IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 11 July 2022, accepted 24 July 2022, date of publication 1 August 2022, date of current version 17 August 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3195807

RESEARCH ARTICLE

PSEBVC: Provably Secure ECC and Biometric Based Authentication Framework Using Smartphone for Vehicular Cloud Environment

VINOD KUMAR[1], AMMAR MOHAMMED ALI AL-TAMEEMI[2], ADESH KUMARI[3], MUSHEER AHMAD[4], MAYADAH WAHEED FALAH[5], AND AHMED A. ABD EL-LATIF[6,7]

[1]Department of Mathematics, PGDAV College, University of Delhi, New Delhi 110065, India
[2]Chemical Engineering Department, University of Technology-Iraq, Baghdad 10066, Iraq
[3]Department of Mathematics, Dyal Singh College, University of Delhi, New Delhi 110003, India
[4]Department of Computer Engineering, Jamia Millia Islamia, New Delhi 110025, India
[5]Building and Construction Engineering Technology Department, AL-Mustaqbal University College, Hillah 51001, Iraq
[6]EIAS Data Science Laboratory, College of Computer and Information Sciences, Prince Sultan University, Riyadh 11586, Saudi Arabia
[7]Department of Mathematics and Computer Science, Faculty of Science, Menoufia University, Shebeen El-Kom 32511, Egypt

Corresponding authors: Adesh Kumari (adeshbhucker@gmail.com) and Ahmed A. Abd El-Latif (aabdellatif@psu.edu.sa)

This work was supported by the EIAS Data Science Lab, College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia.

ABSTRACT The Vehicular Cloud Environment (VCE) is a brand-new study field in cloud and vehicular network.It gives cars networking and sensor capabilities for V2I or V2V communication with roadside infrastructure. Cloud applications are frequently used in traffic control and road safety. A hybrid technical solution that utilizes vehicle resources, cloud infrastructure, and Internet of Things (IoT) settings is needed for effective vehicular communication networking. VCE is a smart vehicular communication architecture that promotes system security, enhanced vehicle control, and self-driving cars. Due to the integration of unknown vehicles and infrastructure via the public network, security and privacy seem to be significant challenges with VCE. In this regard, we propose a PSEBVC, which is a provably secure elliptic curve cryptography (ECC) and biometric based authentication system for VCE employing smartphones. In the face of active and passive adversaries, the offered framework obtains the majority of security features and properties for secure communication. We also propose and prove a formal security model based on the random oracle concept. We also demonstrate the security analysis using the Scyther tool. In the same scenario, we evaluate the performance of our protocol against that of other frameworks. The proposed system, according to our findings, is both secure and efficient in terms of communication and processing overhead. The proposed architecture, according to our findings, provides all needed security criteria while also permitting effective communication.

INDEX TERMS Elliptic curve cryptography, V2V communication, V2I communication, authentication, cloud computing, security and privacy.

I. INTRODUCTION

A vehicular ad hoc network (VANET) is a network of vehicles equipped with sensor, communication and network capabilities that connects V2V or V2I for data sharing [1]. According to [2], it can be utilised for a range of things, including entertainment, aberrant vehicle behaviour alert, accident reporting, smart parking, congestion warning, and advertising. Despite the fact that users and drivers are at the root of the phenomenal growth in vehicle usage, a considerable amount of onboard capacity remains chronically unfertilized. To maximise the utilisation of idle apps and boost vehicle capacity, cloud environments are preferred for developing vehicular networking and applications, thus dominating the

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

appearance of VCE [3]. To handle idle vehicular utilizations such as storage and execution for selective methods, VCE integrates amazing roadside and traffic authority information [4]. A cloud computing can be created by coordinating idle onboard resources at a parking lot or on the highway to gather information, technical data and make agreements to enhance the passenger and driver experience at the facility. The vehicular cloud (VC) is a profitable approach for encouraging excessive usage of cars or vehicles that are linked and operated to benefit users. A VC is typically transient and dynamic as a result of vehicle mobility applications. The temporary VC is an important subsidiary for the traditional cloud in terms of increasing storage and other capacity for conventional cloud (CC). VCC is projected to be capable of generating a variety of vehicle services and applications, including road traffic control, enhanced riding, downloading video streams, driving activities, vehicular crowd sensing, among others [5], [6]. Smartphones can act as a critical interface between networks and drivers as the number of smartphone users grows. A smartphone with biosensors, for example, can collect information on a vehicle's driver's physiological status and communicate it to VCE. The warning bell can be activated to raise an alarm in the event of a danger or mishap [7]. VCE-supported apps become more ascendable, enhanceable, and feasible to implement with the integration of smartphones. VCE is projected to play a key role in the construction of better transportation infrastructure. As a result of connectivity, drivers confront additional hazards and obstacles [8]. Cloud-to-phone communications are subject to malicious attacks if defences are poor.

Users can easily authenticate using their smartphones thanks to the Smartphone option offered by authentication. In order to execute out-of-band authentication, a smartphone app is used. Along with the ID and password, out-of-band authentication normally uses two factors and necessitates a supplementary verification over a different communication channel. The Smartphone approach is being used by a user to sign in on an endpoint, such as a laptop or website. The authentication flow is shown in the following steps [9]:

- The endpoint makes interaction with the authentication server when the authentication request is made.
- The credentials of the user are verified by the authentication server.
- After confirming the login information, the authentication server pushes a message to proxy.authasas.com.
- The server decides which push service is most suited for the Smartphone's platform before sending the push message to it.
- The user's smartphone is then informed via the push message that an authentication request has been launched.
- The smartphone app contacts the authentication server when the user launches it to determine whether authentication is required. The Accept and Reject options serve as indicators of authentication. The server is then informed of the user's choice.
- The endpoint is then authorised when the server has verified the authentication.

A. SCENARIO FOR USING THE SMARTPHONE METHOD FOR AUTHENTICATION

On the website myexample.com, the user wants to log in. When he uses the Smartphone authentication method to log into the website, a push notification is issued to his or her smartphone. He/she sees the Accept and Reject buttons when he opens the Smartphone app that is downloaded to his/her phone. The authentication request is sent back to the authentication framework through the mobile network (secure) if he chooses the Accept option. The user authenticates to myexample.com without providing an OTP code. User can use a backup OTP for offline authentication when your smartphone doesn't have a network connection.

B. RELATED WORK

The literature that has been published that is pertinent to the suggested protocols is briefly reviewed in this section. In 2008, Zhang *et al.* [10] presented a pairing-based cryptography-based identity-based authentication architecture. Vehicles and roadside units (RSU) are not used to store documents in this work. Furthermore, their method provides batch confirmation for multiple data exchanges. Conditional privacy-preserving authentication (CPPA) frameworks were proposed by Lu *et al.* [11] and Raya and Hubaux [12]. Jiang *et al.* [13] presented a binary tree-based authentication system in 2009, in which the RSU could interpret the data collected from the genuine entity right away. Shim [14] demonstrated that an attacker can substantially modify information on two fake messages in the work of Jiang *et al.* [13]. Shim has used a pseudo number method to provide a conditional privacy-preserving authentication solution for secure VANETs. During the verification phase, Lo and Tsai [15] investigated Shim's technique and discovered an error. According to [16], Li and Liu created a lightweight key agreement work for VANET in 2013 to increase the key agreement method's capacity while masking the vehicles' vulnerable information. Lee and Lai then presented [17], a batch verification system for the VANET that includes group verification. In 2015, He *et al.* [18] published an ID-based authentication strategy for VANETs based on Schnorr's signature programme [19]. The authors of this proposed protocol presented a solution to the [10] protocol proposed by Zhang *et al.* In 2016, Oulhaci *et al.* published [20], a protocol for a secure and distributed VANET message authentication system. The following year, Lee *et al.*, based on the Chinese remainder theorem, developed a safer and faster batch key-agreement process for communication channels. Zhang *et al.* [21] published an unique privacy-preserving authentication technique for VANETs in 2017. RSU is bad because it is in charge of a VANET vehicle's private key. Furthermore, Zhang [22] propose a VANET system that uses cryptographic mix-zones to combat malicious attackers while ensuring privacy. [23] was released by Asaar *et al.* in 2018. In the same year, Li *et al.* submitted EPA-CPPA: An efficient and secure anonymous conditional privacy-preserving authentication system for VANETs [24]. A VCE integrated authentication solution was recently introduced by Jiang *et al.* [25]. However, according to [26], it is vulnerable to mutual authentication, forward security, de-synchronization, impersonation, insider attacks, and parallel attacks, for mobile cloud computing services, He *et al.* presented an efficient authentication approach. However, it falls short when it comes to impersonation and undetectable attacks. In 2019, Zhang *et al.* proposed a chinese remainder theorem based conditional privacy-preserving authentication scheme in vehicular VANET [27]. In same year, Cui *et al.* suggested a reputation system-based lightweight message authentication framework and protocol for 5G-enabled vehicular networks [28]. In 2020, Irshad *et al.* proposed a provably secure and efficient authenticated key agreement scheme for energy internet-based vehicle-to-grid technology framework [29]. This work the authors discussed the security drawbacks of Gope and Sikdar [30], further provided enhance protocol in same direction. In same year, Zhang *et al.* proposed an edge computing-based privacy-preserving authentication framework and protocol for 5G-enabled vehicular networks [31]. In 2021, Chaudhry *et al.* proposed a lightweight authentication scheme for 6G-IoT enabled maritime transport system [32]. In 2022, Son *et al.* presented a design of blockchain based lightweight V2I handover authentication protocol for VANET [33] which fails against de-synchronization property. Most recently, there are four different authentication and key agreement schemes [34]–[37] have presented by the authors. In same year, Kumar *et al.* proposed a robust authentication protocol for IoMT-based cloud-healthcare infrastructure which is secure and efficient [38].

C. MOTIVATION AND CONTRIBUTION

The significance of privacy and security issues around VCE cannot be overstated. Mutual authentication between entities is required before sharing any sensitive data. Although VCE-based authentication schemes [25], [26], [33], [39]–[41] have been introduced throughout the previous few decades, their success is unsatisfactory. Furthermore, these frameworks interfere with protocol's core obligations, resulting in a basic breach. Now, we aim to introduce a new secure ECC and biometric-based authenticated key agreement system using smartphone. Many important features of the proposed framework include:

- PSEBVC uses CC to establish authentication between U and VC.
- PSEBVC can also provide a variety of security features and options.
- Using CC, the session key is created between U and VC.
- We discuss the security simulation via Scyther tool.
- A random oracle model is used to create a formal security model and security analysis for PSEBVC.
- PSEBVC is more efficient than other protocols, according to the performance analysis phase.

D. ROAD MAP OF THE PAPER

The remainder of the paper is organised as follows. We present the useful mathematical preliminaries in Section II. The PSEBVC protocol is covered in Section III. Section IV: PSEBVC security analysis. Section V: PSEBVC performance analysis. Finally, we talk about a conclusion. In addition, as shown in Table 1, we provide symbols/notation.

TABLE 1. Notations.

Symbol	Description
U	i^{th} User
q	Large prime
SP	The smartphone of U
$i \stackrel{?}{=} j$	Whether i and j are equal
VC	j^{th} Vehicular cloud
CC	Conventional cloud
$\approx$	Approximate value
SP	i^{th} User's smartphone
$E_K(M)/D_K(M)$	Using the key K, encrypt/decrypt message M.
$\mathcal{A}$	Adversary
pw_U	Password of U
$i \cdot\cdot \Rightarrow j : \{M\}$	Through a secure channel, i sends M to j
B_{SP}	Biometric of U
ID_i	The i^{th} participant's identity
$\triangle T$	Maximum communication time delay
Z_q^*	Multiplicative group of order $q-1$
$SK_{ij}(.)$	Entities i and j share the session key.
$i \cdot\cdot \rightarrow j : \{M\}$	i uses the public channel to send M to j
SL	The simulator

II. PRELIMINARIES

A. SECURE HASH FUNCTION

Definition: A one-way hash function $h_i : \{0, 1\}^* \rightarrow \{0, 1\}^l$ accepts a string input of any length $x \in \{0, 1\}^*$ and outputs a string of a finite length $h(.) \in \{0, 1\}^l$.

The following qualities are what a top hash function should have [42]:

- For any input value x, it is possible to derive the digest, $h(x)$.
- **One-way:**For a given hash value, $y = h(x)$, it is not computationally viable to obtain x.
- **Weak-collision resistance:** For any given input x, it is computationally impossible to get any additional input y with $x \neq y$ such that $h(x) = h(y)$.
- **Strong-collision resistance:** Additionally, finding two inputs (x, y) with $x \neq y$ such that $h(x) = h(y)$ cannot be done computationally

B. ASSUMPTIONS FOR THE MUTUAL AUTHENTICATION PROTOCOL

In order to evaluate the invoked mutual authentication mechanism, we make some assumptions:

1. The secret numbers, the random number, and the hash results are all stored on the cloud server. They achieve the desired safe length l.
2. The encryption E_k, decryption D_k, and hash function $h(.)$ are able. In other words, no one can detect the collision of $h(M)$, where M is the string and $E_k(M)$ is an encrypted string that cannot be cracked in polynomial time without knowing k.
3. The entity has low entropy in both its identification and one-time password (OTP). There are two dictionaries: an identities dictionary and an OTP dictionary. They can be predicted in polynomial time by an attacker.
4. The previous session's keys can be obtained by the enemy through known-key attacks.

C. ELLIPTIC CURVE CRYPTOGRAPHY OVER FINITE FIELD

Let q stand for the huge prime and F_q for the prime finite field of order q. The equation of elliptic curve (EC) is defined as $\nu^2 = \mu^3 + c\mu + d \ mod \ q$, where $c, d \in F_q$. EC is said to be non singular if $4c^3 + 27d^2 \ mod \ q \neq 0$. The additive EC group defined as $G = \{(\mu, \nu) : \mu, \nu \in F_q; (\mu, \nu) \in \mathcal{E}\} \bigcup \{\Phi\}$, where Φ is the zero or identity element of G and G satisfies the following operations [43]:

1. If $\bigvee = (\mu, \nu) \in G$, then $-\bigvee = (\mu, -\nu)$ and $\bigvee + (-\bigvee) = \Phi$.
2. If $\bigvee_1 = (\mu_1, \nu_1), \bigvee_2 = (\mu_2, \nu_2) \in G$, then $\bigvee_1 + \bigvee_2 = (\mu_3, \nu_3)$, where $\mu_3 = \delta^2 - \mu_1 - \mu_2 \ mod \ q$, $\nu_3 = \delta(\mu_1 - \mu_3) - \nu_1 \ mod \ q$, and

$$\delta = \begin{cases} \frac{\nu_2-\nu_1}{\mu_2-\mu_1} \ mod \ q \ if \ \bigvee_1 \neq \bigvee_2 \\ \frac{3\mu_1^2+c}{2\nu_1} \ mod \ q \ if \ \bigvee_1 = \bigvee_2 \end{cases}$$

3. Let $\bigvee = (\mu, \nu) \in G$ then, scalar mortification of G defined as: $n.\bigvee = \bigvee + \bigvee + \cdots\cdots\cdots + \bigvee$ ($n-$ $times$).

1) COMPUTATIONALLY DIFFICULT PROBLEM BASED ON ECC

* **(ECDLP) Elliptic curve discrete logarithms problem** : If $W1, W2 \in G$, then it is hard to evaluate $\nu \in Z_q^*$ such that $W2 = \nu W1$ [42].
* **(ECCDHP) Elliptic curve computational Diffie-Hellman problem** :The generator of G is g for $\alpha, \beta \in Z_q^*$. For the given $(g, \alpha g, \beta g)$, it is difficult to compute $\alpha\beta g$ in G

D. BASIC OF BIOMETRIC AND FUZZY EXTRACTOR

In an error-tolerant manner, a fuzzy extractor (Y, m, l, t, ϵ) extracts a closely random string σ_i from its biometrics input ω, where Y denotes the metric space, m the min-entropy of any computiion on Y, l the number of bits in the borrowed biometric key, and t the mistake acceptance dawn. The effort varies depending on the extractor, but it always leaves the same amount of overs surrounding the mined σ_i relics [44]. Two processes Gen and Rep of define the fuzzy extractor:

- **Gen:** is a probabilistic generation approach that accepts $\omega \in Y$ and returns a derived string $\sigma_i \in \{0, 1\}^l$, referred to as the biometric key, and a supplementary string τ_i, referred to as the public propagation parameter, which is $(\sigma_i, \tau_i) \leftarrow Gen(\omega)$.
- **Rep:** is a deterministic reproduction technique that allows σ_i to be recovered from the conforming auxiliary series τ_i and any vector ω' that is near to ω. For all $\omega, \omega' \in Y$ satisfying the hamming distance $d(\omega, \omega') \leq t$ if $(\sigma_i, \tau_i) \leftarrow Gen(\omega)$, then $Rep(\omega', \tau_i) = \sigma_i$.

III. THE PSEBVC FRAMEWORK

In this session, we will go over our PSEBVC protocol. The proposed architecture is shown in Figure 1. In the architecture, there are three entities as follows:

- **Smartphone user:** Due to their portability and ability to run a variety of applications, smartphones have gained a lot of popularity. However, smartphones' portability also places weight and size restrictions on them. As a result, some resources on smartphones, such as computing and storage resources, are constrained. Smartphones' processing speed and memory capacity are continually increasing, however they still fall short of some mobile applications' needs for computationally demanding mobile applications. Smartphones perform poorly while running several complicated apps, such as image processing, gaming, and so forth. Cloud computing is used to assist effective application execution on smartphones because of the vast resources on the cloud platform [45].
- **Vehicular cloud:** The automobiles, buses, and trucks that are on the road may come together to produce a localised tiny vehicular cloud. The network endpoint devices needed to turn buses and large trucks into network access points, such WiFi hotspots, may be transported on board. The WiFi endpoint on the buses may be accessed by the other vehicles for Internet information. Due to the large number of devices that buses and

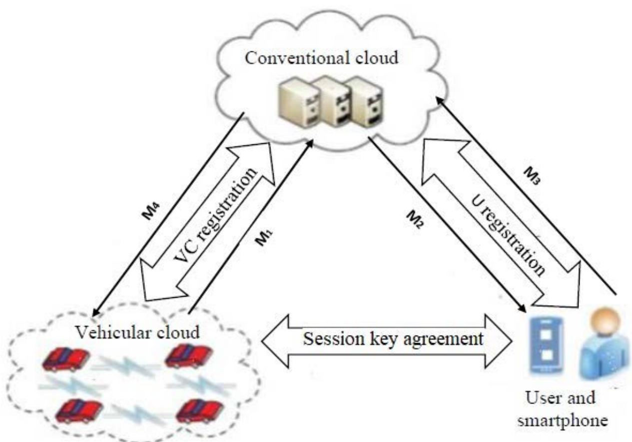


FIGURE 1. The VCE registration and authentication architecture.

large trucks can transport as well as the fact that they frequently follow a set schedule, the network coverage and signal need to be predictable and steady in order for neighbouring automobiles to have a strong connection to them [46].

- **Conventional cloud:** It refers to the internet-based distribution of various services, including data and software, on various servers. It refers to the provision of various services via a local server. It occurs on third-party servers that are hosted by third-party hosting firms.

PSEBVC uses CC to securely communicate with U and VC while also preserving the session key. PSEBVC is divided into five stages. The following phases are described in detail:

A. INITIALIZATION PHASE

The CC takes the following steps:

Step 1. CC chooses the large prime number q and prime finite field $Z_q^\star$.

Step 2. CC selects a nonsingular EC with the equation $v^2 = u^3 + cu + d \bmod q$ over F_q.

Step 3. CC selects a random value $X_{CC} \in Z_q^\star$.

Step 4. CC generates g from G.

Step 5. CC chooses hash function $h(.)$. Where $h_i : \{0,1\}^* \rightarrow \{0,1\}^l$.

Step 5. CC publishes parameters $\{F_q, EC, h(.), q, g, Gen(.), Rep(.)\}$. Where X_{CC} is keep secret.

B. USER REGISTRATION PHASE

U employs the CC registration form, which is described further below:

Step 1. U inputs ID_U, pw_U, to register with CC together imprints B_U, generates random number $r_U \in Z_q^\star$, computes $(\sigma_U, \tau_U) = Gen(B_U)$, $S_U = h(pw_U \| \sigma_U) \oplus r_{SP}$ and $U \Rightarrow CC : M_{R1} = \{ID_U, S_U, t_{R1}\}$.

Step 2. On receiving M_{R1}, CC verifies $t_{R2} - t_{R1} \leq \Delta t$. Then, CC computes $\alpha = h(ID_U \| x_{CC} \| \gamma)$, where x_{CC} represents the CC secret key and γ represents the registration counter $\gamma = 0$ if U is a new registered user. Otherwise, $\gamma = \gamma + \gamma + \gamma \cdots$ Further, CC inserts $\{ID_U, \gamma\}$ in database. Then, CC, computes $\alpha_1 = \alpha \oplus S_U$, and stores $\{\alpha_1, \gamma, g, q, G, h(.)\}$ in database for ID_U. Further, $CC \Rightarrow U : M_{R2} = \{\alpha_1, \gamma, g, q, G, h(.)\}$.

Step 3. On receiving M_{R2}, U computes $\alpha_2 = \alpha_1 \oplus \sigma_U$, $\alpha_3 = h(ID_U \| pw_U \| \alpha_2 \| r_{SP})$ and stores $\{\alpha_1, \gamma, g, q, G, h(.), \tau_U, \alpha_2, \alpha_3\}$ in database.

The process of RP is shown in Table.2.

C. VEHICULAR CLOUD REGISTRATION PHASE

VC receives the registration form CC, as shown below:

Step 1. Sends $VC \Rightarrow CC : \{ID_{VC}\}$.

Step 2. On receiving ID_{VC}, CC verifies ID_{VC} in database. After that, CC calculates $\xi = h(ID_{VC} \| x_{CC})$ and $CC \Rightarrow VC : \{\xi\}$.

TABLE 2. The phase of user registration via a secure channel.

User U with SP	Conventional cloud CC
Inputs ID_U, pw_U and imprints B_U Generates random number $r_{SP} \in Z_q^\star$ Computes $(\sigma_U, \tau_U) = Gen(B_U)$ Computes $S_U = h(pw_U \| \sigma_U) \oplus r_{SP}$ Sends $M_{R1} = \{ID_U, S_U, t_{R1}\}$ $\cdots\cdots\Rightarrow$	
	$t_{R2} - t_{R1} \leq \Delta t$ Computes $\alpha = h(ID_U \| x_{CC} \| \gamma)$ where x_{CC} is the private key of CC, The registration counter is denoted by the symbol γ If U is a first-time user, $\gamma = 0$. Otherwise, $\gamma = \gamma + \gamma + \gamma$..... Includes $\{ID_U, \gamma\}$ in database Computes $\alpha_1 = \alpha \oplus S_U$ Stores $\{\alpha_1, \gamma, g, q, G, h(.)\}$ in SC_U corresponding to ID_U Sends $M_{R2} = \{L_1, SC_U\}$ $\Leftarrow\cdots\cdots$
Computes $\alpha_2 = \alpha_1 \oplus \sigma_U$ Computes $\alpha_3 = h(ID_U \| pw_U \| \alpha_2 \| r_{SP})$ Store $\{\tau_U, \alpha_2, \alpha_3\}$ into SC_U	

TABLE 3. Phase of vehicle cloud registration via secure channel.

Vehicular cloud VC	Conventional cloud CC
Sends $\{ID_{VC}\}$ $\cdots\cdots\Rightarrow$	
	Verifies the ID_{VC} in databae Computes $\xi = h(ID_{VC} \| X_{CC})$ Sends $\{\xi\}$ $\Leftarrow\cdots\cdots$
Selects $r_{VC} \in Z_q^\star$ Computes $PK_{VC} = r_{VC}.g$	

Step 3. On receiving ξ, VC generates random number $r_{VC} \in Z_q^\star$ and sets as private key. Further, VC computes public key $PK_{VC} = r_{VC}.g$.

D. LOGIN, AUTHENTICATION AND KEY MANAGEMENT PHASE

In this session, U and VC will authenticate one another using CC and maintain the following session key:

Step 1. U login with ID'_U and pw'_U, imprints B'_U and receives $\sigma'_U = Rep(B'_U, \tau'_U)$. Then, SP computes $\alpha'_2 = \alpha'_1 \oplus \sigma'_U$, $\alpha'_3 = h(ID'_U \| pw'_U \| \alpha'_2 \| r_{SP})$ and verifies $\alpha'_3 \stackrel{?}{=} \alpha_3$. Then, SP generates random number $x \in Z_q^\star$, computes $H_1 = h(ID_U \| \alpha_1 \| x)$, encrypts $E_1 = E_{h(ID_U \| \alpha_1 \| t_1)}(H_1, x)$ and $U \rightarrow CC : M_1 = \{E_1, t_1\}$.

Step 2. On receiving M_1, CC verifies $t_2 - t_1 \leq \Delta t$. Then, CC decrypts $(H_1, x) = D_{h(ID_U \| \alpha_1 \| t_1)}(E_1)$ and verifies $H_1^* \stackrel{?}{=} h(ID_U \| \alpha_1 \| x)$. Further, CC generates number $z \in Z_q^\star$, computes $H_2 = h(H_1^* \| ID_{VC} \| t_3)$, $ID_{U1} = ID_U \oplus h(H_1^* \| H_2)$, encrypts $E_2 = E_{h(ID_{VC} \| \xi \| t_3)}(H_1^*, z, ID_{U1}, x, H_2, \gamma, t_3)$ and $CC \rightarrow VC : M_2 = \{E_2, t_3\}$.

Step 3. On receiving M_2, VC verifies $t_4 - t_3 \leq \Delta t$. Then, VC decrypts $(H_1^*, z, ID_{U1}, x, H_2, \gamma, t_3) = D_{h(ID_{VC} \| \xi \| t_3)}(E_2)$ and again verifies $H_2^* \stackrel{?}{=} h(H_1^* \| ID_{VC} \| t_3)$. Further, VC computes $ID_U^* = ID_{U1} \oplus h(H_1^* \| H_2^*)$, generates random number $y \in Z_q^\star$, computes $H_3 = h(ID_U^* \| ID_{VC} \| z \| H_2^* \| y)$, session key $SK_{VC} = h(ID_U^* \| ID_{VC} \| \gamma \| z \| H_1^* \| H_3$

TABLE 4. Login, authentication and key management phase via public channel.

User U	**Conventional cloud** CC	**Vehicular cloud** VC
U login with ID'_U and pw'_U Imprints B'_U and gets $\sigma'_U = Rep(B'_U, \tau'_U)$ Computes $\alpha'_2 = \alpha'_1 \oplus \sigma'_U$ Computes $\alpha'_3 = h(ID'_U \parallel pw'_U \parallel \alpha'_2 \parallel r_{SP})$ Checks $\alpha'_3 \stackrel{?}{=} \alpha_3$ Selects $x \in Z_q^\star$ Calculates $H_1 = h(ID_U \parallel \alpha_1 \parallel x)$ Encrypts $E_1 = E_{h(ID_U \parallel \alpha_1 \parallel t_1)}(H_1, x)$ Sends $M_1 = \{E_1, t_1\}$ $\cdots\cdots\cdots\cdots\rightarrow$		
	Verifies $t_2 - t_1 \leq \triangle t$ Decrypts $(H_1, x) = D_{h(ID_U \parallel \alpha_1 \parallel t_1)}(E_1)$ Verifies $H_1^* \stackrel{?}{=} h(ID_U \parallel \alpha_1 \parallel x)$ Generates $z \in Z_q^\star$ Computes $H_2 = h(H_1^* \parallel ID_{VC} \parallel t_3)$ Computes $ID_{U1} = ID_U \oplus h(H_1^* \parallel H_2)$ Encrypts $E_2 = E_{h(ID_{VC} \parallel \xi \parallel t_3)}(H_1^*, z, ID_{U1}, x, H_2, \gamma, t_3)$ Sends $M_2 = \{E_2, t_3\}$ $\cdots\cdots\cdots\cdots\rightarrow$	
		Verifies $t_4 - t_3 \leq \triangle t$ Decrypts $(H_1^*, z, ID_{U1}, x, H_2, \gamma, t_3) = D_{h(ID_{VC} \parallel \xi \parallel t_3)}(E_2)$ Verifies $H_2^* \stackrel{?}{=} h(H_1^* \parallel ID_{VC} \parallel t_3)$ Computes $ID_U^* = ID_{U1} \oplus h(H_1^* \parallel H_2^*)$ Selects $y \in Z_q^\star$ Calculates $H_3 = h(ID_U^* \parallel ID_{VC} \parallel z \parallel H_2^* \parallel y)$ Computes $SK_{VC} = h(ID_U^* \parallel ID_{VC} \parallel \gamma \parallel z \parallel H_1^* \parallel H_3 \parallel xyg \parallel t_5)$ Computes $ID_{VC1} = ID_{VC} \oplus h(H_1^* \parallel H_2^* \parallel t_5)$ Encrypts $E_3 = E_{h(H_1^* \parallel x \parallel t_5)}(H_3, z, y, H_2^*, ID_{VC1}, t_5)$ Sends $M_3 = \{E_3, t_5\}$ $\leftarrow\cdots\cdots\cdots\cdots$
	Verifies $t_6 - t_5 \leq \triangle t$ Sends $M_4 = \{E_3, t_5, t_7\}$ $\leftarrow\cdots\cdots\cdots\cdots$	
Verifies $t_8 - t_7 \leq \triangle t$ Computes $(H_3, z, y, H_2^*, ID_{VC1}, t_5) = D_{h(H_1 \parallel x \parallel t_5)}(E_3)$ Computes $ID_{VC}^* = ID_{VC1} \oplus h(H_1 \parallel H_2^* \parallel t_5)$ Verifies $H_3^* \stackrel{?}{=} h(ID_U \parallel ID_{VC}^* \parallel z \parallel H_2^* \parallel y)$ Computes $SK_U = h(ID_U \parallel ID_{VC}^* \parallel \gamma \parallel z \parallel H_1 \parallel H_3^* \parallel xyg \parallel t_5)$		

$\|xyg\|t_5)$, $ID_{VC1} = ID_{VC} \oplus h(H_1^*\|H_2^*\|t_5)$, encrypts $E_3 = E_{h(H_1^*\|x\|t_5)}(H_3, z, y, H_2^*, ID_{VC1}, t_5)$ and $VC \rightarrow CC : M_3 = \{E_3, t_5\}$.

Step 4. On receiving M_3, CC verifies $t_6 - t_5 \leq \triangle t$. Further, $CC \rightarrow U : M_4 = \{E_3, t_5, t_7\}$.

Step 5. On receiving M_4, SP verifies $t_8 - t_7 \leq \triangle t$. Further, SP decrypts $(H_3, z, y, H_2^*, ID_{VC1}, t_5) = D_{h(H_1\|x\|t_5)}(E_3)$, computes $ID_{VC}^* = ID_{VC1} \oplus h(H_1\|H_2^*\|t_5)$ and verifies $H_3^* \stackrel{?}{=} h(ID_U\|ID_{VC}^*\|z \|H_2^*\|y)$. After that, SP sets session key $SK_U = h(ID_U\|ID_{VC}^*\|\gamma\|z\|H_1\|H_3^*\|xyg\|t_5)$.

Hence, the authentication procedure is completed by U and VC, and both parties agree on a session key $SK = SK_{VC} = SK_U$.

E. PASSWORD AND BIOMETRIC CHANGE PHASE

U takes the following procedures to alter his or her personal password and biometric:

Step 1. U inputs ID'_U, pw'_U, imprint B'_U and archives $\sigma'_U = Rep(B'_U, \tau'_U)$. Then, SP computes $\alpha'_2 = \alpha'_1 \oplus \sigma'_U$, $\alpha'_3 = h(ID'_U\|pw'_U\|\alpha'_2\|r_{SP})$. Further, SP checks whether $\alpha'_3 \stackrel{?}{=} \alpha_3$ holds true or not.

Step 2. The session is terminated if SP does not validate the condition. Otherwise, U selects a new password pw_U^{NEW} as well as a new biometric B_U^{NEW}. Then, SP computes $(\sigma_U^{NEW}, \tau_U^{NEW}) = Gen(B_U^{NEW})$, $S_U^{NEW} = h(pw_U^{NEW}\|\sigma_U^{NEW}) \oplus r_{SP}$ and $U \rightarrow CC : M_{NR1} = \{ID_U, S_U^{NEW}, T_{NR1}\}$.

Step 3. CC validates $t_{NR2} - t_{NR1} \leq \triangle t$ when it receives M_{NR1}. Then CC checks the database for $\{ID_U, \gamma\}$. If yes, $\alpha_1^{NEW} = \alpha \oplus S_U^{NEW}$ and $CC \rightarrow U : M_{NR2} = \{\alpha_1^{NEW}, SC_U\}$ are computed.

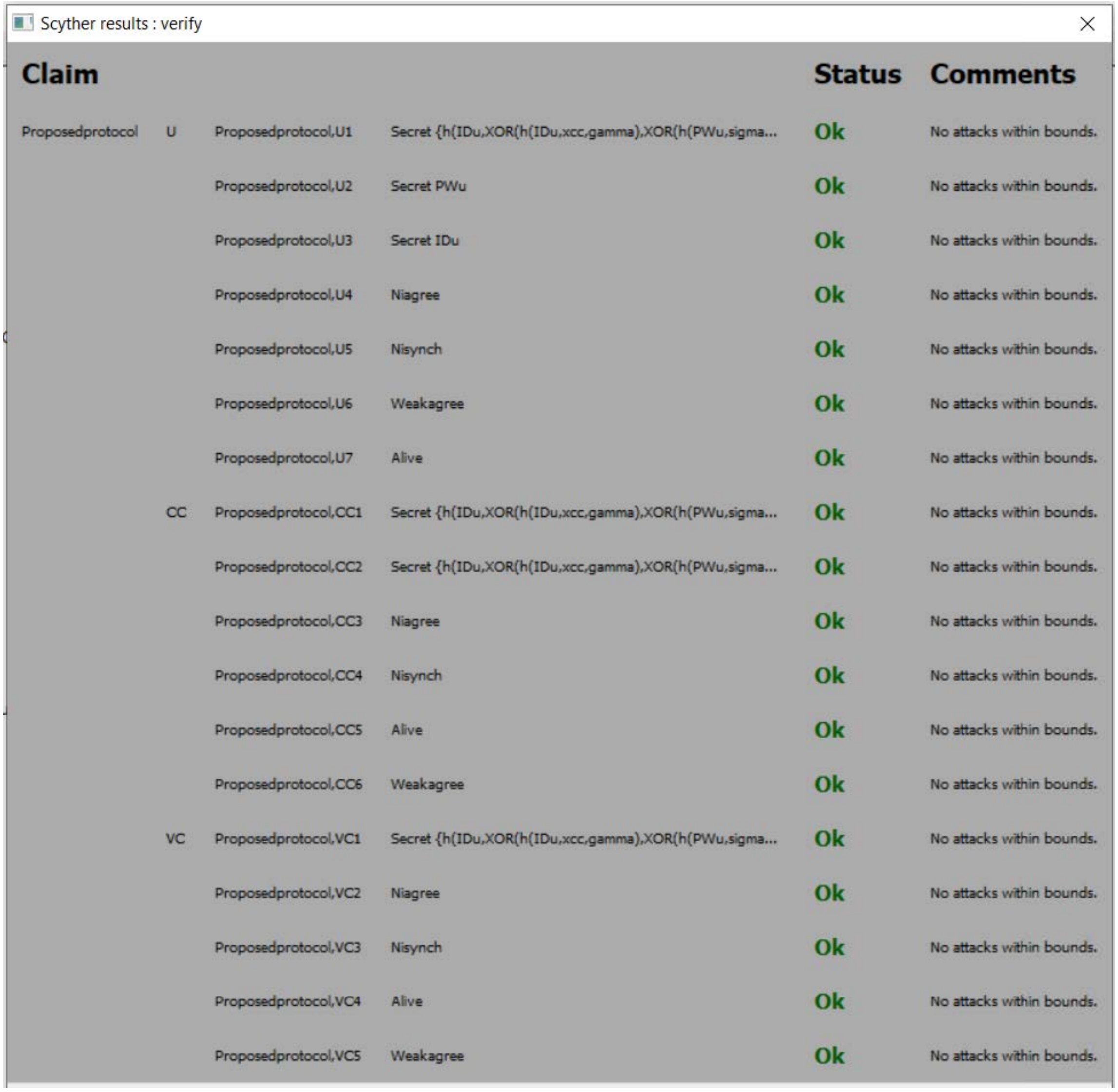
Step 4. On receiving M_{NR2}, SP generates $r_{SP}^{NEW} \in Z_q^\star$, computes $\alpha_2^{NEW} = \alpha_1^{NEW} \oplus \sigma_U^{NEW}$ and $\alpha_3^{NEW} = h(ID_U\|pw_U^{NEW}\|\alpha_2^{NEW}\|r_{SP}^{NEW})$. After that, U replaces pw_U by pw_U^{NEW}, α_1 by α_1^{NEW}, α_2 by α_2^{NEW}, α_3 by α_3^{NEW}, σ_U by σ_U^{NEW} and τ_U by τ_U^{NEW}. Finally, U stores $\{\tau_1^{NEW}, \alpha_2^{NEW}, \alpha_3^{NEW}\}$ in database.

IV. SECURITY ANALYSIS OF THE PSEBVC

We will talk about PSEBVC's security analysis in this session. The PSEBVC verifies three types of security analyses:

A. SECURITY ANALYSIS VIA SCYTHER TOOL

Scyther is a vulnerability analysis tool with a user interface built in Python. This interface facilitates doing protocol security analysis and analysing the findings straightforward for the intended users. To test the proposed approach, the Scyther tool was employed. Scyther can perform a variety of attacks on authentication protocols and display the results. Secret, Nisynch, Nialive, and Niagree are four statements that our Scyther tool model examines. The following



Scyther results : verify

Claim				Status	Comments
Proposedprotocol	U	Proposedprotocol,U1	Secret {h(IDu,XOR(h(IDu,xcc,gamma),XOR(h(PWu,sigma...	Ok	No attacks within bounds.
		Proposedprotocol,U2	Secret PWu	Ok	No attacks within bounds.
		Proposedprotocol,U3	Secret IDu	Ok	No attacks within bounds.
		Proposedprotocol,U4	Niagree	Ok	No attacks within bounds.
		Proposedprotocol,U5	Nisynch	Ok	No attacks within bounds.
		Proposedprotocol,U6	Weakagree	Ok	No attacks within bounds.
		Proposedprotocol,U7	Alive	Ok	No attacks within bounds.
	CC	Proposedprotocol,CC1	Secret {h(IDu,XOR(h(IDu,xcc,gamma),XOR(h(PWu,sigma...	Ok	No attacks within bounds.
		Proposedprotocol,CC2	Secret {h(IDu,XOR(h(IDu,xcc,gamma),XOR(h(PWu,sigma...	Ok	No attacks within bounds.
		Proposedprotocol,CC3	Niagree	Ok	No attacks within bounds.
		Proposedprotocol,CC4	Nisynch	Ok	No attacks within bounds.
		Proposedprotocol,CC5	Alive	Ok	No attacks within bounds.
		Proposedprotocol,CC6	Weakagree	Ok	No attacks within bounds.
	VC	Proposedprotocol,VC1	Secret {h(IDu,XOR(h(IDu,xcc,gamma),XOR(h(PWu,sigma...	Ok	No attacks within bounds.
		Proposedprotocol,VC2	Niagree	Ok	No attacks within bounds.
		Proposedprotocol,VC3	Nisynch	Ok	No attacks within bounds.
		Proposedprotocol,VC4	Alive	Ok	No attacks within bounds.
		Proposedprotocol,VC5	Weakagree	Ok	No attacks within bounds.

FIGURE 2. **Scyther test results.**

are the main points: Niagree is a noninjective synchronisation that assures that the content of the message exchanged between the sender and the recipient is not tampered with and that the communication is completed according to protocol. Nisynch makes sure that communication packets are sent in the correct order and that the protocol is running smoothly. Secret claims guarantee the confidentiality of all messages sent and received [47]. As demonstrated in Figure 2 of the Scyther tool result screen, the protocol passed all of the attack tests. The results of the Scyther tool show that attacking each level's authentication methods is impossible; consequently, the authentication strategy has been confirmed secure.

B. FORMAL SECURITY EVALUATION

We go over the random oracle model's security model and analysis in this phase:

1) FORMAL SECURITY MODEL

In this phase, we adopt random oracle method based on [42], [48]–[50]. We make some changes to fit our protocol.

We use two ECC assumptions based on Section II-C analysis to prove the correctness of PSEBVC.

- $*$ **Elliptic curve decisional Diffie-Hellman problem (ECDDHP)**: Let $\lambda g, \mu g, \nu g \in G$. The probability for $\mathcal{A}$ to decide whether $\nu g = \lambda \mu g$ polynomial time κ is $Advt_{\mathcal{A}}^{ECDDHP}(\tau)$ and ϵ is an ignorably small positive real number, where $Advt_{\mathcal{A}}^{ECDDHP}(\kappa) \leq \epsilon$ [42].
- $*$ **Elliptic curve gap Diffie-Hellman problem (ECGDHP)**: Let $\lambda g, \mu g \in G$. The probability of $\mathcal{A}$ computing $\lambda \mu g$ in polynomial time κ using an ECDDHP oracle is $Advt_{\mathcal{A}}^{ECDDHP}(\kappa) \leq \epsilon$ [42].

2) PROOF OF FORMAL SECURITY EVALUATION

Theorem: The protocol Π operates on an ECC-added additive cyclic group G with a high prime order q. Whereas the password dictionary $\mathcal{D}$ has a size of $\mathcal{N}$. If $\mathcal{A}$ performs no other queries than q_s Send queries, q_h Hash queries, and q_e Execute queries.Then

$$Adv_{\Pi}^{sfs-ake}(\mathcal{A}) \leq \frac{O(q_s+q_e)^2}{(q-1)} + \frac{O(q_h)^2+O(q_s+q_e)^2}{2^l} + \frac{O(q_h)+O(q_s)}{2^{l-1}} + \frac{O(q_s)}{\mathcal{N}} + O((q_h(q_s+q_e)^2+1)Adv_{\mathcal{A}}^{ECDDH}(\kappa'),$$

where $\kappa' = t+(O(q_e)+O(q_s))T_M$ and T_M is the time for one multiplication in G.

Proof: We demonstrate the preceding theorem using the game arrangement. We use 9 Games ranging from G_0 to G_8 in this example. Su_j is the event in game G_j for $\mathcal{A}$ accurately estimating the coin θ via the analysis session. Because these games only have one user U, $\mathcal{A}$ wishes to perform user identity ID_U. The procedure is as follows:

* G_0: With the random oracle appoarch, the actual game for the login and authentication phase of the protocol is G_0, and we have
$$Advt_{\Pi}^{sfs-ake}(\mathcal{A}) = 2Prob[Su_0] - 1 \tag{1}$$
Furthermore, if there are several occurrences, a random θ^* is used as a response. There are several unusual occurrences, such as the ones listed below:
  - Since $\mathcal{A}$ has not guessed θ^*, the game will end or be removed.
  - $\mathcal{A}$ does more queries than the based on upper bound.
  - $\mathcal{A}$ spends more time than the planned upper bound.
* G_1: The total of all SL queries is used for this game. Three lists to help you focus on the answers to the questions.
  - L_H: All hash searches have a solution, which is represented by this object.
  - L_P: The transcript of the communication is represented by this object.
  - L_E: It's the result of $\mathcal{A}$'s rigorous query of the two random oracles.

  Table 5 displays the queries. G_1 and G_0 are indistinguishable with the preceding information, and we notice that
$$Prob[Su_1] - 1 = Prob[Su_0] \tag{2}$$
* G_2: We're looking for ways to get rid of the affects in the transcripts. We explained the likelihood of them in the same way we explained the birthday paradox:
  - In the situation, $a, b, d \in Z_q^{\star}$ could be a smash special session and upper bound.
$$\frac{O(q_s+q_e)^2}{2(q-1)} + \frac{O(q_s+q_e)^2}{2^{l+1}}$$
  - It's possible that the hash outputs will collide, resulting in an upper bound on the position $\frac{O(q_h)^2}{2^{l+1}}$.

  Except for the appearance of collisions, G_2 and G_1 are equivalent. We'll look into it.
$$|Prob[Su_2] - Prob[Su_1]| \leq \frac{O(q_s+q_e)^2}{2(q-1)} + \frac{O(q_h)^2+O(q_s+q_e)^2}{2^{l+1}} \tag{3}$$
* G_3: The probability for M_1 is acknowledged here, and $\mathcal{A}$ forges M_1. We connect some steps on *Send* (U^i, CC^t, M_1) because the simulator wants to verify if M_1 is in L_P and $(ID_U\|\alpha_1\|\star, H_1) \in L_E$. If this query fails, it will be terminated. If checks are taken into account, G_3 and G_2 are proportional. Then we'll be able to achieve
$$|Prob[Su_3] - Prob[Su_2]| \leq \frac{O(q_s+q_e)}{2^l} \tag{4}$$
* G_4:The likelihood of forging M_2 is considered here. Because SL responds with VC, we add few steps on *Send* (CC^t, VC^j, M_2) the simulator wants to verify if $M_2 \in L_P, (\star\|ID_{VC}\|t_3, H_2), (H^*\|H_2, ID_{U1}), \in L_E$. It will be stopped if this query fails. If the verifiers are taken into account, G_4 and G_3 are equivalent. As we can see
$$|Prob[Su_4] - Prob[Su_3]| \leq \frac{O(q_s+q_e)}{2^l} \tag{5}$$
* G_5: The probability of a bogus message M_3 is examined here. Since SL is the reader, the response is provided by SL. On *Send* (VC^j, CC^t, M_3), we add some steps. SL wants to know if $M_3 \in L_P$ and $(H_1^*\|ID_{VC}\|t_3, H_2^*), (H_1^*\|H_2^*, ID_U^*), (ID_U^*\|ID_{VC}\|\star\|H_2^*\|\star, H_3)(ID_U^*\|ID_{VC}\|\gamma\|\star\|H_1^*\|H_3\|\star\|t_5, SK_{VC}), (H_1^*\|H_2^*\|t_5, ID_{VC1}), (H_1^*\|\star\|t_5) \in L_E$. If the query fails, it will be terminated. If verification is under consideration, G_5 and G_4 are same. As a result, we discovered
$$|Prob[Su_5] - Prob[Su_4]| \leq \frac{O(q_h+q_s)}{2^l} \tag{6}$$
* G_6: The probability of a bogus message M_4 is examined here. Since SL returns a CC response. We add some steps on *Send* (CC^t, U^i, M_4), SL wants to validate if $M_4 \in L_P$. If this query fails, it will be terminated. If checks are being consulted, G_5 and G_4 are the same. As a result, we discovered that
$$|Prob[Su_6] - Prob[Su_5]| \leq \frac{O(q_s)}{2^l} \tag{7}$$
* G_7: We use ECGDHP in this case. We believe that $\mathcal{A}$ breaks the chance if he obtains a specific session key via hash-oracle and is the realisation. This is how we change the hash-oracle: On one possibility $\mathcal{A}$ queries $(H_1\|\star\|t_5), (H_1\|H_2^*\|t_5)(ID_U\|ID_{VC}^*\|\star\|H_2^*\|\star, H_3^*), (ID_U\|ID_{VC}^*\|\gamma\|\star\|H_1\|H_3^*\|\star\|t_5, SK_U)$. SL first verifies if $(H_1\|\star\|t_5), (H_1\|H_2^*\|t_5)(ID_U\|ID_{VC}^*\|\star\|H_2^*\|\star, H_3^*), (ID_U\|ID_{VC}^*\|\gamma\|\star\|H_1\|H_3^*\|X\|t_5, SK_U) \in L_E$. The session key is returned if it fails. Otherwise,

the ECGDHP oracle is obtained from SL by inspector $X \stackrel{?}{=} ECDDHP(xg, yg)$. If the query fails, it will be deleted. Otherwise, SL sends $SK \in \{0, 1\}^l$ and $(ID_U \| ID^*_{VC} \| \gamma \| y \| H_1 \| H^*_3 \| X \| t_5$ to L_E. We noticed there are two forms of attacks in G_7: passive and active. To obtain all of the information, $\mathcal{A}$ performs a *Corrupt* query:

- Assaults based on guessing $\mathcal{N}$ password from the dictionary could be taken by $\mathcal{A}$. Whereas $\mathcal{A}$ can use *Send query* q_s with $\frac{Q_s}{\mathcal{N}}$ limits the probability of $\mathcal{A}$ guessing the precise password by loading a session.
- It is employed in passive attacks. The following situations have occurred:
  - ◇ To begin, $\mathcal{A}$ scans the message, then $\mathcal{A}$ inquires about *Execute queries*. Finally, $\mathcal{A}$ asks H-query to complete the task, which breaks ECGDHP. We can look for xyg. With the probability $1/q_h$, from L_E. So, the probability in this way is bounded by $q_h.Advt^{ECDDHP}_{\mathcal{A}}(\kappa + O(q_e)T_M)$.
  - ◇ $\mathcal{A}$, on the other hand, *Send queries* one after the other. In the first type passive attack, $\mathcal{A}$ can search that the upper bound probability for this case is $q_h Advt^{ECDDHP}_{\mathcal{A}}(\kappa + O(q_s)T_M)$

The probability for these passive attack is $q_h Advt^{ECDDHP}_{\mathcal{A}}(\kappa + O(q_e)T_M) + q_h Advt^{ECDDHP}_{\mathcal{A}}(\kappa + O(q_s)T_M) \leq q_h Advt^{ECDDHP}_{\mathcal{A}}(2\kappa + [O(q_s) + O(q_e)]T_M)$, where $\kappa' = (2\tau + [O(q_s) + O(q_e)]T_M)$. Then, we have

$$|Prob[Su_7] - Prob[Su_6]| \leq \frac{q_s}{\mathcal{N}} + q_h Advt^{ECDDHP}_{\mathcal{A}}(\kappa') \tag{8}$$

∗ G_8: Perfect forward security was employed in the previous game. All based *Corrupt* inquiries can be resolved by Adversary. However, according to the *sfs − fresh* technique, *Corrupt* queries should be queried after the *Test* query. As a result, $\mathcal{A}$ can only evade archaic enquiries and documents. Here, we can achieve $(ID_U \| IDVC \| \gamma \| y \| H_1 \| H_3 \| X \| t_5), SK) \in L_E$. The probability of getting xg and yg in the same session is $1/(q_s + q_e)^2$ and we have

$$|Prob[Su_8] - Prob[Su_7]| \leq Q_h(q_s + q_e)^2 Advt^{ECDDHP}_{\mathcal{A}}(\kappa') \tag{9}$$

In the total of the above games, $\mathcal{A}$ has no extra advantage in guessing the session key and $Prob[Su_8] = \frac{1}{2}$. As a result, the theorem is established.

C. INFORMAL SECURITY ANALYSIS

In this session, we'll talk about PSEBVC's informal security investigation. PSEBVC verifies the following security threats, characteristics, and attributes:

- **Supports anonymity property:** In PSEBVC, CC computes $ID_{U1} = ID_U \oplus h(H^*_1 \| H_2)$ and sends to VC and VC computes $ID^*_U = ID_{U1} \oplus h(H^*_1 \| H^*_2)$ of U and do it. Further, VC computes $ID_{VC1} = ID_{VC} \oplus h(H^*_1 \| H^*_2 \| t_5)$ and sends to U and U computes anonymous identity $ID^*_{VC} = ID_{VC1} \oplus h(H_1 \| H^*_2 \| t_5)$ of VC and uses it. The anonymity characteristic is thus supported by PSEBVC.
- **Mutual authentication:** In PSEBVC, U computes $H_1 = h(ID_U \| \alpha_1 \| x)$. Further, CC verifies $H^*_1 \stackrel{?}{=} h(ID_U \| \alpha_1 \| x)$ and computes $H_2 = h(H^*_1 \| ID_{VC} \| t_3)$. Furthermore, VC verifies $H^*_2 \stackrel{?}{=} h(H^*_1 \| ID_{VC} \| t_3)$ and computes $H_3 = h(ID^*_U \| ID_{VC} \| z \| H^*_2 \| y)$. Finally, U verifies $H^*_3 \stackrel{?}{=} h(ID_U \| ID^*_{VC} \| z \| H^*_2 \| y)$. Thus, U and VC verify each other's authenticity. As a result, PSEBVC satisfies the property of mutual authentication.

TABLE 5. Simulation of queries.

Simulation of queries
If a collection of values (s, r) exists in L_H, r is returned as the response for a hash query. Otherwise, SL returns r and sets (s, r) in L_H with a random value $r \in \{0, 1\}^l$. Such steps must be accomplished in the database (l, s, r) in order to run a hash query.
For a $Send\ (U, INIT)$ query, SL executes the following steps: U login with ID'_U and pw'_U, imprints B'_U and gets $\sigma'_U = Rep(B'_U, \tau'_U)$ Computes $\alpha'_2 = \alpha'_1 \oplus \sigma'_U, \alpha'_3 = h(ID'_U \| pw'_U \| \alpha'_2)$ Verifies $\alpha'_3 \stackrel{?}{=} \alpha_3$ Generates $x \in Z^\star_q$ Computes $H_1 = h(ID_U \| \alpha_1 \| a)$ Encrypts $E_1 = E_{h(ID_U \| \alpha_1 \| t_1)}(H_1, x)$ Returns $M_1 = \{E_1, t_1\}$
For a Send (U^i, CC^t, M_1) query, SL performs the following actions: Verifies $t_2 - t_1 \leq \triangle T$ Decrypts $(H_1, x) = D_{h(ID_U \| \alpha_1 \| t_1)}(E_1)$ Verifies $H^*_1 \stackrel{?}{=} h(ID_U \| \alpha_1 \| x)$ Generates number $z \in Z^\star_q$ Computes $H_2 = h(H^*_1 \| ID_{VC} \| t_3), ID_{U1} = ID_U \oplus h(H^*_1 \| H_2)$ Encrypts $E_2 = E_{h(ID_{VC} \| P \| t_3)}(H^*_1, z, ID_{U1}, x, H_2, \gamma, t_3)$ Returns $M_2 = \{E_2, t_3\}$
For a Send (CC^t, VC^j, M_2) query, the following stages are computed by SL: Verifies $t_4 - t_3 \leq \triangle T$ Decrypts $(H^*_1, z, ID_{U1}, x, H_2, \gamma, t_3) = D_{h(ID_{VC} \| P \| t_3})(E_2)$ Verifies $H^*_2 \stackrel{?}{=} h(H^*_1 \| ID_{VC} \| t_3)$ Computes $ID^*_U = ID_{U1} \oplus h(H^*_1 \| H^*_2)$ Generates $y \in Z^\star_q$ Computes $H_3 = h(ID^*_U \| ID_{VC} \| z \| H^*_2 \| y)$ Calculates $SK_{VC} = h(ID^*_U \| ID_{VC} \| \gamma \| z \| H^*_1 \| H_3 \| xyg \| t_5)$ Computes $ID_{VC1} = ID_{VC} \oplus h(H^*_1 \| H^*_2 \| t_5)$ $\kappa')$ Encrypts $E_3 = E_{h(H^*_1 \| x \| t_5)}(H_3, z, y, H^*_2, ID_{VC1}, t_5)$ Returns $M_3 = \{E_3, t_5\}$
For a Send (VC^t, CC^t, M_3) query, SL takes the following steps: Verifies $t_6 - t_5 \leq \triangle T$ Returns $M_4 = \{E_3, t_5, t_7\}$
For a Send (CC^t, U^i, M_4) query, SL takes the following steps: Verifies $t_8 - t_7 \leq \triangle T$ Decrypts $(H_3, z, y, H^*_2, ID_{VC1}, t_5) = D_{h(H_1 \| x \| t_5)}(E_3)$ Computes $ID^*_{VC} = ID_{VC1} \oplus h(H_1 \| H^*_2 \| t_5)$ Verifies $H^*_3 \stackrel{?}{=} h(ID_U \| ID^*_{VC} \| z \| H^*_2 \| y)$ Computes session key $SK_U = h(ID_U \| ID^*_{VC} \| \gamma \| z \| H_1 \| H^*_3 \| xyg \| t_5)$
For an *Execute* (U^i, CC^t, VC^j) query, each *Send* query is completed in order. Massage (M_1, M_2, M_3, M_4) is returned
For a $Reveal\ (I^K)$ query, return SK_U or SK_VC if the occurrence I^K was created and a secure session key was generated. Otherwise, a $\perp$ will be returned.
All of I^K's communication messages are output for a $Corrupt\ (I^K)$ query. If I^K is not $sfs - fresh$ for a $Test(I^K)$ query, $\perp$ is returned. If not, a coin is tossed (ρ). If $\kappa = 0$, the outcome is a random length l value. The right session key is obtained if $\kappa = 1$.

- **Off-line password guessing attack:** Assume that $\mathcal{A}$ guesses $pw_{\mathcal{A}} = pw_U$ of U but password of U's is pw_U and U computes $(\sigma_U, \tau_U) = Gen(B_U)$, $S_U = h(pw_U \| \sigma_U) \oplus r_U$. B_U is U's biometric, σ_U is the derived string, and r_U is U's random selected by U. If possible, $pw_{\mathcal{A}} = pw_U$. Further, $\mathcal{A}$ try to communicate in the login and authentication phase. Where, he/she verifies $\alpha'_3 \stackrel{?}{=} \alpha_3$ which does not verify. Where, $\alpha'_2 = \alpha'_1 \oplus \sigma'_U$, $\alpha'_3 = h(ID_U \| pw_U \| \alpha'_2)$. PSEBVC thus defends against this attack.
- **Replay attack:** To combat replay attacks, the PSEBVC uses a time-stamp and random numbers. U, CC, and VC take the following stages in PSEBVC:
  - In PSEBVC, CC verified $t_2 - t_1 \leq \Delta t$, $t_6 - t_5 \leq \Delta t$, where Δt denotes the longest time delay, and in the login and authentication process, CC generates the value $z \in Z_q^\star$.
  - VC verifies $t_4 - t_3 \leq \Delta t$. VC generates random value $z \in Z_q^\star$ and uses in PSEBVC.
  - U verifies $t_8 - t_7 \leq \Delta t$. U, generates $x \in Z_q^\star$ and uses in the login and authentication phase.

  Even if $\mathcal{A}$ replays the message intercepted over the insecure channel, he or she is unable to locate the secure data. PSEBVC is thus impervious to replay attacks.
- **Provision of key agreement:** In PSEBVC, U and VC agree on the session key $SK = SK_{VC} = SK_U$ after authenticating one another using $xyg = yxg$. The session key is updated using the random values x and y. According to ECCDHP, calculating xyg or yxg is challenging.
- **Strong forward security:** In PSEBVC, CC verifies $t_2 - t_1 \leq \Delta t$, $t_6 - t_5 \leq \Delta t$ and $H_1^* \stackrel{?}{=} h(ID_U \| \alpha_1 \| x)$. Further, VC verifies time-stamps condition $t_4 - t_3 \leq \Delta t$, $H_2^* \stackrel{?}{=} h(H_1^* \| ID_{VC} \| t_3)$ and computes session key $SK_{VC} = h(ID_U^* \| ID_{VC} \| \gamma \| z \| H_1^* \| H_3 \| xyg \| t_5)$. In last, U verifies time-stamps condition $t_8 - t_7 \leq \Delta t$, verifies $H_3^* \stackrel{?}{=} h(ID_U \| ID_{VC}^* \| z \| H_2^* \| y)$ and sets session key $SK_U = h(ID_U \| ID_{VC}^* \| \gamma \| z \| H_1 \| H_3^* \| xyg \| t_5)$. Hence, PSEBVC manages session key $SK = SK_{VC} = SK_U$.
- **Man-in-the-middle attack:** The login and authentication phase includes time-stamp and hash criteria at every stage. It is difficult to check the hash requirements under the definition of a collision-free one-way hash function since it is secured, but if an adversary can enter one of these phases by looking at the time stamps, he/she must do so next. The login and authentication process will therefore fail for the attacker. The suggested technique thus protects against this attack.
- **User impersonation attack:** Obtaining password pw_U and identity ID_U, and creating $M_{R1} = \{ID_U, S_U, t_{R1}\}$ are the most typical approaches for $\mathcal{A}$ to mimic a legitimate user. According to a password guessing assault conducted off-line. For $\mathcal{A}$, they are impossible. The parameters $(\sigma_U, \tau_U) = Gen(B_U)$, $S_U = h(pw_U \| \sigma_U) \oplus r_U$. As a result, PSEBVC may be immune to impersonation attacks.

TABLE 6. Various cryptographic operations have different time costs (ms).

Operation	Running time	U cost	VC cost	CC cost
T_{ECM}	Elliptic curve multiplication	0.537	0.050	0.041
T_{ECA}	Elliptic curve point addition	0.601	0.051	.0045
T_E	Exponentiation	200.670	2.097	1.695
T_M	Multiplication	0.731	0.007	0.006
T_S	Symmetric encryption/decryption	13.434	1.587	0.978
T_{BP}	Bilinear pairing	361.282	5.562	4.11
T_H	Hash function	11.260	0.728	0.483

- **De-synchronization attack:** The VC, CC, and U have no parameters to change, but U checks the login and verification criteria anytime it wishes to update the password. Finally, there is no need for U or VC synchronization with PSEBVC. Because of this, PSEBVC cannot be used in a de-synchronization attack.
- **Insider attack:** In user's registation phase, U submits $(\sigma_U, \tau_U) = Gen(B_U)$, $S_U = h(pw_U \| \sigma_U) \oplus r_U$ Thus, the S_U and B_U cannot be obtained by the administrator of the. PSEBVC is thus resistant to insider attacks.
- **Message authentication:** In PSEBVC, CC gets $M_1 = \{E_1, t_1\}$ and verifies $t_2 - t_1 \leq \Delta t$, $t_6 - t_5 \leq \Delta t$ and $H_1^* \stackrel{?}{=} h(ID_U \| \alpha_1 \| x)$. VC receives the message $M_2 = \{E_2, t_3\}$ and verifies $t_4 - t_3 \leq \Delta t$ and $H_2^* \stackrel{?}{=} h(H_1^* \| ID_{VC} \| t_3)$. U receives the message $M_3 = \{E_3, t_5\}$ and verifies $t_8 - t_7 \leq \Delta t$ and $H_3^* \stackrel{?}{=} h(ID_U \| ID_{VC}^* \| z \| H_2^* \| y)$. The communication message will not be recognised if any of the checks fail. PSEBVC verifies message authenticity between U, CC, and VC.
- **Parallel section attack:** In a parallel approach, the attacker $\mathcal{A}$ builds a new request by reusing a prior message in the public channel, the session key is then computed by impersonating the proper user U. Before building the right approach request or session key in PSEBVC, $\mathcal{A}$ must first comprehend the message's parameters. Our analysis revealed that $\mathcal{A}$ is unable to access the session key. PSEBVC can therefore tolerate parallel attack.

V. PERFORMANCE ANALYSIS

The performance analysis of the PSEBVC is discussed in this section:

A. COMPARISON OF THE SECURITY AND UTILITY FEATURES

Table 7 compares the PSEBVC framework's capabilities, characteristics, and security against those of other frameworks. We compared the PSEBVC framework's functionality, characteristics, and security to those of other frameworks like Jang *et al.* protocol [25], He *et al.* protocol [26], Odelu *et al.* [39], Mo *et al.* [41] Irshad *et al.* [51], Jia *et al.* [52] and Son *et al.* [33]. in Table 7. It's worth noting that Jang et al's protocol fails in the face of MA, IM, DS, IA, ME and PB. He *et al.*'s protocol fails against DS and UA. Odelu *et al.*'s protocol fails against IM and ME. Mo *et al.*'s protocol fails

TABLE 7. The costs of communication and computation are compared.

Protocol	U C-cost(ms)	VC C-cost(ms)	CC C-cost(ms)	Total C-cost(ms)	CM cost	OF	SA	MA	RA	SF	IM	DS	IA	SK	ME	UA	PB
He et al. [26]	$3T_{ECM}+1T_{ECA}+3T_E+5T_H \approx 660.522$	$2T_{BP}+1T_{ECM}+3T_E+5T_H \approx 21.105$	NA	≈ 681.2267	3456 bits	√	√	√	√	√	√	×	√	√	√	×	√
Odelu et al. [39]	$3T_{ECM}+1T_{ECA}+3T_E+5T_H \approx 660.522$	$2T_{BP}+1T_{ECM}+3T_E+5T_H \approx 21.105$	NA	≈ 681.2267	3432 bit	√	√	√	√	√	×	√	√	√	×	√	√
Jiang et al. [25]	$3T_{ECM}+4T_S+9T_H \approx 156.687$	$T_S+3T_H \approx 3.771$	$3T_{ECM}+5T_S+4T_H \approx 6.944$	≈ 167.403	3592 bits	√	√	×	√	√	√	×	×	√	×	√	×
Mo et al. [41]	$3T_M+1T_{ECA}+3T_E+7T_H \approx 81.632$	$3T_M+1T_{ECA}+3T_E+6T_H \approx 70.372$	NA	≈ 152.004	2848 bit	√	√	√	√	√	×	×	×	×	×	×	√
Irshad et al. [51]	$4T_{ECM}+1T_{BP}+3T_{ECA}+8T_H \approx 455.313$	$3T_{ECM}+2T_{BP}+3T_{ECA}+13T_H \approx 20.891$	NA	≈ 476.204	2560 bits	√	√	√	√	√	×	×	√	√	√	×	×
Jia et al. [52]	$4T_{ECM}+1T_E+5T_H \approx 259.118$	$5T_{ECM}+1T_{BP}+3T_{ECA}+5T_H \approx 9.605$	NA	≈ 268.723	2656 bits	√	√	√	√	√	√	×	√	√	√	×	×
Son et al. [33]	$11T_H \approx 123.86$ ms	$4T_{ECM}+1T_{ECA}+12T_H \approx 8.987$	NA	≈ 132.847	3112 bits	√	√	√	√	√	√	×	√	√	√	√	×
PSEBVC	$2T_S+7T_H \approx 105.688$	$2T_S+6T_H \approx 7.542$	$2T_S+5T_H \approx 4.371$	≈ 117.601	864 bits	√	√	√	√	√	√	√	√	√	√	√	√

"Where C-cost: Computation cost, CM-cost: Communication cost, SA: Supports anonymity property, MA: Mutual authentication, IM: Impersonation attack, RA: Replay attack, SK: Session key security, UA: Untraceable attack, ME: Message authentication, PK: Provision of key agreement, OF: Off-line password guessing attack, SF: Strong forward security, DS: De-synchronization attack, IA: Insider attack, and PB: Password and biometric change phase"

against IM, DS, IA, SK, ME and UA. Irshad *et al.*'s protocol fails against IM, DS, UA and PB. Jai *et al.*'s protocol fails against DS, UA and PB. Son *et al.*'s protocol fails against DS and PB. PSEBVC generates all functional features and attributes, including OF, SA, MA, RA, SF, IM, DS, IA, SK, ME, UA, and PB.

B. COMPUTATION COST COMPARISON

On the based of Jing *et al.*'s protocol [25], In PSEBVC, the computing cost of cryptography operations is measured in milliseconds (ms). The details of the calculation cost of various processes are provided in table 6. A smartphone serves as the computation platform for the VCE structural system.

The Huawei Mate 7 comes with Google Android 4.4.2 as the operating system, Hisilicon Kirin 9252.45 GHz processor, and 3 GB of memory. Among the notebook's characteristics are an Intel I7-4460S processor, 16 GB RAM, a 3.1 GHz processor, and MacOS 10.12.4 for VCs. CC's desktop computing was done on a Dell Alienware with a Windows 10 64-bit, Intel I7-6700k 4.0 GHz processor and 32 GB RAM. The JPBC Library is used for the pairing process, whereas the regular Java library is used for the other operations. Concatenation ($\|$) and XOR ($\oplus$) operations had extremely low computing costs. PSEBVC's computation coat was compared to that of related protocols. Jang *et al.* protocol [25], He *et al.* protocol [26], Odelu *et al.* [39], Mo *et al.* [41], Irshad *et al.* [51], Jia *et al.* [52] and Son *et al.* [33]. The following is a breakdown of the cost of computation:

- In Jiang *et al.*'s protocol, the cost of computing is $3T_{ECM}+4T_S+9T_H \approx 156.687$ ms, the protocol of He *et al.* is $3T_{ECM}+1T_{ECA}+3T_E+5T_H \approx 660.522$ ms, Odelu *et al.*'s protocol is $3T_{ECM}+1T_{ECA}+3T_E+5T_H \approx 660.522$ ms, the protocol of Mo *et al.* is $3T_M+1T_{ECA}+3T_E+7T_H \approx 81.632$ ms, Irshad *et al.*'s protocol is $4T_{ECM}+1T_{BP}+3T_{ECA}+8T_H \approx 455.313$ ms, Jia *et al.*'s protocol is $4T_{ECM}+1T_E+5T_H \approx 259.118$ ms, Son *et al.*'s protocol is $11T_H \approx 123.86$ ms and PSEBVC is $2T_S+7T_H \approx 105.688$ ms. The U computation cost expenditure is detailed in the table 7.
- In Jiang et al's protocol, the cost of computing VC is $1T_S+3T_H \approx 3.771$ ms, The protocol of He *et al.* is $2T_{BP}+1T_{ECM}+1T_{ECA}+3T_E+5T_H \approx 21.105$ ms, Odelu *et al.*'s protocol is $2T_{BP}+3T_{ECM}+1T_{ECA}+3T_E+5T_H \approx 21.105$ ms, Mo *et al.*'s protocol is $3T_M+1T_{ECA}+3T_E+6T_H \approx 70.372$ ms, Irshad *et al.*'s protocol is $3T_{ECM}+2T_{BP}+3T_{ECA}+13T_H \approx 20.891$ ms, Jia *et al.*'s protocol is $5T_{ECM}+1T_{BP}+3T_{ECA}+5T_H \approx 9.605$ ms, Son *et al.*'s is $4T_{ECM}+1T_{ECA}+12T_H \approx 8.987$ and PSEBVC is $2T_S+6T_H \approx 7.542$ ms. The detail of VC computation cost spending is displayed in the table 7.
- In Jiang *et al.* protocol, the computing cost of CC is $3T_{ECM}+5T_S+4T_H \approx 6.944$ ms, and PSEBVC is $2T_S+5T_H \approx 4.371$ ms. The table 7 shows the CC computation cost spent in detail.

The overall computing cost of the PSEBVC and related protocols is discussed in the table 7:

- The total computation cost of the PSEBVC is $\approx$117.601 ms
- Jiang *et al.*' protocol has a total computational cost $\approx$167.403 ms, which is $\approx$42.348% more than PSEBVC's total computation cost.
- He *et al.* protocol has a total computing cost of $\approx$681.2267 ms, which is $\approx$479.269% more than PSEBVC's total computation cost.
- Odelu *et al.* procedure has a total computing cost of $\approx$681.2267 ms, which is $\approx$479.269% more than PSEBVC's total computation cost.
- Mo *et al.* protocol has a total computation cost of $\approx$152.004 ms, which is $\approx$29.254% more than PSEBVC's total computation cost.
- Irshad *et al.*'s protocol has a total computing cost of $\approx$476.204 ms, which is $\approx$304.931% more than PSEBVC's total computation cost.
- Jia *et al.*'s protocol has a total calculation cost of $\approx$268.732 ms, which is $\approx$128.504% more than PSEBVC's total computation cost.
- Son *et al.* procedure has a total computing cost of $\approx$132.847 ms, which is $\approx$12.964% more than PSEBVC's total computation cost.

The Fig 3 shows the total computation cost in details.

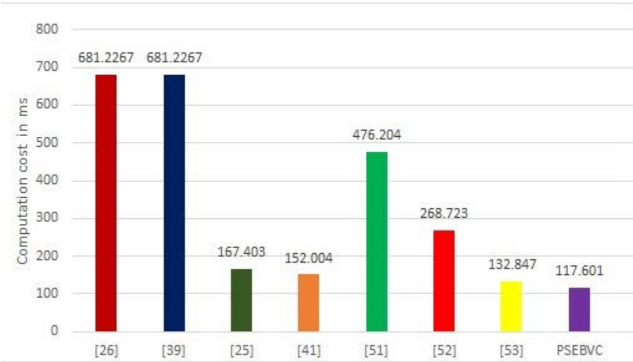


FIGURE 3. Cost of the entire computation in ms.

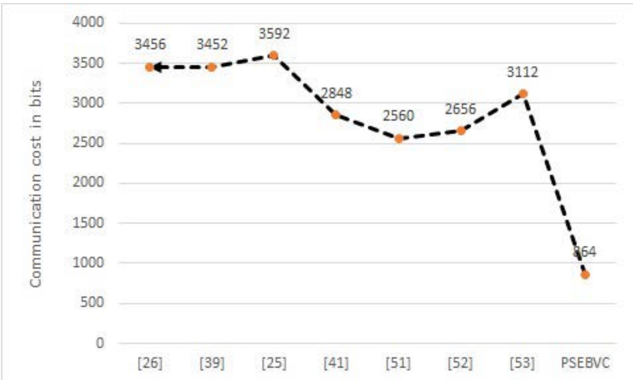


FIGURE 4. Communication cost in bits.

C. COMPARISON OF THE COMMUNICATION COST

The Jang *et al.*'s [25] protocol includes binary lengths of 32, 96, 128, 32, 160, 256, and 1024 bits, respectively, for identity, random number, ticket key, ticket lifetime, hash output, and symmetric key encryption/decryption. The entire computing cost of the PSEBVC and related protocols is discussed as follows from the table 7:

- The PSEBVC's communication cost is 864 bits.
- Jiang et al's protocol has a communication cost of 3592 bits which is ≈315.740% more than PSEBVC's communication cost.
- He *et al.*'s protocol has a communication cost of 3456 bits which is ≈300.000% more than PSEBVC's communication cost.
- Odelu *et al.*'s protocol has a communication cost of 3432 bits which is ≈297.222% more than PSEBVC's communication cost.
- The communication cost of Mo *et al.*'s protocol is 2848 bits which is ≈229.629% more than PSEBVC's communication cost.
- Irshad et al's protocol has a communication cost of 2560 bits which is ≈196.296% more than PSEBVC's communication cost.
- Jia *et al.*'s approach has a communication cost of 2656 bits which is ≈207.407% more than PSEBVC's communication cost.
- Son *et al.*'s approach has a communication cost of 3112 bits which is ≈260.185% more than PSEBVC's communication cost.

The communication cost of the proposed protocol and related protocols is shown in Fig 4.

VI. CONCLUSION

The VCE-assisted structure is a critical method to network system building that is gaining traction. V2V or V2I resource use on roadside units is facilitated. The security and privacy of the VCE system are causing growing concern among users. In this paper, we present a new biometric authentication system for VCE that is aided by ECC. The PSEBVC is resistant to a variety of assaults and meets all of the necessary security requirements. The Scyther tool has been used in the study to demonstrate the protocol's security. Additionally, we have demonstrated the suggested framework and provided a random oracle-based security model. We have also demonstrated the protocol's affordability in terms of communication and processing. As a result, our recommended methodology might be more appropriate for VCE and useful for practical purposes.

ACKNOWLEDGMENT

This work was supported by the EIAS Data Science Lab, College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia.

REFERENCES

[1] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, "Connected vehicles: Solutions and challenges," *IEEE Internet Things J.*, vol. 1, no. 4, pp. 289–299, Aug. 2014.
[2] K. Zheng, L. Hou, H. Meng, Q. Zheng, N. Lu, and L. Lei, "Soft-defined heterogeneous vehicular network: Architecture and challenges," *IEEE Netw.*, vol. 30, no. 4, pp. 72–80, Jul. 2016.
[3] R. Yu, Y. Zhang, S. Gjessing, W. Xia, and K. Yang, "Toward cloud-based vehicular networks with efficient resource management," *IEEE Netw.*, vol. 27, no. 5, pp. 48–55, Sep. 2013.
[4] M. Whaiduzzaman, M. Sookhak, A. Gani, and R. Buyya, "A survey on vehicular cloud computing," *J. Netw. Comput. Appl.*, vol. 40, pp. 325–344, Apr. 2014.
[5] J. Ni, A. Zhang, X. Lin, and X. S. Shen, "Security, privacy, and fairness in fog-based vehicular crowdsensing," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 146–152, 2017.
[6] K. Zhang, J. Ni, K. Yang, X. Liang, J. Ren, and X. S. Shen, "Security and privacy in smart city applications: Challenges and solutions," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 122–129, Jan. 2017.
[7] H. Abid, L. T. T. Phuong, J. Wang, S. Lee, and S. Qaisar, "Cloud: Vehicular cyber-physical systems and cloud computing," in *Proc. 4th Int. Symp. Appl. Sci. Biomed. Commun. Technol.*, 2011, p. 165.
[8] G. Yan, D. Wen, S. Olariu, and M. C. Weigle, "Security challenges in vehicular cloud computing," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 284–294, Mar. 2013.
[9] *Advanced Authentication 6.3 Administration Guide*. Accessed: Dec. 2019. [Online]. Available: https://www.netiq.com/documentation/advanced-authentication-63/server-administrator-guide/data/smartphone.html
[10] C. Zhang, R. Lu, X. Lin, P.-H. Ho, and X. Shen, "An efficient identity-based batch verification scheme for vehicular sensor networks," in *Proc. 27th Conf. Comput. Commun. (INFOCOM)*, Apr. 2008, pp. 246–250.
[11] R. Lu, X. Lin, H. Zhu, P.-H. Ho, and X. Shen, "ECPP: Efficient conditional privacy preservation protocol for secure vehicular communications," in *Proc. 27th Conf. Comput. Commun. (INFOCOM)*, Apr. 2008, pp. 1229–1237.
[12] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *J. Comput. Secur.*, vol. 15, no. 1, pp. 39–68, 2007.
[13] Y. Jiang, M. Shi, X. Shen, and C. Lin, "BAT: A robust signature scheme for vehicular networks using binary authentication tree," *IEEE Trans. Wireless Commun.*, vol. 8, no. 4, pp. 1974–1983, Apr. 2009.

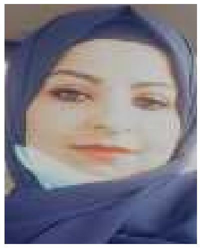
[14] K.-A. Shim, "Reconstruction of a secure authentication scheme for vehicular ad hoc networks using a binary authentication tree," *IEEE Trans. Wireless Commun.*, vol. 12, no. 11, pp. 5386–5393, Nov. 2013.
[15] N.-W. Lo and J.-L. Tsai, "An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 5, pp. 1319–1328, May 2016.
[16] J.-S. Li and K.-H. Liu, "A lightweight identity authentication protocol for vehicular networks," *Telecommun. Syst.*, vol. 53, no. 4, pp. 425–438, Aug. 2013.
[17] C.-C. Lee and Y.-M. Lai, "Toward a secure batch verification with group testing for VANET," *Wireless Netw.*, vol. 19, no. 6, pp. 1441–1449, Aug. 2013.
[18] D. He, S. Zeadally, B. Xu, and X. Huang, "An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 12, pp. 2681–2691, Dec. 2015.
[19] C. P. Schnorr, "Efficient signature generation by smart cards," *J. Cryptol.*, vol. 4, no. 3, pp. 161–174, Jan. 1991.
[20] T. Oulhaci, M. Omar, F. Harzine, and I. Harfi, "Secure and distributed certification system architecture for safety message authentication in VANET," *Telecommun. Syst.*, vol. 64, no. 4, pp. 679–694, Apr. 2017.
[21] L. Zhang, Q. Wu, J. Domingo-Ferrer, B. Qin, and C. Hu, "Distributed aggregate privacy-preserving authentication in VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 3, pp. 516–526, Mar. 2017.
[22] L. Zhang, "OTIBAAGKA: A new security tool for cryptographic mix-zone establishment in vehicular ad hoc networks," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 12, pp. 2998–3010, Dec. 2017.
[23] M. R. Asaar, M. Salmasizadeh, W. Susilo, and A. Majidi, "A secure and efficient authentication technique for vehicular ad-hoc networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5409–5423, Jun. 2018.
[24] J. Li, K.-K. R. Choo, W. Zhang, S. Kumari, J. J. Rodrigues, M. K. Khan, and D. Hogrefe, "EPA-CPPA: An efficient, provably-secure and anonymous conditional privacy-preserving authentication scheme for vehicular ad hoc networks," *Veh. Commun.*, vol. 13, pp. 104–113, Jul. 2018.
[25] Q. Jiang, J. Ni, J. Ma, L. Yang, and X. Shen, "Integrated authentication and key agreement framework for vehicular cloud computing," *IEEE Netw.*, vol. 32, no. 3, pp. 28–35, May 2018.
[26] D. He, N. Kumar, M. K. Khan, L. Wang, and J. Shen, "Efficient privacy-aware authentication scheme for mobile cloud computing services," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1621–1631, Jun. 2018.
[27] J. Zhang, J. Cui, H. Zhong, Z. Chen, and L. Liu, "PA-CRT: Chinese remainder theorem based conditional privacy-preserving authentication scheme in vehicular ad-hoc networks," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 2, pp. 722–735, Mar. 2021.
[28] J. Cui, X. Zhang, H. Zhong, Z. Ying, and L. Liu, "RSMA: Reputation system-based lightweight message authentication framework and protocol for 5G-enabled vehicular networks," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6417–6428, Aug. 2019.
[29] A. Irshad, M. Usman, S. A. Chaudhry, H. Naqvi, and M. Shafiq, "A provably secure and efficient authenticated key agreement scheme for energy internet based vehicle-to-grid technology framework," *IEEE Trans. Ind. Appl.*, vol. 56, no. 4, pp. 4425–4435. Jan. 2020.
[30] P. Gope and B. Sikdar, "An efficient privacy-preserving authentication scheme for energy internet-based vehicle-to-grid communication," *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6607–6618, Nov. 2019.
[31] J. Zhang, H. Zhong, J. Cui, M. Tian, Y. Xu, and L. Liu, "Edge computing-based privacy-preserving authentication framework and protocol for 5G-enabled vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7940–7954, Jul. 2020.
[32] S. A. Chaudhry, A. Irshad, M. A. Khan, S. A. Khan, S. Nosheen, A. A. AlZubi, and Y. B. Zikria, "A lightweight authentication scheme for 6G-IoT enabled maritime transport system," *IEEE Trans. Intell. Transp. Syst.*, early access, Dec. 22, 2021, doi: 10.1109/TITS.2021.3134643.
[33] S. Son, J. Lee, Y. Park, Y. Park, and A. K. Das, "Design of blockchain-based lightweight V2I handover authentication protocol for VANET," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 3, pp. 1346–1358, May 2022.
[34] M. Tanveer, A. H. Zahid, M. Ahmad, A. Baz, and H. Alhakami, "LAKE-IoD: Lightweight authenticated key exchange protocol for the Internet of Drone environment," *IEEE Access*, vol. 8, pp. 155645–155659, 2020.
[35] A. A. Khan, V. Kumar, M. Ahmad, B. B. Gupta, M. Ahmad, and A. A. A. El-Latif, "A secure and efficient key agreement framework for critical energy infrastructure using mobile device," *Telecommun. Syst.*, vol. 78, no. 4, pp. 539–557, Dec. 2021.
[36] M. Tanveer, A. U. Khan, H. Shah, A. Alkhayyat, S. A. Chaudhry, and M. Ahmad, "ARAP-SG: Anonymous and reliable authentication protocol for smart grids," *IEEE Access*, vol. 9, pp. 143366–143377, 2021.
[37] M. Tanveer, A. U. Khan, T. Nguyen, M. Ahmad, and A. Abdei-Latif, "Towards a secure and computational framework for Internet of Drones enabled aerial computing," *IEEE Trans. Netw. Sci. Eng.*, early access, Feb. 15, 2022, doi: 10.1109/TNSE.2022.3151843.
[38] V. Kumar, M. S. Mahmoud, A. Alkhayyat, J. Srinivas, M. Ahmad, and A. Kumari, "RAPCHI: Robust authentication protocol for IoMT-based cloud-healthcare infrastructure," *J. Supercomput.*, vol. 2022, pp. 1–30, May 2022.
[39] V. Odelu, A. K. Das, S. Kumari, X. Huang, and M. Wazid, "Provably secure authenticated key agreement scheme for distributed mobile cloud computing services," *Future Gener. Comput. Syst.*, vol. 68, pp. 74–88, Mar. 2017.
[40] D. He and D. Wang, "Robust biometrics-based authentication scheme for multiserver environment," *IEEE Syst. J.*, vol. 9, no. 3, pp. 816–823, Sep. 2015.
[41] J. Mo, Z. Hu, H. Chen, and W. Shen, "An efficient and provably secure anonymous user authentication and key agreement for mobile cloud computing," *Wireless Commun. Mobile Comput.*, vol. 2019, pp. 1–12, Feb. 2019.
[42] V. Kumar, S. Jangirala, and M. Ahmad, "An efficient mutual authentication framework for healthcare system in cloud computing," *J. Med. Syst.*, vol. 42, no. 8, pp. 1–25, Aug. 2018.
[43] V. Kumar, M. Ahmad, and A. Kumari, "A secure elliptic curve cryptography based mutual authentication protocol for cloud-assisted TMIS," *Telematics Informat.*, vol. 38, pp. 100–117, May 2019, doi: 10.1016/j.tele.2018.09.001.
[44] A. A. Khan, V. Kumar, and M. Ahmad, "An elliptic curve cryptography based mutual authentication scheme for smart grid communications using biometric approach," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 3, pp. 698–705, Mar. 2022.
[45] H. Zhang, Q. Zhang, and X. Du, "Toward vehicle-assisted cloud computing for smartphones," *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, pp. 5610–5618, Dec. 2015.
[46] C. Celes, A. Boukerche, and A. A. F. Loureiro, "Mobility trace analysis for intelligent vehicular networks: Methods, models, and applications," *ACM Comput. Surv.*, vol. 54, no. 3, pp. 1–38, Apr. 2021.
[47] C. J. Cremers, "The scyther tool: Verification, falsification, and analysis of security protocols," in *Proc. Int. Conf. Comput. Aided Verification*. Princeton, NJ, USA: Springer, 2008, pp. 414–418.
[48] M. Abdalla, M. Izabachene, and D. Pointcheval, "Anonymous and transparent gateway-based password-authenticated key exchange," in *Proc. Int. Conf. Cryptol. Netw. Secur.* Hong Kong: Springer, 2008, pp. 133–148.
[49] Z. Zhu, "An efficient authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 36, no. 6, pp. 3833–3838, Dec. 2012.
[50] F. Wu, L. Xu, S. Kumari, and X. Li, "A new and secure authentication scheme for wireless sensor networks with formal proof," *Peer-to-Peer Netw. Appl.*, vol. 10, no. 1, pp. 16–30, Jan. 2017.
[51] A. Irshad, S. A. Chaudhry, M. Shafiq, M. Usman, M. Asif, and A. Ghani, "A provable and secure mobile user authentication scheme for mobile cloud computing services," *Int. J. Commun. Syst.*, vol. 32, no. 14, p. e3980, Sep. 2019, doi: 10.1002/dac.3980.
[52] X. Jia, D. He, N. Kumar, and K.-K.-R. Choo, "A provably secure and efficient identity-based anonymous authentication scheme for mobile edge computing," *IEEE Syst. J.*, vol. 14, no. 1, pp. 560–571, Mar. 2020, doi: 10.1109/JSYST.2019.2896064.

VINOD KUMAR received the Master of Philosophy degree in mathematics from Chaudhary Charan Singh University, Meerut, India, the Master of Technology degree in computer science and data processing from IIT Kharagpur, India, and the Ph.D. degree in elliptic curve cryptography (ECC)-based authentication protocols in cloud computing from the Department of Applied Sciences and Humanities, Jamia Millia Islamia, New Delhi, India. He has qualified CSIR National Eligibility Test (NET) in mathematical sciences, in 2011. In 2011, he also qualified Graduate Aptitude Test in Engineering (GATE) in mathematics. He has over more than eight years of experience in teaching, research, and industry in the field of mathematics, information security and related field. He is currently working as an Assistant Professor with the Department of Mathematics, PGDAV College, University of Delhi, New Delhi. He has supervised five M.Tech. research scholar in the fields of security and optimization. His research interests include remote user authentication protocols, signature protocols, information and network security, cloud computing, cryptographic security protocols, vehicular networking, supply chain management, smart grid security, machine learning, bitcoin, cryptocurrency, number theory, optimization techniques, and applied mathematics. He is also a Lifetime Member of Operational Research Society of India (ORSI), India & MathTech Thinking Foundation (MTTF), India. He has received ''Recognition/Reviewer Certificate Award'' from many reputed journals. He has presented 24 research papers/talk in conferences/workshops. He has authored or coauthored of 35 research papers in reputed international journals and conferences, such as IEEE/Elsevier/Springer/Wiley/Taylor & Francis. Also, he has coauthored a book titled *Elementary Real Analysis*. He has participated in dozen reputed FDP/workshops. He is also serving as a reviewer of dozens of reputed journals, including SCI Indexed of IEEE, Elsevier, Springer, Wiley, and Taylor & Francis. He has been associated with many conferences as a TPC member and the session chair.

AMMAR MOHAMMED ALI AL-TAMEEMI received the B.S. degree in computer science from the University of Technology Baghdad, Baghdad, Iraq, the M.S. degree in computer science (computer programing) from Harbin engineering University, China, in 2012, and the Ph.D. degree in information security from the University of Technology Baghdad, in 2021. His research interests include cryptography, chaos theory, cloud computing, biometric techniques, image processing, and pattern recognition applications.

ADESH KUMARI received the master's degree in mathematics from MDU Rohtak, India, and the Ph.D. degree from the Department of Mathematics, Jamia Millia Islamia, New Delhi, India. She has presented eight research papers/talk in conferences/workshops. She has authored or coauthored of 15 research papers in reputed international journals and conferences, such as IEEE/Elsevier/Springer/Wiley/Taylor & Francis. Also, she has coauthored a book titled *Elementary Real Analysis*. She has participated in dozen reputed FDP/workshops. Her research interests include remote user authentication protocols, smart card security, vehicular networking, information security, and cloud computing.

MUSHEER AHMAD received the B.Tech. and M.Tech. degrees from the Department of Computer Engineering, Aligarh Muslim University, India, in 2004 and 2008, respectively, and the Ph.D. degree in chaos-based cryptography from the Department of Computer Engineering, Jamia Millia Islamia, New Delhi, India. From 2007 to 2010, he has worked with the Department of Computer Engineering, Aligarh Muslim University. Since 2011, he has been working as an Assistant Professor with the Department of Computer Engineering, Jamia Millia Islamia. He has published over 90 research papers in international reputed refereed journals and conference proceedings of the IEEE/Springer/Elsevier. He has more than 2100 citations of his research works with an H-index of 29, i-10 index of 60, and cumulative impact factor of more than 170. Recently, he is listed among World's Top 2% Scientists in a study conducted by Elsevier and Stanford University and report published by Elsevier. His research interests include multimedia security, chaos-based cryptography, cryptanalysis, machine learning for security, image processing, and optimization techniques. He has served as a reviewer and a technical program committee member of many international conferences. He has also served as a Referee of some renowned journals, such as*Information Sciences*, *Signal Processing*, *Journal of Information Security and Applications*, IEEE Journal on Selected Areas in Communications (JSAC), IEEE Transactions on Cybernetics (TCYB), IEEE Transactions on Circuits and Systems for Video Technology (TCSVT), IEEE Transactions on Industrial Informatics (TII), IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), IEEE Transactions on Neural Networks and Learning Systems (TNNLS), IEEE Transactions on Intelligent Transportation Systems (TITS), IEEE Transactions on Network Science and Engineering (TNSE), IEEE Transactions on NanoBioscience (TNB), IEEE Transactions on Circuits and Systems—I: Regular Papers (TCAS-1), IEEE Transactions on Big Data (TBD), IEEE Transactions on Reliability (TR), IEEE Internet of Things Journal (IOTJ), IEEE Multimedia, IEEE Access, *Expert Systems with Applications*, *Wireless Personal Communications*, *Neural Computing and Applications*, *International Journal of Bifurcation and Chaos*, *Chaos Solitons & Fractals*, *Physica A*, *Signal Processing: Image Communication*, *Neurocomputing*, *IET Information Security*, *IET Image Processing*, *Security and Communication Networks*, *Optik*, *Optics and Laser Technology*, *Complexity*, *Computers in Biology and Medicine, Computational and Applied Mathematics*, and *Concurrency and Computation*.



MAYADAH WAHEED FALAH received the bachelor's degree in civil engineering and the M.Sc. degree in civil engineering (structural engineering) from the Department of Civil Engineering, Engineering College, University of Babylon, Babil, Iraq, in 2009 and 2013, respectively. She is currently pursuing the Ph.D. degree. She has a lot of research published in the field of civil engineering. In 2013, she joined the Building and Construction Engineering Technology Department, Al-Mustaqbal University College, as an Academic Staff Member. She is currently a Lecturer and a Researcher in the field of civil engineering.

AHMED A. ABD EL-LATIF received the B.Sc. degree (Hons.) in mathematics and computer science and the M.Sc. degree in computer science from Menoufia University, Egypt, in 2005 and 2010, respectively, and the Ph.D. degree in computer science and technology from the Harbin Institute of Technology (HIT), Harbin, China, in 2013. He is currently an Associate Professor in computer science at Menoufia University, Egypt, and the School of Information Technology and Computer Science, Nile University, Egypt. He is the author and coauthor of more than 130 papers, including refereed IEEE/ACM/Springer/Elsevier journals, conference papers, and book chapters. He received many awards, the State Encouragement Award in Engineering Sciences, in 2016, Arab Republic of Egypt; the Best Ph.D. Student Award from the Harbin Institute of Technology, in 2013; and Young Scientific Award, Menoufia University, Egypt, in 2014. He is a fellow of Academy of Scientific Research and Technology, Egypt. His research interests include multimedia content encryption, secure wireless communication, the IoT, applied cryptanalysis, perceptual cryptography, secret media sharing, information hiding, biometrics, forensic analysis in digital images, and quantum information processing. He has many collaborative scientific activities with international teams in different research projects. Furthermore, he has been reviewing papers for more than 115 international journals, including *IEEE Communications Magazine*, IEEE Internet of Things Journal, *Information Sciences*, IEEE Transactions on Network and Service Management, IEEE Transactions on Services Computing, *Scientific Reports* (Nature), *Journal of Network and Computer Applications*, *Signal Processing*, *Cryptologia*, *Journal of Network and Systems Management*, *Journal of Visual Communication and Image Representation*, *Neurocomputing*, and *Future Generation Computer Systems*. He is an Associate Editor of *Journal of Cyber Security and Mobility*.