Computers and Electrical Engineering 98 (2022) 107692

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

A low-cost privacy preserving user access in mobile edge computing framework☆

Azeem Irshad [a], Shehzad Ashraf Chaudhry [b,*], Anwar Ghani [a], Ghulam Ali Mallah [c], Muhammad Bilal [d], Bander A. Alzahrani [e]

[a] Department of computer science & software engineering, International Islamic University, Islamabad, Pakistan
[b] Department of Computer Engineering, Faculty of Engineering and Architecture, Istanbul Gelisim University, Istanbul, Turkey
[c] Department of Computer Science, Shah Abdul Latif University, Khairpur, Sindh 66111, Pakistan
[d] Department of Computer Engineering, Hankuk University of Foreign Studies, Yongin-si, Gyeonggido, 17035, Korea
[e] Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

ARTICLE INFO

Keywords:
Pervasive and mobile edge computing
Authentication
Symmetric key operations
PUF
IoT

ABSTRACT

The computational offloading from conventional cloud datacenter towards edge devices sprouted a new world of prospective applications in pervasive and Mobile Edge Computing (MEC) paradigm, leading to substantial gains in the form of increased availability, bandwidth with low latency. The MEC offers real-time computing and storage facility within the proximity of mobile user-access network, hence it is imperative to secure communication between end user and edge server. The existing schemes do not fulfill real time processing and efficiency requirements for using complex crypto-primitives. To this end, we propose a novel two-factor biometric authentication protocol for MEC enabling efficient and secure combination of Physically Unclonable Functions (PUFs) with user-oriented biometrics employing fuzzy extractor-based procedures. The performance analysis depicts that our scheme offers resistance to known attacks using lightweight operations and supports 30% more security features than comparative studies. Our scheme is provably secure under Real-or-Random (ROR) formal security analysis model.

1. Introduction

The Mobile Edge Computing (MEC) has revolutionized the next generation sensing applications. The MEC architecture in combination with various sensors and internet of things (IoT) devices has facilitated a plethora of applications for smart homes, smart grid, health fitness, transportation, environment, agriculture and industry [1]. Most of the existing applications depend on centralized cloud data center for data collection, processing and disseminating the sensed information. This increasing reliance on backend creates a bottleneck on many fronts–for example, congestion and communication overhead, low real-time access, availability, bandwidth and processing, maintenance of bulk data storage, and high latency. The edge computing paradigm may well address the above concerns due to high proximity from the end users. The MEC server such as Radio Access Networks–(RAN) may be set between end user-based

☆ This paper is for special section VSI-cei. Reviews were conducted and processed by Guest Editor Dr. I. Razzak and recommended for publication.

* Corresponding author.
E-mail address: ashraf.shehzad.ch@gmail.com (S.A. Chaudhry).

https://doi.org/10.1016/j.compeleceng.2022.107692
Received 3 May 2021; Received in revised form 24 November 2021; Accepted 6 January 2022
Available online 12 January 2022
0045-7906/© 2022 Elsevier Ltd. All rights reserved.

edge devices/servers and cloud data center for monitoring and initial processing as. The mobile devices being deficient in computing may get the data processed on edge servers on real time basis, rather than remote cloud centers, and avail the benefits of high bandwidth and less latency [2]. For instance, the edge servers may dispense cache content for media services or may conduct preliminary processing of collected data between patient's bracelet and medical cloud center. In such a scenario, the privacy issues, nature of authorization or level of trust needs to be properly managed due to the distributed nature of network domains. The significance of MEC is ever increasing in 5G networks due to the low latency commitments.

The main concerns faced by distributed nature of MEC-based ecosystem are security and privacy issues. The communication information may be intercepted, blocked, replayed or tampered by adversaries on insecure channel. The mutual authentication can be ensured by establishing an Authenticated Key Agreement (AKA) between participating entities and construct session key before exchanging any critical information. Many MEC-based authentication protocols such as [3–4] employ costly bilinear pairing-based operations which are too costly operations on power deficient edge devices. There are many symmetric key encryption-based edge computing protocols, however these protocols do not provide anonymity to user, and suffer de-synchronization attacks.

Our key contribution is to design a novel two-factor authentication protocol for MEC with the combination of Physical Uncloneable Function (PUF) and fuzzy extractor-based functions. In our scheme, the device of user needs to be equipped with a PUF circuit to help in establishing a mutually agreed session between mobile user and MEC server [5]. The security solution can work without password, using only smart card and biometrics in registration or login procedures. We used formal security analysis under Real-or-Random (ROR) model to measure the security properties of the session key.

A. *Contribution*: The key contribution of the proposed study is given below:

1 We designed a novel two-factor authenticated key agreement protocol for MEC with the use of PUF and fuzzy-extractor-oriented functions.
2 The designed protocol ensures protection to stolen biometric factors.
3 There is no hassle of stealing password by the adversary during registration and logic phases, since our proposed model supports password free authentication.
4 Our contributed model, a lightweight symmetric key-based protocol, supports Perfect Forward Secrecy (PFS), anonymity, untraceability, and resistance to all known attacks.

B. Scheme's organization: The rest of our scheme is presented as follows: Section II describes literature review. Section III illustrates few preliminary details. Section IV depicts our contributed scheme. Section V analyzes the security on formal lines, and presents the informal discussion. Section VI evaluates the performance of contributed model with other protocols. The last section presents the summary of this scheme.

2. Literature review

We briefly discuss the literature review of authentication protocols in MEC setting as below: The Roman et al. [5] presents the analysis of security risks involved in mobile edge computing and fog computing architectures. Later, Mollah et al. [6] pointed security loopholes in edge computing, and presents the comparison on privacy issues in MEC. Thereafter, Irshad et al. [7] presented a bilinear pairing based authentication protocol for multiple servers in mobile cloud computing. Nevertheless, Xiong et al. [8] identified few attacks in [7], and put forward an enhanced scheme for MEC structure. The above schemes were computation-intensive due to costly crypto-operations. Kaur et al. [9] introduced a lightweight and efficient authentication scheme for resource-deficient MEC models. Later, Ke et al. [10] presented an efficient mobility oriented hierarchical edge computing model for low end IoT devices. Next, Tsai and Lo [11] demonstrated an authentication protocol for distributed edge structure. For MEC setting, recently, Jia et al. [12] presented an ID-based authentication protocol; however Li et al. [13] proved that it was found to be vulnerable for man-in-the-middle attack, and lacked perfect forward secrecy. Also, [13] presented an improved and efficient protocol. However, [13] has few design limitations, and cannot resist replay attack, tracing attack, and denial of service attack by the adversary. Later, Barman et al. [14] presented a multi-server authentication protocol using fuzzy extractor; however the scheme was vulnerable to stolen device attack and privileged insider attack. Then, Zhao et al. [15] designed a PUF-based authentication protocol for multi-server framework, however, the scheme was prone to de-synchronization and man-in-the-middle attacks. Wu et al. [16] presented an authentication protocol distributed cloud computing environment, but susceptible to privileged insider and stolen device attacks. Afterwards, Amin et al. [17] introduced a lightweight authentication scheme for IoT devices in distributed cloud infrastructure; nevertheless the scheme was defenseless against offline password guessing threat, and lacks anonymity.

It is evident from the above literature that none of MEC-based scheme provides most of the security features with efficiency, since most of the symmetric key schemes are vulnerable to attacks. Otherwise, if the schemes are secure in some way, these are too costly for being inducted in resource deficient pervasive and mobile edge computing paradigm.

3. Mathematical preliminaries

This section narrates few preliminary concepts that might help the readers to grasp the article.

A. *Fuzzy extractor*: Biometric authentication is a growing segment of the technology landscape around us. The biometric authentication employs distinct physical characteristics of user's traits for the identification of user's authenticity. The fuzzy extractor (*FE*) is

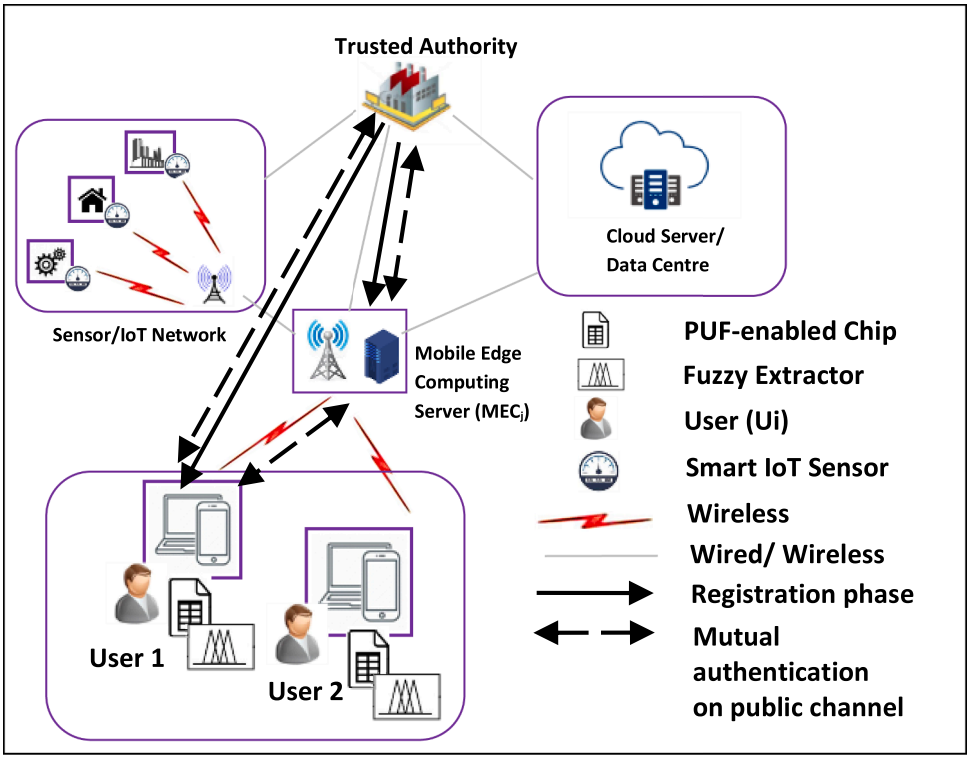

Fig. 1. System model.

used to eliminate the possible noise while capturing the biometric input. The *FE* (*d, δ*) comprises two algorithms such as $FE_{\mathscr{G}}()$ and $FE_{\mathscr{R}}(.)$. The $FE_{\mathscr{G}}(.)$ being a probabilistic key generation algorithm takes random bit string ϛ as input and gives output a key *K* as well as auxiliary data *ad*, i.e. $(K, ad) = FE_{\mathscr{G}}(ϛ)$. Similarly, the $FE_{\mathscr{R}}(.)$ being a deterministic reconstruction algorithm extracts the key *K* out of noisy input string ϛ' and the auxiliary data *ad*, i.e. $K=FE_{\mathscr{R}}(ϛ', ad)$, in case the hamming distance between strings ϛ and ϛ' is at most *d*. The fuzzy extractor supports authenticity in the achievement of robust cryptographic key, in case the minimum entropy for input is at least *δ*, and *K* is in proximity to the uniform random distribution for $\{0, 1\}^K$. This is because the frequent exposure of auxiliary data might result in additional loss of the minimum entropy; however the auxiliary data must remain secure during protocol execution.

B. *Physically uncloneable function*: A PUF is based on an integrated circuit (IC) that is characterized with Challenge-Response Pair (CRP). The underlying concept behind PUF is to use challenges that output incoherent responses, that is, in contrary to what is desired for having a reliable PUF [18]. Such meta-stable challenges output the responses that might vary impulsively. A PUF may be referred as a physical feature of a device; alternatively it is equivalent to biometric features of humans, such as Retina, fingerprint etc. The built-in features of a particular PUF can never be reproduced or cloned due to its physical signature or makeup. In mathematical terms, it is regarded as a function that takes an input (Challenge) in the form of bits' string and returns unique output (Response). The PUF-based function can be expressed as $R=PUF_i(C)$. Where, the variables *C* as well as *R* serve as the challenge and response pairs (CRP). A PUF always returns the same *R* for the same challenge, if tested again and again.

C. *Adversary model*: We evaluate the security properties of our scheme under widely known Canetti-Krawczyk Model (*CK*) adversary models [19]. The capabilities of adversary under these models are given below.

- A malicious attacker $\mathscr{A}$ may eavesdrop, delete, insert, modify, block or replay the intercepted messages communicated among the legal participants.
- $\mathscr{A}$ could steal the mobile device of a user and recover its stored contents using power differentiation analysis.
- $\mathscr{A}$ may initiate known threats including masquerading, spoofing, man-in-the-middle, and replay attack.
- $\mathscr{A}$ may compromise short term temporary secrets from the user's repository under *CK* model.

D. *System model*: In general, the edge computing paradigm is composed of many diverse network elements of distributed domains. These network elements may include mobile edge devices, edge servers, sensor nodes, cloud servers, and trusted authority. Our contributed model takes a subset of this edge computing paradigm and performs the mutual authentication between user devices and edge servers via trusted authority. Hence, our system model comprises mobile devices (U_i), MEC servers (MEC_j), and trusted authority (TA) as shown in Fig. 1. Needless to say, the MEC_j server is also connected on its backend with sensor and IoT devices as well as cloud computing servers. Our proposed model achieves two-factor oriented mutual authentication between MEC_j server and end user employing fuzzy-extractor as well as PUF_i on its end. The user devices are befitted with PUF_i along with biometric sensor. The trusted

Table 1
Symbols with definitions.

Symbols	Definition
TA:	Trusted Authority
U_i, MEC_j:	Mobile User, Mobile Edge Computing node
ID_i, MID_j:	Identities of U_i and MEC_j
K_T, K_u, K_{mi}:	Private secret keys of *TA*, U_i and MEC_j
PUF_i:	Physical Cloneable Functions for Ui
B_i	Fingerprint biometric impression
$FE_{\mathscr{G}}$ / $FE_{\mathscr{R}}$:	Fuzzy Extractor Generation and Reproduction
N_u, N_s:	Random nonces
$E_k()/D_k()$:	Symmetric encryption/decryption:
$\mathscr{A}$:	Adversary
SK:	Session key between U_i and S_j
$\oplus$, $\|\|$, *h()*	XOR, Concatenation, A secure one-way hash function

authority registers both end users as well as MEC_j servers on secure channel. Thereafter, both may be authenticated on insecure channel with the help of TA.

4. Proposed scheme

To gain the equivalent security properties of an authentication protocol employing costly public key-based crypto-primitives [20], many light-weight Symmetric Key-based Authentication protocols (SKA) have been presented. Those SKA schemes also employed biometrics to improve the security. However, these SKA schemes are yet unable to achieve an equivalent PFS, or resistance to de-synchronization attacks. To this end, for achieving the discussed security goals with light-weight operations, we propose a PUF-induced two-factor biometric authentication protocol authenticating a mobile user and IoT-based server. The proposed scheme strives to establish a session key agreement scheme without user's password, by engaging PUF as well as fuzzy extractor-oriented procedures for manipulating fingerprint-based biometrics. The system model comprises three entities, Trusted Authority (*TA*), mobile user (U_i), and Mobile Edge Computing node (MEC_j). In the initialization setup, the mobile users U_i chooses their private secret keys K_u. The *TA* selects its private key K_T, and a medium integer n_0 ($2^4 \leq n_0 \leq 2^8$). Our proposed scheme comprises two phases: 1) MEC_j registration phase 2) Ui's registration phase, and 3) login and mutual authentication phase. The details are given below:

A. *MEC_j's Registration phase:* In this phase, the TA registers MEC_j by choosing its identity MID_j and a random number x_i. Then it computes $K_{mi}=h(K_T||MID_j)$, and initializes hash frequency tag $= f_m = f_t = 0$. Then it stores $\{MID_j, K_{mi}, x_i, f_t\}$ safely and submits $\{MID_j, K_{mi}, x_i, f_m\}$ towards MEC_j. Some important notations used in this scheme are depicted in Table 1.

B. *U_i Registration phase*: In proposed scheme, the user U_i performs its registration steps with TA over a confidential channel as elaborated below.

1 The user U_i chooses its identity ID_i and inputs his/her fingerprint or thumb impression on mobile device. Next, the user recovers biometric template B_i of fingerprint, engenders two random integers r and e, and a random challenge Ch_i.
2 Next, the user calculates the PUF-based output $Z_i = PUF_i\ (Ch_i)$ and recovers the private secret key K_u as well as the auxiliary data FA by using biometric template B_i of fingerprint, i.e., $(K_u, FA) = FE_{\mathscr{G}}(B_i)$. Then, the user calculates $V = h(ID_i\ ||\ K_u)$, $Ch_i^* = Ch_i \oplus h(K_u)$, and $UID_i = ID_i \oplus Ch_i^* \oplus h(K_u\ ||\ r)$. In the end, the user submits $\{UID_i, (Ch_i^*, Z_i), V, Reg_{req}, Loc_i\}$ to TA over confidential channel, which includes the registration request Reg_{reg} as well, as shown in Fig. 2.
3 The TA verifies UID_i's uniqueness after receiving the Reg_{req} from user. Then, it chooses two random integers v, ζ and calculates $w = E_{KT}\ (v, UID_i)$ and $G_i = h(K_T||\ v) \oplus V$, and submits $\{G_i, w, \zeta, n_0\}$ to end user on secure channel. It also stores the parameters $\{UID_i, \zeta, <Ch_i^*, Z_i>\}$ in its repository.
4 After collecting the message G_i from server, the user calculates $W = h(ID_i\ ||\ K_u\ ||\ r)$, $R = h(ID_i\ ||\ e)\ mod\ n_0$, $r^* = r \oplus R$, $H_i = w \oplus h(r\ ||K_u)$, and $FA^* = h(ID_i\ ||\ r) \oplus FA$. Then, it stores $\{h(), G_i, H_i, W, r^*, FA^*, \zeta, e\}$ safely to finalize the registration.

C. *Login procedure*: In this phase, the user attempts for logging into the server by inserting few inputs including his/her identity ID_i, and imprinting fingerprint into the mobile device. The biometric imprinting outputs a biometric template B_i. Next, the following steps are performed for login into the device.

1 By employing the user's identity ID_i, the device computes $R^* = h(ID_i\ ||\ e)\ mod\ n_0$, $r = r^* \oplus R^*$, and recovers the auxiliary FA by computing $FA = h(ID_i||r) \oplus FA^*$. After recovering the secret key K_u by employing $K_u = FE_{\mathscr{R}}(B_i, FA)$, it further computes and verify the equality $W? = h(ID_i\ ||\ K_u\ ||\ r)$. If it does not match, the login phase is aborted. Or else, Ui proceeds for generating a random nonce N_u and timestamp T_u, and computes $V = h(ID_i||K_u)$ as well as $K = G_i \oplus V$ equivalent to $h(K_s||v)$.
2 Next, it calculates $M_1 = N_u \oplus K$, $UID_i = ID_i \oplus h(K_u\ ||\ r)$, $w = H_i \oplus h(r\ ||K_u)$, $M_2 = h(K\ ||\ UID_i||\ N_u\ ||\ w||\ T_u)$ and $UID_i^* = UID_i \oplus N_u$. Here, the parameter N_u is masked by employing the private secret K_u. Finally, the user submits the request message $\{w, UID_i^*, M_1, M_2, T_u\}$ towards TA for verification.

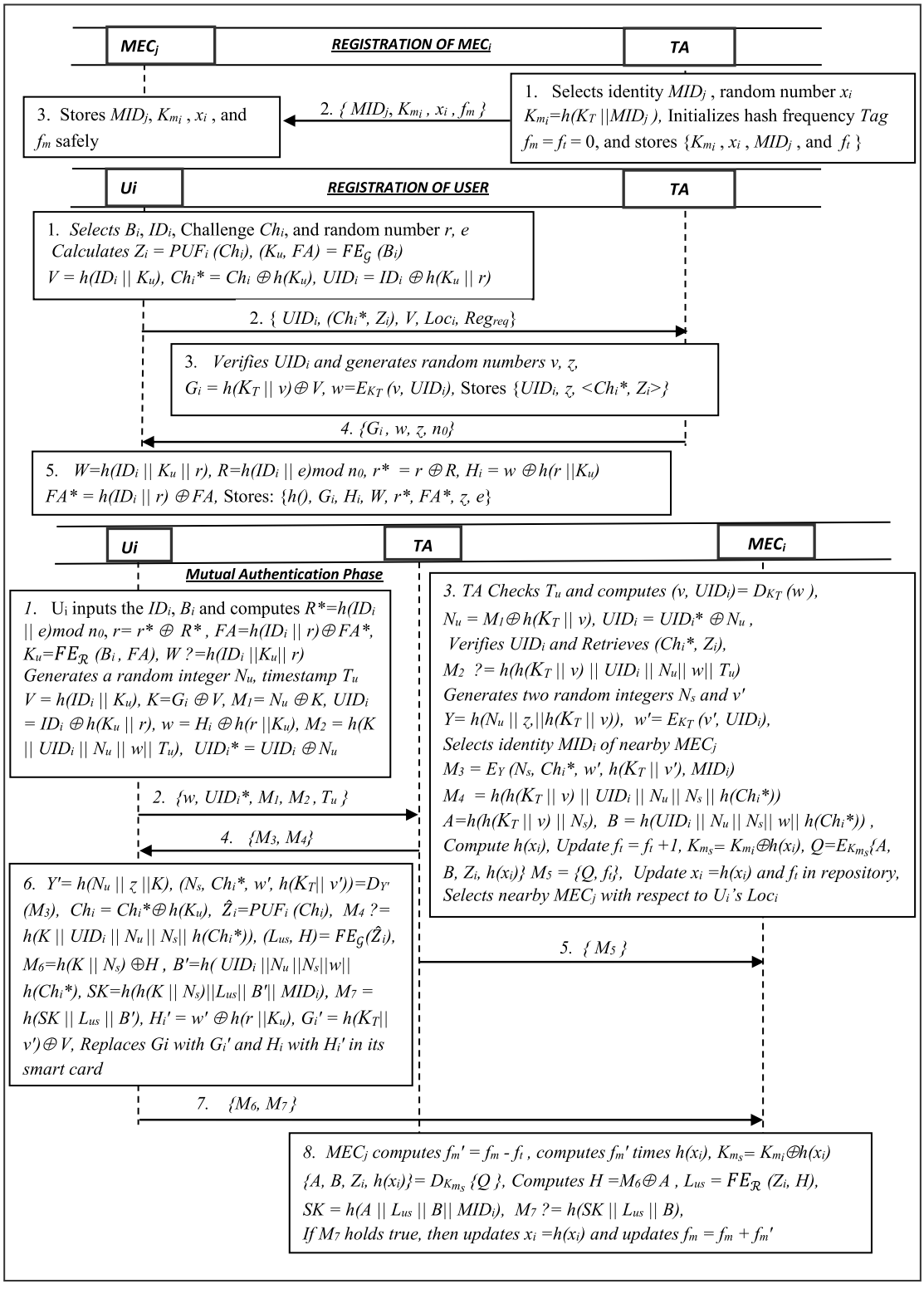

Fig. 2. Proposed model.

D. *Authentication and key agreement procedure:* In this phase, the *TA* after receiving the request *{w, UID_i*, M_1, M_2, T_u}* performs the following steps to complete the authentication and key agreement phase.

1 The TA initially verifies timestamp T_u and computes (v, UID_i) = D_{KT} (w) and recovers N_u after computing $N_u = M_1 \oplus h(K_T || v)$, and then further calculates $UID_i = UID_i^* \oplus N_u$. Next, after verifying the authenticity of UID_i, it retrieves the corresponding user's challenge-response pair (Ch_i*, Z_i) from its repository. Next, the TA computes$M_2$for verifying the validity of equality M_2 ?= $h(h(K_T|| v) || UID_i || N_u|| w|| T_u)$. If it is not true, the TA aborts the session. Onwards, it generates two random numbers N_s and v', and then computes Y= $h(N_u || z || h(K_T ||v))$ by employing the shared secret value. Next, it selects the identity MID_i of nearby MEC_j, and computes w'= E_{KT} (v',UID_i), M_3=E_Y(N_s, Ch_i*, w', $h(K_{mi}||v')$, MID_i) and M_4 = $h(h(K_T ||v)|| UID_i||N_u||N_s||h(Ch_i^*))$. Further, it computes A=$h(h(K_T ||v)||N_s)$ and B=$h(UID_i|| N_u || N_s|| w|| h(Ch_i^*))$. Next, TA computes $h(x_i)$ and updates f_t=f_t+1, and computes K_{ms}= $K_{mi} \oplus h(x_i)$, Q= E_{Kms}{A, B, Z_i, $h(x_i)$}, and M_5 = {Q, f_t}. Then, it updates x_i =$h(x_i)$ and f_t in repository. Next, it selects nearby MEC_j with respect to U_i's Loc_i. Finally, it submits {M_3, M_4} to user and {M_5} to MEC_j.

2 The user after receiving {M_3, M_4} computes Y'= $h(N_u|| z||K)$ and recovers (N_s, Ch_i*, w', $h(K_T|| v')$) bydecryptingM_3using Y' and verifies UID_i using Ch_i*.Then, it computes Ch_i=$Ch_i^* \oplus h(K_u)$, $\widehat{Z}_i$=$PUF_i(Ch_i)$ and verifies M_4 ?= $h(K||UID_i||N_u||N_s|| h(Ch_i^*))$. Then, it extracts L_{us} and auxiliary data H by employing fuzzy extractor $FE_{\mathscr{E}}(.)$, i.e. (L_{us}, H) = $FE_{\mathscr{E}}(\widehat{Z}_i)$. Then, it calculates M_6=$h(K || N_s) \oplus$H, B' = $h(UID_i|| N_u || N_s|| w|| h(Ch_i^*)$, the session key i.e., SK = $h(h(K||N_s)||L_{us}|| B'|| MID_i)$, and M_7 = h(SK || L_{us} || B'). Moreover, it computes G_i' = $h(K_T|| v')\oplus$ V, H_i'= w'$\oplus$ $h(r||K_u)$, and replaces G_i with G_i' and $H_i$as H_i' in smart card. Ultimately, it submits {M_6, M_7} to the MEC_j.

3 MEC_j, after receiving the messages M_5, M_6, and M_7, calculates f_m' = f_m - f_t, computes f_m' times $h(x_i)$, K_{ms}= $K_{mi} \oplus h(x_i)$, {A, B, Z_i, h (x_i)}=D_{Kms}{Q}. Next, it recovers auxiliary data as H=$M_6 \oplus$A, and L_{us} by using reconstruction procedure $FE_{\mathscr{R}}$ (.), i.e. L_{us} = $FE_{\mathscr{R}}$ (Z_i, H). Finally, it calculates session key as SK = $h(A||L_{us}||B||MID_i)$ and certifies M_7 ?= h(SK||L_{us}||B). If M_7 holds true, it certifies the computed session key, and updates x_i =$h(x_i)$ and f_m =f_m+f_m'; otherwise, the MEC_j aborts the session.

5. Security analysis

In formal analysis section, we analyze the security of contributed model on formal lines. We follow the uniformly accepted Real-or-Random (ROR) model [20] which is used to prove the session key security of the contributed security solutions. According to ROR model, an adversary must be able to differentiate an actual session key of the instance from a random key. There are three entities involved in the login and mutual authentication phases, which are user *U_i*, *TA*, and *MEC_j*. In addition, we analyze our scheme informally as well. We now describe ROR model in the following.

A. *Security model:* Participants: Let $\prod_{M_j}^{m}$ be the *m-th* instance of server *MEC_j*, $\prod_{U_i}^{u}$ be the *u-th* instance of user *U_i*, and $\prod_{TA}^{t}$ be the *t-th* instance of user *TA*, termed as oracles.

Collaboration: The collaborator for the instance $\prod_{U_i}^{u}$ for *U_i* is considered as the corresponding instance $\prod_{M_j}^{m}$ of *MEC_j* and vice-versa. The collaborator ID of $\prod_{M_j}^{m}$ is $pid_{U_i}^{u}$ for $\prod_{U_i}^{u}$. The partial transcript for the communicated messages between Ui and server *MEC_j* is unique, forming a session ID $sid_{U_i}^{u}$ between the same participants.

Freshness: The instance $\prod_{M_j}^{m}$ or $\prod_{U_i}^{u}$ is termed as fresh, provided the corresponding session key SK is not revealed to the adversary $\mathscr{A}$.

Adversary: Employing the ROR model, $\mathscr{A}$ can not only read messages in transmission, but also may alter, delete, or hold the parameters during the communication. Alternatively, $\mathscr{A}$ has full control over channel with an additional approach to the undermentioned queries:

- *Execute* ($\prod^{m}$, $\prod^{u}$): By using this query, the communication messages between legal entities Ui and MEC_j are eavesdropped by $\mathscr{A}$, for modeling an eavesdropping attack.
- *Send*($\prod^{m}$, *msg*): The *Send* query enables a participant instance for transmitting and receiving the message *msg* which is modeled as an active threat.
- *Corrupt_Device*($\prod_{U_i}^{u}$ *):* This query simulates the attack of stolen user's device. By employing this query, the crucial parameters could be revealed to the $\mathscr{A}$.
- *Reveal*($\prod^{m}$): This query could reveal the current session key to attacker as constructed between $\prod^{m}$ and its partner.

- *Test($\prod^{m}$):* The semantic security of *SK* as constructed between *Ui* and MEC_j concerning the indistinguishability of the ROR model [20], is implemented by using the *Test* query. Before the beginning of game, an unbiased coin *c* is flipped, while *A* keeps the result secret for taking the decision later on, regarding its output, i.e. it would be used to verify whether the *Test* query's output is consistent. If the session key is found to be fresh upon the execution of this query, the $\prod^{m}$ delivers *SK* if c=1, or it will return a random number, if c=0. Otherwise, it outputs null (⊥).

B. *Semantic security of SK:* Regarding the ROR security model, the challenge of $\mathscr{A}$ is to differentiate between the real session key *SK* as well as random secret key. $\mathscr{A}$ is permitted to issue many Test queries to either of the instances, i.e. $\prod_{M_j}^{m}$ or $\prod_{U_i}^{u}$. The Test query's outcome must correspond to random bit *c*. When the experiment ends, the adversary $\mathscr{A}$ judges the guessed bit *c'* with the purpose to win. *A* wins the game if the bits match, i.e. $c' = c$. The $\mathscr{A}$'s benefit for compromising the semantic security of contributed model $\prod$ in time *t* is characterized by $Adv_{\prod}^{Ake}(t) = |\,2.\,Pr\,[Sucs]\text{-}1\,|$, where the *Sucs* represents the event that adversary may win the game. The protocol $\prod$ stands secure in ROR model when the advantage $Adv_{\prod}^{Ake} \leq \lambda$ for any adequately small $\lambda > 0$.

Random Oracle: In this scheme, the participating entities and $\mathscr{A}$ approach collision-resistant cryptographic hash function as well as secure PUF, as simulated by the random oracles.

Definition1(*Hashing function*): The cryptographic hashing function $h:\{0,1\}^* \rightarrow \{0,1\}^n$, being deterministic one, produces a fixed length, say *n*-bit output string by taking the variable-sized input of binary string. If $Adv_{\mathscr{A}}^{h\text{-}fun}(\tau)$function represents $\mathscr{A}$'s advantage for finding the hash collision,

$Adv_{\mathscr{A}}^{h\text{-}fun}(\tau) = Pr[(I_1, I_2) \Leftarrow_R \mathscr{A} : I_1 \neq I_2 \text{ and } h(I_1) = h(I_2)]$. An (ξ, τ)-adversary compromising the *h(.)* function suggests that $Adv_{\mathscr{A}}^{h\text{-}fun}(\tau) \leq \xi$with at most running time τ.

Definition 2 (*Protected PUF*): The PUC-based IC takes the challenge as an input of string of bits, and provides an output response ρ as an arbitrary string of bits. The response ρ related to any PUF device PUF_i for challenge $\mathscr{C}$ may be symbolized as $\rho = PUF_i(\mathscr{C})$. Here, the PUF_i is regarded as (*d, m, l, δ, ϵ*)-secure if the under-mentioned requirements are met, that is:

(1) Assuming the two PUF-based devices $PUF_{i1}(.)$ and $PUF_{i2}(.)$, and $\mathscr{C}_1 \epsilon \{0,1\}^K$, $Pr[HD(PUF_{i1}(\mathscr{C}_1), PUF_{i2}(\mathscr{C}_2)) > d] \geq 1\text{-}\varepsilon$, where HD denotes Hamming distance.

(2) Considering a PUF $PUF_i(.)$ with any input $\mathscr{C}_1,\ldots, \mathscr{C}_m \epsilon \{0,1\}^K$, $Pr[\widehat{H}_{\infty}(PUF_{i1}(\mathscr{C}_i), PUF_{i2}(\mathscr{C}_j))_{1 \leq i,j \geq m, i \neq j} > \delta] \geq 1\text{-}\varepsilon$, which suggests that the minimum entropy for output of PUF, i.e. $\widehat{H}_{\infty}$must be greater than δ with higher probability, if the corresponding intra-distance, i.e. the distance between both responses of PUF out of same PUF instance and employing the identical challenge is less than *d*, while the corresponding inter-distance, i.e. the distance between both responses of PUF from diverging instances of PUF utilizing the identical challenge is larger than *d*.

C. *Security proof:* The theorem 1 sufficiently proves that the contributed scheme ensures session key-based security.

Theorem 1: If we presume the attacker $\mathscr{A}$ to be a probabilistic polynomial time (PPT) attacker executing in time *t* against the proposed protocol $\prod$ and *l* is number of bits in fingerprint-based biometric impression B_i. In that case, the advantage of adversary for compromising the semantic security of $\prod$ and recovering session key SK is calculated as:

$$Adv_{\Pi}^{AKE}(t) \leq \frac{q_h^2}{|hash|} + \frac{q_{Pf}^2}{|PUF|} + 2\left(C' \cdot q_{me}^{m'}, \frac{q_{me}}{2^l}\right),$$

Where q_h, q_{P_f}, q_{me} represent the number of hash, PUF, and send-queries, and $|hash|$, $|PUF|$ denote range space for hash function, and $P_f(.)$, respectively, while the parameters C' and m' be the Zipf's parameters [21].

Proof1: Following the proofs in [20], we define a sequence of five games symbolized as G_j, where $[0 \leq j \leq 4]$ to prove the session key's security of contributed scheme. Let $Sucs_j$ represent an event wherein the attacker may guess the bit *c* in G_j effectively. The detailed explanation of these games is given in the following.

Game G_0: This game is deemed to be a real attack by $\mathscr{A}$ against our authenticated key exchange (AKE) scheme $\prod$ in ROR security model. Given that, the bit *c* must be selected in the beginning of G_0, it is quite evident that

$$Adv_{\Pi}^{AKE}(t) = |2.\Pr[Sucs_0] - 1| \tag{1}$$

Game G_1: The game G_1 is translated from G_0 by modeling $\mathscr{A}$'s eavesdropping attack by invoking the *Execute* ($\prod^{t}$, $\prod^{r}$) oracle query. Thereafter, $\mathscr{A}$ requires to query the Test oracle for verifying the difference of factual session key *SK* from a random integer. The session

key SK in contributed scheme is evaluated as $SK=h(h(K \,||\, N_s)||\, L_{us} \,||\, B'||\, MID_i)$ between U_i and MEC_j. It is computed by employing $h(K \,||\, N_s)$, L_{us}, B', and MID_i factors. However, the eavesdropping of $\{w, UID_i^*, M_1, M_2, T_u, M_3, M_4, M_5, M_6, M_7\}$ parameters does not help the adversary in computing the SK's parameters $h(K \,||\, N_s)$, L_{us}, B', and MID_i. Since, the calculation of those parameters requires the exposure of long term private secret keys, i.e. K_T and K_u as well as the compromise of PUF_i held by the users. Hence, the probability regarding wining G_1 through eavesdropping of messages is not increased. Then, it follows:

$$\Pr[Sucs_0] = \Pr[Sucs_1] \tag{2}$$

Game G_2: The game $G_1$is translated to G_2 by including the simulations of *Send* as well as *hash* oracle queries. In this way, it may be regarded as an active attack while the adversary might strive to deceive a legal entity into accepting a fictitious and modified content. The attacker is allowed to issue multiple *Hash* oracle queries for monitoring the hash-based collisions. It is noteworthy that all publicly exchanged messages in mutual authentication phase involve the entity's identity, randomly defined nonces, and high entropy long term secrets. Hence, no occurrence of collision is found if $\mathscr{A}$ issues *Send* oracle queries. By the application of results from birthday paradox, we get:

$$|\Pr[Sucs_2] - \Pr[Sucs_1]| \leq \frac{q_h^2}{2|\text{ hash }|} \tag{3}$$

Game G_3: The G_3 is translated from G_2 by adding the simulations of *Send* as well as PUF oracle queries. Hence, following the similar argument given in G_2, owing to secure PUF function (*Ref*. Definition 2), we get:

$$|\Pr[Sucs_3] - \Pr[Sucs_2]| \leq \frac{q_{P_f}^2}{2|PUF|} \tag{4}$$

Game G_4: In the last game, the simulation of *Corrupt_Device* is included. So, $\mathscr{A}$ might recover stored information $\{h(), G_i, H_i, W, r^*, FA^*, \zeta\}$ in Ui's device. However, $\mathscr{A}$ cannot extract either identity or the private secret key K_u which is protected under fuzzy extractor-based fingerprint $B_i \in \{0, 1\}^l$. With the application of PUF, the probability of guessing fingerprint impression B_i is $\frac{1}{2^l}$ [20]. There is no password involved in the registration phase, and hence no possibility of guessing a password. Although, the Ui's identity ID_i may be guessed for being low-entropy string, it is concatenated with private key K_u under collision resistant property of hash. Hence, it follows

$$|\Pr[Sucs_4] - \Pr[Sucs_3]| \leq \left(C' \cdot q_{me'}^{m'}, \frac{q_{me}}{2^l}\right) \tag{5}$$

All of the queries are employed by $\mathscr{A}$, the last chance of winning the game is mere random guessing the bit c by executing Test oracle query. Therefore, we get

$$\Pr[Sucs_4] = \frac{1}{2} \tag{6}$$

According to (1), (2) as well as (6), we get:

$$\begin{aligned} \frac{1}{2} \cdot Adv_{\Pi}^{AKE}(t) &= \left|\Pr[Sucs_0] - \frac{1}{2}\right| \\ &= \left|\Pr[Sucs_0] - \frac{1}{2}\right| \\ &= \left|\Pr[Sucs_1] - \frac{1}{2}\right| \\ &= |\Pr[Sucs_1] - \Pr[Sucs_4]| \end{aligned} \tag{7}$$

By employing the triangular inequality and Eqs. (3), (4), as well as (5), we get the under-mentioned result:

$$\begin{aligned} &|\Pr[Sucs_1] - \Pr[Sucs_4]| \leq |\Pr[Sucs_1] - \Pr[Sucs_3]| + |\Pr[Sucs_3] - \Pr[Sucs_4]| \\ &\leq |\Pr[Sucs_1] - \Pr[Sucs_2]| + |\Pr[Sucs_2] - \Pr[Sucs_3]| + |\Pr[Sucs_3] - \Pr[Sucs_4]| \\ &\leq \frac{q_h^2}{2|hash|} + \frac{q_P^2}{2|PUF|} + \left(C' \cdot q_{me}^{m}, \frac{q_{me}}{2^l}\right) \end{aligned} \tag{8}$$

Ultimately, after solving Eqs. (7) and (8), we get to the Eq. (9) as a result.

$$Adv_{\Pi}^{AKE}(t) \leq \frac{q_h^2}{2|hash|} + \frac{q_{P_f}^2}{2|PUF|} + \left(C' \cdot q_{me}^{m'}, \frac{q_{me}}{2^l}\right) \tag{9}$$

D. *Informal analysis*: The proposed scheme attains all security stipulations for edge computing infrastructure as elaborated below:

1 Supports mutual authentication: In our scheme both participants U_i and MEC_j mutually authenticate each other with the help of TA. The U_i authenticates MEC_j on the basis of $M_4 \stackrel{?}{=} h(K \,||\, UID_i \,||\, N_u \,||\, N_s||\, h(Ch_i^*))$, it knows that the key $K \approx h(K_T||v)$ is only shared with TA, who is authenticating the MEC_j server. The identity MID_j for MEC_j is recovered from M_3, and included in session key.

```
-- Query not attacker(sk[])
Completing...
Starting query not attacker(sk[])
RESULT not attacker(sk[]) is true.
-- Query inj-event(endMECj(idi)) ==> inj-event(beginMECj(idi))
Completing...
Starting inj-event(endMECj(idi)) --> inj-event(beginMECj(idi))
RESULT inj-event(endMECj(idi)) --> inj-event(beginMECj(idi)) is true.
-- Query inj-event (endUi(idi 1527)) --> inj-event(beginUi(idi 1527)) Completing...
Starting query inj-event(endUi(idi1527)) ==> inj-event(beginUi (idi 1527))
RESULT inj-event(endUi(idi_1527)) ==> inj-event(beginUi (idi_1527)) is true.
-- Query inj-event (endTA(idi 132)) --> inj-event(beginTA(idi 132)) Completing...
Starting query inj-event(endTA(idi 132)) ==> inj-event(beginTA (idi 132))
RESULT inj-event(endTA(idi_132)) ==> inj-event(beginTA (idi_132)) is true.
```

Fig. 3. ProVerif results.

Similarly, MID_j also authenticates U_i on the basis of M_5, and subsequent verification of $M_7 ?= h(SK \,||\, L_{us} \,||\, B)$ as received from U_i. The MID_j authenticates both entities during verification of M_7.

2 Resists impersonation and replay attacks: If an adversary attempts to replay, modify or impersonate any legitimate entity by intercepting the $\{w, UID_i^*, M_1, M_2, T_u\}$, $\{M_3, M_4\}$, $\{M_5\}$, $\{M_6, M_7\}$ messages, it may not be able to initiate these attacks. If the first message is replayed or maliciously manipulated on its way towards *TA*, it is traced during the verification of $M_2 ?= h(h(K_T \,||\, v) \,||\, MID_j \,||UID_i||\, N_u||\, w||\, T_u)$ with fresh timestamp T_u. The messages $\{M_3, M_4\}$ are verified using $M_4 ?= h(K \,||\, UID_i \,||\, N_u \,||\, N_s||\, h(Ch_i^*))$. The MEC_j authenticates U_i and *TA* on the basis of $\{M_5\}$ and $\{M_6, M_7\}$ by performing $M_7 ?= h(SK \,||\, L_{us} \,||\, B)$.

3 Supports forward secrecy: If the private secret key, i.e. either K_u (U_i) or K_T (TA) or K_{mi} (MEC_j) is leaked to the adversary, then the latter may not be able to compute previous session key i.e. $SK = h(A||L_{us}||B||MID_i)$ and $SK = h(h(K||N_s)||L_{us}||\, B'||\, MID_i)$ as constructed between U_i and MEC_j [22]. This is because, the adversary requires PUF_i along with access to K_u, or *TA*'s repository in case the K_T is leaked to the adversary. Similarly, if K_{mi} is exposed, then the adversary must require access to previous x_i parameter to decrypt the *TA*'s messages, the MEC_j takes the hash of previous x_i and replaces the old factor with the newly updated parameter, upon successful establishment of session. In this manner the proposed scheme complies with the property of perfect forward secrecy even if the high entropy private secret keys are exposed to the attacker.

4 Resists de-synchronization attack: Most of symmetric key-based schemes suffer from de-synchronization attacks [23]; however, our scheme is resistant of this attack. In case, $\mathscr{A}$ blocks any of the messages on insecure channel, then the former may not de-synchronize the communication. We employ a pseudonym m_w for U_i and TA's synchronization which is updated in each session and is also not stored in TA's repository.

5 Resists DoS attack: Our scheme employs PUF_i function, however, we use fuzzy extractor to reduce the noise from the output of PUF_i and utilize it later on, which nullifies the chances of denial of service on the part of logging into the device. Moreover, no adversary may exploit the maintained repository in TA, this is because the TA may compute search the intended pseudo-identity in at most $O(1)$ complexity [24].

6 Resists ephemeral information leakage threat: Our scheme is immune to ephemeral information leakage threat, since in case the adversary is able to access the short term secrets of user such as N_u or N_s, the former must requires access to PUF_i in addition to K, ζ and Z_i parameters for computing Y'. The parameter Y' can only be used to recover the legitimate Ch_i by decrypting M_3 message as received from the TA. Then the PUF_i outputs Z_i upon the input of Ch_i. Now this Z_i is passed through fuzzy extractor which outputs L_{us} factor which ultimately enables to compute the legitimate session keys $SK = h(A||\, L_{us} \,||\, B||\, MID_i)$ as constructed in the past session. However this computation is dependent upon the access to PUF_i as well other crucial parameter which is a hard assumption for the adversary to have all those pre-requisites simultaneously in hand to initiate the attack.

7 Resists key compromise impersonation (KCI) threat If either short term or long term secret key of user is compromised to the adversary, then the later also requires access to $h(Ch_i^*)$ and ζ parameters to construct a valid or verifiable message $\{M_3, M_4\}$, i.e. $M_3 = E_Y(N_s, Ch_i^*, w', h(K_T \,||\, v'), MID_i)$ and $M_4 = h(h(K_T \,||\, v) \,||\, UID_i||\, N_u \,||\, N_s \,||\, h(Ch_i^*))$, to impersonate as a server. Hence, our scheme is resistant to KCI attack.

E. *Security verification using proverif:* We used ProVerif tool [25] for automated analysis in order to validate the security aspects of proposed protocol such as the confidentiality of session key as well as mutual authentication under Canetti-Krawczyk (CK) Model. By using the strong features of π calculus, it can support hash function, digital signatures, as well as public key encryption-based complex primitives. In order to demonstrate the simulation for system model, we modeled three events including Ui, MECj and TA, in order to simulate the system model. For the purpose, the events beginTA(bitstring) and endTA(bitstring) initialize the other events related to Ui and MECj by registering these processes. Then, the events beginUi(bitstring) and endUi(bitstring) are employed by Ui for authenticating MECj. Similarly, the events beginMECj(bitstring) and endMECj(bitstring) are modeled by MECj for authenticating Ui. After the computation of query results, it is evaluated that the order of the three pairs of events remains stable. The results in Fig. 3 portray that the contributed proposed model attains mutual authenticity by establishing an agreed session key among the three processes, i.e. Ui, MECj and TA.

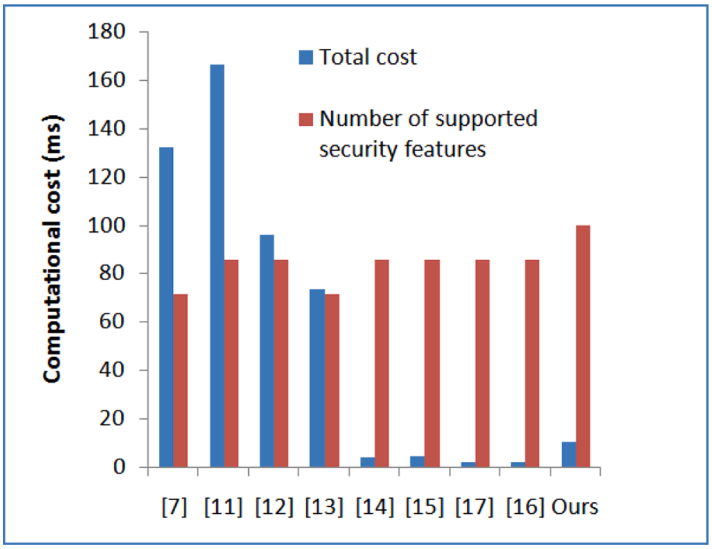

Fig. 4. Computational cost.

Table 2
Experimental cost of primitives.

Operations	*Ui*	MEC_j
T_h	0.029	0.009
T_{SYM}	0.062	0.019
T_{PUF}	0.145	0.68
T_{FEG}/ T_{FER}	3.67	2.06
T_{ME}	12.42	5.78
T_{EPM}	10.92	5.16
T_{PA}	0.065	0.031
T_{BP}	26.68	13.77

6. Efficiency analysis and discussion

This section evaluates the performance of recent contemporary protocols in edge computing and mobile cloud computing frameworks [7,11–16] against the proposed model in terms of computational and communicational latencies. The cost of registration phase is omitted during the comparison of computational overheads of various schemes in Table 4; this is because the registration phase is executed only once, while the mutual authentication phase is performed on frequent basis. The comparison of computational costs for various schemes is demonstrated in Fig. 4. The Table 3 presents the comparative analysis on security features for proposed and related schemes. This table depicts that the scheme [11] is vulnerable to impersonation attack, and also does not fulfill anonymity as well as perfect forward secrecy. The scheme [7] is prone to denial of service attack and fails to prove perfect forward secrecy despite the use of bilinear pairing operations. The scheme [12] does not offer resistance to man-in-the-middle attack (MIDM), while [13] is prone to replay attack, impersonation, MIDM and temporary information leakage attack. The above mentioned schemes employ public key cryptographic operations for edge computing paradigm, however suffering many attacks other than employing costly operations. Later, few symmetric key based cloud computing schemes [9,16] had been presented for edge framework. In which, the scheme [9] is susceptible to Stolen Device Attack (SDA) and privileged insider attack. The scheme [13] is found to be defenseless against de-synchronization and MIDM attack. Amin et al. [17] does not provide anonymity to the user and is also vulnerable to SDA. Likewise, Wu et al. [16] is prone to SDA, and privileged insider attack.

In order to evaluate the experimental overhead of computation for various crypto-primitives we performed simulation on the user's end by using a Smartphone (Lenovo Zuk Z1) comprising Quad-core 2.6 Ghz-Processor, 6GB RAM with Android OS V5.1.2, and on the MEC_j's end by using PC (HP-E8300-Core i5), 2.93 Ghz-processor bearing 6GB RAM with Ubuntu 16.12 OS). The JCE library [17] was installed to evaluate the execution latency of all primitive operations as employed in the proposed scheme. Besides, the 128-bit arbiter PUF is engaged in the execution of PUF operation, and the BCH code is used to simulate the generation $FE_{\mathscr{G}}$ (.) and reproduction $FE_{\mathscr{R}}$ (.) procedures of fuzzy extractor. The Table 2 lists the experimental cryptographic cost of various primitives such as PUF i.e. T_{PUF}, the elliptic curve point multiplication (ECC) i.e. T_{EPM}, modular exponentiation i.e. T_{ME}, hashing function i.e. T_h, the symmetric encryption or decryption i.e. T_{SYM}, fuzzy extractor-based generation and reproduction function i.e. T_{FEG} and T_{FER}, bilinear pairing operation i.e. T_{BP}, inverse operation i.e. T_{INV}, and point addition i.e. T_{PA}. The computational costs of inverse operation and Exclusive-OR are supposed to be negligible. According to Table 4, the proposed scheme bears $1T_{FEG}+1T_{FER}+1T_{PUF}+1T_{SYM}+14T_H$ crypto-primitives at

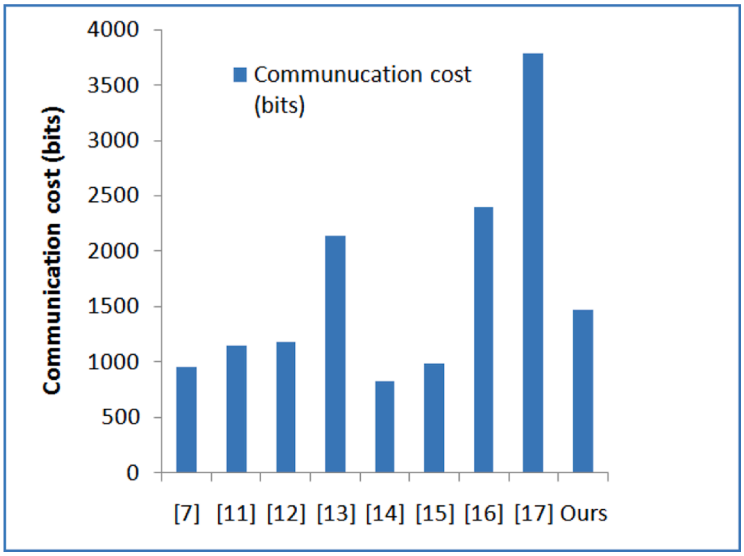

Fig. 5. Communication cost.

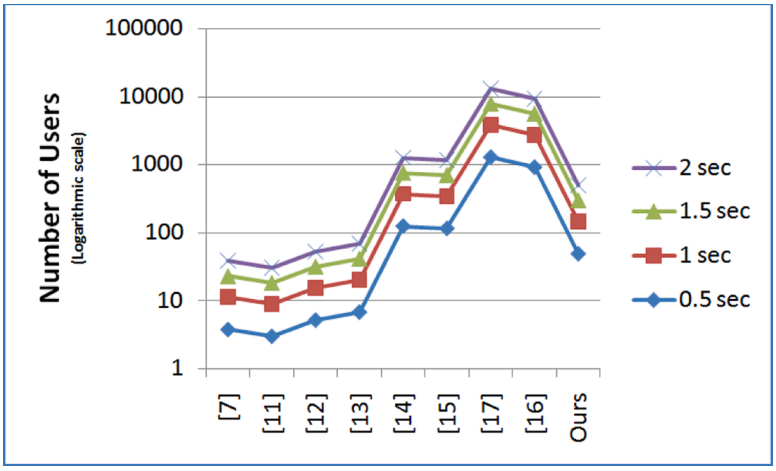

Fig. 6. Number of users authenticated.

user's end, and $1T_{FER} + 4T_{SYM} + 9T_H$ on the side of MEC_j. On average, our scheme takes 10.17 ms to complete the mutual authentication phase excluding communication cost. The other comparative schemes [5–12] incur 132.30ms, 166.28ms, 95.9ms, 73.44ms, 3.994ms, 4.32ms, 0.544ms, and 0.387ms respectively. The [5–8] bear either bilinear pairing, or elliptic curve point multiplication, or modular exponentiation operations. The other schemes [9,16] are based on symmetric key operations, although with many security limitations. To design a secure, yet efficient protocol we suggested a symmetric key operations-based protocol engaging hash, PUF, XOR, and fuzzy extractor operations. The cost of fuzzy extractor operations is higher than hash and symmetric key operations, but less than pairing, ECC or exponentiation operations. Our scheme is not only efficient but also supports strong security features including perfect forward secrecy, anonymity, untraceability, resistance to de-synchronization and denial of service attacks. The Fig. 4 signifies that our scheme takes less computational cost and bears more security features, unlike other schemes.

To compute the communication costs we assume that the timestamp and identity strings take 32 bits, hashing digests take 160-bit, point multiplication pairs take 320 bits. The Table 5 and Fig. 5 shows that our scheme bears 1472 bits communication cost, while the rest of schemes [7,11–16] take 960 bits, 1152 bits,1184 bits, 2144 bits, 832 bits, 992 bits, 3786 bits, 2400 bits respectively. Although, our scheme bears more communication cost than [7,11–14], our scheme is secure that those schemes as depicted from the Figs. 4–6. The Fig. 6 depicts the number of users being authenticated in different time variants, which suggests that our scheme authenticates less number of users than [16] and [15], but it is more secure than those schemes. Likewise, our scheme authenticates more number of users than [7,11–13] with at least same or more security features. In addition, it is evident from Table 3 and Fig. 4 that the proposed scheme supports 30% more security features than the comparative studies.

Table 3
Comparison of security functionalities.

	[7]	[11]	[12]	[13]	[14]	[15]	[16]	[17]	Ours
F1	×	✓	✓	✓	✓	✓	×	✓	✓
F2	✓	✓	✓	✓	✓	✓	×	✓	✓
F3	✓	✓	✓	✓	×	✓	✓	×	✓
F4	✓	✓	✓	✓	×	✓	✓	×	✓
F5	✓	✓	✓	×	✓	✓	✓	✓	✓
F6	×	✓	✓	×	✓	✓	✓	✓	✓
F7	✓	✓	×	×	✓	×	✓	✓	✓
F8	✓	✓	✓	×	✓	✓	✓	✓	✓
F9	×	×	✓	✓	✓	✓	✓	✓	✓
F10	✓	✓	✓	×	×	✓	✓	✓	✓
F11	✓	×	✓	✓	✓	✓	✓	✓	✓
F12	✓	✓	✓	✓	✓	×	✓	✓	✓
F13	×	×	×	×	✓	✓	✓	✓	✓

F1: Supports Anonymity and untraceability, F2: Login viability without password, F3: Resist stolen device attack, F4: Resist insider attack, F5: Resist replay attack, F6: Resist impersonation attack, F7: Resist Man-in-the-middle attack, F8: Resist temporary information leakage attack, F9: Supports perfect forward secrecy, F10: Supports mutual authentication,F11: Resist Denial of service attack,F12: Resist De-synchronization attack, F13: Supports efficient symmetric key operations, ✓: The property is satisfied, ×: Property is not satisfied.

Table 4
Computational costs.

Scheme	*Mutual Authentication phase* U_i	MEC_j+TA	*Total* (U_i+MEC_j+ TA)	*Latency (ms)*
[7]	$5T_{EPM}$ +$2T_{PA}$+ $1T_{ME}$+$1T_{INV}$+$5T_H$	$5T_{EPM}$ +$2T_{BP}$+ $2T_{PA}$+ $2T_{ME}$+ $5T_H$	$10T_{EPM}$ +$2T_{BP}$+ $4T_{PA}$ + $3T_{ME}$+$10T_H$ +$1T_{INV}$	≈ 132.302
[11]	$5T_{EPM}$ +$1T_{BP}$+ $2T_{PA}$+ $2T_{ME}$+ $6T_H$+$1T_{INV}$	$4T_{EPM}$+$2T_{BP}$+ $3T_{PA}$+ $2T_{ME}$+ $3T_H$	$9T_{EPM}$ +$3T_{BP}$+ $5T_{PA}$ +$4T_{ME}$+$9T_H$ +$1T_{INV}$	≈ 166.284
[12]	$4T_{EPM}$ + $1T_{ME}$+$5T_H$	$5T_{EPM}$ +$1T_{BP}$+ $3T_{PA}$+ $5T_H$	$9T_{EPM}$ +$1T_{BP}$+ $3T_{PA}$+ $1T_{ME}$ + $10T_H$	≈ 95.953
[13]	$4T_{EPM}$ + $4T_{PA}$ +$5T_H$	$3T_{EPM}$ +$1T_{BP}$ + $3T_{PA}$ +$2T_H$	$7T_{EPM}$ +$1T_{BP}$ + $7T_{PA}$ +$7T_H$	≈ 73.446
[14]	$1T_{FEG}$ +$9T_H$	$7T_H$	$1T_{FEG}$ +$16T_H$	≈ 3.994
[15]	$1T_{FEG}$+$1T_{PUF}$ +$14T_H$	$11T_H$	$1T_{FEG}$+$1T_{PUF}$ +$25T_H$	≈ 4.32
[16]	$9T_H$	$14T_H$	$23T_H$	≈ 0.387
[17]	$11T_H$	$25T_H$	$36T_H$	≈ 0.544
Ours	$1T_{FEG}$+$1T_{FER}$ + $1T_{PUF}$+ $1T_{SYM}$+$14T_H$	$1T_{FER}$ +$4T_{SYM}$+$9T_H$	$1T_{FEG}$+$2T_{FER}$ +$1T_{PUF}$+ $5T_{SYM}$+$23T_H$	≈ 10.17

Table 5
Communicational overhead (bits).

Schemes	[7]	[11]	[12]	[13]	[14]	[15]	[16]	[17]	Ours
Comm. cost	960	1152	1184	2144	832	992	2400	3786	1472

7. Conclusion

The security of edge paradigm with respect to internet of things and cloud computing is still in its infancy. In this paper, we proposed a novel two-factor biometric authenticated key agreement scheme for mobile edge computing that employs PUF and fuzzy extractor operations to strengthen a protocol based on symmetric key operations. Our scheme achieves mutual authentication, forward secrecy, anonymity and untraceability which are missing features in most of the symmetric key-based protocols. We proved the robustness of security features with formal analysis, and the experimental results also depict that our scheme is efficient, nevertheless maintaining strong security features. Eqs. (1), (2) and(6)

Azeem Irshad received his PhD degree from International Islamic University, Islamabad, Pakistan. He has authored more than 75 international journal and conference publications, including 37 SCI-E journal publications. His research work has been cited over 958 times with 15 h-index and 22 i-10-index. He received Top Peer-Reviewer Award from Publons in 2018 by serving more than 60 reputed international journals.

Shehzad Ashraf Chaudhry is working as an Associate Professor with the Department of Computer Engineering, Faculty of Engineering and Architecture, Istanbul Gelisim University, Istanbul, Turkey. He has authored over 140 scientific publications and with an H-index of 32 and an I-10 index 67, his work has been cited over 3000 times. He has won several awards for his cutting-edge research.

Anwar Ghani is a faculty member at the Department of Computer Science & Software Engineering, International Islamic University Islamabad. He received his Doctorate in Computer Science and MS Computer Science from the Department of Computer Science & Software Engineering, International Islamic University Islamabad in 2016 and 2011. His broad research interests include wireless sensor networks, Information Security, and IoT.

Ghulam Ali Mallah is Professor of Computer Science at SALU Khairpur Mirs. He is a member of various professional forums

including IEEE & ACM. He has about 50 national & international research articles at his credit. He has supervised many MS & PhD students. He is the winner of 04 HEC-Funded R&D projects and ICT Excellence Award.

Muhammad Bilal was a Postdoctoral Research Fellow at Smart Quantum Communication Center, Korea University. Currently, he is an Assistant Professor with the Division of Computer and Electronic Systems Engineering, Hankuk University of Foreign Studies, South Korea. His research interests include design and analysis of network protocols, network architecture, network security, IoT, named data networking, Blockchain, cryptology, and future Internet.

B.A. ALZAHRANI received Ph.D. in Computer Science from University of Essex, UK, in 2015. He is an Associate Professor in the Faculty of Computing and Information Technology, King Abdulaziz University, KSA. He has led 10+ national R&D projects and co-authored 55+ research articles in peer reviewed journals and conferences. His current research interests include WSN, security of ICN/SDN, secure content-routing.

CRediT authorship contribution statement

Azeem Irshad: Writing – original draft, Writing – review & editing. **Shehzad Ashraf Chaudhry:** Conceptualization, Validation, Formal analysis. **Anwar Ghani:** Supervision, Writing – review & editing. **Ghulam Ali Mallah:** Writing – review & editing, Formal analysis. **Muhammad Bilal:** Validation. **Bander A. Alzahrani:** Visualization, Methodology, Investigation, Validation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Abbas N, Yan Z, Taherkordi A, Skeie T. Mobile edge computing: a survey. IEEE Int Things J 2018;5(1):450–65.
[2] Tran TX, Hajisami A, Pandey P, Pompili D. Collaborative mobile edge computing in 5G networks: new paradigms, scenarios,and challenges. IEEE Commun Mag 2017;55(4):54–61.
[3] Shahidinejad A, Ghobaei-Arani M, Souri A, Shojafar M, Kumari S. Light-edge: a lightweight authentication protocol for IoT devices in an edge-cloud environment. IEEE Consum Electron Mag 2021. https://doi.org/10.1109/MCE.2021.3053543.
[4] Cheng G, Chen Y, Deng S, Gao H, Yin J. A Blockchain-based mutual authentication scheme for collaborative edge computing. IEEE Trans Comput Soc Syst 2021. https://doi.org/10.1109/TCSS.2021.3056540.
[5] Roman R, Lopez J, Mambo M. Mobile edge computing, fog et al.: a survey and analysis of security threats and challenges. Future Gener Comput Syst 2018;78:680–98.
[6] Mollah MB, Azad MAK, Vasilakos A. Security and privacy challenges in mobile cloud computing: Survey and way ahead. J Netw Comput Appl 2017;84:38–54.
[7] Irshad A, Sher M, Ahmad HF, Alzahrani BA, Chaudhry SA. An improved multi-server authentication scheme for distributed mobile cloud computing services. KSII TIIS 2016;10(12):5529–52.
[8] Xiong L, Peng D, Peng T, Liang H. An enhanced privacy-aware authentication scheme for distributed mobile cloud computing services. KSII Trans Int Inf Syst 2017;11(12):6169–87.
[9] Kaur K, Garg S, Kaddoum G, Guizani M, Jayakody D. A lightweight and privacy-preserving authentication protocol for mobileedge computing. In: Proceedings of the IEEE global communication conference; 2019. p. 1–6.
[10] Ke Z, Leng S, He Y, Maharjan S, Yan Z. Mobile edge computing and networking for green and low-latency internet of things. IEEE Commun Mag 2018;56(5):39–45.
[11] Tsai JL, Lo NW. A privacy-aware authentication scheme for distributed mobile cloud computing services. IEEE Syst J 2015;9(3):805–15.
[12] Jia X, He D, Kumar N, Choo KKR. A provably secure and efficient identity-based anonymous authentication scheme for mobile edge computing. IEEE Syst J 2019;14(1):560–71.
[13] Li Y, Cheng Q, Liu X, Li X. A Secure Anonymous Identity-Based Scheme in New Authentication Architecture for Mobile Edge Computing. IEEE Syst J 2020;15(1):935–46.
[14] Barman S, Das AK, Samanta D, Chattopadhyay S, Rodrigues J, Park Y. 'Provably secure multi-server authentication protocol using fuzzy commitment. IEEE Access 2018;6:38578–94.
[15] Zhao J, Bian W, Xu D, Jie B, Ding X, Zhou W, Zhang H. A secure biometrics and PUFs-based authentication scheme with key agreement for multi-server environments. IEEE Access 2020;8:45292–303.
[16] Wu F, Li X, Xu L, Sangaiah AK, Rodrigues JJ. 'Authentication protocol for distributed cloud computing: an explanation of the security situations for internet-of-things-enabled devices. IEEE Consum Electron Mag 2018;7(6):38–44.
[17] Amin R, Kumar N, Biswas G, Iqbal R, Chang V. A light weight authentication protocol for IoT-enabled devices in distributed cloud computing environment. Future Gener Comput Syst 2018;78:1005–19.
[18] Chatterjee U, Govindan V, Sadhukhan R, Mukhopadhyay D, Chakraborty RS, Mahata D, Prabhu MM. Building PUF based authentication and key exchange protocol for IoT without explicit CRPs in verifier database. IEEE Trans Dependable Secur Comput 2018;16(3):424–37.
[19] Canetti R, Krawczyk H. Analysis of key-exchange protocols and their use for building secure channels. In: Proceedings of the 22nd International conference on theory and application of cryptology technology. Innsbruck, Austria: Springer; 2001. p. 453–74.
[20] M. Abdalla, P. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting,'' in Proc. IWPKC, Les Diablerets, Switzerland, vol. 3386, pp. 65–84, 2005.
[21] Wang D, Cheng H, Wang P, Huang X, Jian G. 'Zipf's law in passwords. IEEE Trans Inf Forensics Secur 2017;12(11):2776–91.
[22] Shabbir M, Shabbir A, Iwendi C, Javed AR, Rizwan M, Herencsar N, Lin JCW. Enhancing security of health information using modular encryption standard in mobile cloud computing. IEEE Access 2021;9:8820–34.
[23] Irshad A, Ahmad HF, Alzahrani BA, Sher M, Chaudhry SA. An efficient and anonymous chaotic map based authenticated key agreement for multi-server architecture. KSII Trans Internet Inf Syst (TIIS) 2016;10(12):5572–95.
[24] Chaudhry SA. Correcting "PALK: Password-based anonymous lightweight key agreement framework for smart grid. Int J Electr Power Energy Syst 2021;125:106529.
[25] Blanchet B. ProVerif automatic cryptographic protocol verifier user manual. Paris, France: CNRS; 2005.