2009

Noninteractive, Anonymously Authenticated and Traceable Message Transmission for VANETs

Maged Ibrahim, *Helwan University*

Hindawi Publishing Corporation
International Journal of Vehicular Technology
Volume 2009, Article ID 702346, 14 pages
doi:10.1155/2009/702346

Research Article

Noninteractive, Anonymously Authenticated, and Traceable Message Transmission for VANETs

Maged Hamada Ibrahim

Department of Electronics, Communications and Computers, Faculty of Engineering, Helwan University, 1 Sherif Street, 11792 Helwan, Cairo, Egypt

Correspondence should be addressed to Maged Hamada Ibrahim, mhii72@hotmail.com

Received 15 July 2009; Accepted 24 November 2009

Recommended by Kui Wu

Unlike sensor and other ad-hoc wireless networks, vehicular ad-hoc networks (VANETs) are characterized by its high mobility which allows a very short communication interval among onboard units (OBUs) and between an OBU and road-side units (RSUs). This major characterization motivates the design of communication protocols that are noninteractive or at least require a very limited number of rounds between units. The challenging issue is that such protocols must satisfy a number of security services that could be complex by their nature. In secure VANETs protocols, anonymity and traceability are two important services, yet, achieving a satisfactory security level for both of them—with acceptable complexity—is not an easy task due to the contradicting requirements: anonymous transmission must not be traceable by any individual while if a transmission is traceable, then anonymity is threatened. Existing secure VANETs protocols for anonymous and traceable transmissions either, provide unconditional anonymity where traceability and revocation are impossible, or grant trust to a thirdparty not to reveal the identity of a unit unless there is a legal reason. In this paper, we propose the first secure VANET protocol that allows authenticated transmission among OBUs and RSUs and at the same time enjoys the following properties. (i) The transmission among OBUs and RSUs is noninteractive (i.e., a one-move transmission without any interactive setup requirements). (ii) The authenticated transmission between any pair of units is anonymous (i.e., no single authority knows any information about the identity of the communicating OBU). (iii) In serious road crimes (e.g., hit-and-run, road rage, etc.) and under court order, an OBU could be traced to its clear identity. We also show how our protocol could be used to setup a confidential session between any pair of units without relying on extensive number of interactive rounds.

1. Introduction

Advances in wireless technology have imposed a major impact on the revolution of human's lifestyle by providing a convenient and flexible way to access the internet services and various types of personal communication applications. Recently, technologies built on IEEE 802.11 and IEEE 1609 standards, 5.9 GHz Dedicated Short Range Communications (DSRC) protocols [1–5], are proposed to support advanced vehicle safety applications through effective, reliable, and secure vehicle-to-vehicle (V2V) (also known as Inter-Vehicle Communication (IVC)) and vehicle-to-Infrastructure (V2I) communications, which are also known as Vehicle Safety Communications (VSC) technologies. U.S. Department of Transportation (USDOT) works with seven automotive manufacturers—BMW, DaimlerChrysler, Ford, GM, Nissan, Toyota, and VW—to form the Vehicle Safety Communications Consortium (VSCC) to establish the VSC project to evaluate vehicle safety applications enabled or enhanced by external vehicle communications. For example, if a possible red light violation is detected at an intersection, the potential violator will receive a warning to slow down to avoid unintentional red light violations. Meanwhile, a warning on the running red light event will be given to the other drivers at the intersection thereby minimizing the possibility of collision.

1.1. Vehicular Ad-Hoc Networks. Nowadays, car manufactories and telecommunication industries have been gearing up

to equip each car with the technology that allows vehicles to communicate with each other as well as with a roadside infrastructure that may be located in some critical sections of the road, such as at every traffic light or any intersection or stop sign, in order to improve the driving experience and make driving safer. For example, Microsoft Corp.'s MSN TV and KVH [6] Industries Inc. have introduced an automotive vehicle Internet access system called TracNet [7], which can bring the Internet service to any in-car video screens. It also turns the entire vehicle into an IEEE 802.11-based WiFi hotspot, so passengers can use their wireless-enabled laptops to go online like they are home or in the office. Furthermore, by using those equipped communication devices, also known as On-Board Units (OBUs), vehicles can communicate with each other as well as with the Roadside units (RSUs) located in the critical points of the road. A self-organized network can be formed by connecting the vehicles and RSUs, which is called Vehicular Ad Hoc Network (VANET), and the RSUs are further connected to the backbone network via the high-speed network connections. An increasing interest has been raised recently on the applications through V2V and V2I communications, aiming to improve driving safety and traffic management while providing drivers and passengers with Internet access. It is estimated that the market for vehicular communications will reach to multibillions dollars by 2012.

1.2. Efficiency and Security Issues in VANETs. In a vehicular network, drivers are bound to a single identity to prevent spoofing attacks. For instance, in the congestion avoidance scheme, we would like to prevent one vehicle from claiming to be hundreds in order to create the illusion of a congested road. Strong authentication also provides valuable forensic evidence and allows us to use external mechanisms, such as traditional law enforcement, to deter or prevent attacks on vehicular networks. However, drivers value their privacy and are unlikely to adopt systems that require them to abandon their anonymity. For example, if we try to prevent spoofing in a manner that reveals each vehicle's permanent identity, then we may violate drivers' privacy expectations. Balancing anonymity concerns with traceability needs represents a challenging problem when designing a security protocol. Both of these services are important and it is insecure to design a protocol that allows one service without the other since, if anonymity is a right for a citizen, traceability is a right for the law.

Vehicular networks require near real-time responses as well as hard real-time guarantees. While some applications may tolerate some margin in their response times, they will all typically require faster responses than those expected in traditional sensor networks, or even ad hoc networks. However, attempts to meet real-time demands typically make applications vulnerable to Denial of Service (DoS) attacks. For example, on open speed roads, a delay of even seconds can render the message meaningless and could be disastrous.

Current plans for vehicular networks rely on the emerging standard for dedicated short-range communications (DSRC), based on an extension to the IEEE 802.11 technology. Yin et al. provide a detailed, low-level evaluation of the performance of a simulated DSRC network and find that while the current DSRC standard provides an acceptable latency, the reliability is still lacking [8]. According to their simulations, on average, only 50%–60% of a vehicle's neighbors will receive a broadcast message. *Since vehicles moving with high speed will remain within communications range for only a few seconds, opportunities to retry a broadcast will be limited.* On a positive note, DSRC features a high data rate.

Many applications use protocols that rely on probabilistic schemes to provide security. However, given the critical life-threatening nature of many proposed vehicular applications, even a small probability of error will be unacceptable. Applications should rely on deterministic schemes or probabilistic schemes with security parameters large enough to make the probability of failure infinitesimally small. Furthermore, for many applications, security must focus on prevention of attacks, rather than detection and recovery. In an ad hoc network, it may suffice to detect an attack and alert the user, leaving recovery and clean-up to the humans. However, in many safety-related vehicular network applications, detection will be insufficient, since by the time the driver can react, the warning may be too late. Instead, security must focus on preventing attacks in the first place, which will require extensive foresight into the types of attacks likely to occur.

Traditional sensor networks frequently assume a relatively static network, and even ad hoc networks typically assume limited mobility, often focusing on hand-held PDAs and laptops carried by users. For vehicular networks, mobility is the norm, and it will be measured in miles, not meters, per hour. Also, the mobility patterns of vehicles on the same road will exhibit strong correlations. Each vehicle will have a constantly shifting set of neighbors, many of whom it has never interacted with before and is unlikely to interact with again. The transitory nature of interactions in a vehicular network will restrict the utility of reputation-based schemes. For example, rating other vehicles based on the reliability of their congestion reports is unlikely to prove useful, a specific driver is unlikely to receive multiple reports from the same vehicle. Furthermore, *since vehicles may only be within communication range for a matter of seconds, we cannot rely on protocols that require significant interaction between the sender and receiver.*

2. Previous and Related Work

In the VII system [3, 4, 9, 10], each vehicle is equipped with an OBU which integrates the technologies of wireless communications, micro-sensors, embedded systems, and GPS on vehicles. With the help of the OBUs, vehicles are able to communicate with the RSUs on the road sides, which by their role are connected to the internet. OBUs, RSUs, and the network backbone form an infrastructure integration called VII system.

Since OBUs and RSUs need to authenticate each other, several contributions have been given to satisfy this service in the proper way. However, most of these contributions [11–20] rely on a policy that grants trust to the RSU or the security center (also called authentication server) in the network not to trace the identity of a particular OBU (vehicle or driver) without a legal reason. Such policy threatens anonymity and may prevent drivers from joining the service. On the other hand, the recent work of [21] provides unconditional anonymity in the sense that it is impossible to trace the identity of any OBU. The protocol requires four interactive rounds between the OBU and the RSU in order to establish a session key which will be used for all data transmission during the session. The protocol employs the idea of *verifiable common encoding* to allow the OBU to authenticate an RSU and vice versa which requires a large number of public key encryptions/decryptions almost equal to the number of registered users and hence suffers from high computations and communications complexities. Since anonymity is unconditional, revocation of a particular identity is also impossible. However, the authors in [21] introduced the elegant idea of adaptive anonymity as a complexity-anonymity tradeoff.

In our system we will avoid the group signature schemes that rely on the existence of a group authority since, in this case, all users in the group must trust this authority, yet, our work in this paper is inspired by the recent advances in a different version of group signature schemes known as Democratic Group Signatures [22–24] where there is no group authority managing the users in the group.

Democratic group signatures (DGS) [22] eliminates the role of a group manager by (i) allowing the group members themselves to initialize and setup the group, (ii) controlling it over dynamic changes in a collective manner, and (iii) distributing traceability rights to each group individual. In this case, every group member has the individual right to trace and disclose the identity of the signer and hence, anonymity is provided against non-members. The model in [22] requires unlinkability of signatures, that is, the signature verifier cannot distinguish signatures issued by the same group member without this member being traced and disclosed. DGS schemes differ from threshold signature schemes (e.g., [25–30]) in the sense that, in DGS each group member is granted the right to generate a signature on a given message individually; a non-member verifier recognizes the signature as anonymously generated by the group. On the other hand, in threshold signatures, the signature on a given message is generated by the majority of the group (exceeding a certain threshold), yet, no coalition of minority (less than or equals the threshold) can generate the signature. Linkable democratic group signatures (LDGS) [23] realize linkability of signatures issued by the group members in a way that preserves the anonymity of the signer. More precisely, a non-member verifier is able to distinguish signatures issued by the same signer for future reference without being able to trace the identity of this signer. To achieve this property, LDGS actually employs the idea of pseudonym systems. In this scenario, each group member (in addition to his unique identity) will be assigned a unique pseudonym. Given a certain group member, all signed messages generated by this particular member will carry his unique pseudonym. A non-member verifier is able to link signed messages of the same signer via his pseudonym, yet, the verifier gains no information about the signer's identity from this pseudonym. On the other hand, each group member knows the secret tracing trapdoor parameter, by which, he is able to extract the identity of the signer from the signer's unique pseudonym.

Onion routing is a technique for anonymous communication over a computer network. Messages are repeatedly encrypted and then sent through several network nodes called onion routers. Each onion router removes a layer of encryption to uncover routing instructions, and sends the message to the next router where this is repeated [31, 32]. This prevents these intermediary nodes from knowing the origin, destination, and contents of the message. The idea of onion routing (OR) is to protect the privacy of the sender and recipient of a message, while also providing protection for message content as it traverses a network. Onion routing accomplishes this according to the principle of Chaum's mix cascades (also known as MixNets) [33]: Messages travel from source to destination via a sequence of proxies ("onion routers"), which reroute messages in an unpredictable path. To prevent an adversary from eavesdropping on message content, messages are encrypted between routers. The advantage of onion routing (and mix cascades in general) is that it is not necessary to trust each cooperating router; if one or more routers are compromised, anonymous communication can still be achieved. This is because each router in an OR network accepts messages, re-encrypts them, and transmits to another onion router. An attacker with the ability to monitor every onion router in a network might be able to trace the path of a message through the network, but an attacker with more limited capabilities will have difficulty even if he controls one or more onion routers on the message's path. Onion routing does not provide perfect sender or receiver anonymity against all possible eavesdroppers that is, it is possible for a local eavesdropper to observe that an individual has sent or received a message. It does provide for a strong degree of unlinkability, in the notion that an eavesdropper cannot easily determine both the sender and receiver of a given message. Even within these confines, onion routing does not provide any absolute guarantee of privacy; rather, it provides a continuum in which the degree of privacy is generally a function of the number of participating routers versus the number of compromised or malicious routers. Onion routing provides protection (anonymity and privacy) against intermediate nodes (routers) but does not provide source and destination anonymity and hence is not suitable for VANET systems, also notice that there is no intermediate nodes between an OBU and an RSU.

Ring signatures introduced in [34] and further studied in [35, 36] do not require any group manager to form a group. For signature generation, every user builds a set of public keys that includes his public key and the public keys of other users. A generated signature does not reveal the public key of the signer, but a set of public keys of all possible

signers. Therefore, ring signatures cannot be used for a direct communication between a verifier and a signer. Additionally, ring signatures provide unconditional anonymity, that is, no party can reveal the signer identity. Although the LDGS introduced in [23] adds nice properties to DGS, it suffers from a serious weakness: the power to trace and disclose the identity of the signer is in the hands of each member of the group. Such individual traceability violates the anonymity attribute of group signatures, since, in practice, it maybe possible to guarantee the honesty of one authority (group manager) as in GS but it is difficult and almost impossible to guarantee the full honesty of all the group members as in LDGS. Malicious behavior of at least one of the group members violates anonymity and consequently, destroys the main merit of group signatures. Moreover, in LDGS, such traitors are undetectable.

The recent contribution in [24] improves the security of the LDGS introduced in [23] to resist traitors members within the group. The tracing trapdoor parameter must not be known to individuals, yet, it could be shared on a threshold basis such that, the identity-pseudonym link of any member is kept secret from the members unless there exists a legal reason (e.g., a dispute). In case of a dispute, the majority of the members join together to recover the identity of the signer (related to a given pseudonym). The protocol is robust against possible malicious behavior of the traitors. The work assumes that there exist minority traitors (at most 1/4 of the members). However, this bound can be improved to 1/3 but with high computation and communication complexities. Our construction in this paper is based on the traitors resistant TR-LDGS introduced in [24].

3. Motivations and Contributions

Motivations. We argue that more efforts must be made to balance the issue of anonymity and traceability in a way that gives drivers some confidence in the system they deal with. Existing protocols do not provide the appropriate balance between anonymity and traceability in the sense that they either rely on granting trust to a single authority, or providing unconditional anonymity where traceability and revocation are impossible. Due to the high mobility of VANETs and hence, the short time available for the communication between an OBU and an RSU, it is preferred to design protocols that allow transmission in a noninteractive way (one move transmission). Protocols that require many rounds for the purpose of setting up a secure link between an OBU and an RSU must be avoided.

Contributions. Our contribution in this paper is to construct a protocol for secure data transmission in VANETs. We mainly focus on secure OBU-RSU communication. Yet, at the end of the paper, we show how to realize a direct OBU-OBU communication, that is, any two OBUs may communicate directly and securely given any existing RSU in range. In our design we will try our best to make the transmission noninteractive aiming to minimize the time required to transmit a message between an RSU and an OBU. We are able to design a completely noninteractive protocol for authenticated data transmission. For realizing confidentiality service, the transmission from the OBU to the RSU is still noninteractive, however, in order for the RSU to send a first time confidential message to a particular OBU, it requires the OBU to anonymously announce its presence. We balance the contradiction between anonymity and traceability by employing a set of n tracing authorities that share a tracing trapdoor parameter k among themselves on a threshold basis. Each OBU will communicate using a blinded version of its identity. In order to trace a particular identity, the majority of these authorities must collaborate together to extract the identity from its blinded version. In this context, drivers do not need to put any trust in any of the authorities, yet, their trust is distributed among the n authorities. There is also a security center whose purpose is the registration of the identity of the OBU and then to send this registered identity to all RSUs.

4. System Attributes and Outlines

In this section we discuss the merits and the outlines of our proposed protocol.

4.1. Security Attributes. Our protocol allows an OBU unit to noninteractively transmit messages to an RSU and vice versa with the following efficiency and security requirements:

(i) *Authentication.* The RSU is confident that the received message is originated from some OBU in a vehicle that was previously registered with the vehicle security center. On the other hand, the OBU is confident that the received message is originated from some authorized RSU unit.

(ii) *Anonymity.* From a received message, nobody gains any information about the identity of any particular vehicle. Notice that only OBU anonymity is considered. Our anonymity assumption holds as long as at most t authorities are malicious.

(iii) *Traceability.* In case of accidents or any severe problems on the road (e.g., hit and run) under court order, the majority of the tracing authorities (with at most t authorities may maliciously behave or even refuse to cooperate) are able to jointly trace the identity of a particular vehicle (OBU). Moreover, the authorities are able to trace the identities of all vehicles passing by a certain area (a particular RSU) in a certain period of time (the estimated time interval within which the accident took place).

(iv) *Linkability.* It realizes linkability of signatures issued by the OBUs in a way that preserves their anonymity. More precisely, an RSU is able to distinguish signatures issued by the same OBU for future reference without being able to trace the identity of this OBU.

(v) *Nonrepudiation.* In case of accidents and under court order, when a vehicle identity is traced and revealed by the authorities from a given blinded-identity

associated with a given message, it is not possible to repudiate the accusation.

(vi) *Revocation.* It must be easy for the security center to revoke the identity of any OBU and to prevent it from participating in the communication with any RSU.

(vii) *Robustness.* Since we assume that the minority of the tracing authorities could behave maliciously, the protocol should be robust against any malicious behavior during the setup and the tracing protocol.

4.2. Assumptions, Model, and Outlines. According to the system global parameters, we assume that each OBU comes with the unique pair (x, id) installed where x is a private key, $\mathrm{id} = f(x)$, and f is a suitable one-way function. The OBU could be queried to reveal the identity id while the private key x never leaves the OBU. We also assume that any OBU could behave maliciously. In our work, we assume that an RSU is authorized to create sensitive messages for the safety of the vehicles on te road and hence, we cannot runaway of assuming that any RSU is honest-but-curious, since a malicious RSU may manipulate such information, leading to disasters.

We assume the existence of a set of n authorities, as we shall call them the "*tracing authorities*" ($\mathcal{TA} = \{TA_1, \ldots, TA_n\}$), where $n > 4t$ for a threshold $t \geq 1$. The authorities in the set $\mathcal{TA}$ are assumed fully connected via private and authenticated channels (please refer to Section 7 for more security analysis on this issue). We assume also the existence of an authority called the *security center* (SC). The SC is assumed honest-but-curious, that is, it is honest in the sense that it follows the execution steps of the protocol word for word but it is willing to learn any information leaked during execution. We assume that most t tracing authorities could behave maliciously while the majority of them are honest-but-curious. In this sense, we preserve the privacy of the users, since, if at most t authorities collaborate trying to reveal the identity of a particular user, they fail to do so. Only under a valid court order, all honest authorities will obey this order and join to extract the clear identity of the user.

TA's Setup. In the setup phase of $\mathcal{TA}$, a random tracing trapdoor parameter k of bit-length κ (where κ is a security parameter) is to be jointly shared among the set $\mathcal{TA}$ such that each $TA_i \in \mathcal{TA}$ holds a share k_i of k on a t-degree polynomial. Using secure multiparty computations, the authorities $\mathcal{TA}$ compute a blinded version bk of k which is publicly known. While bk is publicly known, we emphasize that k is kept secret and never recovered. The $\mathcal{TA}$ run secure multiparty computations to share the inverse k^{-1} which will be used to trace an identity whenever required.

OBU-SC Interaction (Registeration). In our secure VANET system, the OBU interacts with the SC only one time and this is during registration. An OBU approaches the SC for registration of its identity as follows: the OBU is queried for its identity id and a proof of correctness of this identity (i.e., a proof that id is on the correct form). The SC forwards the accepted identity to all RSUs. From the published blinded tracing parameter bk, an OBU computes its blinded-identity as $\mathrm{bid} = bk^x$. Our protocol ensures that none of the $\mathcal{TA}$ authorities nor the SC authority has any information about which blinded-identity maps to which identity unless the set of $\mathcal{TA}$ collaborate. Notice that $\mathrm{bid} = \mathrm{id}^k$ and hence given a blinded-identity, bid the identity could be revealed by computing $\mathrm{bid}^{k^{-1}}$.

Tracing an OBU. Under court order when a certain blinded-identity bid is required to be traced to its particular identity id, bid is given as input to the tracing authorities $\mathcal{TA}$ where they perform secure multiparty computations to extract the identity id from this bid using the shared trapdoor parameter inverse k^{-1}.

RSU-SC Communication. The SC communicates with the RSUs in only two situations: (i) After registering a new OBU, the SC sends its blinded identity to all RSUs; (ii) In case an OBU is traced to its clear identity and needs to be revoked, the SC instructs the RSUs to remove the corresponding blinded identity from the set $\mathcal{ID}$. Both transmissions are noninteractive and require sending short messages and consequently the overheads are insignificant.

OBU-RSU Communication. Now when the OBU is put on the road, the communication with any RSU is as follows (notice that the blinded tracing trapdoor parameter bk is known to all RSUs): for an OBU to send a message m to an RSU, m is hashed and signed using the OBU's private key x. The blinded-identity bid is attached to the message. The OBU also prepares a NIZK proofs of knowledge to prove to the RSU that

(i) the OBU knows the private key used to produce the signature;

(ii) the private key used in the signature is consistent with the private key used in bid;

(iii) the bid attached to the message corresponds to one of the identities held by the RSU without revealing which identity.

The message is accepted by the RSU only if the above proofs are accepted.

The transmission of authenticated messages from an RSU to an OBU is trivially done using standard digital signature schemes (since anonymity of the RSU is irrelevant) and hence we do not consider this issue no more in the paper. We only notify that, an RSU always attaches the bid of the OBU to whom the message is dedicated.

5. Our Basic Tools

In this section we describe the basic tools that will be used to build our secure VANET system. These tools are partitioned

into two categories: threshold cryptography tools and proofs of knowledge tools. The reader must be familiar with these tools in order to follow the description of our VANET protocol.

5.1. Threshold Cryptography Tools. In this subsection we describe the threshold cryptographic tools used in building our VANET protocol.

5.1.1. Secret Sharing over a Prime Field. Let $s \in Z_q$ be a secret held by some dealer where Z_q is a prime field. In order to share this secret among a set $\mathcal{P} = \{P_1, \ldots, P_n\}$ of $n > t$ players [37], the dealer, defines a polynomial $f(x) = \sum_{j=0}^{t} a_j x^j \bmod q$, he sets $a_0 = s$ and each other coefficient $a_{j\neq 0} \in_R Z_q$. For all $i = 1, \ldots, n$, the dealer secretly delivers $f(i)$ to player P_i. In the reconstruction phase, each player P_i broadcasts $f(i)$, the players are able to compute s from any $t+1$ shares using Lagrange interpolation formula, $s = f(0) = \Sigma_{i\in\mathcal{B}} \lambda_i f(i) \bmod q$, where $\mathcal{B} \subset \mathcal{P}$, $|\mathcal{B}| = t+1$, and λ_i is Lagrange coefficient for player P_i.

5.1.2. Verifiable Secret Sharing. Verifiable secret sharing (VSS) extends polynomial secret sharing in a way that allows the recipients of the shares to verify that their shares are consistent (i.e., that any subset of $t+1$ shares determines the same unique secret). Assuming $n > 2t$, the protocol can tolerate malicious behaviors (e.g., illegal collaboration, sending wrong values, deleting values, etc.) of at most t players. We distinguish two different contributions of VSS; the conditionally secure VSS due to Feldman [38] and the unconditionally secure VSS due to Pedersen [39]. To achieve best security, both of them will be used in our protocol.

Feldman's VSS. Two large primes p and q are chosen such that $q \mid p-1$. The primes p and q and an element $g \in Z_p^*$ of order q are published as the system public parameters. The dealer shares the secret s among the players on a t-degree polynomial $f(x) = \sum_{j=0}^{t} a_j x^j \bmod q$; the dealer also broadcasts the $t+1$ commitments $c_j = g^{a_j} \bmod p$ for all $j = 0, \ldots, t$. These commitments allow each player P_i to verify the consistency of his share $f(i)$ by checking that, $g^{f(i)} = \prod_{j=0}^{t} c_j^{i^j} \bmod p$. If this check fails for any share $f(i)$, P_i broadcasts a complaint. If more than t players broadcasted a complaint, then at least one of them is honest and consequently the dealer is disqualified. Otherwise, the dealer reveals the share $f(i)$ for each complaining player P_i, if the share is correct, P_i is disqualified, otherwise, if the share does not satisfy the commitments or if the dealer does not respond, the dealer is disqualified. During reconstruction of the secret, any player can check the validity of the share broadcasted by any other player via the published commitments to filter out bad shares and safely perform the interpolation. When it comes to the distributed generation of a secret key k and the joint computation of $g^k \bmod p$, Feldman's VSS alone is not secure due to the attacks described in the security analysis section.

Pedersen's VSS. The idea is to use double exponentiation which allows randomization. The public parameters are p, q, g, and h where p, q, and g are as in Feldman's VSS and h is another element in Z_p^* subject to the condition that $\log_g h$ is unknown and assumed hard to compute. In addition to the polynomial $f(x) = \sum_{j=0}^{t} a_j x^j \bmod q$ which holds the secret s as the free term, the dealer sets up a randomizing t-degree polynomial $r(x) = \sum_{j=0}^{t} b_j x^j \bmod q$. He secretly delivers $(f(i), r(i))$ to player P_i for all $i = 1, \ldots, n$. The dealer also publishes the commitments $c_j = g^{a_j} h^{b_j} \bmod p$ for all $j = 0, \ldots, t$. Each player P_i verifies the consistency of his share $f(i)$ by checking that $g^{f(i)} h^{r(i)} = \prod_{j=0}^{t} c_j^{i^j} \bmod p$. If this check fails for any share $f(i)$, P_i broadcasts a complaint. If more than t players broadcast a complaint, then at least one of them is honest and consequently the dealer is disqualified. Otherwise, the dealer reveals the pair $(f(i), r(i))$ for each complaining player P_i, if the pair is correct, P_i is disqualified, otherwise, if the pair does not satisfy the commitments or if the dealer does not respond, the dealer is disqualified. During reconstruction of the secret, any player can check the validity of the share broadcasted by any other player via the published commitments to filter out bad shares and safely perform the interpolation.

5.1.3. Joint Secret Sharing. Joint secret sharing allows the players to jointly share some secret among themselves without the help of the dealer.

Joint Random Secret Sharing (JRSS). JRSS [40] allows a set of n players to jointly share a random secret without the help of the dealer. Each player $P_i \in \mathcal{P}$ picks a random integer $k_i \in Z_q$ and plays the role of the dealer to share k_i among the players over a t-degree polynomial $f_i(x) = k_i + \sum_{j=1}^{t} a_j x^j \bmod q$. Each player $P_i \in \mathcal{P}$ simply sums the shares he receives from the other players to compute a share $f(i) = \sum_{j=1}^{n} f_j(i)$ which is a point on a t-degree polynomial $f(x)$ with its free term equals a random secret $k = \sum_{i=1}^{n} k_i \bmod q$.

Joint Random Verifiable Secret Sharing (JRVSS). To withstand malicious behavior of at most $t < n/2$ players during the JRSS, JRVSS combines the JRSS with Feldman's VSS for computational security or Pedersen's VSS for unconditional security. Simply, each player $P_i \in \mathcal{P}$ picks a random secret integer $k_i \in Z_q$ and plays the role of the dealer in the VSS protocol to share this secret among the other players. Complaints are solved as in the VSS protocol. Finally, each player sums what he has to compute his share on a t-degree polynomial $f(x)$ with its free term $f(0) = \sum_{i=1}^{n} k_i \bmod q$.

Joint Zero Secret Sharing (JZSS). JZSS is a special case of the JRSS in which the random secret shared by each player is a zero. At the end of the JZSS, each player holds a share $f(i)$ on a t-degree polynomial $f(x)$ with its free term $f(0) = 0$.

Joint Zero Verifiable Secret Sharing (JZVSS). As in the JRVSS, to with stand malicious behavior of at most $t < n/2$ players during the JZSS, JZVSS combines the JZSS with

Feldman's VSS for computational security or Pedersen's VSS for unconditional security. Simply, each player $P_i \in \mathcal{P}$ plays the role of the dealer in the VSS protocol to share a zero among the other players. Complaints are solved as in the JRVSS protocol. Finally, each player sums what he has to compute his share on a t-degree polynomial $f(x)$ with its free term $f(0) = 0$. Notice that, from the published commitments, each player can verify that the shared secret is really a zero. This is true since the commitment to a zero will be $g^0 = 1$.

5.1.4. The Multiplication Protocol. Given two secrets a and b shared over t-degree polynomials $A(x)$ and $B(x)$ respectively, the `multiplication` protocol [41] computes $\xi = ab \bmod q$ in a robust way with no information revealed about a or b. Each player P_i locally computes $C(i) = A(i)B(i) \bmod q$. In this case, each $C(i)$ is a share on a $2t$-degree polynomial $C(x) = A(x)B(x) \bmod q$ with $C(0) = \xi$. However, publishing and interpolating the shares $C(1), \ldots, C(n)$ reveals information about $A(x)$ and $B(x)$ [21], consequently, randomizing the shares of $C(x)$ is necessary. To randomize the shares of $C(x)$ without changing $C(0)$, the players run a `JZVSS` to share a zero over a $2t$-degree polynomial $R(x)$ with $R(0) = 0$. Finally, each player $P(i)$ computes and publishes $D(i) = C(i) + R(i)$. The result ξ could be reached by interpolating the $2t$-degree polynomial $D(x)$ with the help of the Berlekamp-Welch decoding scheme [22] to filter out corrupted shares. Since we are interpolating a polynomial of degree deg $= 2t$ and we have a maximum of t malicious players (i.e., at most t possible faults), using the Berlekamp-Welch bound, the number of shares needed in order to correctly interpolate the polynomial is at least deg + 2faults + $1 = 4t + 1$. Hence, we need $n > 4t$.

5.1.5. The Reciprocal Protocol. In the tracing protocol of our TR-LDGS, we are faced with the following problem. Given a secret k which is shared among the players, generate a sharing of the reciprocal of k modulo q with no information revealed about k. Each player P_i holds a share $f(i)$ which is a point on a t-degree polynomial $f(x)$ with $f(0) = k$. To compute shares of k^{-1}, we need $n > 4t$, the n players run the reciprocal protocol [25] as follows:

(i) The players run the `JRVSS`; at the end, each player holds a share $v(i)$ of a random secret v over a polynomial of degree t.

(ii) The players run the `multiplication` protocol and reconstruct the value $\xi = kv \bmod q$.

(iii) Finally each player P_i computes his share of the reciprocal as $\xi^{-1}v(i) \bmod q$, which is a share over a t-degree polynomial with its free term equals $k^{-1} \bmod q$.

5.2. Proofs of Knowledge. In this subsection we describe the proof of knowledge tools used in building our TR-LDGS.

5.2.1. Proof of Knowledge of Discrete log. Let p and q be two large primes, where $q \mid p - 1$, and g is a generator. We review Schnorr's protocol [42] that allows a prover $\mathcal{P}$ to prove to a verifier $\mathcal{V}$ that he knows the $D\log$ of y to the base g modulo p. Let $y = g^x \bmod p$, where x is $\mathcal{P}$'s secret. The protocol is as follows:

(i) $\mathcal{P} \to \mathcal{V}$ picks $r \in_R Z_q$, computes and sends $A = g^r$.

(ii) $\mathcal{V} \to \mathcal{P}$ picks and sends $c \in_R Z_q$.

(iii) $\mathcal{P} \to \mathcal{V}$ computes and sends $s = r + cx$.

(iv) $\mathcal{V}$ accepts if $g^s = ty^c$, else, rejects.

We denote the above protocol by $\Pi_{D\log} \leftarrow P_{D\log}(g, y, x)$.

5.2.2. Proof of Equality of Two Discrete Logarithms. We review the protocol of [42, 43] that is believed to be a zero knowledge proof of equality of two discrete logarithms. In this protocol, the public parameters are two large primes p and q such that $q \mid p-1$, two elements $\alpha, \beta \in Z_p^*$, and the two quantities $G_1, G_2 \in Z_p^*$. The prover ($\mathcal{P}$) proves to a verifier ($\mathcal{V}$) that he knows $x \in Z_q^*$ such that $G_1 = \alpha^x \bmod p$ and $G_2 = \beta^x \bmod p$. The protocol is as follows:

(i) $\mathcal{P} \to \mathcal{V}$:
Choose $r \in_R Z_q^*$ and send ($A = \alpha^r \bmod p$, $B = \beta^r \bmod p$).

(ii) $\mathcal{V} \to \mathcal{P}$:
Choose $c \in_R Z_q^*$ and send c.

(iii) $\mathcal{P} \to \mathcal{V}$:
Send $y = r + cx \bmod q$.

(iv) $\mathcal{V}$:
Check that $\alpha^y = AG_1^c \bmod p$ and $\beta^y = BG_2^c \bmod p$.

The standard method to convert the above protocol to a non-interactive protocol (we denote it $\Pi_{\text{LogEq}} \leftarrow P_{\text{LogEq}}(\alpha, \beta, G_1, G_2, x)$) is by using a sufficiently strong hash function $\mathcal{H}$ and setting $c = \mathcal{H}(A, B)$. The protocol Π_{LogEq} becomes as follows:

(i) $\mathcal{P} \to \mathcal{V}$:
Choose $r \in_R Z_q^*$ and send ($A = \alpha^r \bmod p$, $B = \beta^r \bmod p$, $c = \mathcal{H}(A, B)$ and $y = r + cx \bmod q$).

(ii) $\mathcal{V}$: Check that $\alpha^y = AG_1^c \bmod p$ and $\beta^y = BG_2^c \bmod p$.

5.2.3. Proof of Existence of Discrete log Equality. Let $y_i = \alpha^{x_i} \bmod p$ for $i = 1, \ldots, n$, and let $z = \beta^{x_i} \bmod p$ for some $i \in \{1, \ldots, n\}$. A prover $\mathcal{P}$ demonstrates to a verifier $\mathcal{V}$ that he knows one of the logarithms of y_i ($i \in \{1, \ldots, n\}$) to the base α and that $\log_\alpha y_i = \log_\beta z \bmod q$ without revealing which i. Let $w\log$ the relation holds for $i = 1$ (i.e., $x_1 = \log_\alpha y_1 = \log_\beta z \bmod q$). The protocol is as follows [29]:

(i) $\mathcal{P} \to \mathcal{V}$:
Choose $k_i \in_R Z_q^*$ for $i = 1, \ldots, n$, $c_j \in_R Z_q^*$ for $j = 2, \ldots, n$ and compute

(a) $r_1 = \alpha^{k_1} \bmod p$, $r_i = \alpha^{k_i} y_i^{-c_i} \bmod p$ for $i = 2, \ldots, n$,

(b) $t_1 = \beta^{k_1} \bmod p$, $t_i = \beta^{k_i} z^{-c_i} \bmod p$ for $i = 2, \ldots, n$.

Send the values $(r_1, \ldots, r_n, t_1, \ldots, t_n)$.

(ii) $\mathcal{V} \rightarrow \mathcal{P}$:
Choose and send $c \in_R Z_q^*$.

(iii) $\mathcal{P} \rightarrow \mathcal{V}$:
Calculate $c_1 = c - \sum_{i=2}^{n} c_i \bmod q$, $s_1 = x_1 c_1 + k_1 \bmod q$, and set $s_i = k_i$ for $i = 2, \ldots, n$. Send $(c_1, \ldots, c_n, s_1, \ldots, s_n)$.

(iiii) $\mathcal{V}$:
Check that $c = \sum_{i=1}^{n} c_i$ and that for all $i = 1, \ldots, n$: $\alpha^{s_i} = y_i^{c_i} r_i \bmod p$ and $\beta^{s_i} = z^{c_i} t_i \bmod p$.

The above interactive proof can be transformed into a non-interactive proof that we will denote it by:

$$\prod\nolimits_{\exists \text{LogEq}} \longleftarrow P_{\exists \text{LogEq}}(\alpha, \beta, y_1, \ldots, y_n, z) \tag{1}$$

using a strong hash function $\mathcal{H}$. This can be done by setting

$$c = \mathcal{H}\left(y_1, \ldots, y_n, \alpha, z, \beta, \alpha^{s_1} y_1^{-c_1}, \ldots, \alpha^{s_n} y_n^{-c_n}, \beta^{s_1} z^{-c_1}, \ldots, \beta^{s_n} z^{-c_n}\right). \tag{2}$$

5.3. El-Gamal Cryptosystem. Achieving the confidentiality service in our secure VANET system relies on the El-Gamal cryptosystem [44]. It is known that the El-Gamal cryptosystem works for any family of groups for which the discrete logarithm is considered intractable. Part of the security of the scheme actually relies on the Diffie-Hellman assumption, which implies the hardness of computing discrete logarithms modulo of a large prime. We will present our results for subgroups G_q of order q of Z_p^*, where p and q are large primes such that $q \mid p - 1$. Other practical families can be obtained for elliptic curves over finite fields. We will now briefly describe the El-Gamal cryptosystem, where the primes p and q and at least one generator g of G_q are treated as system public parameters. The key pair of a receiver in the El-Gamal cryptosystem consists of a private key s (randomly chosen by the receiver) and the corresponding public key $h = g^s$. Given a message $m \in Z_p$, encryption proceeds as follows. The sender chooses a random $r \in Z_q$, and sends the pair $(A, B) = (g^r, h^r m)$ as the ciphertext to the receiving party. To decrypt the ciphertext (A, B) the receiver recovers the plaintext as $m = B/A^s$, using the private key s. The El-Gamal encryption is known to be a CPA-secure cryptosystem, there exist extensions to this cryptosystem that achieve CCA1 and CCA2 security.

6. Detailed Protocol Description

Given a security parameter κ, our system works in subgroups G_q of order q of Z_p^*, where p and q are large primes where $q \mid p-1$. Other practical families could be obtained from Elliptic curves over finite fields. The primes p and q and a generator g of G_q are the system global parameters. In number theory, a generator g is picked as follows: let $p = \mu q + 1$, select $\alpha \in_R Z_p^*$ and $\alpha \neq 1$, if $(\beta = \alpha^{\mu} \bmod p \neq 1)$ then β is a valid generator g of order q since in this case $g^q \bmod p = 1$. If $\beta = 1$ (with very low probability), repeat for another α.

Each OBU is assumed to come with the unique pair (x, g^x), where $x \in_R Z_q^*$ of bit-length κ is the OBU's private key and g^x is the OBU's identity id.

6.1. Setup Protocol by the TAs

Protocol Setup-TA. The set of tracing authorities $\mathcal{TA} = \{TA_1, \ldots, TA_n\}$ join together to compute shares of a random integer $k \in Z_q$ of bit-length κ as follows:

(i) Run a JRVSS with Pedersen's VSS as the VSS in place. At the end, each authority TA_i holds a share $K(i)$ of a random secret $k \in_R Z_q^*$ over a t-degree polynomial $K(x)$ with $K(0) = k$.

(ii) The authorities that are not disqualified in the JRVSS in the previous step broadcast the commitments to their shared polynomial based on Feldman's VSS. More precisely, if $K_i(x) = k_i + \sum_{j=1}^{t} a_j$ is the polynomial of authority TA_i then TA_i broadcasts g^{k_i} and $g^{a_j} \bmod p$ for all $j = 1, \ldots, t$.

(iii) For any authority TA_i who receives at least one valid complaint in the previous step, the other authorities join to reconstruct his polynomial $K_i(x)$ and values g^{k_i} and $g^{a_j} \bmod p$ for all $j = 1, \ldots, t$ in the clear.

(iv) Finally, the remaining good authorities join to safely compute $g^k = \prod_{i=1}^{n} g^{k_i} \bmod p$.
At this point, the tracing authorities share the tracing trapdoor parameter k over a t-degree polynomial and have jointly computed the blinded tracing trapdoor, $bk = g^k \bmod p$. They publicize bk (to the RSU's and the security center SC). As a preparation for the tracing protocol, the TAs need to compute shares of $k^{-1} \bmod q$, so they proceedas follows

(v) The TA's run the reciprocal protocol, at the end, each TA_i holds a share $D(i)$ on a t-degree polynomial, $D(x)$ with its free term $D(0) = k^{-1} \bmod q$.

(vi) Each TA broadcasts Feldman's VSS commitments (i.e., to the base g) of all her chosen random polynomials during the reciprocal protocol. These commitments allow the TAs to validate the quantities $V_i = g^{D(i)} \bmod p$ for all i.

The setup-TA protocol ends with each tracing authority TA_iholding a pair of shares $\langle K(i), D(i) \rangle$, where $K(i)$ is a share of the tracing trapdoor parameter k on a t-degree polynomial $K(x)$ while $D(i)$ is a share of the reciprocal of the tracing trapdoor parameter k^{-1} on a t-degree polynomial $D(x)$. Notice that, after the computation of the blinded tracing trapdoor parameter bk, the set of shares $\langle K(1), \ldots, K(n) \rangle$ of k and the commitments $\langle W_1, \ldots, W_n \rangle$ where $W_i = g^{K(i)}$, become useless and could be erased. Each $TA_i \in \mathcal{TA}$ holds (and preserves)

(i) A share $D(i)$ of k^{-1},

(ii) The tuple $\langle V_1, \ldots, V_n \rangle$ as commitments to the shares $\langle D(1), \ldots, D(n) \rangle$ where $V_i = g^{D(i)}$.

6.2. Registration of OBUs

Protocol Reg-OBU. The SC interacts with each OBU as follows:

(i) On the arrival of an OBU, the SC queries this OBU for its unique identity id.

(ii) The OBU sends its identity id and a proof, $\Pi_{D\log} \leftarrow P(g, \mathrm{id}, x)$, that it knows $\log_g \mathrm{id}$.

(iii) The SC runs algorithm $V_{D\log}(g, \mathrm{id}, \Pi_{D\log})$ to verify the validity of id. If not successful, then rejects this id, else, accepts the id and proceed.

(iv) The SC forwards the accepted identity to all RSUs.

Notice that each OBU is able to locally compute its own blinded identity as $\mathrm{bid} = bk^x$ which is id^k.

6.3. OBU-RSU Communication. For an OBU to send a signed message m to an RSU, the OBU proceeds as follows:

(i) hashes m as $h = H(m, r)$ for a random string r,

(ii) computes $s = h^x$,

(iii) prepares $\Pi_{\mathrm{LogEq}} \leftarrow P_{\mathrm{LogEq}}(bk, h, \mathrm{bid}, s, x)$ as a proof that $\log_{bk} \mathrm{bid} = \log_h s$,

(iv) prepares $\Pi_{\exists \mathrm{LogEq}} \leftarrow P_{\exists \mathrm{LogEq}}(bk, g, \mathrm{bid}, \mathcal{ID}, x)$ as an or-proof that there exists one of the m identities $\mathrm{id} \in \mathcal{ID}$ such that $\log_{bk} \mathrm{bid} = \log_g \mathrm{id}$.

For an OBU to be able to prepare $\Pi_{\exists \mathrm{LogEq}}$, it must know all the registered identities. This is easily achieved by allowing each RSU to periodically broadcasts the set $\mathcal{ID}$. Finally the OBU prepares the message to be transmitted as the tuple: $\langle \mathrm{bid}, m, r, h^x, \Pi_{\mathrm{LogEq}}, \Pi_{\exists \mathrm{LogEq}} \rangle$.

6.4. Direct OBU to OBU Communications. Direct OBU to OBU anonymous and authenticated communication is possible under the assumption that the RSU in range periodically broadcasts the set $\mathcal{ID}$ of the registered OBUs. Two OBUs may communicate using their blinded identities and as in the RSU-OBU communication, each OBU includes a prove of knowledge $\Pi_{\exists \mathrm{LogEq}}$ to proof to the other OBU that its identity belongs to the group of registered identities and a proof of knowledge Π_{LogEq} that the signature on a transmitted message is valid. Moreover, two OBUs may use their blinded identities as an El-Gamal public key to start confidential session in the same way as in the RSU-OBU communication (see Section 9).

6.5. Tracing Protocol by the TAs

Protocol Trace-id. Under a court order to trace an identity corresponding to a particular blinded-identity bid, bid is given as an input to all the tracing authorities $\mathcal{TA}$. They join together to reveal the identity id. Each authority $TA_i \in \mathcal{TA}$ performs as follows:

(i) broadcasts $Y_i = (\mathrm{bid})^{D(i)} \bmod p$, where $D(i)$ is TA_i's share of $k^{-1} \bmod q$,

(ii) broadcasts $\Pi_{\mathrm{LogEq}} \leftarrow P_{\mathrm{LogEq}}(g, \mathrm{bid}, V_i, Y_i, D(i))$ to prove that $\log_g V_i = \log_{\mathrm{bid}} Y_i \bmod q$,

From any $t+1$ quantities, Y_i's, that pass the proof Π_{LogEq} successfully, interpolation in the exponent is performed to compute $(\mathrm{bid})^{1/k} = \mathrm{id}$. Interpolation in the exponent is as simple as computing

$$\prod_{i \in \mathcal{B}} (\mathrm{bid})^{D(i)\lambda_i} = (\mathrm{bid})^{\sum_{i \in \mathcal{B}} D(i)\lambda_i} = (\mathrm{bid})^{k^{-1}} = \mathrm{id}, \quad (3)$$

where $|\mathcal{B}| = t + 1$, and λ_i is Lagrangian coefficient of authority TA_i.

6.6. OBU Revocation. The revocation of an OBU is very simple, to revoke an identity id (after being traced), the security center simply instructs the RSUs to remove this identity from the set $\mathcal{ID}$, consequently, the proof $\Pi_{\exists \mathrm{LogEq}}$ attached to a message of a revoked OBU will fail and the messages from this OBU will not be accepted by the RSU.

7. Security Analysis

Setup and Trace Protocol. We first consider the tracing authorities (the setup and trace protocols). The tracing trapdoor parameter is a random, uniformly distributed value k which is distributed on a threshold basis and the value $bk = g^k$ is made public. The protocol is called t-secure, that is, in the presence of at most t malicious authorities:

(i) *Correctness.*

  (a) All subsets of $t+1$ valid shares reconstruct to the same unique secret parameter k.

  (b) Each authority is able to compute the common public value $bk = g^k$.

  (c) k is uniformly distributed in Z_q and hence, bk is uniformly distributed in the subgroup generated by g.

(ii) *Secrecy*. No information on k can be learned by the coalition of at most t members except for what is implied by the value $bk = g^k$.

In the trace protocol, the inverse of the tracing trapdoor parameter k^{-1} is also distributed on a threshold basis where, whenever there is a legal reason, the value $\mathrm{bid}^{k^{-1}} = \mathrm{id}$ is jointly computed and delivered to the court. The security of the trace protocol (correctness and secrecy) is similar to the setup protocol.

JRVSS with Feldman's VSS alone is insecure, since malicious authorities can influence the distribution of the result of Feldman's VSS to a non-uniform distribution. More precisely, the attack works as follows: assume that two

malicious authorities (say TA_1 and TA_2) want to bias the distribution towards values bk whose last bit is 0. TA_1 gives authorities $TA_3, \ldots, TA_{t+2}$ shares which are inconsistent with his broadcast values, the rest of the authorities receive consistent shares. Thus, there will be t complaints against TA_1, yet t complaints are not enough for disqualification. The traitors compute $\alpha = \sum_{i=1}^{n} g^{k_i}$ and $\beta = \sum_{i=2}^{n} g^{k_i}$ (where $k_i = \lambda_i K(i)$). If α ends with 0 then TA_1 will do nothing and continue the protocol as written. If α ends with 1 then it will force the disqualification of TA_1, this is achieved by asking TA_2 to also broadcast a complaint against TA_1, which brings the number of complaints to $t+1$. This action sets the public value bk to β which ends with 0 with probability 1/2. Thus effectively the traitors authorities have forced strings ending in 0 to appear with probability 3/4 rather than 1/2. One must notice that synchronous broadcast does not prevent such attack to take place. Hence, the third requirement for correctness and the secrecy requirement dramatically fail. In Pedersen's VSS, the view of the authorities is independent of the value of the secret k. One may refer to [24, 30] for more details and simulation.

RSU-OBU Communication Security. During communication, when the RSU (verifier) receives a signed message from an OBU_i (signer) for the first time, the signer must prove that the included blinded-identity (bid_i) is valid (i.e., related to an identity in the set of identities $\mathcal{ID}$ held by the RSUs), hence, the signer OBU_i includes the proof $\Pi_{\exists\text{LogEq}}$ which proves to an RSU that the included bid_i corresponds to some $\text{id}_i \in \mathcal{ID}$ with no information revealed about which index i and hence the correspondence of bid_i to id_i is still unknown to an RSU. However, $\Pi_{\exists\text{LogEq}}$ does not ensure that the exponent of $s = h^{x_i}$ is consistent with the exponent of $\text{bid}_i = bk^{x_i}$. To do so, the OBU_i must include a proof Π_{LogEq} which ensures to the RSU that the exponents are equal, or else, the message is rejected. Notice that the proof $\Pi_{\exists\text{LogEq}}$ is included only in the first message transmitted from the OBU_i, once bid_i is verified by the RSU as correct and registered, the RSU stores this bid_i and the proof $\Pi_{\exists\text{LogEq}}$ is omitted from subsequent messages in this session. we emphasize that, the proof Π_{LogEq} must be included within each transmitted message.

Secrecy against SC. We assumed that the SC is honest-but-curious which means that the SC does not behave maliciously, yet, it may misuse any information falls in its hands. From the description of our protocol, the SC interacts with an OBU only once during registration where the OBU sends the clear identity $\text{id} = g^x$. Assuming DHP is infeasible, the SC gains no information in the cryptographic sense about the private parameter x. The proof of knowledge of discrete log $P(g, \text{id}, x)$ reveals no information about the private parameter x. It is important to notice that nobody is able to compute the blinded identity bid of any OBU except the OBU itself, this follows from CDHP where it is infeasible to compute $\text{bid} = g^{xk}$ given $\text{id} = g^x$ and $bk = g^k$, and consequently, nobody is able to detect the correspondence of any id to its bid except the OBU itself (of course, unless the tracing protocol is run). Since the blinded identity $\text{bid} = bk^x = g^{kx}$ is computed by an OBU in the absence of the SC, given the tuple $\langle \text{id} = g^x, \text{bid} = bk^x, \text{bid}' = bk^{x'} \rangle$ where g and bk are known to SC, it is infeasible for an SC to decide on the equality of the exponents (better than a random coin flip) and hence cannot construct a correct mapping. Even if an SC eavesdrops on the channel between the OBU and an RSU, the security of the proof of existence of discrete-log equality $\Pi_{\exists\text{LogEq}}$ does not help it to construct such mapping. In case an SC communicates with an RSU, notice that this communication is one-way, that is, the SC receives nothing from the RSU, it only sends a new registered id or a revoked id.

TAs Communication Channels. We assumed (as in the standard settings of verifiable threshold secret sharing schemes) that the TAs are fully connected via private and authenticated channels. Authentication and privacy are achieved using public-key infrastructure where each authority has her own certified private/public key pair. Although such assumption (regardless of the complexities associated with PKIs, such as scalability) is accepted in the theory of threshold cryptography, an important question arises "*Who certifies the public-keys of the authorities?*" The direct answer of course is "a root CA" and here comes our argument: by this direct answer, we return back to violate the anonymity service which is the main service we are solving in this paper since we still give full trust to a single authority. In this context, we provide two solutions to solve this problem. The first solution is to avoid using PKIs at all. Each authority registers her public-key with every other authority in the set, and this registration must be performed off-line to avoid man-in-the-middle attacks (i.e., the authorities must meet personally before the setup protocol). This solution if combined with the advances in the field of forward secure signatures (please see Section 9.3) does not add much complexities to our protocol but requires some extra presetup work. The second solution is to recall the work in [45] where a provably secure construction is given for secure multiparty computation (with meaningful level of security) without authentication among the parties and hence no need for authenticated channels. However, the computations and communications complexities of the protocol become very high and almost unbearable by many networks.

8. Efficiency Considerations

In this section we discuss some issues concerning the efficiency of our VANET system.

8.1. Adaptive Anonymity. As one may notice the highest computation and communication complexity is during the setup-TA and trace-id protocol, yet, the setup-TA protocol is run only once to setup a tracing trapdoor parameter while the trace-id protocol is run only to trace an identity which is not supposed to be done frequently. A serious complexity issue is the amount of computations performed by an OBU and the bandwidth required for transmission. The quantity that most affects the efficiency of transmission is $\Pi_{\exists\text{LogEq}}$

since the computation of this quantity and the bandwidth is proportional to the number of registered identities in the system. For a huge population, the computation and the verification of this quantity become a serious burden on both the OBU and the RSU (specially the OBU). Hence, we introduce the idea of *adaptive anonymity*, in which, the level of anonymity (that could be selected by the OBU) ranges from *zero-anonymous* to *n-anonymous* where n is the number of registered identities. The OBU, according to its available resources, could select any level of anonymity it desires. Zero-anonymous transmission means that an OBU may select to disable anonymity and instead of attaching its blinded-identity to the message, it attaches its identity in the clear and consequently, the quantity $\Pi_{\exists \text{LogEq}}$ becomes obsolete. For nonzero-anonymous transmission, the level of anonymity may be selected adaptively as follows: The OBU picks $1 < \ell \leq N$ as its selected anonymity level, where N is the number of registered identities. Let $\text{id} = g^x$ be the identity of this particular OBU while bid is its blinded-identity. Next, the OBU picks a subset $\mathcal{B} \subseteq \mathcal{ID}$ where $|\mathcal{B}| = \ell$ and $\text{id} \in \mathcal{B}$. Then, the OBU generates the proof $\Pi_{\exists \text{LogEq}} \leftarrow P_{\exists \text{LogEq}}(bk, g, \text{bid}, \mathcal{B}, x)$ and prepares the message to be transmitted as the tuple:

$$\left\langle \text{bid}, m, r, h^x, \Pi_{\text{LogEq}}, \Pi_{\exists \text{LogEq}}, \mathcal{B} \right\rangle. \tag{4}$$

The complexity (computations and bandwidth) of the transmission is now proportional to the anonymity level ℓ and hence, the adaptive anonymity approach provides a complexity-anonymity tradeoff.

8.2. On the Number of Tracing Authorities. We have described our protocol assuming a high level of security against a corruptive adversary that may corrupt (completely masquerades) a minority (at most t) of the authorities and consequently the number of authorities is required to be at least $4t + 1$. However, in many circumstances (depending on the network environment) a corruption of an authority is ensured to be impossible and hence, all authorities are assumed honest-but-curious (equivalently, the adversary has only eavesdropping and halting capabilities), in this case, the lower bound on the number of authorities may be reduced to $3t + 1$ where $2t + 1$ authorities perform the computations while t of them may be disconnected at any time without disrupting the correct output. Moreover, there is no need for verifiable secret sharing (i.e., Feldmann's and Pedersen's VSS become obsolete), only JRSS and JZSS are incorporated without the need for verifiability and hence, the computations complexity and bandwidth are significantly reduced. Furthermore, if an adversary has only eavesdropping capabilities without being able to halt an authority, all authorities are assumed online and active all the time, consequently, we may reduce the number of authorities by t, hence becomes only $2t+1$ where the adversary is able to eavesdrop no more than t of them.

9. Other Security Services

In this section we discuss other security services and enhancements that could be provided by our VANET system.

9.1. Confidential Communications. Until now, we were only considering authenticated (and nonrepudiable) transmission from an OBU to an RSU and vice versa. Now we turn our attention to the confidentiality service. First, we need to show why confidential transmission is necessary, since if the transmission is anonymous and the identity of the message originator is unknown, then there is no strong reason to consider confidential transmission. Yet, there is still a strong reason for this service: consider the situation where a particular OBU is revoked. We have already shown that an RSU will not accept messages from a revoked OBU, but, what about messages transmitted from an RSU? Since this OBU is revoked, it must not accept any service messages from the RSU (unless the information included in the transmitted message is for free) or else we assume the service is still running for this OBU. Nothing prevents the OBU from accepting messages from an RSU since these messages are sent in the clear. In the context of this scenario, confidential transmission is necessary to be realized. Our goal in this section is to realize this service while maintaining anonymity, the possibility of tracing identities and all the other system attributes. Our idea is that the RSU replies to an OBU using the OBU's blinded-identity as its public key for an El-Gamal cryptosystem where bk is the generator for the system, this requires spending a little more time in the TA-setup phase to test and ensure that bk is also a generator. Simply, bk is a generator if $(bk)^q = 1 \bmod p$. More precisely, the RSU performs as follows:

(i) on the reception of a signed message (or request) from an OBU, it checks the validity of the attached blinded-identity bid as before,

(ii) picks a random session-key K_s for any available symmetric cryptosystem,

(iii) prepares an El-Gamal encryption of this session-key using bid as the public key. The ciphertext is the pair $\langle A, B \rangle$ where $A = bk^r$ and $B = \text{bid}^r K_s$,

(iv) generates the ciphertext $C = E_{K_s}(m)$ for a message m,

(v) sends the tuple $\langle A, B, C, \text{bid} \rangle$ as a hybrid encryption of the message m.

Only the OBU that knows the private key x corresponding to this particular blinded-identity will be able to decrypt the message m. The OBU decrypts for $K_s = B/A^x$ and decrypts C for m. Moreover, this session-key may be used between The RSU and the OBU as an established private session.

9.2. Proactive Security. In (t, n)-threshold, secret sharing schemes security is assured if throughout the entire lifetime of the secret, it is assumed that the adversary is restricted to compromise at most t of the n locations. For long-lived and sensitive secrets (tracing trapdoor parameter k in our system), this assumption about the adversary may

be insufficient since the adversary may behave in a mobile manner, that is, she jumps among the authorities collecting as much information as she can. She has the whole life-time of the secret to do so. An easy strategy is to periodically refresh the secret parameter itself by reinitializing the setup phase which leads to a fresh k and hence a fresh blinded tracing trapdoor parameter bk and erasing all previous versions. However, such an attempt disables traceability of blinded identities generated using previous versions of bk and hence such a strategy becomes insecure.

Therefore, what is actually required to protect the secrecy of the tracing trapdoor parameter k is to be able to periodically renew its shares without changing its value in such a way that any information learned by the adversary about individual shares becomes obsolete after the shares are renewed. Similarly to avoid the gradual destruction of the information by corruption of shares it is necessary to periodically recover lost or corrupted shares without compromising the secrecy of any of the shares [46].

Proactive secret sharing is mainly based on the $(+,+)$-homomorphic property of polynomial secret sharing and secure computations in the exponent of Feldman's VSS. The life-time of the secret is divided into time periods. The time period must be small enough to ensure that the adversary will not exceed the threshold t in her attack during any period, and big enough to reduce complexity to the minimum. At the beginning of every time period, the update of shares is triggered and it consists of the following stages (please refer to [46] for more details):

(1) update of personal private/public keys,

(2) detection of lost or corrupted shares,

(3) verifiable shares renewal,

(4) erasure of old shares.

Here we must emphasize that the erasure of previous versions of shares and private keys is a must and all parties (except the malicious minority) must be honest about erasing their old parameters; since we are working in the static adversary model, the proactive VANET system is secure only in the erasure model.

9.3. Forward Security. Since for proactive security to be effective in protecting the tracing trapdoor parameter and since we assume static adversary model, all authorities must erase their past information after shares update. Since all private and authenticated communications among the authorities are performed using private/public keys. An adversary may target these private keys so that she is able to decrypt and recover past ciphertexts that she may have recorded on the channels. Here, we recall the advances in forward-secure cryptography. The goal of forward-secure cryptosystems is to provide the benefits of frequent rekeying without incurring the costs of changing public keys (and associated overhead). They enable the user to frequently erase the secret key while maintaining the same public key. The notion of a forward-security originated from the notion of "perfect forward secrecy" for key agreement, which protects (locks-down) past traffic even after long-term keys are compromised.

A generic forward-secure signature scheme is constructed from any ordinary (non-forward-secure) scheme used as a black-box [47]. The security of the forward-secure scheme is then reduced to the security of the underlying ordinary scheme (i.e., if some adversary can efficiently compromise the forward-secure scheme, then we can construct an adversary compromising the security of the underlying ordinary scheme). The efficiency of a generic scheme is usually measured in terms of the number of the invocations of the underlying ordinary scheme, the number of the ordinary keys, and so forth.

Forward-secure public key encryption proved harder to achieve, and the first and so far the only result in that area was obtained in [48]. The constructions share similarities with previous tree-based, forward-secure signature schemes. In the construction, however, time periods are associated with all the nodes of the tree (in a preorder traversal) instead of associating time periods with the leaves only; this improves the efficiency of the key-generation and key-update algorithms.

10. Conclusions

In this paper we introduced an efficient and secure protocol for message transmission in VANETs that integrates anonymity, traceability, revocation, and confidentiality services in an efficient and robust way. Our protocol realizes noninteractive transmission which is suitable for VANETs as a high mobility network. The protocol uses the idea of adaptive anonymity to further reduce complexity as possible. Our protocol, although has some complexities in the setup phase, yet provides a very simple and efficient way of communication among units on the road. Up to our knowledge, our protocol is the first to achieve all these security services in a noninteractive way of communication.

Acknowledgment

The author is greatly indebted to the anonymous reviewers of the IJVT for their valuable comments and suggestions that brought this paper to its final version.

References

[1] H. J. Miller and S.-L. Shaw, *Geographic Information Systems for Transportation*, Oxford University Press, Oxford, UK, 2009.

[2] http://www.standards.its.dot.gov/.

[3] Dedicated Short Range Communications (DSRC), http://www.leearmstrong.com/DSRC/DSRCHomeset.htm.

[4] R. Bishop, "A survey of intelligent vehicle applications worldwide," in *Proceedings of IEEE Intelligent Vehicles Symposium (IVS '00)*, pp. 25–30, Dearborn, Mich, USA, October 2000.

[5] C. Schroth, M. Strassberger, R. Eigner, and S. Eichler, "A framework for network utility maximization in VANETs," in *Proceedings of the 3rd ACM International Workshop on Vehicular Ad Hoc Networks (VANET '06)*, pp. 86–87, Los Angeles, Calif, USA, September 2006.

[6] http://www.kvh.com/.

[7] http://trac-net.com/.

[8] J. Yin, T. Elbatt, G. Yeung, et al., "Performance evaluation of safety applications over DSRC vehicular ad hoc networks," in *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks (VANET '04)*, pp. 1–9, Philadelphia, Pa, USA, October 2004.

[9] T. Mak, K. Laberteaux, and R. Sengupta, "A multi-channel VANET providing concurrent safety and commercial services," in *Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks*, pp. 1–9, Cologne, Germany, September 2005.

[10] ITSA, DoT, "National intelligent transportation systems program plan: a ten-year vision," Report, Intelligent Transportation Society of America and Departemnt of Transportation, Washington, DC, USA, January 2002, http://www.itsa.org/itsa/files/pdf/National10YearPlanITSFull.pdf.

[11] J.-P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *IEEE Security and Privacy*, vol. 2, no. 3, pp. 49–55, 2004.

[12] M. Raya and J.-P. Hubaux, "The security of vehicular ad hoc networks," in *Proceedings of the 3rd ACM Workshop on Security of Ad Hoc and Sensor Networks*, pp. 11–21, Alexandria, Va, USA, November 2005.

[13] X. Sun, X. Lin, and P.-H. Ho, "Secure vehicular communications based on group signature and id-based signature scheme," in *Proceedings of IEEE International Conference on Communications (ICC '07)*, pp. 1539–1545, Glasgow, UK, June 2007.

[14] G. Calandriello, P. Papadimitratos, A. Lioy, and J.-P. Hubaux, "Efficient and robust pseudonymous authentication in VANET," in *Proceedings of the 4th ACM International Workshop on Vehicular Ad Hoc Networks, in Conjunction with the 13th Annual International Conference on Mobile Computing and Networking (MobiCom '07)*, pp. 19–28, Montreal, Canada, September 2007.

[15] F. Dötzer, "Privacy issues in vehicular ad hoc networks," in *Proceedings of the 5th International Workshop on Privacy Enhancing Technologies (PET '05)*, vol. 3856 of *Lecture Notes in Computer Science*, pp. 197–209, Cavtat, Croatia, May-June 2005.

[16] J. Choi, M. Jakobsson, and S. Wetzel, "Balancing auditability and privacy in vehicular networks," in *Proceedings of the 1st ACM International Workshop on Quality of Service and Security in Wireless and Mobile Networks*, pp. 79–87, Montreal, Canada, October 2005.

[17] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki, "CARAVAN: providing location privacy for VANET," in *Proceedings of Embedded Security in Cars (ESCAR '05)*, Cologne, Germany, November 2005.

[18] X. Lin, X. Sun, P.-H. Ho, and X. Shen, "GSIS: a secure and privacy-preserving protocol for vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3442–3456, 2007.

[19] J. Guo, J. Baugh, and S. Wang, "A group signature based secure and privacy-preserving vehicular communication framework," in *Proceedings of the Mobile Networking for Vehicular Environments, in Conjunction with the 26th IEEE International Conference on Computer Communications (INFOCOM '07)*, pp. 103–108, Anchorage, Alaska, USA, May 2007.

[20] E. Fonseca, A. Festag, R. Baldessari, and R. L. Aguiar, "Support of anonymity in VANETs—putting pseudonymity into practice," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC '07)*, pp. 3400–3405, Hong Kong, March 2007.

[21] Y. Xi, K.-W. Sha, W.-S. Shi, L. Schwiebert, and T. Zhang, "Probabilistic adaptive anonymous authentication in vehicular networks," *Journal of Computer Science & Technology*, vol. 23, no. 6, pp. 916–928, 2008.

[22] M. Manulis, "Democratic group signatures: on an example of joint ventures," in *Proceedings of the ACM Symposium on Information, Computer and Communications Security (ASIACCS '06)*, p. 365, Taipei, Taiwan, March 2006.

[23] M. Manulis, A.-R. Sadeghi, and J. Schwenk, "Linkable democratic group signatures," in *Proceedings of the 2nd International Conference on Information Security Practice and Experience (ISPEC '06)*, vol. 3903 of *Lecture Notes in Computer Science*, pp. 187–201, Hangzhou, China, April 2006.

[24] M. H. Ibrahim, "Resisting traitors in linkable democratic group signatures," *International Journal of Network Security*, vol. 9, no. 1, pp. 51–60, 2009.

[25] M. H. Ibrahim, "Eliminating quadratic slowdown in two-prime RSA function sharing," *International Journal of Network Security*, vol. 7, no. 1, pp. 106–113, 2008.

[26] M. H. Ibrahim, "Efficient dealer-less threshold sharing of standard RSA," *International Journal of Network Security*, vol. 8, no. 2, pp. 139–150, 2009.

[27] M. H. Ibrahim, I. I. Ibrahim, I. A. Ali, and A. H. El-Sawy, "Fast fully distributed and threshold RSA function sharing," in *Proceedings of Information Systems: New Generation Conference (ISNG '04)*, pp. 11–15, Las Vegas, Nev, USA, November 2004.

[28] M. H. Ibrahim, I. I. Ibrahim, and A. H. El-Sawy, "Fast three party shared generation of RSA keys without distributed primality test," in *Proceedings of Information Systems: New Generation Conference (ISNG '04)*, pp. 5–10, Las Vegas, Nev, USA, November 2004.

[29] R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin, "Robust threshold DSS signatures," *Information and Computation*, vol. 164, no. 1, pp. 54–84, 2001.

[30] R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin, "Secure distributed key generation for discrete-log based cryptosystems," *Journal of Cryptology*, vol. 20, no. 1, pp. 51–83, 2007.

[31] http://www.onion-router.net/.

[32] M. G. Reed, P. F. Syverson, and D. M. Goldschlag, "Proxies for anonymous routing," in *Proceedings of the 12th Annual Computer Security Applications Conference (CSAC '96)*, pp. 95–104, San Diego, Calif, USA, December 1996.

[33] D. L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, no. 2, pp. 84–88, 1981.

[34] R.L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *Proceedings of the 7th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT '01)*, vol. 2248 of *Lecture Notes in Computer Science*, pp. 552–565, Springer, Gold Coast, Australia, December 2001.

[35] J. K. Liu, V. K. Wei, and D. S. Wong, "Linkable spontaneous anonymous group signature for ad hoc groups," in *Proceedings of the 9th Australasian Conference on Information Security and Privacy (ACISP '04)*, vol. 3108 of *Lecture Notes in Computer Science*, pp. 325–335, Springer, Sydney, Australia, July 2004.

[36] A. Bender, J. Katz, and R. Morselli, "Ring signatures: stronger definitions, and constructions without random oracles," in *Proceedings of the 3rd Theory of Cryptography Conference*, vol. 3876 of *Lecture Notes in Computer Science*, pp. 60–79, Springer, New York, NY, USA, March 2006.

[37] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[38] P. Feldman, "A practical scheme for non-interactive verifiable secret sharing," in *Proceedings of the 28th Annual Symposium on the Foundations of Computer Science (SFCS '87)*, pp. 427–437, IEEE, Los Angeles, Calif, USA, October 1987.

[39] T. Pedersen, "Non-interactive and information theoretic secure verifiable secret sharing," in *Proceedings of the 11th Annual International Cryptology Conference on Advances in Cryptology (Crypto '92)*, J. Feigenbaum, Ed., vol. 576 of *Lecture Notes in Computer Science*, pp. 129–140, Springer, Santa Barbara, Calif, USA, August 1992.

[40] I. Ingemarsson and G. J. Simmons, "A protocol to set up shared secret schemes without the assistance of mutually trusted party," in *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques on Advances in Cryptology (EUROCRYPT '90)*, I. B. Damgard, Ed., vol. 473 of *Lecture Notes in Computer Science*, pp. 266–282, Springer, Aarhus, Denmark, 1990.

[41] A. Wigderson, M. B. Or, and S. Goldwasser, "Completeness theorems for noncryptographic fault-tolerant distributed computations," in *Proceedings of the 20th Annual Symposium on the Theory of Computing (STOC '88)*, pp. 1–10, ACM, Chicago, Ill, USA, May 1988.

[42] C. P. Schnorr, "Efficient signature generation by smart cards," *Journal of Cryptology*, vol. 4, no. 3, pp. 161–174, 1991.

[43] E. D. Chaum and T. P. Pedersen, "Wallet databases with observers," in *Proceedings of the 12th Annual International Cryptology Conference on Advances in Cryptology (Crypto '92)*, vol. 740 of *Lecture Notes in Computer Science*, pp. 89–105, Santa Barbara, Calif, USA, August 1992.

[44] T. El Gamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," in *Proceedings of Advances in Cryptology (Crypto '84)*, vol. 196 of *Lecture Notes in Computer Science*, pp. 10–18, Paris, France, April 1984.

[45] B. Barak, R. Canetti, Y. Lindell, R. Pass, and T. Rabin, "Secure computation without authentication," in *Proceedings of the 25th Annual International Cryptology Conference on Advances in Cryptology (Crypto '05)*, vol. 3621 of *Lecture Notes in Computer Science*, pp. 361–377, Santa Barbara, Calif, USA, August 2005.

[46] A. Herzberg, S. Jarecki, H. Krawczyk, and M. Yung, "Proactive secret sharing or: how to cope with perpetual leakage," in *Proceedings of the 15th Annual International Cryptology Conference on Advances in Cryptology (Crypto '95)*, vol. 963 of *Lecture Notes in Computer Science*, pp. 339–352, Santa Barbara, Calif, USA, August 1995.

[47] M. Bellare and S. Miner, "A forward-secure digital signature scheme," in *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology (Crypto '99)*, M. Wiener, Ed., vol. 1666 of *Lecture Notes in Computer Science*, pp. 431–448, Springer, Santa Barbara, Calif, USA, August 1999.

[48] R. Canetti, S. Halevi, and J. Katz, "A forward-secure public-key encryption scheme," in *Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT '03)*, E. Biham, Ed., vol. 2656 of *Lecture Notes in Computer Science*, p. 646, Springer, Warsaw, Poland, May 2003.