Privacy-preserving SVANETs
Privacy-preserving Simple Vehicular Ad-hoc Networks

Jan Hajny, Lukas Malina, Zdenek Martinasek and Vaclav Zeman
Dpt. of Telecommunications, Brno University of Technology, Technicka 12, Brno, Czech Republic

Keywords: Security, Cryptography, Privacy, Digital Signature, Authentication, Protocols.

Abstract: The paper deals with the cryptographic design and experimental implementation of a scheme for (but not limited to) vehicular ad-hoc networks (VANETs). In contrast to existing solutions, our scheme does not need any complex infrastructure (like costly road-side units or special on-board devices) and is based just on users' smart-phones and Internet connection. We call this simplified concept SVANETs (Simple Vehicular Ad-Hoc Networks). In addition, our cryptographic scheme supports drivers' privacy by employing advanced cryptographic constructions like Σ-protocols and proof of knowledge protocols. Our scheme is computationally efficient and practically implementable on current hardware. To prove the efficiency and practical implementability, we provide the first implementation results, which were obtained from our experimental implementation on the Android platform.

1 INTRODUCTION

Vehicular ad-hoc networks (VANETs) provide, so far only theoretically, mechanisms for the communication among cars in daily traffic. By implementing VANETs, it would be possible to share information about traffic accidents, road conditions, traffic density or road closures. Moreover, VANETs would also allow easier monitoring of traffic in cities and on highways. This would improve route planning. Improved traffic monitoring would significantly improve the efficiency, time demands and ecology of traveling.

VANETs allow the communication among cars, which are equipped with special devices called On-Board Units (OBUs). These built-on-purpose devices are wirelessly connected to stationary devices along roads called Road-Side Units (RSUs). Additionally to vehicles with OBUs and RSUs, many existing schemes also use additional third party entities (e.g., registration authorities, revocation authorities, etc.). We consider this concept too complex and impractical for a real-world implementation. It would be too demanding (both financially and logistically) to equip roads with special electronic devices on side (RSUs). Also, it would be very difficult to equip all cars with new, built-on-purpose devices (OBUs). Thus, we propose a new concept called SVANETs[1] in this paper, which needs only smart-phones in participating cars. By simplifying the concept of VANETs, we hope that these communication networks will become more efficient and subsequently more commercially interesting and easier to deploy.

The security of VANETs plays a crucial role in the whole system. First, it is necessary to provide confidentiality and authenticity of messages. In addition to classical security requirements, like the confidentiality and authenticity of messages, the VANETs must also provide new means of privacy protection. Many security problems of existing VANETs are connected to the privacy of users. It must be assured that drivers are not traceable by attackers or by any other entity in the system. The protection of users' privacy plays an important role when the system is about to be deployed commercially and in a large scale. A system which allows the monitoring of drivers and their unwanted tracing would be surely rejected by drivers.

In this paper, we propose a novel cryptographic scheme which is both highly computationally efficient and supporting all security requirements. It provides both the authenticity of messages and the privacy of users. The cryptographic scheme described in this paper is a practical representation of the SVANET concept and uses cryptographic techniques from attribute authentication systems (Hajny and Malina, 2013).

[1] We chose SVANETs name because they provide the same functionality as VANETs, though over Internet.

Hajny J., Malina L., Martinasek Z. and Zeman V..
Privacy-preserving SVANETs - Privacy-preserving Simple Vehicular Ad-hoc Networks.
DOI: 10.5220/0004498802670274
In *Proceedings of the 10th International Conference on Security and Cryptography* (SECRYPT-2013), pages 267-274
ISBN: 978-989-8565-73-0

1.1 Related Work

Many existing VANET schemes (e.g., (Plossl et al., 2006; Raya et al., 2006; Haas et al., 2009)) provide basic security features like message confidentiality, authenticity and non-repudiation by using simple cryptographic methods. Some schemes add the protection of drivers' privacy. It is becoming an important requirement on modern VANETs to provide the prevention against the observing of geographic position of cars and driver's daily routes. In existing VANETs, privacy is usually ensured by two main approaches, i.e., pseudonyms and group signatures.

Privacy preserving solutions based on pseudonyms have been proposed in (Gerlach et al., 2007). The work (Raya and Hubaux, 2007) uses anonymous certificates which are stored in vehicles (usually in a tamper-proof device). This approach uses a set of short-lived pseudonyms and fast changing of these pseudonyms provides the privacy of vehicles. Nevertheless, in dense urban VANETs, this approach is burdened by the preloading and storing of a large number of anonymous certificates with pseudonyms. For low-performance devices like mobile phones, the management of pseudonyms becomes a too complex operation.

The second approach is based on Group Signatures (GS). They provide user anonymity by producing message signatures on behalf of a group in VANETs. Generally, GS guarantee the anonymity of honest users and the traceability of misbehaving users. The group signature scheme called BBS scheme (Boneh et al., 2004) serves as a crucial building block for many security solutions in VANETs (e.g., (Lin et al., 2007) and (Zhang et al., 2008)). Nevertheless, these schemes need several expensive operations like bilinear pairings, modular exponentiations and multiplications during the verification phase and are not appropriate for dense VANETs where tens or hundreds of signatures must be verified in short time. The work (Malina et al., 2012) speeds up multiple group signatures by using short linkability which provides categorized batch verification. Also this solution is based on the BBS scheme and therefore it needs slow bilinear pairing operations.

Based on the analysis of existing schemes, we lack a practical scheme which is able to provide both basic security features (message confidentiality and authenticity) as well as advanced privacy-preserving features (like anonymity, untraceability and unlinkablity of drivers, no trusted third parties, all defined in Section 3.1). Even though some schemes come very close to our requirements, these schemes are currently too complex for smart-phone implementation.

1.2 Our Contribution

In our proposal, we get rid of costly Road-Side Units (RSU) and replace On-Board Units (OBUs) with users' smart-phones. We call the new concept, described in Section 3.2 in detail, Simple VANETs (SVANETs). Our concept supports both the basic security features, such as message confidentiality and authenticity, and advanced privacy-enhancing features. Still, the scheme remains highly practical on mobile devices.

2 PRELIMINARIES

The SVANET scheme proposed in this paper is based on advanced cryptographic constructions. Most constructions are well know and their description can be found in the cited literature. The SVANETs scheme is the practical application of the attribute authentication technology presented in (Hajny and Malina, 2013) where further cryptographic details can be found.

2.1 Assumptions

The Generalized Discrete Logarithm Problem (GDLP) Assumption. Based on (Menezes, 1996) it is assumed that given a finite cyclic group G of order q, a generator g and an element $\beta \in G$ it is hard to find an integer $0 \leq x \leq q-1$ such that $g^x \equiv \beta$.

Factorization Hardness Assumption. Based on (Okamoto and Uchiyama, 1998) it is assumed that it is hard to factor $n = r^2 s$, where r, s are large safe primes.

2.2 Cryptographic Primitives

Discrete Logarithm Commitment Schemes. A cryptographic commitment scheme can be used in scenarios where a user (U) is required to bind to a number without disclosing it. Therefore, there are two properties which must be fulfilled. They are the *hiding property* and the *binding property*. A practical example of using the commitment schemes is the situation where a user generates a random number w and computes a commitment $c = commit(r, w)$ using a randomness r. Then, the user can disclose c to some verifier (V). Although c is disclosed, the verifier is unable to learn w from it (hiding property) and the user is unable to change his w without changing c (binding property).

DL Proof of Knowledge (PK) Protocols. The Proof of Knowledge of Discrete Logarithm (PKDL) protocol can be used by a Prover to give a proof about the knowledge of a discrete logarithm of some public value c with respect to a generator g and modulus p. Using PKDL, the Prover is able to convince V that he knows $w = \log_g c \bmod p$ without actually disclosing it. More information about these protocols can be found in (Cramer, 1996; Quisquater et al., 1989; Cramer et al., 2000; Fiat and Shamir, 1987; Damgård and Fujisaki, 2002; Camenisch and Stadler, 1997; Hajny and Malina, 2013).

DL Proof of Representation Protocols. In cryptographic constructions, it is very common to compute multi-exponentiations in multiplicative groups. The Pedersen commitment (Pedersen, 1992) is the typical example of multi-exponentiation where two generators are used. Using k generators and exponents, the multi-exponentiation can be denoted as

$$c = \prod_{1 \le i \le k} g_i^{w_i} \bmod p, \tag{1}$$

where $(w_1, w_2, ..., w_k)$ is called the representation of c with respect to generators $(g_1, g_2, ..., g_k)$ in mod$p$. With $(g_1, g_2, ..., g_k), q, p, c$ being public, the Verifier might ask the Prover to give a proof of knowledge of $(w_1, w_2, ..., w_k)$. In our proposal, a proof of representation for $k = 2$ is used. PK protocols can be used for proofs of representation too.

Group Signatures from PK Protocols. In the proof of knowledge (resp. proof of representation) protocols introduced above, the Prover always proves the knowledge of some secret value with respect to some public value to the Verifier. In these protocols, the Prover always has to compute a correct answer to some challenge. These protocols can be easily converted to group signature schemes (Fiat and Shamir, 1987). The public value (value c in our examples) can be considered a public key, the secret (value w in our examples) can be considered a private key and the challenge e can be replaced by the *message* (or its hash) being signed. We denote the group signatures constructed using the proof of knowledge (representation) protocols as $SPK(message)$, thus for our examples and *message* we get $SPK\{w : c = g^w\}(message)$ and $SPK\{w_1, w_2 : c = g_1^{w_1} g_2^{w_2}\}(message)$.

Okamoto-Uchiyama Trapdoor One-way Function. Let $n = r^2 s$ and r, s be large primes. Pick $g \in \mathbb{Z}_n$ such that $g \bmod r^2$ is a primitive element of $\mathbb{Z}^*_{r^2}$. Then $c = g^x \bmod n$ is a trapdoor one-way function with r as a trapdoor (Okamoto and Uchiyama, 1998). Value x can be computed using the trapdoor as $x = \dfrac{((c^{r-1} \bmod r^2) - 1)/r}{((g^{r-1} \bmod r^2) - 1)/r} \bmod r$. The function is secure if the factorization of n is hard. Size recommendations for n are the same as for RSA scheme.

2.3 Notation

A Discrete Logarithm (DL) commitment c to a value w is denoted as $c = commit(w)$. For various proofs of knowledge or representation, the efficient notation introduced by Camenisch and Stadler (Camenisch and Stadler, 1997) is used. Thus, a PKDL protocol can be denoted as $PK\{w : c = g^w\}$. The proof of knowledge of representation is denoted as $PK\{w_1, w_2 : c = g_1^{w_1} g_2^{w_2}\}$. The proof of discrete log equivalence with respect to different generators g_1, g_2 is denoted as $PK\{w : c_1 = g_1^w \wedge c_2 = g_2^w\}$. A group signature on *message* is denoted as $SPK\{w : c = g^w\}(message)$. All these protocols can be realized by Σ-protocols (Cramer, 1996). A digital signature using some existing scheme (e.g., RSA) by a user U on some *message* is denoted as $Sig_U(message)$. The symbol ":" means "such that", "$|$" means "divides", "$a||b$" is the concatenation of strings a and b, "$|x|$" is the bitlength of x and "$x \in_R \{0,1\}^l$" is a randomly chosen bitstring of maximum bitlength l. "$x \in_R \mathbb{Z}_q$" denotes a randomly chosen integer less than q. "$\mathbb{Z}_q^*$" denotes an integer multiplicative group modulo q.

3 SCHEME DESCRIPTION

3.1 Requirements

In the Introduction and Related Work sections, we identified the major problems of existing solutions for inter-vehicular communication. In particular, we stated that it is very difficult to provide both authenticity of messages and user anonymity, both in a single efficient VANET scheme. Therefore, our goal is to design a scheme providing both message authenticity and privacy-enhancing features. Namely, we provide following privacy-enhancing features.

- **Anonymity of Users:** in VANETs, a large number of messages is shared among cars. To prevent privacy violation, the messages cannot disclose the identity of drivers.
- **Untraceability of Users:** to protect drivers' privacy, it must be impossible to trace vehicles by intercepting VANET messages.

- **Unlinkability of Signatures:** all valid messages in a VANET are signed to limit fraud messages. The signatures from a particular car cannot be mutually linkable as it would allow unwanted tracing of cars.
- **Revocation:** in case of policy violation, it must be possible to revoke malicious drivers (cars) from the system.

3.2 SVANETs

In our proposal, we get rid of all built-on-purpose devices. By this, we significantly reduce the cost of the vehicular network and make the penetration to the real world much easier. We use only smart-phones and their connection to the Internet through anonymous routing protocols (Reed et al., 1998). The SVANET is based on a mobile phone application which is able to share all traffic information in a privacy-preserving manner. The registration of users as well as the management of the network is done by cloud services. In contrast to classical VANETs or mobile phone traffic applications, we propose a cryptographic protection which prevents all entities (including service administrators and cloud-based management) from breaking the privacy of drivers (i.e., tracing, identification or monitoring). Thus, we do not need any trusted service as many existing proposals do. Our SVANET concept is depicted in Figure 1.

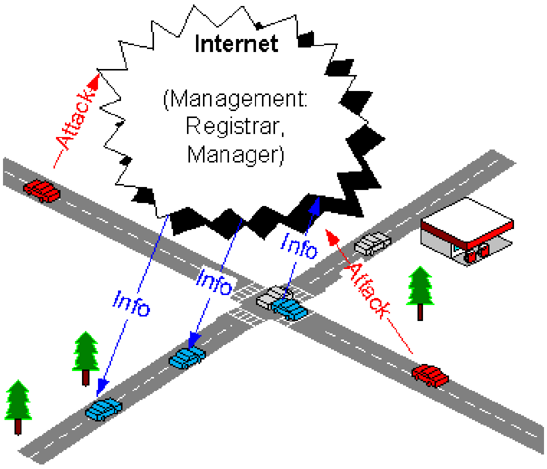


Figure 1: SVANET concept.

In the Figure 1, there are cars equipped with smart-phone devices. These devices are used for sending and receiving traffic information. The information about, for example, road conditions, traffic jams or accidents can be shared. The information is shared among cars using the Internet connection, based on the cellular data communication (GPRS, EDGE, 3G etc.) and the anonymous routing protocols (Reed et al., 1998)[2]. For Android, there is already an implementation called Orbot available, thus we do not deal with anonymous routing in this paper but rather use it as a service. All the management of the network is done using Internet cloud services, thus it is not necessary to use any road-side devices or services. In contrast to existing solutions, we employ strong cryptographic techniques to protect all privacy-sensitive data of users.

3.3 Entities

The SVANET scheme is composed of following entities.

- **Users**: entities which communicate in SVANETs. They can sign messages and verify signatures broadcasted in SVANET using their public and private keys. There can be more roles in the system, e.g., drivers, police, ambulance, firefighters etc. The signature assures that the sender is from a particular authorized group but does not release any other privacy-sensitive information (like user's identity).
- **Registrar**: the entity who registers new users and verifies their qualification for joining the system.
- **Manager**: the entity who is able to manage users (revoke invalid users, attackers, malicious users etc.).

3.4 Protocols

The scheme is composed of 5 cryptographic protocols. They are the `Setup`, `Register`, `Sign`, `Verify` and `Revoke`.

The `Setup` protocol is run by the **Manager** and **Registrar** to establish all system parameters.

The `Register` protocol is run between the **User** and the Registrar with Manager and outputs a keypair for signing/verifying of SVANET messages. During the registration, the User provides his identity and all data required by the Registrar (e.g., ID card, driving licence etc.). This information disclosure does not limit User's privacy since the registration is completely unlinkable to subsequent message signing/verification. Also, the User is assigned a role in the SVANET (general driver, police, ambulance, etc...).

The `Sign` protocol is used for signing messages. By the signature, the User certifies to the authenticity and integrity of the message. Only the group origin is

[2] TOR can add to the complexity of the protocol.

revealed by the signature, User's concrete identity is hidden.

The `Verify` protocol is used for the verification of messages. The authenticity and integrity of messages can be verified by the `Verify` protocol. Only the group origin is disclosed (for example, the verifier learns that the message is coming from police, ambulance etc.), the concrete sender's identity stays hidden. All messages of a single particular driver are mutually unlinkable.

The `Revoke` protocol is used only in cases where some policy violation is detected. In that case, the User's signing key can be revoked. In most severe violation cases, even the identity of senders can be revealed, but only if the Registrar and Manager cooperate. By such a distribution of power, we limit the privacy violation opportunities of single entities.

3.4.1 Cryptographic Specification of Protocols

Setup Protocol. $(params, K_M, K_R) \leftarrow \texttt{Setup}(k,l,m)$ protocol: the goal of the protocol is to generate system parameters *params*, Manager's key K_M and Registrar's key K_R. The protocol inputs security parameters k,l,m (k is the bitlength of the hash function used, l relates to the bitlength of Users' secrets, and m is the verification error parameter). The Registrar generates a group H defined by a large prime modulus p, generators h_1, h_2 of prime order $q : |q| = 2l$ and $q | p-1$. The Manager generates group G for the Okamoto-Uchiyama Trapdoor One-Way Function. G is defined by the modulus $n = r^2 s$ with r,s large primes ($|r| > 360, |r| > 4.5l, |n| \geq 1024$, $r = 2r'+1$, $s = 2s'+1$, r', s' are primes), generator $g_1 \in_R \mathbb{Z}_n^*$ of order $ord(g_1 \bmod r^2) = r(r-1)$ in $\mathbb{Z}_{r^2}^*$ and $ord(g_1) = rr's'$ in $\mathbb{Z}_n^*$. The Manager also randomly chooses its secrets $S_1, S_2, S_3 : |S_1| = 2.5l, |S_2| = l, |S_3^{-1} \bmod \phi(n)| = l$ and $GCD(S_1, \phi(n)) = GCD(S_2, \phi(n)) = GCD(S_3, \phi(n)) = 1$. Finally, Manager computes group public key $G_{PK} = g_1^{S_1} \bmod n$ (public, common for all group members, linked to a specific description, e.g. "general drivers") and values $g_2 = g_1^{S_2} \bmod n, g_3 = g_1^{S_3} \bmod n$. There might be more public group keys (different G_{PK_i}'s and S_{1_i}'s) related to different groups of users in the SVANET. In that case, each unique G_{PK_i} represents one group[3]. In the rest of the paper, we consider for simplicity only one group with one public key G_{PK}.

The values $q, p, h_1, h_2, n, g_1, g_2, g_3, G_{PK}$ are made public as system parameters *params*, while r, s, S_1, S_2, S_3 are securely stored by the Manager as K_M key. Additionally, we use a traditional digital signature scheme (e.g., RSA). Registrars and Users are equipped with a private/public key-pair for digital signatures. This can be accomplished by existing techniques for PKI. The Registrar's private key is denoted as K_R.

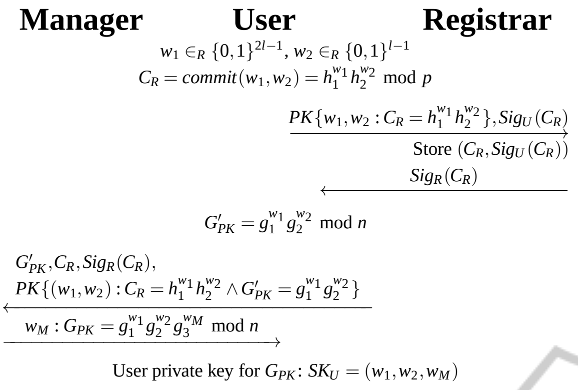
Register Protocol. $SK_U \leftarrow \texttt{Register}(params, K_R, K_M)$ protocol: the first part of the `Register` protocol runs between User's smart-phone and the Registrar. The communication is not anonymous here, thus the Registrar can physically check the identity of the User, his licence etc. Then, User's smart-phone generates User's contribution to his private key (w_1, w_2) and commits to these values. The commitment C_R is digitally signed[4] by the User and sent with an appropriate construction correctness proof $PK\{w_1, w_2 : C_R = h_1^{w_1} h_2^{w_2}\}$ to the Registrar. The Registrar checks the proof, the signature and replies with his digital signature on the commitment. In this phase, the User generated and committed to his private key contribution. It will be used in all his future signatures. The Registrar approved a new User by signing the committed key contribution.

The second part of the protocol runs between the User's smart-phone and the Manager. In this phase, the Manager checks the signature of the Registrar on User's commitment C_R and computes his contribution w_M to User's private key such that $G_{PK} = g_1^{w_1} g_2^{w_2} g_3^{w_M} \bmod n$ holds. As a result, the User's smart-phone learns all parts of the private key SK_U, namely User's part (w_1, w_2) and M's part w_M. This triplet forms the discrete logarithm representation of the group public key G_{PK} such that $G_{PK} = g_1^{w_1} g_2^{w_2} g_3^{w_M} \bmod n$. This representation can be computed only in cooperation with M (who knows the factorization of n). Although G_{PK} is shared among all group members, the private key (w_1, w_2, w_M) is unique for each user, since (w_1, w_2) is randomly generated by each User's smart-phone and w_M is generated by M. Due to the discrete logarithm assumption, Users are stuck to their keys and they are unable to compute other valid keys without knowing K_M. The private key SK_U never leaves the smart-phone and is stored in phone's hardware-protected memory. All operations involving (w_1, w_2, w_M) are computed in the phone. The `Register` protocol is depicted in Figure 2.

Sign Protocol. $signature \leftarrow \texttt{Sign}(params, SK_U, message)$ protocol: the protocol is used by User's smart-phone to construct a signature on message *message*. In the protocol, the User proves the

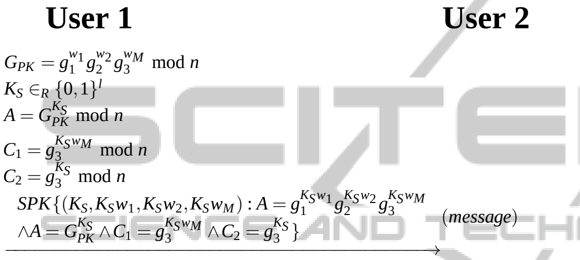
[3] A public list of groups and their assigned keys G_{PK_i}'s is maintained by the Manager.

[4] Here, we rely on already established PKI, e.g., RSA signatures.



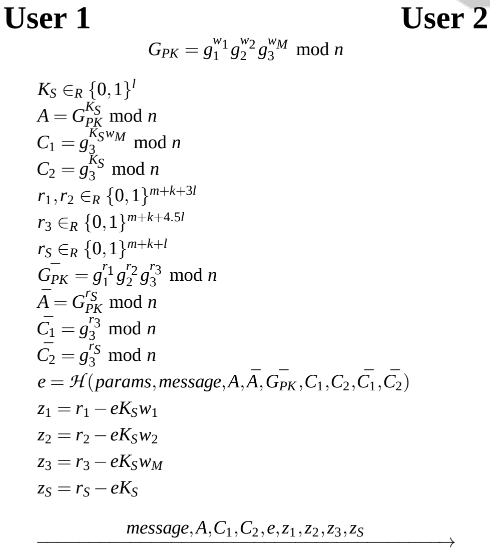
$$w_1 \in_R \{0,1\}^{2l-1},\ w_2 \in_R \{0,1\}^{l-1}$$
$$C_R = commit(w_1,w_2) = h_1^{w_1} h_2^{w_2} \mod p$$
$$\xrightarrow{PK\{w_1,w_2 : C_R = h_1^{w_1} h_2^{w_2}\}, Sig_U(C_R)}$$
$$\text{Store } (C_R, Sig_U(C_R))$$
$$\xleftarrow{Sig_R(C_R)}$$
$$G'_{PK} = g_1^{w_1} g_2^{w_2} \mod n$$
$$\xleftarrow{G'_{PK}, C_R, Sig_R(C_R),\ PK\{(w_1,w_2) : C_R = h_1^{w_1} h_2^{w_2} \wedge G'_{PK} = g_1^{w_1} g_2^{w_2}\}}$$
$$\xrightarrow{w_M : G_{PK} = g_1^{w_1} g_2^{w_2} g_3^{w_M} \mod n}$$
User private key for G_{PK}: $SK_U = (w_1, w_2, w_M)$

Figure 2: Register Protocol.



$$G_{PK} = g_1^{w_1} g_2^{w_2} g_3^{w_M} \mod n$$
$$K_S \in_R \{0,1\}^l$$
$$A = G_{PK}^{K_S} \mod n$$
$$C_1 = g_3^{K_S w_M} \mod n$$
$$C_2 = g_3^{K_S} \mod n$$
$$\xrightarrow{SPK\{(K_S, K_S w_1, K_S w_2, K_S w_M) : A = g_1^{K_S w_1} g_2^{K_S w_2} g_3^{K_S w_M} \wedge A = G_{PK}^{K_S} \wedge C_1 = g_3^{K_S w_M} \wedge C_2 = g_3^{K_S}\}(message)}$$

Figure 3: Sign Protocol in Camenisch-Stadler Notation.



$$G_{PK} = g_1^{w_1} g_2^{w_2} g_3^{w_M} \mod n$$
$$K_S \in_R \{0,1\}^l$$
$$A = G_{PK}^{K_S} \mod n$$
$$C_1 = g_3^{K_S w_M} \mod n$$
$$C_2 = g_3^{K_S} \mod n$$
$$r_1, r_2 \in_R \{0,1\}^{m+k+3l}$$
$$r_3 \in_R \{0,1\}^{m+k+4.5l}$$
$$r_S \in_R \{0,1\}^{m+k+l}$$
$$\overline{G_{PK}} = g_1^{r_1} g_2^{r_2} g_3^{r_3} \mod n$$
$$\bar{A} = G_{PK}^{r_S} \mod n$$
$$\bar{C_1} = g_3^{r_3} \mod n$$
$$\bar{C_2} = g_3^{r_S} \mod n$$
$$e = \mathcal{H}(params, message, A, \bar{A}, \overline{G_{PK}}, C_1, C_2, \bar{C_1}, \bar{C_2})$$
$$z_1 = r_1 - eK_S w_1$$
$$z_2 = r_2 - eK_S w_2$$
$$z_3 = r_3 - eK_S w_M$$
$$z_S = r_S - eK_S$$
$$\xrightarrow{message, A, C_1, C_2, e, z_1, z_2, z_3, z_S}$$

Figure 4: Sign Protocol in detail.

knowledge of his private key $SK_U = (w_1, w_2, w_M)$. The session is randomized by a session key K_S. User creates a commitment C_2 to the session key K_S and proves its correctness. The protocol transcript forms the *signature*. The protocol is illustrated in Figure 3 in CS notation. The Sign protocol follows the standard mechanisms specified in Section 2.2, thus can be implemented as shown in Figure 4.

Verify Protocol. The Verify protocol is the verification of the *SPK* protocol transcript generated by the Sign protocol. Following the standard verification equations (see paper (Hajny and Malina, 2013) for more cryptographic details), the Verify protocol consists of equations (2-6) which are evaluated by the recipient. The last equation (7) checks whether the User is revoked or not.

$$\overline{G_{PK}} = A^e g_1^{z_1} g_2^{z_2} g_3^{z_3} \mod n \quad (2)$$
$$\bar{A} = A^e G_{PK}^{z_S} \mod n \quad (3)$$
$$\bar{C_1} = C_1^e g_3^{z_3} \mod n \quad (4)$$
$$\bar{C_2} = C_2^e g_3^{z_S} \mod n \quad (5)$$
$$e \overset{?}{=} \mathcal{H}(params, message, A, \bar{A}, \overline{G_{PK}}, C_1, C_2, \bar{C_1}, \bar{C_2}) \quad (6)$$
$$C_1 \overset{?}{\not\equiv} C_2^{rev} \mod n \quad (7)$$

Revoke Protocol. $rev \leftarrow$Revoke$(params, signature, K_M, K_R)$ protocol: the protocol is only executed if a User needs to be revoked from the system or if the Manager wants to reveal malicious users (and has a strong evidence for doing so). The transcript of the Sign protocol can be forwarded to the Manager in case of policy violations. The Manager can decide about the type of revocation. User revocation or identification are available.

User Revocation. The Manager knows the factorization of n thus he knows the trapdoor to the Okamoto-Uchiyama trapdoor function. From C_2, he learns the session key K_S and from C_1, his contribution w_M to the User private key SK_U. The Manager can publish revocation information $rev = w_M$ on a public blacklist of revoked keys. Then, each User is able to check if the keys used for message signing are blacklisted or not by checking $C_1 \overset{?}{\equiv} C_2^{rev} \mod n$. The equation holds only for revoked Users. In dense urban areas, the blacklist has to be periodically reset to limit its size. In that case, the user keys have a temporal validity only. Using this type of revocation, no identity is revealed and no valid users have to update their keys. The revocation does not influence non-revoked users in any sense. Users only need to periodically download the blacklist with short *rev* values. Also Registrars can initiate the revocation, by sending C_R to the Manager who is able to link C_R to w_M. The revocation information *rev* is then published by the Manager in the same way as if revocation is initiated by Users.

Identification. Sometimes, it is necessary to identify malicious users. In that case, the Manager reveals w_M and finds corresponding C_R since both values are linked by the Register protocol. C_R is then forwarded to the Registrar who can de-anonymize the User since he has a database of digitally signed C_R's.

The identification is non-repudiable since C_R is digitally signed and perfectly binds the User to the key inside.

4 SECURITY ANALYSIS

We provide the security analysis of the proposed SVANET scheme in this section. The scheme is built using well-established cryptographic protocols. These protocols, defined in Section 2.2, provide provable security features. Since all protocols are from the Σ-protocol family (Cramer, 1996), they provide following features (the proofs of features can be found in (Cramer et al., 2000)).

- **Completeness**: honest users who know the private keys are always accepted by the verification protocol.[5]
- **Soundness**: dishonest users who do not know private keys are always rejected by the verification protocol.[6]
- **Zero-knowledge**: all protocols have the Zero-Knowledge property which mathematically proves that no secret information about user private keys is released by the signatures. [7]

Using the above specified features of Σ-protocols, it is straightforward, that group signatures can be constructed only by honest group members who know private keys. Now, we prove privacy-enhancing features stated in Section 3.1.

- **Anonymity of Users:** by having the ZK property, it is possible to prove that the signatures release no information except that they belong to some group member. Thus, signer's identity is not released.
- **Untraceability of Users:** no entity (including Registrar and Manager) can trace a particular user since they cannot de-anonymize signatures alone due to the DL assumption. Furthermore, they cannot link the Register and Sign protocols without cooperating.
- **Unlinkability of Signatures:** all signatures are randomized by a secret value K_S, thus all signatures are mutually unlinkable.
- **Revocation:** in case of policy violation, the Registrar and the Manager can join their secret information to identify signature owners. No entity can misuse its secret knowledge to de-anonymize users alone.

5 IMPLEMENTATION RESULTS

Recently, many VANET schemes with privacy-enhancing features were proposed. Unfortunately, many solutions are based on cryptographic primitives which require bilinear pairing operations. These operations are much more demanding than classical operations (more than 100 times slower, based on (Caro, 2012)). The computational demands are the main reasons why existing proposals remain theoretical only. Currently, it is unfeasible to implement schemes which use bilinear pairing on mobile devices such as smart-cards or mobile phones (Caro, 2012). The signing operation would take seconds which is unacceptable for real-life applications.

The SVANET scheme proposed in this paper is based on plain modular arithmetic. The signing algorithm needs only 8 modular exponentiations, 6 modular multiplications and 4 subtractions. Since modular operations are fast on current hardware (usually provided by dedicated libraries or default APIs), we consider the signing phase to be very efficient. The complexity of a signature verification is very similar - 8 modular exponentiations are needed if nobody is revoked. Each revoked vehicle adds to the complexity by 1 exponentiation. Our next goal is to limit this linear dependency by using batch verification techniques (Malina et al., 2012).

We implemented all the required operations on two Android devices. One represents an older mobile phone (Samsung Galaxy S i9000) and the second represents a new smart-phone (Samsung Galaxy Nexus I9250M). The time of signature generation using our scheme and 1024 b group is presented in Table 1 in milliseconds. The numbers represent the average of 100 measurements. Based on these results, the time of signing on an average smart-phone is under 100 ms which we consider very practical.

6 CONCLUSIONS

In this paper, we proposed a novel concept called SVANETs. This concept makes inter-vehicular communication efficient and affordable because it eliminates all the costly hardware devices, replacing them

[5] This is proven by the design of the protocol - honest users can always construct correct responses z_1, z_2, z_3, z_S during the Sign protocol.

[6] This is proven by the existence of the knowledge extractor, see (Cramer et al., 2000) for more information.

[7] This is proven by the existence of the Zero-Knowledge simulator, see (Cramer et al., 2000) for more information.

Table 1: Signing Performance in Milliseconds.

Operation	Samsung Galaxy S1	Nexus i9250
Random Num. Gen. (160b)	0,05	0,04
Random Num. Gen. (560b)	0,12	0,08
Hash SHA1	0,11	0,02
Modular Power (160b)	6,13	4,30
Modular Power	14,83	9,69
Modular Multiplication	0,16	0,14
Multiplication	0,03	0,03
Subtraction	0,01	0,02
Total	**102,38 ms**	**67,58ms**

by drivers' smart-phones. In addition, we propose a new cryptographic scheme which makes these SVANETs both secure and privacy-friendly. With the proposed cryptographic scheme, it is possible to retain both authenticity of messages and anonymity of drivers. The proposed scheme allows smart-phones to send digitally signed messages on behalf of a particular group of drivers. Our implementation results show that the scheme is highly practical and implementable on today's smart-phones.

ACKNOWLEDGEMENTS

This research work is funded by projects SIX CZ.1.05/2.1.00/03.007; the Technology Agency of the Czech Republic projects TA02011260 and TA03010818; the Ministry of Industry and Trade of the Czech Republic project FR-TI4/647.

REFERENCES

Boneh, D., Boyen, X., and Shacham, H. (2004). Short group signatures. In *Proc. Adv. Cryptology-Crypto 04, ser. LNCS 3152*, pages 41–55. Springer-Verlag.

Camenisch, J. and Stadler, M. (1997). Proof systems for general statements about discrete logarithms. Technical report.

Caro, A. D. (2012). The java pairing based cryptography library (jpbc): Benchmark. http://gas.dia.unisa.it/projects/jpbc/benchmark.html#testbed3.

Cramer, R. (1996). *Modular Design of Secure, yet Practical Cryptographic Protocols*. PhD thesis, University of Amsterdam.

Cramer, R., Damgrd, I., and MacKenzie, P. (2000). Efficient zero-knowledge proofs of knowledge without intractability assumptions. In *Public Key Cryptography*, volume 1751 of *Lecture Notes in Computer Science*, pages 354–372. Springer Berlin Heidelberg.

Damgård, I. and Fujisaki, E. (2002). A statistically-hiding integer commitment scheme based on groups with hidden order. In *Proceedings of the 8th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology*, ASIACRYPT '02, pages 125–142, London, UK. Springer-Verlag.

Fiat, A. and Shamir, A. (1987). How to prove yourself: Practical solutions to identification and signature problems. In Odlyzko, A., editor, *Advances in Cryptology - CRYPTO 86*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194. Springer Berlin / Heidelberg.

Gerlach, M., Festag, A., Leinmuller, T., Goldacker, G., and Harsch, C. (2007). Security architecture for vehicular communication. In *The 5th International Workshop On Intelligent Transportation*.

Haas, J., Hu, Y.-C., and Laberteaux, K. (2009). Real-world VANET security protocol performance. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1 –7.

Hajny, J. and Malina, L. (2013). Unlinkable attribute-based credentials with practical revocation on smart-cards. In *Proceedings of the 11th international conference on Smart Card Research and Advanced Applications*, CARDIS'12, pages 62–76, Berlin, Heidelberg. Springer-Verlag.

Lin, X., Sun, X., han Ho, P., and Shen, X. (2007). Gsis: A secure and privacy preserving protocol for vehicular communications. In *IEEE Transactions on Vehicular Technology*, volume 56, pages 3442–3456.

Malina, L., Castella-Roca, J., A., V.-G., and Hajny, J. (2012). Short-term linkable group signatures with categorized batch verification. In *the FPS*, pages 1 –17.

Menezes, A. J. (1996). *Handbook of Applied Cryptography*. CRC Press.

Okamoto, T. and Uchiyama, S. (1998). A new public-key cryptosystem as secure as factoring. In *Advances in Cryptology - EUROCRYPT 98*, volume 1403 of *Lecture Notes in Computer Science*, pages 308–318. Springer Berlin / Heidelberg.

Pedersen, T. P. (1992). Non-interactive and information-theoretic secure verifiable secret sharing. In *Proceedings of the 11th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '91, pages 129–140, London, UK, UK. Springer-Verlag.

Plossl, K., Nowey, T., and Mletzko, C. (2006). Towards a security architecture for vehicular ad hoc networks. In *Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on*, page 8.

Quisquater, J.-J., Guillou, L., Annick, M., and Berson, T. (1989). How to explain zero-knowledge protocols to your children. In *Proceedings on Advances in cryptology*, CRYPTO '89, pages 628–631, New York, NY, USA. Springer-Verlag New York, Inc.

Raya, M. and Hubaux, J.-P. (2007). Securing vehicular ad hoc networks. *J. Comput. Secur.*, 15:39–68.

Raya, M., Papadimitratos, P., and Hubaux, J.-P. (2006). Securing vehicular communications. *Wireless Communications, IEEE*, 13(5):8 –15.

Reed, M., Syverson, P., and Goldschlag, D. (1998). Anonymous connections and onion routing. *IEEE Journal on Selected Areas in Communications*, 16(4):482–494.

Zhang, C., Lu, R., Lin, X., Ho, P.-H., and Shen, X. (2008). An efficient identity-based batch verification scheme for vehicular sensor networks. In *INFOCOM*, pages 246–250. IEEE.