


# A Novel Pairing-Free Lightweight Authentication Protocol for Mobile Cloud Computing Framework

Azeem Irshad , Shehzad Ashraf Chaudhry , Osama Ahmad Alomari, Khalid Yahya ,  
and Neeraj Kumar , *Senior Member, IEEE*

**Abstract**—The mobile cloud computing (MCC) refers to an infrastructure that integrates cloud computing and mobile computing, and it has changed a great deal, the service provisioning of applications, which requires to get the data processed after collection from vast sensor and Internet-of-Things-based network. The ever increasing number of handheld mobile gadgets has exacerbated the need for robust and efficient authenticated key agreements. We could witness a number of MCC-based multiserver authentication schemes lately to foster the secure adaptation of the technology; however, the demonstrated solutions are either insecure or employing too costly bilinear pairing operations for implementation. In view of limitations, as illustrated in previous studies, we propose a novel pairing-free multiserver authentication protocol for MCC environment based on an elliptic curve cryptosystem that is not only efficient, but also free from security loopholes as demonstrated. The performance evaluation section discusses and distinguishes the findings among latest studies. The strength of the contributed scheme is proved theoretically under formal security model.

**Index Terms**—Anonymity, attacks, authentication, crypt-analysis, Internet-of-Things (IoT), mobile cloud computing (MCC).

## I. INTRODUCTION

THE mobile cloud computing (MCC) framework lets the mobile users benefit from rich computational capabilities seamlessly, irrespective of the limited resources of mobile gadgets. The MCC might serve as a potential model for future mobile applications requiring instant feedback and processing from massive data produced of Internet-of-Things (IoT) and wireless sensor network based nodes on the field [1], [2]. The mobile devices are becoming more popular and offering people with convenience. The MCC platform enables mobile users

in accessing Internet during mobility, and in turn lets them surmount the obstacles in the way of achieving efficiency and performance. To avail quality multimedia services on demand, the mobile gadgets may fall short of computing and memory resources. This may be a bottleneck for applications whose dynamic needs scale abruptly. In most desktop applications, these shortcomings can be settled with cloud computing that works on the principle of utility computing. However, the demands of running applications for mobile devices are not well addressed by cloud computing. To bridge the gap, a technology framework integrating mobile computing and cloud computing, as termed MCC [3], has emerged recently that provides cheap, incessant, and on-the-spot services to mobile users.

The basic motivation for MCC is to provide quality services to resource-constrained mobile clients. However, unlike cloud computing, it is difficult to secure MCC environment for its total reliance on insecure wireless technology. Thus, the security challenges, particularly user authentication in MCC, are getting spotlight of research community for few years [4], [5].

The multiserver authentication (MSA) in MCC not only endorse the mobile user's authenticity to various servers using a single password, but also helps unleashing the mobile user of the hassle of performing multiple servers' registrations. To achieve this, some authors presented bilinear pairing-based MSA schemes for MCC, however with high computation cost for costly pairing operations. Some of the schemes also bear high communication cost, which should be less in MCC due to power constraint of low-end gadgets. Hence, a secure and efficient MSA-based scheme is required for MCC framework that relies minimally on pairing operations for mutual authentication between user and server, leading to reduced communication in mutual authentication phase.

### A. Research Contribution

The salient points of contribution in this work are as follows.

- 1) To design an efficient and secure authentication protocol by avoiding computation-intensive pairing operations.
- 2) To design a protocol framework in which servers are not required to store either user-oriented' verifiers or maintaining repositories pertaining to users' identities.
- 3) To design a scalable scheme that supports adding a new user or a server freely in network, without requiring a related alteration in smart cards' (SCs) or servers' repository.

Manuscript received March 2, 2020; revised April 22, 2020; accepted May 27, 2020. Date of publication June 19, 2020; date of current version August 26, 2021. (Corresponding author: Neeraj Kumar.)

Azeem Irshad is with the Department of Computer Science and Software Engineering, International Islamic University, Islamabad 44000, Pakistan (e-mail: irshadazeem2@gmail.com).

Shehzad Ashraf Chaudhry and Osama Ahmad Alomari are with the Department of Computer Engineering, Faculty of Engineering and Architecture, Istanbul Gelisim University, Istanbul 34310, Turkey (e-mail: sashraf@gelisim.edu.tr; oalomari@gelisim.edu.tr).

Khalid Yahya is with the Mechatronics Engineering Department, Faculty of Engineering and Architecture, Istanbul Gelisim University, Istanbul 34310, Turkey (e-mail: koyahya@gelisim.edu.tr).

Neeraj Kumar is with the Department of Computer Science and Engineering, Thapar University, Patiala 147004, India, with the Department of Computer Science and Information Engineering, Asia University, Taiwan, and also with the King Abdulaziz University, Jeddah 21589, Saudi Arabia (e-mail: neeraj.kumar@thapar.edu).

Digital Object Identifier 10.1109/JSYST.2020.2998721

- 4) To design a scheme in which the user should be able to access the services of server after verifying it on the basis of user's identity and public key of registration center (RC).

### B. Organization of the Article

In this article, Section II describes the literature review. Section III presents few preliminary concepts needed to grasp the scheme. Section IV demonstrates the proposed scheme. Sections V and VI exhibit formal security analysis and security discussion, respectively. Section VII demonstrates performance analysis. Section VIII concludes this article.

## II. LITERATURE REVIEW

The problems in a conventional single-server authentication scheme, i.e., [6] lead to the development of MSA oriented paradigm. Li *et al.* [7] pioneered MSA protocol based on neural networks. Later, many MSA schemes [8], [9] could be witnessed. Then, another MSA scheme was presented on symmetric cryptoprimitives by Tsai [10], although some attacks were pointed by Liao *et al.* [11], onwards. However, Lin *et al.* [8], [11] do not comply with anonymity and were based on static identity. To comply with dynamic identity concept, Liao *et al.* [11] presented a dynamic ID-based anonymous MSA protocol, which was followed by other schemes [12]–[15]. These schemes provide computational efficiency however, unable to fulfill perfect forward secrecy. Later, some elliptic curve cryptography (ECC) based MSA schemes were presented to boost the security. The ECC-based schemes [16]–[18] show advantage over symmetric schemes, yet lacking relevance for MCC applications due to costly RC's involvement in authenticating cloud server and user. To encounter these issues, Tsai *et al.* [19] presented a privacy-aware authentication scheme employing bilinear pairing operation, which is followed by other schemes [20]–[27] sharing similar drawbacks. Meanwhile, to eliminate the involvement of RC, Mishra [28] and Lin *et al.* [29] introduced their works, however suffer scalability problems, in case more users or servers are added into the system. More recently, IoT and cyber-physical system-based cloud computing schemes were introduced [30]–[36]; however, these schemes were computation intensive or employed costly pairing operations besides solving other problems. In this context, we propose an efficient and security enhanced MSA protocol for MCC that achieves mutual authentication without using bilinear pairing operations and involving RC, employing only ECC-based operations.

## III. PRELIMINARIES

This section illustrates some preliminaries as following.

### A. MSA in MCC Context

In a single-server system (SSS), a mobile user needs to register with each server in the system, with whom the former may get the services. After getting registered from multiple servers in SSS, the mobile user has to maintain many SCs and credentials, which may sound frustrating for a mobile user. Fig. 2 depicts a

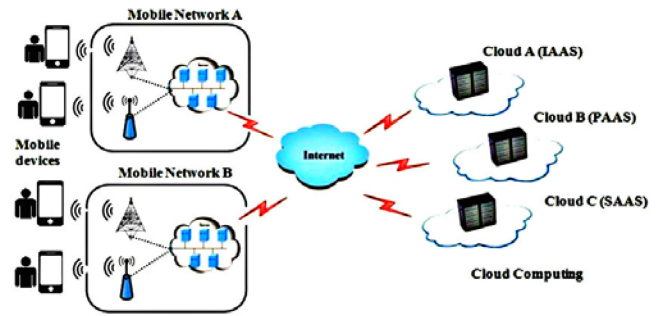


Fig. 1. MCC architecture.

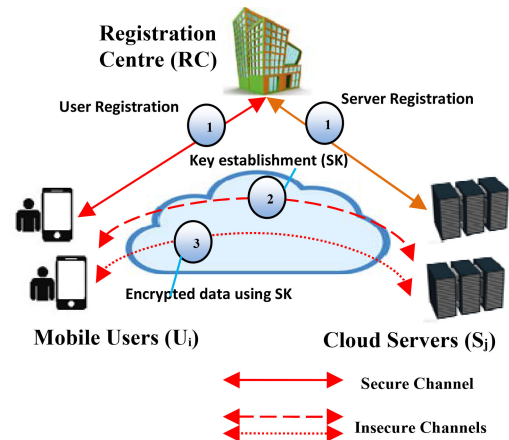


Fig. 2. Registration and mutual authentication for MSA in MCC.

multiserver environment, where mobile users and cloud servers register with a trusted authority, say RC [29], [30]. Using multiserver paradigm in MCC as depicted in Fig. 1, a mobile user may seek services from various cloud servers, after performing mutual authentication procedure using a single SC, identity and password as issued from RC during registration. In conventional MSA key agreements, mostly a user gets mutually authenticated with cloud server  $S_j$ , however, with the mandatory participation of RC in each session. In MCC, the elimination of RC from mutual authentication is even more significant for reducing computation and communication load due to the presence of power deficient mobile gadgets.

### B. Elliptic Curve Cryptography

The ECC, being one of public key cryptography-based algorithms, was pioneered by Koblitz [38]. The infinite field in ECC-based elliptic curves may comprise of both, even ( $F_2^m$ ) and odd fields ( $F_p$ ). The motivation of using ECC is due to the lesser key sizes as compared to RSA. For defining the elliptic curves in ECC, an elliptic curve equation is delineated as  $E_p(a, b)$ :  $y^2 = x^3 + ax + b \pmod{p}$  over a finite field  $F_p$ , where  $a, b \in F_p$  and  $4a^3 + 27b \neq 0 \pmod{p}$ . The following two definitions substantiate the security properties of ECC.

*Definition 1:* It is hard to extract  $\omega \in Z_p^*$  from  $\Xi = \omega P$ , where  $\Xi$  and  $P \in E_C(\delta, \sigma)$ . If  $\mathcal{A}$ 's advantage to derive  $\omega$  from

TABLE I  
NOTATIONS DEFINITION

Notations	explanation
$U_i, S_j, RC$	$i^{th}$ User, $j^{th}$ Server, Registration Centre
$ID_i, ID_j$	Identity of $U_i$ , Identity of $S_j$
$PW_u, B_i$	Password and biometric impression of $U_i$
$U_{pr}, S_{pr}$	Private key of $U_i$ , Private key of $S_j$
$U_{pb}, S_{pb}$	Public key of $U_i$ , Public key of $S_j$
$m_s, m_sP$	Private secret of RC, Public key of RC
$x_s, x_s'$	Ephemeral random secrets
$E_k(), D_k():$	Encryption, Decryption using key $K$
$h(), H_f():$	Secure hash function, Bio-hashing function
$\oplus,   :$	XOR, Concatenation

$\Xi$  in  $\tau_2$  is  $\text{Adv}_{\mathcal{A}}^{\text{ECDLP}}(\tau_2)$ , then  $\text{Adv}_{\mathcal{A}}^{\text{ECDLP}}(\tau_2)$  be advantage probability for random  $\tau_2$ . We define ECDLP as hard, if  $\text{Adv}_{\mathcal{A}}^{\text{ECDLP}}(\tau_2) \leq \mu_2$  holds for a small function  $\mu_2$

$$\text{Adv}_{\mathcal{A}}^{\text{ECDLP}}(\tau_2) = \Pr[\omega \in Z_p^* | \Xi = \omega P]. \quad (1)$$

**Definition 2:** To define an elliptic curve-based computational Diffie–Hellman problem (CDHP) on  $E_C$ , it is hard to build  $[\omega, \partial]P \in E_C(\delta, \sigma)$  from  $\Xi, \Omega$  given  $\Xi = [\omega]P, \Omega = [\partial]P, \Xi, \Omega, P \in E_C(\delta, \sigma)$ , and  $\omega, \partial \in Z_p^*$ . Thus, CDHP problem is hard if advantage probability  $\text{Adv}_{\mathcal{A}}^{\text{CDHP}}(\tau_3) \leq \mu_3$  in random time  $\tau_3$  for any negligibly small  $\mu_3$

$$\begin{aligned} \text{Adv}_{\mathcal{A}}^{\text{CDHP}}(\tau_3) &= \Pr[[\omega, \partial]P \in E_C(\delta, \sigma) | \Xi \\ &= [\omega]P \wedge \Omega = [\partial]P]. \end{aligned} \quad (2)$$

### C. Biohashing

The biohashing tool maps user's biometric input vector into random vectors, which then produces a code particular to each user, as called Biocode. The concept of Biohashing was surfaced when Lumini and Nanni [37] proposed a paired authenticator comprising of iterated inner products, further bearing token-based pseudorandom integers, such a unique biometric impression leads to construction of distinct compact codes.

## IV. PROPOSED MODEL

The multiserver architecture involves three participants, that is, user  $U_i$ , server  $S_j$ , and RC. RC chooses its master secret key  $m_s$ , long-term secret  $x$ , and public key  $m_s P$ , while  $m_s$  is only known to RC,  $x$  is shared with all servers, and  $m_s P$  is known publicly. The proposed model includes four subphases, i.e., server registration, user registration, login and authentication phase, and finally password update phase. Some notations used in the proposed scheme are depicted in Table I.

### A. Server Registration Phase

In this section, the server  $S_j$  registers itself through RC and adopts the understated steps for registration

- 1) First,  $S_j$  sends its identity  $ID_j$  to RC, as shown in Fig. 3.
- 2) Next, RC generates a random number  $a_j$  and calculates  $S_{pb} = a_j P$  and  $S_{pr} = a_j + m_s h(ID_j, S_{pb})$ . Then, it sends  $\{S_{pb}, S_{pr}, x\}$  to  $S_j$  using secure channel.
- 3)  $S_j$ , receives the message and stores  $\{S_{pr}, x\}$  safely and publishes  $S_{pb}$ .

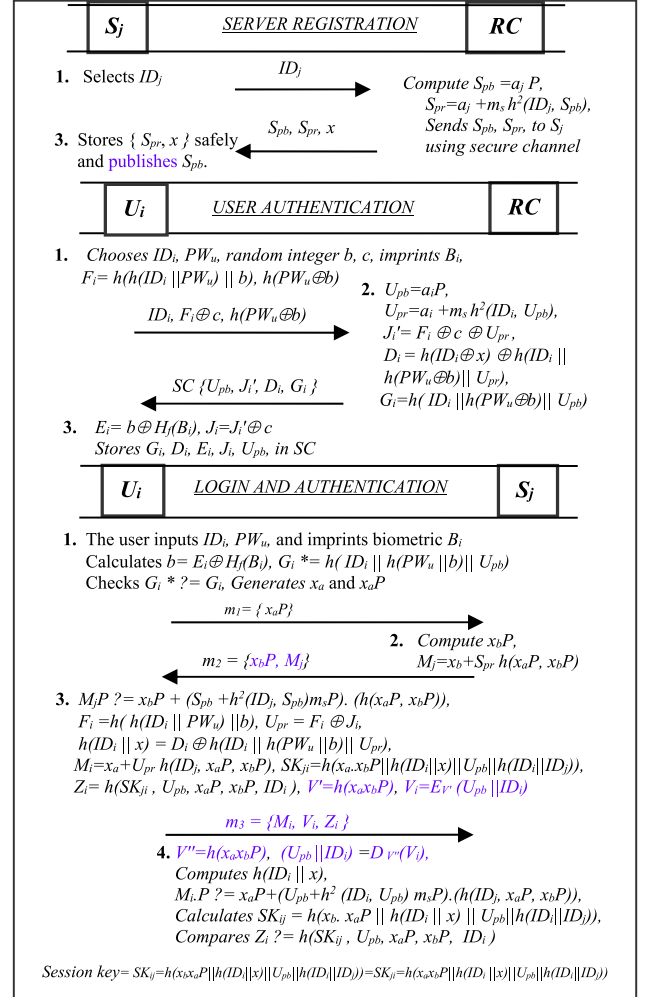


Fig. 3. Proposed model registration and mutual authentication.

### B. User Registration Phase

For registration,  $U_i$  performs few steps with RC as follows.

- 1) Initially,  $U_i$  selects  $ID_i, PW_u$ , random numbers  $b, c$ , imprints biometric  $B_i$ , and calculates  $F_i = h(h(ID_i || PW_u) || b)$  and  $h(PW_u \oplus b)$ . Next, it submits  $\{ID_i, F_i \oplus c, h(PW_u \oplus b)\}$  to RC using confidential channel.
- 2) Next, RC generates a random integer  $a_i$  and computes  $U_{pb} = a_i P, U_{pr} = a_i + m_s \cdot h(ID_i, U_{pb}), J_i' = F_i \oplus c \oplus U_{pr}, D_i = h(ID_i || x) \oplus h(ID_i || h(PW_u \oplus b) || U_{pr}), G_i = h(ID_i || h(PW_u \oplus b) || U_{pb})$  and stores  $\{U_{pb}, J_i', D_i, G_i\}$  in SC. Next, RC delivers the SC safely to  $U_i$ .
- 3) After receiving SC,  $U_i$  computes  $E_i = b \oplus H_f(B_i), J_i = J_i' \oplus c$  and stores in SC, as well. Finally, SC holds  $\{U_{pb}, J_i, D_i, G_i, E_i\}$ .

### C. Login and Authentication Phase

$U_i$  establishes a jointly approved session key with  $S_j$ , as shown in Fig. 3. The procedure is given as follows.

- 1)  $U_i$  inputs the identity  $ID_i$ , password  $PW_u$ , and imprints biometric  $B_i$  using biometric sensor. Thereafter,

SC calculates  $b = E_i \oplus H_f(B_i)$ ,  $G_i^* = h(\text{ID}_i || h(\text{PW}_u \oplus b) || U_{pb})$  and checks  $G_i^* ? = G_i$ . On failure, the SC abandons the session. Otherwise, SC generates a random number  $x_a$  and computes  $x_a P$ . Further, it submits  $m_1 = \{x_a P\}$  to  $S_j$ .

- 2)  $S_j$  receives  $m_1$  and generates random number  $x_b$ . Then it computes  $x_b P$  and  $M_j = x_b + S_{pr} h(x_a P, x_b P)$ . Next, it sends the message  $m_2 = \{x_b P, M_j\}$  toward  $U_i$  for verification.
- 3)  $U_i$  then computes and verifies  $M_j.P ? = (x_b P + S_{pb} + h(\text{ID}_j, S_{pb})m_s.P) \cdot (h(x_a P, x_b P))$ . If true,  $U_i$  calculates  $F_i = h(h(\text{ID}_i || \text{PW}_u) || b)$ ,  $U_{pr} = F_i \oplus J_i$ ,  $h(\text{ID}_i || x) = D_i \oplus h(\text{ID}_i || h(\text{PW}_u \oplus b) || U_{pr})$ ,  $M_i = x_a + U_{pr} h(\text{ID}_j, x_a P)$ ,  $SK_{ji} = h(x_a x_b P || h(\text{ID}_i || x) || U_{pb} || h(\text{ID}_i || \text{ID}_j))$ ,  $V' = h(x_a x_b P)$ ,  $V_i = E_{V'}(U_{pb} || \text{ID}_i)$  and  $Z_i = h(SK_{ji}, U_{pb}, x_a P, x_b P, \text{ID}_i)$ . Finally, it sends  $m_3 = \{M_i, V_i, Z_i\}$  to  $S_j$ .
- 4)  $S_j$  receives  $m_3$ , computes  $V'' = h(x_a x_b P)$ ,  $(U_{pb} || \text{ID}_i) = D_{V''}(V_i)$ , and recovers  $(U_{pb}, \text{ID}_i)$ . Next, it verifies  $M_i.P ? = (x_a P + U_{pb} + h(\text{ID}_i, U_{pb})m_s.P) \cdot (h(\text{ID}_j, x_a P, x_b P))$ . If both are true,  $S_j$  further computes  $h(\text{ID}_i || x)$  and  $SK_{ij} = h(x_b x_a P || h(\text{ID}_i || x) || U_{pb} || h(\text{ID}_i || \text{ID}_j))$  and compares  $Z_i ? = h(SK_{ij}, U_{pb}, x_a P, x_b P, \text{ID}_i)$ . If this holds true,  $S_j$  finally validates  $U_i$  as a legitimate user, with the established session key  $SK_{ij}$ .

#### D. Password Alteration Phase

$U_i$  may change its password by taking the following steps.

- 1)  $U_i$ , after inserting SD into device, inputs its identity ( $\text{ID}_i^*$ ), password ( $\text{PW}_u^*$ ), and imprints biometric pattern  $B_i^*$  into the scanner. Then SC calculates  $b = E_i \oplus H(B_i^*)$ ,  $G_i^* = h(\text{ID}_i || h(\text{PW}_u \oplus b) || U_{pb})$ , and checks  $G_i^* ? = G_i$ . If the equation does not match, it terminates session, or asks the user to insert a new password  $\text{PW}_u^{\text{new}}$ .
- 2) Next, the SC calculates  $F_i = h(h(\text{ID}_i || \text{PW}_u) || b)$ ,  $U_{pr} = F_i \oplus J_i$ ,  $h(\text{ID}_i || x) = D_i \oplus h(\text{ID}_i || h(\text{PW}_u \oplus b) || U_{pr})$ ,  $G_i^{\text{new}} = h(\text{ID}_i || h(\text{PW}_u^{\text{new}} \oplus b) || U_{pb})$ ,  $F_i' = h(h(\text{ID}_i || \text{PW}_u^{\text{new}}) || b)$ ,  $J_i^{\text{new}} = F_i' \oplus U_{pr}$ , and  $D_i^{\text{new}} = h(\text{ID}_i || x) \oplus h(\text{ID}_i || h(\text{PW}_u^{\text{new}} \oplus b) || U_{pr})$ .
- 3) Ultimately, the parameters  $G_i$ ,  $D_i$ , and  $J_i$  are exchanged with  $G_i^{\text{new}}$ ,  $D_i^{\text{new}}$ , and  $J_i^{\text{new}}$ , respectively, in SC.

#### V. FORMAL SECURITY ANALYSIS

In order to verify and evaluate the resilience of our protocol against recognized threats, we follow the security models [39], which warrant the scheme's security features (SFs) under random oracle model (ROM) as follows.

*Security goals:* The security goals for formal analysis are:

- 1) to establish a mutually agreed session key between participants proven under ROM;
- 2) to protect the privacy of the mobile user from adversary during mutual authentication phase.

*Interacting roles:* The members  $U_i \in \mathcal{U}$  or  $S_j \in \mathcal{S}$  meet in an authentication protocol  $\Pi$ . In a session, we symbolize an  $\vartheta$ th

TABLE II  
SECURITY FEATURES

<i>Execution</i> ( $\Pi_{U_i}^\vartheta, \Pi_{S_j}^\tau$ ): This query may be used by $\mathcal{A}$ to initiate passive threats upon eavesdropping executions between $\Pi_{U_i}^\vartheta$ and $\Pi_{S_j}^\tau$ . The query outputs messages as exchanged during execution of protocol $\Pi$ .
<i>SendClient</i> ( $\Pi_{U_i}^\vartheta, m$ ): This query is initiated by $\mathcal{A}$ to intercept $m$ and sends to $\Pi_{U_i}^\vartheta$ after modifying it. Its output suggests that $m$ is modified and generated by $\Pi_{U_i}^\vartheta$ . $\mathcal{A}$ launches $\Pi$ after invoking <i>SendClient</i> ( $\Pi_{U_i}^\vartheta, \text{Start}$ ).
<i>SendServer</i> ( $\Pi_{S_j}^\tau, m$ ): This query is launched by $\mathcal{A}$ to initiate an attack on $S_j$ . Its output reveals output as generated by $\Pi_{S_j}^\tau$ after receiving $m$ .
<i>Reveal</i> ( $\Pi_{U_i}^\vartheta$ ): $\mathcal{A}$ may utilize this query to initiate a session key attack, and may recover the session key for $\Pi_{U_i}^\vartheta$ .
<i>Corrupt</i> ( $U_i \leftrightarrow \rightsquigarrow$ ): The <i>Corrupt</i> query returns to an attacker, the long term secret $\text{PW}_u$ of the involved participant $U_i$ .
<i>Test</i> ( $\Pi_{U_i}^\vartheta$ ): $\mathcal{A}$ could submit a single <i>Test</i> query to the oracle. In return, the oracle would randomly select a bit $\mathcal{b} \in \{0, 1\}$ and outputs session key, if $\mathcal{b} = 1$ . Otherwise, it outputs a random value as response from $\{0, 1\}^*$ .

instance for  $U_i$  as  $\Pi_{U_i}^\vartheta$  and  $\tau$ th instance of  $S_j$  as  $\Pi_{S_j}^\tau$ . An instance  $\Pi_{U_i}^\vartheta$  or  $\Pi_{S_j}^\tau$  is regarded as *accepted*, if it holds session keys  $sk_{U_i}^\vartheta$  or  $sk_{S_j}^\tau$ , respectively.

*Long-term keys:* The  $U_i \in \mathcal{U}$  uses  $\text{PW}_u$  and a biometric  $B_i$  for login verification.  $U_i$  gets private and public key as  $U_{pr}$  and  $U_{pb}$ , while  $S_j \in \mathcal{S}$  as  $S_{pr}$  and  $S_{pb}$ , respectively, from RC.

*Adversarial model:* Some capabilities of an adversary in proposed scheme are assumed as following.

- 1) A probabilistic and polynomial time  $\mathcal{A}$ , with full authority on channels, may seize and manipulate messages on insecure communicating channels.  $\mathcal{A}$  may get SC contents and session-based ephemeral variables.
- 2)  $\mathcal{A}$  may be a malicious insider within an organization.
- 3)  $\mathcal{A}$  may communicate with any legitimate entity such as user  $U_i$ , server  $S_j$  or even RC to implement the oracle queries in Table II and get the competence to initiate the attack.

$\mathcal{A}$  might launch the understated queries in any sequence.

*Definition 3. Fresh Oracle:* The oracle  $\Pi_{U_i}^\vartheta$  is called *fresh* if

- 1)  $\Pi_{U_i}^\vartheta$  is in *accepted* state or  $\Pi_{U_i}^\vartheta$  and its partner constructs an agreed session key after exchange of communication messages.
- 2) Upon approval, the *Reveal* queries are not demanded or ever sent over to  $\Pi_{U_i}^\vartheta$  or the corresponding partner.

*Definition 4. Protocol's robustness* We depict the strength of authentication protocol ( $\Pi$ )'s using game *Game* ( $\Pi, \mathcal{A}$ ), where  $\mathcal{A}$  may initiate different queries, while it may use *Test* query just one time to the fresh oracle. If it uses *Test* query after accepting oracle  $\Pi_{U_i}^\vartheta$ ,  $\mathcal{A}$  would produce output for single bit  $\mathcal{b}'$ . The intent of  $\mathcal{A}$  is to make a guess of the bit  $\mathcal{b}'$  accurately in test session. We can describe the  $\mathcal{A}$ 's advantage as

$$\text{Adv}_{\mathcal{A}}^{\text{APLMCC}}(\mathcal{A}) = |2 \Pr(\mathcal{b}' = \mathcal{b}) - 1|. \quad (3)$$

The APLMCC is secure, if  $\text{Adv}_{\mathcal{A}}^{\text{APLMCC}}$  for  $\mathcal{A}$  is insignificant. For proving the security strength of contributed protocol, we employ DDH assumption as given in Definition 5.

*Definition 5. Diffie-Hellman (DDH) assumption:* This assumption is explained by two experiments,  $\text{Exp}_{P,g}^{\text{ddh-real}}(\mathbb{A})$  and  $\text{Exp}_{P,g}^{\text{ddh-rand}}(\mathbb{A})$ . The attacker  $\mathbb{A}$  is provided  $x_a P$ ,  $x_b P$ ,  $x_a x_b P$  in  $\text{Exp}_{P,g}^{\text{ddh-real}}(\mathbb{A})$ , and  $x_a P$ ,  $x_b P$ ,  $x_c P$  in

$\text{Exp}_{P,g}^{\text{ddh-rand}}(\mathbb{A})$ , where  $x_a$ ,  $x_b$ , and  $x_c$  are selected from  $Z_n^*$ ,  $g$  be large prime and  $P$  is generator of a finite cyclic group  $G$ . The  $\mathbb{A}$ 's advantage for violating DDH property is

$$\text{Adv}_{P,g}^{\text{ddh}}(\mathbb{A}) = \max \left\{ \left| \Pr \left[ \text{Exp}_{P,g}^{\text{ddh-real}}(\mathbb{A}) = 1 \right] - \Pr \left[ \text{Exp}_{P,g}^{\text{ddh-rand}}(\mathbb{A}) = 1 \right] \right| \right\}. \quad (4)$$

*Theorem 1:* We assume  $\mathcal{D}$  as a uniformly distributed dictionary of all possible passwords with length  $|\mathcal{D}|$ , and  $\Pi$  as proposed protocol. Using DDH assumption, we have

$$\text{Adv}_{\mathcal{D},\Pi}(\mathcal{A}) \leq \frac{qr_h^2}{2^l} + \frac{(qr_s + qr_e)^2}{g^2} + 2qr_e \cdot \text{Adv}_{P,g}^{\text{DDH}}(\mathbb{A}) + 2 \max \left\{ \frac{qr_h}{2^k}, \frac{qr_s}{|\mathcal{D}|} \right\} \quad (5)$$

while  $qr_s$ ,  $qr_e$ , and  $qr_h$  characterize the number of *SendClient*, *Execution*, and *Hash* queries, respectively; and  $l$  denotes length of a user's identity as well as password.

*Theorem proof (Game attacks):* This theorem is verified by series of games, beginning real attack  $G_0$ , and terminating at  $G_4$ , where  $\mathcal{A}$  does not have significant advantage. We may represent an event  $\varepsilon_n$  for each game  $G_n$ , while  $n$  ranges  $\{0 \leq n \leq 4\}$  in order that  $\mathcal{A}$  may accurately guess bit  $\mathcal{B}$  in the *Test* query.

*Game  $G_0$ :* This game  $G_0$  is a factual attack in ROM, and all instances are simulated as actual execution in oracle. For event  $\varepsilon_0$ ,  $\mathcal{A}$  may guess  $\mathcal{B}$  in *Test* query as

$$\text{Adv}_{\mathcal{D},\Pi}(\mathcal{A}) = \left| 2\Pr[\varepsilon_0] - \frac{1}{2} \right|. \quad (6)$$

*Game  $G_1$ :* The game  $G_1$  is alike game  $G_0$ , but it models hash oracles  $h$  and  $H$  by keeping hash list  $L_h$  bearing entries  $(In, Op)$ . The hash list outputs  $Op$  after consulting  $(In, Op)$  on receiving hash query. Otherwise, it selects  $Op$  randomly, modifies the entry  $(In, Op)$ , and submits to  $\mathcal{A}$ . From  $\mathcal{A}$ 's perspective, the real attacks on protocol and  $G_1$  simulation are indistinguishable. Since the oracles queries having *Send*, *Corrupt*, *Reveal*, *Hash*, *Execute*, and *Test* are simulated in the same manner as a real attack, hence

$$\Pr[\varepsilon_1] = \Pr[\varepsilon_0]. \quad (7)$$

*Game  $G_2$ :* This Game is just similar to  $G_1$  and we model these oracles with the exception that if  $\mathcal{A}$  makes a guess of participants' credentials, given that the partial transcripts  $\{x_aP, x_bP\}$  find a match with real values or face collisions, the executions will be terminated. If we refer to birthday paradox, the probability of collisions for random output of hash-oracles is not higher than  $\frac{qr_h^2}{2^{l+1}}$ . Similarly, its probability in the transcripts is at most  $\frac{(qr_s + qr_e)^2}{2g^2}$ . Hence

$$|\Pr[E_2] - \Pr[E_1]| \leq \frac{qr_h^2}{2^{l+1}} + \frac{(qr_s + qr_e)^2}{2g^2}. \quad (8)$$

*Game  $G_3$ :* In this game, the simulation of queries is altered to *SendClient* oracle. First, we select a session at random, executed by involved instances  $\prod_{U_i}^{\vartheta}$  and  $\prod_{S_j}^{\tau}$ .

- 1) On receiving *SendClient* ( $\prod_{U_i}^{\vartheta}$ , *Start*) query, we select at random, a value  $a \in Z_n^*$  to compute  $x_aP$  and submit to  $\mathcal{A}$ .
- 2) On receiving *SendServer* ( $\prod_{S_j}^{\tau}$ ,  $x_aP$ ) query, we select  $b \in Z_n^*$  and compute  $x_bP$ ,  $M_j$ ,  $x_b \cdot x_aP$ ,  $S_{pb}$  similar to real protocol, then would return  $\{x_bP, M_j\}$  to  $\mathcal{A}$ .
- 3) Next, on receiving *SendClient* ( $\prod_{U_i}^{\vartheta}$ ,  $(x_bP, M_j)$ ), we compute  $M_i$ ,  $V_i$ ,  $Z_i$ , and return  $\{M_i, V_i, Z_i\}$  to  $\mathcal{A}$ .

Hence, it is noticeable that  $G_3$  is completely indistinguishable of the previous game  $G_2$ . Hence

$$\Pr[\varepsilon_3] = \Pr[\varepsilon_2]. \quad (9)$$

*Game  $G_4$ :* In game  $G_4$ , the queries' simulation is again modified to *SendClient* oracle. The process of computing  $x_b x_aP$  is also changed to minimize its dependence on  $PW_u$  and temporary variables. When *SendServer* ( $\prod_{S_j}^{\tau}$ ,  $x_aP$ ) and *SendClient* ( $\prod_{U_i}^{\vartheta}$ ,  $(x_bP, M_j)$ ) are requested, we compute  $x_cP$ , where  $x_c$  is randomly selected from  $Z_n^*$ . The distinction between  $G_3$  and  $G_4$  becomes evident from the following equation:

$$|\Pr[E_4] - \Pr[E_3]| \leq qr_e \cdot \text{Adv}_{P,g}^{\text{DDH}}(\mathcal{A}). \quad (10)$$

Hence,  $\mathcal{A}$  might design a DDH solver after distinguishing  $G_3$  and  $G_4$ . In game  $G_4$ , we chose a randomly chosen key  $x_c$ , which could be evaluated from legal  $SK_{ij}$  using two cases.

*Case 1:* In this case,  $\mathcal{A}$  puts a query  $Z_i = h(SK_{ij}, U_{pb}, x_aP, x_bP, ID_i)$  to  $h$ . The event probability for this amounts to  $\frac{qr_h}{2^l}$ .

*Case 2:*  $\mathcal{A}$  requests *SendClient* query with the exception of *SendClient* ( $\prod_{S_j}^{\tau}$ ,  $m$ ) and successfully masquerades  $U_i$  as  $S_j$ .  $\mathcal{A}$  is not allowed to expose the static key  $PW_u$  of  $U_i$ . Therefore,  $\mathcal{A}$  needs to access  $U_i$ 's  $PW_u$  for impersonation threat. Its probability is calculated as  $1/|\mathcal{D}|$ . Since there are at most  $qr_s$  number of sessions for that type, hence the probability of occurred event is less than  $qr_s/|\mathcal{D}|$ , and we may infer the following equation:

$$\Pr[E_4] = \frac{1}{2} + \max \left\{ \frac{qr_h}{2^k}, \frac{qr_s}{|\mathcal{D}|} \right\} \quad (11)$$

Using (3)–(11), we have

$$\begin{aligned} \Pr[E_4] &= \frac{1}{2} + \max \left\{ \frac{qr_h}{2^k}, \frac{qr_s}{|\mathcal{D}|} \right\} \text{Adv}_{\mathcal{D},\Pi}(\mathcal{A}) = 2 \left| \Pr[\varepsilon_0] - \frac{1}{2} \right| \\ &= 2 \left( |\Pr[\varepsilon_0] - \Pr[\varepsilon_4]| + \max \left\{ \frac{qr_h}{2^k}, \frac{qr_s}{|\mathcal{D}|} \right\} \right) \\ &\leq 2 \left( |\Pr[\varepsilon_1] - \Pr[\varepsilon_2]| + |\Pr[\varepsilon_3] - \Pr[\varepsilon_4]| + \max \left\{ \frac{qr_h}{2^k}, \frac{qr_s}{|\mathcal{D}|} \right\} \right) \\ &\leq \left( \frac{qr_h^2}{2^l} \frac{(qr_s + qr_e)^2}{g^2} + qr_e \text{Adv}_{P,g}^{\text{DDH}}(\mathcal{A}) \right) 2 \max \left\{ \frac{qr_h}{2^k}, \frac{qr_s}{|\mathcal{D}|} \right\}. \end{aligned}$$

## VI. SECURITY DISCUSSION

The informal security discussion related to our scheme is illustrated as follows.

*Theorem 2:* Our scheme ensures user's anonymity and untraceability and resists offline password-guessing attack in line with threat model assumptions, while the probability of adversary's success is negligible.

*Proof:* In our scheme,  $U_i$  forwards its identity  $ID_i$  by computing  $V' = h(x_a x_b P)$ ,  $V_i = E_{V'}(U_{pb} || ID_i)$ . Since  $x_a x_b P$  can only be constructed by a valid  $S_j$ , we deduce that  $ID_i$  remains confidential with  $S_j$  having identity  $ID_j$ . Besides,

the exchanged messages do not contain any distinguishable factor to assist  $\mathcal{A}$  in tracing  $U_i$ . Besides,  $\mathcal{A}$  may intercept  $\{x_aP, M_i, V_i, x_bP, M_j, ID_j, Z_i\}$  on insecure channel. It may also extract SC contents  $\{G_i, D_i, E_i, J_i, U_{pb}\}$  using the side-channel attack. Nonetheless,  $\mathcal{A}$  cannot get  $PW_u$  from these contents since  $PW_u$  is not used in the construction of parameters except  $G_i$  and  $J_i$ . To guess  $PW_u$  from  $G_i$  and  $F_i$ ,  $b$  is required, which cannot be derived from  $E_i$  until biometric value is known. Hence, it is proved that our scheme not only provides anonymity and untraceability to  $U_i$  but also resists offline password-guessing attack. ■

*Theorem 3:* Our scheme is immune to user and server impersonation attacks in consideration with the assumptions of threat model.

*Proof:* In our scheme,  $\mathcal{A}$  cannot initiate an impersonation attack as long as  $\mathcal{A}$  is not able to access  $PW_u$ , or private keys of participants. Since to forge the user,  $\mathcal{A}$  cannot modify  $M_i$  corresponding to  $x_bP$  until  $\mathcal{A}$  gets access to  $PW_u$  and  $U_{pr}$  of a particular user. Likewise, any malicious server may attempt a server spoofing attack toward  $U_i$ ; however, it cannot construct  $M_j$  against  $x_aP$  for not having  $S_{pr}$  of a server.  $\mathcal{A}$  can only replay  $M_j$ , which is detected upon the  $U_i$ 's verification for  $M_j.P? = (x_bP + S_{pb} + h(ID_j, S_{pb})m_s.P) \cdot (h(x_aP, x_bP))$ . Therefore, it is proved that our scheme is resistant to masquerading attacks. ■

*Stolen card attack:* In this attack,  $\mathcal{A}$  steals SC and attempts to guess user's password and launch an impersonation attack. Nevertheless, in our scheme to launch any type of replay, forgery, or modification attack,  $\mathcal{A}$  will need access to  $U_{pr}$  for generating  $M_i$  and ultimately  $m_3 = \{M_i, V_i, Z_i\}$ , which is not possible without accessing biometric  $B_i$ . Hence, the stolen SC contents may not help adversary in this attack.

*Known-key security:* The knowledge of session key may not assist  $\mathcal{A}$  in guessing either user's or server's private key or session keys (past or future) in proposed scheme.

*Perfect forward secrecy:* The contributed protocol adheres to perfect forward secrecy, even if private keys of RC,  $U_i$  or  $S_j(m_s, x, U_{pr}, S_{pr})$  are lost involuntarily. That is, the contributed scheme develops a session key as  $SK_{ji} = h(x_ax_bP || h(ID_i || x) || U_{pb} || h(ID_i || ID_j))$  by using  $x_a$  and  $x_b$  as  $x_ax_bP$ , which is intractable due to ECDLP, and cannot be guessed in polynomial time.

*No session-specific temporary information attack:* Our scheme provides protection to session keys in case the corresponding session specific temporary variables are exposed accidentally, i.e.,  $x_a$  or  $x_b$ , since  $\mathcal{A}$  requires access to user's dynamic key  $h(ID_i || x)$  for constructing session key  $SK_{ji} = h(x_ax_bP || h(ID_i || x) || U_{pb} || h(ID_i || ID_j))$ . However,  $h(ID_i || x)$  cannot be extracted from  $D_i$  in SC until  $h(PW_u \oplus b)$  and  $U_{pr}$  are also known. Hence, our scheme defies a session-specific temporary information attack.

*No key-compromise impersonation (KCI) attack:* The KCI attack is possible when some of the leaked parameter of user's domain leads to server masquerading attack [41], [42]. In our protocol, even if user's parameter  $U_{pr}$  is compromised, it may not lead to server masquerading attack, as  $\mathcal{A}$  cannot construct a fresh  $M_j = x_b + S_{pr} h(x_aP, x_bP)$  without knowledge of

TABLE III  
SECURITY FEATURES

	[19]	[20]	[21]	[22]	[23]	[24]	[25]	[26]	[30]	[31]	[32]	Ours
SF1	✓	✓	×	✓	×	×	×	✓	✓	✓	✓	✓
SF2	✓	✓	×	✓	✓	✓	✓	✓	×	✓	✓	✓
SF3	✓	✓	✓	✓	×	×	×	✓	×	✓	✓	✓
SF4	✓	✓	✓	×	×	×	×	✓	✓	✓	✓	✓
SF5	✓	✓	×	×	×	×	×	✓	✓	✓	✓	✓
SF6	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓
SF7	×	×	×	×	×	×	×	✓	✓	✓	✓	✓
SF8	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓
SF9	✓	✓	✓	✓	✓	✓	✓	×	×	✓	✓	✓
SF10	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SF11	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SF12	×	✓	×	×	×	×	×	✓	×	✓	✓	✓
SF13	✓	✓	×	✓	✓	✓	✓	✓	×	✓	✓	✓
SF14	✓	✓	×	✓	✓	✓	✓	×	×	×	✓	✓
SF15	×	✓	×	×	✓	✓	✓	✓	✓	×	✓	✓
SF16	✓	✓	×	×	✓	✓	✓	✓	×	✓	✓	✓
SF17	×	✓	✓	×	✓	✓	✓	✓	×	✓	✓	✓
SF18	✓	✓	✓	✓	×	✓	✓	×	×	×	✓	✓
SF19	✓	✓	✓	✓	✓	✓	✓	×	×	×	×	✓
SF20	✓	✓	✓	✓	✓	✓	✓	×	×	×	×	✓

SF1: Provides anonymity. SF2: Provides mutual authenticity. SF3: Resist malicious insider threat. SF4: Resist offline password-guessing threat. SF5: Resists stolen SC threat. SF6: Resist replay threat. SF7: Pairing-free protocol. SF8: Provides session key verification. SF9: No verifier table. SF10: Provides perfect forward secrecy. SF11: Provides known key secrecy. SF12: Resists impersonation attack. SF13: Resists desynchronization attack. SF14: Resists DoS attack. SF15: Resists session-specific temporary information attack. SF16: Resists KCI attack. SF17: Resists private key guessing attack. SF18: No time synchronization required. SF19: Scalable (easy server addition into network). SF20: Immune to traceability attack.

$S_{pr}$ . A user may foil any such attack by verifying  $M_j.P? = (x_bP + S_{pb} + h(ID_j, S_{pb})m_s.P) \cdot (h(x_aP, x_bP))$ . Hence, the proposed scheme is resistant to KCI attack.

## VII. PERFORMANCE AND EVALUATION

With a consideration to contribute an MCC scheme without bilinear pairing operations and achieving the same objectives as given in Section I-A, we proposed a novel scheme and achieved the following results in our favor. This section analyzes the performance of the proposed model against the works presented in [19]–[26] and [30]–[32] in terms of vulnerability analysis and computational or communicational costs. It is obvious from Table III that the schemes [19], [21]–[25], [30] for MCC do not provide immunity from impersonation attacks. Moreover, the works presented in [19]–[21], [22], and [31] do not provide immunity of session-specific ephemeral information attack, while [21]–[23], [25] do not resist against stolen SC attacks. The schemes [21], [30] are not secure against KCI attack, while schemes presented in [26] and [32] suffer traceability. Second, those protocols do not comply with mutual authentication, and also suffer the desynchronization problem during biometric imprinting in login phase. The schemes [21], [23]–[25] do not maintain user's anonymity, while schemes presented in [21], [26], and [31] are unable to resist against DoS attack due to the verifiers' maintenance at server. The schemes [26]–[32] affect the scalability of the system, i.e., a new server cannot be added without upgrading SCs of users. Besides, the work presented in [30] does not provide session key verification to participants and is prone to insider attack. The schemes [23], [26]–[31] suffers time synchronization problem. Likewise, offline password-guessing attack could be initiated against [22]–[25]. One of the salient features that distinguish our technique with [19]–[22] is that it provides mutual authentication without employing the bilinear pairing operations. While the

TABLE IV  
COMPUTATIONAL COST COMPARISON (MS)

	User (U)	Server (S)	RTT
[19]	$3T_{ECPM} + 4T_H + 4T_{SYM} \approx 34.484$	$1T_{ECPM} + 2T_{BP} + 2T_H + 2T_{PA} \approx 12.952$	47.43
[20]	$2T_{ECPM} + 4T_H + 1T_{MTP} + 1T_{SYM} + 2T_{Exp} \approx 57.012$	$1T_{ECPM} + 5T_H + 2T_{BP} + 2T_{PA} + 2T_{Exp} \approx 13.408$	70.42
[21]	$3T_{ECPM} + 4T_H + 1T_{MTP} + 1T_{PA} \approx 65.46$	$2T_{ECPM} + 5T_H + 2T_{BP} + 2T_{PA} + 1T_{Exp} \approx 15.154$	80.81
[22]	$3T_{ECPM} + 4T_H + 1T_{MTP} + 1T_{PA} \approx 63.72$	$5T_H + 2T_{Exp} + 2T_{BP} + 1T_{PA} \approx 11.40$	75.12
[23]	$4T_{ECPM} + 8T_H + 2T_{PA} \approx 45.606$	$4T_{ECPM} + 5T_H + 2T_{PA} \approx 8.202$	53.80
[24]	$8T_{ECPM} + 8T_H + 2T_M \approx 90.386$	$4T_{ECPM} + 4T_H + 2T_M + 2T_{BP} \approx 18.78$	109.1
[25]	$2T_{ECPM} + 6T_H \approx 22.858$	$3T_{ECPM} + 7T_H + 4T_{SYM} \approx 6.224$	29.08
[26]	$12T_H + 1T_{FE} \approx 12.032$	$8T_H \approx 0.08$	12.11
[30]	$2T_{ECPM} + 4T_H + 3T_{SYM} \approx 23.123$	$1T_{ECPM} + 4T_H + 2T_{SYM} \approx 2.104$	25.22
[31]	$5T_H + 4T_{ECPM} \approx 45.247$	$4T_H + 4T_{ECPM} \approx 8.144$	53.39
[32]	$5T_H + 2T_{SYM} + 2T_M \approx 0.625$	$3T_H + 2T_{SYM} + 2T_M \approx 0.064$	0.068
Ours	$3T_{ECPM} + 10T_H + 1T_{SYM} \approx 34.48$	$3T_{ECPM} + 10T_H + 1T_{SYM} \approx 6.197$	40.67

TABLE V  
COMPUTATIONAL COSTS OF OPERATIONS

	User (ms)	Server (ms)
$T_{ECPM}$	11.228	2.026
$T_{MTP}$	29.433	5.388
$T_{Exp}$	2.361	0.325
$T_{SYM}$	0.133	0.019
$T_{BP}$	28.592	5.317
$T_{PA}$	0.079	0.024
$T_M$	0.013	0.003
$T_H$	0.067	0.010

work presented in [24] is also pairing-free MCC authentication protocol, however, it undergoes many limitations. The schemes [25], [26], [30] are efficient in terms of computation; however, these schemes suffer from other limitations, as given in Table IV. For mutual authentication, our scheme employs an ECC-based point multiplication that is a comparatively less computation intensive than a bilinear pairing operation.

Table V lists few symbols for timings of used operations that represent  $T_H$  for one-way hash operation,  $T_{SYM}$  for symmetric key-based operation,  $T_{BP}$  for bilinear pairing,  $T_{ECPM}$  for EC-based point multiplication,  $T_M$  for multiplication,  $T_{PA}$  for point addition,  $T_{Exp}$  for performing modular exponentiation,  $T_{MTP}$  for computing map-to-point function, and  $T_{FE}$  for fuzzy extraction operation.

The computational costs are computed using MIRACL library [40] upon implementation on Android-based client, i.e., a mobile device (Lenovo Zuk Z1, Quad-core 2.5 GHz processor with 3 GB of RAM and Android V5.1.1 Operating system). For implementing a server, we used a personal computer (HP-based E8300 having Core i5, 2.96 GHz processor with 6 GB of RAM and Ubuntu 16.12 Operating system). We conducted experiments on client-based mobile device and server-based personal computer, and the related costs are given in Table V. Table IV compares computational costs of [19]–[26] and [30]–[32] against our scheme. It is obvious that on user's end, schemes in [20]–[22] bear high computational cost, i.e., 65.46, 63.72, and 90.38 ms, respectively, due to costly map-to-point hash function and bilinear pairing operations. The schemes [19], [23]–[26], [30]–[32] and our scheme take 34.48, 45.60, 90.38, 22.85, 12.03, 23.12, 45.24, 0.625, and 34.48 ms respectively. The schemes [19]–[24] are computation intensive either due to costly map-to-hash operations or bilinear pairing operations or

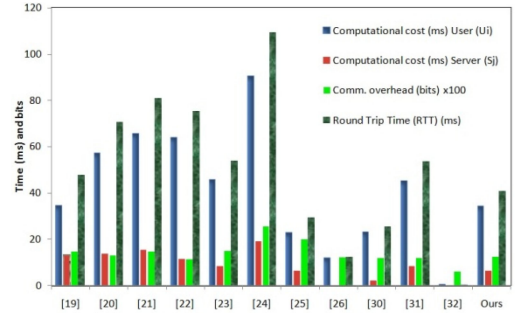


Fig. 4. Pictorial view of comparative analysis.

TABLE VI  
COMMUNICATION DELAY

	Comm. cost (bits)	Number of exchanged messages
[19]	1472	4
[20]	1312	4
[21]	1472	3
[22]	1152	3
[23]	1504	4
[24]	2560	3
[25]	2016	3
[26]	1216	4
[30]	1184	2
[31]	1184	3
[32]	608	4
Ours	1248	3

higher number of ECC-based point multiplications. Despite the heavy computational costs, these schemes are prone to many limitations, including vulnerability to lacking anonymity, insider threats, offline guessing attacks, stolen SC threats, impersonation, and session specific ephemeral secrets leakage threats. Although schemes in [25], [26], and [30] take less computational overheads than our scheme but are exposed to many security drawbacks, as given in Fig. 4 as well as Table III. Moreover, schemes in [26] and [30]–[32] are not scalable and restrict free addition of users or service providers into the system since there is a tradeoff between efficiency and computational cost. The balance of the two needs to be maintained for a secure and efficient protocol. Our scheme takes far less computational cost than [19]–[24], [31] since it is pairing-free MSA protocol as depicted in the round trip time (RTT) column of Table IV as well as Fig. 4. The RTT, being the total network latency, is the sum of total execution time for user as well as server.

Our scheme takes a bit more computations than [25], [26], [30]; however, it is secure than all those counterparts as evident from Table III. To the best of our knowledge, yet there is no pairing-free MSA protocol providing secure mutual authentication to clients. The security properties of the proposed scheme are complemented by theoretical proofs.

Table VI depicts the communication overhead in bits for various schemes [19]–[26], [30]–[32]. We assume the communication cost for various parameters, i.e., for user identity as 160-bits, random number as 160-bits, hash function digest (SHA-1) as 160-bits, elliptic curve point (pair) as 320-bits, timestamp as 32-bits, modular exponentiation as 160-bits, and symmetric encryption as 128-bits. The communication cost of the proposed protocol is computed as 1248-bits which is efficient

as compared to [19]–[21], [23]–[25], although a little higher than other schemes [22], [26], [30]–[32]. There is a tradeoff between overheads and strong security, the security of our scheme is verified through rigorous random oracle-based formal analysis. Therefore, in comparison with previous schemes, our scheme provides effective mutual authentication to participants, immunity from known attacks, and computational and communicational efficiency in MCC framework without involving registration authority.

### VIII. CONCLUSION

The security hazards have become an impediment in the way of adapting MCC paradigm. To secure the MCC environment, many schemes have been demonstrated lately in order to meet the dynamic needs of mobile users for resources. The schemes either have security loopholes or being inefficient on account of costly pairing-based implementations. To defy the identified limitations, we proposed a novel pairing-free MSA protocol for MCC environment that is far efficient than other contemporary schemes. The proposed scheme not only provides computational cost efficiency but also preserves the features of costly pairing schemes, such as achieving secure mutual authentication, anonymity, scalability etc. The SFs are supported with rigorous and formal theoretical analysis.

### REFERENCES

- [1] C. Stergiou, K. E. Psannis, B. G. Kim, and B. Gupta, "Secure integration of IoT and cloud computing," *Future Gener. Comput. Syst.*, vol. 78, pp. 964–975, 2018.
- [2] M. Díaz, C. Martín, and B. Rubio, "State-of-the-art, challenges, and open issues in the integration of Internet of things and cloud computing," *J. Netw. Comput. Appl.*, vol. 67, pp. 99–117, 2016.
- [3] N. Fernando, S. W. Loke, and W. Rahayu, "Mobile cloud computing: A survey," *Future Gener. Comput. Syst.*, vol. 29, no. 1, pp. 84–106, 2013.
- [4] M. B. Mollah, M. A. K. Azad, and A. Vasilakos, "Security and privacy challenges in mobile cloud computing: Survey and way ahead," *J. Netw. Comput. Appl.*, vol. 84, pp. 38–54, 2017.
- [5] S. Fatima and S. Ahmad, "An exhaustive review on security issues in cloud computing," *KSII Trans. Internet Inf. Syst.*, vol. 13, no. 6, pp. 3219–3237, 2019.
- [6] C. L. Hsu, "Security of Chien et al.'s remote user authentication scheme using smart cards," *Comput. Standards Interfaces*, vol. 26, no. 3, pp. 167–169, 2004.
- [7] L. H. Li, I. C. Lin, and M. S. Hwang, "A remote password authentication scheme for multiserver architecture using neural networks," *IEEE Trans. Neural Netw.*, vol. 12, no. 6, pp. 1498–1504, Nov. 2001.
- [8] I. C. Lin, M. S. Hwang, and L. H. Li, "A new remote user authentication scheme for multi-server architecture," *Future Gener. Comput. Syst.*, vol. 1, no. 19, pp. 13–22, 2003.
- [9] W. S., Juang, "Efficient multi-server password authenticated key agreement using smart cards," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 251–255, Feb. 2004.
- [10] J. L., Tsai, "Efficient multiserver authentication scheme based on one-way hash function without verification table," *Comput. Standards Interfaces*, vol. 27, no. 3/4, pp. 115–121, 2008.
- [11] Y. P., Liao and S. S. Wang, "A secure dynamic ID based remote user authentication scheme for multiserver environment," *Comput. Standards Interfaces*, vol. 31, no. 1, pp. 24–29, 2009.
- [12] H. C. Hsiang and W. K. Shih, "Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment," *Comput. Standards Interfaces*, vol. 31, no. 6, pp. 1118–1123, 2009.
- [13] C. C. Lee, T. H. Lin, and R. X. Chang, "A secure dynamic ID based remote user authentication scheme for multiserver environment using smart cards," *Expert Syst. Appl.*, vol. 38, no. 11, pp. 13863–13870, 2011.
- [14] X. Li, Y. P. Xiong, J. Ma, and W. D. 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.
- [15] X. Li, J. Ma, W. Wang, Y. Xiong, and J. Zhang, "A novel smart card and dynamic ID based remote user authentication scheme for multi-server environments," *Math. Comput. Model.*, vol. 58, no. 1/2, pp. 85–95, 2013.
- [16] E. J. Yoon and Y. K. Yoo, "Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem," *J. Supercomput.*, vol. 63, no. 1, pp. 235–255, 2013.
- [17] 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.
- [18] V. Odelu, A. K. Das, and A. Goswami, "A secure biometrics-based multi-server authentication protocol using smart cards," *IEEE Trans. Inf. Forensics Secur.*, vol. 10, no. 9, pp. 1953–1966, Sep. 2015.
- [19] J. L. Tsai and N. W. Lo, "A privacy-aware authentication scheme for distributed mobile cloud computing services," *IEEE Syst. J.*, vol. 9, no. 3, pp. 805–815, Sep. 2015.
- [20] D. He *et al.*, "Efficient privacy-aware authentication scheme for mobile cloud computing services," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1621–1631, Jun. 2018.
- [21] T. H. Chen, H. L. Yeh, and W. K. Shih, "An advanced ECC dynamic ID-based remote mutual authentication scheme for cloud computing," in *Proc. 5th FTRA Int. Conf. Multimedia Ubiquitous Eng.*, 2011, pp. 155–159.
- [22] W. B. Hsieh and J. S. Leu, "An anonymous mobile user authentication protocol using self-certified public keys based on multi-server architectures," *J. Supercomput.*, vol. 70, no. 1, pp. 133–148.52, 2014.
- [23] H. Li, F. Li, C. Song, and Y. Yan, "Towards smart card based mutual authentication schemes in cloud computing," *KSII Trans. Internet Inf. Syst.*, vol. 9, no. 7, pp. 2719–2735, 2015.
- [24] M. Wazid, A. K. Das, S. Kumari, X. Li, and F. Wu, "Provably secure biometric-based user authentication and key agreement scheme in cloud computing," *Secur. Commun. Netw.*, vol. 9, no. 17, pp. 4103–4119, 2016.
- [25] M. Alizadeh, S. Abolfazli, M. Zamani, S. Baharun, and K. Sakurai, "Authentication in mobile cloud computing: A survey," *J. Netw. Comput. Appl.*, vol. 61, pp. 59–80, 2016.
- [26] A. Irshad, M. Sher, H. F. Ahmad, B. A. Alzahrani, and S. A. Chaudhry, "An improved multi-server authentication scheme for distributed mobile cloud computing services," *KSII Trans. Internet Inf. Syst.*, vol. 10, no. 12, pp. 5529–5552, 2016.
- [27] R. Amin, S. K. Islam, G. P. Biswas, D. Giri, M. K. Khan, and N. Kumar, "A more secure and privacy-aware anonymous user authentication scheme for distributed mobile cloud computing environments," *Secur. Commun. Netw.*, vol. 9, no. 17, pp. 4650–4666, 2016.
- [28] D. Mishra, "Design and analysis of a provably secure multi-server authentication scheme," *Wireless. Pers. Commun.*, vol. 86, no. 3, pp. 1095–1119, 2016.
- [29] H. Lin, F. Wen, and C. Du, "An improved anonymous multi-server authenticated key agreement scheme using smart cards and biometrics," *Wireless Personal Commun.*, vol. 84, no. 4, pp. 2351–2362, 2015.
- [30] A. Karati, R. Amin, S. H. Islam, and K. K. R. Choo, "Provably secure and lightweight identity-based authenticated data sharing protocol for cyber-physical cloud environment," *IEEE Trans. Cloud Comput.*, vol. 9, no. 1, pp. 318–330, Jan.-Mar. 2021.
- [31] R. Amin, N. Kumar, G. P. Biswas, R. Iqbal, and V. Chang, "A light weight authentication protocol for IoT-enabled devices in distributed cloud computing environment," *Future Gener. Comput. Syst.*, vol. 78, pp. 1005–1019, 2018.
- [32] S. A. Chaudhry, I. L. Kim, S. Rho, M. S. Farash, and T. Shon, "An improved anonymous authentication scheme for distributed mobile cloud computing services," *Cluster Comput.*, vol. 22, no. 1, pp. 1595–1609, 2019.
- [33] P. Wang, B. Li, H. Shi, Y. Shen, and D. Wang, "Revisiting anonymous two-factor authentication schemes for IoT-enabled devices in cloud computing environments," *Secur. Commun. Netw.*, vol. 2, 2019.
- [34] M. Vivekanandan, V. N. Sastry, and U. S. Reddy, "Biometric based user authentication protocol for mobile cloud environment," in *Proc. IEEE 5th Int. Conf. Identity, Secur. Behav. Anal.*, 2019, pp. 1–6.
- [35] Y. Yu, L. Hu, and J. Chu, "A secure authentication and key agreement scheme for IoT-based cloud computing environment," *Symmetry*, vol. 12, no. 1, p. 150, 2020.
- [36] M. Nikooghadam and H. Amintoosi, "Secure communication in Cloud-IoT through design of a lightweight authentication and session key agreement scheme," *Int. J. Commun. Syst.*, 2020, Art. no. e4332.



- [37] A. Lumini and N. Loris, "An improved bio-hashing for human authentication," *Pattern Recognit.*, vol. 40, no. 3, pp. 1057–1065, 2007.
- [38] N. Kobitz, "Elliptic curve cryptosystems," *Math. Comput.*, vol. 48, pp. 203–209, 1987.
- [39] M. Ballare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *Proc. ACM Conf. Comput. Commun. Secur.*, 1993, pp. 62–73.
- [40] Shamus Software, Ltd., Miracl Library. [Online]. Available: <http://www.shamus.ie/index.php?page=home>
- [41] A. A. Nasr, K. Dubey, N. A. El-Bahnasawy, S. C. Sharma, G. Attiya, and A. El-Sayed, "HPFE: A new secure framework for serving multi-users with multi-tasks in public cloud without violating SLA," *Neural Comput. Appl.*, vol. 32, pp. 6821–6841, 2020.
- [42] K. Dubey, M. Y. Shams, S. C. Sharma, A. Alarifi, M. Amoon, and A. A. Nasr, "A management system for servicing multi-organizations on community cloud model in secure cloud environment," *IEEE Access*, vol. 7, pp. 159535–159546, 2019.



**Azeem Irshad** received the master's degree from Arid Agriculture University, Rawalpindi, Pakistan, and the Ph.D. degree in computer science from International Islamic University, Islamabad, Pakistan.

He has authored more than 64 international journal and conference publications, including 33 SCI-E journal publications. His research work has been cited over 646× with 12h-index and 14 i-10-index. His research interests include strengthening of authenticated key agreements in cloud-IoT, smart grid, pervasive edge computing, CPS, 5G networks, WSN, SIP and multiserver architectures.

ad hoc networks, e-health clouds, SIP and multiserver architectures.

Dr. Irshad was a recipient of the Top Peer-Reviewer Award from Publons in 2018 with 126 verified reviews. He has served as a Reviewer for more than 40 reputed journals including *IEEE Systems Journal*, *IEEE Communications Magazine*, *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*, *IEEE Consumer Electronics Magazine*, *IEEE Sensors Journal*, *IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY*, *IEEE Industry Applications Society*, *Computer Networks*, *Information Sciences*, *CAEE*, *Cluster Computing*, *AIHC*, *JNCA*, and *FGCS*, notably.



**Shehzad Ashraf Chaudhry** received the master's and Ph.D. degrees in computer science (with distinction) from International Islamic University Islamabad, Pakistan, in 2009 and 2016, respectively.

He is currently working as an Associate Professor with the Department of Computer Engineering, Faculty of Engineering and Architecture, Istanbul Gelisim University, Istanbul, Turkey. He has authored more than 100 scientific publications appeared in different international journals and proceedings, including more than 72 in SCI/E journals. With an

H-index of 23 and an I-10 index 43, his work has been cited more than 1650 times. He has also supervised more than 35 graduate students in their research. His current research interests include lightweight cryptography, elliptic/hyper elliptic curve cryptography, multimedia security, E-payment systems, MANETs, SIP authentication, smart grid security, IP multimedia subsystem, and next-generation networks. He occasionally writes on issues of higher education in Pakistan.

Dr. Chaudhry was a recipient of the Gold Medal for achieving 4.0/4.0 CGPA in his Masters. Considering his research, Pakistan Council for Science and Technology granted him the Prestigious Research Productivity Award, while affirming him among Top Productive Computer Scientist in Pakistan. He has served as a TPC member of various international conferences and is an Active Reviewer of many ISI indexed journals.



**Osama Ahmad Alomari** received the B.Sc. degree in computer science from Al Al-Bayt University, Mafrqa, Jordan, in 2005, the M.Sc. degree in computer science from the National University of Malaysia, Bangi, Malaysia, in 2012, and the Ph.D. degree in computer science (artificial intelligence) from Universiti Sains Malaysia, George Town, Malaysia, in 2018.

He is an Assistant Professor with the Department of Computer Engineering, Istanbul Gelisim University, Istanbul, Turkey. In general, his research interests are optimization, pattern recognition, feature selection, microarray data analysis, machine, and deep learning, and network security.



**Khalid Yahya** received the Ph.D. degree in electrical engineering from Kocaeli University, Kocaeli, Turkey, in 2018.

He is presently working as an Assistant Professor of Mechatronics Engineering with Istanbul Gelisim University, Istanbul, Turkey. He has authored or coauthored more than a dozen papers in prestigious journals and conferences. He is an Active Reviewer of many conferences and journals. His current research interest includes microelectronic circuit analysis and design, renewable energy resources, power electronics, and MPPT designs for energy harvesting systems and information security.



**Neeraj Kumar** (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from Shri Mata Vaishno Devi University, Katra (J&K), India, in 2009.

He was a Postdoctoral Research Fellow with Coventry University, Coventry, U.K. He is currently a full Professor with the Department of Computer Science and Engineering, Thapar University, Patiala, India. He has authored more than 400 technical research papers published in leading journals and conferences from the IEEE, Elsevier, Springer, Wiley, etc. Some

of his research findings are published in top cited journals such as the *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS*, *IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING*, *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, *IEEE TRANSACTIONS ON CONSUMER ELECTRONICS*, *IEEE NETWORK*, *IEEE COMMUNICATIONS*, *IEEE WIRELESS COMMUNICATIONS*, *IEEE Internet of Things Journal*, and *IEEE Systems Journal*. He has guided many research scholars leading to Ph.D. and M.E./M.Tech. He is in the editorial board of the *Journal of Network and Computer Applications* (Elsevier), *IEEE SYSTEMS JOURNAL*, *ACM Computing Survey*, *IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING*, *IEEE NETWORKS*, *IEEE COMMUNICATION MAGAZINE*, and *International Journal of Communication Systems* (Wiley).