Certificateless and Lightweight Authentication Scheme for Vehicular Communication Networks

Waleed Hathal, Haitham Cruickshank, Zhili Sun, and Carsten Maple

***Abstract*—Reducing the number of road accidents is a key agenda item for governments across the world. This has led to an increase in the amount of attention given to Vehicular Communication Systems (VCS), which are seen as an important technology that can offer significant improvements in road safety. Using VCS, vehicles can form a dynamic self-configuring network that enables a vehicle to communicate with other vehicles (V2V) and roadside infrastructure (V2I). However, such wireless communication channels are vulnerable to attacks, and therefore an authentication scheme for communications should be designed before the deployment. Prior work has focused on utilising digital signature approaches to achieve the security requirements, but due to the special characteristics of VCS, such approaches are not well suited for safety related applications of VCS, since they incur high communication and computation overheads. To combat this issue, we propose a certificateless and lightweight authentication scheme to provide means of secure communications for VCS. In this work we introduce authentication tokens, which replace digital certificates to reduce the burden of certificate management on a Trusted Authority (TA). In addition, the utilisation of tokens ensures that mutual authentication is achieved for V2I communication. Moreover, we employ TESLA as the underlying broadcast authentication protocol to achieve the required security goals for safety message broadcasting. According to the security analysis and extensive simulation of our scheme, the results show that it can withstands various types of attacks. Also it has better performance in term of verification delay, scalability and communication overhead compared to lightweight authentication schemes that are based on similar techniques. Therefore, the scheme is well suited for VCS.**

***Index Terms*—Authentication tokens, BAN logic, schnorr signature, TESLA, vehicular communication systems (VCS).**

I. Introduction

THE rapid advancement of mobile and wireless communication technologies has accelerated the realisation of Vehicular Communication Systems (VCS), which is an important network platform for Intelligent Transportation Systems (ITS). VCS has attracted significant attention of governments across the world, to enable cooperative and automated mobility. Applications that are becoming increasingly prevalent including road safety, such as lane merging and alerts for traversing intersections, to value-added services, such as navigation, toll payment services and internet access [1]–[4]. Vehicles are increasingly being equipped with multiple sensors to collect and process different data to be shared with other vehicles and road infrastructures, hence allowing enhanced safety and comfort for road users. The so-called Internet of Vehicles (IoV) is gathering momentum and will increase the number of vehicles that will form part of VCS [5].

Vehicles will be equipped with a wireless device, known as an On-Board Unit (OBU), to allow a vehicle to exchange traffic related messages with its peers and infrastructures. Exchanging safety related messages periodically allows vehicles to be aware of their surroundings, hence road safety can be achieved. The US standards specify a Basic Safety Message (BSM), where ETSI standards specify Cooperative Awareness Message (CAM) [6]. In which these safety messages incorporate information about a vehicle's status such as speed, location and direction collected by the sensors equipped in the vehicle. Communications between vehicles and infrastructures may be based on the Dedicated Short Range Communication (DSRC) protocol, which specifies that a vehicle should broadcast BSM every 100-300 [7], [8].

Although VCS provides a platform for a variety of applications that can help reduce road accidents and improve driving experience, there are a number of challenges that require addressing prior to their deployment. Wireless communication networks are exposed to various attacks, such as message modification, deletion or replay attacks which can ultimately lead to traffic disruption or accidents [9]. Therefore, an authentication framework is imperative to provide receiving vehicles with security primitives such as, source authentication, integrity, non-repudiation and confidentiality. Since OBUs are expected to receive high volume of messages, a secure authentication scheme should satisfy the aforementioned security requirements and should be efficient and scalable for a large number of requests [10], [11].

Authentication, integrity and non-repudiation can be achieved by utilising Public Key Infrastructure (PKI), in which a digital signature algorithm is used to sign and verify messages. Although PKI-based schemes can achieve the security requirements and may offer a high security level, but such schemes still suffer from high communication overhead due to the size of signature and certificate [12]–[15]. In addition, signing as well as verifying messages, invokes a computation overhead, which is not suitable for many safety applications with a high

Manuscript received March 12, 2020; revised July 13, 2020 and September 24, 2020; accepted November 3, 2020. Date of publication December 3, 2020; date of current version January 22, 2021. This work was supported by the PETRAS Internet of Things Research Hub. The review of this article was coordinated by Prof. Z. Zheng. *(Corresponding author: Waleed Hathal.)*

Waleed Hathal, Haitham Cruickshank, and Zhili Sun are with the Institute of Communication Systems, University of Surrey, GU2 7XH Guildford, U.K. (e-mail: waleed.hathal@surrey.ac.uk; H.cruickshank@surrey.ac.uk; Z.Sun@surrey.ac.uk).

Carsten Maple is with the WMG department, University of Warwick, CV4 7AL Coventry, U.K. (e-mail: cm@warwick.ac.uk).

Digital Object Identifier 10.1109/TVT.2020.3042431

volume of messages. One of the most problematic concerns is Certificate Revocation List (CRL); CRLs can become sizeable and their distribution is non-trivial in vehicular ad-hoc networks. Therefore, they can have a high communication overhead, which can lead to a high impact on packet-loss ratio. To tackle the CRL issues other researches have focused on utilising identity-based batch verification or aggregate signature techniques instead of traditional PKI [16]–[18]. Furthermore, due to the expected high volume of messages many existing researches have proposed lightweight authentication schemes to overcome the high computational and communication overheads of public key cryptography [19]–[23]. In [19], a lightweight authentication scheme for vehicular networks is proposed, which employs Identity Based Signature (IBS) based on the standard Rivest–Shamir–Adleman (RSA) to ensure message authentication. It also achieves privacy preserving for vehicles by allowing Roadside Unites (RSUs) to convert a vehicle's signature to Trusted Authority's (TA) signature, while the TA can retrieve the real identity of that vehicle. However, Zhang *et al.* [20] showed that the scheme in [19] has a security defect, which enables attackers to launch a common modulus attack to reveal a vehicle's private key. Moreover, Cui, *et al.* [21] introduced a message authentication scheme based on edge computing concept to reduce the computation load on the vehicles' side. In which RSUs can authenticate messages of nearby vehicles and broadcast the verification results to all vehicles to reduce the redundancy of message verification on vehicles. Although the proposed scheme can effectively reduce the computation on the vehicles' side, but according to [22] it suffers from the following attacks; man-in-the-middle, impersonation and concatenation attacks. In addition, symmetric key cryptography has been exploited to provide lightweight authentication for VCS. Timed Efficient Stream Loss-tolerant Authentication (TESLA) has been used in [23], [24] along with Elliptic Curve Digital Signature Algorithm (ECDSA) to provide source authentication and non-repudiation respectively. However, the former scheme is not applicable for time sensitive applications due to the message buffering which introduces longer verification time. As for the latter scheme, it suffers from high communication overhead due to the extra elements that are added in a message to overcome the buffering issue of TESLA.

Unfortunately, safety oriented applications cannot tolerate high communication and computation complexity in vehicular networks. Therefore, to tackle these issues we propose a secure certificateless authentication scheme, which provides a secure Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communications. The main contributions of this paper are summarised as follows:

1) Firstly, authentication tokens are introduced in our scheme instead of digital certificates to reduce the burden of certificate management on TA. In addition, the utilisation of authentication tokens allow vehicles to establish secure and efficient mutual authentication with the TA before joining the network.
2) Secondly, to ensure source authentication for periodically broadcasted BSM in a timely fashion, TESLA is utilised as the underlying broadcast authentication scheme. Also, since TESLA does not support non-repudiation, we linked the authentication tokens with the TESLA key for each vehicle to ensure non-repudiation is provided. In addition, Schnorr signature algorithm is used to validate the authentication tokens.
3) Finally, since TESLA is being used as the underlying authentication scheme and does not support instant message verification we take advantage of a vehicle's past trajectory to construct a table of future movement prediction. Also, to ensure that such a method of providing instant authentication does not effect the efficiency of the proposed scheme, we utilise the Chinese Remainder Theorem (CRT) to obtain a single value for all possible future movement.

The rest of the paper is organised as follows: Section II presents related work on authentication schemes in VCS. The problem definition, system model and security objectives are presented in Section III. Section IV presents the underlying cryptographic tools that are used in this paper. The proposed scheme is presented in Section V. The security analysis of the proposed scheme is presented in Section VI. Performance evaluation is provided in Section VII. Finally, we conclude the paper in Section VIII.

II. Related Work

Various techniques have been studied in recent years to ensure authentication for VCS. In [25] Raya and Hubaux proposed a scheme based on PKI, where vehicles store a large number of anonymous certificates with the corresponding public and private key pairs to allow vehicles to use a key pair to sign BSM for a period of time. Although the scheme achieves authentication and anonymity but it suffers from high storage overhead due to the large number of certificates, it incurs high verification delay and the certificate revocation list (CRL) can be a bottleneck if large vehicles are revoked. To overcome the storage overhead issue of [25], the authors of [14], [26] proposed a group signature schemes, which allows vehicles to obtain a temporary anonymous certificate when it passes by an RSU. However, CRL distribution and checking is still an issue. Therefore, the authors of [7], [27] presented authentication schemes where they have replaces the CRL with HMAC to achieve message integrity and avoid the long process CRL checking.

Furthermore, public-key encryption is a widely utilised technique in VCS that can provide secure and confidential communications. An end-to-end authentication scheme is proposed in [28]. It is based on Elliptic Curve encryption to ensure all the transmitted data between vehicles are encrypted and can only be access by authorised entities. Also, the authors have used sandboxing technique as an extra layer of security to prevent intrusion for in-vehicle security for downloaded services. Kanchan *et al.* [29] proposed a privacy-preserving scheme called SAPSC, which utilises cloud computing for group communications. They have adopted signcryption method, which allows messages to be signed and encrypted simultaneously. Moreover, in [30], [31] Homomorphic encryption has been used to provide confidentiality and privacy, where the former scheme uses Paillier Homomorphic encryption and one-way hash function to generate

pseud-identity for vehicles. Also, the authors have introduced a decentralised mutual identity authentication by allowing RSUs to verify vehicles' pseud-identities instead of on cloud. Whereas the latter scheme adopts Boneh-Lynn-Shacham (BLS) group signature to ensure non-repudiation along with Homomorphic encryption to ensure confidentiality and integrity. Another group key management scheme is proposed in [32], where the authors have combined prime factorisation, CRT, the use of the noise parameter and discrete logarithm to develop asymmetric group key management. Although, the schemes [28]–[32] achieve the security requirements, they still suffer from high computation due to the adoption of expensive cryptographic operations such as bilinear pairing. Also, the certificate management and revocation is still an issue in these schemes.

Other studies have adopted the identity-based signature to address the problem of certificate management. In [33] the authors proposed an ID-based scheme called (SPECS), which satisfy the privacy requirements. Also they employed Bloom filter and binary search technique to achieve lower message overheads and better success rate. Nonetheless, Horng *et al.* [34] pointed that SPECS can be exposed to impersonation attacks, hence, they proposed an improved ID-based scheme, which support batch authentication to speed up the verification process. Moreover, the authors of [35] proposed an identity-based authentication scheme, which adopts cuckoo filter and binary search technique to obtain high success rate for the batch authentication. In [36], the authors proposed an ID-based authentication scheme using proxy vehicles to enable RSUs to simultaneously verify a large number of signatures with minimal overheads. The authors of [37] proposed an attribute-based framework based on attribute-based signature (ABS) to ensure message authentication and privacy are achieved. The idea of designing such a scheme is to conceal vehicles' identities at a lower cost of communication and computation overheads. Furthermore, Luo and Ma in [38] proposed a multi-authority efficient access scheme for vehicular cloud computing, where ciphertext-policy attribute-based encryption (CP-ABE) is utilised to ensure access for legitimate vehicles and revocation mechanism. Also, their scheme can prevent static corruption of authorities.

Significant studies have been investigating the use of lightweight broadcast authentication schemes for VCS. Studer *et al.* [24] proposed an authentication scheme, which combines ECDSA with TESLA to provide non-repudiation and message authentication respectively. However, the scheme does not providing instant message verification, which is not suitable to be used for safety oriented applications in VCS. Similarly, Lyu *et al.* [23] presented a scheme that combines TESLA and ECDSA to ensure non-repudiation and message authentication. They, proposed to predict a vehicle's future position to provide instant message verification. However, the scheme suffers from high communication overhead, due to the added leaf values of the Merkle Hash Tree (MHT) in BSM. Ying *et al.* [39] proposed an anonymous and lightweight Authentication based on Smart Card (ASC) scheme for VCS, where they utilise low-cost cryptographic operations to authenticate source of messages. However, this scheme suffers from high computation cost at the TA side, which make it vulnerable to DoS attacks.

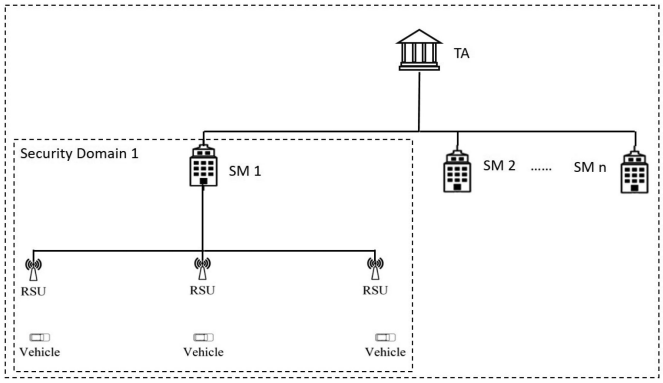

Fig. 1. System Model.

Moreover, in [40], [41] the authors have focused on designing a privacy preserving certificateless aggregation signature scheme to reduce the communication overhead by reducing the size of the signature. However, both schemes did not consider V2V communications, which is an important part since vehicles have less processing powers compared to RSUs. Du *et al.* [42] and Kamil *et al.* [43] pointed out that the proposed scheme by [40] is not secure against signature forgery.

III. Problem Definition

In this section, we define the system model, problem statement, threat model and security requirements.

A. Problem Statement

As per the DSRC standard, each vehicle should broadcast BSM every 100-300 ms to its peers and infrastructure; each message should be signed and verified using ECDSA according to IEEE Std 1609.2-2016. Due to the special characteristics of VCS, the use of ECDSA is not seen as compatible with the safety-oriented applications. Since ECDSA requires significant processing power to verify messages, it is not compatible for time sensitive applications (e.g. safety applications). Furthermore, the use of ECDSA implies that a digital certificate should be attached to each packet that is broadcasted, thereby allowing receivers to verify legitimate senders. As a result, this can cause high communication overhead, which in turn, leads to an increase in the packet loss rate. As a result, designing an efficient lightweight authentication scheme for time sensitive applications is critical, to ensure secure communications while maintaining low communication and computation overheads.

B. System Model and Assumptions

Fig. 1 demonstrates the system model of the proposed scheme, which consists of TA, SM, RSUs and OBUs. Details of all network entities and assumptions are described as follows:

1) TA: The TA is assumed to be equipped with sufficient computation and storage capabilities. It is responsible for registering and generating credentials for SMs and OBUs in the network. In addition, it generates Primary Tokens (PT), which are encrypted using its master key. The PT

are generated for registered OBUs, to be used for the initial authentication phase (when joining the network). The TA is assumed to be fully trusted by all the network entities. Having an anchor point of trust in the system to issue credentials for all entities in the network is critical. Therefore, the assumption of TA being trusted.

2) SM: The TA divides the whole precinct into several domains, where each domain is managed by a SM. The purpose behind introducing SMs is to reduce the burden on the TA by splitting the load on the SMs. In addition, a SM is responsible for maintaining all of the RSUs in its domain. Furthermore, SMs are responsible for generating and updating Secondary Tokens (ST) for in domain vehicles during the initial authentication process. Also, a SM has secure connection with the TA, RSUs and other SMs through the use of transport layer security (TLS). Moreover, we assume that it has sufficient computational and storage capabilities. Since SMs are deployed and maintained by the TA regularly, they are assumed to be secured and trusted.[1]
3) RSU: The RSUs are deployed at roadsides, where they act as a bridge between OBUs and the core network. They also broadcast road information to all OBUs within the communication range. Furthermore, the RSUs are equipped with sufficient computational and storage capabilities.
4) OBU: An OBU is a radio device that is fitted in the vehicles, which provides a means of communication amongst vehicles, and between vehicles and RSUs. The OBUs have limited computational and storage capabilities compared to RSUs. In addition, OBUs have a Tamper-Proof Device (TPD), which is used to store all the credentials such as Identity, PT, ST and TESLA keys.

C. Threat Model

Internal and external Adversaries are two types of adversaries in VCS, where an external adversary is considered to be powerful and can monitor and analyse the traffic in the network. As an external adversary is not part of the system, he/she cannot decrypt messages. Furthermore, for the whole network to be observed and analysed that needs multiple colluding of external adversaries. On the other hand, an internal adversary is a compromised vehicle. In addition, since an internal adversary is a part of the network, he/she considered to be potent.

As the wireless medium is considered to be insecure we present all the possible attacks. An adversary can (a) modify or replay messages, (b) delete or inject false messages, (c) impersonate a legitimate entity of the network, (d) block future messages to prevent authentication, (e) eavesdropping and (f) perform a Denial-of-Service (DoS) attack. Therefore, the aforementioned attacks are prevented in this study.

[1]The assumption of trusted SMs is beneficial in terms of reducing the burden on the TA. It is also worth mentioning that other works [44], [45] have considered it.

D. Security Requirements

Since the volume of broadcasted messages in VCS is expected to be very high, it is crucial to design an authentication mechanism, which can ensure authentication in a timely fashion with low communication and computation overheads. In this paper, we list the main security goals for the authentication scheme as follows:

1) Authenticity and integrity: Receivers should be able to validate the origin of a message and verify that it was sent by a legitimate vehicle. Further, receivers should be able to validate that the content of a message has not been modified by unauthorised party.
2) Non-repudiation: This attribute is used to ensure that a sender is assured of the message delivery and a receiver is assured that the message was sent by a legitimate sender. Therefore, both parties cannot deny sending or receiving the message. In addition, this prevents an illegitimate vehicle from claiming to be another vehicle.
3) Confidentiality: It allows two parties to share a secret through an insecure channel, while preventing any unauthorised entity from knowing the shared secret. Although confidentiality is not a requirement for BSM, but in our scheme we encrypt the PT to keep a vehicle's data confidential.
4) Resistant to various attacks: These include impersonation attacks, message modification attack, replay attacks, blocking messages and broadcasting false messages to other vehicles. Also, it is important that the scheme can resist DoS attacks.
5) Low communication and computation overheads: In order to achieve higher success rate of message authentication and lower message delay, the message verification phase should be lightweight and efficient with low security overheads. Therefore, maintaining low computation overhead can prevent computational-based DoS attacks.

IV. Preliminaries

In this section, we introduce some basic knowledge related to the fundamentals of our proposed scheme; these are hash chain, TESLA authentication scheme, CRT and the Schnorr signature algorithm. The notations used throughout this paper are listed in Table I.

A. Hash Function, Hash Chain and HMAC

A one-way hash function is considered to be secured if and only if the below properties are fulfilled [46]:

- $h(\cdot)$ takes a message of an arbitrary length as an input and outputs a message digest of a fixed-length.
- Given x, it is easy to compute $h(x) = y$, but it is hard to compute $h^{-1}(y) = x$, when given y.
- Given x, it is computationally infeasible to find $x' \neq x$, such that $h(x') = h(x)$.

Fig. 2 defines a hash chain where $S_k = h(S_{k-1}), k = 1, 2, \cdots, i$ and $S_0 = SD$, where SD is the initial seed value. As per the

TABLE I
DEFINITION OF NOTATIONS

Notations	Descriptions
PT	primary token
ST	secondary token
SM_i	ith security manager
RSU_i	ith road-side unit
SD	seed
SK_x, PK_x	private key and corresponding public key of entity x
S_x	master secret key of entity x
V_i	ith vehicle
CK	commitment key (first TESLA key on the chain)
$Sign(SK_x, M)$	Signing a message M using Schnorr signature with secret key SK_x
$Verify(PK_x, M, \sigma_{x,M})$	Verifying the Schnorr signature $\sigma_{x,M}$ of a message M with public key PK_x
$\{PT\}_{S_{TA}}$	PT encrypted with the master secret key of TA
C_x	challenge generated by entity x
X	CRT output for prediction table entries
H	a collision free one-way hash function
K_{ST}	secret key used for secondary token
K_{KU}	secondary token key update
TS	timestamp
$\|\|$	message concatenation

Fig. 2. Hash Chain.

hash function definition, given S_k it is computationally feasible to compute S_{k+1} but it is infeasible to compute S_{k-1}.

HMACs are keyed hash functions used to provide source authentication and message integrity involving a cryptographic hash function and a secret cryptographic key. Examples of hash functions include the SHA family leading to HMACs such as HMAC-SHA1, HMAC-SHA512/224 and HMAC-SHA3.

B. TESLA Broadcast Authentication Protocol

One of the challenges of broadcast authentication is securing transmitted data and allowing all receivers to ensure that a message was sent by a legitimate sender and has not been modified. TESLA scheme is a broadcast authentication protocol that is based on symmetric cryptography. It employs one-way hash chains to generate private keys to ensure source authentication [47]. In addition, since it utilises hash functions, it is efficient in terms of communication and computation overheads. Although, TESLA is based on symmetric cryptography, it can achieve asymmetric properties with the help of delayed disclosure of keys. Moreover, it can tolerate arbitrary packet loss due to its lightweight operation.

Fig. 2 shows how a sender can generate their private keys starting with a seed value SD and using a hash function (H) repeatedly to generate the previous keys. The first key of the chain ($SD = S_0$) serves as the commitment key to the entire chain, which allows receivers to authenticate the future values on the chain. Furthermore, TESLA uses a second hash function (H') to generate keys that are used for computing Message Authentication Codes (MAC). We use TESLA in our scheme to authenticating periodic safety messages exchanged between vehicles.

C. Chinese Remainder Theorem (CRT)

CRT is a theorem of number theory, which has been used extensively in cryptographic algorithms such as RSA, to reduce the computation overhead [48]. CRT can be used to achieve data protection and can be designed as a one-way limitation. It states the following: let $m_1, m_2, \ldots, m_k$ be pairwise co-prime positive integers, where the Greatest Common Divisor (GCD) $(m_i, m_j) = 1$ and $i \neq j$. Let $n_1, n_2, \ldots, n_k$ be arbitrary sequence of integers then the CRT defines the congruent equations as follows:

$$\begin{aligned} x &\equiv n_1 \mod m_1 \\ x &\equiv n_2 \mod m_2 \\ &\vdots \\ x &\equiv n_k \mod m_k \end{aligned} \tag{1}$$

Then x has a unique solution: $x = \sum_{n=1}^{k} b_i\, M_i\, y_i \mod M$, where, $1 \leq i \leq k$

$$M = \prod_{i=1}^{k} m_i M_i = \frac{M}{m} \tag{2}$$

$$y_i = M_i^{-1} \, mod \, m_i \tag{3}$$

The m_i values are considered to be the moduli of the CRT and x is the solution of the CRT problem. As a consequence of the CRT, any positive integer $N < n$ can be represented as a k-tuple, $n_1, n_2, \ldots, n_k$ and vice versa. The N value will be non-deterministic in case if there is less than a k-tuple. The use of CRT in our work is to generate a single verification value for the predicted movement of a vehicle, and to provide instant authentication for safety messages since TESLA alone cannot provide instant message authentication.

D. Schnorr Signature Algorithm

We have adopted the Schnorr signature algorithm [49] as the underlying signature algorithm to sign and verify STs in our scheme. It is known for its efficiency in terms of communication and computation overheads. Also, it is provably secure in the random oracle model. Assuming SM_i whose public key is PK_{SM_i} and private key is SK_{SM_i}, where $PK_{SM_i} = p^{SK_{SM_i}}$ and $SK_{SM_i} \in \mathbb{Z}_q^*$. Let the signing procedure of a message M by SM_i be denoted as $\sigma_{SM_i,M} = Sign(SK_{SM_i}, M)$. Whereas the verification procedure by other entities (receivers) be denoted as $Verify(PK_{SM_i}, M, \sigma_{SM_i,M})$.

V. PROPOSED AUTHENTICATION SCHEME

In this section, we describe our proposed authentication scheme, which consists of five phases: 1) System initialisation; 2) registration; 3) initial authentication (vehicles joining the network); 4) message signing and 5) message verification. During the system initialisation phase the TA generates its public and

private key pair and other public parameters. The registration phase is intended to assign each entity in the network a unique identity and generate a PT for each vehicle. Prior to accessing the network a vehicle must authenticate itself to the TA using the PT obtained in the registration phase. Additionally, a ST will be generated by a SM for each vehicle to reduce the communication and computation burden on TA for future authentications. Once a vehicle is authenticated with the TA, it can sign and verify BSM using TESLA keys.

A. System Initialisation Phase

The initial parameters are issued by the TA for the whole system, and these parameters can be updated regularly by the TA if the master key is believed to be compromised, or the TA wants to enhance the system security level. Since we adopt Schnorr signature algorithm in our scheme, the TA has to do the following steps to generate its public/private key pair and other parameters:

1) primes p and q such that $q|p-1$, $q \geq 2^{140}$ and $p \geq 2^{512}$;
2) $\alpha \in \mathbb{Z}_p$ with order q, i.e., $\alpha^q = 1 \pmod p$ and $\alpha \neq 1$;
3) a one-way hash function $h{:}(0,1)^* \rightarrow (0,1)^l$;
4) choose a random number r as the TA's private key $r \in \mathbb{Z}_q^*$ so that $SK_{TA} = r$ and the TA computes its public key as $PK_{TA} = p^{SK_{TA}}$. Also, TA generates a master secret key (S_{TA}) to be used for encrypting/decrypting PTs. Then it publishes the tuple $(p, q, \alpha, h, PK_{TA})$ to all network entities as the system parameters.

B. Registration Phase

SM registration: The TA performs the following steps to assign a unique identity and generates key pair for SM_i in security domain i.

1) TA obtains the private key of SM_i by randomly choosing a number $SK_{SM_i} \in \mathbb{Z}_q^*$ and computes the public key $PK_{SM_i} = p^{SK_{SM_i}}$.
2) The generated identity and public key for SM_i is then signed by the private key of TA $\sigma_{SK_{TA},M} = Sign(SK_{TA}, PK_{SM_i}||SM_i)$.
3) TA sends the credentials $SK_{SM_i}||\sigma_{SK_{TA},M}$ to SM_i through a secure channel. As per the assumed model, there exists a secure channel between the TA and SMs.

RSU registration: Each SM is responsible for generating and updating the security credentials for its local RSUs. SM_i undertakes the following steps to generate key pair and identity for each RSU. For RSU_i, this is:

1) SM_i obtains the private key of RSU_i by randomly choosing a number $SK_{RSU_i} \in \mathbb{Z}_q^*$ and computes the public key $PK_{RSU_i} = p^{SK_{RSU_i}}$.
2) The generated identity and public key for RSU_i is then signed by the private key of SM_i $\sigma_{SK_{SM_i},M} = Sign(SK_{SM_i}, PK_{RSU_i}||RSU_i)$.
3) SM_i sends the credentials $SK_{RSU_i}||\sigma_{SK_{SM_i},M}$ to RSU_i through a secure channel.

Vehicle registration: The vehicle registration is conducted offline during the vehicle inspection or manufacturing and it is the responsibility of the TA. Each vehicle is equipped with a TPD, which is used to store credentials and performs the cryptographic operations. The TA assigns an identity to V_i and uploads the credentials as follows.

1) TA generates a random secret key (S_{V_i}) to be shared only between the TA and V_i.
2) TA creates a PT, which includes the identity, secret key of V_i and the expiration time of the PT.
3) TA uses its master secret key S_{TA} to encrypt the PT and uploads it in the TPD along with the S_{V_i} and the identity of V_i.

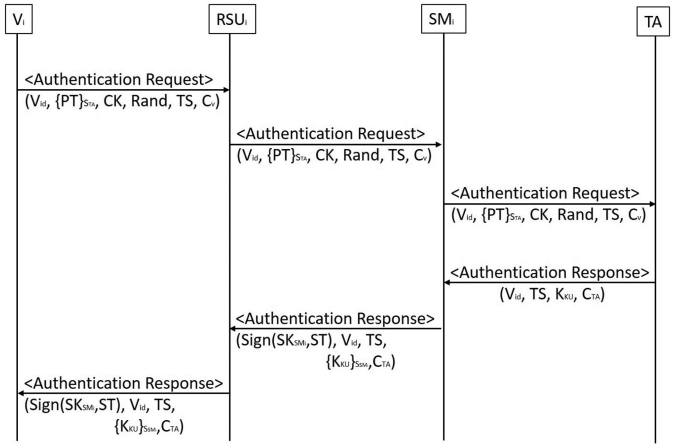

Fig. 3. Initial authentication phase flow.

C. Initial Authentication Phase

Prior to accessing the network a vehicle must perform the initial authentication phase with the TA to be able to join the network. When a vehicle V_i joins a domain it sends an authentication request message to the TA through the first RSU it meets. Fig. 3 describes the message flow of the initial authentication handshake that takes place between vehicles and the TA. First V_i sends an authentication message request containing: $(V_{id}, \{PT\}_{S_{TA}}, CK, Rand, TS, C_v)$ where V_{id} is the vehicle ID, $\{PT\}_{S_{TA}}$ is the encrypted PT by TA, CK is the commitment key of the key chain that was generated by V_i (TESLA keys), $Rand$ is a fresh nonce generated by V_i, TS is the time stamp and C_v is a generated challenge by V_i using its secret key (S_{V_i}) that was obtained from the TA. The request challenge C_v is computed by V_i as $HMAC(S_{V_i}, TS||Rand||CK)$. Upon receiving the authentication request RSU_i forwards the message to SM_i, where it will store V_{id} and CK and forward the request to TA. Once the TA receives the authentication request message for V_i it will authenticate it as described in Algorithm 1.

After authenticating V_i, the TA will send an authentication response message to SM_i containing: $(V_{id}, TS, K_{KU}, C_{TA})$ where V_{id} is the vehicle ID, TS is the time stamp, K_{KU} is the key to be used by V_i to update its ST and C_{TA} is the challenge response calculated by the TA, using the same nonce that was generated by V_i. When SM_i receives the authentication response message from TA it then generates and signs a ST for V_i as shown in Algorithm 2. The response message, which includes the signed ST is then forwarded to V_i through RSU_i.

Algorithm 1: The Process of TA Verifying Authentication Request From V_i.

Require: $\{PT\}_{S_{TA}}, TS, CK, Rand, C_v$
1: TA decrypts $\{PT\}_{S_{TA}}$ using its master secret key.
2: Get V_i's secret key and PT expiry date.
3: **if** PT is not expired **then**
4: Verify C_v by recomputing the challenge request $C'_v = HMAC(S_{V_i}, TS||Rand||CK)$
5: **if** $C_v = C'_v$ **then**
6: Generates K_{KU} for V_i to update its ST by computing $HMAC(S_{V_i}, Rand)$.
7: TA Computes the challenge response $C_{TA} = HMAC(S_{V_i}, TS||Rand||K_{KU})$.
8: **end if**
9: **end if**

Algorithm 2: The Process of Generating ST for V_i by SM_i.

Require: $\sigma_{SM_i,ST} = Sign(SK_{SM_i}, ST), K_{KU}$
1: SM_i generates the ST for V_i by computing $K_{ST} = H(H(K_{KU}))$.
2: Generate ST for V_i, which includes $(V_{id}, K_{ST}, CK, Exp)$
3: ST is signed by the secret key of SM_i, SK_{SM_i} as follows:
4: Choose uniformly at random $0 \leq k < r$,
5: Compute $S_0 = p^k$, (p is an element of prime order r),
6: Compute $S_1 = H(ST||S_0)$,
7: Compute $S_2 = k + SK_{SM_i}S_1 \bmod r$,
8: The signature of ST is (S_1, S_2).
9: K_{KU} is encrypted by the master secret key of SM_i to be used for ST update $\{K_{KU}\}_{S_{SM_i}}$.

Upon receiving the response message V_i verifies the message as described in Algorithm 3. After authenticating the generated ST, V_i can communicate with other vehicles and RSUs using the self generated TESLA keys.

D. Message Signing

A vehicle can broadcast BSM to its peers and RSUs once it has been authenticated by the TA and obtained a ST from the local SM. Since TESLA is being used as the underlying authentication scheme and does not support instant message verification, we must overcome this limitation. Therefore, we take advantage of a vehicle's past trajectory to construct a prediction table by modelling all the possible future movements every two consecutive messages, such as M_{i-1} and M_i as shown in Fig. 4. The idea of using a vehicle's future movements to provide instant authentication with TESLA is inspired by [23]. To predict future movements we use a local coordinate, which is placed at the beginning position ($\vec{P_0}$) of the time frame. Moreover, a pair of perpendicular vectors ($\vec{x}$ and $\vec{y}$) are used to set the accuracy of position prediction. Hence, a future position can be expressed as $\vec{P_i} = \vec{P_0} + a_i\vec{x} + b_i\vec{y}$ and the movement

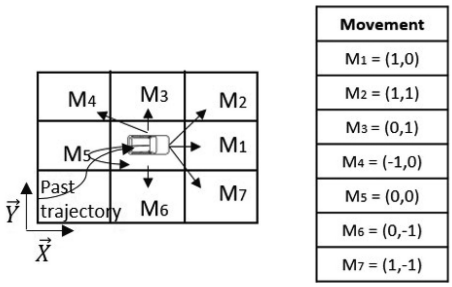

Fig. 4. Prediction table construction.

between two intervals such, I_{i-1} to I_i can be expressed as $\vec{M_i} = \vec{P_i} - \vec{P_{i-1}} = (a_i - a_{i-1})\vec{x} + (b_i - b_{i-1})\vec{y}$.

Once a prediction table is constructed (as shown in Fig. 4) V_i calculates a single value (X) to tie all of the movement entries (M_k) using the CRT. The CRT produces a unique solution (X) to simultaneous linear congruences as shown in (4).

$$\begin{aligned} X &\equiv H(I_i||T_i||M_1||Nonce) \mod n_1 \\ X &\equiv H(I_i||T_i||M_2||Nonce) \mod n_2 \\ &\vdots \\ X &\equiv H(I_i||T_i||M_7||Nonce) \mod n_7 \end{aligned} \tag{4}$$

As shown in 4 for an entry M_k (from prediction table Fig. 4) in PT_i. The arbitrary integers of the congruence equations are labelled as $H(I_i||T_i||M_k||Nonce)$, where the nonce is used to prevent message forgery. For sake of clarity each congruent equation of the CRT in 4 represents a single entry from the prediction table.

Once V_i calculates X_1 value for the next Beacon (B_1), it can broadcast the first beacon (B_0) in its time frame. B_0 contains the following: $(m_0, HMAC(K_T, m_0), HMAC(K_T, X_1), ST(S_1, S_2), K_0)$, where $(m_0 = T_0, I_0, K_0, H(X_1), \vec{P_0}, \vec{x}, \vec{y})$. The attachment of signed ST helps receivers to ensure (K_0) corresponds to V_i, hence non-repudiation is provided.

To generate a signature for future beacons e.g. (B_i), V_i chooses K_i (from TESLA key chain) for interval I_i. Then it performs the steps of constructing prediction table and calculating a single value using CRT to get X_{i+1}. Therefore, B_i includes the following: $(m_i, HMAC(K_i, m_i), HMAC(K_i, X_{i+1}), X_i, K_{i-1})$, where $m_i = T_i, I_i, K_{i-1}, X_i$.[2] Fig. 5 shows all the elements a vehicle attaches to beacons at different intervals.

E. Message Verification

Receivers cannot authenticate B_0 instantly due to the delayed key disclosure (K_T)[3]. However, receivers should authenticate the attached ST and verify that CK of the sending vehicle is corresponding to the ST. If CK is validated then the HMAC of m_0 and X_1 should be stored until next interval. Upon receiving

[2] B_1 includes K_0 (CK of TESLA) and K_T to authenticate B_0
[3] K_T is computed by V_i as $H(K_{KU})$.

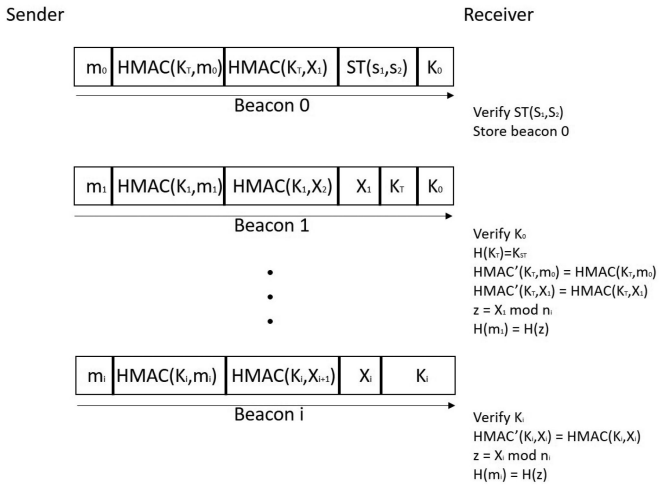

Fig. 5. Description of transmitted BSM and how verification is carried out.

Algorithm 3: The Process of V_i Verifying Authentication Response.

Require: $TS, K_{KU}, ST, Rand, C_{TA}, \sigma_{SM_i,ST} = Sign(SK_{SM_i}, ST)$
1: V_i computes $K_{KU} = HMAC(S_{V_i}, Rand)$ using its secret key.
2: V_i verifies C_{TA} by computing $C'_{TA} = HMAC(S_{V_i}, TS||Rand||K_{KU})$.
3: **if** $C_{TA} = C'_{TA}$ **then**
4: K_{KU} is stored in V_i's TPD to be used when updating ST.
5: V_i generates and stores K_{ST} for BSM broadcasting.
6: Using the public key of SM_i, V_i verifies ST $Verify(PK_{SM_i}, ST, \sigma_{SM_i,ST})$).
7: **if** ST is verified **then**
8: V_i can communicate with other vehicles.
9: **end if**
10: **end if**

B_1 each vehicle should verify the attached K_T by computing the hash value $H(K_T)$ and compare it with the K_{ST}, which is included in the sender's ST. If K_T is valid then B_0 can be authenticated as $HMAC(K_T, m_0)$ and if it is matching the HMAC of m_0 stored then it is verified. Moreover, m_1 can be instantly verified by computing the $HMAC(K_T, X_1)$ and if it matches the stored value then receivers can compute modulus as $z = X_1 \bmod n_i$. Finally, the hash value of z should match the hash of m_1 given that the message was not modified.

To validate future beacons ($B_i, i > 1$), receivers can compute the CK by using the current key K_i. If the key is valid and fresh then by computing $HMAC(K_i, X_1)$ then calculating the $z = X_i \bmod n_i$ and finally hashing and comparing $H(z)$ and $H(m_i)$. If both values are identical then m_i is valid. Otherwise, receivers should store m_i and its HMAC value along with K_i until next interval.

F. Secondary Token Update Procedure

The secondary token update procedure is performed between a vehicle V_i and SM_i before the current ST of V_i expires. The following steps illustrate how an ST update is conducted:

1) Before sending an update request to SM_i, V_i needs to generate a key chain (TESLA keys) and a random number. Then, it computes a challenge $C_v = HMAC(K_{KU}, TS||Rand||CK)$. The update request message includes $\{V_{id}, TS, Rand, CK, C_v, \{K_{KU}\}_{S_{SM_i}}\}$.
2) Upon receiving the update request, SM_i uses its master secret key S_{SM_i} to decrypt the key update K_{KU}, that was generated by the TA in the initial authentication phase. The SM_i then verifies C_v by recomputing the challenge $C'_v = HMAC(K_{KU}, TS||Rand||CK)$. If $C_v = C'_v$, then SM_i generates a key K_{ST} by computing $H(H(K_{KU}||Rand))$ for the new ST. Finally, the SM_i constructs a new ST for V_i and signs it using its private key using Schnorr signature.
3) Once V_i obtains the response from SM_i, it computes the new K_{ST} and stores it, and verifies the signed ST.

The frequency of updating the ST depends on the need of each vehicle, as a vehicle can obtain multiple STs at a single time. Moreover, the key update K_{KU} for each vehicle is valid for a certain duration. The duration for the key update is decided by the SM_i.

VI. Security Analysis

In this section, we analyse the security of our proposed scheme. Since the signature algorithm of the proposed authentication scheme is based on Schnorr signature [49], it was proved to be secure in the random oracle model as long as the Discrete Logarithmic Problem (DLP) is hard to be solved [50]. Definition 1 below presents the mathematical problem used to analyse the security of Schnorr signature.

Definition VI.1: DLP: DLP Consider an element $x = g^s \bmod p$, where $x \in G$. It is easy to calculate x given p and s, but it is hard to determine s given x and p.

Therefore, our analysis are based on the widely accepted BAN logic to formally proof that our scheme achieves mutual authentication [51]. Furthermore, we carry out informal analysis to show that the scheme meets the security requirements described in Section III.

A. Formal Security Analysis

Table II shows the notations used in the BAN-logic. Below are some of the BAN-logical postulates, which are important for validating our scheme:

1) The message-meaning rule: $\frac{P|\equiv Q \leftrightarrow^k P, P \lhd \{X\}_k}{P|\equiv Q|\sim X}$, $\frac{P|\equiv Q \leftrightharpoons^k P, P \lhd \langle X \rangle_k}{P|\equiv Q|\sim X}$.
2) The freshness-conjuncatenation rule: $\frac{P|\equiv \#(X)}{P|\equiv \#(X,Y)}$.
3) The nonce-verification rule: $\frac{P|\equiv \#(X), P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$.
4) The jurisdiction rule: $\frac{P|\equiv Q \Rightarrow X, P|\equiv Q|\equiv X}{P|\equiv X}$, $\frac{P|\equiv (X,Y)}{P|\equiv X}$, $\frac{P \lhd (X,Y)}{P \lhd X}$, $\frac{P|\equiv Q|\sim (X,Y)}{P|\equiv Q|\sim X}$.

TABLE II
BAN LOGIC NOTATIONS AND MEANINGS

Notations	Description
P $\|\equiv X$	Principal P believes a statement X
P $\triangleleft X$	Principal P sees statement X
P $\Rightarrow X$	Principal P has jurisdiction over statement X
#(X)	Formula X is fresh
$\{$ X $\}_K$	Formula X encrypted under the key K
P $\overset{k}{\leftrightarrow}$ Q	P and Q share the key K to communicate
$\overset{k}{\mapsto} P$	Principal P has k as a public key (the matching secret key is denoted k^{-1})
P $\overset{X}{\Longleftrightarrow}$ Q	Formula X is secret known only to P and Q
P $\|\sim X$	Principal P once said the statement X

Before performing the BAN-logic proof on our scheme we first need to detail the assumptions, goals and idealised forms. The assumptions listed below are apparent and necessary for validating our scheme.

- A_1: $V_i \mid\equiv TA \Rightarrow PT$
- A_2: $V_i \mid\equiv V_i \leftrightarrow^{S_{V_i}} TA$
- A_3: $TA \mid\equiv \#(Rand, TS, CK)$
- A_4: $SM_i \mid\equiv \#(TS)$
- A_5: $TA \mid\equiv TA \leftrightarrow^{S_{V_i}} V_i$
- A_6: $SM_i \mid\equiv TA \mid\sim K_{KU}$
- A_7: $V_i \mid\equiv \quad \mapsto PK_{SM_i} SM_i$
- A_8: $V_i \mid\equiv \#(TS)$
- A_9: $V_i \mid\equiv \#(Rand, TS, CK)$

Next we set our goals as follows:

- G_1: $TA \mid\equiv C_v$
- G_2: $V_i \mid\equiv ST$
- G_3: $V_i \mid\equiv V_i \leftrightarrow^{K_{KU}} SM_i$

Since the authentication request message is transparently forwarded by RSU_i and SM_i they are not included in the BAN analysis. Also, the authentication response is transparently forwarded by RSU_i, hence it is not included as well. The idealised message sequences of our proposed protocol are as follows:

Message (m_1) $V_i \rightarrow TA$:
$(V_{id}, TS, Rand, CK, \langle TS, Rand, CK\rangle_{S_{V_i}},$
$\{V_{id}, Exp, S_{V_i}\}_{K_{S_{TA}}})$
Message $(m_2) TA \rightarrow SM_i$:
$(V_{id}, TS, K_{KU}, C_{TA})$
Message $(m_3) SM_i \rightarrow V_i$:
$(V_{id}, TS, \langle ST\rangle_{Sign_{SK_{SM_i}}}, \{K_{KU}\}_{S_{SM_i}}, C_{TA})$

Below we analyse the idealised form of the proposed authentication scheme based on the logical postulates of BAN logic and the assumptions we made above. The main procedures of proof are as follows:

From m_1 we could show:
$TA \triangleleft V_{id}, TS, Rand, CK, \langle TS, Rand, CK\rangle_{S_{V_i}},$
$\{V_{id}, Exp, S_{V_i}\}_{K_{S_{TA}}})$.
Based on A_1 the jurisdiction rule, we can show:
$TA \triangleleft \{V_{id}, Exp, S_{V_i}\}_{K_{S_{TA}}}$.
From A_5 and the message-meaning rule, we can prove:
$TA \mid\equiv V_i \mid\sim \langle TA, Rand, CK\rangle_{S_{V_i}}$.
From A_3 and the freshness-conjuncatenation rule, the following can be shown:
$TA \mid\equiv \#(Rand, TS, CK)$
Based on $TA \mid\equiv V_i \mid\sim (TA, Rand, CK)$ and the nonce-verification rule, we can show:
$TA \mid\equiv C_v$ (G_1).

As it was mentioned in section III.B, messages exchanged between entities within the core network are secured using TLS. Hence, from m_2 we can show:
$(SM_i \triangleleft (V_{id}, TS, K_{KU}, C_{TA})$
Based on A_6 the assumption that m_2 is securely transmitted by TA to SM_i using TLS, the following can be shown:
$SM_i \mid\equiv TA \mid\sim (V_{id}, TS, K_{KU}, C_{TA})$
From A_4 and the freshness-conjuncatenation rule, the following can be shown:
$SM_i \mid\equiv \#(TS)$
From m_3 we can show:
$V_i \triangleleft (V_i, TS, \langle ST\rangle_{Sign_{SK_{SM_i}}}, \{K_{KU}\}_{S_{SM_i}}, C_{TA})$
From A_7 and the jurisdiction rule, we can show:
$V_i \triangleleft (\langle ST\rangle_{Sign_{SK_{SM_i}}})$
From A_7 and the message-meaning rule, we can prove:
$V_i \mid\equiv SM_i \mid\sim \langle ST\rangle_{Sign_{SK_{SM_i}}}$.
Based on the previous prove $(V_i \mid\equiv SM_i \mid\sim \langle ST\rangle_{Sign_{SK_{SM_i}}})$, we can show:
$V_i \mid\equiv ST$ (G_2)
From A_8 and the freshness-conjuncatenation rule, the following can be shown:
$V_i \mid\equiv \#(TS)$
From A_2 we can show that:
$V_i \mid\equiv TA \mid\sim \langle C_{TA}\rangle_{S_{V_i}}$
Therefore, from A_9 V_i can compute K_{KU}. Thus, we can proof:
$V_i \mid\equiv TA \mid\sim K_{KU}$.
Hence, the following can be proven:
$V_i \mid\equiv V_i \leftrightarrow^{K_{KU}} SM_i$ (G_3)

After successfully proofing all of the mentioned goals $G_1 - G_3$ using the widely accepted BAN logic, we can say that the proposed scheme can achieve mutual authentication between vehicles and the core network entities.

Furthermore, since our scheme relies heavily on the cryptographic hash and HMAC functions to validate the authenticity of BSM, we rely on previous assertions that these cryptographic functions are secure. As TESLA does not support instant authentication we make use of the movement prediction to generate a HMAC which is broadcast before sending the beacon message, thereby, allowing receivers to authenticate messages instantly. However, if broadcasting the HMAC of the movements prediction is not secure then our technique is fundamentally flawed and cannot be assumed secure. Below we show that sending the HMAC before the beacon is secure. We note that the authors of [47] have previously proved the security of TESLA scheme.

Theorem 1: If the underlying HMAC algorithms and hash chain are secure, our scheme provides a negligible probability that an attacker could forge a legitimately authenticated message in the context of VANETs, independent of the attacker's computational capability.

In order to prove theorem 1, we use the following lemma.

Lemma 1: Assuming that both the key chain and the HMAC algorithms are secure, then broadcasting the HMAC of a movement prediction is secure.

Proof: Based on the broadcasted HMAC of the movement prediction, the attackers' aim is to generate bogus messages and impersonate a legitimate vehicle. In order to successfully achieve this aim, attackers may attempt the following attacks.

Firstly, an attacker might try to discover a different movement prediction value $\hat{X_{i+1}}$, which results in obtaining the same HMAC as the original $X_{i+1} : HMAC(K', X_{i+1}) = HMAC(K', \hat{X_{i+1}})$. However, if an attacker is able to generate such an outcome, then it shows that the utilised HMAC function is not secure under the adaptive chosen message attack. Secondly, an attacker might try to generate a valid message and HMAC by obtaining undisclosed TESLA key before it is sent by the legitimate vehicle. However, to successfully find such a key an attacker should defeat the one-way property of the hash chain, which is computationally infeasible. ■

B. Informal Security Analysis

In this subsection, we informally show that our scheme meets the security requirements described in Section III and can resist other known attacks.

Source authentication: During the initial authentication phase the TA can verify the source of the message by decrypting the PT $\{V_{id}, EXP, S_{V_i}\}_{S_{TA}}$ using its master secret key to obtain the secret key of the V_i S_{V_i} that was generated and stored in the V_i's TPD during the offline registration phase. Once the PT is decrypted, the TA computes $HMAC(S_{V_i}, TS||Rand||CK)$ and compares it with the challenge that was generated by V_i to verify that the message was generated by a legitimate user. Moreover, the validation of the message source in V2V communication can be efficiently verified in our scheme as follows: Upon receiving the first packet, receivers check the validity of the attached ST using the public key of SM_i. If it is valid then the packet is buffered until the next interval, at which time it will be possible to authenticate the source of the message. The K_T can be obtained from the second packet to compute the HMAC of the first packet and receivers can be assured that the message was sent by a legitimate vehicle. Since a secure one-way function is employed in our scheme the source authentication of the subsequent packets in a time frame can be achieved through the delayed key disclosure and the prediction value X. For instance, for interval I_i, receivers obtains the TESLA key K_i that is included in the packet to validate the source of the message by calculating the HMAC of the message and comparing it with the prediction value X that was in the previous beacon (B_{i-1}).

Message integrity and freshness: The integrity of a message can be ensured by validating the key K_i by following the one-way key chain, then computing the HMAC of m_i and comparing it with the stored HMAC of X_{i+1}. As mentioned before in the security proof subsection, each undisclosed keys cannot be obtained by an attacker. Hence, as long as the receiver checks the timestamp of the received message and ensure that it is fresh then message integrity is guaranteed.

Non-repudiation: Since ST includes $(V_{id}, CK$, generation time, expiry time), when a broadcast message is received, the receivers should verify the validity of current interval's key (K_i) by following the one-way key chain to CK. In the case that K_i is valid, then the property of non-repudiation is achieved since TESLA keys are only known to the sender and cannot be disclosed by any other entity, as proved in lemma 1. Also, it is important that the receivers check the timestamp of the message to be able to establish the freshness of the message and overcome the possibility of replay attacks. Since we utilise the Schnorr signature to sign ST, this helps receivers in authenticating the commitment key (CK).

Confidentiality: During the initial authentication phase it is important in the proposed scheme that the PT and K_T are kept confidential. Confidentiality of PT is achieved by encrypting it using the secret key of TA (S_{TA}), meaning no other entity has access to the PT except for TA. As for the temporary key K_T, which is used by a vehicle to allow receivers to authenticate the first packet in a time frame, this is encrypted by the vehicle's secret key and this is only known to the vehicle and the TA.

Resistance against various types of attacks: We show that our scheme can withstand the impersonation attack, message modification attack, block or broadcast bogus messages, replay attack and DoS attack as follows.

- Impersonation attack: For an adversary to impersonate an SM and generate a valid signature for a ST, the adversary should solve the DLP in order to generate a valid signature, since the Schnorr signature depends on the hardness of DLP. Furthermore, for an adversary to impersonate a vehicle he/she should defeat the property of the one-way key chain. Hence, as we proved in the aforementioned subsection that solving DLP and defeating one-way chain are computationally infeasible.
- Message modification attack: During interval (I_i), a sender generates the predicted movement (X_{i+1}) for next interval and computes the associated HMAC. Once receivers obtain the beacon message during interval I_{i+1}, they can compute the HMAC of the message using the TESLA key attached in the packet to compare and verify that the message contents have not been changed.
- Block or broadcast bogus messages: It is important that an authentication scheme resists packet loss. Since the underlying authentication scheme in our proposed work is based on TESLA, receivers can recover the key chain and validate messages. If an adversary blocks a beacon (B_i), receivers can still authenticate incoming beacons. However, as the movement prediction that was generated in (B_i) is not stored then receivers have to buffer the beacon (B_i) until the next key is disclosed. Also, an adversary may try to inject bogus messages in the network, but receivers only authenticate vehicles with a valid ST.
- Replay attack: For the initial authentication phase we prevent replay attacks by including a nonce and compute $HMAC(S_{V_i}, TS||Rand||CK)$ to allow the TA to check the validity and freshness of the message. For V2V communication, each sender should include a nonce in the movement prediction to prevent replay attacks.

TABLE III
SIMULATION PARAMETERS

Parameters	Value
Number of vehicles	20-550
Communication range of vehicles	300 m
Communication range of RSU	1000 m
Simulation time	60 s
MAC layer protocol	802.11p
Beacon interval	300 ms
Simulation area	1000m x 1000m
vehicle speed	0-25 m/s

VII. PERFORMANCE EVALUATION

As we focus on reducing the authentication overheads in VCS. In this section, we evaluate the authentication efficiency of the proposed scheme in terms of communication and computation overheads for both V2V and V2I communications. We have simulated a city scenario, where we conducted our simulation using NS-3. The simulation parameters are listed in Table III. To comply with the DSRC standards, we have set the interval for BSM dissemination every 300 ms. Furthermore, this work adopted SUMO for the mobility traces, where vehicles are randomly distributed on the roads. In addition, vehicles have random mobility in the simulated map with speed ranging from 0 to 25 m/s. We simulated our proposed scheme on our desktop machine equipped with an Intel Core i7, 16 GB RAM, and a display card Intel UHD Graphics 620.

A. Vehicle Initial Authentication Phase Performance

In this subsection we evaluate the performance of the initial authentication phase for vehicles when joining the network. We mainly focus on the communication and computation overheads as the performance metrics for this phase. Furthermore, we use Authentication based on Smart Card (ASC) scheme as a benchmark [39], because vehicles in ASC go through similar steps to join the network. Due to the fact that the wireless communication channel is a shared medium, it is important to maintain low communication overhead. Furthermore, since the TA is expected to have high volume of authentication requests from vehicles wishing to join the network, it is critical to retain a low computation cost for the TA while validating requests from all vehicles.

Communication overhead: Since the wireless communication channel is a shared medium we focus on the additional communication overhead caused by the attached security elements. Therefore, we only consider the overheads on the wireless channel and do not consider the wired link between the core network entities.[4] Without loss of generality, it is assumed that the size of the output of the AES encrypted data is 32 bytes, HMAC/hash are 20 bytes, the size of the Schnorr signature is 42 bytes, the size of timestamp is 4 bytes and the random number is 20 bytes. The initial authentication phase in both our scheme and ASC consists of two messages being exchanged between vehicles and the TA.[5] The security overhead of both messages (request and response) of our scheme are 76 and 86 bytes respectively. Therefore, the total communication overhead of our scheme is 162 bytes. On the other hand, the security overhead of authentication request and response in the ASC scheme are 64 bytes and 44 bytes respectively. Therefore, the total overhead of ASC scheme is 108 bytes. Fig. 6 shows the communication overhead of the initial authentication phase for both schemes with different number of requests. It can be seen that both schemes increase linearly when the number of requests increases. Compared to our scheme, ASC actually has lower communication overhead. As a result of attaching signature in the response message in our scheme, the communication overhead is slightly higher than ASC scheme. Therefore, the communication overhead of ASC is 66.67% of our scheme. Despite that the communication overhead of our scheme is higher than ASC, we show that our scheme achieves better computation cost at the expense of a slightly higher communication overhead, where the extra communication overhead does not effecting the performance of our scheme.

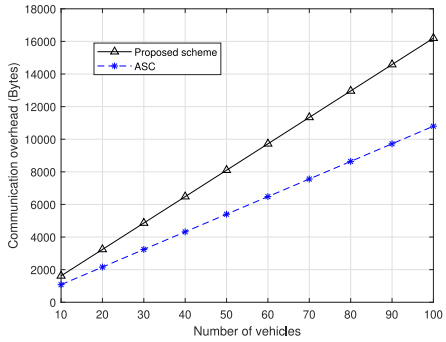

Fig. 6. Total communication overhead for the vehicle initial authentication phase against number of requests.

TABLE IV
COMPUTATION COST OF NETWORK ENTITIES FOR INITIAL AUTHENTICATION PHASE

Schemes	OBU	RSU	SM	TA
Our scheme	$2T_{mac} + T_h + T_{ver} + T_{e/d}$	–	T_s	$2T_{mac} + T_h + 2T_{e/d}$
ASC	$T_{ex} + 5T_h + T_{e/d}$	T_h	–	$T_{ex} + 5T_h + T_{e/d}$

Computation overhead: Table IV shows and compares the computation cost of each entity during the initial authentication phase in ASC and our scheme. According to our implementation the time of the cryptographic operations are; a hash function operation takes 0.013 ms, an HMAC operation takes 0.019 ms AES encryption/decryption operation takes 0.06 ms, modulo exponential operation takes 0.45 ms and Schnorr signature generation and verification operations are 1.57 ms and 3.113 ms respectively. Let the time complexity of a hash operation be T_h, HMAC operation be T_{mac}, AES encryption/decryption be T_e & T_d, Schnorr signature generation and verification be T_{sign} & T_{ver} and lastly modulo exponential operation be T_{ex}. As it

[4]Wired connection between core networks entities are not considered because the wired connection offers higher bandwidth compared to the wireless communication channel.

[5]Authentication request messages initiated by vehicles and authentication response messages sent by the TA.

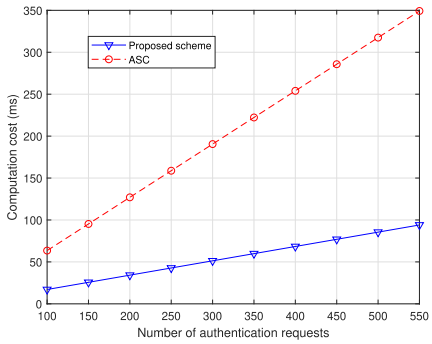

Fig. 7. TA computation cost during vehicle initial authentication phase.

can be seen from Table IV that an OBU in our scheme executes a hash function, 2 HMAC operations, an AES encryption and decryption and lastly Schnorr signature verification. Therefore, the total computation cost for an OBU is 3.284 ms. Whereas for a SM it generates and sign a secondary token, hence the computation cost is 1.57 ms. Lastly, the TA executes a hash function, 2 HMAC operations and an AES encryption and decryption, which results in 0.171 ms. Therefore, the total computation for the initial authentication phase of our scheme is 4.975 ms. On the other hand, in ASC an OBU has to execute 5 hash functions, an exponential operation and an AES encryption and decryption operations, which results in 0.635 ms. Whereas a RSU executes 1 hash operation, which is 0.013 ms. Finally, the TA in ASC executes the same cryptographic operations as an OBU. Hence, the total computation for the initial authentication phase of ASC scheme is 1.283 ms.

As the TA controls a large region, which implies that it is expected to receive a high volume of authentication requests. Therefore, Fig. 7 compares the TA computation cost of our scheme with ASC. It can be seen that the computation cost of TA in our scheme is 26.93% of ASC. That is a result of the network model presented in our scheme, where the TA assigns a SM to a domain. To reflect the efficiency of our scheme over ASC, if there were 50000 authentication requests it would cost 8.6 seconds and 31.8 seconds for our scheme and ASC respectively.

B. Secondary Token Updating Overhead

In this subsection, we present the communication and computation overheads for updating a secondary token by vehicles and SMs. The total communication overhead of the update procedure is 138 bytes, where the ST update request contains 96 bytes as the request message includes; $C_v, Rand, CK, TS$ and $\{K_{KU}\}_{S_{SM_i}}$. As for the response message, 42 bytes are utilised as the message sent by an SM includes a signed secondary token.

For the computation overhead a vehicle should compute a HMAC operation to generate a challenge request, hence the request cost is 0.019 ms. Upon receiving the request an SM should validate the challenge request and generate and sign a new ST. Therefore, the total cost of SM is 1.615 ms. Finally, a vehicle is required to execute 2 hash operations and verify the Schnorr signature. Hence, validating the response cost is 3.139 ms.

TABLE V
BSM COMMUNICATION OVERHEAD

Schemes	PBA	TESLA	VAST	ECDSA	Our scheme
Overhead(Byte)	160	40	145	181	80

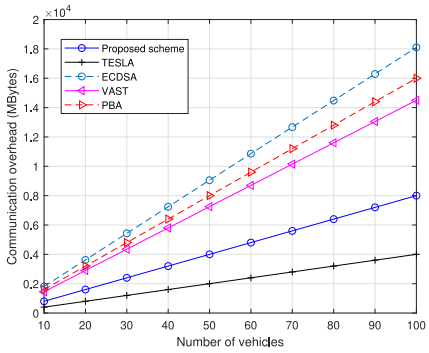

Fig. 8. Communication overhead of BSM against number of vehicles.

C. Periodic Safety-Related Message Broadcast Performance

In this subsection, we evaluate the authentication overhead of authenticating BSM exchanged between vehicles and RSUs. Our authentication analysis focuses on three elements: 1) communication overhead; 2) message signing cost; and 3) message verifying cost.

1) Communication Overhead: Table V shows the communication overhead comparisons of the schemes for a single BSM. The communication overhead presented in Table V are introduced by including the signature, certificate, MAC and symmetric key, while the message itself is not considered. In PBA scheme, a sender attaches two HMAC values of size 40 bytes, 20 bytes for TESLA symmetric key and five MHT leafs of size 20 bytes each to support instant authentication. Therefore, the total communication cost for their scheme is 160 bytes. In TESLA scheme, a sender only attaches a 20 bytes HMAC of the message and 20 bytes TESLA symmetric key, hence, the total communication overhead is 40 bytes for one message. While in VAST scheme, a certificate of size 63 bytes, 20 bytes HMAC, a signature of size 42 bytes, 16 bytes for a symmetric key and 4 bytes for index ID are attached in one message. Therefore, the communication cost for VAST scheme is 145 bytes. Communication cost for ECDSA is 181 bytes due to the attached certificate of size 125 bytes and 56 bytes for the signature. For our scheme the total communication cost is 80 bytes, as a single message includes two HMAC values of size 20 bytes each, a 20 bytes TESLA symmetric key and 20 bytes for the prediction outcome of the CRT.

Fig. 8 shows the relationship between communication overhead and the number of vehicles in communication range. Obviously, as the number of vehicles increases, the communication overhead increases linearly. It can be seen that TESLA scheme has the lowest communication overhead amongst the schemes. Whereas our scheme has a lower communication overhead than the other schemes, with the exception of TESLA. This is due to the HMAC being appended, as well as the prediction values to provide instant authentication. Although TESLA has lower

TABLE VI
SIGNING AND VERIFYING COST OF BSM

Schemes	PBA	TESLA	VAST	ECDSA	Proposed scheme
Signing cost	$61T_h + 2T_{mac}$	T_{mac}	$T_{mul} + T_{mac}$	T_{mul}	$8T_h + 8T_m + 2T_{mac}$
Verifying cost	$6T_h + T_{mac}$	T_{mac}	$4T^*_{mul} + 2T_{mac}$	$4T_{mul}$	$T_h + T_m + T_{mac}$

$*$ Note: Receivers in VAST scheme only authenticate digital signature and certificate of a sender when the property of non-repudiation is desired.

communication overhead but it does not allow receivers to authenticate messages instantly, which is a drawback. Furthermore, we can show from the comparison between all the schemes in Table V that our scheme is 50% of PBA, 55.2% of VAST and 44.2% of ECDSA. It is worth mentioning that if an accurate movement prediction table (with large prediction entries) to be considered, PBA scheme will have a large communication overhead due to the large number of leafs in the MHT, which need to be included in the message to enable receivers to verify the message. On the other hand, our scheme will have no increase in the communication overhead because we only include the prediction value, which is computed by the CRT.

2) Message Signing Cost: The comparison of message signing cost is presented in Table VI. Let T_h denote the time taken to perform one hash function operation, T_{mac} denote the time taken to perform one HMAC operation, T_{mul} denote the time taken to execute one point multiplication operation and T_m denote the time taken to perform one modulo operation. The implementation time of T_h, T_{mac}, T_{mul} and T_m are 0.013 ms, 0.019 ms, 1.75 ms and 0.00491 ms respectively.

As shown in Table VI the cost of message signing for PBA requires 61 hash operations and 2 HMAC operations. Therefore, the required time to sign a single message is $0.013 \times 61 + 0.019 \times 2 \approx 0.831$ms. As for TESLA scheme, a sender is only required to execute one HMAC operation, hence approximately 0.019 ms is needed to sign a single message. In VAST, signing a single message requires one point multiplication operation and HMAC operation. Thus, the time required to sign a message is approximately $1.75 + 0.019 \approx 1.769$ms. A point multiplication operation is executed when signing a message in ECDSA, which requires approximately 1.75 ms. Finally, for our scheme signing a single message requires 8 hash operations, 8 modulo operations and 2 HMAC operations. Hence, the time required is $0.013 \times 8 + 0.00\,491 \times 8 + 0.019 \times 2 \approx 0.18\,128$ ms.

It is obvious that TESLA scheme has a better performance in term of message signing when compared to the other schemes. This is because it only requires one operation (T_{mac}). Whereas, in our scheme the signing computation overhead is slightly higher than that of TESLA, but lower than other schemes. This is because in our scheme a sender have to construct the prediction table, and the compute the CRT value to obtain a single value for all movement prediction. As such, the computational overhead for signing is slightly higher than TESLA. Unlike the TESLA scheme, a sender only computes a HMAC of a message which will enable senders to sign messages efficiently but requires receivers to buffer the message until the next interval. On the other hand, PBA and our scheme have the same approach of providing instant message authentication where a sender should also construct a prediction table for future movements. PBA uses

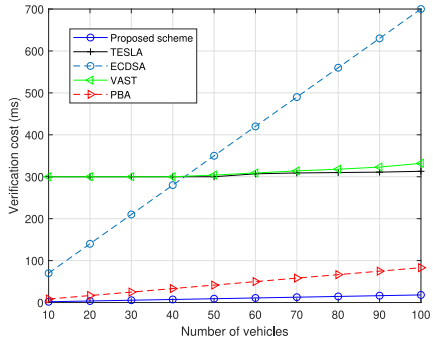

Fig. 9. Verification delay against number of vehicles.

the MHT to compute a single prediction value. The MHT has 6 layers which results in executing a large number of hash operations. As for ECDSA and VAST schemes they both incur the highest message signing overhead due to the point multiplication execution used in signing a message.

3) Message Verifying Cost: The cost of message verification is shown in Table VI. As it can be seen, TESLA has the lowest verification cost due to only requiring one HMAC operation. As for our scheme it has superior performance when compared with the rest of schemes. This is because a receiver is only required to execute one operation of hash function, HMAC and modulo operation in order to be able to authenticate a message. In PBA, each receiver is required to execute an HMAC operation and 6 hash functions. Therefore, they have a slightly higher computation overhead. For VAST, if the property of non-repudiation is required, then a receiver's computation overhead is 4 point multiplication operations and 2 HMAC operations. Two of the point multiplication operations are executed to verify the certificate of the sender and the other two are executed to verify the signature on the message.

To reflect the efficiency of our scheme against the others, we present the message verification delay in Fig. 9. It can be observed that our scheme has the lowest overhead in terms of message verification, since it only utilises HMAC and modulo operations. In PBA, a receiver is required to execute a HMAC operation and multiple hash functions, depending on the size of the MHT. The authors of the PBA scheme have utilised six layers of MHT which means a receiver has to run 6 hash functions. Therefore, their verification overhead is 44.89% higher than our scheme. Although, TESLA scheme requires less time for verification when compared to our scheme and others, but it lacks the instant authentication property which makes it undesirable for safety application. It can be seen in Fig. 9 that TESLA has a 300 ms delay. As for VAST scheme, a receiver can verify a message based on a HMAC operation only when non-repudiation is

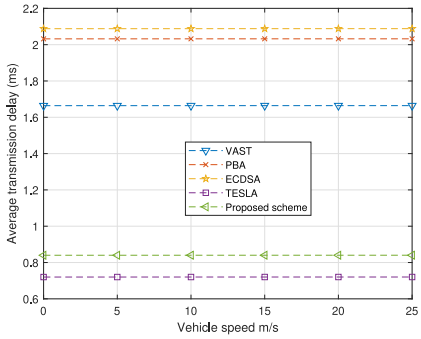

Fig. 10. Impact of vehicles' speed on average transmission delay.

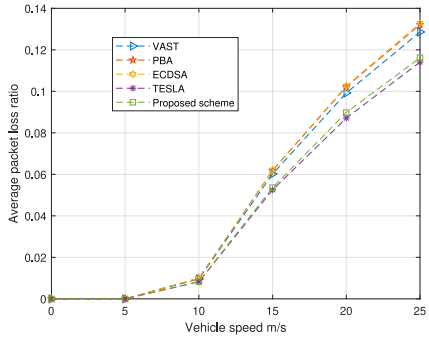

Fig. 11. Impact of vehicles' speed on average packet loss ratio.

not required. However, when there is a necessity for a receiver to achieve non-repudiation, then a receiver is required to run four point multiplication operations to validate the certificate and signature of senders. Also, VAST has a 300 ms authentication delay since they employ the conventional TESLA, which has no support for instant authentication. The verification of VAST scheme is 80% higher than our scheme. Finally, for ECDSA the verification delay is the highest due to the requirement for four point multiplication operations to be executed for verifying each message.

For more insightful knowledge to the reader we thought of showing the effect of a vehicle's speed on average transmission delay and average packet loss ratio. For this part of simulation we assume that the number of vehicles is 50. The simulation results on the average transmission delay and average packet loss rate are shown in Figs. 10 and 11. In addition, we define the average transmission delay T_D of the message between the receiver and the sender in equation (5):

$$T_D = Avg(\Sigma_{i=1}^{n} Avg(\Sigma_{j=1}^{N_j}(T_r^j - T_s^j))) \tag{5}$$

where n represents the number of vehicles; N_j represents the number of received messages from vehicle V_j; T_s^j and T_r^j represent the time of sending the message and the time of receiving the message, respectively. Note that T_r^j-T_s^j corresponds to the time it takes for the message to be transmitted between two vehicles. From Fig. 10, we can see that the average transmission delay of different schemes tends to be stable when vehicle's speed is less than 30 m/s. It also can be observed that our scheme outperforms PBA, VAST and ECDSA. Whereas, TESLA has slightly lower delay and that is because of the packet size difference. Therefore, from Fig. 10 we can conclude that our scheme meets the latency requirement of VCS (≤ 20 ms)[52].

Moreover, the average packet loss ratio is the percentage of lost messages in the total number of messages as defined in equation (6):

$$PL = Avg(\Sigma_{i=1}^{n} num_l^i (num_r^i + num_l^i)^{-1}) \tag{6}$$

where $AVG(.)$ is an averaging function; n is number of vehicles; num_r^i is the number of received messages by V_i and num_l^i is the number of lost messages. It can be seen from Fig. 11, when a vehicle's speed is higher than 10 m/s the effect on average packet loss ratio increases. This rise happens because when a vehicle's speed is high, the probability that a vehicle will move out of the communication range of other vehicles increases. Fortunately, the average loss ratio of our scheme and TESLA are the lowest compared with the other benchmarks.

VIII. Conclusion

In this paper, we have proposed a certificateless lightweight authentication scheme for vehicular communication systems, where we have introduced authentication tokens to replace the expensive digital certificates. The possession of authentication tokens allow vehicles in our scheme to achieve mutual authentication with the TA when joining the network. Furthermore, we have achieved source authentication for broadcasted safety messages based on the lightweight TESLA authentication scheme. As the conventional TESLA protocol does not support non-repudiation, each token corresponds to a vehicle's TESLA keys and it is signed using Schnorr signature. Therefore, preventing illegitimate vehicles from accessing the network. Moreover, since TESLA does not verify messages instantly, a movement prediction based on the trajectory of a vehicle is used to allow vehicles to authenticate BSM without the any delays. In addition, as movement predictions can be large, which might lead to an increase in the communication overhead. We have computed a single value for all the possible movements based on the Chinese Remainder Theorem (CRT). Hence, reducing the communication and computation overheads. Furthermore, based on the extensive security and performance analysis, the proposed scheme has shown resistance to common attacks, scalability, and practicality. While it outperforms the lightweight authentication schemes that are based on similar techniques.

References

[1] P. Vijayakumar, M. Azees, A. Kannan, and L. J. Deborah, "Dual authentication and key management techniques for secure data transmission in vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1015–1028, Apr. 2016.

[2] B. E. Y. Belmekki, A. Hamza, and B. Escrig, "Cooperative vehicular communications at intersections over Nakagami-m fading channels," *Veh. Commun.*, vol. 19, Oct. 2019, Art. no. 100165.

[3] B. E. Y. Belmekki, A. Hamza, and B. Escrig, "On the outage probability of cooperative 5G NOMA at intersections," in *Proc. IEEE 89th Veh. Technol. Conf. (VTC2019-Spring)*, 2019, pp. 1–6.

[4] J. Huang, L. Yeh, and H. Chien, "Abaka: An anonymous batch authenticated and key agreement scheme for value-added services in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 1, pp. 248–262, Jan. 2011.

[5] P. Asuquo *et al.*, "Security and privacy in location-based services for vehicular and mobile communication: An overview, challenges, and countermeasures," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4778–4802, Dec. 2018.

[6] ETSI, "Intelligent transport systems (ITS), vehicular communication, basic set of applications, analysis of the collective perception service (CPS), informative report for the collective perception service," *Eur. TeleCommun. Standards Inst. (ETSI)*, Tec. Rep. TR 103 562 V2.1.1, Dec. 2019.

[7] X. Zhu, S. Jiang, L. Wang, and H. Li, "Efficient privacy-preserving authentication for vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 2, pp. 907–919, Feb. 2014.

[8] J. B. Kenney, "Dedicated short-range communication (dsrc) standards in the united states," *Proc. IEEE*, vol. 99, no. 7, pp. 1162–1182, Jul. 2011.

[9] J. T. Isaac, S. Zeadally, and J. S. Camara, "Security attacks and solutions for vehicular ad hoc networks," *IET Commun.*, vol. 4, no. 7, pp. 894–903, Apr. 2010.

[10] S. Tangade, S. S. Manvi, and P. Lorenz, "Decentralized and scalable privacy-preserving authentication scheme in VANETs," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8647–8655, Sep. 2018.

[11] L. Zhang, Q. Wu, A. Solanas, and J. Domingo-Ferrer, "A scalable robust authentication protocol for secure vehicular communication," *IEEE Trans. Veh. Technol.*, vol. 59, no. 4, pp. 1606–1617, May 2010.

[12] A. Wasef, R. Lu, X. Lin, and X. Shen, "Complementing public key infrastructure to secure vehicular ad hoc networks [security and privacy in emerging wireless networks]," *IEEE Wireless Commun.*, vol. 17, no. 5, pp. 22–28, Oct. 2010.

[13] S. Guo, D. Zeng, and Y. Xiang, "Chameleon hashing for secure and privacy-preserving vehicular communication," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 11, pp. 2794–2803, Nov. 2014.

[14] R. Lu, X. Lin, H. Zhu, P. Ho, and X. Shen, "Ecpp: Efficient conditional privacy preservation protocol for secure vehicular communication," in *Proc. IEEE INFOCOM 2008 27th Conf. Comput. Commun.*, Apr. 2008, pp. 1229–1237.

[15] "IEEE standard for wireless access in vehicular environments–security services for applications and management messages," *IEEE Std 1609.2-2016 (Revision of IEEE Std 1609.2-2013)*, pp. 1–240, Mar. 2016.

[16] K. Shim, "Reconstruction of a secure authentication scheme for vehicular ad hoc networks using a binary authentication tree," *IEEE Trans. Wireless Commun.*, vol. 12, no. 11, pp. 5386–5393, Nov. 2013.

[17] D. He, S. Zeadally, B. Xu, and X. Huang, "An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks," *IEEE Trans. Inf. Forensics Secur.*, vol. 10, no. 12, pp. 2681–2691, Dec. 2015.

[18] K.-A. Shim, "Cpas: An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks," *IEEE Trans. Veh. Technol.*, vol. 61, no. 4, pp. 1874–1883, May 2012.

[19] Z. Wei, J. Li, X. Wang, and C. Gao, "A lightweight privacy-preserving protocol for VANETs based on secure outsourcing computing," *IEEE Access*, vol. 7, pp. 62 785–62 793, 2019.

[20] G. Zhang, Y. Liao, Y. Fan, and Y. Liang, "Security analysis of an identity-based signature from factorization problem," *IEEE Access*, vol. 8, pp. 23 277–23 283, 2020.

[21] J. Cui, L. Wei, J. Zhang, Y. Xu, and H. Zhong, "An efficient message-authentication scheme based on edge computing for vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1621–1632, May 2019.

[22] T. Limbasiya and D. Das, "Lightweight secure message broadcasting protocol for vehicle-to-vehicle communication," *IEEE Syst. J.*, vol. 14, no. 1, pp. 520–529, Mar. 2020.

[23] C. Lyu, D. Gu, Y. Zeng, and P. Mohapatra, "Pba: Prediction-based authentication for vehicle-to-vehicle communication," *IEEE Trans. Dependable Secure Comput.*, vol. 13, no. 1, pp. 71–83, Jan. 2016.

[24] A. Studer, F. Bai, B. Bellur, and A. Perrig, "Flexible, extensible, and efficient VANET authentication," *J. Commun. Netw.*, vol. 11, no. 6, pp. 574–588, Dec. 2009.

[25] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *J. Comput. Secur.*, vol. 15, no. 1, pp. 39–68, Jan. 2007.

[26] C. D. Jung, C. Sur, Y. Park, and K. Rhee, "A robust and efficient anonymous authentication protocol in VANETs," *J. Commun. Netw.*, vol. 11, no. 6, pp. 607–614, Dec. 2009.

[27] A. Wasef and X. Shen, "Emap: Expedite message authentication protocol for vehicular ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 1, pp. 78–89, Jan. 2013.

[28] G. Kumar, R. Saha, M. K. Rai, and T. Kim, "Multidimensional security provision for secure communication in vehicular ad hoc networks using hierarchical structure and end-to-end authentication," *IEEE Access*, vol. 6, pp. 46 558–46 567, 2018.

[29] S. Kanchan, G. Singh, and N. S. Chaudhari, "Sapsc: Signrecrypting authentication protocol using shareable clouds in VANET groups," *IET Intell. Transport Syst.*, vol. 13, no. 9, pp. 1447–1460, Sep. 2019.

[30] C. Sun, J. Liu, Y. Jie, Y. Ma, and J. Ma, "Ridra: A rigorous decentralized randomized authentication in VANETs," *IEEE Access*, vol. 6, pp. 50 358–50 371, 2018.

[31] X. Zhang and D. Wang, "Adaptive traffic signal control mechanism for intelligent transportation based on a consortium blockchain," *IEEE Access*, vol. 7, pp. 97 281–97 295, 2019.

[32] A. Mansour, K. M. Malik, A. Alkaff, and H. Kanaan, "ALMS: Asymmetric lightweight centralized group key management protocol for VANETs," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–16, Feb. 2020.

[33] T. Chim, S. Yiu, L. C. Hui, and V. O. Li, "Specs: Secure and privacy enhancing communication schemes for VANETs," *Ad Hoc Netw.*, vol. 9, no. 2, pp. 189–203, 2011.

[34] S. Horng *et al.*, "b-specs+: Batch verification for secure pseudonymous authentication in VANET," *IEEE Trans. Inf. Forensics Secur.*, vol. 8, no. 11, pp. 1860–1875, Nov. 2013.

[35] J. Cui, J. Zhang, H. Zhong, and Y. Xu, "Spacf: A secure privacy-preserving authentication scheme for VANET with Cuckoo filter," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10 283–10 295, Nov. 2017.

[36] M. R. Asaar, M. Salmasizadeh, W. Susilo, and A. Majidi, "A secure and efficient authentication technique for vehicular ad-hoc networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5409–5423, Jun. 2018.

[37] H. Cui, R. H. Deng, and G. Wang, "An attribute-based framework for secure communication in vehicular ad hoc networks," *IEEE/ACM Trans. Netw.*, vol. 27, no. 2, pp. 721–733, Apr. 2019.

[38] W. Luo and W. Ma, "Efficient and secure access control scheme in the standard model for vehicular cloud computing," *IEEE Access*, vol. 6, pp. 40 420–40 428, 2018.

[39] B. Ying and A. Nayak, "Anonymous and lightweight authentication for secure vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 10 626–10 636, Dec. 2017.

[40] J. Cui, J. Zhang, H. Zhong, R. Shi, and Y. Xu, "An efficient certificateless aggregate signature without pairings for vehicular ad hoc networks," *Inf. Sci.*, vol. 451/452, pp. 1–15, 2018.

[41] H. Zhong, S. Han, J. Cui, J. Zhang, and Y. Xu, "Privacy-preserving authentication scheme with full aggregation in VANET," *Inf. Sci.*, vol. 476, pp. 211–221, 2019.

[42] H. Du, Q. Wen, and S. Zhang, "An efficient certificateless aggregate signature scheme without pairings for healthcare wireless sensor network," *IEEE Access*, vol. 7, pp. 42 683–42 693, 2019.

[43] I. A. Kamil and S. O. Ogundoyin, "An improved certificateless aggregate signature scheme without bilinear pairings for vehicular ad hoc networks," *J. Inf. Secur. Appl.*, vol. 44, pp. 184–200, 2019.

[44] A. Lei, H. Cruickshank, Y. Cao, P. Asuquo, C. P. A. Ogah, and Z. Sun, "Blockchain-based dynamic key management for heterogeneous intelligent transportation systems," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1832–1843, Aug. 2017.

[45] W. S. Hathal, H. Cruickshank, P. Asuquo, Z. Sun, and S. Bao, "Token-based lightweight authentication scheme for vehicle to infrastructure communications," in *Proc. Living Internet Things (IoT 2019)*, 2019, pp. 1–6.

[46] W. Mao, *Modern Cryptography: Theory and Practice*. Englewood Cliffs, NJ, USA.: Prentice Hall Professional Technical Reference, 2003.

[47] A. Perrig, R. Canetti, J. D. Tygar, and D. Song, "The tesla broadcast authentication protocol," *Rsa Cryptobytes*, vol. 5, no. 2, pp. 2–13, 2002.

[48] N. S. Szabo, j. a. Tanaka, and Richard I, *Residue Arithmetic and Its Applications to Computer Technology.* New York, NY, USA: McGraw-Hill, 1967.

[49] C. P. Schnorr, "Efficient signature generation by smart cards," *J. Cryptol.*, vol. 4, no. 3, pp. 161–174, Jan. 1991.

[50] D. Pointcheval and J. Stern, "Security arguments for digital signatures and blind signatures," *J. Cryptol.*, vol. 13, no. 3, pp. 361–396, 2000.

[51] M. Burrows, M. Abadi, and R. M. Needham, "A logic of authentication," *Proc. Roy. Soc. London. A. Math. Phys. Sci.*, vol. 426, no. 1871, pp. 233–271, 1989.

[52] J. Zhang, H. Zhong, J. Cui, Y. Xu, and L. Liu, "An extensible and effective anonymous batch authentication scheme for smart vehicular networks," *IEEE Internet Things J*, vol. 7, no. 4, pp. 3462–3473, Feb. 2020.

Waleed Hathal received the B.Eng. degree in electrical and electronic engineering and the M.Sc. degree (with distinction) in advanced digital systems from the University of Hertfordshire, U.K., in 2013 and 2014, respectively. He received the Ph.D. degree from the University of Surrey in 2020. He is currently a Member with the 5G Innovation Centre (5GIC), University of Surrey where he works as a Research Fellow. His research interests include cybersecurity for connected autonomous vehicles (CAV), 5G, smart cities, and IoT.

Haitham Cruickshank is a Reader with the Institute for Communication Systems (ICS), University of Surrey, Guildford U.K. He is experienced Researcher and worked several U.K., EU and ESA security related projects. He coauthored several ETSI specifications on Intelligent Transport Systems (ITS) privacy and broadband satellite network security architectures. He has more than 163 papers including 34 refereed journals, 120 conferences, four books chapters and five IETF/ETSI standards. His main research interests include privacy and security in communication networks, user and IoT, and future networking architecture in mobile, and satellite and Internet. This includes work on 5G and maximizing network performance in providing security and privacy to users despite the large amount of personal data sharing in this system.

Zhili Sun received the B.Sc. degree in mathematics from Nanjing University and the Ph.D. degree from the Department of Computing, Lancaster University. He is a Chair Professor with the Institute of Communication Systems (ICS), University of Surrey, U.K. He has authored three books and authored or coauthored more than 240 papers in international journals and conferences. His research interests include satellite communications and networks, wireless mobile and sensor networks, mobile operating systems, traffic engineering, and IP networks and security.

Carsten Maple is Deputy Pro-Vice-Chancellor with the University, charged with leading the strategy in North America. He is also the Principal Investigator with the NCSC-EPSRC Academic Centre of Excellence in Cyber Security Research with the University and Professor of Cyber Systems Engineering in WMG. He is a Co-Investigator with the PETRAS National Centre of Excellence for IoT Systems Cybersecurity where he leads on Transport and Mobility. He has an International Research Reputation and extensive experience of institutional strategy development and interacting with external agencies. He has authored or coauthored more than 250 peer-reviewed papers and is coauthor of the U.K. Security Breach Investigations Report 2010, supported by the Serious Organised Crime Agency and the Police Central e-crime Unit. He is also coauthor of Cyberstalking in the U.K., a report supported by the Crown Prosecution Service and Network for Surviving Stalking. His research has attracted millions of pounds in funding and has been widely reported through the media. He has given evidence to government committees on issues of anonymity and child safety online. Additionally, he has advised executive and non-executive directors of public sector organisations and multibillion pound private organisations.