IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 8, 2019, accepted July 16, 2019, date of publication July 25, 2019, date of current version August 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2931217

A Mutual Authentication Scheme for Secure Fog Computing Service Handover in Vehicular Network Environment

FAVIAN DEWANTA[1,2], (Member, IEEE), AND MASAHIRO MAMBO[3], (Member, IEEE)

[1]Division of Electrical Engineering and Computer Science, Graduate School of Natural Science and Technology, Kanazawa University, Kanazawa, Japan
[2]School of Electrical Engineering, Telkom University, Bandung, Indonesia
[3]Faculty of Electrical, Information and Communication Engineering, Institute of Science and Engineering, Kanazawa University, Kanazawa 920-1192, Japan

Corresponding author: Favian Dewanta (favian@telkomuniversity.ac.id)

The work of F. Dewanta was supported in part by a BUDI-LN Scholarship from the LPDP (Indonesia Endowment Fund for Education) and the Kementrian Ristek-Dikti (Ministry of Research, Technology, and Higher Education) of Republic of Indonesia, and also in part by the Telkom University.

ABSTRACT Handover schemes play a vital role on fog computing service (FCS) provided through vehicular network. It not only determines the quality of services (QoSs) but also the security and safety of vehicular network system against adversaries. As a part of handover process, authentication between vehicles and a new fog node (FN) significantly contributes to protecting private information and infrastructure of vehicular network at once. In this paper, we propose a lightweight and secure mutual authentication scheme for handover process considering limited access FCS in the vehicular network environment and also service reservation scenario at login and service request phase. In the proposed scheme, mutual authentication process is assisted by a cloud server (CS) during login and service request phase in which CS distributes the credentials for on-the-road authentication between the vehicles and FN installed on road side unit (RSU). We demonstrate that our proposed scheme is lightweight due to employing one-way hash function and exclusive-or operation extensively. In addition, our scheme is efficient in terms of computational cost as well as computation cost. We show that our scheme achieves 1.1–56.67 times faster computation and also reduces the total message size by 30%–58.21% in comparison with the previous authentication schemes in the most relevant environment. The informal and formal security analyses show that this authentication scheme can protect the secrecy of transactions of all interacting entities against various known attacks. In addition, validation using SPAN software based on AVISPA also confirms that the proposed authentication scheme can satisfy mutual authentication goal and, at the same time, also protect against replay and man-in-the-middle attack.

INDEX TERMS Fog computing, mutual authentication, vehicular network.

I. INTRODUCTION

The next generation of autonomous cars and intelligent transportation systems (ITS) are predicted to generate gigabytes or even terabytes of data every day. These data are generated by several sensors and actuators in the cars, e.g. proximity-sensor, camera, and GPS, and also some distributed packages of sensors and actuators that are located in a certain part of roads as traffic or weather monitoring systems [1]. These data are consumed by users, government, or even several companies to increase safety, comfort, and driving convenience through some strict data sharing procedures protecting privacy of each party.

In terms of real time application and Big Data processing system in ITS, relying only on cloud computing system is not enough due to long latency and limited bandwidth. In this case, edge computing, like cloudlet [2], fog computing [3], etc., is needed as the complement and proxy of cloud server at the same time, and also for guaranteeing response time of application. Moreover, the evolution of network technology makes inter-work among multiple vehicles, edge nodes, and cloud servers feasible and apparent as stated in surveys [4], [5].

As a part of vehicular network system, handover process for fog computing service (FCS) should be designed in

The associate editor coordinating the review of this manuscript and approving it for publication was Maurice J. Khabbaz.

a secure and real-time way because it can lead to catastrophic loss if adversaries can spoof and also expose some credentials and private information which are exchanged by vehicles and fog node (FN). Especially, mutual authentication between vehicles and FN is considered as one of important procedures for authorizing FCS handover in the vehicular network environment. Therefore, in this paper, we propose a lightweight and secure mutual authentication scheme for guaranteeing the legitimacy of both vehicles and FN even under various known attacks.

Even though several researchers have already mentioned mutual authentication scheme and proposed their algorithms in the vehicular network as explained later in Section II, the discussion of mutual authentication and session key generation is still needed due to the existence of weakness found by security analysis of their schemes as shown more detail in Section VI and VIII. Major advantages of our proposed scheme over their schemes are twofold; high security and efficiency. As for the former, we conduct comprehensive security analysis and show that only the proposed scheme does not have security flaws in any one of attacks analysed in Table 5. As for the latter, our scheme extensively employs one-way hash function and exclusive-or operation so as to deliver better performance than previous work in terms of computational cost and communication cost as explained in Section VIII. In other words, we can argue that our work can provide more lightweight mutual authentication scheme for secure FCS handover which is needed by real time applications in vehicular network.

A. CONTRIBUTIONS

Our work contributes a number of prominent features as listed below.

- First, we incorporate a FCS reservation mechanism in our design of secure FCS Handover by taking into account intrinsic properties of FCS in vehicular network, i.e. limited computational resources in a FN and the need of FCS reservation regarding specifications of computational resources.
- Second, we put FN on RSU/eNode-B and avoid utilizing RSU as an internet gateway to an application server. We introduce limited access FCS in vehicular network environment, i.e. FN providing FCS to vehicles isolates vehicular network from public network/internet, with the purpose of providing secure environment against outsider attacks and supporting low latency applications in vehicular network.
- Third, we propose a lightweight mutual authentication scheme that can enable secure service handovers following vehicle's movement by employing one-way hash function and exclusive-or extensively.
- Fourth, we provide comprehensive analysis for evaluating our proposed scheme by conducting formal security analyses, e.g. random oracle model, BAN logic, and AVISPA, informal security analysis against various known attacks, and efficiency analysis with respect to the most relevant references.

B. DEMONSTRATION OF SECURITY PROPERTIES

Eventually, we elaborate and demonstrate security properties of our proposed scheme as given in the following list.

- We provide formal security analysis to show that secrecy is kept well in Section IV and also that mutual authentication can be reached by means of BAN Logic in Section V.
- By informal security analysis in Section VI, we show that our authentication method is still secure against several known attacks, such as arbitrary guessing attack, user/vehicle/fog node impersonation attack, stolen-OBU/vehicle attack, replay attack, and also combination of stolen verifier, stolen-OBU and sniffing attack.
- Unlike previous schemes, we introduce dynamic credentials (service tag $S_{i,t}$ and initial key K_x) which can make adversaries need more effort to reveal private information and to track vehicle's path based on sniffed messages. Moreover, by employing these dynamic credentials, we also show that this proposed scheme is secure against ephemeral secret key leakage (ESL) attack based on CK adversary model, and at the same time preserves forward secrecy as shown in Section VI.
- We also perform protocol verification in Section VII to validate our design functionality and security by using SPAN [6] which is based on AVISPA (Automated Validation of Internet Security Protocols and Applications) tool [7]. In fact, AVISPA has been widely used in industry and academia to verify, validate, and show security weakness in the protocol specification written in HLPSL (High Level Protocol Specification Language) [8]–[10].

C. FCS IN VEHICULAR NETWORK ENVIRONMENT

We recognize that the concept of FCS in vehicular network still enables vehicles to access cloud server (CS) for non-real-time big data processing. This concept has an advantage in a sense that vehicles can have more options of services from CS. However, it also has a drawback in terms of exposing vehicles to a huge number of internet users that potentially can threaten vehicle's critical systems affecting safety of passengers or driver. Thus, we design limited access FCS in vehicular network as a base environment of our proposed mutual authentication scheme by authorizing RSU's access control policy that can block internet connection from/to vehicles.

Considering a secure FCS handover in vehicular network, we can find similar work proposed by Yao *et al.* [11]. Their work enables fog computing service hosted by other vehicle with rich computational resources by means of VF (vehicular fog) construction and VFS (vehicular fog service) access method. However, their work presents more discussions on reliability aspects and less considerations on security. It provides only informal security discussion about confidentiality,

integrity, and non-repudiation. In addition, the authors do not elaborate their countermeasure strategy in detail especially for their authentication method.

FCS in vehicular network is not only incorporated by vehicle computational resource, but also established by fog node (FN) installed in road side unit (RSU)/base station as discussed in Li *et al.* [12]. In such a service migration scheme as explained in [12], a vehicle and a new fog node (FN), i.e. we can also say RSU/eNode-B in other way, should verify each other prior to resuming FCS in an insecure network. Therefore we can adopt the existing authentication schemes between vehicle and RSU in vehicular network [13]–[16] in order to craft authentication scheme for RSU/eNode-B-based FCS. Note that [12] does not discuss authentication scheme in the investigation of service migration scenarios.

However, to comply with our own limited access FCS, we design online login and service request phase in order to (1) verify user and create dynamic credentials for mutual authentication phase at the same time, (2) reserve FCS related to base layer and instance layer things, and also (3) deploy those credentials to a number of FNs which a vehicle will pass by. Note that we assume the vehicle is owned by a user requesting FCS. It is worth mentioning this phase because to support faster stateful service handover we need to deploy specific base layer, i.e. guest OS, kernel, etc., based on user request to the assigned FNs in the beginning of FCS so that service handover only requires to transfer instance layer, i.e. application, database, etc., as discussed in [17] and [18]. In the previous work of service migration, usually it is assumed by the authors that base layer has already been deployed at the beginning prior to having FCS. In a real practice, specifications of base layer can be different for each user, i.e. memory size, disk/storage size, CPU type, etc., depending on quality of services (QoS) requested by users. In addition, as a matter of fact, FN's computational resources are far more limited in comparison with CS. As a consequence, FN's computational resources can be allocated only for active vehicles on the road. In such a situation, it is important for users to reserve FCS in login and service request phase prior to the start of driving with their vehicles. Thus, to fit with the intrinsic properties of FCS, these practical things should be considered in the design of our proposed authentication scheme especially in the login and service request phase. Not only a matter of handover speed, this difference on login and service request phase also consequently increases security against ephemeral secret leakage attack as discussed in Section VI.

As a summary, Table 1 shows comprehensive information of environmental usage and underlying cryptographic functions. As far as authors know, only our secure FCS handover considers limited access FCS environment and service reservation scenario to reflect real condition of FCS implementation in vehicular network. As explained in Section II, references [13]–[16] show the most relevant schemes to ours and there are three similar properties with ours in terms of authentication techniques, which are (1) usage of vehicle to infrastructure (RSU) network, (2) authentication between vehicle and RSU, and (3) usage of Trusted Authority (TA) for setting up authentication. We can also observe dissimilarities with ours which affect the design, security and efficiency of our mutual authentication scheme for secure FCS handover.

TABLE 1. Comparison of Environment, Approach, and Underlying Cryptographic Function.

Schemes	E1	E2	E3	E4	E5	E6	E7	E8	E9
Our	✓	✓	×	✓	✓	✓	✓	✓	×
[13]	✓	✓	✓	✓	×	×	×	✓	✓
[14]	✓	✓	✓	✓	×	×	×	✓	×
[15]	✓	✓	✓	✓	×	×	×	✓	×
[16]	✓	✓	×	✓	×	×	×	✓	✓

E1: Vehicle to infrastructure (RSU) network; E2: Authentication between Vehicle and RSU; E3: Enabling RSU as an internet gateway for vehicle; E4: Utilizing Trusted Authority (TA); E5: Enabling fog computing service (FCS); E6: Service Reservation; E7: Limited Acces FCS; E8: One-way hash function; E9: Elliptic curve cryptography; ✓: Satisfied; ×: Not satisfied.

II. RELATED WORK

Authentication schemes have been used for establishing secure vehicular network. They become fundamental requirement to enable reliable and trusted communication among involved parties in vehicular networks [19]. Moreover, in the time of huge growing of ad-hoc network and IoT technology, a lot of devices with tiny computation power are easily connected and at the same time increasing security issues caused by network intrusions and passive attacks. As a consequence, authentication schemes are considered as the main requirement of many applications in vehicular network [20].

Various authentication techniques have been studied and proposed in the past. They can be categorized into asymmetric and symmetric cryptography-based authentication, which are PKIC-based authentication, ECDSA-based authentication, MAC-based authentication, hash function-based authentication, and TESLA-based authentication [21]. As for VANET authentication scheme, it can be categorized into ID authentication, property authentication, and location authentication [22]. Among those mentioned authentication techniques, we intentionally select ID authentication by using hash function as our main approach to establish lightweight trusted communication and also create new session key in vehicular network application.

In recent years, there are several papers that already discussed and had some similarity issues related to our work. Li *et al.* [23] propose efficient and secured dynamic identity-based authentication protocol for multi-server architecture using smart card. Their work attempts to remedy the previous work that contains some dangerous flaws, such as vulnerable to leak-of-verifier attack, stolen smart card attack and impersonation attack. In another network domain, Xue *et al.* [24] propose temporal-credential-based mutual

authentication between user, gateway node, and sensor by using smart card as user's verifier. The authors promote the concept of temporal credential stored inside the smart card to protect the private information of the user. The authors claim that their approach is slightly better as compared to the previous research in terms of providing mutual authentication and key agreement by using hash value instead of directly using plain password and user ID. To the best of our knowledge, even though the work of Xue *et al.* [24] offers more privacy protection feature in comparison to Li *et al.* [23], it still fails to address efficient password change mechanism which requires user to resubmit credential to the gateway node and then receive new smart card. Meanwhile, in [23] user only needs to put smart card on card reader, login, and type a new password in order to create the new password. Thus, our work attempts to combine their advantages into one scheme which promote temporal credential usage and at the same time provide efficient mechanism through personal device application as elaborated in Section III.

One of ideas in using out-of-band channel to conduct authentication in wearable devices is proposed by Liu *et al.* [25]. They employ the QR code to encode a set of random numbers, key, and message verifier on wearable device. The scheme is able to create a secure and private channel between wearable device and mobile terminal and also feasible to implement it by using android OS based mobile phone. Unfortunately, the authentication process takes long times (4.2 - 4.6 *seconds*) due to the process of encoding and decoding QR codes. Thus, we drop QR code-based authentication approach for the sake of providing real time and secure service handover in our scheme.

In the case of IoT environment, authentication and key generation are often discussed under multi-server or multi-gateway environment for the sake of establishing redundancy and fault tolerant scheme. As for the example, Wu *et al.* [26] discuss the authentication and key agreement method for multi-gateway wireless sensor network environment. They claim that the previous paper by Amin and Biswas [27] is not discussing all possible attacks. It turns out that there are more attacks that should be considered, such as sensor capture attack, user forgery attack, gateway forgery attack, sensor forgery attack and off-line guessing attack. By using their authentication method, they can show that their method is effective against those attacks. Another related work is conducted by Kumari *et al.* [28] by designing a provably secure biometrics-based multi-cloud server authentication scheme. This work reveals that secure user authentication schemes for multi-cloud-server are considered as an open issue and challenge because the previous works [26], [27] still have not solved crucial issues, such as user impersonation attack, server spoofing attack, denial-of-service attack, and also perfect forward secrecy under the multi-cloud server situation. It is worth to mention that both of [28] and [26] are not compatible to be adopted into our case concerning aspect of security, latency, and computational resource. As for [28], their multiple cloud server authentication scheme is not appropriate to our case considering the number of vehicles in the road and also the number of FNs' computational resources installed in RSU. Suppose each vehicle possesses multiple FNs' computational resources at the same time. Then, if the number of vehicles is bigger than or equal to the number of available FNs' computational resources, other vehicles are unable to access computational resources of FNs. As long as each vehicle processes computational resources of one FN, we can expect that such a problem does not occur. Thus, to our best knowledge constructing one vehicle - one FN server scheme is more appropriate to be applied in vehicular network-based FCS as discussed in our proposed scheme. As for [26], their idea of multi-gateway for IoT environment is fit for cloud computing service based scheme in which all of the application servers are available through internet connection but at the same time can cause serious damage to the vehicular network system. In our opinion, accessing application server through internet can disrupt vehicle's real time application due to high latency and long delay. Moreover, data exchange between vehicles and application servers through the internet can be easily sniffed and analyzed by outsider entities in which they can possibly reveal private information or even intrude the system to gain control over vehicles or important data. Thus, establishing FCS and isolating vehicular network system from internet access are necessary as discussed in Section III.

Yao *et al.* [11] discuss a method for enabling reliable and secure fog computing service provision in vehicular network. Their proposed scheme defines three-layered system framework, i.e. trusted authority (TA), RSUs, and OBUs of vehicles, and two methods, i.e. VF construction and VFS access method, in order to realize their purpose. In the latter part, they show that their proposed scheme is relatively lightweight to be applied in latency-sensitive vehicular fog computing environment. However, the authors do not present enough security analyses on their work except only one section of informal security analysis in order to show that their work is able to provide confidentiality, integrity, and non-repudiation properties. Instead of security analysis, the authors seem to have more concern on reliability analysis by giving more discussion of network analysis with respect to computational and communication cost, and throughput analysis. Since their work does not give any detail description on authentication protocol part, e.g. message size, variable size, method for verifying, etc., unfortunately we cannot check their protocol and compare the components of their protocol with the components of our proposed protocol and also the components of other related work. Therefore, their work is not included in the discussion of performance evaluation of related work.

Even though one seems to notice that our work discusses similar issues to Wazid *et al.* [14] and Dua *et al.* [13], we can argue that our work is unique in terms of network architecture and paradigm of vehicular network. In [14] and [13], the authors only mention three types of mutual authentications which are 1) between vehicles; 2) between vehicles and their respective cluster heads; and 3) between cluster

heads and RSU. In other words, connection between each vehicle and RSU is limited only through cluster heads. Moreover, their concept of vehicular network utilizing RSU as the gateway for accessing application server through internet is similar to the previous proposal of multi-gateway in IoT environment which can cause severe issue as mentioned in early paragraph. As a result, this network model is not appropriate for FCS in the vehicular network environment by considering the need of direct connection between each vehicle and FN and also the real time and security aspect of vehicles concurrently.

Despite utilizing the similar approach of mutual authentication, our work is different from the work by Mohit *et al.* [15] in terms of authentication mechanism. In [15], the purpose of authentication is to collect data from vehicle sensor with the auxiliary of sink node in the middle of communication between user and vehicle sensor. Their scheme indeed works differently to our proposed scheme in which authentication occurs directly between vehicle and FN prior to conducting FCS handover. Furthermore, we find some weaknesses in [15] in comparison to our proposal as discussed in Section VIII-C.

Our work is also different from Feng *et al.* [16] in terms of authentication usage and technique. Our work simply utilizes hash and XOR function in order to verify entities due to the need for real time applications. In contrast, Feng *et al.* [16] utilizes ECDSA for detecting and preventing multi-source Sybil attack in vehicular network.

Eventually, the environment and properties of our work with regard to the most related work [13]–[16] are described in Table 1. As for latter part, the performances of our proposed scheme regarding security features, computational cost, and communication cost are shown in Section VIII.

III. PROPOSED MUTUAL AUTHENTICATION AND SESSION KEY GENERATION

A. NEED FOR SECURE AND LOW LATENCY NETWORK

The current vehicles not only rely on the works of their own sensing and control system, but also have to coordinate with other vehicles to perform certain difficult driving tasks such as lane changing maneuvers, emergency brake, overtaking other vehicle, and others. Therefore, communication among vehicles and other environment supporting system should be done in a real-time manner, e.g. less than 100 *ms* for some critical systems as mentioned in [29]. In such a condition, placing application server and supporting system closer to vehicles as the proxy of cloud server (CS) is more beneficial for delivering low latency service. Moreover, several works already mentioned that fog/edge computing is planned to support several services in vehicular network [30]–[32]. In addition, by separating base VM (virtual machine) and VM overlay as being done by [18] in cloudlet case, VM overlay handover between edge computing server operated by Openstack [33] can be transferred in a more efficient way.

In our scenario, fog computing service (FCS) is not provided by other vehicles as discussed by Hou *et al.* [34] and Yao *et al.* [11], but rather we consider to put FCS on RSU/eNode-B as described by other researchers in their work [4], [5], [21]. The advantages for selecting RSU/eNode-B as the place for installing FCS are twofold, enabling to install higher computational resources with respect to vehicle computational resources and availability of existing 4G-LTE coverage infrastructure owned by telecommunication service provider. In addition, service migration in fog computing enabled 4G-LTE network following vehicle's movement has been discussed for several scenarios by Li *et al.* [12].

Besides low latency criteria, vehicular network-based control system should provide a high level of security particularly for critical system related to the safety of driver and passengers inside vehicles. We can argue that isolating this critical system from outsider or internet user can limit several numbers of potential attacks and increase security level at the same time. This approach is commonly used in industrial automation control system of manufacture/power plant by creating network air gap or installing firewall between corporate network and process/control/field area network [35]. As analogous to the industrial network security approach, instead of accessing application server in internet network, in our proposed scheme vehicles can only access fog computing service (FCS) installed in road side unit (RSU)/eNode-B located within local network of telecommunication service provider as the basis of vehicular network. This limited access FCS property is conducted by RSU by means of network access policy scenario. As a consequence, vehicular network application can be conducted in a secure and real-time manner.

B. OVERVIEW OF PROPOSED SCHEME

In our vehicular network environment, mutual authentication and session key generation come into action when vehicles encounter a new FCS coverage. To make it clear, suppose FNs with high computational power have been installed along the road as shown in Figure 1. Each of FNs is connected to several RSUs so that every vehicle can access and use the FCS through local vehicular network, e.g. IEEE 802.11 a/b/g/n/p, IEEE 1609, 4G-LTE, 5G, etc. Then, to keep using the FCS, vehicles need to conduct mutual authentication and session key generation with a new FN. After successfully conducting handover process, then the former service in the former FN_1 is deleted in order to enable other vehicles to use computational resources of FN_1 as shown in Figure 1.

Prior to elaborating into further detail, we present overview of the proposed mutual authentication scheme as listed below.

- **Registration phase**: Users register their vehicles by using personal devices to cloud server (CS). Note that CS is also capable to control and manage FNs in the background system in order to serve FCS for vehicular network. Then, CS issues some credentials to users for verifying login process and service request through their personal devices.
- **User login and service request phase**: Users verify their previous registered account through their

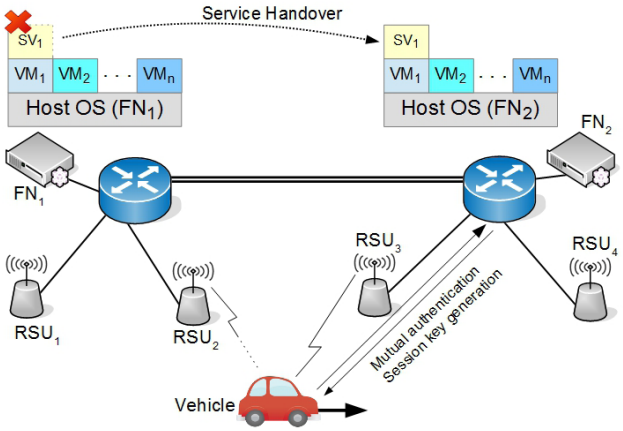

FIGURE 1. FCS handover on vehicular network environment.

personal devices and request FCS to CS after successfully being authenticated by the system. Upon the request, CS returns credentials $(K_x, S_{i,t})$ to the user. Then user also processes them further before transferring new calculated credentials $(VS_{i,t}, VK_x)$ to vehicle. At the same time, CS sends other credentials (HS, HV, K_x) to assigned FNs for later mutual authentication with vehicle.

- **Mutual authentication and session key generation phase**: After receiving credentials from users, vehicles communicate with an FN installed close to several RSUs and conduct mutual authentication and session key generation prior to having FCS.
- **Service termination phase**:Users can terminate the FCS anytime and also request for its service log through their personal devices.
- **User and password change phase**: Users are able to change their identity and password by using their personal devices and send them to CS.

In practice, all of computations and transactions between users and CS are done by personal devices. It is appropriate to use personal devices in the current life style in which people utilize their personal devices to access and manage many things for the sake of convenience. Users only need to input user's identity, password, and vehicle's identity, and also define FCS that is going to be used while driving. Then, we can say that personal devices are representation of the users and consequently the terms will be mentioned interchangeably in the following sentences. For example, by saying users store variables from CS, it means that their personal devices keep those variables into their storage/memory card.

In addition, depending on the context, FN means services/application servers from one of several VM instances/containers that are installed inside the FN. We elaborate to describe details of each procedure below by using symbols shown in Table 2. To construct an efficient scheme, we extensively use a cryptographically secure one-way hash function $h(\cdot)$ as utilized by previous schemes [14], [15], [21].

In order to successfully conduct mutual authentication between V_i and FN_i, at first it should be infeasible for adversaries to compromise CS. All communication channels between CS and U_i and also between CS and FN_i should be tamper-proof and invisible by any adversary. U_i can transfer credentials $VS_{i,t}$ and VK_x in a secure manner by means of direct typing to OBU's interface and/or an encrypted short range wireless protocol. By ensuring that all credentials can be received by both V_i and FN_i as mentioned previously, this authentication protocol between V_i and FN_i can verify both V_i and FN_i and exchange variables K_y and K_z for crafting session key. Note that both V_i and FN_i know the formula to calculate session key $SK_f = SK_v = h(HV||K_y||K_z)$ in which SK_f, SK_v, HV, K_y, K_z are session key calculated by fog node, session key calculated by vehicle, hash of VID_i, key calculated by vehicle for crafting session key, and key calculated by fog node for crafting session key respectively.

Note that, CS is used as an integrated part of our mutual authentication scheme due to limitation of computational resources that can be installed in RSU/eNode-B. In our concept of FCS for vehicular network, FN_i only allocates computational resources for active vehicle V_i in the road. At the same time, we also limit V_i from accessing CS through

TABLE 2. Symbols Used in Our Proposed Scheme.

Symbol	Description
U_i	i-th user
CS	Cloud server
FN_i	i-th Fog Node
V_i	i-th Vehicle
UID_i	Identity of i-th user
UPW_i	Password of i-th user
VID_i	OBU identity of i-th Vehicle
$S_{i,t}$	Service tag of i-th user at time t
CID_i	Identity of i-th user assigned by the cloud server
RN_i	Nonce generated by i-th user for registration process
LN_i	Nonce generated by i-th user for login process
N_v, N_f	Nonce for authentication and session key generation
SK_f/SK_v	Session key calculated by fog node / vehicle
K_x	Initial key prior to conducting authentication
K_y	Key calculated by vehicle for crafting session key
K_y	Key calculated by fog node for crafting session key
$t_1, t_2, t_3,$ t_4, t_5, t_6	Timestamp
X^*	Input/received value of X fed in checking process
X'	Computed value of X fed in checking process
$h(\cdot)$	One-way hash function
$\|\|$	Concatenation
$\oplus$	XOR operation
$\mathcal{A}$	Adversary
$\dashrightarrow$	Insecure channel
$\longrightarrow$	Secure channel

internet since V_i is critical system in which its performance can affect safety of user U_i if data flow from and to V_i are exposed through internet access. Thus, it is better to keep V_i in local vehicular network to limit possible outsider attacks. As a consequence, prior to enabling FCS, U_i conducts login and service request phase through his/her personal device in order to allocate computational resources of FN_i for V_i. Then, CS generates fresh credentials in login and service request phase that are distributed to assigned FN_i and passed to V_i after being received and proceeded by U_i. Finally, V_i and FN_i can conduct mutual authentication and session key generation phase without communicating with CS in a secure and efficient way prior to having service handover.

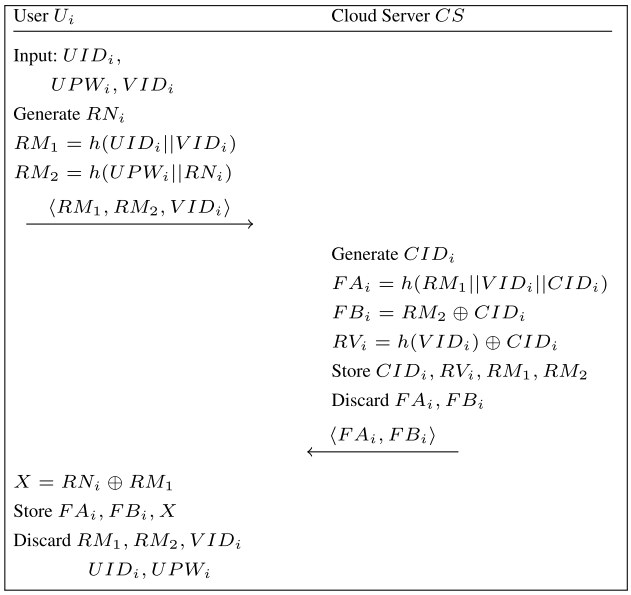

FIGURE 2. User registration phase via a secure channel.

C. REGISTRATION

Prior to conducting registration process as given in Figure 2, user U_i requires to prepare UID_i and UPW_i which should be unique for every user. As for vehicle's identity VID_i, this identity refers to OBU's unique number assigned by manufacturer. In this paper, OBU also functions as the device for computation and communication between vehicle and the environment including RSUs and other vehicles [36], [37]. The registration process, all transmitted messages of which are sent out through secure channel, is described as follows.

- U_i selects identity UID_i, password UPW_i, vehicle identity VID_i, and generates a random number RN_i as nonce. U_i computes registration message $RM_1 = h(UID_i||VID_i)$ and $RM_2 = h(UPW_i||RN_i)$ and send $\langle RM_1, RM_2, VID_i \rangle$ to CS.
- CS generates random number for registered user identity CID_i, computes $FA_i = h(RM_1||VID_i||CID_i)$, $FB_i = RM_2 \oplus CID_i$, and $RV_i = h(VID_i) \oplus CID_i$. Then, CS replies with $\langle FA_i, FB_i \rangle$, stores the parameters CID_i, RV_i, RM_1, and RM_2, and also discards FA_i and FB_i in order to protect against linkage attacks between users and CS.
- Then, U_i computes $X = RN_i \oplus RM_1$, stores FA_i, FB_i, and X, and also discards RM_1 and RM_2 for the same reason.

D. LOGIN AND SERVICE REQUEST

Figure 3 shows data flow and detail execution of login and service request. In the login and service request, U_i needs to input his/her identity, password, and vehicle ID. If the login process is successful, U_i can proceed with specifying several number of service paths to the CS. These requested service's paths are distributed to the specific FNs that cover the paths together with other credentials for on-road authentication with vehicle V_i. However, this paper will not discuss about how CS handle and process those requested paths nor how to limit user's credentials distribution to the specific FNs because discussing those issues can cause more complicated discussion and distract readers from the main purpose of this paper. Instead of that, we will leave discussion about how to establish secure communication among CS and FNs as the future work. The details of login and service request are described as follows.

- U_i inputs UID_i^*, UPW_i^*, and VID_i^* to start login and user authentication process. Prior to verifying inputs, application will create $RM_1' = h(UID_i^*||VID_i^*)$, $RN_i' = X \oplus RM_1'$, $RM_2' = h(UPW_i^*||RN_i')$, and also compute $CID_i' = RM_2' \oplus FB_i$. Then, the application on personal device will check whether FA_i is equal to $h(RM_1'||VID_i^*||CID_i')$.
- Upon authenticating the parameters, the application will generate a random number LN_i and a service path request SA_i, and get timestamp T_1 in order to compute $LM_1 = h(RM_2'||LN_i||SA_i||T_1)$, $LM_2 = SA_i \oplus LN_i$ and $LM_3 = h(RM_1'||CID_i') \oplus LN_i$. Then, U_i sends $\langle LM_1, LM_2, LM_3, CID_i', T_1 \rangle$ to the CS for user verification.
- Prior to proceeding the request, CS has to verify the message by calculating $LN_i' = h(RM_1||CID_i) \oplus LM_3^*$, $SA_i' = LM_2^* \oplus LN_i'$ and checking whether LM_1^* is equal to $h(RM_2||LN_i'||SA_i'||T_1)$. After authenticating, CS will generate service tag $S_{i,t}$ for limited time usage of fog computing service, timestamp T_2, and an initial key K_x prior to conducting later authentication. Then, CS computes $LM_4 = h(CID_i||LN_i'||S_{i,t}||K_x||T_2)$, $LM_5 = h(LN_i'||K_x) \oplus S_{i,t}$, $LM_6 = h(RM_1||RM_2) \oplus K_x$ and replies U_i request with $\langle LM_4, LM_5, LM_6, T_2 \rangle$.
- After that, CS computes hash value of service tag $HS = h(S_{i,t})$, VID_i related hash value $HV = h(RV_i \oplus CID_i||HS)$, and sends $\langle SA_i, HS, HV, K_x \rangle$ to the fog node via secure channel.
- Lastly, U_i will retrieve $K_x' = h(RM_1'||RM_2') \oplus LM_6^*$, $S_{i,t}' = h(LN_i||K_x') \oplus LM_5^*$, and verify whether LM_4^* is equal to $h(CID_i'||LN_i||S_{i,t}'||K_x'||T_2)$. Eventually, U_i computes $VS_{i,t} = S_{i,t}' \oplus VID_i^*$ and $VK_x = h(VID_i^*||h(S_{i,t}')) \oplus K_x'$ which will be entered into the OBU of vehicle.

Note that all of transactions in login and service request phase are done via a secure channel by using TLS/SSL protocol. But still, we provide informal security analysis

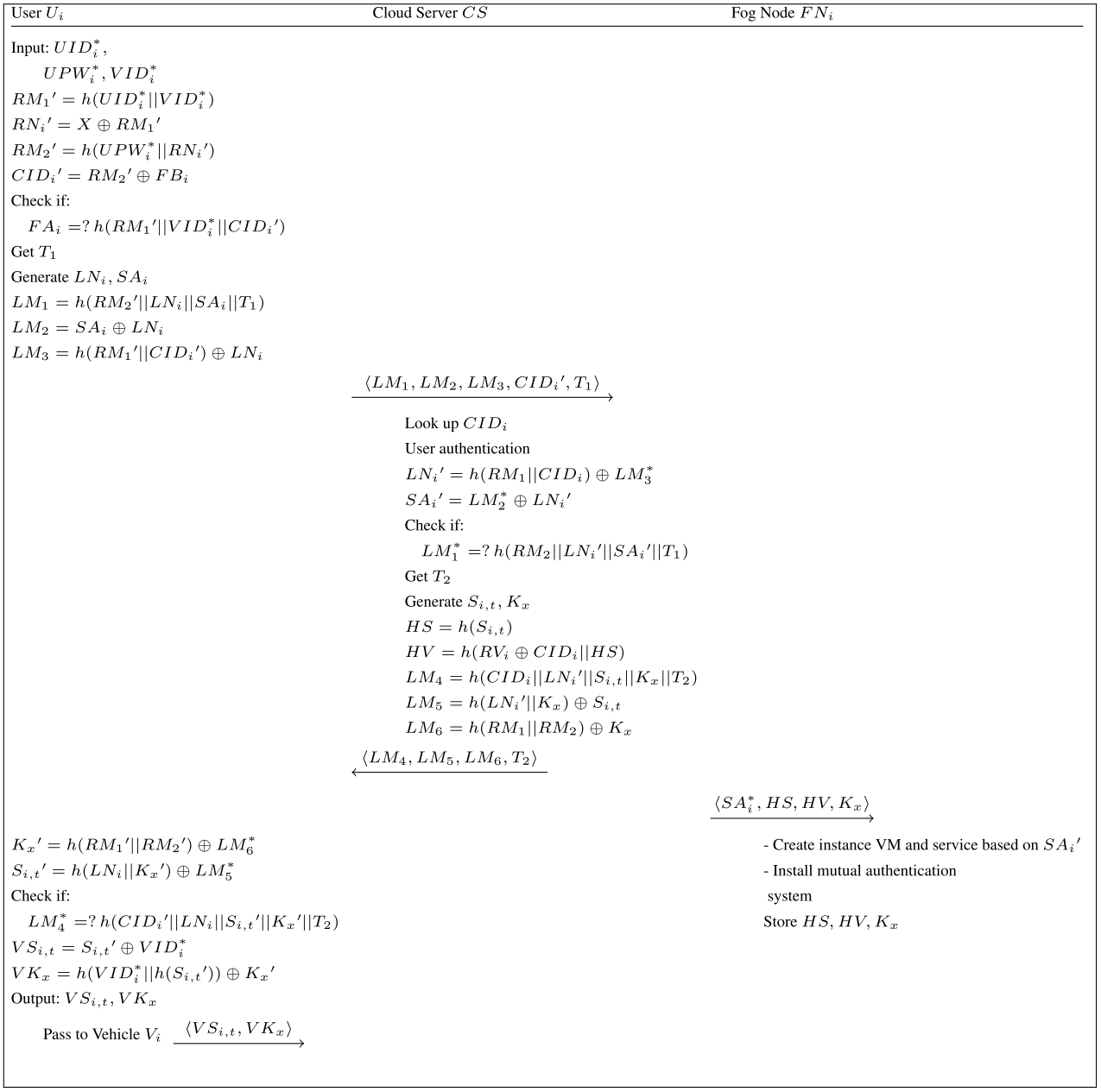

FIGURE 3. Data flow of login and service request phase via a secure channel.

for ensuring that these transactions can be of no benefit to adversaries in exploiting the flaws in case of using insecure network.

For giving more clarity, Figure 4 shows the visualization of login and service request process. At the first step, U_i sends service specification to the *CS*. Then, in the second step, *CS* verifies the service request based on resource availability on the FN_i which is accessed by V_i. At the third step, *CS* sends acknowledge or rejects message together with the later mutual authentication parameters if resource is available. Lastly, user inputs all parameters for mutual authentication and session key generation to the OBU of vehicle.

E. MUTUAL AUTHENTICATION AND SESSION KEY GENERATION

After both V_i and FN_i receive parameters from login process, mutual authentication and session key generation can be performed whenever service handover occurs among all available FN_i in the vehicle path. In this case, V_i will at first initiate the process whenever it senses a different fog node domain area. Intuitively, the sensing process can be understood from the case of vehicular mobility based on IEEE 802.11 protocol network. In our discussion, the connectivity of RSU and fog node server is guaranteed. Because of that, it makes sense that both network and service handover are

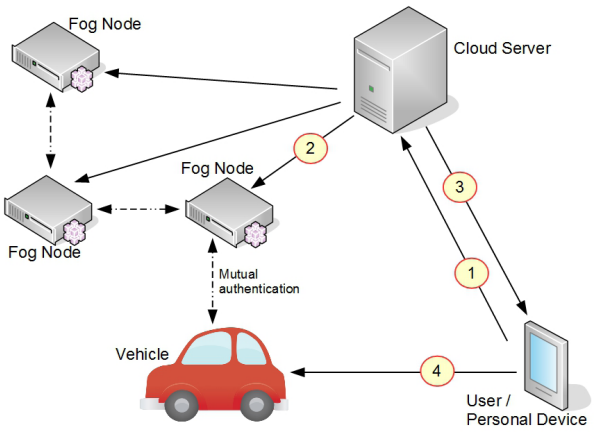

FIGURE 4. Visualization of login and service request process. 1) User login and request fog node service. 2) Cloud server assigns a task to fog node and transfers authentication credentials. 3) Cloud server sends ACK and credentials to user, then 4) User inputs the processed credentials into the Vehicle.

assumed to work smoothly and in a predictive way. Such a method has already been elaborated in [38], [39] and we would not discuss this issue since it is out of scope of this paper.

The process of mutual authentication and session key generation is shown in Figure 5. The detail processes are provided as follows:

- Its process is started with the generation of random number N_v, timestamp t_1, and it is followed by computation of $HS' = h(VS^*_{i,t} \oplus VID_i)$, $HV' = h(VID_i||HS')$, $K'_x = HV' \oplus VK^*_x$, $K_y = h(N_v||K'_x)$, $VA_i = h(HS'||K'_x) \oplus N_v$, and $VB_i = h(K_y||K'_x||t_1)$.
- Upon receiving $\langle VA_i, VB_i, t1\rangle$ from V_i, FN_i verifies V_i by the following computation. At first, FN_i will compute $N'_v = VA^*_i \oplus h(HS||K_x)$, $K'_y = h(N'_v||K_x)$, and check the message received time. Then, it will investigate whether
  1) timestamp t_1 is correct and within limited delay tolerance satisfying $\delta_t > t_2 - t_1$, and
  2) VB^*_i is equal to $h(K'_y||K_x||t_1)$.
- After authenticating V_i, FN_i will generate N_f, compute $K_z = h(N_f||K_x)$, $VC_i = h(HS||HV||K'_y) \oplus N_f$, $SK_f = h(HV||K'_y||K_z)$, and $VD_i = h(SK_f||t_3)$. Then, this message $\langle VC_i, VD_i, t_3\rangle$ is sent to V_i for proving FN_i's role.
- After calculating $N'_f = VC^*_i \oplus h(HS'||HV'||K_y)$ followed by $K'_z = h(N'_f||K'_x)$, $SK_v = h(HV'||K_y||K'_z)$, V_i will check whether these following criteria are accepted.
  1) timestamp t_3 is correct and within limited delay tolerance satisfying $\delta_t > t_4 - t_3$, and
  2) VD^*_i is equal to $h(SK_v||t_3)$.
- After authenticating FN_i, V_i will assign new next initial key VK_x for later authentication that is $VK_x = SK_v \oplus K'_x \oplus HV'$, and send $\langle h(SK_v||t_3||t_5), t_5\rangle$ to FN_i in order to verify the previous process of mutual authentication and session key generation.
- Upon receiving that message from V_i, FN_i will check whether
  1) timestamp t_5 is correct and within limited delay tolerance satisfying $\delta_t > t_6 - t_5$ with current time t_6, and
  2) VE^*_i is equal to $h(SK_f||t_3||t_5)$.
- Eventually, mutual authentication is achieved after both VN_i and FN_i agree upon the same session key SK_v and SK_f respectively.

Vehicle V_i	Fog Node FN_i
Input: $VS^*_{i,t}, VK^*_x$	
$HS' = h(VS^*_{i,t} \oplus VID_i)$	
$HV' = h(VID_i\|\|HS')$	
$K_x' = HV' \oplus VK^*_x$	
Get t_1	
Generate N_v	
$K_y = h(N_v\|\|K_x')$	
$VA_i = h(HS'\|\|K_x') \oplus N_v$	
$VB_i = h(K_y\|\|K_x'\|\|t_1)$	
$\langle VA_i, VB_i, t_1\rangle$ ⇢	
	Get t_2
	$N_v' = h(HS\|\|K_x) \oplus VA^*_i$
	$K_y' = h(N_v'\|\|K_x)$
	Check if:
	$\delta_t >? t_2 - t_1$
	$VB^*_i =? h(K_y'\|\|K_x\|\|t_1)$
	Get t_3
	Generate N_f
	$K_z = h(N_f\|\|K_x)$
	$SK_f = h(HV\|\|K_y'\|\|K_z)$
	$VC_i = h(HS\|\|HV\|\|K_y') \oplus N_f$
	$VD_i = h(SK_f\|\|t_3)$
	$\langle VC_i, VD_i, t_3\rangle$ ⇠
Get t_4	
$N_f' = h(HS'\|\|HV'\|\|K_y) \oplus VC^*_i$	
$K_z' = h(N_f'\|\|K_x')$	
$SK_v = h(HV'\|\|K_y\|\|K_z')$	
Check if:	
$\delta_t >? t_4 - t_3$	
$VD^*_i =? h(SK_v\|\|t_3)$	
Get t_5	
$VK_x = SK_v \oplus K_x' \oplus HV'$	
$VE_i = h(SK_v\|\|t_3\|\|t_5)$	
Store SK_v, VK_x	
$\langle VE_i, t_5\rangle$ ⇢	
	Get t_6
	Check if:
	$\delta_t >? t_6 - t_5$
	$VE^*_i =? h(SK_f\|\|t_3\|\|t_5)$
	$K_x = SK_f \oplus K_x$
	Store SK_f, K_x

FIGURE 5. Mutual authentication and key generation phase.

F. SERVICE TERMINATION

FN_i service can be terminated by V_i or U_i by sending signal of termination in any time. The log of service is compressed

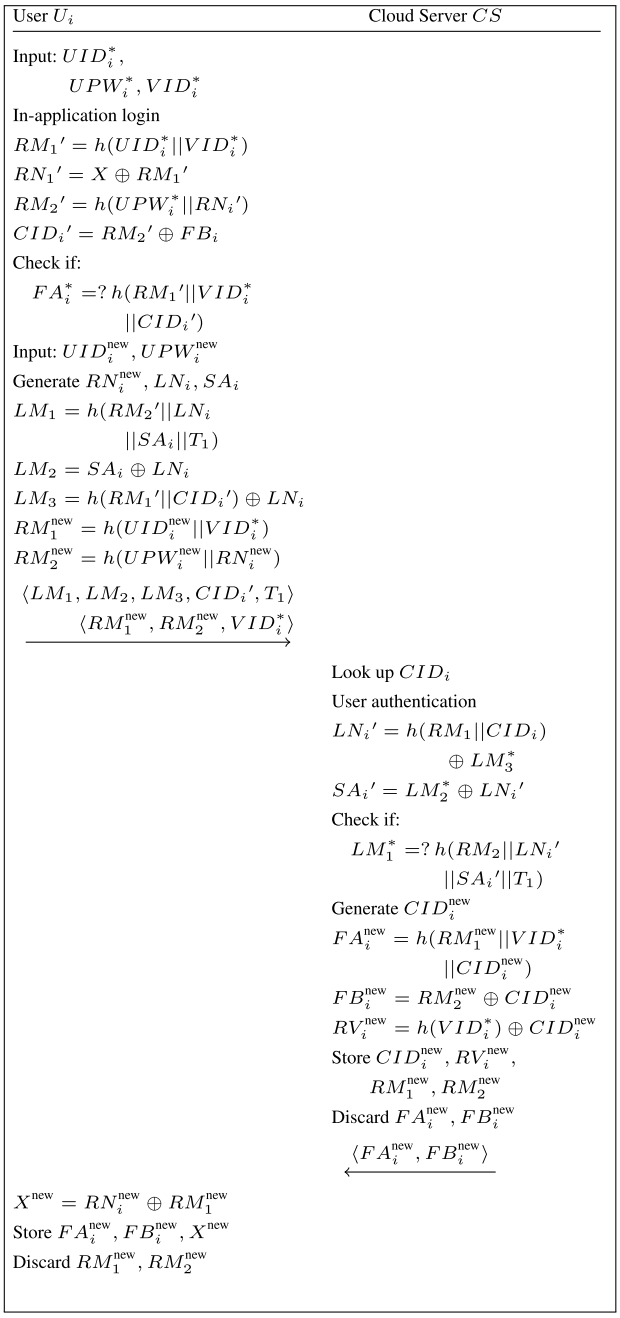

FIGURE 6. Password change phase via a secure channel.

into a zipped file and sent to *CS* for further analysis. U_i is able to download the log file from *CS* and get the summary of FN_i service.

G. USER ID AND PASSWORD CHANGE

A registered user is able to change user ID and password whenever he/she is not using fog computing service. The process of password change is depicted in Figure 6 and the operation is described as follows.

- At first U_i is required to login into the system by entering existing UID_i, UPW_i, and VID_i. Upon successful login, U_i will be inquired about submitting a new user identity UID_i^{new} and a password UPW_i^{new}. Then, a new random number RN_i^{new}, a service request SA_i, and LN_i are picked to calculate LM_1, LM_2, LM_3 and RM_1^{new}, RM_2^{new}.
- Upon receiving $\langle LM_1, LM_2, LM_3, RM_1^{new}, RM_2^{new}, VID_i^*\rangle$, *CS* will authenticate U_i by computing LN_i^* and checking LM_1^* as previously mentioned in login phase.
- If U_i is successfully authenticated, then *CS* will compute FA_i^{new}, FB_i^{new}, RV_i^{new} and also stores RM_1^{new}, RM_2^{new}, CID_i^{new}, and RV_i^{new}.
- After being received by U_i, then FA_i^{new} and FB_i^{new} are saved by U_i together with the latest computed X^{new}.

IV. FORMAL SECURITY ANALYSIS USING REAL OR RANDOM MODEL

The Real Or Random model, is one of familiar approaches for proving the computational indistinguishability [40], [41] of the proposed scheme by inferring the probability ensemble as discussed in the previous works by [14] and [42]. In this ROR-model, we demonstrate that our proposed protocol only reveals a tiny advantage to the adversary in order to obtain secret key. For providing clarity, we define the following models.

Participant: For the entities, vehicle V_i, Fog Node FN_i, and Cloud Server CS_i, we define $\Pi_{V_i}^t$, $\Pi_{FN_i}^u$, and $\Pi_{CS_i}^v$ as the instances t, u, and v of V_i, FN_i, and CS_i respectively.

Partnering: The instances t and u are the partner of each other if they can fulfill the following conditions; 1) both instances are in an accept state, 2) both instances are mutually authenticated, and 3) both instances share an identical session identification (*sid*).

Freshness: A session key is considered to be fresh if $\mathcal{A}$ cannot obtain the key by using reveal query as elaborated in the following part.

Adversary: This model assumes that $\mathcal{A}$ has powerful control over all the communication processes. As a consequence, $\mathcal{A}$ is able to read, modify, and also generate fake messages in order to obtain used session keys. Moreover, $\mathcal{A}$ can perform some action to these following queries as also elaborated in [41] and [43]:

- *Execute*$(\Pi_{V_i}^t, \Pi_{FN_i}^u)$: In this query, $\mathcal{A}$ performs passive attacks by eavesdropping exchanged messages between honest participants of $\Pi_{V_i}^t$ and $\Pi_{FN_i}^u$.
- *Send*$(\Pi_{V_i}^t, m)$: An active attack is performed by executing this query in which $\mathcal{A}$ transmits a message m to the participant $\Pi_{V_i}^t$. The output of this query is the message generated by participant $\Pi_{V_i}^t$ to respond to this query.
- *CorruptVerifier*$(\Pi_{V_i}^t)$: The case of stolen verifier attack of participant $\Pi_{V_i}^t$ is modelled in this query. As a result, $\mathcal{A}$ is able to possess information stored in the verifier (mobile phone) for login/authentication.
- *CorruptOBU*$(\Pi_{V_i}^t)$: It models the case of stolen OBU attack of participant $\Pi_{V_i}^t$. In this query, $\mathcal{A}$ is able to extract both information and computational process from OBU.
- *Test*(Π^t): At first, it initializes $b \leftarrow \{0, 1\}$ by choosing it uniformly at random and outputs the session key if

$b = 1$, random number with the same size of the session key if $b = 0$, and $\perp$ if the session key is not defined yet.

Semantic Security of the Session key: In this formal security model, $\mathcal{A}$ has to distinguish the output of $Test(\Pi^t)$ query, i.e. whether it is the real session key or a random number. In addition, $\mathcal{A}$ is allowed to query to more than one participant, either $\Pi^t_{V_i}$ or $\Pi^u_{FN_i}$. Then, the *Guess* of $\mathcal{A}$ is checked against bit b. If $Guess = b$ then $\mathcal{A}$ wins the game, otherwise $\mathcal{A}$ loses the game. Let W denote an event that $\mathcal{A}$ wins the game. Thus, the advantage of $\mathcal{A}$ in breaking the semantic security of the proposed authenticated key exchange (AKE) protocol $\mathcal{P}$ is given in Equation (1). Our proposed protocol $\mathcal{P}$ is considered to be secured if $Adv^{AKE}_{\mathcal{P}} \leq \psi$ in which ψ is sufficiently small real number bigger than 0.

$$Adv^{AKE}_{\mathcal{P}} = |2Pr[W] - 1|. \tag{1}$$

Random Oracle: Based on [14] and [42], it is assumed that $\mathcal{A}$ gains access to cryptographically secure one way hash function $h(\cdot)$ which is collision-resistant and modeled as a random oracle $\mathcal{H}$.

Theorem 1: If $\mathcal{A}$ be an adversary running in a polynomial time t against our proposed mutual authentication and key exchange protocol $\mathcal{P}$ in the ROR model, then the advantage is

$$Adv^{AKE}_{\mathcal{P}} \leq \frac{q_h^2}{|Hash|} + \frac{2\, q_{login}}{|D_{login}|} + \frac{2 q_{auc}}{|D_{auc}|}$$

where q_h, $|Hash|$, q_{login}, $|D_{login}|$, q_{auc}, and $|D_{auc}|$ are the number of $\mathcal{H}$ queries, the range space of $h(\cdot)$, the number of login queries, the size of D_{login} of dictionary attack in login phase, the number of mutual authentication phase queries in the case of the stolen OBU attack, the size of D_{auc} of dictionary attack in mutual authentication phase between V_i and FN_i respectively.

Proof: The proof of this theorem is delivered by using a sequence of five experiments Exp_i in which $i = 0, 1, 2, 3, 4$ as already demonstrated in the previous works [42] and [14]. We also denote W_i as an event that $\mathcal{A}$ is successful in guessing the random bit b correctly at Exp_i. The detail of the proof is elaborated as follows. □

Experiment Exp_0: It is defined as a real attack on the proposed protocol $\mathcal{P}$. By definition,

$$Adv^{AKE}_{\mathcal{P}} = |2Pr[W_0] - 1|. \tag{2}$$

Experiment Exp_1: This experiment attempts to simulate passive attacks to the session between V_i and FN_i by means of $Execute(\Pi^t_{V_i}, \Pi^u_{FN_i})$ oracle. After eavesdropping exchanged messages, $\mathcal{A}$ will possess variables and continue to guess either the real session key or a random number by querying $Test(\Pi^t)$ oracle. Note that the session key is calculated by using formula $SK_f = h(HV||K'_y||K_z)$ where $K'_y = h(N'_v||K_x)$, $N'_v = h(HS||K_x) \oplus VA^*_i$, $K_z = h(N_f||K_x)$, and also $SK_v = h(HV'||K_y||K'_z)$ where $HS' = h(VS_{i,t} \oplus VID_i)$, $HV' = h(VID_i||HS')$, $K_y = h(N_v||K'_x)$, $K'_x = HV' \oplus VK^*_x$, $K'_z = h(N^*_f||K_x)$, and $N^{*\prime}_f = h(HS'||HV'||K_y) \oplus VC^*_i$. We can easily know that the probability to guess the session key correctly is not increased by eavesdropping VA_i, VB_i, VC_i, VD_i, and VE_i. As a result, we get

$$Pr[W_0] = Pr[W_1]. \tag{3}$$

Experiment Exp_2: This experiment attempts to deceive a target into accepting our modified message by means of $Send(\Pi^t_{V_i}, m)$ or $Send(\Pi^u_{FN_i}, m)$ oracle and the possibility of digest collision in the one-way hash function. In addition, $\mathcal{A}$ is allowed to launch unlimited number of messages to test the collision of the hash function. By applying the birthday paradox, we can obtain

$$|Pr[W_1] - Pr[W_2]| \leq \frac{q_h^2}{2|Hash|}. \tag{4}$$

Experiment Exp_3: Exp_2 is transformed into Exp_3 by simulating $CorruptVerifier(\Pi^t_{V_i})$. In this experiment, $\mathcal{A}$ is able to possess some variables FA_i, FB_i, and X which are kept by personal device and attempts to obtain the session key by login to the system and then input the computed credential into vehicle. However, it is computationally infeasible to get UID_i, UPW_i, and VID_i from FA_i, FB_i, and X due to the protection of $h(\cdot)$. Thus, $\mathcal{A}$ needs to guess UID_i, UPW_i, and VID_i. By considering the number of allowed login error, we can derive

$$|Pr[W_2] - Pr[W_3]| \leq \frac{q_{\text{login}}}{|D_{\text{login}}|}, \tag{5}$$

in which $|D_{\text{login}}|$ is described in the following equation.

$$|D_{\text{login}}| = |D_{\text{UID}_i}||D_{\text{UPW}_i}||D_{\text{VID}_i}|. \tag{6}$$

Experiment Exp_4: By considering the oracle of $CorruptOBU(\Pi^t_{V_i})$, Exp_3 is transformed into Exp_4 which allows $\mathcal{A}$ to possess some variables and computational knowledge. By using $CorruptOBU(\Pi^t_{V_i})$ query, A is able to possess VID_i. In order to deceive FN_i and obtain the session key, $\mathcal{A}$ needs to guess more variables which are $VS_{i,t}$ and VK_x for creating VA_i and VB_i so that fog node can authenticate vehicle from the received VA_i and VB_i. As a consequence,

$$|Pr[W_3] - Pr[W_4]| \leq \frac{q_{\text{auc}}}{|D_{\text{auc}}|}. \tag{7}$$

in which $|D_{\text{auc}}|$ is described in the following equation.

$$|D_{\text{auc}}| = |D_{\text{VS}_{i,t}}||D_{\text{VK}_x}|. \tag{8}$$

In Exp_4 all oracles are simulated and $\mathcal{A}$ is left to guess the bit b eventually whether it is equal to 0 or 1. As a result, it can be derived that

$$Pr[W_4] = \frac{1}{2}. \tag{9}$$

By modifying Equation (2), we can derive

$$\frac{1}{2} Adv^{AKE}_{\mathcal{P}} = |Pr[W_0] - \frac{1}{2}|. \tag{10}$$

By applying the triangular inequality, we can derive

$$\begin{aligned}|Pr[W_1] - Pr[W_4]| &\leq |Pr[W_1] - Pr[W_2]| \\ &\quad + |Pr[W_2] - Pr[W_3]| \\ &\quad + |Pr[W_3] - Pr[W_4]| \\ &\leq \frac{q_h^2}{2|Hash|} + \frac{q_{\text{login}}}{|D_{\text{login}}|} + \frac{q_{\text{auc}}}{|D_{\text{auc}}|}. \quad (11)\end{aligned}$$

By combining Equation (3), (9), and (11), we can derive

$$|Pr[W_0] - \frac{1}{2}| \leq \frac{q_h^2}{2|Hash|} + \frac{q_{\text{login}}}{|D_{\text{login}}|} + \frac{q_{\text{auc}}}{|D_{\text{auc}}|}. \quad (12)$$

Finally, we can complete the proof by combining Equation (10) and (12) as follows.

$$Adv_{\mathcal{P}}^{AKE} \leq \frac{q_h^2}{|Hash|} + \frac{2\, q_{\text{login}}}{|D_{\text{login}}|} + \frac{2q_{\text{auc}}}{|D_{\text{auc}}|}.$$

V. FORMAL SECURITY ANALYSIS USING BAN LOGIC AND SIMULATION

This section is intended to discuss about formal security analysis using BAN Logic [44] as a formal method approach. It is important to mention that BAN Logic is limited authentication method that cannot capture all possible attacks in the protocol [45]. It can only uncover the belief of each agents and examine the correctness of protocol flow with respect to the goal under the assumption that all agents perform honest operation [46]. Then, through this formal analysis, we can prove that our proposed protocol can provide mutual authentication between honest V_i and FN_i.

Prior to elaborating the formal analysis, we need to define the goals for inferring the formal verification process. For understanding the notations and rules in this formal analysis proof, we suggest to referring to works of Srinivas *et al.* [45] and Kumari *et al.* [47]. Then, the **goal** of the formal verification is to prove that our protocol can satisfy the following statements.

1) $V \mid\equiv FN \overset{SK}{\rightleftharpoons} V$
2) $V \mid\equiv FN \mid\equiv V \overset{SK}{\rightleftharpoons} FN$
3) $FN \mid\equiv V \overset{SK}{\rightleftharpoons} FN$
4) $FN \mid\equiv V \mid\equiv FN \overset{SK}{\rightleftharpoons} V$

Note that we use notation SK in formal verification to address session key which is stated in different notation in algorithm (SK_f for fog node and SK_v for vehicle). Then, for clearer and easier logical reasoning, those stated goals are interpreted as **sub-goals** as given below.

SG_1. $V \mid\equiv N_v$
SG_2. $V \mid\equiv FN \mid\equiv N_v$
SG_3. $FN \mid\equiv N_v$
SG_4. $FN \mid\equiv V \mid\equiv N_v$
SG_5. $V \mid\equiv N_f$
SG_6. $V \mid\equiv FN \mid\equiv N_f$
SG_7. $FN \mid\equiv N_f$
SG_8. $FN \mid\equiv V \mid\equiv N_f$
SG_9. $V \mid\equiv SK$
SG_{10}. $V \mid\equiv FN \mid\equiv SK$
SG_{11}. $FN \mid\equiv SK$
SG_{12}. $FN \mid\equiv V \mid\equiv SK$

In that list, sub-goals 1, 5, and 9 are intended to satisfy goal 1. It is clearly understood that N_v is a part of SK_v or SK_f so that sub-goal 1 leads to goal 1. In the case of FN, sub-goals 3, 7, and 11 are to support goal 3. The rest, sub-goals 2, 6, and 10 and also sub-goals 4, 8, 12 are for supporting goal 2 and also goal 4, respectively.

The formal verification procedure that will be elaborated afterwards is conducted under the following assumptions.

A_1. $V \mid\equiv \#(N_v, N_f, SK)$
A_2. $FN \mid\equiv \#(N_v, N_f, SK)$
A_3. $V \mid\equiv FN \overset{\{HS,K_x\}}{\rightleftharpoons} V$
A_4. $FN \mid\equiv V \overset{\{HS,K_x\}}{\rightleftharpoons} FN$
A_5. $FN \mid\equiv V \Rightarrow N_v$
A_6. $FN \mid\equiv V \overset{\{HS,HV,K_y\}}{\rightleftharpoons} FN$
A_7. $V \mid\equiv FN \overset{\{HS,HV,K_y\}}{\rightleftharpoons} FN$
A_8. $V \mid\equiv FN \Rightarrow N_f$
A_9. $V \Rightarrow SK$
A_{10}. $V \mid\equiv FN \Rightarrow SK$
A_{11}. $FN \Rightarrow SK$
A_{12}. $FN \mid\equiv V \Rightarrow SK$

The standard and idealized message of our protocol is also displayed below.

M_1. $V \rightarrow FN$: $VA_i :< \{N_v\} >_{\{HS,K_x\}}, VB_i, t_1$
M_2. $FN \rightarrow V$: $VC_i :< \{N_f\} >_{\{HS,HV,K_y\}}, VD_i, t_3$
M_3. $V \rightarrow FN$: VE_i, t_5

Then, based on BAN Logic rules, assumptions, goals, and sub-goals, formal verification is demonstrated as follows.

Prior to sending M_1, V receives CID_i and $VS_{i,t}$. Then prior to computing VA_i and VB_i, V chooses a random number N_v for later to be used as future SK for both V and FN. By considering A_1, it is implicitly said that

St_1: $V \mid\equiv N_v$. ($\boldsymbol{SG_1}$)
M_1: $V \rightarrow FN$: $VA_i :< N_v >_{\{HS,K_x\}}, VB_i, t_1$
From M_1, we can get
St_2: $FN \lhd VA_i :< N_v >_{\{HS,K_x\}}, VB_i, t_1$.
By combining St_2, A_4, and *message-meaning rule*, we can get
St_3: $FN \mid\equiv V \mid\sim N_v$.
By combining St_3, A_2, and *nonce-verification rule*, we can get
St_4: $FN \mid\equiv V \mid\equiv N_v$. ($\boldsymbol{SG_4}$)
By combining St_4, A_5 and *jurisdiction rule*, we can obtain
St_5: $FN \mid\equiv N_v$. ($\boldsymbol{SG_3}$)
After recognizing N_v, FN creates N_f. By considering A_2, we can implicitly obtain
St_6: $FN \mid\equiv N_f$. ($\boldsymbol{SG_7}$)
After generating nonce N_f, FN will calculate SK : $\{HS, K_y, K_z\}$ as stated in A_{11}. By doing this, it automatically confirms that
St_7: $FN \mid\equiv SK$. ($\boldsymbol{SG_{11}}$)

Then, FN computes $VD_i : \{SK, t_3\}$ and sends the reply message M_2 to V as

M_2: $FN \rightarrow V : VC_i :< N_f >_{\{HS,HV,K_y\}}, VD_i, t_3$

From M_2, we can obtain

St_8: $V \lhd VC_i :< N_f >_{\{HS,HV,K_y\}}, VD_i, t_3$.

By combining St_8, A_7, and *message-meaning rule* for VC_i we can obtain

St_9: $V \mid\equiv FN \mid\sim N_f$.

By combining St_9, A_1, and *nonce-verification rule*, we can get

St_{10}: $V \mid\equiv FN \mid\equiv N_f$. $(\boldsymbol{SG_6})$

By combining St_{10}, A_8, and *jurisdiction rule*, we can obtain

St_{11}: $V \mid\equiv N_f$. $(\boldsymbol{SG_5})$

By believing that V is able to compute SK as mentioned in A_9, and also by combining with St_{11}, we can derive

St_{12}: $V \mid\equiv SK$. $(\boldsymbol{SG_9})$

By believing that FN also calculates SK in the same way as mentioned in A_{10}, and also combining with St_{10} and *message-meaning rule* for VD_i, we can derive

St_{13}: $V \mid\equiv FN \mid\sim SK$.

By combining St_{13}, considering freshness of SK as stated in *freshness-rule*, and also applying *nonce-verification rule*, we can derive

St_{14}: $V \mid\equiv FN \mid\equiv SK$. $(\boldsymbol{SG_{10}})$

Note that SK is derived from $\{HS, K_y, K_z\}$, by means of *elimination rule*, it is automatically the proof of

St_{15}: $V \mid\equiv FN \mid\equiv N_v$. $(\boldsymbol{SG_2})$

After recognizing K_z, SK, and K_y, then V replies the message as follows.

M_3: $V \rightarrow FN : VE_i, t_5$

From M_3, we can obtain

St_{16}: $FN \lhd VE_i, t_5$.

By combining St_{16}, St_7, A_{12}, and *message-meaning rule* we can obtain

St_{17}: $FN \mid\equiv V \mid\sim SK$.

By combining St_{17}, A_2, and *nonce-verification rule*, we can obtain

St_{18}: $FN \mid\equiv V \mid\equiv SK$. $(\boldsymbol{SG_{12}})$

Note that SK is derived from $\{HS, K_y, K_z\}$, by means of *elimination rule*, it is automatically the proof of

St_{19}: $FN \mid\equiv V \mid\equiv N_v$. $(\boldsymbol{SG_8})$

By following those steps, it is clear that our protocol can be proven to achieve all sub-goals as mentioned in Steps St_1, St_4, St_5, St_6, St_7, St_{10}, St_{11}, St_{12}, St_{14}, St_{15}, St_{18}, and St_{19}. Both V and FN believe that they can exchange nonce N_v and N_f and also share the same secret session key SK.

VI. INFORMAL SECURITY ANALYSIS

This section elaborates the security analysis of the proposed protocol informally as done in the previous relevant work [13]–[16]. We consider adversaries are able to eavesdrop the communication channels and modify the messages to obtain private information. Then, we show that the proposed method can protect user U_i, cloud server CS, fog node FN_i, and vehicle V_i from well-known attacks as listed below.

A. IDENTITY/PASSWORD GUESSING AND STOLEN VERIFIER ATTACK

Suppose adversary $\mathcal{A}$ is successful to possess verifier, in this case personal device, and also collect stored information that contains parameters FA_i, FB_i, and X for user authentication. Based on the information, $\mathcal{A}$ attempts to login into the system by arbitrarily guessing any strings for UID_i^*, UPW_i^*, and VID_i^*. As for guessing, $\mathcal{A}$ needs to deduce from the definition of $FA_i = h(RM_1||VID_i||CID_i)$, $FB_i = h(UPW_i||RN_i) \oplus CID_i$. It is obvious that $\mathcal{A}$ cannot infer UID_i, UPW_i, and VID_i from FA_i, FB_i, and X. Moreover, due to one-way hash function property that is used to calculate $RM_1 = h(UID_i||VID_i)$ and $RM_2 = h(UPW||RN_i)$, $\mathcal{A}$ can no longer deduce registered UID_i, UPW_i, and VID_i.

In other case, suppose $\mathcal{A}$ does not possess login verifier and can only eavesdrop the transmitted message from vehicle and fog node $VA_i = h(HS'||K_x') \oplus N_v$, $VB_i = h(K_y||K_x'||t_1)$, $VC_i = h(HS||HV||K_y') \oplus N_f$, $VD_i = h(SK_f||t_3)$, t_1, t_3, t_5, t_6, and $VE_i = h(SK_v||t_3||t_5)$. It is clear that $\mathcal{A}$ cannot obtain any information of UID_i, UPW_i, and VID_i by means of sniffing the transmitted message between vehicle and fog node. As a result, we can conclude that our protocol is proof against identity/password guessing attack in the case of ability to sniff transmitted message and stolen verifier attack.

B. USER IMPERSONATION ATTACK

This attack is performed under the assumption that adversary $\mathcal{A}$ in some way is able to possess FA_i, FB_i, and X by hacking into mobile phone or personal computer of certain users. Without login into the application, $\mathcal{A}$ attempts to impersonate legal user U_i by directly sending message to cloud server CS. It is already proven that it is impossible to attain UID_i, UPW_i, and VID_i from FA_i, FB_i, and X. Then, $\mathcal{A}$ uses another method by simply generating random number LN_i^{adv}, CID_i^{adv}, SA_i^{adv}, RM_1^{adv}, and RM_2 in order to create $LM_1^{adv} = h(RM_2^{adv}||LN_i^{adv}||SA_i^{adv}||T_1)$, $LM_2^{adv} = SA_i^{adv} \oplus LN_i^{adv}$, and also $LM_3^{adv} = h(RM_1^{adv}||CID_i^{adv}) \oplus LN_i^{adv}$. After receiving $\langle LM_1^{adv}, LM_2^{adv}, CID_i^{adv}, T_1\rangle$, CS then decode the message by calculating $LN_i^* = LM_2^{adv} \oplus h(RM_1||CID_i)$. The verification is done by comparing whether LM_1^{adv} is equal to $h(RM_2||LN_i^*||SA_i^*||T_1)$. However, CS realizes that

- CID_i^{adv} is neither on its database, or in another case
- CID_i^{adv} is in some way found on its database but RM_1^{adv} and RM_1 are different which in fact can lead to different value of LN_i^* and LN_i^{adv}.

As a result, CS cannot authenticate impersonation attack of $\mathcal{A}$ and eventually reject the authentication process.

C. MAN-IN-THE-MIDDLE ATTACKS

The adversary $\mathcal{A}$ stands between U_i and CS in the login process, and in some way is successful to obtain LM_1, LM_2, LM_3, CID_i, T_1 and also LM_4, LM_5, LM_6, T_2. Then, $\mathcal{A}$ interrupts data from CS, and changes that data with its own generated credential $S_{i,t}^{adv}$ and K_x^{adv} so that vehicle

in the future will connect to the fraudulent fog node that is prepared by $\mathcal{A}$. Because of difficulty to reveal LN_i, RM_1, RM_2 out of those obtained messages, $\mathcal{A}$ generates its own $LN_i^{adv}, RM_1^{adv}, RM_2^{adv}$ and also computes $LM_4^{adv} = h(CID_i||LN_i^{adv}||S_{i,t}^{adv}||K_x^{adv}||T_2)$, $LM_5^{adv} = h(LN_i^{adv}||K_x^{adv}) \oplus S_{i,t}^{adv}$, and $LM_6 = h(RM_1^{adv}||RM_2^{adv}) \oplus K_x^{adv}$. Then, the message $\langle LM_4^{adv}, LM_5^{adv}, LM_6^{adv}, T_2\rangle$ is sent to the user for being authenticated and passed to the vehicle. However, due to different value of LN_i, RM_1', RM_2' and $LN_i^{adv}, RM_1^{adv}, RM_2^{adv}, U_i$ cannot authenticate that message. As a result, U_i is protected from man-in-the-middle attack and at the same time from fraudulent cloud server attack.

D. VEHICLE IMPERSONATION ATTACK

In this attack, the adversary $\mathcal{A}$ tries to impersonate vehicle by sending $\langle VA_i^{adv}, VB_i^{adv}, t_1\rangle$ and $\langle VE_i^{adv}, t_5\rangle$ instead of $\langle VA_i, VB_i, t_1\rangle$ and $\langle VE_i, t_5\rangle$. However, without knowing the correct value of $VS_{i,t}^*, VK_x^*$ and N_v, this attack will fail at the first place in which FN_i cannot authenticate $\mathcal{A}$. Hence, our proposed authentication system is proof against vehicle impersonation attack.

E. FOG NODE IMPERSONATION ATTACK

This attack is similar to vehicle impersonation attack, the adversary $\mathcal{A}$ attempts to impersonate FN_i by sending $\langle VC_i^{adv}, VD_i^{adv}, t_3\rangle$ instead of $\langle VC_i, VD_i, t_3\rangle$. However, without knowing the appropriate value of HS, HV, K_x, N_f and N_v^*, that fake message can be easily detected by V_i. As a result, $\mathcal{A}$ will fail to impersonate FN_i.

F. STOLEN-OBU/VEHICLE ATTACK

We assume that adversary $\mathcal{A}$ can steal OBU and obtain VID_i in some way. In order to gain access of fog node service, adversary $\mathcal{A}$ inputs $VS_{i,t}^{adv}$ and VK_x^{adv} and also runs the vehicle in the road to communicate with FN_i. Then, that vehicle initiates communication by computing $HS^{adv} = h(VS_{i,t}^{adv} \oplus VID_i)$, $HV^{adv} = h(VID_i||HS^{adv})$, $K_x^{adv} = HV^{adv} \oplus VK_x^{adv}$, generating N_v^{adv}, calculating $K_y^{adv} = h(N_v^{adv}||K_x^{adv})$, $VA_i^{adv} = h(HS^{adv}||K_x^{adv}) \oplus N_v^{adv}$, $VB_i^{adv} = h(K_y^{adv}||K_x^{adv}||t_1)$ and also sending a message $\langle VA_i^{adv}, VB_i^{adv}, t_1\rangle$. After receiving that message, FN_i calculates $N_v^* = h(HS||K_x) \oplus VA_i^{adv}$, $K_y' = h(N_v^*||K_x)$ and checks the authenticity of vehicle using $VB_i^{adv} = ?h(K_y'||K_x||t_1)$. Due to different value of $K_y' \neq K_y^{adv}$, FN_i cannot authenticate adversary's vehicle. As a result, this proposed authentication method is proof against stolen-OBU/vehicle attack.

G. REPLAY ATTACK

In this kind of attack, adversary $\mathcal{A}$ can eavesdrop the transmitted messages $\langle VA_i, VB_i, t_1\rangle$, $\langle VC_i, VID_i, t_3\rangle$, and $\langle VE_i, t_5\rangle$, and also keep those messages for later attacks to the V_i or FN_i. However, due to expiration of timestamp information, this attack can easily be detected, and those messages can be treated as old messages. Thus, this proposed method can prevent and countermeasure replay attack.

H. COMBINATION OF STOLEN VERIFIER, STOLEN OBU, AND SNIFFING ATTACK

In this attack scenario, $\mathcal{A}$ is able to duplicate login verifier and also possess following key parameters FA_i, FB_i, and X. In addition, $\mathcal{A}$ is also assumed to possess OBU and obtain VID_i string. Then, $\mathcal{A}$ performs message eavesdropping in order to get more information and collect $VA_i, VB_i, VC_i, VD_i, VE_i, t_1, t_3$, and t_5. The purpose of this attack is to find secret information for login into the system and exploiting FCS. Unfortunately, all of those information are not beneficial for $\mathcal{A}$ because of this following reasons.

- It is computationally infeasible to cross-correlate between verifier parameters FA_i, FB_i, X, VID_i and messages $VA_i, VB_i, VC_i, VD_i, VE_i, t_1, t_3, t_5$ with the purpose to obtain information of UID_i and UPW_i.
- In order to use fog node service, $\mathcal{A}$ needs more parameters, for example $VS_{i,t}$ and VK_x, which are generated in login and service request phase. Even though, $\mathcal{A}$ can obtain previous $VS_{i,t}$ by accessing mobile phone memory/disk, $\mathcal{A}$ still cannot communicate with FN_i because this service tag $S_{i,t}$ is unique for each requested service.

I. COLD BOOT ATTACK OF VEHICLE'S OBU

This attack scenario occurs in case $\mathcal{A}$ is able to steal OBU and also performs a memory dump of the target vehicle's OBU by doing hard reset. As a consequence, $\mathcal{A}$ may possess several credentials including $VS_{i,t}^*$ and VK_x^*. In case of using static credential, launching this attack gives $\mathcal{A}$ privilege to access private data of victims on FCS. However, due to using dynamic service tag $S_{i,t}$ and initial key K_x, $\mathcal{A}$ is unable to exploit FCS on the road in the next service session after assigned service time is expired. As a result, our proposed authentication scheme is protected against cold boot attack of vehicle's OBU.

J. PRIVACY/ANONYMITY PRESERVING AND INABILITY TO TRACE

From Figure 5, it can be simply understood that all the transactions between V_i and FN_i eliminate the usage of user identity UID_i and UPW_i. Instead of user identity, this method employs temporary service tag $S_{i,t}$ and temporary initial key K_x to conduct mutual authentication. This $S_{i,t}$ is changed every time new fog computing service is requested and also K_x is recalculated in the process of mutual authentication. In addition, this process is protected by one-way hash function combined with fresh nonce (N_v and N_f) to conceal $S_{i,t}$. As a consequence, this method guarantees anonymity service for the users even though they run their vehicle on the same path every day. Furthermore, by only sniffing the communication channel, it is infeasible to infer user/vehicle identity.

K. EPHEMERAL SECRET LEAKAGE ATTACK

Considering CK adversary model, we need to mention that there are two types of credentials that are used in this scheme for creating session keys (SK_v, SK_f) which are short term/ephemeral credentials (N_v, N_f, K_x) and also long term

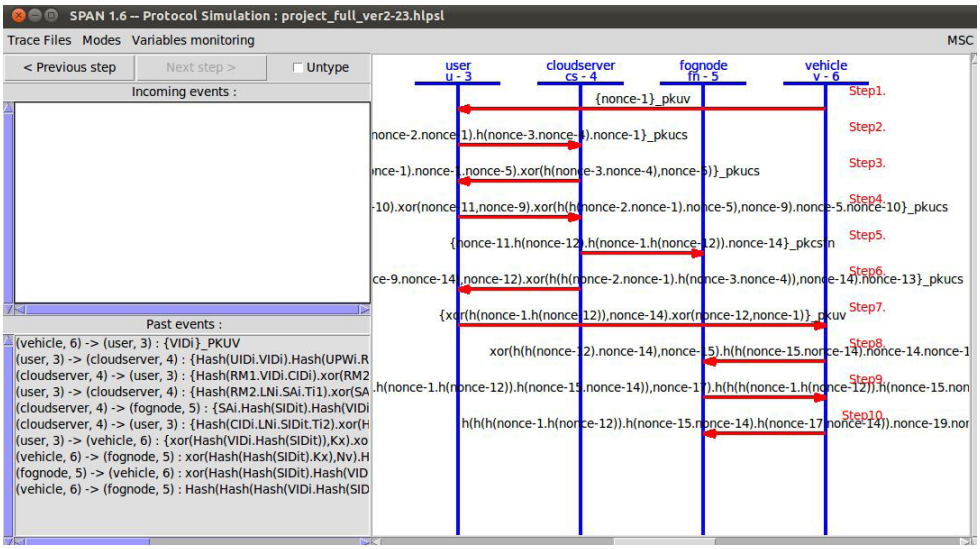

FIGURE 7. Simulation result of executability the proposed method.

credentials ($S_{i,t}$, HS, HV). In the proposed authentication scheme, session key shared between V_i and FN_i is calculated as $SK_f = h(HV||K'_y||K_z)$, where $K'_y = h(N^*_v||K_x)$, $N^*_v = h(HS||K_x) \oplus VA^*_i$, $K_z = h(N_f||K_x)$, and also $SK_v = h(HV'||K_y||K'_z)$, where $HS' = h(VS_{i,t} \oplus VID_i)$, $HV' = h(VID_i||HS')$, $K_y = h(N_v||K'_x)$, $K'_x = HV' \oplus VK^*_x$, $K'_z = h(N^*_f||K_x)$, and $N^{*\prime}_f = h(HS'||HV'||K_y) \oplus VC^*_i$.

In this attack model, we create two scenarios which reflect CK adversary model. Those scenarios are composed of 1) the exposure of short-term credentials and also 2) the exposure of long term credential. However, even though one of scenarios are used, still they cannot give any benefit to $\mathcal{A}$ due to these following reasons.

- Possessing short term credentials (N_v, N_f, K_x) are not helpful for $\mathcal{A}$ in order to reveal long term credentials. Thus, $\mathcal{A}$ cannot compute or guess session key correctly.
- In other case, it is also computationally infeasible by only possessing long term credentials ($S_{i,t}$, HS) for $\mathcal{A}$ to derive session key correctly.

Moreover, revealing session key in particular session does not guarantee $\mathcal{A}$ to be able revealing session key in the previous or future sessions. In addition, the existence of limited time service tags ($S_{i,t}$, HS) gives more complexity to guess correct session key under different sessions and service tags. As a result, we conclude that our proposed authentication scheme is proof against ephemeral secret leakage attack.

L. FORWARD SECRECY PRESERVING

This proposed authentication scheme can preserve forward secrecy by using fresh timestamps t_1, t_3, t_5, random numbers N_v, N_f, initial key K_x, and also service tag $S_{i,t}$ for every service request. Thus, obtaining those mentioned value will not give any benefit to $\mathcal{A}$ in the following sessions as described in the previous sub-sections.

VII. SIMULATION USING AVISPA

After validating the authentication process, we complete the analysis by showing the simulation result using SPAN software which is based on AVISPA [6], [48]. In this paper, we use SPAN software which is installed under 32bit Ubuntu 10.10 VM image with 11 GB memory and run under software Oracle VM Virtual Box [49]. The SPAN software is used for simulating feasibility of HLPSL code execution by means of animation, running OFMC back-end, CL-Atse back-end, SATMC back-end, and also TA4SP back-end. However, due to using XOR operator in our proposed scheme which are not supported by both SATMC and TA4SP back-ends, the results are always ''Inconclusive'' for both SATMC and TA4SP back-ends. Hence, we only provide the security analysis results from OFMC and CL-Atse back-ends as given below.

The idea behind feasibility of HLPSL code execution is to make sure whether the protocol specification works well in terms of syntax and grammatical error. Syntax error checking actually is done by AVISPA, but grammatical error which shows message flow is actually difficult to be performed by AVISPA alone. In this case, SPAN comes to add more feature on AVISPA by showing feasibility of HLPSL code execution by means of message flow animation. Moreover, there is condition caused by grammatical error that makes the HLPSL code compilable but it cannot be executed. As a result, it makes AVISPA back-ends find no attack in the simulated protocol. Thus, it is important to conduct this feasibility test as shown in Figure 7.

The OFMC (on-the-fly model checker) back-end is one of the tools in AVISPA project for verification and demonstrating attack in the tested protocol. It takes initial states, transition relation and goals as the input of the simulation checker. It employs Dolev-Yao intruder model by introducing an intruder in the middle of communication. The intruder is able to keep the message and send it again later to the agents

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/project_full_ver2-23.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.59s
  visitedNodes: 156 nodes
  depth: 18 plies
```

(a) Simulation result using OFMC back-end

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/project_full_ver2-23.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 3476 states
  Reachable  : 687 states
  Translation: 0.31 seconds
  Computation: 0.04 seconds
```

(b) Simulation result using CL-Atse back-end

FIGURE 8. Simulation results in the case of replay attack and man in the middle attack.

in simulation. In the end of analysis, it will show attack trace if it is found by OFMC back-end, or SAFE state as shown in Figure 8(a).

Unlike OFMC, CL-Atse back-end performs faster computation in finding potential attack due to its computational simplification in handling exclusive-or operation [50]. In spite of its faster performance and many optimizations, CL-Atse is considered as the same tools that use same input from HLPSL2IF converter to generate a number of possible attacks in bounded number of sessions. In our simulation, it clearly shows that CL-Atse back-end performs faster computation as compared to OFMC back-end. Moreover, it also shows same result as given by OFMC back-end analysis. For giving better clarity, the simulation result can be checked on Figure 8(b).

VIII. PERFORMANCE EVALUATION

In this section, we discuss performance evaluation between our method and several previous works in terms of computational cost and communication cost. Among several previous works on mutual authentication and session key generation, we select four works Dua *et al.* [13], Feng *et al.* [16], Mohit *et al.* [15], and Wazid *et al.* [14] which are considered as similar and comparable approach as compared to our method. Thus, to the best of our knowledge it is enough to compare our work with these recent works.

A. COMPUTATIONAL COST

It is important to discuss computational cost in any authentication method analysis because computation plays a vital role in determining the speed and real-timeness of the protocol. In this case, our novel method is expected to deliver lower computational cost in order to compete with previous methods.

TABLE 3. Comparison of Computational Cost.

Method	Num. of operation	Comp. cost (*ms*)
Our	$18T_H$	9.00
Dua et al. [13]	$12T_H + 8T_M$	510.00
Mohit et al. [15]	$20T_H$	10.00
Wazid et al. [14]	$24T_H$	12.00
Feng et al. [16]	$4T_H + 5T_M$	317.00

In Table 3, computational cost is described as the total number of operations that significantly affect the speed of mutual authentication process and session key generation process in all involved parties. It covers the process of exclusive-or operation, one-way hash function, and encryption/decryption. However, based on [15], XOR function is not counted and negligible in computation analysis because it is not valuable as compared to one way hash function (T_H =0.0005 *seconds*), symmetric key cryptography (T_S = 0.0087 *seconds*) and elliptic curve scalar point multiplication (T_M = 0.0630 *seconds*).

The result shows that our method can exceed the computational cost of previous results by more than 1.1 and even 56.67 times faster with respect to previous work. It is because our method uses simple and effective computation as compared with others. Other methods that rely only on one-way hash function can reach 10 *ms* in [15] and 12 *ms* in [14]. The other two methods show slowest computation time, which are 510 *ms* in [13] and 317 *ms* in [16], due to using elliptic curve cryptography approach that employs high cost scalar point multiplication.

B. COMMUNICATION COST

In this paper, communication cost is calculated by accumulating the total numbers of messages and size (in *bits*) used to conduct mutual authentication and session key generation. Based on [15], we define one-way hash function as SHA-1 function which is 160 *bits* in size. The ECC-point multiplication is defined as 512 bits. Lastly, the timestamp is defined as long integer type data which is 32 *bits* in size.

TABLE 4. Comparison of Communication Cost.

Method	Num. of message	Comm. cost (*bits*)
Our	3	896
Dua et al. [13]	3	2144
Mohit et al. [15]	4	1280
Wazid et al. [14]	3	896
Feng et al. [16]	4	1416

The communication cost is given in Table 4. Our proposed scheme results lower communication cost with respect to other related work except for Wazid *et al.* [14] result. Our work is considered to be lightweight as same as Wazid *et al.* [14] that only uses 3 messages and 896 *bits* in size. With respect to other three related work [13], [15], [16], our work can reduce the total message size by 30% to 58.21%. As for Mohit *et al.* [15], their result (1280 *bits*) shows higher communication cost in comparison with our result (896 *bits*) due to involving higher number of entities in mutual authentication process. Due to using elliptic curve cryptography technique, Feng *et al.* [16] and Dua *et al.* [13] need to transfer bigger data size so that it affects communication cost. As for Feng *et al.* [16], their protocol uses 1416 *bits* computational cost and 4 messages. Lastly, the work of Dua *et al.* [13] shows the highest communication cost in which it uses 3 messages but 2144 *bits* computational cost.

C. COMPARISON OF SECURITY FEATURES

The following list will explain the discussion on the comparable research features.

- **Feng *et al.* [16]** propose authentication scheme to complete the previous work in [51] which handle event-based reputation system for traffic safety application. Their work is claimed to be secure and effective to prevent and detect multi sources Sybil attack on vehicular network. In addition, they mention that there are three reasons to reject warning messages which are sent by Sybil attacker. Those are 1) reusing expired pseudonym and certificate, 2) combining stolen pseudonym with arbitrary guessed session key, and 3) forging pseudonym and session key. However, they are careless in designing the authentication method to protect against collusion attack. In particular, the potency of collusion attack occurs in the process of local certificate validation which is requested by receiver of fake warning message to the RSU. To successfully deceive the victim (V_{vic}), there should be at least 1 attacker that stands between victim and RSU. Then, that attacker sends HM_{vic} that is equal to HM_1 after receiving validation message request from victim as shown in Figure 9.

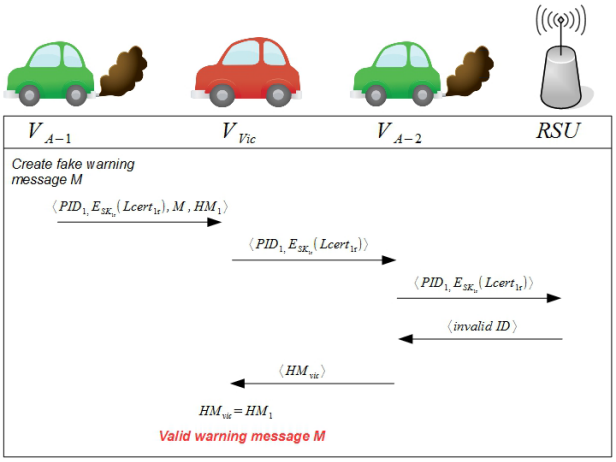

FIGURE 9. Collusion attack on Feng et al. [16].

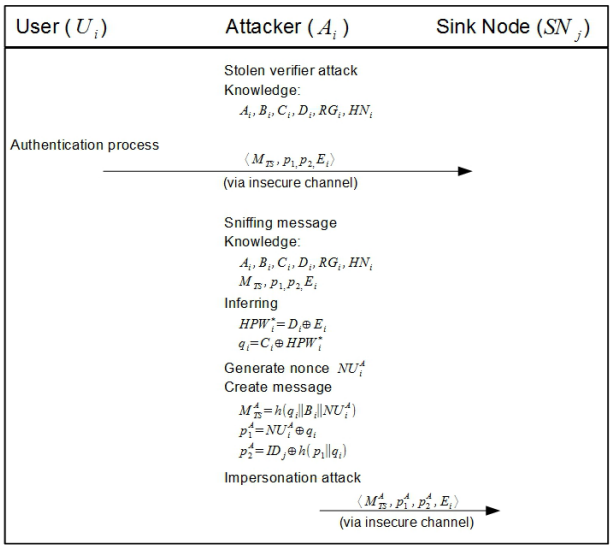

FIGURE 10. Combination of stolen verifier attack, sniffing message and impersonation attack on Mohit *et al.* [15].

- **Mohit *et al.* [15]** propose authentication protocol for wireless sensor network-based smart vehicular system. Their work is claimed to be lightweight in terms of communication and computational cost as compared with previous works. In addition, by using informal security analysis, they can demonstrate that their method is secure against impersonation attack, stolen smart card attack, off-line identity guessing attack, and also providing privacy protection by concealing vehicle trajectory. However, they are failed to protect vehicles against combination of stolen verifier attack, message sniffing, and impersonation attack. As a consequence, attacker can possess hashed password (HPW_i) and eventually able to impersonate the role of user (U_i) as shown in Figure 10.

TABLE 5. Comparison of Protocol Properties, Attack Resistance, and Validation.

Properties	Our	[13]	[16]	[15]	[14]
$M1$	◯	⊗	⊗	⊗	⊗
$M2$	◯	◯	⊗	⊗	⊗
$A1$	✓	✓	✓	✓	✓
$A2$	✓	✓	✓	✓	✓
$A3$	✓	✓	N/A	N/A	✓
$A4$	✓	✓	N/A	✓	✓
$A5$	✓	✓	✓	✓	✓
$A6$	✓	✓	✓	✓	✓
$A7$	✓	✓	×	✓	✓
$A8$	✓	✓	×	×	✓
$A9$	✓	×	N/A	N/A	×
$A10$	✓	✓	×	×	×
$F1$	✓	✓	⊗	⊗	✓
$F2$	✓	✓	⊗	⊗	⊗
$S1$	✓	✓	⊗	⊗	⊗

M1:Online Login; M2:Dynamic credential creation; A1:Identity and password guessing attack; A2:Impersonation attack; A3:Stolen-OBU attack; A4:stolen-verifier attack; A5:Inability to trace; A6:Anonymity; A7:Replay attack; A8:Combination of stolen verifier, stolen OBU, and impersonation attack; A9:Cold boot attack of vehicle's OBU; A10: Ephemeral Secret Leakage (CK adversary model); F1:Formal security analysis using RoR model; F2:Formal security analysis using BAN Logic; S1:Software-based protocol validation; ✓:Resistant against attack; ×:Not resistant against attack; ◯: used in mentioned paper; ⊗: Not used in mentioned paper; N/A: Not applicable

Moreover, lack of formal verification and software-based validation makes their work less comprehensive in verifying the performance of authentication scheme and session key creation.

- Another lightweight authentication scheme is also proposed by **Wazid *et al.* [14]**. By means of informal security analysis and real-or-random (ROR) model, the authors claim that their authentication scheme is secure against various known attacks. However, their work is not appropriate for infrastructure-based FCS because only limited number of vehicles that can access RSU through authentication process. Moreover, their work utilizes static credentials which makes their protocol vulnerable against cold boot attack unless they change credentials through password update phase or using trusted platform module (TPM). In addition, they do not consider protection against ESL attack in the CK-adversary model.
- This recent authentication approach by **Dua *et al.* [13]** can satisfy several critical properties and even demonstrate secure system against ESL attack in the CK-adversary model. However, they do not provide some important security analysis, e.g. stolen vehicle/OBU, stolen verifier, which may occur in real life. Moreover, their authentication approach addresses different network architecture and different purpose as discussed previously.

Lastly, the comparison of protocol properties, attack resistance, and validation method is shown in Table 5. In that table, the discussed items are listed and mentioned along with other parameters as defined with $M1 - M2$, $A1 - A10$, $F1 - F2$, and $S1$. Those notations mean properties of authentication method, attack resistance, formal verification, and software validation respectively. Note that the attack resistance recapitulation in Table 5 is based on informal security analysis in this paper and also taken from the discussion of the selected papers in Table 5.

IX. CONCLUSIONS AND FUTURE WORK

This paper presented a mutual authentication scheme for secure fog computing service handover in the vehicular network environment. The proposed scheme is lightweight and efficient in securing private information due to employing one-way hash function and exclusive-or operation. Formal security analyses by means of the Real-Or-Random model and the BAN Logic show that our proposed method can satisfy and guarantee the security of mutual authentication process. As a part of formal analyses, software-based validation by using SPAN software based on AVISPA also confirms that our method is secured against replay and man in the middle attacks. Furthermore, by using informal security analyses, no weakness of the proposed method was found against various known attacks, including off-line guessing attack, replay attack, impersonation attack, stolen verifier attack, and combination attack. Performance evaluation in terms of computational and communication cost shows that our method is competitive as compared with the previous methods with the best performance. Eventually, our proposed scheme outperforms other related work in terms of security guarantee, computational cost, and communication cost.

For the future work, we will extend this work by discussing authentication scheme between fog node and cloud server. We plan to elaborate in details the mechanism of limited service handover scheme to effectively follow vehicle trajectory and protect users from vehicle theft incidents at the same time. We also consider to equip our work with fault tolerant schemes with respect to heavy data traffic, natural disaster, and any other scenarios.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for providing constructive feedback and insightful suggestions.

REFERENCES

[1] OpenFog Consortium. (2017). *OpenFog Reference Architecture for Fog Computing*. [Online]. Available: https://www.openfogconsortium.org/wp-content/uploads/OpenFog_Reference_Architecture_2_09_17-FINAL-1.pdf

[2] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, Oct. 2009.

[3] P. Hu, S. Dhelim, H. Ning, and T. Qiu, "Survey on fog computing: Architecture, key technologies, applications and open issues," *J. Netw. Comput. Appl.*, vol. 98, pp. 27–42, Nov. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804517302953

[4] E. Ndashimye, S. K. Ray, N. I. Sarkar, and J. A. Gutiérrez, "Vehicle-to-infrastructure communication over multi-tier heterogeneous networks: A survey," *Comput. Netw.*, vol. 112, pp. 144–166, Jan. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128616303826

[5] F. Yang, J. Li, T. Lei, and S. Wang, "Architecture and key technologies for Internet of vehicles: A survey," *J. Commun. Inf. Netw.*, vol. 2, no. 2, pp. 1–17, Jun. 2017. doi: 10.1007/s41650-017-0018-6.

[6] Y. Boichut, T. Genet, Y. Glouche, and O. Heen, "Using animation to improve formal specifications of security protocols," in *Proc. SAR-SSI*, 2007, pp. 169–182.

[7] AVISPA. *Automated Validation of Internet Security Protocols and Applications*. Accessed: Jul. 2019. [Online]. Available: http://www.avispa-project.org/

[8] Y. Chevalier, L. Compagna, J. Cuellar, P. Hankes Drielsma, J. Mantovani, S. Moedersheim, and L. Vigneron, "A high level protocol specification language for industrial security-sensitive protocols," in *Proc. Workshop Specification Automated Process. Secur. Requirements (SAPS)*, Linz, Austria, 2004, p. 13. [Online]. Available: https://hal.inria.fr/inria-00099882

[9] A. Gotsman, F. Massacci, and M. Pistore, "Towards an independent semantics and verification technology for the HLPSL specification language," *Electron. Notes Theor. Comput. Sci.*, vol. 135, no. 1, pp. 59–77, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1571066105050528

[10] S. Kumari and H. Om, "Authentication protocol for wireless sensor networks applications like safety monitoring in coal mines," *Comput. Netw.*, vol. 104, pp. 137–154, Jul. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128616301384

[11] Y. Yao, X. Chang, J. Misic, and V. Mišić, "Reliable and secure vehicular fog service provision," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 734–743, Feb. 2019.

[12] J. Li, X. Shen, L. Chen, D. P. Van, J. Ou, L. Wosinska, and J. Chen, "Service migration in fog computing enabled cellular networks to support real-time vehicular communications," *IEEE Access*, vol. 7, pp. 13704–13714, 2019.

[13] A. Dua, N. Kumar, A. K. Das, and W. Susilo, "Secure message communication protocol among vehicles in smart city," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4359–4373, May 2018.

[14] M. Wazid, A. K. Das, N. Kumar, V. Odelu, A. G. Reddy, K. Park, and Y. Park, "Design of lightweight authentication and key agreement protocol for vehicular ad hoc networks," *IEEE Access*, vol. 5, pp. 14966–14980, 2017.

[15] P. Mohit, R. Amin, and G. P. Biswas, "Design of authentication protocol for wireless sensor network-based smart vehicular system," *Veh. Commun.*, vol. 9, pp. 64–71, Jul. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2214209616301127

[16] X. Feng, C.-Y. Li, D.-X. Chen, and J. Tang, "A method for defensing against multi-source sybil attacks in VANET," *Peer-Peer Netw. Appl.*, vol. 10, no. 2, pp. 305–314, 2017. doi: 10.1007/s12083-016-0431-x.

[17] A. Machen, S. Wang, K. K. Leung, B. J. Ko, and T. Salonidis, "Live service migration in mobile edge clouds," *IEEE Wireless Commun.*, vol. 25, no. 1, pp. 140–147, Feb. 2018.

[18] K. Ha, P. Pillai, W. Richter, Y. Abe, and M. Satyanarayanan, "Just-in-time provisioning for cyber foraging," in *Proc. 11th Annu. Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, 2013, pp. 153–166. [Online]. Available: http://doi.acm.org/10.1145/2462456.2464451

[19] *Intelligent Transport Systems (ITS); Security; ITS Communications Security Architecture and Security Management*, Eur. Telecommun. Standards Inst., Sophia Antipolis, France, 2016. [Online]. Available: http://www.etsi.org/deliver/etsi_ts/102900_102999/102940/01.02.01_60/ts_102940v010201p.pdf

[20] R. G. Engoulou, M. Bellaïche, S. Pierre, and A. Quintero, "VANET security surveys," *Comput. Commun.*, vol. 44, pp. 1–13, May 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366414000863

[21] S. S. Manvi and S. Tangade, "A survey on authentication schemes in VANETs for secured communication," *Veh. Commun.*, vol. 9, pp. 19–30, Jul. 2017, [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2214209616300018

[22] F. Kargl, Z. Ma, and E. Schoch, "Security engineering for VANETs," in *Proc. 4th Workshop Embedded Secur. Cars (escar)*, 2006, pp. 1–10.

[23] X. Li, Y. Xiong, J. Ma, and W. Wang, "An efficient and security dynamic identity based authentication protocol for multi-server architecture using smart cards," *J. Netw. Comput. Appl.*, vol. 35, no. 2, pp. 763–769, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804511002244

[24] K. Xue, C. Ma, P. Hong, and R. Ding, "A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 36, no. 1, pp. 316–323, Jan. 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804512001403

[25] S. Liu, S. Hu, J. Weng, S. Zhu, and Z. Chen, "A novel asymmetric three-party based authentication scheme in wearable devices environment," *J. Netw. Comput. Appl.*, vol. 60, pp. 144–154, Jan. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804515002143

[26] F. Wu, L. Xu, S. Kumari, X. Li, J. Shen, K.-K. R. Choo, M. Wazid, and A. K. Das, "An efficient authentication and key agreement scheme for multi-gateway wireless sensor networks in IoT deployment," *J. Netw. Comput. Appl.*, vol. 89, pp. 72–85, Jul. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804516303150

[27] R. Amin and G. Biswas, "A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks," *Ad Hoc Netw.*, vol. 36, pp. 58–80, Jan. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870515001274

[28] S. Kumari, X. Li, F. Wu, A. K. Das, K.-K. R. Choo, and J. Shen, "Design of a provably secure biometrics-based multi-cloud-server authentication scheme," *Future Gener. Comput. Syst.*, vol. 68, pp. 320–330, Mar. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X16303776

[29] S. Zhang, J. Chen, F. Lyu, N. Cheng, W. Shi, and X. Shen, "Vehicular communication networks in the automated driving era," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 26–32, Sep. 2018.

[30] J. Ni, K. Zhang, X. Lin, and X. S. Shen, "Securing fog computing for Internet of Things applications: Challenges and solutions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 601–628, 1st Quart., 2018.

[31] S. Khan, S. Parkinson, and Y. Qin, "Fog computing security: A review of current applications and security solutions," *J. Cloud Comput.*, vol. 6, no. 1, p. 19, Aug. 2017. doi: 10.1186/s13677-017-0090-3.

[32] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges," *Future Gener. Comput. Syst.*, vol. 78, pp. 680–698, Jan. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X16305635

[33] Openstack. *Open Source Software for Creating Private and Public Clouds*. Accessed: Jul. 2019. [Online]. Available: https://www.openstack.org/

[34] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen, "Vehicular fog computing: A viewpoint of vehicles as the infrastructures," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 3860–3873, Jun. 2016.

[35] M. Cheminod, L. Durante, and A. Valenzano, "Review of security issues in industrial networks," *IEEE Trans. Ind. Informat.*, vol. 9, no. 1, pp. 277–293, Feb. 2013.

[36] X. Li, Y. Feng, F. Wang, and Q. Qian, "When smart phone meets vehicle: A new on-board unit scheme for VANETs," in *Proc. IEEE Int. Conf. Pervasive Intell. Comput.*, Liverpool, U.K., Oct. 2015, pp. 1095–1100.

[37] F. Visintainer, L. Altomare, A. Toffetti, A. Kovacs, and A. Amditis, "Towards manoeuver negotiation: Autonet2030 project from a car maker perspective," *Transp. Res. Procedia*, vol. 14, pp. 2237–2244, Jan. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2352146516302459

[38] M. Mouton, G. Castignani, R. Frank, and T. Engel, "Enabling vehicular mobility in city-wide IEEE 802.11 networks through predictive handovers," *Veh. Commun.*, vol. 2, no. 2, pp. 59–69, Apr. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2214209615000108

[39] D. Li, X. Li, and J. Wan, "A cloud-assisted handover optimization strategy for mobile nodes in industrial wireless networks," *Comput. Netw.*, vol. 128, pp. 133–141, Dec. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128617302359

[40] O. Goldreich, *Foundations of Cryptography*, vol. 1. New York, NY, USA: Cambridge Univ. Press, 2006.

[41] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Public Key Cryptography*, S. Vaudenay, Ed. Berlin, Germany: Springer, 2005, pp. 65–84.
[42] A. K. Das, S. Kumari, V. Odelu, X. Li, F. Wu, and X. Huang, "Provably secure user authentication and key agreement scheme for wireless sensor networks," *Secur. Commun. Netw.*, vol. 9, no. 16, pp. 3670–3687, Nov. 2016. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/sec.1573
[43] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks," in *Advances in Cryptology*, B. Preneel, Ed. Berlin, Germany: Springer, 2000, pp. 139–155.
[44] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Trans. Comput. Syst.*, vol. 8, no. 1, pp. 18–36, 1990. [Online]. Available: http://doi.acm.org/10.1145/77648.77649
[45] J. Srinivas, S. Mukhopadhyay, and D. Mishra, "Secure and efficient user authentication scheme for multi-gateway wireless sensor networks," *Ad Hoc Netw.*, vol. 54, pp. 147–169, Jan. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870516302980
[46] P. Syverson and I. Cervesato, "The logic of authentication protocols," in *Foundations of Security Analysis and Design*, R. Focardi and R. Gorrieri, Eds. Berlin, Germany: Springer, 2001, pp. 63–137.
[47] S. Kumari, X. Li, F. Wu, A. K. Das, H. Arshad, and M. K. Khan, "A user friendly mutual authentication and key agreement scheme for wireless sensor networks using chaotic maps," *Future Gener. Comput. Syst.*, vol. 63, pp. 56–75, Oct. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X16300930
[48] O. Heen, T. Genet, S. Geller, and N. Prigent, "An industrial and academic joint experiment on automated verification of a security protocol," in *Proc. Mobile Wireless Netw. Secur. (MWNS)*. Singapore: World Scientific, 2008, pp. 39–53.
[49] T. Genet, "A short SPAN+AVISPA tutorial," IRISA, Rennes, France, Res. Rep. hal-01213074v5, Oct. 2015. [Online]. Available: https://hal.inria.fr/hal-01213074v5/document
[50] P. Lafourcade, V. Terrade, and S. Vigier, "Comparison of cryptographic verification tools dealing with algebraic properties," in *Formal Aspects Security Trust*, P. Degano and J. D. Guttman, Eds. Berlin, Germany: Springer, 2010, pp. 173–185.
[51] N.-W. Lo and H.-C. Tsai, "A reputation system for traffic safety event on vehicular ad hoc networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2009, no. 1, Dec. 2009, Art. no. 125348. doi: 10.1155/2009/125348.

FAVIAN DEWANTA received the B.Eng. degree in telecommunication engineering from Telkom University, Bandung, Indonesia, in 2009, and the M.Eng. degree in IT convergence engineering from the Kumoh National Institute of Technology, Gumi, South Korea, in 2013. He is currently pursuing Ph.D. degree in information security with the Division of Electrical Engineering and Computer Science, Graduate School of Natural Science and Technology, Kanazawa University, Japan. Since 2015, he has been a Lecturer with the School of Electrical Engineering, Telkom University. His research interests include authentication, trust establishment, information security, edge/fog computing, the IoT, vehicular networks, wireless sensor networks, industrial networks, and also real-time systems.

MASAHIRO MAMBO received the B.Eng. degree from Kanazawa University, Japan, in 1988, and the M.S.Eng. and Dr.Eng. degrees in electronic engineering from the Tokyo Institute of Technology, Japan, in 1990 and 1993, respectively. After working at the Japan Advanced Institute of Science and Technology, JAIST, Tohoku University, and the University of Tsukuba, he joined Kanazawa University, in 2011. He is currently a Professor with the Faculty of Electrical, Information and Communication Engineering, Institute of Science and Engineering. His research interests include information security, software protection, and privacy protection. He has served as the Co-Editor-in-Chief of the *International Journal of Information Security* (Springer), the Steering Committee Chair of the International Conference on Information Security, and the Chair of the Technical Committee on Information Security (ISEC), Engineering Sciences Society (ESS), Institute of Electronics, Information and Communication Engineers (IEICE).