Non-Interactive Privacy-Preserving Sybil-Free Authentication Scheme in VANETs

Mahdi Akil
Karlstad University, Sweden
mahdi.akil@kau.se

Leonardo Martucci
Karlstad University, Sweden
leonardo.martucci@kau.se

Jaap-Henk Hoepmann
Radbound University
jhh@cs.ru.nl

***Abstract*—In vehicular ad hoc networks (VANETs), vehicles exchange messages to improve traffic and passengers' safety. In VANETs, (passive) adversaries can track vehicles (and their drivers) by analyzing the data exchanged in the network. The use of privacy-enhancing technologies can prevent vehicle tracking but solutions so far proposed either require an intermittent connection to a fixed infrastructure or allow vehicles to generate concurrent pseudonyms which could lead to identity-based (Sybil) attacks. In this paper, we propose an anonymous authentication scheme that does not require a connection to a fixed infrastructure during operation and is not vulnerable to Sybil attacks. Our scheme is built on attribute-based credentials and short lived pseudonyms. In it, vehicles interact with a central authority only once, for registering themselves, and then generate their own pseudonyms without interacting with other devices, or relying on a central authority or a trusted third party. The pseudonyms are periodically refreshed, following system wide epochs.**

I. Introduction

Vehicular ad hoc networks (VANETs) are spontaneously created self-organizing mobile computer networks comprised of vehicles and support devices. The objective of VANETs is to establish the communication infrastructure on top of which services can be offered to drivers, road management companies and transportation agencies. VANET applications include driver assistance, such as car platooning, parking spot locator, and traffic notification, in addition to safety features, such as lane change assistance, collision warning, and accident notifications [19]. Transportation agencies would be able to use VANETs to improved real-time traffic management, planning roadworks and upgrades to the transportation network.

A key aspect of VANETs is the capacity of vehicles to set vehicle-to-vehicle (V2V) wireless communication channels. V2V communication is used to forward and exchange information within the VANET. For the aforementioned applications, the information is exchanged on a continuous and regular basis and includes: the vehicle's permanent identifiers, its geographical position, and its path. Information about vehicles can be linked to individual drivers and therefore it is personal data. A passive adversary eavesdropping the V2V communication can track vehicle identifiers, position, and path [22]. To prevent this, VANETs need to offer security and privacy guarantees.

To improve the privacy and security in VANETs, communication schemes based on anonymous credentials (ACs) have been proposed [16], [29]. The disadvantage of existing proposals is that they either rely on the availability of a trusted third party at all times or are vulnerable to Sybil attacks [13].

In this paper, we propose a non-interactive privacy-preserving and Sybil-free anonymous authentication scheme for VANETs based on attribute-based credentials [12]. Our scheme has a decentralized architecture that allows vehicles to generate their own pseudonyms without the intervention of a trusted third party or central authority (CA). The pseudonyms are derived from a (certified) master secret, which is an AC issued by the CA, and allow vehicles to mutually authenticate themselves anonymously. The pseudonyms in our scheme are fixed within an epoch and unlinkable across epochs, where epochs are system wide fixed time intervals.

Our scheme is based on non-interactive zero-knowledge proofs of knowledge (NIZKP) and Camenisch-Lysyanskaya (CL) signatures [9], [15], [25]. In this paper, we focus on data communication privacy. Nonetheless, vehicles could still be tracked if the applications that are built on top of our protocol allow for that.

The remainder of this paper is structured as follows. The background on VANETs and a short description of attribute based credentials are presented in Section II. Section III describes the system and the adversary model, and Section IV outlines the detailed explanation of our proposed privacy-preserving authentication scheme. The security, privacy and performance analysis of our proposal are presented in Section V. Section VI presents the related work and its limitations. Section VII discusses the flexibility of our model and its extensions. The conclusions, limitations, and future directions are summarized in Section VIII.

II. Background

In this section, we briefly introduce VANETs, their typical network architecture, and their limitations. We also describe attribute-based credentials, which are a category of ACs.

A. Privacy-enhancing vehicular ad hoc networks

In VANETs, vehicles are mobile network nodes that interact with each other to share information. Vehicles exchange

Symposium on Vehicles Security and Privacy (VehicleSec) 2023
27 February 2023, San Diego, CA, USA
ISBN 1-891562-88-6
https://dx.doi.org/10.14722/vehiclesec.2023.23007
www.ndss-symposium.org

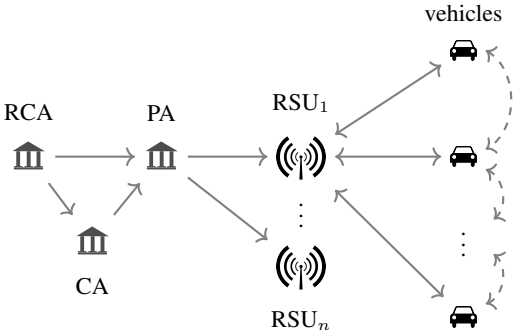

Fig. 1: A privacy-enhancing VANET architecture with a RCA, a CA, a PA, n RSUs and vehicles. The vehicles exchange data between themselves (V2V) and with the RSUs.

messages that are used by onboard services and applications, and by road traffic managing services.

In the literature, a privacy-enhancing VANET architecture, i.e., a VANET in which vehicles are identified by pseudonyms, usually contains the following elements: (*a*) a certificate authority (CA) hierarchy, (*b*) a pseudonym authority, (*c*) road-side units and (*d*) vehicles [4], [18], [26]. An example of this architecture is depicted in Figure 1.

- The *root certificate authority* (RCA) is responsible for vehicle registration and is at the root of the CA hierarchy. The RCA may issue certificates to other CAs. The RCA and CAs issue digital certificates to pseudonym authorities, which grant them rights to issue pseudonyms.
- The *pseudonym authority* (PA) issues pseudonyms for registered vehicles. The pseudonyms are often valid for a limited amount of time, and eventually expire.
- *Road-side units* (RSUs) are access points deployed along roads to improve network availability and performance. RSUs relay network traffic between vehicles and the PA.
- *Vehicles* are equipped with on-board units (OBUs) that broadcast messages to neighboring vehicles, including its position, direction, speed, acceleration, deceleration, and surrounding traffic information. This information allows a vehicle to perceive its surrounding environment and improve the overall traffic. The OBU also securely store the cryptographic keys and other secret information.

Limitations: This privacy-enhancing VANET architecture has the following two important limitations:

- Vehicles have to contact the PA to get a new set of pseudonyms. If there are no unused pseudonyms in the pool, then vehicles have to contact the PA and request a new pool of pseudonyms. In cases where vehicles do not have access to an RSU (e.g., rural roads), they would have to reuse pseudonyms, which weakens the privacy claims of solutions based on this type of architectures, as pseudonyms should ideally not be reused in uncontrollable or unbounded conditions.
- This type of architecture requires a large number of the RSUs to be deployed, which leads to a significant expense on infrastructure [14]. Also, certificate revocation lists (CRL) are distributed via the RSUs. Without access to an updated CRL, vehicles are not able to check if a certificate is revoked or not.

B. Attribute-based credentials

The non-interactive privacy-preserving Sybil-free authentication scheme we propose in this paper uses a construction of attribute-based credentials (ABCs) to solve the limitations of privacy-enhancing VANETs that are based on the architecture presented in Section II-A. In this section, we introduce ABCs.

1) ABCs: are the digital equivalent of a passport, drivers license, or diploma. They contain one or more *attributes* (typically properties or qualifications of a user, like age), that the *issuer* of the ABC believes belong to its owner. ABCs can have several attributes that can be shown independently.

An ABC allows a *prover* (typically a user) to prove a selection of attributes to a *verifier* (typically a service provider) in a privacy-preserving manner [12]. This proof is secure in the sense that a user can only prove the possession of attributes contained in ABCs that have been issued to her earlier, and *unlinkable* in the sense that no issuer can link the use of any of the ABCs it issued to their later use in a proof, and that no verifier can tell whether the same ABC is used in two different proofs (unless this is revealed through the attributes disclosed) [10]. The unlinkability property is what makes ABCs privacy preserving [6].

The parties involved in an ABC scheme are the following:

- The *scheme authority* that is responsible for defining the global cryptographic parameters and certify issuers.
- *Issuers* are the entities responsible for issuing ABCs for the users and publish the public parameters.
- *Provers* (or users) are entities who hold valid ABC that contains attributes which are vouched for by the issuer.
- *Verifiers* are parties that can verify attributes received from provers using the published public parameters.

The ABCs issued to a single user are bound to the user's private key k_U and are signed by an issuer I. k_U is an attribute that is always hidden. We write $C_I(a_0, .., a_L)$ for an ABC issued by I, to user U, with $k_U = a_0$, and containing attributes $\{a_1, .., a_L\}$. In the remainder of this paper we assume that users hold a single ABC containing all their attributes.

Users can prove ownership of a selection of attributes in their ABCs to a verifier using zero knowledge proofs (ZKP) that: (*a*) reveals the values of the set of disclosed attributes A_d, while keeping the values of the set of hidden attributes A_h confidential (including the private key $a_0 = k_U$), (*b*) reveals the identity of the issuer I, and (*c*) proves that the credential is properly signed without revealing the actual signature. This makes the use of ABCs unlinkable. This selective disclosure proof can include a signature over a message m chosen by the user, independent of the attributes contained in the ABC.

The ZKP over a message m, proving ownership of a credential issued by I, containing the revealed set of attributes A_d and hidden set A_h is denoted as $PK\{A_h|C_I(A_h; A_d)\}(m)$. It proves that the message m was constructed by a user who has the set of attributes A_d, which were signed by I.

There are many ways to construct ABCs, e.g., U-Prove from Microsoft [21] is based on blind signatures and IBM's idemix uses ZKP [10]. In Section IV-A, we show how our scheme is implemented based on idemix.

2) Domain specific pseudonyms: A *pseudonym* is a privacy-preserving identifier used to minimize personal data disclosure. A pseudonym is an "identifier of a subject which is different from the subject's real name" [23]. Some ABCs allow users to generate domain specific pseudonyms, derived from their secret key, in such a way that:

- a user can only generate one valid pseudonym for a specific domain (specified by its unique string or number),
- a user can prove the validity of a pseudonym, and
- two different pseudonyms generated for two different domains, and belonging to a given user are unlikable.

We write $N(a_0, dom)$ as a deterministic domain specific pseudonym of user U with $a_0 = k_U$ for the domain specified by dom (typically a string). A selective disclosure proof over a message m of an ABC issued by I that discloses attributes A_d and the pseudonym N generated for the domain dom is

$$PK\{A_h | C_I(A_h; A_d) \wedge N(a_0, dom)\}(m),$$

where $a_0 \in A_h$.

3) Inspection: enables accountability. A verifier may report an prover linked to misbehavior (however defined in the system) to an *inspector*. All messages exchanged in the VANET includes $[a]$, which is an encryption of attribute $a \in A_h$ that can identify the prover. A verifier forwards $[a]$ to the inspector, who decrypts it and obtain a. The disclosure proof is

$$PK\{A_h | C_I(A_h; A_d) \wedge [a]\}(m, [a]).$$

4) ABC properties: The main ABC properties are [2], [6]:

- Security properties:
  - *authenticity:* the ABC is signed by the issuer and it guarantees that the attributes belong to the user,
  - *integrity:* the attributes contained in an ABC have not been tampered,
  - *non-transferability:* the ABC is bound to the user that was involved in the issuing protocol.
- Privacy properties:
  - the ABC's hidden attributes are not revealed,
  - pseudonyms cannot be linked to the information gathered by the issuer during the issuing protocol,
  - pseudonyms generated by the same user for different domains cannot be linked.

ABCs can demonstrate the possession of a credential or of selected certified attributes. Therefore, ABCs are suitable to be used in VANETs where a user needs to prove that she holds a credential in order to exchange messages. Moreover, ABCs allow users to verify credential authenticity without having to communicate with a trusted third party.

III. Our system model

In this section, we describe the main goals of our scheme, our system architecture, and adversary model.

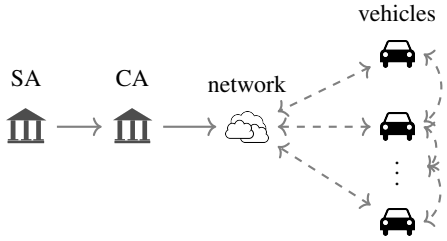

Fig. 2: Our new system architecture requires neither PAs nor RSUs. Only intermittent access to the network is expected.

A. Objectives

Our goals is to allow vehicles to broadcast messages that can be authenticated using an ABC. In addition, we require that (*a*) all messages from a vehicle should be linked for a short period of time ϵ (an epoch) for safety reasons [1], and (*b*) two messages m_1 and m_2 sent from the same vehicle in different time periods (epochs) ϵ_1 and ϵ_2 should not be linked.

In scenarios where vehicles are required to stay linkable for longer periods, that might be achieved by an application running on top of our protocol.

The security and privacy properties aimed by our system are the following:

- *Authenticated broadcasting:* verifiers can verify that a received messages is from vehicles that own a valid ABC.
- *Conditional anonymity:* a prover can send messages while maintaining her anonymity. Only the CA can de-anonymize provers.
- *Impersonation:* a prover u_1 is not able to encrypt a hidden attribute belonging to prover u_2.
- *Non-repudiation:* a prover is not able to dispute not being the sender of m, if she is the actual sender of m.
- *Offline verification:* messages can be generated and verified without the interference of a trusted third party.
- *Linkability within epochs:* all messages from a prover u_1 sent during an epoch ϵ_1 are linkable.
- *Unlinkability across epochs:* two messages from a prover u_1 sent in different epochs ϵ_1 and ϵ_2 are unlinkable.
- *Sybil-freeness:* a prover is not able to have two or more pseudonyms in a given epoch.

B. Our VANET system architecture

Our VANET environment consists of the following entities: a scheme authority (SA), a (trusted) certificate authority (CA), the participating vehicles, and a communication network. Our environment is illustrated in Figure 2.

- The SA generates and distributes the public parameters of the system and certifies the CAs. The role of the SA could be fulfilled by a trusted global authority, just like ICAO is trusted for issuing standards and managing a global key directory for electronic passports.
- The CA issues ABCs to vehicles and may de-anonymize vehicles that it registered. The role of the CA is typically implemented by the national transportation authority.

- *Vehicles* are equipped with an OBU that holds the vehicle's private key in a tamper-proof storage. OBUs are installed and initialized by the vehicles' manufacturers. Vehicles act as both *provers* and *verifiers* in our system.
- A *communication network* for (*a*) V2V message exchange and for (*b*) intermittent communication between vehicles and the CA.

In our model, (*a*) the SA publishes its public parameters globally, (*b*) every country has a CA which is registered with the SA, (*c*) vehicles register with the CA of their country, and (*d*) every country has a different set of inspection keys.

For example, a vehicle registered in Norway and located in Sweden can communicate with cars registered in Sweden, but cannot be de-anonymized by the Swedish authorities because the Norwegian CA and the Swedish CA have different inspection keys. If the Norwegian vehicle misbehaves in Sweden, then the Swedish CA needs to contact the Norwegian CA to de-anonymize the vehicle.

C. The adversary model

The adversary's goal is to either expose the vehicle's private key k_U from its a pseudonym or to link two or more pseudonyms generated in different epochs to a given vehicle.

An adversary can eavesdrop, modify and inject data between the communicating parties. She may attempt to impersonate participants, and launch Sybil attacks. We assume that an adversary might be a participating vehicles in the VANET, i.e, she may obtain one valid ABC from the CA.

The limitations of the adversaries are the following: we assume that they are not able to compromise the CA, break the underlying cryptographic building blocks, and identify the pseudonym holders based on metadata, such as IP, MAC addresses, or imperfections on the radio transmission signatures.

IV. Implementation of our model

In this section, we review the roles of the SA and the CAs in our model, provide an overview of how vehicle are registered, pseudonyms are generated, and messages are verified.

We implement our system model following the abstract description of the ABC scheme in Section II-B. There are three stages in our model: system initialization, vehicle registration, and V2V communication. The overview of the operations within each of these three stages is presented below:

- System initialization:
  - SA generates system parameters and sends them to the CAs and car manufacturers.
  - CA generates the key pair for issuing ABCs.
  - CA generates a key pair for verifiable encryption, which will be used for inspection [8].
  - SA certifies the public keys of the CAs.
- Vehicle registration:
  - A vehicle generates and stores its private key $k_U = a_0$ in its OBU.
  - A vehicle and the CA engage in the credential issuing protocol. If successful, the vehicle obtains its ABC $C_{CA}(a_0, \ldots, a_L)$, where $\{a_1, \ldots, a_L\}$ are the vehicle attributes, and a public inspection key pk_{in}.
  - The vehicle stores C_{CA} and pk_{in}. The CA stores the vehicle attributes $\{a_1, \ldots, a_L\}$.
- V2V communication:
  - The vehicle generates an epoch pseudonym as a domain specific pseudonym using epoch ϵ as the domain.
  - To broadcast a message m, the vehicle encrypts one of its attribute a_i as m' using pk_{in} and broadcasts m, m', and the proof of knowledge of the pseudonym.
  - A successful verification of a received proof of knowledge guarantees that (*a*) the message is from a registered vehicle, (*b*) the pseudonym is valid for the current epoch, and (*c*) a_i is properly encrypted for inspection.
  - If verifier wants to report a misbehaving vehicle, then she sends m' to the CA (the *inspector*). The CA decrypts m' and obtain the hidden attribute a_i, which can be linked to the reported vehicle.

A. Our model implementation with Idemix

We leverage the idemix protocol [11] to develop a non-interactive authentication scheme for VANETs and to achieve the goals listed in Section III-A. In this section, the aforementioned stages are explained in further detail following the Idemix protocol.

1) Group and system parameters: are published by the SA. The SA picks two random numbers $g' \in_R \mathbb{Z}^*_\Gamma$ and $r \in_R [0 \ldots \rho]$, computes the generators g and h, where $g = (g')^b (\mod \Gamma)$ and $h = g^r$. It then publishes the group parameters $\{\Gamma, \rho, g, h\}$. The commitment group $\mathbb{Z}^*_\Gamma$ with order $\Gamma - 1 = \rho.b$ for some large prime ρ, where b is a cofactor smaller than ρ preferably. The order $\Gamma - 1$ ensures that $\mathbb{Z}^*_\Gamma$ has a large subgroup of prime order ρ and that the discrete logarithms are hard to compute [11]. The bit lengths of Γ and ρ are given by l_Γ and l_ρ respectively.

2) CA generates the key pair for ABCs: the CA initiates the key generation step from CL-signature scheme [7]. Its output is the public-key pair $\{pk_I, sk_I\}$ that is used for issuing certificates, i.e., signed lists of user attributes $\{a_0, \ldots, a_L\}$.

3) CA generates a key pair for verifiable encryption: the CA decrypts an encrypted attribute of misbehaving vehicles. To generate the inspection key pk_{in}, the CA runs the key generation algorithm from the CS-signature scheme [8]. The outputs are public-key pair $\{pk_{in}, sk_{in}\}$. Each country has a unique inspection key.

4) Vehicle registration: a vehicle generates its private key k_u, which is chosen uniformly at random from the interval $[1, \rho]$, using the SA's public parameters.

To obtain an ABC on its list of attributes $\{a_0, \ldots, a_L\}$, a vehicle interacts with the CA, which runs the CL signing algorithm [7]. Its outputs are the ABC $C_{CA}(a_0, \ldots, a_L)$ and the inspection public key pk_{in}, which are sent to the vehicle in a secure communication channel. The CA also stores the vehicle's attributes $\{a_1, \ldots, a_L\}$

5) V2V communication: requires a system wide epoch, which is a time interval ϵ. Once an epoch ϵ_i is over, it is

Algorithm 1: Send a ZKP over message m (PK)

Input: $\{\epsilon, C_{CA}, \text{m}\}$
if *new* ϵ_i *is starting* **then**
  Generate $N_{(\epsilon_i)}$
while ϵ_i *is not finished* **do**
  encrypt a hidden attribute a_i, $m' = [a_i]$ $a_i \in A_h$
  calculate PK $\{a_0 \mid C_{CA} \wedge N_{(\epsilon_i)} \wedge m'\}(m, m')$
  send PK to nearby vehicles

Algorithm 2: Verify a ZKP over message m (PK)

Input: PK_{v_1}
if $\epsilon + 1$ *started* **then**
  Discard all saved pseudonyms
if PK_{v_1} *received* **then**
  verify PK_{v_1}
  if *verification == True* **then**
    Read m
    N_ϵ is valid
    m' is valid
    Save N_ϵ
  else if *verification == False* **then**
    discard m
else
  Wait

immediately followed by the next epoch ϵ_{i+1}. The algorithm for generating and sending a PK is shown in Algorithm 1.

- A vehicle generates its domain pseudonym N_ϵ for epoch ϵ. The pseudonym $N_\epsilon = N(a_0, \epsilon) = g_\epsilon^{a_0}$, where g_ϵ is the hash of ϵ, i.e., $g_\epsilon = H(\epsilon)^{(\Gamma-1)/\rho} \mod \Gamma$, and H is a hash function mapping $\{0,1\}^* \rightarrow Z_\Gamma$ [11].
- To broadcast a message m, a vehicle (prover) needs to convince the verifier that it:
  - possesses an ABC issued by the CA while not revealing its attributes,
  - can encrypt a hidden attribute,
  - can generate a signature on m to demonstrate possession of a valid secret that is certified by the CA.
- The sender encrypts a hidden attribute a_i as $m' = [a_i]$ using the CS encryption algorithm [8], and generates the ZKPK of $\{ABC, N_\epsilon, m'\}$, which outputs PK : $\{a_0 | C_{CA}(a_0, .., a_L) \wedge N_\epsilon \wedge m'\}(m, m')$. PK is then sent to the neighbouring vehicle recipients over the V2V network.
- A recipient verifies PK using the public key of the CA non-interactively, i.e., without the need to contact the CA, see algorithm 2. If the CL verification algorithm returns $True$, the recipient is convinced that:
  - the message m is from a registered vehicle,
  - the pseudonym N_ϵ is valid, for the current epoch ϵ,
  - m' is an encrypted hidden attribute.

At the end of the the epoch, the receiver will discard all pseudonyms.

6) Deanonymization: A verifier may report a vehicle by sending m' to the CA. The CA may proceed to deanonymize the vehicle using CS decryption algorithm with m' as input. The output is the hidden attribute $m = a_1$.

V. Security, privacy and performance evaluation

In this section, we look into the security and privacy system goals defined in Section III-A and evaluate the implementation of our model against them. We provide a computational performance evaluation of our model based on the number of exponentiation operations required to perform the algorithms involved in V2V communication (Section IV-A5).

A. Security and privacy evaluation

The protocol leverages the CL signature scheme [7]. The unforgeability of the signatures hold under strong RSA assumption and the computational DH assumption [11]. We also leverage non-interactive ZKPs using the Fiat-Shamir heuristic which are also semantically secure [15].

- *Authenticated broadcasting:* the PK only passes verification if it was generated by a prover with a valid ABC.
- *Conditional anonymity:* is realized by encrypting a hidden attribute a_i to a ciphertext m', which is appended to every message sent by the *prover*. Every message sent by the prover has a different m' appended to it, as the prover picks a new random number for every message it sends. A *prover* can be de-anonymized if the verifier forwards the prover's m' to the CA, which then decrypts it. The output is the hidden attribute a_i, which de-anonymizes the prover.
- *Impersonation:* the CA issues an ABC for vehicle v_1 only if all the presented attributes are valid and belong to v_1. So, a vehicle v_2 cannot get a certificate for a_1 that belongs to v_1. Therefore, v_2 cannot impersonate v_1.
- *Non-repudiation:* m' is the encryption of the certified hidden attribute a_1 and it is appended to a message sent by prover v_1. The CA can decrypt m' and obtain a_1 which is linked to v_1. Hence, v_1 cannot deny being the sender of that message.
- *Offline verification:* a vehicle can verify a received PK by itself using the CA's public key, i.e., without contacting a third party.
- *Linkability within epochs:* when a prover sends multiple messages in an epoch ϵ, i.e., using the same N_ϵ, a verifier will link all messages to the same N_ϵ.
- *Unlinkability across epochs:* at the beginning of each epoch ϵ_i a prover generates a new N_{ϵ_i}. Messages sent in different epochs ϵ_1 and ϵ_2 are unlinkable because $N_{\epsilon_1} \neq N_{\epsilon_2}$.
- *Sybil-free:* our construction prevents Sybil attacks in the following two ways:
  - because of the nature of the registration procedure, a single vehicle has exactly one OBU and since the CA has to verify the vehicle's attributes before generating

an ABC to the vehicle. A vehicle can never register more than one instance of itself.
  - a prover cannot generate multiple N_ϵ during the same epoch since they are bound to ϵ and to the prover's secret key a_0, where ϵ changes only when the epoch is finished and a_0 is constant.
- *Replay attacks:* this scheme is susceptible to replay attacks within epochs. This is typical for most protocols that do not keep a state between senders and receivers. It could be mitigated with timestamps shared between senders and receivers.

B. Computational performance evaluation

Our protocol is based on *idemix* which is based on CL signatures. Therefore, the security of our protocol relies on the security of the CL signatures. CL signatures are based on RSA and its security relies on the modulus n of length l_n. $l_n = 2048$ is still secure [24]. The rest of the security parameters are $l_k = 160$, $l_e = l_k + 2$ and $l_v > l_n + l_k + l$ where l is a security parameter. We chose $l_v = 3072$.

The computational performance of our scheme can be estimated by the number of exponentiation operations used to send and receive messages:

- *Sending a PK* requires 15 exponentiation operations:
  - seven to prove that ABC C_{CA} is valid,
  - two for generating the domain pseudonym N_ϵ,
  - six for encrypting a hidden attribute a_1.
- *Verifying a received proof PK* requires 10 exponentiation operations:
  - eight for verifying C_{CA},
  - two to verify the pseudonym.

Multiplication and addition operations are disregarded because their computational complexity is low when compared to exponentiations. Also, the *issuing* and *inspection* protocols are not considered in the evaluation because they are performed by the CA.

We simulated the exponentiation calculation using GMP library[1] in c programming language on a *good* specs virtual machine running Ubuntu $20.04.4$ @ 4.0 GHZ Intel $i7-6700(8)$, 16 GB RAM. We replicated the same experiments using a *bad* specs machine having Ubuntu $20.04.4$ @ 2.3 GHZ Intel Xeon, 2 GB RAM . We repeated the experiment 1000 times. The average of each operation is shown in tables I and II. In table I, we can see that the average time taken to send a message is 7.15 ms using the *good* machine and it increases to 17.93 ms when using the *bad* machine. However, note that since the pseudonym generation operations is done only once during the beginning of an epoch so in most cases it would take 5.84 ms and 14.41 ms to send a PK. Table II, shows the total time to verify a PK, the time varies between 5.82 ms and 14.02 ms between the *good* and the *bad* machine.

We compare between a good machine and a bad machine to show that the operations are machine dependant and we see no reason why in the near future faster OBUs will be produced which will also reduce the current timings.

[1]https://gmplib.org/

TABLE I: Average total time to send a PK, and the breakdown of computation times for different operations, using a good and a poor VM (in ms).

Send PK	CL-sig	CS-enc	Pseud-Gen	Total
Good VM	3.95	1.89	1.31	7.15
Poor VM	9.19	5.22	3.52	17.93

TABLE II: Average total time to verify a PK, and the breakdown of computation times for different operations, using a good and a poor VM (in ms).

Verify PK	CL-sig	Pseud-Ver	Total
Good VM	4.51	1.31	5.82
Poor VM	10.5	3.52	14.02

VI. Related work

In this section we discuss privacy-preserving schemes that have been proposed in the literature aimed at VANETs. We selected the most relevant recent work that can be better compared to ours. For a more comprehensive overview of security and privacy in VANETs we refer to [22].

Camenisch et al. [5] propose a novel zone encryption scheme that allows vehicles to exchange encrypted messages. In their scheme they divide earth into zones where in each zone vehicles have to agree on a symmetric key that is used to encrypt messages. Encrypted messages makes tracking a lot harder by preventing an outsider from mapping messages to a specific vehicle and the encryption keys are different for each zone. Vehicles still need to authenticate themselves and for that they request short-term credential from the issuer. The credential is valid for an epoch and at the end of the epoch vehicles need to contact the issuer to get a new credential. Their system provide high level of privacy but is vulnerable to non-repudiation and cannot protect against Sybil attacks [5].

In [16], they propose a system where every time a registered vehicle enters a new RSU location, the vehicle first communicates with the corresponding RSU by generating a randomized token based on its secret key to request a pseudo identity that will be used to communicate with other vehicles in the vicinity. However, there is nothing preventing vehicles from generating as many randomized tokens as they want making this scheme vulnerable to Sybil attacks.

Verheul et al. [27] propose IFAL, a system where pseudonym certificates are pre-loaded in a vehicle for its entire life. Certificates are encrypted and can only be decrypted with an activation code that is sent to the vehicle via SMS before each epoch which allows the vehicle to derive the pseudonym. The same key can be used for an entire epoch to derive pseudonyms. CRLs are not needed in their system because if a vehicle misbehaves they will stop sending the activation codes which prevents that vehicle from deriving new valid pseudonyms. However, a misbehaving vehicle will be able to keep using the system for an entire epoch (which is 90 days).

2FLIP [29] is a privacy-preserving authentication scheme in which drivers are required to authenticate themselves with a biometric scheme (fingerprint). Vehicles generate a new pseudonym for every transmitted message. 2FLIP uses

TABLE III: Comparison between our proposed scheme and the related work.

Properties \ Paper	Our scheme	[27]	[5]	[29]	[16]
Authenticated broadcasting	✓	✓	✓	✓	✓
Conditional anonymity	✓	✓	✓	✓	✓
Impersonation	✓	✓	✓	✓	✓
Non-repudiation	✓	✓	-	✓	✓
Offline verification	✓	✓	✓	✓	✓
Linkability within epochs	✓	✓	✓	-	-
Unlinkability across epochs	✓	✓	✓	✓	✓
Sybil free	✓	✓	-	-	-
Self-pseudonym generation	✓	-	-	✓	-
Encrypted messages	-	-	✓	-	-
No RSU reliance	✓	-	✓	-	-
No CRL reliance	✓	✓	-	-	-

transaction pseudonyms [23] for sending messages, i.e., a new unlinkable pseudonym is produced for every message transmitted. However, they are not ideal for a VANET setting as pseudonyms should be linkable for at least a short period of time. Controlled linkability allows participating vehicles to better perceive their surrounding environment. 2FLIP is also vulnerable to Sybil attacks [1].

A comparison in the privacy and security properties offered by our scheme and the related work is shown in Table III and shows that our scheme is the only one that provides all security and privacy properties without relying on RSUs or CRLs. In our scheme, differently from [5], broadcasted messages are not encrypted. It is arguable if VANET message exchanges need to be encrypted, as they have a public utility value (road safety). In our understanding, it is the identity of those participating in the VANET that should be preserved, and not necessarily the secrecy of the message.

VII. Discussion

In this section, we review the flexibility of our model, define misbehaving, discuss revocation and provide details about how our system is Sybil resistant.

1) Flexibility of our model: we're aware that in our system we require a lot of trust in the CA since it is able to decrypt vehicles credentials and link them to the vehicle but keep in mind that the CA is not monitoring all the communication in the VANET but it only receives a portion of reported misbehaving vehicles. However, our model could be extended to include a separate entity or entities such as, *inspector* which can be responsible for de-anonymizing vehicles instead of the CA. The inspection key could be divided between multiple organizations where at least two of them need to come together to join their key parts and perform inspection [28]. The system could also be extended to give *provers* the ability to prove possession of a known certified attribute and the ability to share that attribute. For example, certain areas only allow vehicles with specific properties to enter and for that a *prover* could share their attributes (electric vehicle or a vehicle that uses a specific type of fuel).

2) Misbehaving vehicles: misbehaving vehicles in our systems are vehicles that don't abide by traffic laws (passing a red light, speeding, etc) or try to manipulate the system by sending an excessive amount of messages in a short time to nearby vehicles in order to try and reroute others to another road while keeping the empty roads for themselves. A misbehaving vehicle will not have its credentials revoked but instead will be de-anonymized by the CA in order to receive a fine.

3) Revocation: in the literature, general reasons have been suggested to revoke a vehicle such as: vehicle robbery, misbehaving and breaking the laws and vehicle destruction (end of life) [5], [27]. Previous research has stressed the importance of revocation and has proposed measures for revocation. However, we believe that revocation is not very necessary because if a vehicle is stolen, the owner reports the theft and she won't be held responsible for future misbehaving. Just like what happens now in real life, if you don't stop at a red light (misbehave), you don't get your vehicle taken away from you but instead you receive a fine. Our system uses the same technique, the CA identifies and fines the misbehaving vehicle when it breaks the law, as for vehicle destruction, we also don't see a reason to revoke that credential because it will simply die with the vehicle and for this, procedures need to be set to make sure that the OBU of the vehicle is also properly destroyed with it. If the attributes of the vehicles change then the owner needs to contact the CA which will issue a new ABC on the new attributes.

4) Sybil resistance: to prevent Sybil attacks and provide the privacy-preserving benefits of pseudonyms at the same time, the pseudonyms need to be limited to only one per vehicle at any epoch. This is possible either by restricting the issuing of pseudonyms or by providing means to detect multiple pseudonyms belonging to a same vehicle. Restricting the issuing of pseudonyms is the better approach, as it completely avoids the problem of detection, which is present in some Sybil-free pseudonym approaches [3], [20]. The proposal from Khodaei et al. [17] falls into this category as it is an interactive protocol: pseudonyms are sent to a trusted third party, which is considered to be always available, so that they can be verified. Our proposal does not have the aforementioned problems because it limits the generating of pseudonyms to only one per vehicle by binding it to the current epoch and the vehicle's private key. Therefore, only one pseudonyms can be generated per epoch and verifiers can check the validity and Sybil-freeness of the received pseudonyms without interacting with third parties.

VIII. Conclusions and Future Work

We leveraged idemix to develop a non-interactive, privacy-preserving, Sybil-free authentication scheme in VANETs.

Our security and privacy evaluation shows that we managed to achieve the goals mentioned in Section III-A: (*a*) *Authenticated broadcasting* allows receivers to be sure that a received message is from a vehicle that has a valid ABC. (*b*) *Self-pseudonym generation* allows vehicles to generate their own pseudonyms without interacting with third parties. (*c*) *Conditional anonymity*, so misbehaving vehicles can be deanonymized by the CA. (*d*) No *impersonation*, so vehicles cannot assume the identities of other vehicles. (*e*)

Non-repudiation prevents a vehicle from denying being the sender of a message that it had sent. (*f*) *Offline verification* gives vehicles the ability to verify received messages without interacting with third parties. (*g*) *Linkability within epochs* to link messages sent by a same vehicle within an epoch. (*h*) *Unlinkability across epochs* so messages sent in different epochs by a same vehicle cannot be linked. (*i*) Our analysis also proves that our protocol is resilient against *Sybil attacks*.

For future work, we plan improve our protocol design so it provide the same security and privacy properties with a less computational effort while providing means to protect against replay attacks. We will provide a formal verification of our system protocols. In addition, we are going to investigate other foundational ABC schemes, such as U-Prove and ABC4Trust constructions, and how could they be implemented in VANETs.

Acknowledgment

This work was supported by the Swedish Foundation for Strategic Research (SSF) SURPRISE (Secure and private connectivity in smart environments) project.

References

[1] Akil, M., Islami, L., Fischer-Hübner, S., Martucci, L.A., Zuccato, A.: Privacy-preserving identifiers for iot: a systematic literature review. IEEE Access **8**, 168470–168485 (2020)

[2] Alpár, G., Jacobs, B.: Credential design in attribute-based identity management (2013)

[3] Andersson, C., Kohlweiss, M., Martucci, L.A., Panchenko, A.: A Self-Certified and Sybil-Free Framework for Secure Digital Identity Domain Buildup. In: Information Security Theory and Practices: Smart Devices, Convergence and Next Generation Networks, Proc. of the 2^{nd} IFIP WG 11.2 Int. Workshop (WISTP 2008). LNCS 5019, Springer (2008)

[4] Baldini, G., Hernández-Ramos, J.L., Steri, G., Matheu, S.N.: Zone keys trust management in vehicular networks based on blockchain. In: 2019 Global IoT Summit (GIoTS). pp. 1–6. IEEE (2019)

[5] Camenisch, J., Drijvers, M., Lehmann, A., Neven, G., Towa, P.: Zone encryption with anonymous authentication for v2v communication. In: 2020 IEEE EuroS&P. IEEE (2020)

[6] Camenisch, J., Dubovitskaya, M., Lehmann, A., Neven, G., Paquin, C., Preiss, F.S.: Concepts and languages for privacy-preserving attribute-based authentication. In: IFIP Working Conf. on Policies and Research in Identity Management. pp. 34–52. Springer (2013)

[7] Camenisch, J., Lysyanskaya, A.: A signature scheme with efficient protocols. In: Int. Conf. on Security in Communication Networks. pp. 268–289. Springer (2002)

[8] Camenisch, J., Shoup, V.: Practical verifiable encryption and decryption of discrete logarithms. In: Annual Int. Cryptology Conf. pp. 126–144. Springer (2003)

[9] Camenisch, J., Stadler, M.: Efficient group signature schemes for large groups. In: Annual Int. Cryptology Conf.pp. 410–424. Springer (1997)

[10] Camenisch, J., Van Herreweghen, E.: Design and implementation of the idemix anonymous credential system. In: Proc. of the 9th ACM Conf. on Computer and Communications Security (2002)

[11] Camenisch, J., et al.: Specification of the identity mixer cryptographic library, version 2.3. 1, December 7, 2010

[12] Chaum, D.: Security without identification: Transaction systems to make big brother obsolete. Communications of the ACM **28**(10) (1985)

[13] Douceur, J.R.: The sybil attack. In: Int. workshop on peer-to-peer systems. pp. 251–260. Springer (2002)

[14] Farradyne, P.: Vehicle infrastructure integration (vii)-architecture and functional requirements. Draft Version **1** (2005)

[15] Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Conf. on the theory and application of cryptographic techniques. pp. 186–194. Springer (1986)

[16] Gayathri, N., Thumbur, G., Reddy, P.V., Rahman, M.Z.U.: Efficient pairing-free certificateless authentication scheme with batch verification for vehicular ad-hoc networks. IEEE Access **6**, 31808–31819 (2018)

[17] Khodaei, M., Noroozi, H., Papadimitratos, P.: Scaling pseudonymous authentication for large mobile systems. In: WISEC (2019)

[18] Khodaei, M., Papadimitratos, P.: Efficient, scalable, and resilient vehicle-centric certificate revocation list distribution in vanets. In: WISEC (2018)

[19] Lee, M., Atkison, T.: Vanet applications: Past, present, and future. Vehicular Communications **28**, 100310 (2021)

[20] Martucci, L.A., Kohlweiss, M., Andersson, C., Panchenko, A.: Self-certified sybil-free pseudonyms. In: Proc. of the first ACM conference on Wireless network security. pp. 154–159 (2008)

[21] Paquin, C., Zaverucha, G.: U-prove cryptographic specification v1. 1. Technical Report, Microsoft Corporation (2011)

[22] Petit, J., Schaub, F., Feiri, M., Kargl, F.: Pseudonym schemes in vehicular networks: A survey. IEEE communications surveys & tutorials **17**(1), 228–255 (2014)

[23] Pfitzmann, A., Hansen, M.: A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management (2010)

[24] PUB, F.: Digital signature standard (dss). FIPS PUB (2019), https://doi.org/10.6028/NIST.FIPS.186-5-draft

[25] Schnorr, C.P.: Efficient signature generation by smart cards. Journal of cryptology **4**(3), 161–174 (1991)

[26] Serna, J., Morales, R., Medina, M., Luna, J.: Trustworthy communications in vehicular ad hoc networks. In: IEEE WF-IoT. IEEE (2014)

[27] Verheul, E., Hicks, C., Garcia, F.D.: Ifal: Issue first activate later certificates for v2x. In: 2019 IEEE European Symposium on Security and Privacy (EuroS&P). pp. 279–293. IEEE (2019)

[28] Veseli, F., Serna, J.: Evaluation of privacy-abc technologies-a study on the computational efficiency. In: IFIP International Conference on Trust Management. pp. 63–78. Springer (2016)

[29] Wang, F., Xu, Y., Zhang, H., Zhang, Y., Zhu, L.: 2flip: A two-factor lightweight privacy-preserving authentication scheme for vanet. IEEE Transactions on Vehicular Technology **65**(2), 896–911 (2015)