Privacy Preserving Based on Seamless Authentication with Provable Key Verification using mIoMT for B5G-enabled Healthcare Systems

B D Deebak, Seong Oun Hwang, *Senior Member, IEEE,*

***Abstract*—B5G-enabled healthcare systems interconnect a wide range of Internet of Medical Things (IoMT) using supportive networks such as heterogeneous networks and cognitive radio networks to enhance the medical infrastructure. In healthcare, IoMT integrates access technologies, computing infrastructure, and services to connect healthcare systems to handle intensive computation without sharing private data. As a result, healthcare systems accessing a massive IoMT (mIoMT) utilize real-time data sharing to enhance the overall resource efficiency of remote patient monitoring. To optimize the IoT-generated data, the application interface of the computing device regulates self-management messaging systems with healthcare providers. By utilizing direct communication with the networks, they offer a long-lasting service, enhancing the performance trade-off. Since the network has more of a digital existence in the physical universe, a convergence of cloud-server integration with IoT inherently causes more security challenges to preserving the privacy of edge computing systems. Therefore, in this paper, we present privacy preserving based seamless authentication with provable key verification (PPSA-PKV) for securing B5G-enabled healthcare systems. To preserve the identities of the registered users, the proposed PPSA-PKV applies a collision-free cryptographic hash function and elliptic-curve arithmetic. Security analyses including formal and informal show high-level privacy protection for the proposed PPSA-PKV with seamless verification compared to other state-of-the-art approaches. The simulation analysis shows that the proposed PPSA-PKV incurs less delay ($\approx 0.14sec$) and improves throughput ($\approx 1865bits$) to fulfill the energy efficiency (at an average 0.294J) of B5G networks. Lastly, a learning model using a support vector machine (SVM) demonstrates the monitoring process of edge data centers to detect malicious authentication requests.**

***Index Terms*—Massive Internet of Medical Things, B5G, Data Sharing, Healthcare, Privacy Preserving, Authentication, Learning.**

I. Introduction

Digital convergence transforms peoples' lifestyles through all-time connected devices to deliver a new dimension of real-time applications ensuring cost-effective solutions in the healthcare industry. Unlike the communication paradigm of IoT, IoMT interconnects healthcare monitoring and emergency alerting systems with smart devices to share and link the technologies via different networking channels to deliver exceptional services in a timely way [1]. Rapid advancements in multi-tier heterogeneous networks have unified device-to-device (D2D) along with mobile edge computing (MEC) to fulfill the requirements of next-generation medical service systems. The new generation of relaying technologies resolves various complex problems using data processing systems to deliver insights via communication channels to promote preventive care [2]. The processing system of IoMT is comprised of medical devices and application interfaces with limited processing units both computation and communication to streamline the routine tasks enabling personalized treatment plans and predictive analysis in healthcare services. Precisely, wearable devices are predicted to be more than 500 billion to the Internet by the end of 2025 [3].

In recent times, devices like edge nodes and gateway have been associated with homogeneous and heterogeneous IoMT to resolve security issues related to privacy protection without compromising their performance trade-off. Therefore, in the healthcare industry, B5G-enabled health systems operate the technological infrastructure using trusted service providers to perform seamless authentication and privacy preservation. To address system vulnerabilities (wireless infusion and radiology information), in the literature, Shen et al. [4] constructed a key agreement protocol using group data sharing which provides data outsourcing in a cloud computing environment. The constructed mechanism structures the design blocks symmetrically to meet key constraints of fault tolerance and to determine common secret keys owned by the groups in order to provide better security efficiency in cooperative environments like electricity, agriculture, and retail.

Wu et al. [5] intended to design an authentication protocol with lightweight operators to meet the operational specifications of resource-constrained medical devices. They use a two-factor strategy to secure the access channel in order to ensure secrecy and untraceability, however, this mechanism cannot handle the processing workload by using a gateway node to manage user and server registration to achieve a common session key agreement. To prevent security leakage attacks, Khan et al. [6] devised an authentication framework with privacy preserving. Moreover, this mechanism hyper-elliptic curve cryptography to reduce the computation capacity of the onboard units. However, this framework cannot exploit the key characteristics of the transportation systems such as privacy and confidentiality to perform advanced security measurements (i.e., network and application). Lee et al. [7] applied group authenticated key agreement to construct an anonymous protocol with lightweight properties aiming to solve the issues related to transmission and security efficiencies.

In this protocol, the authors exploit the properties of physically unclonable functions (PUF) to differentiate the execution scenario associated with authentication and authenticated key agreement phases. While analyzing in-transmit data features of e-Health systems, the cryptographic solutions reveal that the key agreement strategies such as privacy preserving [8], [9], two- [10], three- [11], and multi-factor [12] can protect the security of the communication systems. Most importantly, they prefer symmetric cryptosystems to minimize the system overheads including computation and communication to deliver customized application services with high-level security and data protection. However, enduring data integrity and confidentiality significantly require low-power biomedical devices to perform flexible remote treatment. To analyze the integrity of computing devices, a few recent studies [13]–[15] have researched the properties of device authentication and data privacy via remote monitoring systems.

*Corresponding Authors (Seong Oun Hwang)

This work was supported by the Brain Pool Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and Information Communication Technology (ICT) under Grant No. 2022H1D3A2A02081848.

B D Deebak and Seong Oun Hwang are with the Department of Computer Engineering, Gachon Univesity, Gachon University, Seongnam 13120, South Korea e-mail: (deebak@gachon.ac.kr and sohwang@gachon.ac.kr).

In [13], an authentication protocol with forward secrecy was proposed to handle the computation complexity of low-power mobile devices, whereas in [14], [15], two different strategies namely lightweight authentication with preserving user anonymity (LA-PUA) and privacy-aware authentication with access control (PAA-AC) were formulated to generate a secure channel between the computing devices and end-users in order to record their credentials with the registration center. However, the authentication strategies utilized in LA-PUA and PAA-SC cannot resist user tracking and forgery attacks due to ineffective usage of user revocation while sharing their secret parameters with the communication entities. To minimize utility costs and service outages during user revocation, the convergence in digital healthcare systems necessitates B5G technology and IoMT with edge computing platforms and application services [16].

However, promoting the B5G-enabled edge computing platform and its application services are in need of device privacy and secure connectivity to protect the potential features of data services [17]. On the other hand, the population residing in urban areas intensifies the usage rate of data workloads on the distribution of the B5G networks via mIoMT to obtain effective resource scheduling in advance enterprise services. As a result, the edge data center utilizes machine learning models like support vector machines (SVM) to monitor the features relevant to physical and operational statistics [18]. To achieve service optimization and secure operation through edge computing, this paper applies a security-by-design approach using SVM, ensuring end-to-end authentication and performance maintenance with provable key verification.

A. Motivation

Technologies, applications, and services in B5G provide full coverage of broadband connectivity to operate high-resolution sensing services and localization via edge intelligence to fulfill the desired objectives of autonomous systems. However, to secure B5G-enabled healthcare systems, enterprises need to examine connected devices via a key management system in order to conduct state-of-the-art research providing massive connectivity in the cooperative networks [19]. As a result of this, the services with time-engineered applications need a robust key agreement technique with proper mutual authentication to address security and privacy issues related to B5G networks in healthcare environments. Key agreement techniques in mIoMT focus on data access, sharing, and reuse on the basis of a few predefined network parameters to demonstrate the privacy risks associated with system updates and backups. To highlight the technological awareness, the researchers utilize a few additional features with key agreement processes in order to secure data transfer between the communication entities [20]–[24].

In [20], a secure user sign-in authentication (SUSA) was proposed to secure the communication of cloud-based distributed networks, whereas in [21], [23], seamless authentication (SA) and mutual authentication with dynamic security (DA-DS) were constructed to use a shared session key in a dynamic way to enhance the connectivity of IoT devices. However, they cannot maintain data privacy operated by the mIoMT devices to address the issue strongly correlated with mutual authentication and key agreement. Therefore, in this paper, we apply the cryptographic primitives such as hash function and elliptic-curve arithmetic to deal with security vulnerabilities addressed in the existing schemes. Additionally, we design a seamless authentication with privacy preserving to protect mIoMT deployment in B5G networks. Moreover, to improve the performance of collaborative edge computing, the proposed PPSA-PKV uses complex infrastructure. This infrastructure has its network environment to access the convenient services of B5G, including more reliability and increased availability to monitor the process of authentication via the edge data center using an appropriate classifier (i.e., SVM).

B. Contributions

Our significant contributions are as follows.

1) We propose privacy preserving based on seamless authentication with provable key verification (PPSA-PKV) that applies the suitable crypto-primitives such as hash function and elliptic-curve arithmetic to prevent security vulnerabilities in the prevalence of mIoMT, such as forging data [20], guessing passwords guessing [21], eavesdropping attacks [22], etc.
2) We choose multiple distributed servers with provable key verification using sink nodes to preserve the privacy of computing devices and user identities by utilizing a trusted authority.
3) We apply informal and formal analyses to show the security level of the proposed PPSA-PKV in terms of key authentication, agreement, secrecy, etc. Also, we present an extensive analysis of cost efficiencies to prove that the proposed PPSA-PKV can reduce its driving factors in terms of computation and communication significantly to process the biological characteristics of I_{MD} faster than other schemes.
4) We develop a practical testbed using NS3.37 to show that the proposed PPSA-PKV achieves a better quality of services such as delay, throughput, and energy consumption than other schemes to adhere to constraints on sustainable mIoMT deployment.
5) Lastly, we deploy edge data modeling on Raspberry Pi-4 to analyze the communication scenario of mIoMT connected via LPWAN. In addition to this, we utilize a dedicated computing system to exploit a few attributes of the proposed PPSA-PKV and other relevant approaches such as the time of authentication request, verification status, center identity, and location to learn the significance of the statistical learning approach (i.e., classification and prediction using SVM).

C. Structure of the Paper

The rest of the sections of this paper are as follows. Section II reviews the security weaknesses and complexities of the existing works in terms of forgery, tracking, and secrecy. Section III presents the modeling strategies in healthcare including network and threat to assess the core functionality of mIoMT. Section IV presents the algorithmic structure of PPSA-PKV utilizing collision-free hash function and elliptic-curve arithmetic to preserve data privacy and user identities. Section V conducts security analyses both informal and formal to demonstrate the security efficiency of PPSA-PKV. Section VI evaluates performance analyses including computation and communication to assess cost efficiency. Additionally, this section demonstrates a suitable network environment using NS3.37 and a learning model to examine the quality metrics and prediction accuracy. Section VII concludes this research.

II. Related Work

Of late, extensive research work has been carried out for IoMT-based healthcare systems using various privacy preserving techniques [33]. Unfortunately, the technique with privacy preserving demands a large comparable key size to perform resource-intensive computation to determine a shared session key with the real-time entities. As a result, the recent works utilized lightweight authentication with

TABLE I: Comparative Analysis of Existing Privacy Preserving Systems Related To Key Attributes and Complexities

Authentication Schemes/Year	Applied Primitives	Resistant To							Complexities		
		Forgery Attack	Password Guessing Attack	User Tracking	Privileged Insider Attack	Perfect Forward Secrecy	Secure Session Key Agreement	Masquerade Attack	Computation Cost	Communication Cost	Energy Consumption
LSA-D2D [25], 2018	Collision-free one-way hash function	✔	✔	✘	✔	✔	✘	✘	High	High	High
LAS [26], 2019	Collision-free one-way hash function; pseudonym identity-based	✔	✘	✔	✘	✘	✘	✘	High	High	High
EMFUAP-FS [27],2020	Collision-free one-way hash; RSA cryptosystem;fuzzy extractor	✔	✘	✘	✔	✘	✘	✘	High	High	High
IAKP [28], 2021	Collision-free one-way hash; XOR operations	✔	✘	✔	✔	✘	✘	✘	High	High	High
UC-PPAP [29], 2022	Collision-free one-way hash function	✔	✔	✔	✔	✘	✘	✘	High	High	High
PPA-SC [30], 2022	Elliptic curve cryptography	✘	✘	✘	✘	✘	✘	✘	High	High	High
PP-TFA [31], 2022	Collision-free one-way hash function	✘	✔	✘	✘	✔	✘	✘	High	High	High
PPA-HECC [32], 2022	Hyper Elliptic Curve Cryptography	✘	✘	✘	✘	✘	✘	✘	High	High	High
The proposed PPSA-PKV	Collision-free one-way hash function and elliptic-curve arithmetic	✔	✔	✔	✔	✔	✔	✔	Low	Low	Low

Yes - ✔; No - ✘

high-level security to minimize computation complexities [34]–[36]. However, they have addressed various security and privacy issues while ensuring concrete solutions to edge intelligence B5G networks. Other studies focused on different authentication techniques namely knowledge [35], token [36], identity [37], biometric [38], and three-factor [39] to satisfy the requirements of device anonymity with perfect secrecy.

Mohseni-Ejiyeh et al. [25] devised a data-sharing scheme based on lightweight security-aware device-to-device (LSA-D2D) for 5G cellular networks. This LSA-D2D scheme applies a crypto-primitive of bilinear pairing to perform proper encryption and decryption processes to resist various attacks (forgery and spoofing) in open 5G networks. However, they cannot prevent a few potential attacks like password guessing, replays, and the man-in-the-middle. Zhou [40] intended to improve signcryption mechanism with provable security to monitor the patient's status remotely via healthcare applications. Moreover, the improved mechanism regulates certificateless cryptosystem to preclude the private-key escrow problem [41]. Amin et al. [42] formulated an anonymous remote authentication scheme using a wireless medical sensor network to preserve the property of anonymity and mutual authentication for mobile users. They use a specific patient monitoring architecture to provide a user-friendly environment and utilize an appropriate medical sensing unit to analyze its energy consumption ratio. Shuai et al. [26] presented a lightweight authentication scheme (LAS) using on-body sensor networks to assess the security attributes of patient monitoring systems. Unfortunately, the LAS scheme cannot withstand privileged insider and identity theft to preserve the anonymity of the end users.

Wong et al. [43] developed a three-factor anonymous authentication (TF-AA) integrating biometrics and smartcards to provide high-level security in multi-server healthcare systems. However, the TF-AA scheme consumes high computation and communication overhead while controlling user access in privacy-preserving environment. To address the performance issue pertaining to delay-sensitive applications, Jia et al. [44] presented a fog-driven three-party authentication (FD-TPA) using bilinear pairing. Cryptanalysis showed that the FD-TPA scheme cannot guarantee anonymity and secrecy to meet the standard requirements of distributed healthcare applications. Wang et al. [27] reviewed the core structure of multi-factor security to construct an efficient multi-factor user authentication protocol with forward secrecy (EMFUAP-FS) to provide secrecy during real-time data access. The EMFUAP-FS scheme uses a computation strategy of RSA cryptosystem to reduce the energy consumption ratio of the sensing units [45]. However, this scheme cannot perform service authorization based on RSA to resist forgery attacks.

Wu et al. [28] formulated an improved authentication and key agreement (IAKP) to address security issues such as session-specific information attacks and privacy preservation in fog-driven healthcare systems. Masud et al. [29] intended to develop a user-centric privacy preserving authentication protocol (UC-PPAP) for ambient machine-intelligent healthcare systems. Soleymani et al. [30] aimed to structure a privacy preserving authentication with seamless connectivity (PPA-SC) to preserve the privacy of medical repositories. Zhang et al. [31] proposed a privacy preserving three-factor authentication (PP-TFA) to ensure privacy protection with limited computational cost. Khan et al. [32] developed a privacy preserving authentication using hyper-elliptic curve cryptography (PPA-HECC) to meet the prominent features of intelligent transportation systems. However, the scheme is susceptible to several vulnerabilities including privacy preservation.

Xun et al. [33] constructed an authentication model using machine

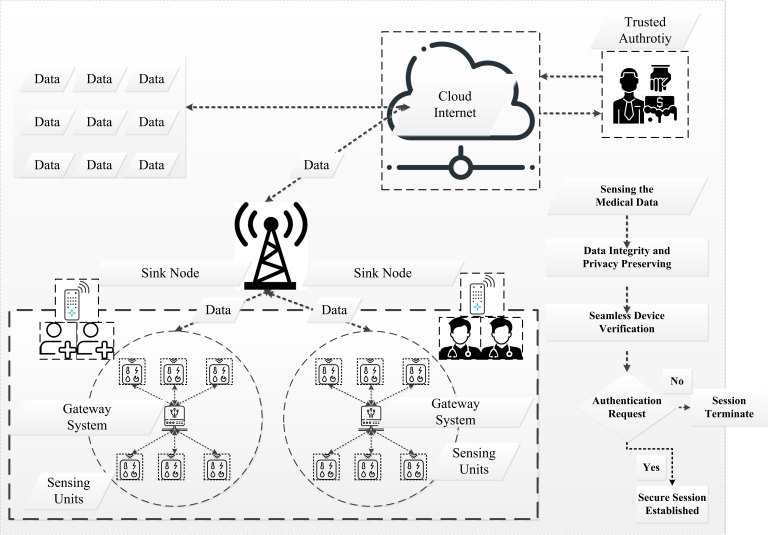


Fig. 1: Network Model in mIoMT Environment.

learning to enhance the process of behavioral characteristics without any additional computation cost in the automobile industry. Xiao et al. [46] developed a computing system using a deep-learning approach to authenticate and verify the data request processing by edge computing devices. However, these existing learning models do not explicitly use their key features relating to the behavior of the communication entities including static and dynamic information [47]. Importantly, the relevant existing schemes are still vulnerable [25]–[32] and incur more extensive computation to guarantee privacy preservation, as shown in Table I. Therefore, the proposed PPSA-PKV applies a collision-free one-way hashing and elliptic-curve arithmetic to develop a robust authentication framework with privacy preservation in order to achieve a high level of resiliency and integrity in the services of mIoMT.

III. Network and Threat Models

In this section, we discuss the design models including the network and the threat to highlight the essential requirements of mIoMT on the basis of building components between the participating entities.

A. Network Model

The development of medical device technologies including hardware and customized software envisages designing a network model to decentralize the process of real-time patient monitoring. However, gathering IoT-generated data is an essential process to fulfill the key requirements of networking systems such as lifetime and self-configuration. Thus, in this network model, an edge-based IoMT is enabled to achieve a cost-effective solution while accessing the data over an insecure channel. Moreover, the associated medical device carrying by PAD_i uses an authorized cloud server to secure its connectivity with a personalized application interface. Although the implantable devices are highly scalable over the communication networks, managing the device connectivity is challenging to control the virtual sensing units through inter-networking protocols and standards. As a result, in this model, we chose a two-level topology to operate and manage the cloud system via a dedicated gateway to access the clustered sensing units [48].

While managing the networks over the cloud, computing devices including the gateway system provisionally manage the storage of sensing data via the sink node to offer a seamless connection upon the verification of trusted authority between end users and the cloud server. Using a cloud-managed network, the edge network associated with the gateway system obtains the sensing data via neighboring clusters to deliver them to the sink nodes using lower-power wireless technologies (6LoWPAN). Considering this modeling scenario, the sink node acts as a network coordinator to identify device-level authentication to provide a proper data-sharing system in the IoMT environment. Therefore, in this paper, we prefer to use lightweight cryptographic operations to formulate a privacy prreserving framework [49].

In this framework, we identify a few possible security threats such as forging and identity theft among the end users and the cloud server to handle massive sensing data in order to verify system anonymity. Additionally, the computing devices utilize a self-compiling system to initiate the process of key authentication, as shown in Fig. 1. The components of the network model and their descriptions are as follows:

1) **User:** Patients/Doctors PAD_i access the gateway system via sink node to prove their legitimacy before initiating the communication with the sensing units.
2) **Gateway:** This system acts as a communication hub between the patient/doctor and the sensing unit to share or exchange sensitive information with the cloud network.
3) **Sensing Unit:** The medical device (i.e., wearable and unwearable) builds the machine interface using sensory technologies within the connection applications to monitor the patient's condition remotely.
4) **Cloud Internet:** This service allows remote servers to host the application services and resources which enable users to access the data via a dedicated Internet.
5) **Sink Node:** This node does not use an outgoing connection explicitly to deliver the data contents to the cloud Internet. However, depending on the application requirements, it can have one or more sink nodes to perform data collection.
6) **Trusted Authority:** This entity acts as a trusted third party to store the authorized certificates of the users legalized by the policy issuer.

B. Threat Model

The adopted model explores the source attributes of PPSA-PKV to analyze the security features of a patient monitoring system which typically follows the guidelines of the Dolev-Yao (DY)[50]. The participating entities such as the cloud server, the patient, and

the experts utilize a few fundamental strategies of DY to perform statistical or crypt-analytics over an insecure channel to extract the transmitted messages or intercept the communication with the derived session key to disrupt the network flow. An active or passive adversary, A_{dv}, is capable of performing cyber threats such as impersonation, replaying, eavesdropping, etc. under the following assumptions.

1) A_{dv} can identify the networking features to intercept or defer the exchange of data transmissions between the participating entities.
2) A_{dv} can capture source attributes of communication systems physically using arrayed medical devices to perform sensitive computation using a power analysis attack.
3) A_{dv} can enumerate the identities of devices or users to associate them with application servers to derive the real identities of end users or remote servers.
4) A_{dv} can attempt to secure the private information of the networks to access or monitor the sensitive information of the patient, which compromises the system's ability to manage the flow of data transmission.
5) A_{dv} can exploit a few significant features of the secret data available in SC_R i.e., $\{DID_i, T_r, CPW_i\}$ to analyze the security weaknesses of the provable key verification phase.

To simulate the attack-like masquerade or denial-of-service on the proposed PPSA-PKV for B5G-enabled healthcare systems, the proposed Real-or-Random (RoR) adopts a few significant modeling features of Abdalla et al. [51]. In the proposed RoR, the security model applies a modeling game among two probabilistic polynomial time Turing machines i.e., challenger C_H and A_{dv} to ensure secure communication during the process of authentication. Let us assume that C_H has the privilege to access the remote system which has already been employed with the key attributes of the proposed PPSA-PKV K_A to perform interception and modification.

In the case of evaluating the security of K_A, C_H aimed to attract A_{dv} to initiate an attack on K_A, however, C_H deliberately concerned about the behavior of A_{dv} in learning the sensitive information of the remote system. As a result of this, C_H designed and developed a gaming system using Oracle to learn the behavioral characteristics of A_{dv}. In realizing this fact, upon proper system initialization, the proposed ROR allows A_{dv} to choose a guessing bit $c \in \{0,1\}$ which generates a series of queries to examine the probability of winning the game by A_{dv}. Moreover, this key assumption evaluates the features of the secret data in real-time to prove the efficiency of the key verification phase. The descriptive proof based on A_{dv} capabilities is explained in Section V[B].

IV. Proposed Mechanism

This section presents the proposed privacy preserving based on seamless authentication with provable key verification to validate the authenticity of the participating entities. The proposed PPSA-PKV applies a collision-free one-way hash function and elliptic-curve arithmetic to protect the privacy of the computing devices. The proposed PPSA-PKV manages a trusted authority, T_A, a patient/doctor, PAD_i, and a cloud server, CS_j, over an insecure channel to examine the connectivity of distributed computing systems. Using authentication and key agreement, the participating entities can generate a session key to initialize a secure channel between PAD_i and CS_j. Moreover, the secured channel uses a dedicated AG_A to preserve the privacy of PAD_i and CS_j upon authorizing their service request with T_A. To verify the process of authentication between the participating entities, the proposed PPSA-PKV has the following assumptions to be made.

1) AG_A acts as an honest node to administer the cryptographic process involved in distributed servers to establish secure communication between different end-users via sink nodes.
2) T_A utilizes a collision-free hash function to access the data transmission cryptographically linked with the public key of the end-users to verify the authenticity of the communication between PAD_i and CS_j; and
3) CS_j selects its specified private key, PR_{Key}, to process the request in computerized form in order to verify the process of data transmission with AG_A.
4) The entities PAD_i, AG_A, and CS_j devise the process of transmission through SN_j to deliver the application services effectively in mIoMT environment. To deal with malicious attacks, the proposed PPSA-PKV uses $G_P(X,Y)$ using a key management system whereby the entities can validate their request using S_{Key} in prior: 1. To achieve numerous authentication processes at once; and 2. To reduce the cost of computation involved in the encryption and decryption process.

The PPSA-PKV scheme comprises five execution phases: pre-deployment, registration, system login, authentication with device verification, and session key update. Table II describes important notations used.

TABLE II: Important Notations in the Proposed PPSA-PKV

Notation	Description
PAD_i	Patient/Doctor where $i=1,2,3...$
T_A	Trusted authority
CS_j	Cloud server where $j=1,2,3...$
AG_A	Authentic gateway access
SC_R	Smart card reader
SU_i	Sensing unit where $i=1,2,3...$
MS_{ID_j}	Masked sensing identity where $j=1,2,3...$
$MI_{rd_{su}}$	Masked information
S_{ID_j}	Identity of the sensing units where $j=1,2,3...$
SI_j	Secret information where $j=1,2,3...$
q	Prime integer
PR_{Key}	Predetermined private key
$G_P(X,Y)$	Group key generator
PU_{Key}	Public key of PAD_i
CID_i	Context identity of PAD_i
CPW_i	Context password of PAD_i
DID_i	Device identity of PAD_i
Pwd_i	Password of PAD_i
T_r, T_s, T_c, T_{su}	Timestamps
$rd_r, rd_i, rd_s, b, rd_{su}$	Random integers
$h(.)$	A collision-free one-way hash function
CK_1, CK_2	Key derivatives of CS_j
SID_j	Server identity
PRK_j, PUK_j	Private and public key of CS_j

Pre-deployment: In this phase, CS_j selects a few significant parameters of a common elliptic curve integer, represented as E_C, over its definite prime field, q, to determine $Y^2 = X^3 + PX + Q$. Additionally, CS_j chooses its group generator, $G_P(X,Y)$ over Z_q^* to access the data request processed by AG_A. The participating entities, PAD_i, utilize $G_P(X,Y)$ to obtain the CS_js' public and private keys, which are required to compute a shared session key to prevent a temporary information attack.

CS_j selects PR_{Key} to find its public key, $PU_{Key} = PR_{Key}.G_P(X,Y)$. It is relevant to note that PR_{Key} is accessible only to T_A, which shares PU_{Key} over an insecure network to PAD_i to perform data-sharing.

User Registration: As a new legal entity, PAD_i provides the key credentials, such as DID_i and Pwd_i, without any physical constraints to generate a valid authentication request. The associated flows are as follows.

Step 1: PAD_i chooses DID_i, T_r, and Pwd_i to compute $CID_i = h(DID_i \| T_r)$ and $CPW_i = h(Pwd_i \| T_r)$ and sends the parameters, including CID_i and CPW_i, to AG_A over a secure communications channel.

Step 2: PAD_i selects a random number, rd_r, to generate the source parameters $\{DID_i, CID_i, CPW_i, rd_r, T_r\}$ in order to process the authentication request as a legitimate one. Accordingly, CS_j obtains the processed request via AG_A to compute the following:

$$\begin{aligned} CK_1 &= h(CID_i \| CPW_i \| rd_r) \\ CK_2 &= h(CID_i \| CPW_i) \oplus h(rd_r \| PR_{key}) \end{aligned} \tag{1}$$

It is important to note that $PU_{Key} = PR_{Key}.G_P(X,Y)$ is computed to discover parameters $\{PR_{Key}, PU_{Key}\}$ between PAD_i and CS_j. To make it more evident, in the proposed PPSA-PKV, each user holds his/her unique information using the associated value of CID_i to provide proof of legitimacy. This ensures that the user logging on to the remote system is already authorized by T_A to provide data access via CS_j in order to control the level of security required by one or more administrators to protect the application environment like healthcare. While dealing with cloud deployment, the computing infrastructure can have more administrators to mobilize the delivery channels such as consultation and remote monitoring using the generated values of CID_i to overcome the risk closely related to the authentication request and security control, managing the application resources.

Step 3: CS_j maintains user identities PAD_i along with the attributes (rd_r, T_r) to determine its secure key, S_{Key}. Additionally, the source attributes $\{CK_1, CK_2, rd_r, T_r, PU_{Key}, G_P(X,Y)\}$ are reserved in the memory of PAD_i associated with CS_j to verify authentic secret key S_{Key}.

Sensing Unit Registration: The steps involved in the registration of the sensing units are as follows.

Step 1: SU_j selects two random integers $b, rd_{su} \in Z_q^*$ to find a few authentic information $A_1 = g^b$ and $A_2 = y^b = g^{bx}$. Additionally, masked sensing identity $MS_{ID_j} = S_{ID_j} \oplus h(A_1 \| A_2 \| T_{su}$ and masked information of rd_{su} i.e., $MI_{rd_{su}} = rd_{su} \oplus X_{AG_A \to SU_j}$ are determined to verify the information request using $V_j = h(X_{AG_A \to SU_j} \| A_1 \| A_2 \| rd_{su} \| S_{ID_j} \| T_{su})$, where T_{su} is the current timestamp of the information request. Lastly, SU_j generates the source parameters $\{V_j, MS_{ID_j}, MI_{rd_{su}}, A_1, T_{su}\}$ to AG_A.

Step 2: On obtaining the parameters $\{V_j,\ MS_{ID_j},\ MI_{rd_{su}},\ A_1,\ T_{su}\}$ from SU_j, AG_A verifies whether $|T_{su} - T_c| < \Delta_T$. In the case of not holding the key freshness T_{su}, AG_A discards the request processed by SU_j. Otherwise, AG_A computes $A_2 = A_1^x = g^{bx} \pmod q$ to recover $S^*_{ID_j} = MS_{ID_j} \oplus h(A_1 \| A_2 \| T_{su}$ and searches the list of authorized sensing units SU_j to provide proper data access via sink node. In the case of $S_{ID_j} \neq S^*_{ID_j}$, AG_A discards the request. Otherwise, AG_A finds $rd^*_{su} = MI_{rd_{su}} \oplus X_{AG_A \to SU_j}$ and validates the request using $V_j = h(X_{AG_A \to SU_j} \| A_1 \| A_2 \| rd^*_{su} \| S^*_{ID_j} \| T_{su})$.

If the validation is unsuccessful, then AG_A discards the request. Otherwise, AG_A finds secret information $SI_j = h(X_{AG_A \to SU_j} \| S^*_{ID_j})$ and masked information $MI_j = SI \oplus h(X_{AG_A \to SU_j} \| A_1 \| A_2 \| rd^*_{su} \| S^*_{ID_j} \| T_{su})$ to verify the information $N_j = h(X_{AG_A \to SU_j} \| A_1 \| A_2 \| rd^*_{su} \| S^*_{ID_j} \| T_{su})$. Lastly, AG_A sends the generated parameters $\{MI_j, N_j, T_{su}\}$ to SU_j via secure channel.

Step 3: After obtaining the parameters $\{MI_j, N_j, T_{su}\}$ from AG_A, SU_j probes the key freshness of T_{su}. If the freshness of the key does not hold, then SU_j terminates the session request. Otherwise, SU_j uses $SI^*_j = MI_j \oplus h(h(X_{AG_A \to SU_j} \| A_1 \| A_2 \| T_{su})$ to verify the processing request $N_j = h(SI^*_j \| X_{AG_A \to SU_j} \| A_1 \| A_2 \| T_{su})$. If the verification is successful, then SU_j stores SI^*_j into its database and accordingly, deletes $X_{AG_A \to SU_j}$ from the database to issue the confidential information (CI) to AG_A. Otherwise, SU_j terminates the request.

Step 4: On obtaining the CI from SU_j, AG_A deletes the other parameters of SN_j i.e., S_{ID_j} and $X_{AG_A \to SU_j}$ to initiate the process of data sharing via sink node to CS_j.

Cloud Server Registration: In this stage, CS_j prefers to use the generated keys $\{PR_{Key}, PU_{Key}\}$ to verify transmission requests exercised by PAD_i over insecure networks.

Step 1: CS_j publishes its identity, SID_j, to T_A via AG_A to check whether any PAD_i is legally registered to share its sensitive information.

Step 2: After obtaining SID_j, T_A uses $PRK_j = h(SID_j \| S_{Key})$ and $PUK_j = PRK_j.P$ to compute the private key and public key of CS_j.

Step 3: Subsequently, T_A publishes parameters $\{SID_j, PUK_j\}$ to issue trustworthy parameters PRK_j and S_{Key} to CS_j.

System Login PAD_i: In this stage, PAD_i uses smartcard reader SC_R to register the credentials, DID_i and Pwd_i, in order to process the session request. The steps involved in the session request are as follows.

Step 1: SC_R chooses $CID_i^{New} = h(DID_i \| T_r)$ and $CPW_i^{New} = h(Pwd_i \| T_r)$ to validate parameters like CID_i and CPW_i from the registration phase.

Step 2: Subsequently, SC_R validates the login attributes through $CK_1 \stackrel{?}{\to} h(CID_i^{New} \| CPW_i^{New} \| T_r)$ to check whether the request is legitimate or not.

Step 3: If accepted, this phase further computes $h(rd_r \| PR_{Key} \| S_{Key}) = CK_2 \oplus h(CID_i \| CPW_i)$. Otherwise, SC_R terminates the login request.

Step 4: SC_R generates random number rd_i to compute $rd_i.PU_{Key}(X,Y)$. Also, it computes $LR_i = h(h(rd_r \| T_r \| PR_{Key}) \| h(rd_r.PU_{Key}(X,Y)))$ and $En_i = En_(PU_{Key})(rd_i)$ to compute the desired source attributes: $\{LR_i, En_i, T_s\}$ in order to validate the session request further.

Step 5: SC_R stores $\{LR_i, En_i, T_r\}$ in to implantable medical device I_{MD} to initiate its data transmission.

Authentication with Provable Key Verification: In this stage, AG_A validates the session request of PAD_i to create an authentication message that verifies the identity of CS_j to share the secured session key with PAD_i and CS_j. Note. Before initializing the transmission process via sink node, AG_A verifies the requests processed by PAD_i and SN_j through the associated search parameters $LR_i = h(h(rd_r \| T_r \| PR_{Key}) \| h(rd_r.PU_{Key}(X,Y)))$ and $N_j = h(X_{AG_A \to SU_j} \| A_1 \| A_2 \| rd^*_{su} \| S^*_{ID_j} \| T_{su})$. In the case of successful verification, AG_A searches the identities of PAD_i and SU_j using SC_R and SI_j to validate the timestamp T_r and T_{su} with the objective of processing the data transmission. Otherwise, AG_A terminates the requests.

Step 1: CS_j obtains login request $\{LR_i, En_i, T_s\}$ processed by SC_R and accordingly, verifies the credentials of PAD_i over $T_s^{'}$. The steps involved in the verification process are as follows.

Step 1.1: If $(T_s^{'} - T_s) \leq \delta_{(T_s)}$ is succcessful, then CS_j proceeds with the login request of PAD_i, i.e., $\{LR_i, En_i, T_s\}$. Otherwise, CS_j rejects the request.

Step 1.2: With successful verification, CS_j accepts message request $En_i = En_(PU_{Key})(rd_i)$ by using PR_{Key} to verify $rd_i = DE_(PR_{Key})(En_i)$.

Step 1.3: Subsequently, CS_j computes $En_1 = h(rd_r \parallel T_r \parallel PR_{Key})$ and $En_2 = h(rd_i.PU_{Key}(X,Y))$ to validate the message request.

Step 1.4: With successful validation, $LR_i \stackrel{?}{\rightarrow} h(En_1 \| h(En_2)))$ accepts the message request from CS_j. Otherwise, CS_j terminates the request.

Step 2: CS_j finds a new session key, $c = h(rd_s.rd_i.PU_{Key}(X, Y))$, to verify the message request by using rd_s.

Step 3: Subsequently, CS_j finds $MR_i = h(rd_s \| rd_i \| rd_r \| T_c)$ to create an authentic message and utilizes T_c to verify the source parameters, rd_s, rd_i, and rd_r.

Step 4: After successful generation of authentication message, PAD_i obtains the deirable inputs $\{MR_i, T_c\}$ from CS_j to verify the source attributees. The steps involved in PAD_i are as follows.

Step 4.1: PAD_i verifies whether $(T_c - T_c^{'}) \leq \delta_{(T_c)}$ holds or not to verify the message request processed by CS_j. If unsuccessful, PAD_i terminates the request.

Step 4.2: PAD_i then computes $MR_i^{New} = h(rd_s \| rd_i \| rd_r \| T_c)$ to check whether the message was already processed by PAD_i. Otherwise, PAD_i terminates the request.

Step 4.3: PAD_i computes session key $SK_{New} = h(rd_s.rd_i.PU_{Key}(X, Y))$ to ensure a secure session with CS_j, where rd_s and rd_i are the parameters of random numbers generated by the login session.

Step 5: Lastly, PAD_i and CS_j agree to share session key SK_{New} via AG_A to secure data transmission.

Password Update: In this stage, PAD_i tries to modify Pwd_i to reinstate the session request with CS_j.

The steps are as follows. *Step 1:* PAD_i utilizes its own SC_R using end-device to provide the confidential Pwd_i.

Step 2: PAD_i computes $CPW_i^{old} = h(Pwd_i \parallel T_r)$ to verify CPW_i. It has a few additional computations to execute, as follows.

Step 2.1: If $CPW_i^{old} == CPW_i$ holds, then PAD_i updates password Pwd_i to re-establish the session with CS_j, and then executes *Step 3*. *Step 2.2:* Otherwise, the request is terminated.

Step 3: PAD_i returns CPW_i^{old} to CPW_i in the smart device system memory via SC_R.

V. Security Analysis

In this section, we show both informal and formal analyses to evaluate the security properties of the authentication and key agreement protocol.

A. Informal Analysis

The numerical results of security properties related to key freshness are as follows.

1) $\mathcal{P}_1$ ***Property of Proper Mutual Authentication****:* To probe mutual authentication, the participating parties (PAD_i, CS_j) exchanges their shared session key via T_A. In the PPSA-PKV scheme, PAD_i authenticates $SK_{New} = h(rd_s.rd_i.PU_{Key}(X,Y))$. In the authentication phases, T_A verifies PAD_i via AG_A by using $En_1 = h(rd_r \parallel T_r \parallel PR_{Key})$ and $En_2 = h(rd_i.PU_{Key}(X,Y))$. Additonally, T_A checks whether PAD_i matches its source attributes with the conditional expression $LR_i \stackrel{?}{\rightarrow} h(En_1 \| h(En_2)))$ to access the data transmission. Although A_{dv} attempts to modify the message request of PAD_i and tries to portray a legal server, A_{dv} cannot find any source attributes, $\{h(.), En_1, En_2, PR_{key}\}$ to verify its authenticity. Therefore, A_{dv} cannot formulate a legitimate request to CS_j to authorize the session estsblishment. Hence, the proposed PPSA-PKV ensures proper mutual authentication between PAD_i and CS_j to secure the communication channel.

2) $\mathcal{P}_2$ ***Property of Reliable Session Key Agreement****:* In PPSA-PKV, PAD_i shares SK_{New} via T_A with CS_j to initiate a secure session. Upon confirming the execution of the login and authentication, PAD_i can mutually share sensitive data with AG_A using SK_{New}. Furthermore, PAD_i data gathered by I_{MD} are encrypted using $SK_{New} = h(rd_s \| rd_i \| PU_{Key}(X,Y))$. This computation validates $MR_i = h(rd_s \parallel rd_i \parallel rd_r \parallel T_c)$, which validates the genuineness of random integers to form a legal request. As T_c changes during execution, different sets of secure session keys can be generated to provide more communication services in parallel. Hence, PPSA-PKV provides reliable session key agreement between PAD_i and CS_j.

3) $\mathcal{R}_1$ ***Resilience against Privileged Insider Attacks****:* The PPSA-PKV scheme hardly ever transmits communication parameters such as $\{h(.), En_1, En_2, PR_{key}\}$ as plaintext to authenticate server access. In order to examine parameters further, PAD_i uses $PU_{Key} = PR_{Key}.G_P(X,Y)$. Thus, an authentic server cannot obtain the secret key without $CK_1 = h(CID_i \parallel CPW_i \parallel rd_r)$ and $CK_2 = h(CID_i \parallel CPW_i) \oplus h(rd_r \parallel PR_{key})$. Moreover, the hashing functions, including $PRK_j = h(SID_j \parallel S_{Key})$ and $PUK_j = PRK_j.P$, are eventually verified using $PRK_j = h(SID_j \parallel S_{Key})$ and $PUK_j = PRK_j.P$ to verify the session of PAD_i. Therefore, A_{dv} cannot infer a valid PAD_i session key without a presumption of $\{rd_r, PRKey_i, S_{Key}\}$. Hence, PPSA-PKV can resist a privileged insider attack.

4) $\mathcal{R}_2$ ***Resilience against Replay Attacks****:* Suppose A_{dv} exploits old captured messages $T_A \rightarrow SID_j, PUK_j$, $PAD_i \rightarrow \{DID_i, CID_i, CPW_i, rd_r, T_r\}$, and $CS_j \rightarrow \{LR_i, En_i, T_s\}$. A_{dv} cannot send an old authorized message because messages are validated by the timestamp: $(T_s^{'} - T_s) \leq \delta_{(T_s)}$ and $(T_c - T_c^{'}) \leq \delta_{(T_c)}$. Hence, PPSA-PKV can resist a replay attack.

5) $\mathcal{R}_3$ ***Resilience against User Masquerades****:* Suppose A_{dv} tries to forge login message $\{LR_i, En_i, T_s\}$ and tries a system login to CS_j with message modification $\{MR_i, T_c\}$. Since the parameter of $\{MR_i^{New}\}$ cannot be tampered with or verified upon authentic gateway access, A_{dv} cannot deduce original data message $MR_i^{New} = h(rd_s \| rd_i \| rand_r \| T_c)$ via fake decryption parameter $\{MR_i^{New}\}$. Hence, PPSA-PKV can repress a masquerading user attack.

6) $\mathcal{R}_4$ ***Resilience against Gateway Masquerades****:* Since A_{dv} is unaware of parameters such as rd_s, rd_i, and $PU_{Key}(X,Y)$ from the data exchange protocol, A_{dv} cannot exploit a gateway masquerade against PPSA-PKV. Hence, the proposed scheme can repress a gateway masquerade.

7) $\mathcal{R}_5$ ***Resilience against Offline Password Guessing****:* In most cases, A_{dv} tries to acquire system parameters over a public network. Assume A_{dv} obtains system parameters $\{CK_1, CK_2, rd_r, T_r, PU_{Key}, G_P(X,Y)\}$ from SC_R storage. In addition, A_{dv} tries to find new secret key $SK^{(}New^*)$ to compute $MR_i^{New} = h(rd_s \parallel rd_i \parallel rd_r \parallel T_c)$. However, a valid $SK^{(}New^*)$ cannot be computed because it is irretrievable from the devices of PAD_i. Thus, PPSA-PKV can be resilient to offline password guessing.

8) $\mathcal{R}_6$ ***Resilience against User Forgery****:* To forge the communication of any legal entity, the adversary requires parameters such as $\{DID_i, CID_i, CPW_i, rd_r, T_r\}$. To succeed in such an effort, A_{dv} tries to compute CID_i and CPW_i consisting of $CK_1, CK_2, rd_r, T_r, PU_{Key}$, and $G_P(X,Y)$. Since PU_{Key} is irretrievable, A_{dv} cannot infer or obtain a legal identity for AG_A in order

to derive a valid message request to authorize the session. Thus, PPSA-PKV can resist user forgery to preserve the identities of the users.

9) $\mathcal{R}_7$ ***Resilience against Gateway Forgery****:* Assume A_{dv} generates values for DID_i,CID_i, and CPW_i. As a result, A_{dv} claims that a legal request may be successfully generated. However, PPSA-PKV cannot permit anyone to generate a valid request without the proper associations for CK_1,CK_2, and PU_{Key}. Importantly, S_{key}, and PU_{Key} are hard to derive because they are associated with high-level security features. Thus, PPSA-PKV can resist gateway forgery.

10) $\mathcal{R}_8$ ***Resilience against Gateway User Tracking****:* As AG_A randomly manages the identities of DID_i,CID_i, and CPW_i for each session established via T_A, PAD_i cannot be tracked by any A_{dv}. In addition, S_{key} is irretrievable; thus, a legal request cannot be generated to track the user session.

11) $\mathcal{R}_9$ ***Resilience against Man-in-the-Middle****:* In most instances, A_{dv} attempts to overhear the transmitted messages processed between PAD_i and CS_j. However, the secret parameters rd_r,rd_i,En_i, and LR_i generated by PAD_i and CS_j are not contributed to any of the associated networks to infer their values. Moreover, secret key S_{Key} and its generated parameter PR_{Key} are also not shared with PAD_i and CS_j to form a legal authentication message. As a result of this, A_{dv} cannot compute a valid session c by using this attack.

12) $\mathcal{P}_3$ ***Property of Perfect Forward Secrecy****:* In most case, A_{dv} tries to obtain system parameters such as DID_i,CID_i, and CPW_i. Moreover, A_{dv} might examine legal message requests such as $T_A \rightarrow \{SID_j,PUK_j\}$, $PAD_i \rightarrow \{DID_i,CID_i,CPW_i,rd_r,T_r\}$, and $CS_j \rightarrow \{LR_i,En_i,T_s\}$ to generate a valid session key, $SK_{New} = h(rd_s.rd_i.PU_Key(X,Y))$. As a rule, PAD_i generates a secret key PR_{Key} that generates a random number, rd_r. As a result, it is evident that A_{dv} cannot obtain legal values for CK_1,CK_2,rd_r,T_r,PU_{Key}, and $G_P(X,Y)$ to discover a secure session. Hence, the proposed PPSA-PKV protects perfect forward secrecy.

Through the procedures of the PPSA-PKV scheme, PAD_i entities can mutually endorse one another to access sensitive data via CS_j. Eventually, PAD_i accesses a patient's private information through CS_j via T_A. As a session key is securely shared among the communication entities, PPSA-PKV can achieve security (mutual authentication and session key agreement), block user and gateway masquerades, and repress privileged insider and replay attacks to meet the requirements of the mIoMT in B5G networks.

B. Formal Analysis using a Random Oracle Model

In this section, the proposed PPSA-PKV protocol considers a formal analysis using the random oracle model [52] to prove that the proposed protocol can provably be secure to meet the security requirements of the mIoMT environment. Also, proof of the protocol is described in Theorem 1 based on RoR to present the significance of session key agreement.

Participants- In the proposed PPSA-PKV, three different entities such as PAD_i, CS_j, and AG_A are utilized to represent i^{th} instances of the participants. It is assumed that A_D can intercept, modify, and remove the transmitted messages during the process of communication to hold the results of random oracle L_1 and L_2. To describe the capabilities of each participant $\mathcal{P}$, we have five oracle lists defining L_{h_1} and L_{h_2}: *store the results of random oracle*, L_1: *initial private keys of* S_D *returned by keyInitial*, L_2: *private key of* S_D *referring to current time by keyUpdate*, L_{sk}: *store the instances of session key*; and L_{ex}: *store the instances of information exchange between the communication entities*. Using this assumption, various computational queries are executed under simulated real-world attacks including *Execute*, *Corrupt-Device*, *Reveal*, *Send*, and *Test* to guarantee secure authentication. Initially, the oracle lists including L_1 and L_2 do not hold any instance of adversary A_D and target user T_U to choose PAD_i^*. The detailed descriptions of the query are as follows:

- h_1 *query* - uses A_D to inquire the identity of T_U i.e., PAD_i. If there is any entry in L_{h_1}, $\mathcal{P}$ transmits (PAD_i,ρ_{h_1}) to A_D. Otherwise, $\mathcal{P}$ chooses a random integer $r_r \in Z_q^*$ to find $\rho_{h_1} = h_1(PAD_i) = g^{bx} \in G_1$ and annexes the corresponding parameters (PAD_i,bx,ρ_{h_1}) to L_{h_1}. Lastly, $\mathcal{P}$ sends (PAD_i,ρ_{h_1}) to A_D.
- *keyInitial query* - obtains the identity of T_U i.e., PAD_i to check whether $PAD_i \neq PAD_i^*$ to verify the behavior of A_D. if the verification holds, then $\mathcal{P}$ sends its initial key $SK_{x,0}$ to A_D via an entry list L_1. Otherwise, $\mathcal{P}$ retrieves (PAD_i,bx,ρ_{h_1}) to compute $SK_{x,0} = (g^b)^x$. Lastly, it transmits $SK_{x,0}$ to A_D and annexes the parameters $(PAD_i,SK_{x,0})$ to L_1.
- *keyUpdate query* - obtains the query (PAD_i,t) to validate whether $PAD_i \neq PAD_i^*$ to retrieve $SK_{x,t}$ over the current time. If the retrievable item is available in L_2, then $\mathcal{P}$ obtains (PAD_i,bx,ρ_{h_1}) and (PAD_i,bx,t,ρ_{h_2}) from L_{h_1} and L_{h_2}, respectively. Lastly, $\mathcal{P}$ computes $SK_{x,t} = (g^b)^x$ and accordingly, sends the parameters $(PAD_i,t,SK_{x,t})$ to L_2.
- *Execute* ($P_{PAD_i^{T_1}}$,$P_{CS_j^{T_2}}$,$P_{AG_A^{T_3}}$) - A_D invokes *Execute* query to observe the transmitted messages transferred over an insecure channel between PAD_i, CS_j, and AG_A.
- *Corrupt-Device*($P_{PAD_i^{T_1}}$) - *Corrupt-Device* implies that A_D can observe the sensitive information stored in SC_R.
- *Send* ($(P^{(t)},Msg)$ - The *Send* query authorizes A_D to return the transmitted messages to $(P^{(t)}$ and notify the responses accordingly. While executing a query, $\mathcal{P}$ checks whether $PAD_i \neq PAD_i^*$ to perform a proper search stage. If the verification holds, then $\mathcal{P}$ searches PAD_i in L_2 to retrieve the parameter $SK_{x,t}$. In case of unavailability, $\mathcal{P}$ obtains $SK_{x,t}$ as shown in *keyUpdate query* and insert the source parameters $(PAD_i,t,SK_{x,t})$ to L_2. Then, $\mathcal{P}$ chooses the random integers $b,x \in Z_q^*$ to compute $S = g^{bx}$ and other relevant parameters as shown in PPSA-PKV. Lastly, it combines the source parameters of PPSA-PKV with L_{ex} and sends it to A_D.
- *Reveal* $(P^{(t)})$ - *Reveal* (P^t) implies that A_D supplies session key SK between PAD_i, CS_j, and AG_A. It is worth noting that SK can be safe and secure if A_D discloses SK using a query of Reveal (P^t). Additionally, upon receiving the queries, $\mathcal{P}$ searches the identity of T_U i.e., PAD_i in L_{sk} to simulate the passive attacks. If not existing, $\mathcal{P}$ aborts the session.
- *Test* ($(P^{(t)})$ - A coin f_c is fairly tossed before the game begins. The outcome of a query is known only to A_D to determine its consistency. If A_D regulates the query with a key freshness SK, then $(P^{(t)}$ will record SK for $f_c = 0$. Otherwise, $(P^{(t)}$ reports a null value $(\perp)$.

A_D conducts *Test* query for each participant which has a separate returned value to check the consistency of the random bit f_c via *Test* query. In order to win the game, A_D checks whether $f_c^{'}$ is equaled to f_c. Additionally, this query is accessible to the participants using a collision-resistant one-way hash function $h(.)$ which is also called a random oracle **Hash**.

Theorem 1 - A_D breaches the security mechanism and acquires the session key of the communication parties thereby the sensitive information of PAD_i is passively collected. Running the A_D is always influential over polynomial time to obtain:

$$A_{D_t} \leq \frac{q_h^2}{|Hash|} + \frac{q_p^2}{|ECA|} + 2.max\{Z_C.q_c^c, \frac{q_c}{2^{I_{MD}}}\} \quad (2)$$

where A_{D_t} is A_D in polynomial time, q_h, q_p, and q_c are the representation of *Hash*, *ECA*, and *Send* queries, respectively. Specifically, *Hash* and *ECA* i.e., $GP(X,Y)$ are the dimension of hash $h(.)$ and *ECA* $ECA(.)$ functions which analyze the privacy protection of the computing devices. Also, C_Z and c define the boundary condition to link its size with other computational parameters, and I_{MD} specifies the number of computing bits in Pwd_i of PAD_i.

Proof. Four games are executed i.e., $GM_{(i)}$ to prove the robustness of session key security where $i \in \{0,1,2,3\}$. In order to show the correctness of guessing any bit f_c, $Succ_{(A_D,i)}$ is represented. As a result, winning game $GM_{(i)}$ denotes its potential probability as $P_r[Succ_{(A_D,GM_{(i)})}]$. The key derivatives of $GM_{(i)}$ are as follows:

$GM_{(1)}$ **-** By virtue of this game, A_D performs a real-time simulated attack over PPSA-PKV. A_D randomly chooses a bit f_c at the outset of $GM_{(0)}$. Consequent to this assumption, we acquire:

$$A_{D_t} = |2.P_r[Succ_{(A_D,GM_{(1)})}] - 1| \quad (3)$$

$GM_{(2)}$ **-** Using this game, A_D drives the query *Execute* $(P_{PAD_i^{T_1}}, P_{CS_j^{T_2}}, P_{AG_A^{T_3}})$ to overhear the transmitted messages $\{DID_i, CID_i, CPW_i, rd_r, T_r\}$, $\{SID_j, PUK_j\}$, and $\{LR_i, En_i, T_s\}$. Accordingly, A_D verifies whether the computed session key SK matches with a real one or not in order to perform the queries including *Reveal* and *Test*. According to PPSA-PKV, the session key of the communication parties is composed of $SK_{New} = h(rd_s.rd_i.PU_{Key}(X,Y))$. To obtain an authentic SK_{New}, A_D is required to fetch the public key of the associated group i.e., PAD_i and random numbers of PAD_i, CS_j, and AG_A. As a consequence, A_D cannot gain the probability of winning this game via the key establishment phase to derive a SK_{New}. Hence, $GM_{(0}$ and $GM_{(1)}$ are considered to be imperceptible to obtain:

$$P_r[Succ_{(A_D,GM_{(2)})}] = P_r[Succ_{(A_D,GM_{(1)})}] \quad (4)$$

$GM_{(3)}$ **-** To secure an authentic SK_{New}, A_D drives two queries successively *Hash* and *Send*. Accordingly, A_D modifies the transmitted messages to explore a few active attacks such as masquerade, denial-of-service, etc. However, the established communication between PAD_i, CS_j, and AG_A is well protected using $h(.)$ as it is already composed of the public key of the associated group and random numbers. On the other hand, A_D cannot derive any authentic key for the associated group without the random integers as they are computationally infeasible to solve. Hence, while applying the birthday paradox, we can derive:

$$|P_r[Succ_{(A_D,GM_{(3)})}] - P_r[Succ_{(A_D,GM_{(2)})}]| \leq \frac{2q_h^2}{2.|Hash|} \quad (5)$$

$GM_{(4)}$ **-** Similar to $GM_{(2)}$, A_D drives two queries *Send* and *ECA* as stated in Section IV. ECA i.e., $GP(X,Y)$ has its group key generator property to analyze its weakness over the confidence level of security in order to achieve user privacy and device protection. Hence, we can derive the output as:

$$|P_r[Succ_{(A_D,GM_{(4)})}] - P_r[Succ_{(A_D,GM_{(3)})}]| \leq \frac{q_p^2}{|ECA|} \quad (6)$$

$GM_{(5)}$ **-** In this game, A_D tries to acquire SK_{New} through *Corrupt-Device* query. Using *Corrupt-Device* query, A_D extracts a few significant parameters $\{CK_1, CK_2, rd_r, T_r, PU_{Key}, G_P(X,Y)\}$ stored in the memory of PAD_i where $CK_1 = h(CID_i \parallel CPW_i \parallel rd_r)$, $CK_2 = h(CID_i \parallel CPW_i) \oplus h(rd_r \parallel PR_{key})$, and $PU_{Key} = PR_{Key}.G_P(X,Y)$. Since A_D is not aware of any authentic key values of the associated group without the random integers, A_D must predict the system parameters to assign their extracted values. However, this prediction is computationally infeasible to compute the values CID_i, CPW_i, and rd_r, simultaneously. Hence, $GM_{(4)}$ and $GM_{(5)}$ are computationally imperceptible. While applying Zipf's law, we can derive:

$$|P_r[Succ_{(A_D,GM_{(5)})}] - P_r[Succ_{(A_D,GM_{(4)})}]| \leq max\{Z_C.q_c^c, \frac{q_c}{2^{I_{MD}}}\} \quad (7)$$

The above games $GM_{(1)}$ to $GM_{(5)}$ are successfully executed to infer a guessing bit f_c in order to secure the game as a result of this contest. Hence, we can derive the result as:

$$|P_r[Succ_{(A_D,GM_{(5)})}]| = \frac{1}{2} \quad (8)$$

From the above equations Eq. (2) and (3), we can derive:

$$\begin{aligned} \frac{1}{2}A_{D_t} &= |P_r[Succ_{(A_D,GM_{(1)})} - \frac{1}{2}]| \\ &= |P_r[Succ_{(A_D,GM_{(2)})} - \frac{1}{2}]| \end{aligned} \quad (9)$$

Using Eq.(6) and (7), we can derive:

$$\begin{aligned} \frac{1}{2}A_{D_t} = |P_r[&Succ_{(A_D,GM_{(2)})}] \\ &- P_r[Succ_{(A_D,GM_{(5)})}]| \end{aligned} \quad (10)$$

Applying the property of trigonometric inequality, we can derive:

$$\begin{aligned} \frac{1}{2}A_{D_t} &= |P_r[Succ_{(A_D,GM_{(2)})}] - P_r[Succ_{(A_D,GM_{(5)})}]| \\ &\leq |P_r[Succ_{(A_D,GM_{(2)})}] - P_r[Succ_{(A_D,GM_{(4)})}]| \\ &+ |P_r[Succ_{(A_D,GM_{(4)})}] - P_r[Succ_{(A_D,GM_{(5)})}]| \\ &\leq |P_r[Succ_{(A_D,GM_{(2)})}] - P_r[Succ_{(A_D,GM_{(3)})}]| \\ &+ |P_r[Succ_{(A_D,GM_{(3)})}] - P_r[Succ_{(A_D,GM_{(4)})}]| \\ &+ |P_r[Succ_{(A_D,GM_{(4)})}] - P_r[Succ_{(A_D,GM_{(5)})}]| \\ &\leq \frac{q_h^2}{2|Hash|} + \frac{q_p^2}{2|ECA|} + max\{Z_C.q_c^c, \frac{q_c}{2^{I_{MD}}}\} \end{aligned} \quad (11)$$

Lastly, multiplying the Eq.(10) by 2, we can derive the appropriate result:

$$A_{D_t} \leq \frac{q_h^2}{|Hash|} + \frac{q_p^2}{|ECA|} + 2.max\{Z_C.q_c^c, \frac{q_c}{2^{I_{MD}}}\} \quad (12)$$

Hence, *Theorem 1* is proven. ■

VI. Performance Analysis

In this section, the performance of the proposed PPSA-PKV and other existing schemes [25]–[32] are evaluated using a dedicated system setup to analyze the cost metrics namely computation and communication. Also, we present a real-time testbed and its detailed configurations using NS3.37 and Raspberry Pi 4 [Model B] to address the analytical solutions related to performance criteria (delay, throughput, and energy consumption) and prediction accuracy.

TABLE III: Computation efficiencies of PPSA-PKV and the related authentication schemes

Schemes	Login and Authentication Process		Total Cost	Execution Time (ms)
	Device	Cloud Server		
LSA-D2D [25]	$3T_{Hash}+2T_{(E/D)}+2T_{EM}+2T_{EC}$	$3T_{Hash}+2T_{(E/D)}+2T_{EM}+7T_{EC}$	$6T_{Hash}+4T_{(E/D)}+4T_{EM}+9T_{EC}$	2.624
LAS [26]	$8T_{Hash}+3T_{EC}$	$35T_{Hash}+7T_{EC}$	$43T_{Hash}+10T_{EC}$	2.137
EMFUAP-FS [27]	$9T_{Hash}+1T_{(E/D)}+1T_{EC}$	$6T_{Hash}+1T_{(E/D)}+7T_{EC}$	$15T_{Hash}+2T_{(E/D)}+2T_{EC}$	6.513
IAKP [28]	$18T_{Hash}+2T_{(E/D)}+8T_{EM}+5T_{EC}$			17.171
UC-PPAP [29]	$8T_{Hash}+18T_{EC}$			1.191
PPA-SC [30]	$22T_{Hash}+3T_{SM}$			3.197
PP-TFA [31]	$11T_{Hash}+3T_{BH}+10T_{E/D}$			9.811
PPA-HECC [32]	$4T_{HESM}$			1.92
The proposed PPSA-PKV	$7T_{Hash}$	$8T_{Hash}$	$15T_{Hash}$	0.627
T_{Hash}- One Way Hash Function; $T_{(E/D)}$- Symmetric Encryption/Decryption; T_{EM}- Elliptic Curve Point Multiplication; T_{EA}- Elliptic Curve Point Addition; T_{EC}- Exponential Computation; and T_{HESM}- Hyper-Elliptic Curve Scalar Multiplication				

A. Computation Analysis

In this analysis, we consider the system phases including login and authentication of the proposed PPSA-PKV and the other schemes to analyze their computation factors. In the analysis, the system phases applied the OpenSSL library between two end-users [53] to examine the cost factors of the attributes. To build a simulation environment, the end-user employed an Intel(R) Core(TM) i7-1165G7 processor integrated with 8GB RAM and a clock speed of 4.8GHz, while the server terminal had a Core i7-1355U with 16GB RAM at 2.3GHz. The end-user and server terminals were configured via an H3C S1024R Ethernet switch to guarantee that the end devices utilized a bandwidth $\approx$ 100Mbps to iterate the computation more than 100 times. According to the NIST [Anon] recommendation [40], we preferred to use the P-192 as the standard elliptic curve to regulate its message digest, and SHA-256 to perform cryptographic hashing.

Table III shows the computation efficiencies of PPSA-PKV and the related authentication schemes. Since PPSA-PKV has a lesser cost factor (0.627 ms), the execution time involved during login and authentication can further be reduced to handle the biological characteristics of I_{MD} faster in order to provide better transmission efficiency than other schemes [25]–[32], as shown in Fig.2a.

TABLE IV: Communication efficiencies for PPSA-PKV and the related authentication schemes

Schemes	Message Rounds	Total Cost (bits)
LSA-D2D [25]	4	2600
LAS [26]	4	8416
EMFUAP-FS [27]	4	5088
IAKP [28]	5	6784
UC-PPAP [29]	4	3840
PPA-SC [30]	5	2560
PP-TFA [31]	3	2144
PPA-HECC [32]	3	2496
The proposed PPSA-PKV	4	1248

B. Communication Analysis

In this analysis, we include a few key parameters, including the lengths for device identity, device password, sensor identity, temporary identity, timestamp, and sequence number (all at 128 bits), elliptic curve point multiplication (320 bits), plus the random number, the secret key, and the one-way hash function (each at 160 bits) to probe the transmitted messages effectively during authentication phase [54]. The proposed PPSA-PKV underwent four rounds of meessages, and the transmitted requests were $\{PR_{Key},\ PU_{Key}\}$, $\{SID_j,\ PUK_j\}$, $\{LR_i,\ En_i,\ T_s\}$, and $\{MR_i,\ T_c\}$ to totally consume an approximate cost of $[128+128+128+128+160+160+128+160+128=1248bits]$. Similarly, the message rounds of other schemes were evaluated to determine their transmission costs. LSA-D2D [25] had four message rounds, and the requests were $\{PID_i, rand_i, P_i, T_s, V_C, h(.)\}$, $\{V_C,\ P_i,\ T_p,\ \sigma_(eNB,\ p)\}$, $\{PID_i,\ PID_j,\ P_i,\ h(.)\}$, and $\{PID_j, ID_{Enb}, H(.), diff, Exp, \sigma_{Enb}\}$ reaching an approximate cost of $[128+128+128+128+320+160+320+128+160+128+128+128+128+160+128+128++160+160+160+160=2600bits]$.

LAS [26] had four message rounds, and the transmitted messages were $\{MID_i, MS_1, V_1, T_1\}$, $\{MS_2, V_2, NC_{k0}\}$, $\{MS_3, V_3\}$, and $\{MS_4, V_4\}$ to utilize $[37\times128+23\times160=8416bits]$. EMFUAP-FS [27] had four message rounds and the message requests were $\{X_i, D_0, D_1, M_1, n_i\}$, $\{D_3, M_2, X_i\}$, $\{M_3, D_5, X_j\}$, $\{M_4, D_5, X_j\}$ to use $[31\times128+5\times160+1\times320]=5088bits]$. IAKP [28] had five message rounds for requested messages $\{ID_i, RID_i\}$, $\{A,\ PID_i,\ N_i,\ T_u\}$, $\{A,\ B,\ PID_i,\ PID_j,\ N_i,\ L_j,\ T_u,\ T_f\}$, $\{C, Auth_i, Auth_j, T_c\}$, and $\{B, C, Auth_i, T_c\}$, spending $[33\times128+14\times160+\times320=6784bits]$. UC-PPAP [29] had four message rounds; messages were $\{N_G^1, \beta, j\}$, $\{N_G^2, \delta, \mu, PI_{SN}^1, T_H^2, C_A^1\}$, $\{N_G^1,\ op_1,\ op_2,\ T_H^4,\ C_A^2\}$, and $\{T_H^4,\ GA_2,\ C_A^2\}$ consuming $[15\times128+12\times160=3840bits]$. PPA-SC [30] had five transmission rounds during session key generation and authentication; the message requests were $\{B_1, B_2, B_3, T_1\}$, $\{B_2, B_4, B_5, B_6, T_2\}$, $\{B_7, B_8, T_3\}$, $\{B_9, B_10, B_11, T_4\}$, and $\{\sigma_{ij}, B_{12}, T_5\}$ involving $[12\times160+5\times128=2560bits]$. PP-TFA [31] had three message requests, and their computation parameters were $\{t_i, \gamma_i, ID_i, C_i\}$, $\{ID_i, ID_j, \gamma_i, \gamma_j, \delta\}$, and $\{t_j, \gamma_j, ID_j, C_j\}$ to consume $[3\times160+13\times128=2144bits]$. PPA-HECC [32] possessed three message requests, and the authenticated messages were $\{Auth_{tm}\}$, $\{Auth_{mt}\}$, and $\{Auth_{tu}\}$ to expend $[4\times320+2\times160+7\times128=2496bits]$. As shown in Table IV and Fig. 2b, the proposed PPSA-PKV had a communication cost lower than the other authentication schemes [25]–[32], improving transmission speeds between the communicating parties.

C. Experimental Results

This section discusses the key configuration of the simulation tool and statistical computing system i.e., for NS3 and SVM to compare the evaluation criteria of the proposed PPSA-PKV and other related schemes [25]–[32].

1) Simulation Setup: In this evaluation, an extensive simulation was conducted using NS3.37 under Ubuntu 22.04 to upgrade the network standard which uses a low-power wide area network (LPWAN) to connect S_D and W_S in mIoMT environment. This environment operated a licensed cellular brand to offer better network coverage (i.e., $1000\times1000m^2$) and durable battery life as opposed to other IoT technologies Moreover, NarrowBrand IoT developed LPWAN to enable a wide range of device connectivity

(a) Computation Cost (sec)

(b) Communication Cost (bits)

Fig. 2: Cost analysis of the proposed PPSA-PKV with other related authentication schemes

with low-bandwidth utilization to optimize system capacity and energy consumption. In order to examine the criteria, the built-in components such as W_S, S_G, and S_D were integrated via T_A. The integrated components utilize the authentication requests of the proposed PPSA-PKV (i.e., $\{DID_i, CID_i, CPW_i, rd_r, T_r\}$ and $\{CK_1, CK_2, rd_r, T_r, PU_{Key}, G_P(X,Y)\}$) and other related schemes to inspect the overall performance of mIoMT environment.

To examine the effectiveness of the PPSA-PKV with other related authentication schemes, we allocate 100 sensing units with $180m$ transmission distance to transmit the data packet either in a horizontal or in a vertical direction at the rate of $88kbps$. Also, during the login and authentication phase, we utilize a routing mechanism of ad hoc on-demand distance vector (AODV) via sink-node to process four different transmitted messages 72 bytes, 144 bytes, 128 bytes, and 56 bytes, respectively over a simulation period $3600sec$. To evaluate the transmission process under data traffic pattern (UDP/CBR), the communication entities W_S and S_D operate fixed access point as S_G whereby the flow of data transmission is regulated with a buffer size $80kbytes$ to validate the quality metrics such as delay, throughput, and energy consumption.

To load node positions, a sink node, device proximity, packet transmission, time interval, and energy consumption model, the simulation network initialized its configuration setup with sensing units (as stated in V). This modified model dealt with the consumption of energies during data transmission and reception which utilizes cryptographic functions and its operational command like *"BYka"* to the existing model of LoRaWAN. Two programming languages such as *C++* and *Python* are practiced to implement the traffic patterns and key agreement mechanisms constructed by a defined topology with the network simulation i.e., LoRaWAN. The network traffic utilizes the computing devices S_D to carry out the transmission of packets for every 5 second which uses a gateway S_G over the energy model to assess the consumption ratio according to Eq.(13).

$$E_C = (\sum_{i=1}^{n} E_{RX_i} + E_{MAC_i} + E_{AES_i}) + \sum_{j=1}^{n/m} E_{BK_j} \qquad (13)$$

where E_{RX_i} is the received data transmission, E_{BK_j} is the computation of *BYka* operation, E_{MAC_i} is computation of MAC, and E_{AES_i} is the encryption operation of MAC. Table V shows the important parameters utilized in NS3 simulation.

TABLE V: Important Parameters Utilized in NS3 Simulation

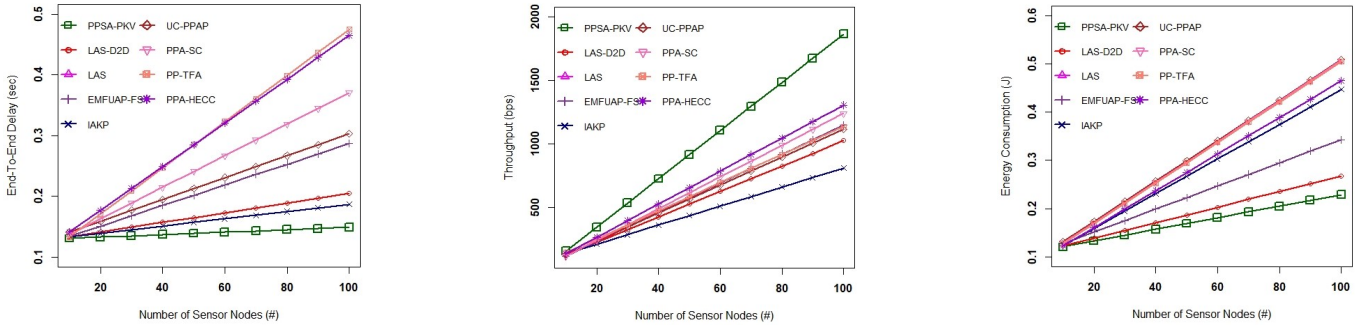
Parameter	Value
Simulation Area	$1000 \times 1000m^2$
Simulation Time	3600 sec
Initial Energy of a Sensor Node	10kJ
Supplied Voltage	3.3V
The power utilized per packet transmission	0.028 A
The power utilized per packet reception	0.0112 A
Number of System Gateway	1
Range of Transmission	180 (m)
Number of Sensor Nodes	100
Size of a Packet	1024 bytes
Flow of Data packets	4 packets
Flow Data Type	UDP/CBR
Rate of Data Transmission	88 Mbps
Size of a Queue	80 kB
Data Traffic Pattern	UDP/CBR
Routing Protocol	AODV

The computing scenario is structured using NS3.37 to signify three major roles:

1) **Smart Device** holds by the end-users to obtain sensitive information with proper data security.
2) **Sink-Node/Server** transfers the sensitive data via the LoRaWAN network to establish a secure data path.
3) **Medical Sensing Units** remotely monitor the activities of remote patients and doctors via an application system.
4) **Smart Gateway** acts as a standard interface to collect and aggregate the sensing data using I/O devices.

2) Comparative Analysis Using NS3: This section compare their examination results over the standard requirements of B5G networks including average response time, throughput, and energy consumption [55].

End-To-End (E2E) Delay is defined as the amount of time taken by the sensing unit to transmit its sensitive information from the source to the destination node. From Fig.3a, it is observed that while the connectivity of the sensing units grows exponentially, the delay time between source to destination is intensified to determine the best path to route the data packets. As a consequence, the proposed PPSA-PKV finds its authentic secret key S_{Key} prior to balancing the routing via proper device verification and also optimizes its cost efficiencies including computation and storage to regulate data traffic and device connectivity thereby the end-to-end delay is well managed to secure



(a) End To End Delay (sec) (b) Throughput Rate (bps) (c) Energy Consumption (J)

Fig. 3: Simulation Analysis Using NS3

the path routing using seamless accessibility. The quantitative analysis demonstrates that the proposed PPSA-PKV delivers less end-to-end delay $\approx 0.14sec$ even if the number of computing devices grows to 100. While analyzing the existing schemes [25]–[32] with a similar number of computing devices, they obtain an excessive end-to-end delay than the proposed PPSA-PKV to guarantee device authenticity.

Throughput Rate is defined as the time a process takes to deliver its message over a communication channel during a particular period. From the Fig.3b, it is stated that the proposed PPSA-PKV has a pre-deterministic secret key S_{Key} to provide seamless authentication with device verification whereby path routing is optimized with less transmission delay to improve the throughput rate of the communication systems. However, the existing schemes cannot determine any specific computation strategy to limit a few constraints of throughput rate including energy consumption and packet loss in order to obtain a secure session SK. The statistical analysis reveals that the proposed PPSA-PKV attains a high throughput rate of $1865bits$ while the number of computing devices is upscaled to 100. When inspecting the existing schemes [25]–[32] with a similar number of computing devices, they acquire less throughput rate than the proposed PPSA-PKV to minimize the rate of packet loss.

Energy Consumption Rate is defined as the amount of energy spent during the transmission of packets between source to destination in order to optimize the consumption ratio of the LoRaWAN network. To analyze the consumption ratio using Eq.13, the sensed information is transmitted periodically with reference to simulation time and a number of computing devices. Using this metric, the distance between the computing devices is assessed with high energy consumption. To address this issue effectively, the parameters such as power utilized per packet transmission and reception are considered with dynamic traffic patterns. As a result, the proposed PPSA-PKV considers route selection to access the key attributes such as node distance, residual energy, and dynamic mobility to improve the lifetime of the computing device, in consequence, the energy consumption ratio is well minimized than other related schemes [25]–[32] to eradicate data duplication as shown in Fig. 3c.

3) Learning Analysis Using SVM: This section shows a deployment scenario of a distributed computing system to demonstrate the constructive concept of the statistical learning approach using SVM. This approach considers a few significant attributes such as the time of the authentication requests, verification status, edge center identity, and location of the proposed PPSA-PKV and other existing approaches [25]–[32] to analyze their efficiency rate (i.e., prediction accuracy). To probe the accuracy rate based on the data samples precisely, the generative dataset is associated with the computing system. As a result, the key attributes of the proposed PPSA-PKV and other existing approaches can be correlated using the time of the authentication requests and verification of the identities. To form a set of labeling data with proper data preprocessing, the proposed and existing approaches perform proper data selection, training, and prediction. Also, they apply an appropriate Haversine formula (i.e., $H(\theta) = sin^2(\theta/2)$ to estimate the distance linked up with the average response time in processing the status of the authentication request under normal conditions.

Initially, the computing system sets the response time $\approx 2sec$ and accordingly, evaluates the processing requests ≈ 1000 in order to analyze the response of the CS_j which resulted in $\approx 1.72ms$. Additionally, the system prefers to use the linear kernel to classify the inseparable data using SVM and initiates a state boundary among the data instance (i.e., defining the class values) to produce an appropriate hyper-plane with different classes. To predict the genuineness of the learning model, the proposed and other existing approaches define a clear data separation with effective dimensional space whereby the SVM model makes the training set more memory efficient to return the prediction accuracy of the authentication requests. Algorithm 1 shows the learning-cum-training process of data acquisition to infer whether the request made in the authentication process is genuine or not. To substantiate the findings, the classifier analyzes the source attributes (i.e., user identities and the time of the authentication request) using d_e whereby the deviation of data values can be determined to abort the malicious request, preventing the attacks at distributed edge devices.

To investigate the results via the computing system, the SVM classifier was implemented using Python. This system utilized a dataset with 2K containing the information of the source attributes processed by the proposed PPSA-PKV and other existing approaches. Moreover, the learning model applied data associated with authentication i.e., user identities and timestamps to detect any intrusion or malicious activities that occurred or not. The metrics intended for the performance evaluation are as follows.

- *Accuracy* defines the learning measurements in identifying the relationship and patterns among the source variables based on appropriate data analysis [56].

$$Accuracy = \frac{(T_P + T_N)}{(T_P + T_N + F_P + F_N)} \tag{14}$$

where T_P is the true positive, T_N is the true negative, F_P is the false positive, and F_N is the false negative.

- *Precision* shows a modeling indicator to define the quality of the corrective prediction made by the target class.

$$Precision = \frac{T - P}{(T_P + F - P)} \tag{15}$$

- *Recall* defines the percentage of data samples making the learning model to identify the class of interest out of aggregated data.

$$Recall = \frac{T-P}{(T_P+F-N)} \tag{16}$$

- *F1-Score* evaluates the precision and recall scores to assimilate the accuracy of the learning model.

$$F1-Score = 2*\frac{(Precision*Recall)}{(Precision+Recall)} \tag{17}$$

Algorithm 1 Data Acquisition and Training using SVM

Require: Load of the cite dataset
Require: Load of the authentication dataset
Ensure: Train the learning model using SVM to predict the authentic data request processed edge data center
1: obtain the Metadata related to Authentication;
2: obtain the location of PAD_i;
3: obtain the verification status of the entities c_r;
4: obtain the time of the authentication request T_r;
5: find the estimated distance using Haversine d_e;
6: specify a target value i.e., 0 or 1;
7: partition the data into *Training* and *Testing*;
8: construct a confusion matrix;
9: apply SVM to classify and predict the rate of accuracy;
10: **if** *status==1* **then**
11: Authentic Data Request;
12: **else**
13: Malicious Data Request;
14: **end if**

To verify the correctness of the data, a proper confusion matrix was generated which makes the algorithm update the information related to identities and the time of the authentication requests during the authentication process of the proposed PPSA-PKV and other existing approaches. Moreover, this learning process exploits the value of d_e over the training samples to predict the genuine and malicious authentication request. To analyze the prediction behavior of the system, the generated value d_e of the proposed PPSA-PKV and other approaches considered active authentication requests. The findings, as shown in Table VI, prove that the behavioral schema i.e., biometric-based can provide better consistency than knowledge-based to deal with long-term authentication. It is also evident that the SVM model can effectively use behavioral-based profiling systems using machine-like standalone ML/AI processors to perform its computation at the edge node with limited energy consumption to achieve accurate decisions. Each node records the capacitive data i.e., d_e using identities and timestamps to examine modeling accuracy using SVM. From Table VI, we can observe that the proposed PPSA-PKV can achieve better evaluation results than other existing approaches [25]–[32] in maintaining the prediction accuracy as it relates the authentication measure with biometric-based. Importantly, this approach can be a part of an integrated security solution to meet the standard policies of security-by-design.

TABLE VI: Evaluation Results of SVM Learning Models

Schemes	Learning Model	Accuracy	Precision	Recall	F1-Score
LSA-D2D [25]	SVM	0.963	0.927	0.957	0.946
LAS [26]	SVM	0.972	0.931	0.943	0.956
EMFUAP-FS [27]	SVM	0.918	0.854	0.867	0.892
IAKP [28]	SVM	0.924	0.863	0.875	0.918
UC-PPAP [29]	SVM	0.947	0.906	0.927	0.971
PPA-SC [30]	SVM	0.891	0.857	0.874	0.881
PP-TFA [31]	SVM	0.979	0.935	0.948	0.963
PPA-HECC [32]	SVM	0.887	0.841	0.867	0.874
The proposed PPSA-PKV	SVM	0.994	0.972	0.987	0.991

VII. Conclusion and Future Works

In B5G-enabled healthcare, the mIoMT is emerging as a device-centric paradigm to build an automation process via real-time data collection integrating healthcare IT systems between the end devices and the cloud server. To preserve the sensing information over insecure channels, various authentication techniques (privacy preserving, three-factor, identity- and biometric-based) have been developed in the past. However, the existing techniques are application-centric, expose them to more computation overheads, and are also susceptible to potential threats (forgery and privileged-insider). Thus, in this paper, we applied two crypto-primitives such as a collision-free hash function and elliptic-curve arithmetic to design an authentication framework with device verification and privacy preserving. To prevent potential threats and to reduce cost factors, the proposed PPSA-PKV uses proper device verification with CS_j by using T_A in the authentication phase. This strategic approach helps the end users solve the security and privacy problems with insecure networks via the T_A. Moreover, as processed by a personalized application server, T_A identifies and verifies the source of data transmission, including message requests to authorize the connectivity. Comparative analysis showed that the proposed PPSA-PKV incurred lower overheads, including computation and communication, thus enhancing the performance of healthcare systems. Above all, while validating the authentication requests using SVM, the prediction accuracy of the proposed PPSA-PKV ensures a better detection ratio $\approx 99.4\%$ than other schemes to eliminate malicious authentication requests.

Security analyses both Informal and formal showed the security levels of the proposed PPSA-PKV in terms of mutual authentication and session key agreement. Moreover, performance analysis demonstrated that the proposed PPSA-PKV achieved better service efficiencies than other schemes, implying its practicality in resource-constrained networks. In the future, we will design a lightweight, privacy preserving model using federated learning to improve system efficiencies of the patient monitoring systems, including convergence rate and fairness.

References

[1] R. Pirmagomedov, D. Moltchanov, A. Samuylov, *et al.*, "Characterizing throughput and convergence time in dynamic multi-connectivity 5g deployments," *Computer Communications*, vol. 187, pp. 45–58, 2022.
[2] A. A. Pise, K. K. Almusaini, T. A. Ahanger, A. Farouk, P. K. Pareek, S. J. Nuagah, *et al.*, "Enabling artificial intelligence of things (aiot) healthcare architectures and listing security issues," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.
[3] G. Tsochev, "Some security problems and aspects of the industrial internet of things," in *2020 International Conference on Information Technologies (InfoTech)*, IEEE, 2020, pp. 1–5.
[4] J. Shen, T. Zhou, D. He, Y. Zhang, X. Sun, and Y. Xiang, "Block design-based key agreement for group data sharing in cloud computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 16, no. 6, pp. 996–1010, 2017.
[5] F. Wu, X. Li, A. K. Sangaiah, *et al.*, "A lightweight and robust two-factor authentication scheme for personalized healthcare systems using wireless medical sensor networks," *Future Generation Computer Systems*, vol. 82, pp. 727–737, 2018.
[6] M. A. Khan, I. Ullah, A. Alkhalifah, *et al.*, "A provable and privacy-preserving authentication scheme for uav-enabled intelligent transportation systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 5, pp. 3416–3425, 2021.
[7] T.-F. Lee, X. Ye, and S.-H. Lin, "Anonymous dynamic group authenticated key agreements using physical unclonable functions for internet of medical things," *IEEE Internet of Things Journal*, vol. 9, no. 16, pp. 15 336–15 348, 2022.
[8] J. Miao, Z. Wang, Z. Wu, X. Ning, and P. Tiwari, "A blockchain-enabled privacy-preserving authentication management protocol for internet of medical things," *Expert Systems with Applications*, vol. 237, p. 121 329, 2024.
[9] S. Das and S. Namasudra, "Lightweight and efficient privacy-preserving mutual authentication scheme to secure internet of things-based smart healthcare," *Transactions on Emerging Telecommunications Technologies*, e4716, 2023.

[10] N. Singh and A. K. Das, "Tfas: Two factor authentication scheme for blockchain enabled iomt using puf and fuzzy extractor," *The Journal of Supercomputing*, pp. 1–50, 2023.

[11] V. O. Nyangaresi, "Privacy preserving three-factor authentication protocol for secure message forwarding in wireless body area networks," *Ad Hoc Networks*, vol. 142, p. 103 117, 2023.

[12] B. Deebak and S. O. Hwang, "Intelligent drone-assisted robust lightweight multi-factor authentication for military zone surveillance in the 6g era," *Computer Networks*, vol. 225, p. 109 664, 2023.

[13] X. Li, J. Peng, M. S. Obaidat, F. Wu, M. K. Khan, and C. Chen, "A secure three-factor user authentication protocol with forward secrecy for wireless medical sensor network systems," *IEEE Systems Journal*, vol. 14, no. 1, pp. 39–50, 2019.

[14] M. Masud, G. S. Gaba, K. Choudhary, M. S. Hossain, M. F. Alhamid, and G. Muhammad, "Lightweight and anonymity-preserving user authentication scheme for iot-based healthcare," *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2649–2656, 2021.

[15] L. Xiong, F. Li, M. He, Z. Liu, and T. Peng, "An efficient privacy-aware authentication scheme with hierarchical access control for mobile cloud computing services," *IEEE Transactions on cloud computing*, vol. 10, no. 4, pp. 2309–2323, 2020.

[16] T. Mazhar, H. M. Irfan, I. Haq, *et al.*, "Analysis of challenges and solutions of iot in smart grids using ai and machine learning techniques: A review," *Electronics*, vol. 12, no. 1, p. 242, 2023.

[17] A. Rahman, K. Hasan, D. Kundu, *et al.*, "On the icn-iot with federated learning integration of communication: Concepts, security-privacy issues, applications, and future perspectives," *Future Generation Computer Systems*, vol. 138, pp. 61–88, 2023.

[18] S. Suhail and R. Jurdak, "Towards trusted and intelligent cyber-physical systems: A security-by-design approach," *arXiv preprint arXiv:2105.08886*, 2021.

[19] S. Namasudra, D. Devi, S. Choudhary, R. Patan, and S. Kallam, "Security, privacy, trust, and anonymity," *Advances of DNA computing in cryptography*, vol. 1, pp. 138–150, 2018.

[20] B. Deebak and F. Al-Turjman, "Secure-user sign-in authentication for iot-based ehealth systems," *Complex & Intelligent Systems*, pp. 1–21, 2021.

[21] A.-T. Fadi and B. D. Deebak, "Seamless authentication: For iot-big data technologies in smart industrial application systems," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 4, pp. 2919–2927, 2020.

[22] R. Lu, X. Lin, and X. Shen, "Spoc: A secure and privacy-preserving opportunistic computing framework for mobile-healthcare emergency," *IEEE transactions on parallel and distributed systems*, vol. 24, no. 3, pp. 614–624, 2012.

[23] U. Jain, S. Pirasteh, and M. Hussain, "Lightweight, secure, efficient, and dynamic scheme for mutual authentication of devices in internet-of-things-fog environment," *Concurrency and Computation: Practice and Experience*, vol. 35, no. 1, e7428, 2023.

[24] S. Namasudra, "Fast and secure data accessing by using dna computing for the cloud environment," *IEEE Transactions on Services Computing*, vol. 15, no. 4, pp. 2289–2300, 2020.

[25] A. Mohseni-Ejiyeh, M. Ashouri-Talouki, and M. Mahdavi, "An incentive-aware lightweight secure data sharing scheme for d2d communication in 5g cellular networks.," *ISeCure*, vol. 10, no. 1, 2018.

[26] M. Shuai, B. Liu, N. Yu, and L. Xiong, "Lightweight and secure three-factor authentication scheme for remote patient monitoring using on-body wireless networks," *Security and Communication Networks*, vol. 2019, 2019.

[27] D. Wang, P. Wang, and C. Wang, "Efficient multi-factor user authentication protocol with forward secrecy for real-time data access in wsns," *ACM Transactions on Cyber-Physical Systems*, vol. 4, no. 3, pp. 1–26, 2020.

[28] T.-Y. Wu, T. Wang, Y.-Q. Lee, W. Zheng, S. Kumari, and S. Kumar, "Improved authenticated key agreement scheme for fog-driven iot healthcare system," *Security and Communication Networks*, vol. 2021, pp. 1–16, 2021.

[29] M. Masud, G. S. Gaba, P. Kumar, and A. Gurtov, "A user-centric privacy-preserving authentication protocol for iot-ami environments," *Computer Communications*, vol. 196, pp. 45–54, 2022.

[30] S. A. Soleymani, S. Goudarzi, M. H. Anisi, A. Jindal, N. Kama, and S. A. Ismail, "A privacy-preserving authentication scheme for real-time medical monitoring systems," *IEEE Journal of Biomedical and Health Informatics*, 2022.

[31] L. Zhang, Y. Zhu, W. Ren, Y. Zhang, and K.-K. R. Choo, "Privacy-preserving fast authentication and key agreement for e-health systems in iot, based on three-factor authentication," *IEEE Transactions on Services Computing*, 2022.

[32] M. A. Khan, I. Ullah, A. Alkhalifah, *et al.*, "A provable and privacy-preserving authentication scheme for uav-enabled intelligent transportation systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 5, pp. 3416–3425, 2021.

[33] M. Usman, M. A. Jan, X. He, and J. Chen, "P2dca: A privacy-preserving-based data collection and analysis framework for iomt applications," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1222–1230, 2019.

[34] J. Al-Muhtadi, K. Saleem, S. Al-Rabiaah, M. Imran, A. Gawanmeh, and J. J. Rodrigues, "A lightweight cyber security framework with context-awareness for pervasive computing environments," *Sustainable Cities and Society*, vol. 66, p. 102 610, 2021.

[35] J. Kim and N. Park, "Lightweight knowledge-based authentication model for intelligent closed circuit television in mobile personal computing," *Personal and Ubiquitous Computing*, pp. 1–9, 2022.

[36] M. Dammak, O. R. M. Boudia, M. A. Messous, S. M. Senouci, and C. Gransart, "Token-based lightweight authentication to secure iot networks," in *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, IEEE, 2019, pp. 1–4.

[37] E. S. Babu, A. K. Dadi, K. K. Singh, S. R. Nayak, A. K. Bhoi, and A. Singh, "A distributed identity-based authentication scheme for internet of things devices using permissioned blockchain system," *Expert Systems*, vol. 39, no. 10, e12941, 2022.

[38] M. Ghahramani, R. Javidan, and M. Shojafar, "A secure biometric-based authentication protocol for global mobility networks in smart cities," *The Journal of supercomputing*, vol. 76, pp. 8729–8755, 2020.

[39] S. F. Aghili, H. Mala, M. Shojafar, and P. Peris-Lopez, "Laco: Lightweight three-factor authentication, access control and ownership transfer scheme for e-health systems in iot," *future generation computer systems*, vol. 96, pp. 410–424, 2019.

[40] C. Zhou, "An improved lightweight certificateless generalized signcryption scheme for mobile-health system," *International Journal of Distributed Sensor Networks*, vol. 15, no. 1, p. 1 550 147 718 824 465, 2019.

[41] S. Shafqat, H. Majeed, Q. Javaid, and H. F. Ahmad, "Standard ner tagging scheme for big data healthcare analytics built on unified medical corpora," *Journal of Artificial Intelligence and Technology*, vol. 2, no. 4, pp. 152–157, 2022.

[42] R. Amin, S. H. Islam, G. Biswas, M. K. Khan, and N. Kumar, "A robust and anonymous patient monitoring system using wireless medical sensor networks," *Future Generation Computer Systems*, vol. 80, pp. 483–495, 2018.

[43] A. M.-K. Wong, C.-L. Hsu, T.-V. Le, M.-C. Hsieh, and T.-W. Lin, "Three-factor fast authentication scheme with time bound and user anonymity for multi-server e-health systems in 5g-based wireless sensor networks," *Sensors*, vol. 20, no. 9, p. 2511, 2020.

[44] X. Jia, D. He, N. Kumar, and K.-K. R. Choo, "Authenticated key agreement scheme for fog-driven iot healthcare system," *Wireless Networks*, vol. 25, pp. 4737–4750, 2019.

[45] Y. Deng, Z. Zeng, K. Jha, and D. Huang, "Problem-based cybersecurity lab with knowledge graph as guidance," *Journal of Artificial Intelligence and Technology*, 2021.

[46] L. Xiao, X. Lu, T. Xu, W. Zhuang, and H. Dai, "Reinforcement learning-based physical-layer authentication for controller area networks," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2535–2547, 2021.

[47] F. S. Hassan and A. Gutub, "Improving data hiding within colour images using hue component of hsv colour space," *CAAI Transactions on Intelligence Technology*, vol. 7, no. 1, pp. 56–68, 2022.

[48] S. Al-Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi, "Internet of things (iot) communication protocols," in *2017 8th International conference on information technology (ICIT)*, IEEE, 2017, pp. 685–690.

[49] B. D. Deebak and A.-T. Fadi, "Lightweight authentication for iot/cloud-based forensics in intelligent data computing," *Future generation computer systems*, vol. 116, pp. 406–425, 2021.

[50] W. Mao, "A structured operational modelling of the dolev-yao threat model," in *Security Protocols: 10th International Workshop, Cambridge, UK, April 17-19, 2002. Revised Papers 10*, Springer, 2004, pp. 34–46.

[51] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," *IEE Proceedings-Information Security*, vol. 153, no. 1, pp. 27–39, 2006.

[52] R. Canetti, A. Jain, and A. Scafuro, "Practical uc security with a global random oracle," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, 2014, pp. 597–608.

[53] S. Amanlou, M. K. Hasan, and K. A. A. Bakar, "Lightweight and secure authentication scheme for iot network based on publish–subscribe fog computing model," *Computer Networks*, vol. 199, p. 108 465, 2021.

[54] B. D. Deebak, F. Al-Turjman, and L. Mostarda, "Seamless secure anonymous authentication for cloud-based mobile edge computing," *Computers & Electrical Engineering*, vol. 87, p. 106 782, 2020.

[55] L. Campanile, M. Gribaudo, M. Iacono, F. Marulli, and M. Mastroianni, "Computer network simulation with ns-3: A systematic literature review," *Electronics*, vol. 9, no. 2, p. 272, 2020.

[56] M. Sahu, N. Padhy, S. S. Gantayat, and A. K. Sahu, "Local binary pattern-based reversible data hiding," *CAAI Transactions on Intelligence Technology*, vol. 7, no. 4, pp. 695–709, 2022.

B D Deebak received his Ph.D. in computer science from SASTRA Deemed University, Thanjavur, India, in 2016. He is currently a Brain Pool Fellow with the Department of Computer Engineering at Gachon University, South Korea.

Seong Oun Hwang (Senior Member, IEEE) received his Ph.D. degree in computer science from the Korea Advanced Institute of Science and Technology, South Korea in 2004. He is currently a Full Professor at the Department of Computer Engineering, Gachon University, South Korea.