

Received 20 December 2022, accepted 6 February 2023, date of publication 13 February 2023, date of current version 21 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3244697

RESEARCH ARTICLE

An Anonymous Authentication With Received Signal Strength Based Pseudonymous Identities Generation for VANETs

AMANG SUDARSONO¹, AND MIKE YULIANA, (Member, IEEE)

Department of Electrical Engineering, Electronics Engineering Polytechnic Institute of Surabaya, Politeknik Elektronika Negeri Surabaya, Surabaya 60111, Indonesia

Corresponding author: Amang Sudarsono (amang@pens.ac.id)

This work was supported in part by the Ministry of Education, Culture, Research, and Technology of Indonesia, through Penelitian Dasar Unggulan Perguruan Tinggi (PDUPT) Scheme, in 2022.

ABSTRACT Anonymous authentication system enables mobile users to anonymously authenticate themselves to an authorized entity such as a *Group Manager (GM)* without revealing any privacy information. It provides unlinkable but accountable communications as well. These features are useful for wireless mobile networks implementation including vehicle ad-hoc networks (VANETs). However, performance of the system has to be sufficient reliable which may be existing systems have not dealt with yet. In this paper, we propose pseudonymous-based anonymous authentication between participating mobile users involved in the communications. We combine shared key generation based on received signal strength (RSS) between two involving entities and unlinkable but accountable pseudonymous-based anonymous authentication with efficient and effective pseudonym self-generation and revocation process. Our proposed shared key generation provides unique *pseudonymous identities (PIDs)*. Based on *PID* at epoch time and updated revocation list obtained from *GM*, we achieve an efficient cost computation of revocation check. We show the measurement scenario of system performance by varying the traffic conditions either quiet or crowded to create communication impairment combined with mobile users' speed varying on 20 km/h, 40 km/h, and 60 km/h respectively and *ping* time interval settings on 7 ms, 10 ms, and 20 ms, respectively. Here, the result of evaluation shows that *PIDs* generation works properly by the number of generated *PIDs* up to 11 with the highest correlation up to 0.99. Meanwhile, quantization algorithm works properly for 3000 or more ICMP packets and achieves zero key disagreement rate (KDR). Total signing and verification times are sufficient practical about 90 ms and 100 ms, respectively.

INDEX TERMS Pseudonym, anonymity, revocation, received signal strength, shared key generation.

I. INTRODUCTION

As the advancement of mobile ad-hoc networks (MANETs) including VANETs, security and privacy are now becoming a matter that is very mandatory to be considered [1], [2], [3], [4], [5]. We realize due to such networks are infrastructure-less networks accessed freely and wirelessly at anytime and anywhere as long as the device can reach them. Therefore, everyone is able to access the networks without permission of

The associate editor coordinating the review of this manuscript and approving it for publication was Zijian Zhang¹.

the network administrator. Here, the adversaries may join to the networks as well as other users do. With this phenomenon, threats and attacks can occur at anytime. This becomes worse if the information exchanged on the network is an information related to the privacy information. This privacy information may be routine traffic information of users, location, driving behaviour, etc. One of security and privacy-preserving solutions is anonymous authentication. However, other security requirements may also be considered. Fundamentally, anonymous authentications have already addressed the anonymity and unlinkability whereas only trusted authorities who have

ability to reveal them. Meanwhile, other entities involving in the anonymous authentication system are not able to uncover the anonymity and unlinkability. Currently, there are many anonymous authentication schemes and their implementations have been proposed [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28]. In these schemes, mobile users may utilize some *PIDs* and change a *PID* to other *PID* to meet unlinkability requirement. Mobile users' *PIDs* may be embedded for the goal to revoke either misbehaving mobile users or perhaps those *PIDs* have been expired already. Therefore, the requirement of pseudonyms generation and its *PIDs* distribution for maintaining secure communications should be considered.

Based on reported of Lindell et. al. [2], there are two security requirements in the anonymous authentication: (1) secure authentication that allows no unauthorized user should be able to defraud the system for granting him/her an access, (2) anonymity that allows no entity should know and learn which user is communicating and interacting with. One of widely public key infrastructure algorithms to fulfill these security requirements is group signature. In this algorithm, the valid members in the group are able to sign a message on behalf of the group by using their member secret key without disclosing their privacy information. Moreover, any generated signature could be verified by all other members in the group by using group public key. Thus, by adopting group signature scheme, we can achieve and deploy an anonymous authentication and its implementations including for VANETs.

All members in the group signature defaultly trust to *GM*. Sun et. al. [8] proposed an efficient key management distribution process using group signature based anonymous authentication in VANETs. The scheme employed batch signature verification to support a distributed certificate service (DCS). Moreover, the scheme does not only reduce significantly revocation cost, but also comply security requirements such as authentication, non-repudiation, revocation, anonymity and unlinkability. Here, the authors introduced four entities involved in the system, such as trusted authority (TA), regional group manager (RM), road-side units (RSUs), and vehicles. In addition, Malina et. al. [4], [5] introduced an efficient group signature for privacy-preserving in the vehicular networks. The proposed system is able to minimize the impact of several common attacks such as denial of services (DoS) and reply attacks. There are four entities involved in the system: TA, *GM*, RSUs, and vehicles. The scheme [4] focused on the practical of registration, join protocol, signing and verification protocol. However, due to conventional asymmetric cryptography usage in the registration and join protocol, the system is less effective because it needs to maintain a key distribution process of membership. In addition, vehicles and RSUs are suffering from secret key and other public key elements. Gao et. al. [10] introduced identity-based signature with pseudonyms instead of public key infrastructure to achieve the efficiency and effectiveness.

However, multiple pseudonyms are presented to preserve the privacy of vehicles may the system be complex when dealing with much more number of RSUs and vehicles. In addition, other implementations of VANETs using group signature have been introduced as well such as in [4], [6], [7], [8], [9], [10], [12], [16], [19], [22], [24], [27], and [28] that focusing on key management and distribution mechanisms, and trying to achieve as effective as anonymous authentication mechanism among vehicles. Meanwhile, the use of pseudonym for anonymous authentication in the VANETs has been well proposed [1], [3], [16], [24], [27], [28]. Here, privacy-preserving based on pseudonymity is performed by various solutions for VANETs implementation. Adversary model also has been presented by introducing several potential attacks globally, locally, actively, passively, internally or externally. In this case, pseudonym lifecycle is well described and explained regarding its issuing, usage, changing, resolution, and revocation as well. An anonymous identity authentication based on pseudonym for the implementation of mobile crowdsensing (MCS) [15], [18], [21], [25] has been proposed. The definition of attack model for MCS network is explained as well. Here, the authors combined public key infrastructure and public key to solve the problem of management in large scale and evaluated the proposed anonymous authentication by testing the function and performance.

Throughout an efficient verifier-local revocation (VLR) group signature algorithm, Rahaman et. al. [25] proposed an enabler anonymous but considering the accountability of communications. They introduced a sublinear revocation with backward unlinkability and exculpability (SRBE) scheme to support the implementations such as smartphone-based crowdsensing, citizen science, and vehicular communications. However, it has a drawback when handling a particular scenario that needs more than one pseudonyms within epoch time. Sucasas et.al, [21], [23] introduced an attribute-based credential (privacy-ABC) to support pseudonym-based authentication through embedded attributes in cloud services implementation. Here, a pseudonym-based signature scheme is proposed to enable unlinkable pseudonym by self-generating the embedded attributes. This scheme offered verifiable delegation and enabling users to share attributes to the service provider. In addition, the used of different pseudonyms is guaranteed to unlink for the same user. However, different pseudonyms from the same user are not able to be used for the same task.

In this paper, we propose pseudonym-based anonymous authentication between participating mobile users involved in the communications. We combine previous scheme [29], [30] of shared key generation based on RSS used for ensuring the similarity of shared-key between two involving entities in the either vehicle-to-infrastructure (V2I) or vehicle-to-vehicle (V2V) communications and unlinkable but accountable pseudonym-based anonymous authentication with efficient and effective of pseudonym self-generation and revocation process as well

as [21], [23], and [25]. In our pseudonymous-based anonymous authentication scheme, throughout *secret key generation (SKG)*-based join protocol, pseudonym self-generation is able to create T number of pseudonyms (PID_{ij}) which run on the participating mobile user (i.e., hereinafter it is called as Mobile- i) for interval time j . Meanwhile, our proposed scheme achieves an efficient cost computation of revocation check, since it is able to avoid computation cost linearly grows proportional to the number of revoked users. To do so, each pseudonym PID_{ij} is generated at index k in $[1, K]$ which embedded into $H(k)^{PID_{ij}}$ together with Mobile- i secret x_i , where K is total index of epoch time and $H()$ denotes a hash function operation. Regarding some notations that appear frequently in this paper, we insert the descriptions as briefly described in Table 1.

Moreover, we employ the randomness characteristic parameters generated by physical layer of wireless network [31], [32] when joining mobile users register themselves to GM for pursuing $PIDs$. Thus, some advantages can be obtained by incorporating SKG process to our proposed anonymous authentication system.

We summarize our technical contributions as follows:

- Generated pseudonyms from SKG process, signing process, and verification process are integrated into a system to fulfill communication scenarios in VANETs.
- Anonymity, privacy-preserving, and pseudonymity requirements for security and privacy protection are able to maintain the computation costs efficiently.
- Signatures generated inside the time epoch are unlinkable. Even if the same pseudonym used in the signing process within epoch time potentially ignites signatures linkable, our proposed scheme is only exposing single pseudonym to be linkable.
- The usage of indexing embedded in the pseudonyms within epoch time is to maintain computation costs when executing revocation check based on $PIDs$, where our proposed scheme only publishes a PID within epoch time, meanwhile other $PIDs$ are kept secret.
- Via SKG process, a PID embedded into $H(k)^{x_i PID_{ij}}$ within epoch time j where $j \in [1, T]$ and index k where $k \in [1, K]$ can be implemented securely and other components can be also encrypted by any symmetric key cryptosystem. Hence, these components are kept secret during transmission.

The evaluation of system performance is carried out by changing the traffic condition either quiet or crowded, varying the speed of mobile users from 20 km/h to 60 km/h and *ping* time interval from 7 ms to 20 ms. Computation cost requires about 12 seconds in average running on Raspberry Pi to conduct SKG process. Meanwhile, total processing time of signing and verification processes in the proposed anonymous authentication only consumes about 350 ms including communication costs. This shows the practicality of our proposed system.

TABLE 1. Notations and descriptions.

Notation	Description
GM	An authorized entity in the anonymous authentication
Mobile- i	A mobile user who are participating in the communications
PID	Pseudonymous identity
PID_{ij}	Pseudonymous identity of Mobile- i at epoch time j
VID_i	The real identity of Mobile- i (i.e., license plate number of vehicle)
REG_i	A database of secret components related to Mobile- i registration
RL	A revocation list
GL	A group list
grt_{ij}	The revocation token of Mobile- i in epoch time j
gpk	A group public key
gsk	A group secret key
gok	A group opening secret key
$msk[i]$	A secret membership key of Mobile- i
$H()$	A cryptographic hash function operation
\mathbb{Z}_q^*	Set of prime numbers
SKG	Shared secret key generation mechanism

The structure of this paper is started by the above introduction. Furthermore, we describe briefly about the overview of anonymous authentication scheme in Section II whereas $PIDs$ may be involved in the process such as join protocol, signing protocol, signing check algorithm, and revocation check algorithm. In Section III, we give some notes of our motivation and the contribution of our work, detail explanation about our proposed anonymous authentication scheme using $PIDs$ generated from SKG process which includes system setup, key generation, join protocol, revocation protocol, anonymous authentication protocol, and open protocol. Then implementation and evaluation are comprehensively discussed in Section IV. And finally, conclusion and future works are discussed in Section V, respectively.

II. PRELIMINARIES

In this section, we briefly describe the fundamental technologies and algorithms adopted in the proposed system. Here, firstly we shortly describe about the fundamental of bilinear pairing as the basic pairing based cryptography. Secondly, we briefly describe about an overview of pseudonymous-based anonymous authentication system. In addition, we describe the corresponding assumption of our proposed anonymous authentication scheme.

A. BILINEAR MAP OF PAIRING

- let multiplicative cyclic groups, \mathbb{G}_1 and \mathbb{G}_2 respectively of prime order q .
- let a generator of \mathbb{G}_1 , g_1 and a generator of \mathbb{G}_2 , g_2 .
- let a computable isomorphism, φ from \mathbb{G}_2 to \mathbb{G}_1 such that an isomorphism function $\varphi(g_2) = g_1$; and
- let a bilinear map, e whereas $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ which has particular characteristic as follows:
 - Bilinearity: for all $U \in \mathbb{G}_1$, $V \in \mathbb{G}_2$ and $a, b \in \mathbb{Z}$, where $e(U^a, V^b) = e(U, V)^{ab}$.
 - Non-degeneracy: $e(g_1, g_2) \neq 1$.

B. OVERVIEW OF PSEUDONYMOUS BASED ANONYMOUS AUTHENTICATION

As well as [16], [21], [23], [24], [25], and [26], pseudonymous-based anonymous authentication may comprise several algorithms and protocols such as key setup algorithm, user pseudonym generation, user join protocol which may be pseudonym generation as a part of join protocol, user revocation check algorithm based on generated *PIDs*, signing algorithm, and verification algorithm. Generally, these processes are executed by a trusted authority (i.e., may be represented by a *GM* that can act as key setup generator, user join manager, user revocation manager, verifier entity, and open manager) and mobile users who are able to act as either signer user or verifier user.

- Key setup generation: on given security parameters, *GM* executes this algorithm to create group public key *gpk* and group secret key *gsk*. The given parameters may be consisted of a specific group order q of a bilinear map. Then, two multiplicative cyclic groups \mathbb{G}_1 and \mathbb{G}_2 have to be selected as well to create a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$. Furthermore, a cryptographic hash function is also selected $H() : \{0, 1\}^* \rightarrow \mathbb{Z}_q^*$. Then, the output of this algorithm is publishing *gpk* and *gsk*. In the some conditions, credentials may also be needed. Here, *GM* involves *gsk* to extract public parameters in *gpk* when creating credential components for a user with respect to the user's *PID*. Note that, to get its credential, a user has to request it to *GM* through a secure channel communication.

- User pseudonym generation: this algorithm optionally may be performed to initially create *PIDs* of users based on user membership index and epoch time T . Based on these generated *PIDs*, a user is able to sign a message anonymously through signing algorithm. Based on these *PIDs*, the verifier also verifies whether the signature is valid or not and makes sure that the valid signature is not in the revocation list. In our proposed system, as well as *SKG* process in [29] and [30], *PIDs* are represented by generated shared secret keys derived from collected RSS values between joining mobile users and *GM*. Here, we employ randomness extraction algorithm to fetch reciprocity of collected RSS values on both sides. Later on, a quantization algorithm is utilized to quantize and convert them into binary form to increase the correlation of collected RSS values between joining mobile users and *GM*. Furthermore, reconciliation and verification are executed to obtain the exact key stream on both sides. These key stream can be represented as *PIDs* of joining mobile users. In our proposed system, to have a set of shared secret keys represented as *PIDs* of joining mobile users, four steps are performed sequentially. Firstly, RSS values are collected through channel probing. Then, the values are quantized by particular quantization algorithm. Furthermore, the values must be synchronized through information reconciliation, and finally the result bits are increased their randomness by computing privacy amplification.

- User join protocol: this protocol is done interactively communication between a joining user (i.e., registering mobile user) and *GM* for pursuing the joining user to join the system. The protocol firstly is started by mobile user to request a particular credential and *gpk* to *GM*. After fetching the credential and *gpk*, sometimes *PIDs* can be created based on such credential. Sometimes, attributes may be attached in the credential and dispatched to another entity to generate *PIDs*. Here, some computations have to be executed by mobile user and some secret values are obtained. The epoch time or time slot may also be added in the *PIDs* which are used when mobile user convinces other entities about its validity in the system due to their transactions or communications done in every time. In addition, this protocol also delivers membership secret key *msk* of joining user. Meanwhile, *GM* stores some secret components of the joining user in the registration database.

- User revocation algorithm: based on valid *PIDs* generated either through pseudonym generation process or join protocol, *GM* sets a certain equation which correlating between these *PIDs* of the user and epoch time when some unexpectation acts occur such as misbehaving activities, secret key loss, secret key expiration, etc. The outputs of this algorithm is a revocation list *RL* which consisting of one or more random components related to the revoked user's *PIDs* in the list.

- Signing algorithm: a user operates this algorithm to convince his/her legality to a verifier when accessing the system anonymously. The algorithm requires *PIDs* of signer user, secret key of signer user *msk*, and public key *gpk*. The algorithm generates a signature on a message M (i.e., with certain arbitrary length of message). Here, signer user should select some random values, commitment values, and other components for signing process together with *PIDs* based on epoch time. To do so, signer user computes them together with his/her own *msk*. This algorithm may yield involving auxiliary public keys and some challenge components. Then, all components are used to sign the message M anonymously.

- Verification algorithm: a verifier runs this algorithm which commonly comprises two steps. Both users in signing and verification algorithms may be mobile users (e.g., one acts as a signer and another acts as the verifier). First step is signature check. Here, verifier has to check the validity of signature generated in the signing process. By executing verification algorithm based on *PID* with its epoch time and index, verifier verifies whether the signature is valid or not. Then, second step is performed to ensure that user is not in the list of revocation list if and only if the verification of signature is valid. In this revocation check, when *PID* on particular epoch time is reported as invalid, verification algorithm result should detect it as invalidity because the *PID* embedded in the revocation component is found in the list. However, when *PID* attached in the signature generation is valid, the verifier ensures it by comparing a particular equation whether the signer user's signature on a message M is valid or not.

C. ASSUMPTIONS

The traceability and unforgeability requirements of our implemented pseudonymous-based anonymous authentication scheme are based on the q -SDH assumption, DLIN assumption, and DL assumption. Since, in this paper does not address q -SDH assumption, we omit this assumption. The definitions of assumptions are also well described in [25] based on the construction frameworks discussed in [33] as follows.

Definition 1 (Decision Linear (DLIN) Assumption): For all PPT algorithm \mathcal{A} , the probability $\Pr[\mathcal{A}(U, V, W, \tilde{U}, \tilde{V}, \tilde{W}, U^a, V^b, W^{a+b}) = 0] - \Pr[\mathcal{A}(U, V, W, U^a, V^b, W^c) = 0]$ is negligible, where $U, V, W \in_R \mathbb{G}_1$ and $a, b, c \in_R \mathbb{Z}_q$.

Definition 2 (DL Assumption): On given inputs $g_1, g_1^a \in \mathbb{G}_1$, where $a \in_R \mathbb{Z}_q^*$, then the output is a . Here, it can be said that (t, ϵ) -DL assumption holds in \mathbb{G}_1 , if no PPT algorithm \mathcal{A} has an advantage at least ϵ to solve DL problem in \mathbb{G}_1 .

III. PROPOSED SYSTEM

In this section, we briefly describe a proposed system of pseudonymous-based anonymous authentication which comprises system setup, key generation, join protocol, revocation process, signing protocol, verification protocol, and open protocol. In the join protocol, PID_{ij} of a participating joining mobile user are generated through the contribution of SKG process. The usage of SKG process is also involved in the signing protocol. Whilst, in the verification protocol, there would be two steps. First step, the mobile verifier does signature check. If and only if the check is valid, then second step is executed by checking whether the mobile user is a revoke user or not.

A. OUR MOTIVATION AND CONTRIBUTION

Our main motivation in adopting SKG process is to utilize the advantages of randomness parameters in physical layer of wireless communication [31], [32] generated from collected RSS values between joining mobile user and GM through a join protocol. We employ the SKG process to securely exchange secret components in the authentication protocol as well. In this case, the excavating randomness parameters derived from RSS values are generated by ICMP packet of a communication either between joining mobile user and GM or between signer mobile user and verifier mobile user through *ping* command. Secondly, instead of a particular equation when generating $PIDs$, SKG process yields a set of winner keys to represent the $PIDs$ of communicating entities. Hence, by this idea we reduce the complexity of pseudonym identities generation as a part of join protocol. By the assisting SKG process may also securely send secret components from signing user to verifier and the usage of shared secret key with particular symmetric key cryptosystem, such as advanced encryption standard (AES-256) [34] may also provide secure data exchange. Here, as well as [29] and [30], we adopt NIST statistical test suite to test the randomness of pseudo-random number generators [35] and *tshark* network protocol

TABLE 2. Comparison of proposed scheme and several existing schemes in term of requirements for privacy-preserving authentication.

Scheme	GS-TDL [12]	Gao et. al. [10]	SRBE [25]	Sucasas et. al. [21], [23]	Our Scheme
Pseudonym self-generation	NA	NA	○ (one PID per credential)	○ (unlimited)	○ (one PID per credential)
Revocation check	○	○	○	○	○
Backward unlinkability	○	○	○	○	○
Non transferability	○	○	○	○	○
Unlinkability but accountability	×	×	×	○ (all $PIDs$ in RL are published)	○ (only one PID in RL is published)

analyzer [36] to accommodate and analyze network traffic either in $PIDs$ generation process when executing join protocol or shared secret key generation when performing authentication protocol. In addition, Kalman Filter [37] also is adopted to increase reciprocity of measured RSS values when mobile user authenticates him/her self to mobile verifier in authentication protocol. Moreover, other remaining algorithms and techniques are adopted as well from [29] and [30].

The contribution of our proposed scheme to satisfy the requirements of anonymity and privacy-preserving of authentication system can be illustrated in Table 2. GS-TDL scheme [12] allows signer users are linkable temporarily when generating multiple signatures at the same epoch time T . Hence, it has a problem when T is set into longer time duration then the scheme to be a general digital signature where the signer user should be always linkable. Meanwhile, when T is set into shorter time duration, the scheme to be a usual group signature where the signer user should be always unlinkable. The advantage of the scheme is verifier-local revocation that enables no signer user is burdened in the revocation computations. However, the scheme has not yet involved any pseudonym in the authentication process. To improve the efficiency and anonymity in the authentication system, Gao et. al. [10] introduced an identity-based short group signature. The scheme has already involved pseudonyms to achieve privacy-preserving. However, the complexity of the scheme is high when dealing with much more numbers of participating users. Whilst, SRBE scheme [25] offers self-generation of pseudonyms at signer user side eventhough only single pseudonym for every credential in signing process. In this case, signer user is able to generate single, unique and unlinkable pseudonym. However, it has a problem when handling a particular scenario that needs more than one pseudonym within epoch time. The need of more than one pseudonym is addressed by Sucasas et. al. [21], [23]. In this scheme, signer user is

allowed to sign a message with involving unlimited and unlinkable but accountable pseudonyms. Here, the used of different pseudonyms is guaranteed to unlink for the same user. However, different pseudonyms from the same user are not able to be used for the same task. Both SRBE scheme [25] and Sucasas et. al. [21], [23] play the index value of pseudonym based on epoch time of generated pseudonyms, thus effectiveness searching of a pseudonym can be achieved (e.g., when applying in the revocation check process). However, all pseudonyms in the revocation list must be published for every epoch time. Meanwhile, our proposed scheme as well as [21], [23], and [25] focuses on pseudonym generation derived from RSS values of communications between mobile users and *GM* when joining user registers him/her self in the system. We focus on a single pseudonym self-generation in signing protocol based on the index value of every pseudonym. Thus, effectiveness and efficiency can be achieved as well as [21], [23], and [25] with satisfaction of privacy-preserving requirements. In addition, revocation check process of our proposed scheme provides unlinkability but only one pseudonym in the revocation list published. Moreover, our proposed system satisfies unlinkable but accountable pseudonymity which means that all pseudonyms generated by the same signing user and generally used for different signing the message should not be linkable to each other. In addition, the users should be able to generate several pseudonyms to participate signing the message. In this situation, different pseudonyms generated from a user cannot either be linked to the same user or be used for the same signing process. In this case, given pseudonyms is impossible to reveal which pseudonyms belong to the same user. Given pseudonyms used in signing the message certainly each pseudonym belongs to a different user. Therefore, unlinkable but accountable feature enables users to participate in signing the message without being linked to each other. This also enables VANETs to be ensured that users will not able to participate in the same signing process with two or more different pseudonyms.

B. PROPOSED ANONYMOUS AUTHENTICATION USING PID_{ij} GENERATED FROM SHARED KEY GENERATION

At first, *GM* sets up several public parameters by selecting two cyclic groups \mathbb{G}_1 and \mathbb{G}_2 of prime order q . Then, a multiplicative group \mathbb{G}_T is executed by a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$. Later on, *GM* chooses $g_1 \in_R \mathbb{G}_1$ and $g_2 \in_R \mathbb{G}_2$. In addition, *GM* sets up a hash function $H() : \{0, 1\}^* \in \mathbb{Z}_q^*$. So far, a group public key is denoted as $gpk = \langle q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2, H \rangle$, and group secret key is indicated as $gsk = \langle d, s, u \rangle$, where $d, s, u \in_R \mathbb{Z}_q^*$. Furthermore, *GM* sets up $D = g_1^d \in \mathbb{G}_1$, $U = g_1^u \in \mathbb{G}_1$, $S = g_2^s \in \mathbb{G}_2$, and appends them to the group public key, $gpk = \langle D, S, U \rangle$. Finally, *GM* issues the group public key $gpk = \langle q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2, H, D, S, U \rangle$ and keeps secret $gsk = \langle d, s, u \rangle$.

Fig. 1 step 5 is *SKG* process to generate a set of secret keys that represents PID_{ij} of Mobile-*i*. The detail procedure for

SKG process is described in Fig. 2 as well as [29], and [30]. In *SKG* process extracted from RSS values between Mobile-*i* and *GM* communication, firstly we collect RSS values by channel probing through ICMP packets. In this case, we collect about 3000 ICMP packets. Let say h_{M_i} is a signal sent from Mobile-*i* to *GM* and h_{GM} is signal sent from *GM* to Mobile-*i*. In the meantime, an eavesdropper (i.e., Eve) intercepts h_{M_i} from Mobile-*i* and h_{GM} from *GM*. This can be represented as the following model:

$$\begin{aligned} R_{(M_i-GM)_x} &= H_{(M_i-GM)_x} \times h_{M_{ix}} + N_{(M_i-GM)_x}, \\ R_{(M_i-E)_x} &= H_{(M_i-E)_x} \times h_{M_{ix}} + N_{(M_i-E)_x}, \\ R_{(GM-M_i)_x} &= H_{(GM-M_i)_x} \times h_{GM_x} + N_{(GM-M_i)_x}, \\ R_{(GM-E)_x} &= H_{(GM-E)_x} \times h_{GM_x} + N_{(GM-E)_x}. \end{aligned} \quad (1)$$

where $R_{(M_i-GM)_x}$, $R_{(M_i-E)_x}$, $R_{(GM-M_i)_x}$, and $R_{(GM-E)_x}$ are RSS values received by *GM* from Mobile-*i*, by Eve from Mobile-*i*, by Mobile-*i* from *GM*, and by Eve from *GM*, respectively. $H_{(M_i-GM)_x}$, $H_{(M_i-E)_x}$, $H_{(GM-M_i)_x}$, and $H_{(GM-E)_x}$ are the channel gain estimated by *GM*, Eve, and Mobile-*i*. $N_{(M_i-GM)_x}$, $N_{(M_i-E)_x}$, $N_{(GM-M_i)_x}$, and $N_{(GM-E)_x}$ are zero mean additive Gaussian noise. Then, by using randomness extraction we reach reciprocity of collected RSS values. Here, we improve the correlation of measured RSS values between Mobile-*i* and *GM* by employing polynomial interpolation which is represented as the following model:

$$\begin{cases} a_0 = s(n-1), \\ a_3 = \frac{1}{6}(s(n) - s(n-3)) + \frac{1}{5}(s(n-2) - s(n-1)), \\ a_1 = \frac{1}{5}(s(n) - s(n-2)) - a_3, \\ a_2 = s(n) - s(n-1) - a_1 - a_3. \end{cases} \quad (2)$$

$$y(l) = a_0 + a_1 l + a_2 l^2 + a_3 l^3. \quad (3)$$

where $s(n)$ is the input of RSS values by the index n , a_0, \dots, a_3 are polynomial interpolation coefficients, and $y(l)$ denotes the output of correlated RSS values by the index l . Note that correlation is used for quantifying the relationship of collected RSS values between Mobile-*i* and *GM* [38]. The next step is employing quantization process to convert every single correlated RSS values into bits stream. In this case, we utilize Multibit M-ary quantization by ordering the correlated RSS values from the smallest to the biggest one. Then, we sort the values into several levels of a block, where the level is determined by the guard level using Equation 4.

$$\int_{q_{i-1}}^{q_i - g_i} f_{\tilde{h}} d\tilde{h} = \frac{1 - \delta}{m}. \quad (4)$$

Guard level g_i is set between two series of quantization q_{i-1} and q_i with assumption of measurement h which is followed by a particular probability distribution $f_{\tilde{h}}$. Meanwhile, δ denotes the ratio of guard level. Here, the guard level excludes the same RSS values. Let say, the level of quantization is m (i.e., from 0 to $m-1$), thus the interval of quantization is $I_0 = (q_0, q_1 - g_1), I_1 = (q_1, q_2 - g_2), \dots, I_{m-1} = (q_{m-1}, q_m - g_m)$,

where q_0 and q_m are minimum and maximum of h . Meanwhile, $q_1 (1 \leq i \leq m - 1)$ is determined by Equation 4.

The result bits stream of quantization needs to be filtered the mismatched bits stream remaining in both sides. In this case, we utilize BCH error code correction (i.e., BCH(31, 6)). Here, the result bits key stream from Multibit M-ary quantization are encoded into codewords. Every codeword consists of parity for exchanging in both sides such that bits error correction process is properly executed. Hence, finally we get a preliminary bits key stream between Mobile- i and GM . The next step is increasing the randomness of bits key stream. Here, we utilize Universal hash function such that the randomness of bits key stream passing the NIST pseudorandomness test suite. There are up to 11 winner keys that passed the test. Then, to ensure the equality of 11 keys between Mobile- i and GM , SHA-256 hash function is used to verify whether the keys in both sides are really equal or not. If it is valid, then a set of 11 keys is represented as the $PID_{ij} \in \mathbb{Z}_q$ of Mobile- i .

C. SYSTEM SETUP

The system is initiated by performing algorithm ψ with security parameter λ as an input. ψ outcomes 3 groups \mathbb{G}_1 , \mathbb{G}_2 , and \mathbb{G}_T of λ -bit of prime order q , and a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$. Moreover, a generator g_1 is chosen from \mathbb{G}_1 and a generator g_2 is chosen from \mathbb{G}_2 at random. The system applies a hash function $H() : \{0, 1\}^* \rightarrow \mathbb{Z}_q$ as well. Where, $(q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2, H)$ is public. The secret keys of the group manager, opening manager, and group public respectively can be generated as follows:

- Choose two random secrets $d, s \in_R \mathbb{Z}_q^*$ and assign them as the secret key of group manager $gsk = \langle d, s \rangle$. Then, choose a random secret $u \in_R \mathbb{Z}_q^*$ and assign it to the secret key of the opening manager $gok = \langle u \rangle$.
- Compute $D = g_1^d \in \mathbb{G}_1$, $S = g_2^s \in \mathbb{G}_2$, $U = g_1^u \in \mathbb{G}_1$ and assign them to group public key $gpk = \langle q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2, H, D, S, U \rangle$. Here, only the group manager is able to proceed $\langle d, s \rangle$ and only opening manager is able to access $\langle u \rangle$.

D. JOINING PROTOCOL

Fig. 1 indicates our proposed interactive join protocol as a part of pseudonymous-based anonymous authentication system. The protocol comprises nine steps. Again, this protocol is an interactively communication protocol between joining mobile node (i.e., Mobile- i) and GM . Meanwhile, Fig. 2 shows a shared secret key generation process when receiving $PIDs$ of Mobile- i on time duration T (i.e., PID_{ij}) where $j \in [1, T]$. This is a part of interactive join protocol in the SKG process. The fifth step of the process is depicted in Fig. 1.

Based on gpk and gsk obtained from GM , Mobile- i registers him/her self to GM by doing the following steps:

- Select a random secret key $x_i \in_R \mathbb{Z}_q^*$ and compute key agreement components $A'_i = g_1^{1/x_i} \in \mathbb{G}_1$ and $Q_i = g_1^{VID_i + h_i}$. In this case, VID_i can be represented by the

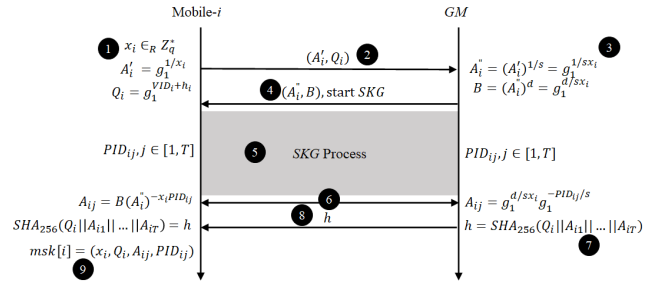


FIGURE 1. Proposed interactive join protocol process.

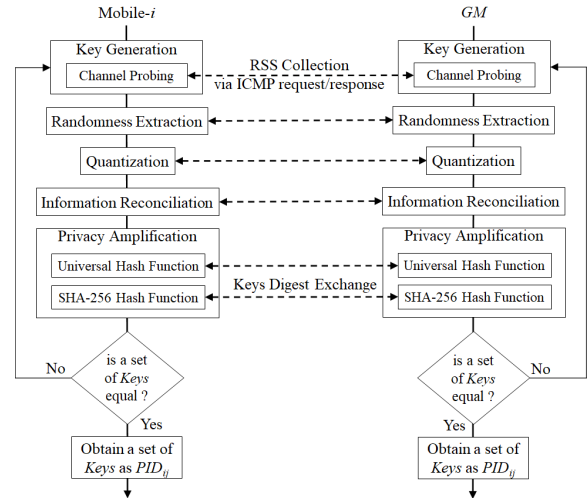


FIGURE 2. Proposed shared secret key generation process to obtain PID_{ij} .

license plate number of Mobile- i , $VID_i = H(\text{plate number})$ and $h_i = H(i)$. Note that VID_i acts as a consistent identity which is the real identity of Mobile- i . This identity is used for registering Mobile- i to join the system. Meanwhile, PID_{ij} is a consistent identity which is not the real identity (i.e., a pseudonym) of Mobile- i generated by a particular pseudonyms generation algorithm based on epoch time j where $j \in [1, T]$. Both of VID_i and PID_{ij} are involved in the authentication process. However, only PID_{ij} is involved in the revocation check process based on epoch time j .

- Send a tuple of (A'_i, Q_i) to GM .
- When GM receives (A'_i, Q_i) from Mobile- i , GM executes $A''_i = (A'_i)^{1/s} = g_1^{1/sx_i}$ and $B = (A'_i)^d = g_1^{d/sx_i}$.
- Then, GM sends a tuple of (A''_i, B) to Mobile- i . Furthermore, SKG process is started.
- SKG process is done interactively between Mobile- i and GM such that PID_{ij} is obtained, where $j \in [1, T]$ and T is the epoch time represented by the number of generated keys in the SKG process. Fig. 3 illustrates scenario of the process.
- Upon obtaining PID_{ij} , Mobile- i executes $A_{ij} = B(A''_i)^{-x_i PID_{ij}}$ where it is equal to $A_{ij} = g_1^{d/sx_i} g_1^{-PID_{ij}/s}$. In this case, PID_{ij} is obtained from SKG process by

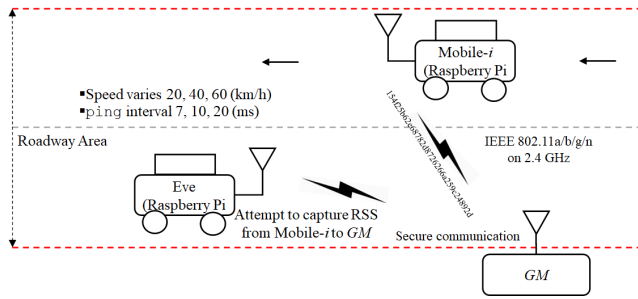


FIGURE 3. Scenario of join protocol Mobile-*i* and GM.

Mobile-*i* and GM simultaneously. Then, GM calculates $A_{ij} = g_1^{d/sx_i} g_1^{-PID_{ij}/s}$. Here, both computed A_{ij} on Mobile-*i* side and GM side are equal.

- To ensure the equality of A_{ij} , hash function is used to guarantee the integrity of A_{ij} together with Q_i . In this case, $A_{ij} = g_1^{d/sx_i} g_1^{-x_i PID_{ij}/sx_i} = g_1^{(d-x_i PID_{ij})/sx_i}$. By sharing the hash value, both Mobile-*i* and GM make sure that A_{ij} and Q_i are kept their integrity.
- Finally, after checking the validity of hash value and ensuring that it is really valid, Mobile-*i* fetches his/her membership secret key $msk[i] = \langle x_i, Q_i, A_{ij}, PID_{ij} \rangle$ and GM stores the secret components corresponding to Mobile-*i* to the database, $REG_i = \langle Q_i, PID_{ij} \rangle$.

E. REVOKING PROTOCOL

Revoke ($grt_{i'}, j, k$): the inputs of this algorithm are revocation token of revoked Mobile-*i'* $grt_{i'j*}$ on $PID_{i'j*}$ on epoch time $j*$ with index $k*$ where $j* \in [1, T]$ and $k* \in [1, K]$ of r -revoked mobile nodes which consist of revocation tokens ($grt_{1j*}, \dots, grt_{rj*}$) elements. The algorithm is executed by GM as follows:

- For revoked Mobile-*i'*, set $grt_{i'j*} = \langle H(k*)^{x_{i'} PID_{i'j*}} \rangle$ and inserts $grt_{i'j*}$ into revocation lists RL .
- Output revocation list RL that consists of all r -revoked mobile nodes' tokens, $grt_{i'j*}$.

F. ANONYMOUS AUTHENTICATION PROTOCOL

Fig. 4 and Fig. 5 illustrate our proposed secret key generation in the anonymous authentication to secure-exchanging important components when a mobile user signs anonymously a message to the mobile verifier. Then, after exchanging the components, mobile user and mobile verifier perform signing and verification anonymous authentication as shown in Fig. 6. In advanced, let say Mobile-*i* and verifier perform SKG process to obtain shared secret key by collecting RSS values via channel probing of 6000 ICMP packets. Again, let assume h_{M_i} is a signal sent from Mobile-*i* to verifier and h_V is signal sent from verifier to Mobile-*i*. An eavesdropper, Eve tries to intercept h_{M_i} from Mobile-*i* and h_V from verifier. The model can be represented as:

$$R_{(M_i-V)_x} = H_{(M_i-V)_x} \times h_{M_{ix}} + N_{(M_i-V)_x},$$

$$R_{(M_i-E)_x} = H_{(M_i-E)_x} \times h_{M_{ix}} + N_{(M_i-E)_x},$$

$$R_{(V-M_i)_x} = H_{(V-M_i)_x} \times h_{V_x} + N_{(V-M_i)_x},$$

$$R_{(V-E)_x} = H_{(V-E)_x} \times h_{V_x} + N_{(V-E)_x}. \tag{5}$$

where $R_{(M_i-V)_x}$, $R_{(M_i-E)_x}$, $R_{(V-M_i)_x}$, and $R_{(V-E)_x}$ are RSS values received by verifier from Mobile-*i*, by Eve from Mobile-*i*, by Mobile-*i* from verifier, and by Eve from verifier, respectively. $H_{(M_i-V)_x}$, $H_{(M_i-E)_x}$, $H_{(V-M_i)_x}$, and $H_{(V-E)_x}$ are the channel gain estimated by verifier, Eve, and Mobile-*i*. $N_{(M_i-V)_x}$, $N_{(M_i-E)_x}$, $N_{(V-M_i)_x}$, and $N_{(V-E)_x}$ are zero mean additive Gaussian noise. Then, randomness extraction is employed to enhance the reciprocity of measured RSS values. Here, we adopted Kalman Filter to enhance the correlation of measured RSS values between Mobile-*i* and verifier. Fig. 7 illustrates the process of Kalman Filter, where z_{l-1} and P_{l-1} are the input parameters with noise measurements R and Q which are predicted in every iteration. The time update used in profiling channel prediction is $\hat{z}_l = Az_{l-1}$ and $\hat{P}_l = AP_{l-1}A^T + Q$. Meanwhile, measurement update used for apriori profiling estimated channel correction is $K_l = (\hat{P}_l H^T) / (H \hat{P}_l H^T + R)$, $z_l = \hat{z}_l + K(y_l - H \hat{z}_l)$, and $P_l = (1 - K_l H) \hat{P}_l$. Where K_l is Kalman Filter gain and z_l is the output correlated RSS values by the index l .

The next step is quantization for converting every single correlated RSS values into bits stream. Firstly, we set two values as threshold of the correlated RSS values: $q+$ and $q-$, where $q+ = \mu + \alpha \cdot \sigma$ and $q- = \mu - \zeta \cdot \sigma$, where μ denotes the mean of RSS values, σ is its standard deviation, and ζ represents a constant, $0 < \zeta < 1$. If RSS values are out of the threshold, they will be omitted. Then, level-crossing is executed to improve the matching bits stream by segmenting RSS values into blocks with particular length (i.e., m -bit), thus the values of each block are either greater than $q+$ or less than $q-$. Here, each m -bit stream of RSS values is evaluated by Mobile-*i* and verifier to determine bits key stream. The main goal utilizing level-crossing is to increase the equality of bits stream in both sides instead of information reconciliation in refining the mismatched bits stream remaining. Hence, as the result, we get a preliminary bits key stream. The next step is utilizing SHA-256 hash function to enhance the randomness of bits key stream in order to pass the NIST pseudorandomness test suite. There are about 3 to 5 winner keys that passed the test. Among the winner keys, the shared secret key is the one that has the highest approximate entropy coefficient. Then, for ensuring the equality of shared secret key between Mobile-*i* and verifier, again SHA-256 hash function is employed to verify it. If it is valid, then the shared secret key will be used to secure the exchanging components. Here, let say the shared secret key is $\gamma \in \mathbb{Z}_q$. Furthermore, shared secret key γ can be involved to secure the secret components in signing and verification processes.

Anonymous authentication protocol is performed by signing mobile node (again, i.e., Mobile-*i*) to verifier mobile node. It consists of two protocols: signing protocol GSign computed by sender entity and verification protocol GVerify computed by receiver entity to verify the signature of sender

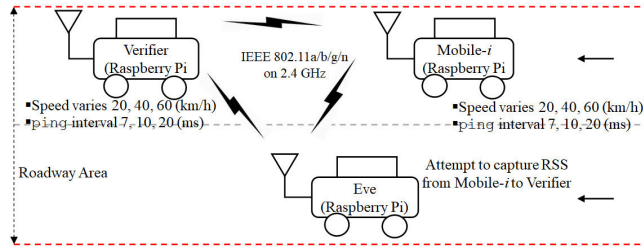


FIGURE 4. Scenario for securing secret components in anonymous authentication protocol.

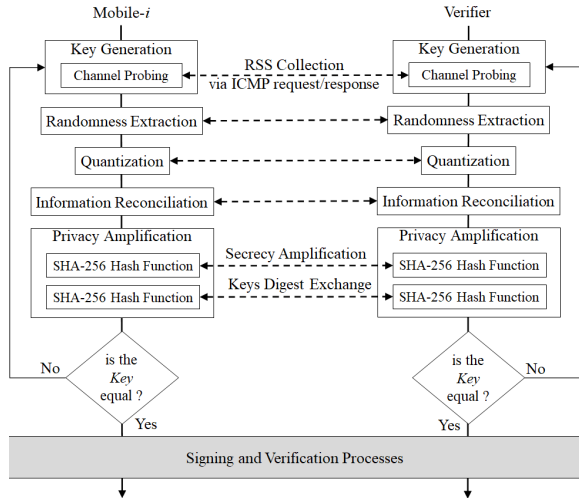


FIGURE 5. Proposed shared secret key generation in the anonymous authentication.

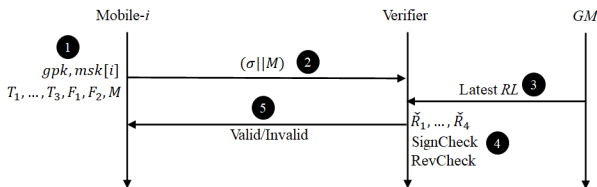


FIGURE 6. Proposed anonymous signing and verification protocol mechanism.

entity. A signature can be successfully verified if and only if verification process declares its validity and the signature is not in the revocation list. Detail of proposed GSign and GVerify protocols are described as follows.

GSign ($gpk, msk[i], j, k, M$): any mobile node (i.e., Mobile- i) in the group can sign an arbitrary message $M \in \{0, 1\}^*$. To create a signature, each mobile node uses his/her private key, hash code of the message, and a random integer to keep each signature randomized. On given a group public key $gpk = \langle D, S, U \rangle$, a Mobile- i 's private key $msk[i] = \langle x_i, Q_i, A_{ij}, h_i, PID_{ij} \rangle$, and a message $M \in \{0, 1\}^*$, the signature can be formed as follows:

- Perform *SKG* process to obtain a shared secret key γ as explained above. Where $\gamma \in \mathbb{Z}_q$.

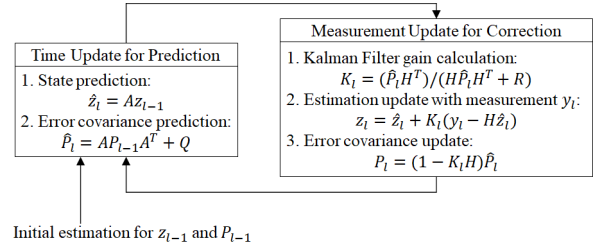


FIGURE 7. Adopted Kalman Filter process to enhance correlation of measured RSS values.

- Select epoch time j on index k and use credential elements $\langle x_i, A_{ij} \rangle$ corresponding to PID_{ij} for selected j on index k , where $j \in [1, T]$ and $k \in [1, K]$.
- Choose $\alpha, \beta \in_R \mathbb{Z}_q^*$ and compute: $T_1 = A_{ij}^{x_i} U^\alpha \in \mathbb{G}_1$ and $T_2 = S^\alpha \in \mathbb{G}_2$. In addition, Mobile- i computes $F_1 = g_1^{VID_i+h_i+\beta} \in \mathbb{G}_1$, $F_2 = U^\beta \in \mathbb{G}_1$, and $T_3 = H(k)^{x_i \cdot PID_{ij} + \gamma} \in \mathbb{Z}_q$ where $H(k)$ is computed hashing of index k and $\gamma \in \mathbb{Z}_q$ is derived from *SKG* process.
- Compute signature of knowledge (*SPK*) Z denoted as follows:

$$Z = SPK\{(x_i, \alpha, \beta, \gamma, PID_{ij}, VID_i, h_i) : e(T_1, S) = \frac{e(D, g_2)e(U, T_2)}{e(g_1^{x_i \cdot PID_{ij}}, g_2)} \wedge T_3 = H(k)^{x_i \cdot PID_{ij} + \gamma} \wedge F_1 = g_1^{VID_i+h_i+\beta} \wedge F_2 = U^\beta\}(M). \quad (6)$$

- Pick blinding factors: $r_{x_i}, r_\alpha, r_\beta, r_\gamma, r_{PID_{ij}}, r_{VID_i}$, and $r_{h_i} \in_R \mathbb{Z}_q^*$, and compute:

$$\begin{aligned} R_1 &= \frac{e(D, g_2)e(U, T_2)}{e(g_1^{r_{x_i} r_{PID_{ij}}}, g_2)}, \\ R_2 &= H(k)^{r_{x_i} r_{PID_{ij}} + r_\gamma}, \\ R_3 &= g_1^{r_{VID_i} + r_{h_i} + r_\beta}, \\ R_4 &= U^{r_\beta}. \end{aligned} \quad (7)$$

- Compute a challenge $c \in_R \mathbb{Z}_q^*$ as: $c = H(gpk, M, j, k, T_1, \dots, T_3, F_1, F_2, R_1, \dots, R_4)$.
- Compute responses: $s_{x_i} = r_{x_i} + cx_i$, $s_\alpha = r_\alpha + c\alpha$, $s_\beta = r_\beta + c\beta$, $s_\gamma = r_\gamma + c\gamma$, $s_{PID_{ij}} = r_{PID_{ij}} + cPID_{ij}$, $s_{VID_i} = r_{VID_i} + cVID_i$, and $s_{h_i} = r_{h_i} + ch_i$.
- Output a group signature: $\sigma = \langle T_1, \dots, T_3, F_1, F_2, c, s_{x_i}, s_\alpha, s_\beta, s_\gamma, s_{PID_{ij}}, s_{VID_i}, s_{h_i} \rangle$, for $j \in [1, T]$.
- Send the signature σ to verifier.

The optional process to secure a-tuple of (M, j, k, σ) through encryption process using shared symmetric key γ obtained from *SKG* process as shown in Fig. 5, this may offer more secure and resistant from eavesdropping and modifications from other parties. Let say Mobile- i encrypts a-tuple of (M, j, k, σ) using AES-256 cryptosystem with shared secret key γ , $C = E_{AES-256}((M, j, k, \sigma), \gamma)$ where C denotes ciphertext of a-tuple of (M, j, k, σ) , γ is shared secret key,

and $E_{AES-256}$ is encryption function of AES-256 cryptosystem. Then, C is sent by Mobile- i to the verifier. On the other side, upon receiving ciphertext C and using shared secret key γ obtained from SKG process, verifier decrypts C to get a-tuple of (M, j, k, σ) by utilizing AES-256 decryption, $(M, j, k, \sigma) = D_{AES-256}(C, \gamma)$ where $D_{AES-256}$ denotes AES-256 decryption function.

GVerify (gpk, j, k, M, σ): on given public key $gpk = \langle D, S, U \rangle$ and a message M , the group signature $\sigma = \langle T_1, \dots, T_3, F_1, F_2, c, s_{x_i}, s_\alpha, s_\beta, s_\gamma, sPID_{ij}, sVID_i, s_{h_i} \rangle$ can be verified as follows:

- Check the $SPK Z$ as follows:
Re-derived $\tilde{R}_1, \tilde{R}_2, \tilde{R}_3$, and \tilde{R}_4 as:

$$\begin{aligned}\tilde{R}_1 &= \frac{e(D, g_2)e(U, T_2)}{e(g_1^{s_{x_i} sPID_{ij}}, g_2)} \cdot e(T_1, S)^{-c}, \\ \tilde{R}_2 &= \frac{H(k)^{s_{x_i} sPID_{ij} + s_\gamma}}{T_3^c}, \\ \tilde{R}_3 &= \frac{g_1^{sVID_i + s_{h_i} + s_\beta}}{F_2^c}, \\ \tilde{R}_4 &= \frac{U^{s_\beta}}{F_3^c}.\end{aligned}\quad (8)$$

- Re-derived the challenge $c' \in \mathbb{Z}_q^*$ as:
 $c' = H(gpk, M, j, k, T_1, \dots, T_3, F_1, F_2, \tilde{R}_1, \dots, \tilde{R}_4)$.
Check that $c' \stackrel{?}{=} c$. Accept σ if the check succeeds and rejects otherwise.
- Revocation Check (RL, j, k, σ): to check whether Mobile- i' is revoked at the epoch time j on index k or not, verifier entity firstly obtains $PID_{i'j}$ of the Mobile- i' at epoch time j on index k by computing $H(k)^{x_{i'} PID_{i'j}} = T_3/H(k)^\gamma$ and the latest RL . Again, $RL = (grt_{1j}, \dots, grt_{zj})$ where $grt_{i'j} = \langle H(k)^{x_{i'} PID_{i'j}} \rangle$ and γ is shared secret key obtained from SKG process. Here, verifier entity searches the value of $grt_{i'j} = H(k)^{x_{i'} PID_{i'j}}$ based on the epoch time j on index k in the RL . In this case, if it is found, Mobile- i' is revoked which means that revocation token of Mobile- i' $grt_{i'j}$ should be in $RL = (grt_{1j}, \dots, grt_{zj})$.

G. OPENING PROTOCOL

Open ($gok, gpk, i, M, \sigma, REG$): this protocol is used for tracing a signature back to the actual signer. The inputs of this protocol are opening manager's private key $gok = \langle u \rangle$ and a signature σ , then opening manager computes the following steps:

- Verify whether σ is a valid signature on a message M or not by executing GVerify algorithm.
- Compute: $Q_i = \frac{F_1}{F_2^{1/u}}$.

The opening manager can then disclose the identity of the vehicle by accessing the above equation, because:

$$Q_i = \frac{F_1}{F_2^{1/u}}$$

$$\begin{aligned}g_1^{VID_i + h_i} &= \frac{g_1^{VID_i + h_i + \beta}}{(U^\beta)^{1/u}} \\ &= \frac{g_1^{VID_i + h_i + \beta}}{(g_1^{u\beta})^{1/u}} = g_1^{VID_i + h_i + \beta} \cdot g_1^{-\beta}.\end{aligned}\quad (9)$$

H. SECURITY ANALYSIS

As well as [25] with respect to construction frameworks discussed in [33], our proposed scheme satisfies the signature correctness and indentity correctness, respectively. We also prove the BU-anonymity, traceability, and exculpability properties under DLIN assumption and DL assumption, respectively. The proofs are provided in Appendix A.

In addition, we evaluate the security property of existential unforgeability under chosen-message attacks [39]. It is defined by using the following sequence games.

- Setup: Let \mathcal{B} be the challenger runs Setup and Join protocols. \mathcal{B} obtains a-tuple $\langle gpk, msk[i], REG_i \rangle$. It gives the adversary \mathcal{A} the resulting gpk and keeps secret the membership secret $msk[i]$. Then, \mathcal{B} performs the following steps:
 - Signature Queries: At any interval time $j \in [1, T]$ at index $k \in [1, K]$, \mathcal{A} issues signature queries to \mathcal{B} M_1, \dots, M_z . Thus, \mathcal{B} is able to sign a chosen message M_l at interval time j with index k , where $l \in [1, z]$.
 - Signing: \mathcal{A} requests a signature σ on chosen message M_l for any random Mobile- i . Then, \mathcal{B} executes GSign protocol $GSign(gpk, j, k, msk[i], M_l)$, obtains a signature σ , and sends it to \mathcal{A} .
- Output: Finally, \mathcal{A} outputs a signature σ^* on a message M_l^* at interval time j^* with index k^* . \mathcal{A} can be said the winner of this game if only if:
 - signature σ^* is successfully verified using GVerify protocol and ensuring whether the signature σ^* is not revoked by using Revocation check algorithm, and
 - \mathcal{A} can not obtain the signature σ^* in making a signing query on message M_l^* .

Even if \mathcal{A} wants to break signature scheme is given $\langle A_{ij}, PID_{i1}, \dots, PID_{iT} \rangle$ for all $i \in [1, n]$, $j \in [1, T]$ and $PID_{ij} \in \mathbb{Z}_q$ at index $k \in [1, K]$, whereas $A_{ij} = g_1^{(d-x_i PID_{ij})/sx_i}$ and PID_{ij} is obtained from SKG process. \mathcal{A} wants to forge secret components $\langle A_{i^*j^*}, x_{i^*}, PID_{i^*j^*}, H(k^*), VID_{i^*}, j^*, k^* \rangle$ by picking randomly $A_{i^*j^*} \in_R \mathbb{G}_1$ and $x_{i^*}, PID_{i^*j^*}, H(k^*), VID_{i^*} \in_R \mathbb{Z}_q^*$, for $j^* \in [1, T]$ and $k^* \in [1, K]$.

Proof: If any value of $\langle A_{i^*j^*}, x_{i^*}, PID_{i^*j^*}, H(k^*), VID_{i^*}, j^*, k^* \rangle$ satisfies the equality of $A_{i^*j^*} = g_1^{(d-x_{i^*} PID_{i^*j^*})/sx_{i^*}}$ and $Q_{i^*} = g_1^{VID_{i^*} + h_{i^*}}$ where $h_{i^*} = H(i^*)$, then the definition of either $e(T_1, S) = e(A_{i^*j^*} U^\alpha, S)$ or $e(T_1, S) = \frac{e(D, g_2)e(U, T_2)}{e(g_1^{x_{i^*} PID_{i^*j^*}}, g_2)}$ equality are satisfied as well.

In this case, \mathcal{A} will successfully forge either if really finds $PID_{i^*j^*}$ equal to PID_{ij} or if x_{i^*} is really equal to x_i such that $A_{i^*j^*} = g_1^{(d-x_{i^*} PID_{i^*j^*})/sx_{i^*}}$. However, since there exists randomly secret components such as α, β, γ , and

blinding factors as well in every signing process, hence this game would be negligible. This is because in GSign protocol, it must compute $T_1 = A_{ij}^{x_i} U^\alpha$, $T_2 = S^\alpha$, and $T_3 = H(k)^{x_i \cdot PID_{ij} + \gamma}$. In addition, it also computes $F_1 = g_1^{VID_i + h_i + \beta}$ and $F_2 = U^\beta$. Here, it needs randomly secret components of α , β , and γ when signing a message M . Moreover, secret key x_i and its PID_{ij} are also involved in the computation. Then, blinding factors have also to be randomly selected: $r_{x_i}^*$, r_α^* , r_β^* , r_γ^* , $r_{PID_{ij}}^*$, $r_{VID_i}^*$, and $r_{h_i}^* \in_R \mathbb{Z}_q^*$ corresponding to $\langle R_1^*, \dots, R_4^* \rangle$. In this case, \mathcal{A} has already requested the query of hashing $H(gpk, j, k, M, T_1, \dots, T_3, F_1, F_2, R_1, \dots, R_4)$, then \mathcal{B} reports failure and terminates the game. This shows that the resulting signature σ is strongly unforgeable, whereas $\sigma = \langle T_1, \dots, T_3, F_1, F_2, c, s_{x_i}, s_\alpha, s_\beta, s_\gamma, s_{PID_{ij}}, s_{VID_i}, s_{h_i} \rangle$, $c = H(gpk, M, j, k, T_1, \dots, T_3, F_1, F_2, R_1, \dots, R_4)$, $s_{x_i} = r_{x_i} + cx_i$, $s_\alpha = r_\alpha + c\alpha$, $s_\beta = r_\beta + c\beta$, $s_\gamma = r_\gamma + c\gamma$, $s_{PID_{ij}} = r_{PID_{ij}} + cPID_{ij}$, $s_{VID_i} = r_{VID_i} + cVID_i$, and $s_{h_i} = r_{h_i} + ch_i$.

IV. IMPLEMENTATION AND EVALUATION

In the implementation, Raspberry Pi acts as an on-board unit (OBU) assembled in every participating user and GM . Fig. 3 illustrates a scenario of interactive join protocol between joining mobile user, let say Mobile- i and GM . Here, we assume GM in stationary position. On the other hand, Fig. 4 represents the scenario when signer user (i.e., Mobile- i) anonymously authenticates him/her self to a verifier user. We assume both Mobile- i and verifier user are in mobile. This scenario is performed by Mobile- i to secure secret components sent to verifier user. In addition, Table 3 shows equipment specifications involved in the implementation. Here, we utilize Python language for developing the system and *tshark-analyzer* [36] for investigating the network traffic.

A. EXPERIMENTAL ENVIRONMENT AND SCENARIO

We start from the scenario of our experiment to evaluate the system performance of join protocol in generating $PIDs$ derived from SKG process. This is an interactive registration process of a joining mobile user let say Mobile- i with speeds vary from 20 km/h to 60 km/h on *ping* time interval 7 ms, 10 ms, and 20 ms, respectively to a GM as illustrated in Table 4. Here, the measurement setting is performed on the road along about 4 km either on the quiet or crowded traffic condition as shown in Fig. 8. We grab about 3000 ICMP packets in total to generate RSS values between Mobile- i and GM through IEEE802.11a/b/g/n 2.4 GHz wireless network standard. Meanwhile, by the same setting with normal traffic condition, we perform SKG process among mobile users let say between Mobile- i and verifier to secure secret components yield in the signing protocol and send securely the secrets to verifier through wireless network. Here, we pick up about 6000 ICMP packets to generate RSS values from these communications. In this case, we also vary the speed of Mobile- i and verifier from 20 km/h to 60 km/h. Our measurements are done through wireless USB adapter TL-WN722N with the involvement of an adversary, say Eve

TABLE 3. Equipment specifications in the implementation.

Specification of	Mobile- i , Verifier and GM
Software	gcc-4.9.2, Python 3.6, libcap-1.8.1, openssl-1.0.1k
O/S	Linux Raspberry pi 4.1.19-v7+
CPU	Raspberry pi 3 model B with ARMv7 1.2GHz processor
RAM	1 GB
NIC	USB Adapter TL-WN722N IEEE802.11a/b/g/n WiFi

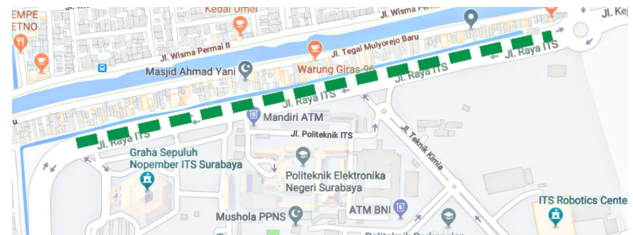


FIGURE 8. Measurement location at Jl. Raya ITS Surabaya.

to always attempt collecting RSS values from the communications either between Mobile- i and GM or between Mobile- i and verifier. Meanwhile, the display of proposed system application in tracking and monitoring mobile nodes is depicted in Fig. 9.

B. MEASUREMENTS OF INTERACTION BETWEEN MOBILE- i AND GM THROUGH JOIN PROTOCOL

As well as [30], we employ reciprocity technique to get a better correlation between RSS values among participating mobile users. We introduced two types of road traffic condition either crowded or quiet traffic. The average of the lowest correlation is about 0.04 when the speed is on 60 km/h in crowded traffic. Whilst, the average of highest one is 0.91 when the speed is on 40 km/h in crowded traffic. On the other hand, the increasing of correlated RSS values upgrades to 0.4 and 0.99, respectively which are illustrated in Table 5. Therefore, we can say that SKG process in this join protocol is working properly.

In the quantization process, RSS values are transformed into bits stream as shown in Table 6. In this case, there are about 1792 output bits and tested their KDR and the key generation

TABLE 4. Scenarios of implementation measurement.

Scenario	Joining Mobile- i 's speed (km/h)	<i>ping</i> time interval (ms)
A	20	7
B	20	10
C	20	20
D	40	7
E	40	10
F	40	20
G	60	7
H	60	10
I	60	20

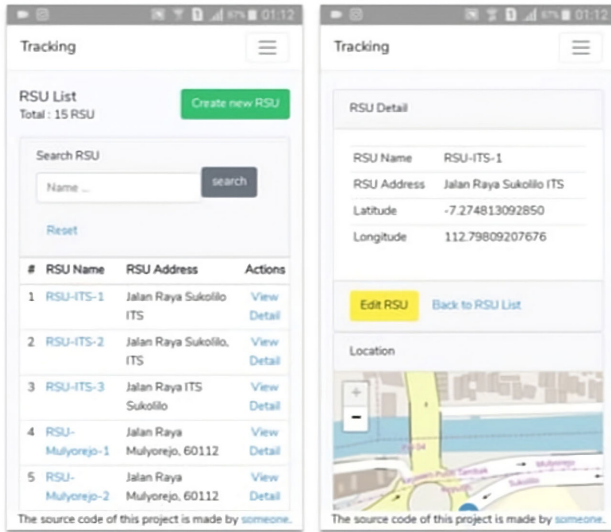


FIGURE 9. Display of proposed system application for mobile users tracking and monitoring.

TABLE 5. Correlations of measurement collected RSS values.

Measurement of speed and traffic condition	Measured RSS	Measured RSS After Reciprocity Technique
20 km/h and quiet	0.445	0.782
20 km/h and crowded	0.781	0.898
40 km/h and quiet	0.897	0.951
40 km/h and crowded	0.910	0.984
60 km/h and quiet	0.119	0.485
60 km/h and crowded	0.042	0.497

TABLE 6. Measurement results of KDR and KGR.

Measurement of speed and traffic condition	KDR (%)	KGR (bit/s)
20 km/h and quiet	53	238
20 km/h and crowded	47	238
40 km/h and quiet	47	238
40 km/h and crowded	46	238
60 km/h and quiet	48	238
60 km/h and crowded	47	238

rate (KGR). The results are smallest KDR can be achieved up to 46% when the speed is on 40 km/h in crowded traffic. Meanwhile, the highest KDR can be achieved up to 53% when the speed is on 20 km/h in quiet traffic. Furthermore, KGR in this measurement can be achieved up to 238 bit/s.

The next step is to increase approximate entropy coefficient such that in satisfying the randomness requirement of shared secret key. To do so, hashing is utilized. In this case, we utilize Universal hash function where measurement result of improving KGR as shown in Table 7. Here, we can show that the hashing is able to contribute to improve the KGR.

A number of generated keys derived from SKG process represents PID_{ij} . These generated keys actually are the winner keys fulfilling randomness requirement as the result of communication between joining Mobile- i and GM. In instant,

TABLE 7. Measurement result of improving KGR by utilizing Universal hash function.

Measurement of speed and traffic condition	Input bits	Number of keys	KGR (bit/s)
20 km/h and quiet	1408	11	109
20 km/h and crowded	1408	11	108
40 km/h and quiet	1408	11	109
40 km/h and crowded	1408	11	108
60 km/h and quiet	1408	11	109
60 km/h and crowded	1408	11	109

TABLE 8. Measurement of number of PID_{ij} .

Measurement of speed and traffic condition	PID Index
20 km/h and quiet	7
20 km/h and crowded	11
40 km/h and quiet	8
40 km/h and crowded	9
60 km/h and quiet	11
60 km/h and crowded	10

the number of generated PID_{ij} for each scenario can be illustrated in Table 8. In this case, we can generate the number of PID_{ij} up to 11. Thus, our setting for the epoch time of PID_{ij} is $T = 11$, whereas $j \in [1, 11]$.

C. MEASUREMENTS OF SKG PROCESS BETWEEN MOBILE- i AND VERIFIER

Again, in this scenario, Mobile- i and verifier conduct SKG process to obtain shared secret key γ in both sides to secure important components used for anonymously signing a message M , such as a-tuple of (M, j, k, σ) . Here, to get reciprocally secret key, we employ Kalman Filter [37] for improving the correlation of measured RSS values among two participating entities as well as [29]. Measurement results say the average of smallest correlation can be achieved up to 0.97 when the speed is on 60 km/h. Meanwhile, the highest one can be achieved up to 0.99 when the speed is either on 20 km/h or 40 km/h. As the impact, the correlated of RSS values is increased up to 0.99 as shown in Table 9. Therefore, we can say that SKG process in this scenario is working properly as well.

Meanwhile, measurements of quantization process as shown in Table 10 show that there are about 3656 output bits which are about 40% of them decreased from 6000 RSS values. However, this decreasing number can be improved by implementing Kalman Filter. Thus, it can be increased up to about 4556 bits. Then, the output bits are evaluated based on the KDR and KGR measurements. As the results, smallest KDR can be achieved perfectly to zero. Whilst, KGR can be achieved up to about 25% improvement.

Table 11 shows the number of winner keys for each measurement in the scenarios. By using our adopted technique, we can achieve the number of winner keys up to 5. This is sufficient to generate shared secret key among two participating

TABLE 9. Correlation scores of RSS values between Mobile-*i*-Verifier, Eve-Mobil-*i*, Eve-Verifier, and Kalman Filter of Mobile-*i*-Verifier.

Scenario of Measurement	Correlation Score			
	Mobile- <i>i</i> -Verifier	Eve-Mobil- <i>i</i>	Eve-Verifier	Kalman Filter of Mobile- <i>i</i> -Verifier
A	0.50	0.06	0.18	0.97
B	0.73	0.06	0.01	0.99
C	0.53	0.21	0.11	0.98
D	0.51	0.32	0.43	0.99
E	0.64	0.27	0.42	0.99
F	0.47	0.06	0.02	0.97
G	0.92	0.27	0.61	0.98
H	0.65	0.21	0.23	0.97
I	0.44	0.16	0.17	0.97

TABLE 10. Measurement results of KDR and KGR.

Measurement	Number of bits (bit)	KDR (%)	KGR (bit/s)
A	998	0	24.05
B	1212	0	21.96
C	1214	0	19.16
D	1403	0	31.79
E	1197	0	21.10
F	987	0	29.15
G	1378	0	32.27
H	1216	0	29.13
I	979	0	31.18

TABLE 11. The number of winner keys measurement and its approximately entropy.

Scenario of Measurement	Number of Winner Keys Index	Average of Approx. Entropy
A	3	0.670
B	5	0.494
C	4	0.765
D	5	0.578
E	4	0.718
F	4	0.682
G	5	0.581
H	4	0.690
I	4	0.693

users when they are authenticating themselves to encrypt the important components such as a-tuple of (M, j, k, σ) .

D. COMPUTATION TIME MEASUREMENTS

Fig. 10 and Fig. 11 show the computation times of proposed system process. Here, total computation time of join protocol as shown in Fig. 10 consists of group computation cost, correlation computation (i.e., polynomial interpolation), quantization, error code correction using BCH codes, Universal hash function, NIST randomness test, integrity check using SHA-256, and communication cost. Group computation cost may include exponentiations in \mathbb{G}_1 , multiplications in \mathbb{G}_1 , and hash function. This computation cost consumes about 20 ms. Whilst, communications cost totally takes about 360 ms. Universal hash function computation and error correction through BCH codes take about 3.58 seconds and 3.27 seconds, respectively. Correlation technique and quantization consume about 1.8 seconds and 3.08 seconds, respectively. NIST randomness test takes about 290 ms and shared

TABLE 12. Computational complexity.

Scheme	Algorithm	$\epsilon()$	$E(\mathbb{G}_1)$	$E(\mathbb{G}_T)$	$\mathcal{O}()$
Sucasas et. al. [21], [23]	Sign	1	10	1	$\mathcal{O}(1)$
	SignCheck	2	12	2	$\mathcal{O}(1)$
	RevCheck	0	0	0	$\mathcal{O}(\log_2 R)$
	Revoke	0	0	0	$\mathcal{O}(\log_2 N)$
SRBE [25]	Sign	3	3	5	$\mathcal{O}(1)$
	SignCheck	4	4	3	$\mathcal{O}(1)$
	RevCheck	0	0	0	$\mathcal{O}(\log_2 R)$
	Revoke	0	0	0	$\mathcal{O}(\log_2 R)$
CLHZ [11]	Sign	5	5	5	$\mathcal{O}(1)$
	SignCheck	7	7	6	$\mathcal{O}(1)$
	RevCheck	0	0	0	$\mathcal{O}(R)$
	Revoke	0	0	0	$\mathcal{O}(1)$
GS-TDL [12]	Sign	4	3	4	$\mathcal{O}(1)$
	SignCheck	9	0	5	$\mathcal{O}(1)$
	RevCheck	0	0	0	$\mathcal{O}(R)$
	Revoke	0	1	0	$\mathcal{O}(R)$
Our Scheme	Sign	6	8	0	$\mathcal{O}(1)$
	SignCheck	4	6	1	$\mathcal{O}(1)$
	RevCheck	0	0	0	$\mathcal{O}(\log_2 R)$
	Revoke	0	0	0	$\mathcal{O}(\log_2 R)$
	Open	0	7	0	$\mathcal{O}(1)$

secret keys verification takes about 80 ms. Therefore, total computation cost is about 12.58 seconds.

On the other hand, similarly the computation cost of *SKG* process for authentication process from Mobile-*i* to verifier shown in Fig. 11 consists of group computation cost, correlation process using Kalman Filter, quantization, level-crossing, randomness using SHA-256, NIST randomness test, shared secret key verification using SHA-256, encryption, and communication cost. Here, the group computation for signing the message *M* is about 90 ms, correlation takes about 2.57 seconds, quantization and level-crossing consume 3.57 seconds and 1.82 seconds, randomness with SHA-256 takes about 60 ms, NIST randomness test takes about 1.12 seconds, shared secret key verification using SHA-256 takes about 50 ms, encryption of a-tuple (M, j, k, σ) consumes about 70 ms, and communication cost is about 120 ms. Hence, total computation cost for this optional *SKG* process in authentication is about 9.53 seconds.

Table 12 shows the complexity comparison of our proposed scheme with existing schemes [11], [12], [21], [23], [25] in term of signing, verification (i.e., signing check and revocation check), and revocation process. Here, signing process

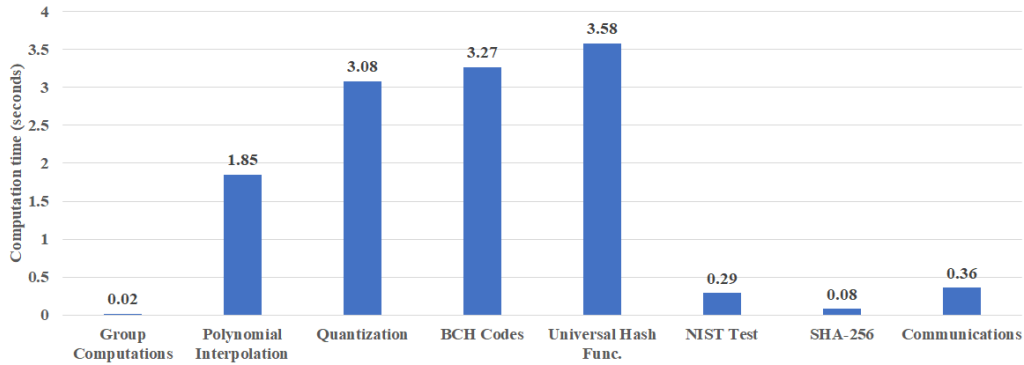


FIGURE 10. Computation time measurement of join protocol.

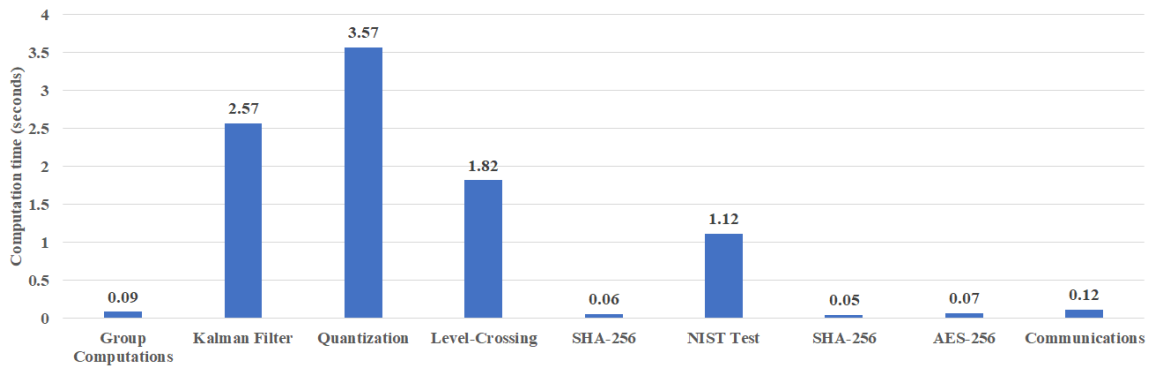


FIGURE 11. Computation time measurement of authentication process from Mobile-i to verifier.

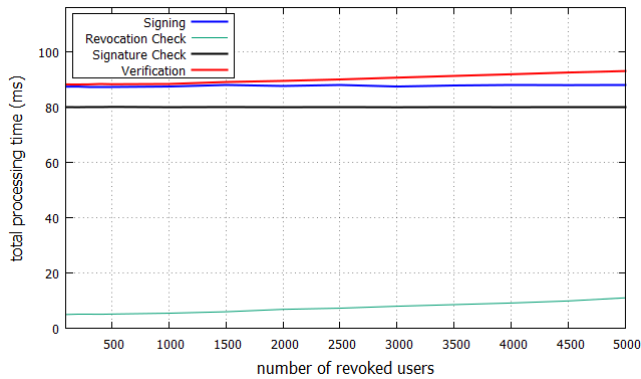


FIGURE 12. Computation time measurement of anonymous authentication system.

in our proposed scheme needs to compute 6 pairings and 8 exponentiations in \mathbb{G}_1 constantly. Meanwhile, in the verification process includes signing check and revocation check. Signing check consumes 4 pairings, 6 exponentiations in \mathbb{G}_1 , and an exponentiation in \mathbb{G}_T constantly. Whilst, revocation check consumes index-based comparison to revocation list RL computation.

Fig. 12 explains computation time measurement of our proposed anonymous authentication scheme based on the variation of revoked user number. In this measurement, we vary

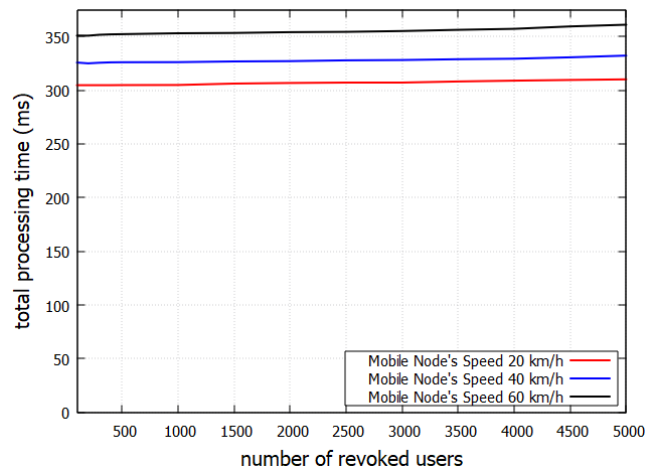


FIGURE 13. Computation time measurement of total anonymous authentication process based on Mobile-i's speed.

the number of revoked users from 100 to 5000. Verification time includes signature check and revocation check processes. Here, signature check is constant about 80 ms in every number of revoked user, meanwhile revocation check time is proportional to the number of revoked users. It varies from about 9 ms to 15 ms when the number of revoked user varies from 100 to 5000 with the total verification time varies from

about 92 ms to 97 ms. On the other hand, at signer user side, signing time is constant about 88 ms.

Fig. 13 describes the total time of anonymous authentication process in every number of revoked user when both Mobile- i and verifier speed at 20 km/h, 40 km/h, and 60 km/h, respectively. Total authentication time includes signing process, communication time, and verification process. It varies from about 304 ms to 310 ms in proportional to the number of revoked users 100 to 5000 when mobile node's speed is 20 km/h. It varies from about 326 ms to 332 ms for the speed 40 km/h and 351 ms to 361 ms for the speed 60 km/h, respectively.

V. CONCLUSION

We have presented an anonymous authentication based on pseudonymous using PIDs generation derived from shared key generation process of measurement collected RSS values in join protocol and the authentication protocol. Adopted SKG process is able to raise shared secret keys which represented as PIDs with zero KDR, high KGR, and better reciprocity. Performance evaluation is done with the traffic condition changing from quiet to crowded and varying the speed from 20 km/h to 60 km/h on ping time interval varied from 7 ms to 20 ms. By 3000 ICMP packets are able to generate up to 11 PIDs and varying speed of mobile users from 20 km/h to 60 km/h on ping time interval from 7 ms to 20 ms. Meanwhile, by 6000 ICMP packets are able to generate 3 to 5 winner keys. Here, quantization algorithm works properly for achieving highest correlation 0.99 and the lowest one is about 0.90. Meanwhile, computation cost requires about 12 seconds in average running on Raspberry Pi. Meanwhile, total processing time of signing and verification processes in the anonymous authentication only consumes about 350 ms including communication costs running on Raspberry Pi.

Future Works. Our future works include more efficient signing, verification, and revocation algorithms of pseudonymous-based anonymous authentication system with involvement of SKG process where PIDs are derived from wireless channel parameters. The implementation of more various applications is also our future works.

APPENDIX A

FORMAL SECURITY OF PROPOSED SCHEME

In this formal security, we consider the definitions and the proofs of features satisfaction of proposed pseudonymous-based anonymous authentication scheme as follows.

A. CORRECTNESS DEFINITIONS

Definition 3 (Signature Correctness): If and only if for all $\langle gpk, REG_i, msk[i] \rangle$ generated by Setup and Join algorithms of every signature generated by Mobile- i at interval time $[1, T]$ using GSign algorithm is valid, then it indicates the correctness of proposed scheme. Meanwhile, GVerify algorithm in interval time $j \in [1, T]$ and index $k \in [1, K]$, whenever Mobile- i is not a revoked user based on the result of Revocation Check algorithm, then it can be

said that signature is correct. Formally, $GVerify(gpk, j, k, M, GSign(gpk, j, k, msk[i], M), M)$ is valid if and only if Mobile- i is not revoked user at interval time $j \in [1, T]$ with index $k \in [1, K]$.

Proof. Here, the correction of Equation 6 executed by Mobile- i to generate a signature σ on a message M can be proven as follows, whereas $T_1 = A_{ij}^{x_i} U^\alpha$, $T_2 = S^\alpha$, and $T_3 = H(k)^{x_i PID_{ij} + \gamma}$. In addition, $F_1 = g_1^{VID_i + h_i + \beta}$ and $F_2 = U^\beta$, thus:

$$\begin{aligned} e(T_1, S) &= \frac{e(D, g_2)e(U, T_2)}{e(g_1^{x_i PID_{ij}}, g_2)} \\ &= \frac{e(g_1^d, g_2)e(U, S^\alpha)}{e(g_1^{x_i PID_{ij}}, g_2)} \\ &= \frac{e(g_1^d, g_2)e(U, g_2^{\alpha s})}{e(g_1^{x_i PID_{ij}}, g_2)} \\ &= e(g_1^d, g_2)e(U, g_2^{\alpha s})e(g_1^{-x_i PID_{ij}}, g_2) \\ &= e(g_1^{d-x_i PID_{ij}}, g_2)e(U, g_2^{\alpha s}) \\ &= e(g_1^{(d-x_i PID_{ij})/s}, g_2^s)e(U^\alpha, g_2^s) \\ &= e(g_1^{(d-x_i PID_{ij})/s} U^\alpha, g_2^s) \\ &= e(g_1^{((d-x_i PID_{ij})/s)x_i} U^\alpha, g_2^s) \\ &= e(A_{ij}^{x_i} U^\alpha, S). \end{aligned}$$

Definition 4 (Identity Correctness): If and only if for all $\langle gpk, REG_i, msk[i] \rangle$ generated by Setup and Join algorithms of every signature generated by Mobile- $i \in [1, n]$ using GSign algorithm in interval time $j \in [1, T]$ with index $k \in [1, K]$, Open algorithm outputs Mobile- i . Formally, $Open(gpk, gok, REG_i, GSign(gpk, j, k, msk[i], M), M)$ outputs Mobile- $i \in [1, n]$ at interval time $j \in [1, T]$ with index $k \in [1, K]$. Where joining Mobile- i whose REG_i is in the group list GL .

Proof: GM runs Open protocol to identify Mobile- i as shown in Equation 9 on inputs valid signature σ and message M . Equation 9 proves that based on the result Q_i which is found in the database GL possessed by GM, the correctness of Mobile- i 's identity can be proven.

B. BU-ANONYMITY DEFINITION

Let \mathcal{A} be an advantage to break proposed group signature scheme run by an adversary and \mathcal{B} be an algorithm to break it run as a challenger. BU-anonymity is the anonymity with backward unlinkability. Where, backward unlinkability means that even after a revocation of a user occurs, the signatures generated by the user are still remain anonymously before the revocation.

Here, Join protocol for exculpability with considering to the following anonymity game.

- Setup: the challenger \mathcal{B} runs Setup protocol. \mathcal{A} is given gpk , then \mathcal{B} runs \mathcal{A} and sets interval time $j = 0$ with index $k = 0$, revoked users list $RL = \emptyset$, and corrected users list $CL = \emptyset$.

- Queries: \mathcal{A} queries the challenger \mathcal{B} by performing the following steps:
 - H-Join: \mathcal{A} requests Mobile- i to join the system using Join protocol. Furthermore, \mathcal{B} computes Join protocol. In this case, \mathcal{B} plays role be both as joining Mobile- i and GM .
 - C-Join: as well as H-Join, \mathcal{A} requests Mobile- i to join the system using Join protocol. Here, \mathcal{A} acts as the joining Mobile- i executes Join protocol, meanwhile the challenger \mathcal{B} acts as a GM . Then, \mathcal{B} adds Mobile- i to CL .
 - Revocation: \mathcal{A} requests revocation process for Mobile- i through Revoking protocol. Here, \mathcal{B} increases j by 1 for index k , adds Mobile- i to RL , and responds the revocation handlers or token for all r -revoked users of RU at interval time j with index k . r denotes the total number of revoked users.
 - Signing: \mathcal{A} requests a signature σ on a message M for Mobile- i using GSign protocol. Then, \mathcal{B} responds the corresponding signature at j and index k , if Mobile- i is not in the CL .
 - Corruption: \mathcal{A} requests secret key $msk[i]$ of Mobile- i . Then, \mathcal{B} responds $msk[i]$ if Mobile- i is not in the CL . Hence, \mathcal{B} adds Mobile- i to CL .
 - Opening: \mathcal{A} requests opening of a signature σ on a message M using Opening protocol. Then, \mathcal{B} responds the identity of Mobile- i as the signer user, if and only if σ is valid.
- Challenge: \mathcal{A} picks a message M and two users: Mobile- i_0 and Mobile- i_1 . If Mobile- i_0 and Mobile- i_1 are not in the CL , \mathcal{B} selects $\phi \in_R \{0, 1\}$ and responds the signature σ on M for Mobile- i_ϕ at current interval $j = j^*$ at index k^* .
- Restricted Queries: \mathcal{A} requests the above queries, but \mathcal{A} is not able to query the corruptions of Mobile- i_0 and Mobile- i_1 , revocation process of Mobile- i_0 and Mobile- i_1 at interval time j^* at index k^* , and opening of the challenged signature.
- Output: finally, \mathcal{A} outputs a result of bit ϕ' that indicating \mathcal{A} 's guess of ϕ . Here, if $\phi' = \phi$, then \mathcal{A} wins. We define the advantage of \mathcal{A} as $|\Pr[\phi' = \phi] - 1/2|$. BU-anonymity requires that for all PPT \mathcal{A} , the advantage of \mathcal{A} on this game is negligible.

C. TRACEABILITY DEFINITION

The proposed group signature scheme can be said traceable, if the probability of winning the following game is negligible for all PPT algorithm \mathcal{A} .

- Setup: Let \mathcal{B} be the challenger runs Setup and Join protocols. \mathcal{B} obtains a-tuple $\langle gpk, msk[i], REG_i \rangle$. \mathcal{B} sends gpk to \mathcal{A} and sets the corrected users list $CL = \emptyset$. Then, \mathcal{B} performs the following steps:
 - Queries: Each interval time j , in the beginning \mathcal{A} announces the starting time of j at index k to \mathcal{B} such that interval time j at index k is synchronous in both sides of \mathcal{A} and \mathcal{B} . Whenever j is incremented for

index k , \mathcal{A} and \mathcal{B} keep them synchronous. At any interval time $j \in [1, T]$ at index $k \in [1, K]$, \mathcal{A} can issue queries to the challenger \mathcal{B} . Thus, \mathcal{B} is able to sign a message M .

- Signing: \mathcal{A} requests a signature σ on a message M for any random Mobile- i . Then, \mathcal{B} executes GSign protocol, $GSign(gpk, j, k, msk[i], M)$, obtains a signature σ , and sends it to \mathcal{A} .
- Corruption: \mathcal{A} requests a secret key of Mobile- i , $msk[i]$. Then, \mathcal{B} adds Mobile- i in CL and responds with $msk[i]$.
- Output: \mathcal{A} outputs a signature σ^* on a message M^* at interval time j^* at index k^* . \mathcal{A} can be said the winner of this game if only if:
 - signature σ^* is successfully verified using GVerify protocol and ensuring whether the signature σ^* is not revoked by using Revocation check algorithm,
 - traces to some Mobile- i^* outside the CL or Open protocol is failed, and
 - \mathcal{A} can not obtain the signature σ^* by making a signing query on message M^* .

Lemma 1: Again, \mathcal{A} be algorithm to break proposed signature scheme is given $\langle g_1, g_1^{a_1}, g_1^{a_2}, g_2, g_2^{a_1}, g_2^{a_2} \rangle$ and $\langle A_{ij}, PID_{i1}, \dots, PID_{iT} \rangle$ for all $i \in [1, n]$, $j \in [1, T]$ and $PID_{ij} \in \mathbb{Z}_q$ at index $k \in [1, K]$, whereas $A_{ij} = g_1^{(d-x_i PID_{ij})/sx_i}$ and PID_{ij} is obtained from SKG process. \mathcal{A} wants to forge secret components $\langle A_{i^*j^*}, x_{i^*}, PID_{i^*j^*}, H(k^*), VID_{i^*}, j^*, k^* \rangle$ by picking randomly $A_{i^*j^*} \in_R \mathbb{G}_1$ and $x_{i^*}, PID_{i^*j^*}, H(k^*), VID_{i^*} \in_R \mathbb{Z}_q^*$, for $j^* \in [1, T]$ and $k^* \in [1, K]$.

Proof: If any values of $\langle A_{i^*j^*}, x_{i^*}, PID_{i^*j^*}, H(k^*), VID_{i^*}, j^*, k^* \rangle$ satisfies the equality of $A_{i^*j^*} = g_1^{(d-x_{i^*} PID_{i^*j^*})/sx_{i^*}}$ and $Q_{i^*} = g_1^{VID_{i^*} + h_{i^*}}$ where $h_{i^*} = H(k^*)$, then the definition of bilinear map either $e(T_1, S) = e(A_{i^*j^*}^{x_{i^*}} U^\alpha, S)$ or $e(T_1, S) = \frac{e(D, g_2)e(U, T_2)}{e(g_1^{x_{i^*} PID_{i^*j^*}}, g_2)}$ equality are satisfied as well. Hence, to prove this assumption, \mathcal{A} demonstrates 2 types of forgers as follows.

- Type 1 Forger: On given any $\langle A_{i^*j^*}, x_{i^*}, PID_{i^*j^*}, H(k^*), VID_{i^*}, j^*, k^* \rangle$ randomly selected and $PID_{i^*j^*}$ at $j^* \in [1, T]$ with index $k^* \in [1, K] \neq PID_{ij}$ at $j \in [1, T]$ and index $k \in [1, K]$ for any $PID_{i^*j^*} \in_R \mathbb{Z}_q^*$, $i \in [1, n]$, $j \in [1, T]$, and $k \in [1, K]$ such that $e(T_1, S) = e(A_{i^*j^*}^{x_{i^*}} U^\alpha, S)$ and $e(T_1, S) = \frac{e(D, g_2)e(U, T_2)}{e(g_1^{x_{i^*} PID_{i^*j^*}}, g_2)}$.
- Type 2 Forger: On given any $\langle A_{i^*j^*}, x_{i^*}, PID_{i^*j^*}, H(k^*), VID_{i^*}, j^*, k^* \rangle$ randomly selected, but $PID_{i^*j^*} \neq PID_{ij}$ for any $i \in [1, n]$, $j \in [1, T]$, and $k \in [1, K]$ such that $e(T_1, S) = e(A_{i^*j^*}^{x_{i^*}} U^\alpha, S)$ and $e(T_1, S) = \frac{e(D, g_2)e(U, T_2)}{e(g_1^{x_{i^*} PID_{i^*j^*}}, g_2)}$.

In this case, Type 1 Forger will successfully forge if really finds $PID_{i^*j^*}$ equal to PID_{ij} . Meanwhile, for Type 2 Forger will successfully forge if $x_{i^*} = x_i$ such that $A_{i^*j^*} = g_1^{(d-x_{i^*} PID_{i^*j^*})/sx_{i^*}}$, but $PID_{i^*j^*} \neq PID_{ij}$ which means that \mathcal{A} can extract PID_{ij} from SKG process. However, since there exists randomly secret components such as α, β, γ , and

blinding factors as well in every signing process and they are kept secret by the signer user, hence this game would be negligible.

D. EXCULPABILITY DEFINITION

The proposed group signature scheme can be said satisfying exculpability feature if no PPT algorithm can forge a signature σ generated by an un-corrupted Mobile- i such that Mobile- i can not dispute. Formally, the probability of winning the following game is negligible for all PPT algorithm \mathcal{A} .

- Setup: The challenger \mathcal{B} runs Setup protocol. Then, \mathcal{B} obtains gpk , gsk , and REG_i . Furthermore, \mathcal{B} stores gpk and sends $\langle gpk, gsk, REG_i \rangle$ to \mathcal{A} . In addition, \mathcal{B} sets a revocation list $RL = \emptyset$. Then, \mathcal{B} performs the following steps.
 - Queries: At the beginning of each interval time j , \mathcal{A} announces it to \mathcal{B} and synchronizes each other. Whenever j is incremented at index k , both \mathcal{A} and \mathcal{B} keep them synchronous. Then, \mathcal{A} can make queries to \mathcal{B} the Join, GSign, and Corruption games.
 - Join: \mathcal{A} requests a new Mobile- i registration by performing Join protocol. Hence, \mathcal{A} obtains $msk[i]$. In this case, \mathcal{A} plays the role as a GM. Then, \mathcal{A} obtains a revocation list RL and adds Mobile- i REG_i in GL . Then, \mathcal{A} performs signing process.
 - Signing: Same as BU-anonymity game.
 - Corruption: \mathcal{A} requests $msk[i]$ of Mobile- i . The challenger \mathcal{B} responds with $msk[i]$. Then, \mathcal{B} updates the current and future revocation lists grt_{ij} , $\forall j \in [1, T]$ at index $k \in [1, K]$ with corresponding revocation list RL .
- Challenge: \mathcal{A} outputs a signature σ^* on a message M^* , interval time j^* at index k^* of Mobile- i^* . It can be said that \mathcal{A} is the winner of this game if:
 - \mathcal{A} does not obtain signature σ^* on message M^* from signing query.
 - signature σ^* verification returns valid.
 - Opening protocol returns the identity of Mobile- i^* and it is found in the group list GL .
 - \mathcal{A} does not corrupt Mobile- i^* .
 - \mathcal{B} can not disclose the secret key $msk[i^*]$ of Mobile- i^* such that \mathcal{A} does not obtain it using $msk[i^*]$.

Proof: Let \mathcal{A} be an adversary who wants to break the exculpability game as above with non-negligible probability. Then, we can construct another PPT algorithm \mathcal{B} to solve DL problem in \mathbb{G}_2 with non-negligible probability.

Meanwhile, in the GSign protocol, Mobile- i computes $T_1 = A_{ij}^{x_i} U^\alpha$, $T_2 = S^\alpha$, and $T_3 = H(k)^{x_i PID_{ij} + \gamma}$. Additionally, Mobile- i computes $F_1 = g_1^{VID_i + h_i + \beta}$ and $F_2 = U^\beta$. Then, let \mathcal{B} selects blinding factors: $r_{x_i}^*$, r_α^* , r_β^* , r_γ^* , $r_{PID_{ij}}^*$, $r_{VID_i}^*$, and $r_{h_i}^* \in_R \mathbb{Z}_q^*$ corresponding to $\langle R_1^*, \dots, R_4^* \rangle$. In this case, \mathcal{A} has already requested the query of hashing $H(gpk, j, k, M, T_1, \dots, T_3, F_1, F_2, R_1, \dots, R_4)$, then \mathcal{B} reports failure and terminates the game. Otherwise, \mathcal{B} defines $H(gpk, j, k, M, T_1, \dots, T_3, F_1, F_2, R_1^*, \dots, R_4^*)$ and generates signature σ^* . Then, \mathcal{B} sends the signature σ^* to \mathcal{A} .

REFERENCES

- [1] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, and A. Lioy, "Efficient and robust pseudonymous authentication in VANET," in *Proc. VANET*, Sep. 2007, pp. 19–27.
- [2] Y. Lindell, "Anonymous authentication," *J. Privacy Confidentiality*, vol. 2, no. 2, pp. 35–63, 2010.
- [3] J. Petit, F. Schaub, M. Feiri, and F. Kargl, "Pseudonym schemes in vehicular networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 228–255, 1st Quart., 2015.
- [4] L. Malina, A. Vives-Guasch, J. Castellà-Roca, A. Viejo, and J. Hajny, "Efficient group signatures for privacy-preserving vehicular networks," *Telecommun. Syst.*, vol. 58, no. 4, pp. 293–311, Apr. 2015.
- [5] L. Malina, J. Hajny, and V. Zeman, "Light-weight group signatures with time-bound membership," *Secur. Commun. Netw.*, vol. 9, no. 7, pp. 599–612, May 2016.
- [6] M. Azees, P. Vijayakumar, and L. J. Deboarh, "EAAP: Efficient anonymous authentication with conditional privacy-preserving scheme for vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2467–2476, Sep. 2017.
- [7] J. Liu, Y. Yu, J. Jia, S. Wang, P. Fan, H. Wang, and H. Zhang, "Lattice-based double-authentication-preventing ring signature for security and privacy in vehicular ad-hoc networks," *Tsinghua Sci. Technol.*, vol. 24, no. 5, pp. 575–584, 2019.
- [8] Y. Sun, Z. Feng, Q. Hu, and J. Su, "An efficient distributed key management scheme for group-signature based anonymous authentication in VANET," *Secur. Commun. Netw.*, vol. 5, no. 1, pp. 79–86, Jan. 2012.
- [9] X. Yue, B. Chen, X. Wang, Y. Duan, M. Gao, and Y. He, "An efficient and secure anonymous authentication scheme for VANETs based on the framework of group signatures," *IEEE Access*, vol. 6, pp. 62584–62600, 2018.
- [10] T. Gao, Y. Li, N. Guo, and I. You, "An anonymous access authentication scheme for vehicular ad hoc networks under edge computing," *Int. J. Distrib. Sensor Netw.*, vol. 14, no. 2, pp. 1–16, 2018.
- [11] C.-K. Chu, J. K. Liu, X. Huang, and J. Zhou, "Verifier-local revocation group signatures with time-bound keys," in *Proc. 7th ACM Symp. Inf. Comput. Commun. Secur. (ASIACCS)*, May 2012, pp. 26–27.
- [12] K. Emura and T. Hayashi, "Road-to-vehicle communications with time-dependent anonymity: A lightweight construction and its experimental results," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1582–1597, Feb. 2018.
- [13] A. Sudarsono, T. Nakanishi, and N. Funabiki, "Efficient proofs of attributes in pairing-based anonymous credential system," in *Proc. 11th Privacy Enhancing Technol. Symp. (PETS)*, in Lecture Notes in Computer Science, vol. 6794. Waterloo, ON, Canada: Springer-Verlag, Jul. 2011, pp. 246–263. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-642-22263-4>
- [14] A. Sudarsono, T. Nakanishi, and N. Funabiki, "A pairing-based anonymous credential system with efficient attribute proofs," *J. Inf. Process.*, vol. 20, no. 3, pp. 774–784, 2012.
- [15] P. Ma, D. Tao, and T. Wu, "A pseudonym based anonymous identity authentication mechanism for mobile crowd sensing," in *Proc. 3rd Int. Conf. Big Data Comput. Commun. (BIGCOM)*, Aug. 2017, pp. 10–14.
- [16] M. Chowdhury, A. Gawande, and L. Wang, "Anonymous authentication and pseudonym-renewal for VANET in NDN," in *Proc. ICN*, Sep. 2017, pp. 220–223.
- [17] C. Shouqi, L. Wanrong, C. Liling, S. Qing, and H. Xin, "An improved anonymous authentication protocol for location-based service," *IEEE Access*, vol. 7, pp. 114203–114212, 2019.
- [18] Y. Shi, Q. Zhang, J. Liang, Z. He, and H. Fan, "Obfuscatable anonymous authentication scheme for mobile crowd sensing," *IEEE Syst. J.*, vol. 13, no. 3, pp. 2918–2929, Sep. 2019.
- [19] Y. Jiang, S. Ge, and X. Shen, "AAAS: An anonymous authentication scheme based on group signature in VANETs," *IEEE Access*, vol. 8, pp. 98986–98998, 2020.
- [20] X. Jia, D. He, N. Kumar, and K.-K.-R. Choo, "A provably secure and efficient identity-based anonymous authentication scheme for mobile edge computing," *IEEE Syst. J.*, vol. 14, no. 1, pp. 560–571, Mar. 2020.
- [21] V. Sucasas, G. Mantas, J. Bastos, F. Damiao, and J. Rodriguez, "A signature scheme with unlinkable-yet-accountable pseudonymity for privacy-preserving crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 19, no. 4, pp. 752–768, Apr. 2020.
- [22] 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, Apr. 2020.

- [23] V. Sucasas, G. Mantas, M. Papaioannou, and J. Rodriguez, "Attribute-based pseudonymity for privacy-preserving authentication in cloud services," *IEEE Trans. Cloud Comput.*, early access, May 27, 2021, doi: 10.1109/TCC.2021.3084538.
- [24] J. Qi and T. Gao, "A privacy-preserving authentication and pseudonym revocation scheme for VANETs," *IEEE Access*, vol. 8, pp. 177693–177707, 2020.
- [25] S. Rahaman, L. Cheng, D. Yao, H. Li, and J. M. Park, "Provably secure anonymous-yet-accountable crowdsensing with scalable sublinear revocation," in *Proc. Privacy Enhancing Technol.*, 2017, pp. 287–306.
- [26] D. Ma, X. Lyu, and R. Zou, "A novel variable K -pseudonym scheme applied to 5G anonymous access authentication," 2021, *arXiv:2106.07158*.
- [27] Y. Sun, R. Lu, X. Lin, X. Shen, and J. Su, "An efficient pseudonymous authentication scheme with strong privacy preservation for vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 59, no. 7, pp. 3589–3603, Sep. 2010.
- [28] M. A. Al-Shareeda, M. Anbar, S. Manickam, and I. H. Hasbullah, "A secure pseudonym-based conditional privacy-preservation authentication scheme in vehicular ad hoc networks," *Sensors*, vol. 22, no. 5, p. 1696, Feb. 2022.
- [29] A. Sudarsono, M. Yuliana, P. Kristalina, and A. R. Barakbah, "An implementation of shared key generation extracted from received signal strength in vehicular ad-hoc communication," in *Proc. 6th Int. Symp. Comput. Netw. (CANDAR)*, Nov. 2018, pp. 57–65.
- [30] A. Sudarsono, M. Yuliana, and P. Kristalina, "A shared secret key generation between vehicle and roadside based preprocessing method," in *Proc. Int. Conf. Comput. Eng., Netw., Intell. Multimedia (CENIM)*, Nov. 2019, pp. 57–64.
- [31] J. Wan, A. B. Lopez, and M. A. Al Faruque, "Exploiting wireless channel randomness to generate keys for automotive cyber-physical system security," in *Proc. ACM/IEEE 7th Int. Conf. Cyber-Phys. Syst. (ICCP)*, Apr. 2016, pp. 1–10.
- [32] C. Zenger, "Physical-layer security for the Internet of Things," Ph.D. dissertation, Dept. Electr. Eng. Inf. Technol., Ruhr-Univ. Bochum, Bochum, Germany, 2017.
- [33] D. Boneh and H. Shacham, "Group signatures with verifier-local revocation," in *Proc. ACM CCS*, Oct. 2004, pp. 168–177.
- [34] W. Stallings, *Cryptography and Network Security*, 7th ed. London, U.K.: Pearson, 2017.
- [35] L. E. Bassham, A. L. Rukhin, J. Soto, J. R. Nechvatal, M. E. Smid, S. D. Leigh, M. Levenson, M. Vangel, N. A. Heckert, and D. L. Banks, "Statistical test suite for random and pseudorandom number generators for cryptographic applications," NIST, Gaithersburg, MD, USA, Tech. Rep. 800-22 Rev 1a, 2010.
- [36] *Tshark—A Network Protocol Analyzer, Tshark-Dump and Analyze Network Traffic*. Accessed: Aug. 22, 2021. [Online]. Available: <https://www.wireshark.org/docs/man-pages/tshark.html>
- [37] G. Welch and G. Bishop, "An introduction to the Kalman filter," Dept. Comput. Sci., Univ. North Carolina Chapel Hill, Chapel Hill, NC, USA, Tech. Rep. TR 95-041, Jul. 2006, pp. 1–16. [Online]. Available: https://www.cs.unc.edu/welch/media/pdf/kalman_intro.pdf
- [38] W. Xu, S. Jha, and W. Hu, "LoRa-key: Secure key generation system for LoRa-based network," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6404–6416, Aug. 2019.
- [39] S. Goldwasser, S. Micali, and R. L. Rivest, "A digital signature scheme secure against adaptive chosen-message attacks," *SIAM J. Comput.*, vol. 17, no. 2, pp. 281–308, Apr. 1988.



AMANG SUDARSONO received the B.E. degree in electrical engineering, telecommunication and multimedia program from the Sepuluh Nopember Institute of Technology, Indonesia, in 2001, and the Ph.D. degree in communication network engineering from Okayama University, Japan, in 2011. From 1997 to 2002, he was at the Network Engineering Division, Metro Cellular Nusantara, Ltd., Indonesia. He joined as a Lecturer with the Division of Telecommunication Engineering, Department of Electrical Engineering, Politeknik Elektronika Negeri Surabaya (Electronics Engineering Polytechnic Institute of Surabaya), Indonesia, in 2002. His research interests include privacy-enhancing authentications (group signatures), network and information security, and cryptography.



MIKE YULIANA (Member, IEEE) was born in 1978. She received the bachelor's, master's, and Ph.D. degrees in electrical engineering from the Sepuluh Nopember Institute of Technology, Surabaya, Indonesia, in 2001, 2007, and 2019, respectively. Her research interests include wireless communication, cryptography, and physical layer security.