A Security-Enhanced Ultra-Lightweight and Anonymous User Authentication Protocol for Telehealthcare Information Systems

Dake Zeng, Akhtar Badshah, Shanshan Tu, *Senior Member, IEEE*, Muhammad Waqas, *Senior Member, IEEE*, and Zhu Han, *Fellow, IEEE*

***Abstract*—The surge in smartphone and wearable device usage has propelled the advancement of the Internet of Things (IoT) applications. Among these, e-healthcare stands out as a fundamental service, enabling the remote access and storage of patient-related data on a centralized medical server (MS), and facilitating connections between authorized individuals such as doctors, patients, and nurses over the public Internet. However, the inherent vulnerability of the public Internet to diverse security threats underscores the critical need for a robust and secure user authentication protocol to safeguard these essential services. This research presents a novel, resource-efficient user authentication protocol specifically designed for healthcare systems. Our proposed protocol leverages the lightweight authenticated encryption with associated data (AEAD) primitive ASCON combined with hash functions and XoR, specifically tailored for encrypted communication in resource-constrained IoT devices, emphasizing resource efficiency. Additionally, the proposed protocol establishes secure session keys between users and MS, facilitating future encrypted communications and preventing unauthorized attackers from illegally obtaining users' private data. Furthermore, comprehensive security validation, including informal security analyses, demonstrates the protocol's resilience against a spectrum of security threats. Extensive analysis reveals that our proposed protocol significantly reduces computational and communication resource requirements during the authentication phase in comparison to similar authentication protocols, underscoring its efficiency and suitability for deployment in healthcare systems.**

***Index Terms*—Authentication, Internet of Things, security, authenticated encryption, ASCON, secure communication.**

I. Introduction

THE advancement of the Internet of Things (IoT) has brought about a significant shift in human life, reshaping industries, and driving efficiency gains [1]–[4]. Its integration has particularly revolutionized healthcare systems, birthing concepts like Medicine 4.0 and Healthcare 4.0, leading to innovations in patient care and management [5]. However, as these systems rely heavily on interconnected devices and data sharing, ensuring security within smart healthcare environments becomes paramount.

D. Zeng and S. Tu are with the College of Computer Science, Beijing University of Technology, Beijing 100124, China (e-mails: pete.zeng@akuhome.com; sstu@bjut.edu.cn).

A. Badshah is with the Department of Software Engineering, University of Malakand, Dir Lower 18800, Pakistan (e-mail: akhtarbadshah@uom.edu.pk).

M. Waqas is with the Centre for Sustainable Cyber Security, School of Computing and Mathematical Sciences, Faculty of Engineering and Science, University of Greenwich, United Kingdom, and with the School of Engineering, Edith Cowan University, Perth, 6027 WA, Australia (e-mail: engr.waqas2079@gmail.com).

Z. Han is with the Department of Electrical and Computer Engineering at the University of Houston, Houston, TX 77004 USA, and also with the Department of Computer Science and Engineering, Kyung Hee University, Seoul, South Korea, 446-701 (e-mail: hanzhu22@gmail.com).

The seamless exchange of authenticated keys among remote users, such as medical professionals, across interconnected systems presents significant challenges. For instance, in remote patient monitoring, medical professionals need secure access to health data from wearable devices. Similarly, during emergency medical responses, authorized personnel must quickly access critical patient information without compromising security. Establishing robust security measures, particularly in key exchange, is essential for these IoT-enabled healthcare networks. The vulnerability of patient-related data to cyber threats in the public domain underscores the need for a secure and efficient authentication and key exchange protocol [6]–[8].

The existing literature on authentication and key exchange protocols in smart healthcare environments has seen notable contributions, ranging from cryptographic techniques to network protocols and access control mechanisms. Prior research has explored various cryptographic primitives and authentication protocols, such as identity-based cryptography [9], attribute-based encryption [10], and certificateless cryptography [11], each with its strengths and limitations. However, gaps persist in achieving a balance between security and efficiency in these dynamic, interconnected environments.

To address these challenges and bolster security measures within IoT-driven healthcare landscapes, our research focuses on enhancing authentication and key exchange specifically tailored for smart healthcare systems. We aim to rectify the limitations found in current protocols utilized for ensuring secure communications in healthcare settings. Numerous protocols exhibit vulnerabilities to a range of attacks, such as server and user impersonation, leakage of ephemeral secrets, denial-of-service incidents, de-synchronization, insider threats leveraging privileges, and a notable absence of crucial functionalities like user anonymity and mutual authentication. Moreover, cryptographic approaches such as public key cryptography and chaotic map-based user authentication and key exchange protocols demand significant computational resources, rendering them impractical for resource-limited IoT devices. Our research addresses this gap by aiming to enhance the security and efficiency of authentication and key exchange specifically tailored for smart healthcare systems. We aim to contribute novel approaches that not only bolster security measures but also streamline resource utilization, enabling seamless access to critical medical information while fortifying defenses against potential attacks. Through our work, we aspire to establish a robust framework for secure key exchange, ensuring the confidentiality and integrity of patient data in IoT-driven healthcare landscapes.

The main contributions of this paper are listed below:

- We introduce a resource-efficient authentication protocol

tailored for smart healthcare systems. This protocol leverages lightweight cryptography, utilizing an Authenticated Sponge CONstruction (ASCON) scheme alongside a hash function and XOR cryptographic primitives. It facilitates the establishment of a secure session key between the user and server following mutual authentication, ensuring encrypted communication for enhanced security between the user and the medical server. Additionally, our protocol ensures user anonymity and untraceability throughout the authentication and key exchange phases.

- We employ the real-or-random (ROR) model as a formal analysis technique to thoroughly evaluate the session key security in our protocol. Additionally, through an in-depth informal security assessment, we demonstrate the protocol's ability to withstand various potential attacks and provide numerous functional features.
- We implemented the related security operations/primitives on two different hardware platforms: a resource-constrained device (Raspberry Pi 4 with 2 GiB RAM, Raspberry Pi OS, 32-bit) for the user device, and a more resource-rich device (Intel® Core™ i5-8300H CPU @ 2.30GHz, 8 GiB RAM, Windows 10.22H2, 64-bit) for the medical server. We then measured the experimental execution times for each primitive to compare the performance of our proposed protocol with other existing state-of-the-art protocols. The results indicate that our protocol is computationally competitive when compared to these related protocols.

The subsequent sections of this paper are organized as follows: Section II provides an extensive review of the literature, while Section III presents the system overview and background knowledge. Section IV elaborates on the proposed protocol, followed by a thorough security analysis in Section V. Section VI evaluates the efficiency and effectiveness of the protocol. Lastly, Section VII presents concluding remarks to summarize the key findings of this study.

II. Related Work

In recent years, there has been extensive research on authentication and key exchange protocols in telehealthcare [12]–[17] based on passwords and smartcards. However, many of these protocols exhibit certain limitations. Firstly, both passwords and smartcards are susceptible to being forgotten, lost, stolen, or replicated. Secondly, if authorized users share their passwords and smartcards with unauthorized users, the system lacks the means to identify the legitimate user. Thirdly, certain protocols [12], [13], necessitate the server's retention of a password table, thereby exposing vulnerabilities to potential security breaches like password disclosure, stolen verifiers, and server-spoofing attacks. Additionally, the typically low entropy of user passwords renders them vulnerable to offline password guessing attacks. To bolster security, the inclusion of biometric characteristics as a third factor in designing robust authentication protocols is employed. The amalgamation of these three factors fortifies resistance against guessing, forgetting, stealing, and duplication issues [18], thereby overcoming the weaknesses inherent in two-factor schemes. Given the numerous advantageous properties offered by the three factors, multiple authentication and key exchange protocols based on three factors have been proposed for the telehealthcare environment [19]–[26].

In the development of a robust three-factor (password, smartcard, and biometric data) authenticated key agreement scheme, a variety of cryptographic algorithms and mechanisms were strategically integrated. This approach, designed for the key agreement scheme, incorporates one-way hash functions, chaotic maps, elliptic curve cryptography (ECC), the Rivest-Shamir-Adleman (RSA) cryptosystem, and fundamental operations such as XOR and concatenation. ECC emerged as a compelling choice due to its ability to provide security levels equivalent to RSA while employing smaller key sizes. This inherent characteristic translated to reduced power consumption, minimized bandwidth requirements, and decreased computational overhead [27]. Moreover, the computational efficiency of chaotic map operations surpassed that of ECC and RSA counterparts. This superiority positioned chaotic maps as an advantageous solution, especially in scenarios requiring heightened computational performance. Additionally, the incorporation of physical unclonable functions (PUFs) added an innovative layer leveraging inherent hardware variations, fortifying security against potential attacks [28], [29]. Furthermore, the use of authenticated encryption with associated data (AEAD) schemes ensured confidentiality, integrity, and authentication of transmitted data, thereby enhancing the overall security architecture, particularly in the context of telecare medicine information systems [30].

Awasthi and Srivatava [19] initially introduced a lightweight three-factor authenticated key agreement protocol for telehealthcare information systems that utilized symmetric key encryption and hash functions. However, Mishra *et al.* [20] identified security vulnerabilities in this protocol. These included susceptibility to password guessing attacks, failure to detect incorrect inputs during password changes that could lead to subsequent denial-of-service attacks. Additionally, Tan's analysis [21] further highlighted deficiencies in [19], exposing vulnerabilities to reflection attacks, lack of user anonymity, and insufficient three-factor security. Tan proposed an improved protocol, but subsequent examination [22] revealed weaknesses to replay attacks and denial-of-service attacks. Yan *et al.* [23] endeavored to rectify these vulnerabilities by presenting a new protocol claimed to be resilient against various attacks. However, Mishra *et al.* [24] contested this assertion, asserting vulnerabilities to offline password guessing attacks, inefficiencies in login and password updating, and a deficiency in user anonymity within Yan *et al.*'s design. In response, they proposed an authentication protocol fortified by hash functions and nonces to enhance security. Further scrutiny by Sarvabhatla *et al.* [25] unearthed security flaws in Mishra *et al.*'s protocol [24], specifically vulnerabilities to offline identity guessing attacks and user impersonation attacks. Subsequently, Amin and Biswas [26] raised concerns about Mishra *et al.*'s protocol [24], highlighting vulnerabilities to server impersonation attacks, session key computation attacks, and smart card theft attacks.

Kumari *et al.* [31] introduced an authentication protocol leveraging ECC to ensure secure access to medical server

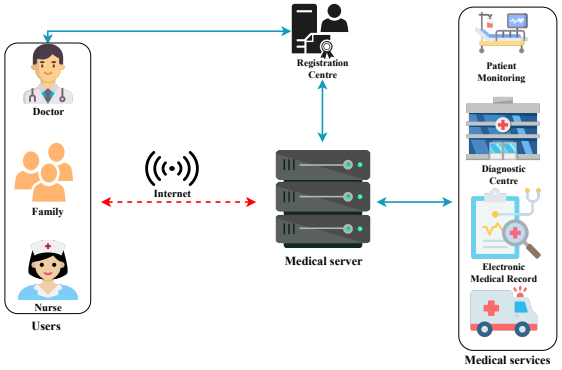


Fig. 1: Telehealthcare system network—users communicate with the medical server over insecure channels (red), while other connections are secure (blue).

information. However, this protocol exhibits vulnerabilities to password guessing, smart card/device loss, privilege insider breaches, user impersonation, and de-synchronization attacks. Additionally, it lacks provisions for user anonymity. Khatoon *et al.* [32] proposed a telehealthcare-based authentication protocol utilizing user bi-linear-pairing. Despite this, their approach remains susceptible to user impersonation and privileged insider attacks, while also lacking support for user anonymity. Tanveer *et al.* [30] devised a resource-efficient authentication protocol for telehealthcare information systems, employing ASCON and hash functions. However, their protocol is susceptible to de-synchronization attacks.

III. Background Knowledge

In this section, we introduce the network and threat models, along with outlining the design objectives and pertinent cryptographic foundations.

A. Network Model

The network model illustrated in Fig. 1 serves as the foundational framework for the proposed protocol. It comprises essential components: the Registration Center (RC), Medical Server (MS), and users (UR_i where $i = 1, 2, 3, \ldots, n$), where 'n' denotes the total number of users. These users include medical professionals such as doctors, nurses, or authorized family members seeking access to information stored within the MS. The RC plays a crucial role in granting access to the MS and UR_i. It deploys the MS and oversees the registration process for UR_i. This access enables medical centers to track patient records and utilize various services. Additionally, the MS stores sensitive registration details related to UR_i and patient health data linked to the patient monitoring system. To ensure secure communication between the MS and UR_i, the establishment of a secure mechanism is necessary for UR_i to access data within the MS. While the initial registration with the RC is assumed to occur over a secure channel, subsequent communication with the MS relies on the proposed authentication and key exchange protocol, which provides the necessary security without requiring a pre-established secure channel. This protocol is therefore an ideal choice for securing UR_i access to the system's resources. The confidential data stored within the system can be securely accessed using this protocol, which serves as a secure means to grant access within the required system.

B. Threat Model

Establishing an appropriate threat model is crucial as the foundation for the proposed protocol. In this context, the Dolev–Yao (DY) threat model [33] is used for the suggested authentication and key exchange protocol. According to this model, an adversary $\mathcal{A}$ can perform various operations on messages during communication, including intercepting, dropping, modifying, and replaying. In the proposed network model, the smart device, which allows users to connect to the medical server for telehealthcare services, is considered an untrusted entity. An adversary $\mathcal{A}$ might steal this device and potentially extract the secret credentials stored in its memory card or smart card [34]. Moreover, the MS (assumed to be in a secure environment) is physically inaccessible to $\mathcal{A}$. However, an insider adversary $\mathcal{A}$ with unauthorized access could compromise the integrity of the MS by retrieving sensitive information from its database. This unauthorized access might allow $\mathcal{A}$ to perform various malicious actions on behalf of a specific user. In addition to the DY model, we also consider the Canetti–Krawczyk (CK) threat model [35]. According to the CK-adversary model, the adversary $\mathcal{A}$ can compromise ephemeral information such as session states and session keys. This model is deemed more effective than the DY model in certain contexts, providing a more comprehensive framework for formulating an authentication protocol and considering various adversary capabilities and behaviors.

C. Cryptographic Preliminaries

In this subsection, we present the essential cryptographic primitives employed in the design of the proposed protocol.

1) Fuzzy Extractor

The Fuzzy Extractor (FE) plays a crucial role in cryptography by facilitating the creation of a distinct secret key from a user's biometric data [36]. It comprises two core functions:

1. **Biometric Key Generation ($Gen(\cdot)$):** This probabilistic function, $Gen(\cdot)$, takes BIO_{UR} (user's biometric information) as input and produces a unique key BK within the range $[0, 1]^{l_{BK}}$. Here, l_{BK} denotes the key's length, and RP represents the reproduction parameter. The probabilistic nature of BK is essential for ensuring that the generated key maintains security and robustness despite the imperfections or noise in the biometric data.

2. **Reproduction ($Rep(\cdot)$):** $Rep(\cdot)$ is a deterministic function that regenerates the key BK using BIO'_{UR} (biometric data provided at login) and RP, ensuring that the difference $|BIO_{UR} - BIO'_{UR}| \leq E_t$, where E_t is the acceptable error tolerance.

These functions, $Gen(\cdot)$ and $Rep(\cdot)$, within the FE framework, are integral in securely generating and reproducing unique cryptographic keys derived from biometric data. The probabilistic aspect of $Gen(\cdot)$ ensures that the process can handle variations and noise in biometric input while preserving the security of the generated key.

2) ASCON

Authenticated Sponge CONstruction (ASCON) is a lightweight AEAD scheme, extensively utilized to meet the computational demands of resource-constrained devices and high-performance computing systems [37]. ASCON is favored for hardware deployment due to its ability to minimize energy consumption and resist side-channel attacks. Moreover, its effectiveness and resilience against various forms of attacks have been widely acknowledged. The selection of ASCON as the optimal solution is driven by specific attributes, including its lightweight design, resistance to multiple attacks, and suitability for resource-limited environments. ASCON employs a sponge construction methodology in its design, serving as a versatile cryptographic structure used for various cryptographic functions. Specifically, the sponge construction comprises a fixed-length internal state and two distinct phases: the absorption phase and the squeezing phase.

1) **Absorption Phase:** During this phase, ASCON initializes an internal state to zero or a fixed initial value. Subsequently, the input data is divided into fixed-size blocks, each of which is combined with the rate portion of the internal state via an XOR operation. After absorbing each block, ASCON updates the internal state using a custom permutation function, known as the Substitution-Permutation Network (SPN).
2) **Squeezing Phase:** Once all input data has been absorbed, the internal state transitions to the squeezing phase. In this process, the output data (such as ciphertext or hash value) is squeezed from the rate portion of the internal state, one block at a time, and the permutation function is reapplied to ensure the output data's security and randomness, continuing until the desired output length is achieved.

Notably, the sponge construction allows the capacity and rate portions to be adjusted according to specific requirements, balancing security and performance, which makes ASCON particularly suitable for resource-constrained environments. The encryption and decryption processes of ASCON, based on the sponge construction, are detailed as follows:

The encryption process in ASCON, represented as $\langle CT, \text{TAG}\rangle = \text{E}_K(N, A, \text{PT})$, where $\langle CT, \text{TAG}\rangle$ signify the resulting ciphertext and authentication tag, K denotes the secret key, N the nonce, A the associated data, and PT represents the plaintext to be encrypted. Specifically, in practical applications, A can refer to any non-encrypted but authenticated data, such as identities, message headers, timestamps, and so on. During the encryption process, A is incorporated into the calculation along with N and PT to generate the specific ciphertext and authentication tags. Subsequently, any slight tampering with A and N during decryption will lead to decryption failure, thus ensuring the integrity and authenticity of the data.

The decryption procedure in ASCON, denoted as $\langle \text{PT}, \perp\rangle = \text{Dec}_K(N, A, CT, \text{TAG})$, yields $\langle \text{PT}, \perp\rangle$ containing the retrieved plaintext if the authentication succeeds, and $\perp$ represents an error if the authentication fails.

ASCON's attributes, including its lightweight design, resistance against a spectrum of attacks, and applicability in resource-constrained scenarios, position it as a compelling choice for systems emphasizing both efficiency and security.

TABLE I: Notations and their descriptions

Notation	Description
UR_i	i-th user
ID_i, PW_i	Identity and password of UR_i
PID_i	Pseudo identity of UR_i
TID_i	Temporal identity of UR_i
SP_i	Secret parameter of UR_i
$Param_i$	Parameters specific to user UR_i
$Param_k$	The k-th parameter in authentication
P^a, P^b	The two 128-bit parts of the 256-bit P
RP_i	Reproduction parameter of UR_i
TAG	Authentication parameter
$\mathbf{E}(\cdot)/\mathbf{D}(\cdot)$	ASCON encryption/decryption function
A	Associative data
N	Nonce
CT	Cipher text
PT	Plaintext
ΔT	Message delay tolerance
MS	Medical server
SK	Session key
RC	Registration center
TS	Timestamp
$\mathcal{A}$	Adversary
$h(\cdot)$	Cryptographic hash function
$\oplus$	Bitwise XOR operation
$\|$	Concatenation
rn	Random number

D. Design Objectives

The suggested protocol aims to achieve the following core design objectives:

- **Mutual authentication**: The suggested protocol ensures a robust mutual authentication between the user and MS during the authentication and key exchange phase. This authentication step validates the authenticity of involved entities and ensures the integrity of received messages.
- **Confidentiality**: The protocol guarantees the robust confidentiality of the session key generated via authentication and key exchange. This key also restricts access to authorized entities.
- **Untraceability**: Certain design measures are incorporated into authentication and key exchange communications to prevent adversaries from tracing the transmission.
- **Anonymity**: The protocol protects the real identities of communicating entities. Doing so safeguards the real identities against privacy breaches and potential risks.
- **Non-linkability**: Implementing measures to prevent the correlation of messages from the same source, mitigating potential credential extraction from multiple interactions.
- **Security Resilience**: The protocol must exhibit robustness against common security attacks encountered in communication environments.

IV. THE PROPOSED FRAMEWORK

The specifics of the proposed protocol are delineated within this section. The protocol encompasses four phases: user registration, authentication and key exchange, password update, and revocation. Table I furnishes a compilation of notations employed for elaborating on the protocol. Subsequent subsec-

tions expound on the operational mechanics of the proposed protocol.

A. Initialization Phase

The RC serves as the trusted authority overseeing the registration of URs and MS. Prior to deploying MS in the designated field, RC selects a distinct identity (ID_{MS}) and a confidential master key (K_{MS}) for MS. Furthermore, RC securely stores the credentials $\{ID_{MS}, K_{MS}\}$ within the tamper-resistant database of MS.

B. User Registration Phase

In this stage, UR_i, the user, undergoes registration with RC. This process is assumed to occur over a secure or private channel to ensure the confidentiality and integrity of the communication. This secure channel is necessary because the registration involves the exchange of sensitive information, such as confidential credentials, between UR_i and the RC. RC is responsible for assigning confidential credentials to UR_i as part of the user registration process. Before gaining access to the network resources, UR_i must authenticate itself with MS. RC executes essential procedures to enroll UR_i and ensure the completion of the registration process.

Step 1. Initially, U_i chooses an identity ID_i along with a specific password PW_i. After this selection, UR_i transmits a registration request message $< ID_i >$ to RC through a secure communication channel.

Step 2. Upon receiving the registration request from U_i, RC initiates the selection of a random secret parameter SP_i and a random number r_i. Subsequently, it calculates a pseudo identity for UR_i, computed as $PID_i = h(ID_i \parallel ID_{MS} \parallel K_{MS})$, where K_{MS} represents master secret key of MS. Following this, RC computes $Y_i = (SP_i \parallel r_i) \oplus h(PID_i \parallel ID_{MS} \parallel K_{MS} \parallel TS_i)$, where TS_i represents the registration timestamp. Subsequently, RC randomly generates a temporary identity TID_i for the user UR_i to facilitate identification during future authentication processes. Additionally, to counter de-synchronization attacks by an adversary $\mathcal{A}$—which could disrupt the synchronization of TID_i updates between the MS and UR_i—RC maintains both the new (TID_i^n) and old (TID_i^o) temporary identities of UR_i in the MS's database. Initially, at the time of registration, the old identity is set as $TID_i^o = null$, and the new identity is set as $TID_i^n = TID_i$. Therefore, even in extreme attack scenarios where the MS updates the temporary identity TID_i of UR_i but the synchronization message to UR_i is intercepted by $\mathcal{A}$, the MS can still recognize UR_i during subsequent re-authentication and complete the identity update, since a copy of the previous temporary identity is stored in its database. Ultimately, RC securely stores the secret credentials specific to UR_i $\{TID_i^o = null, TID_i^n = TID_i, PID_i, Y_i, TS_i\}$ in the MS's database and forwards $\{TID_i, PID_i, SP_i, r_i\}$ to UR_i via a secure channel.

Step 3. Following that, UR_i inputs their biometric data BIO_i into the smart device. The device uses a fuzzy extractor function $Gen(\cdot)$ to generate a stable biometric key BK_i and a public reproduction parameter RP_i, expressed as $(BK_i, RP_i) = Gen(BIO_i)$. Next, UR_i computes a hash value $V_i = h(ID_i \parallel PW_i)$ to secure the user's credentials. The device then encrypts a combination of a random value r_i and the reproduction parameter RP_i with V_i, resulting in $W_i = (r_i \parallel RP_i) \oplus V_i$. An encryption key K_i is derived by combining components of V_i with the biometric key BK_i, specifically as $K_i = (V_i^a \oplus V_i^b) \oplus BK_i$. Next, UR_i computes $\text{PT}_i = \{PID_i \| SP_i\}$, $N_i = r_i$, $A_i = ID_i$, and $\langle CT_i, \text{TAG}_i \rangle = \text{E}_{K_i}(N_i, A_i, \text{PT}_i)$. Finally, UR_i stores the parameters $\{TID_i, W_i, CT_i, \text{TAG}_i\}$ in its memory.

TABLE II: Stored parameters in the network entities.

Stored Parameters in User UR_i Smart Device
$\{TID_i, W_i, CT_i, \text{TAG}_i, Gen(\cdot), Rep(\cdot), h(\cdot)\}$
Stored Parameters in Medical Server MS
$\{\{((TID_i^o = null, TID_i^n = TID_i), PID_i, Y_i, TS_i) \mid i = 1, 2, \cdots, n\}, \{ID_{MS}, K_{MS}, h(\cdot)\}\}$

The parameters stored during the user registration phase are outlined in Table II.

C. Authentication and Key Exchange Phase

During this stage, UR_i conducts local authentication by verifying its secret credentials, and then proceeds to transmit an authentication and key exchange request message to MS. Once mutual authentication is achieved between UR_i and MS, they establish a session key to enable secure and unintelligible communication. Executing the following steps is essential to complete the authentication and key exchange process.

Step 1. UR_i inputs their identity ID_i', password PW_i', and biometric imprint BIO_i'. The smart device computes $V_i' = h(ID_i' \parallel PW_i')$, $(r_i^* \| RP_i^*) = W_i \oplus V_i'$, $BK_i' = Rep(BIO_i', RP_i^*)$, $K_i' = (V_i'^a \oplus V_i'^b) \oplus BK_i'$, $N_i = r_i^*$, $A_i = ID_i'$, and $\langle \text{PT}_i, \perp \rangle = \text{D}_{K_i'}(N_i, A_i, CT_i, \text{TAG}_i)$. Here, BK_i' represents the biometric key associated with UR_i, obtained using the $Rep(\cdot)$ function of FE. The parameter V_i' results from a hash operation performed on ID_i' and PW_i'. The secret encryption key K_i' is formed by concatenating $(V_i'^a \oplus V_i'^b)$ and BK_i', where $V_i'^a$ and $V_i'^b$ are derived from V_i'. Furthermore, N_i and A_i represent nonce and associative data, respectively, used for the ASCON decryption function $\text{D}(\cdot)$. If the verification of TAG_i fails, the system triggers $\perp$, prompting the smart device to terminate the process and generate a login failure message. Otherwise, the smart device retrieves the parameters as $\text{PT}_i = \{PID_i \| SP_i\}$ and proceeds with the process.

Step 2. Upon successful login and retrieval of parameters from PT_i, the smart device generates random numbers rn_1 and rn_2, and the current timestamp TS_1. It then computes $N_1 = rn_1$, $A_1 = TID_i$, $K_1 = SP_i$, and $\text{PT}_1 = rn_2$. Note that PT_1 is a random number to ensure session uniqueness and prevent attacks based on predictable values. Using the ASCON128a encryption function, the device generates the ciphertext CT_1 and the authentication tag TAG_1, where $\langle CT_1, \text{TAG}_1 \rangle =$

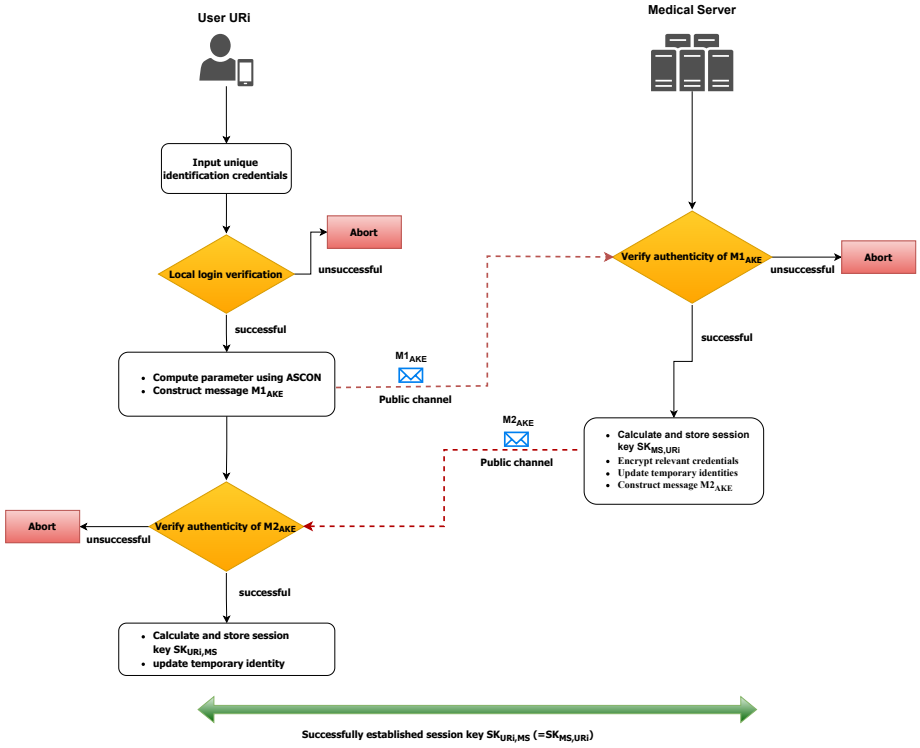


Fig. 2: Flowchart of the proposed authentication and key exchange phase.

$\mathrm{E}_{K_1}(N_1, A_1, \mathrm{PT}_1)$. The device then constructs the request message $M1_{AKE} = \{TID_i, CT_1, \mathrm{TAG}_1, rn_1, TS_1\}$ and transmits it to the MS through an unsecured channel. The inclusion of TAG_1 is essential for ensuring the integrity and authenticity of the transmitted data, allowing the MS to verify the message and safeguard against potential attacks.

Step 3. After the reception of $M1_{AKE}$, the MS verifies $|TS_1 - TS_1'| < \Delta T$. If so, the MS searches for TID_i. If either matches TID_i^o or TID_i^n, then retrieve the corresponding $\{PID_i, Y_i, TS_i\}$. The MS then computes $(SP_i^* \parallel r_i^*) = Y_i \oplus h(PID_i \parallel ID_{MS} \parallel K_{MS} \parallel TS_i)$, $N_2 = rn_1, A_2 = TID_i$, $K_2 = SP_i^*$, and $\langle \mathrm{PT}_1', \perp \rangle = \mathrm{D}_{K_2}(N_2, A_2, CT_1, \mathrm{TAG}_1)$. If the verification of TAG_1 fails, it triggers $\perp$ and aborts the procedure. Otherwise, the MS retrieves rn_2 from the plaintext PT_1'.

Step 4. Next, the MS generates the random numbers rn_3 and rn_4 and the current timestamp TS_2. The MS then computes $X_1 = h(rn_1 \parallel SP_i^* \parallel TS_1 \parallel TS_2)$, $K_3 = X_1^a \oplus X_1^b$, $N_3 = rn_3$, $A_3 = PID_i$, $\mathrm{PT}_2 = (rn_2 \parallel rn_4)$, $\langle C_1 \parallel C_2, \mathrm{TAG}_2 \rangle = \mathrm{E}_{K_3}(N_3, A_3, \mathrm{PT}_2)$, and $X_2 = (rn_3 \parallel rn_4) \oplus h(PID_i \parallel rn_2 \parallel TS_1 \parallel TS_2)$. Consequently, the MS updates the identities as $TID_i^o = TID_i$, $TID_i^n = C_1$ and stores the session key as $SK_{MS,UR_i} = C_2$. The MS then constructs the response message to UR_i as $M2_{AKE} : \{X_2, \mathrm{TAG}_2, TS_2\}$ and transmits it to the UR_i through an unsecured channel.

Step 5. After the reception of $M2_{AKE}$, the smart device verifies $|TS_2 - TS_2'| < \Delta T$. If so, it computes $X_3 = h(rn_1 \parallel SP_i \parallel TS_1 \parallel TS_2)$, $(rn_3^* \parallel rn_4^*) = X_2 \oplus h(PID_i \parallel rn_2 \parallel TS_1 \parallel TS_2)$, $K_4 = X_3^a \oplus X_3^b$, $N_4 = rn_3^*, A_4 = PID_i$, $\mathrm{PT}_3 = (rn_2 \parallel rn_4^*)$, and $\langle C_1' \parallel C_2', \mathrm{TAG}_3 \rangle = \mathrm{E}_{K_4}(N_4, A_4, \mathrm{PT}_3)$. Next, the smart device checks $\mathrm{TAG}_2 \stackrel{?}{=} \mathrm{TAG}_3$ and if it holds, the smart device updates $TID_i = C_1'$ and stores $SK_{UR_i,MS} = C_2'$ as the session key.

In summary, Fig. 2 provides a visual representation of the flow of the authentication and key exchange phase, which assists in understanding the proposed protocol. Additionally, Table III offers a detailed description of the authentication and key exchange process between UR_i and MS.

Remark 1. *In line with industry best practices for AEAD encryption, we have utilized a 128-bit key, nonce, and associated data configuration for our implementation. To ensure compatibility with this standard, the encryption key K_3 is derived from the 256-bit input X_1 by performing an XOR operation on its two halves, X_1^a and X_1^b. This method ensures that K_3 meets the necessary security parameters for* ASCON*'s AEAD functionality.*

D. Revocation Phase

In the event that a user UR_i experiences the loss of their smart device or card, UR_i follows a procedure to obtain a replacement device. To initiate the revocation process, UR_i sends a request containing their identity ID_i through a secure channel. The RC computes $PID_i = h(ID_i \parallel ID_{MS} \parallel K_{MS})$ and subsequently searches for this value within the MS database. If a match is found, the corresponding record is removed. After this step, UR_i initiates the new registration process. For this new registration process, the steps from *Step 1* to *Step 3* in the user registration phase are replicated.

TABLE III: Authentication and key exchange phase between UR_i and MS.

User UR_i	**Medical Server** MS
Input: ID_i', PW_i', BIO_i'; Compute: $V_i' = h(ID_i' \parallel PW_i')$, $(r_i^* \| RP_i^*) = W_i \oplus V_i'$, $BK_i' = Rep(BIO_i', RP_i^*)$, $K_i' = (V_i'^a \oplus V_i'^b) \oplus BK_i'$, $N_i = r_i^*$, $A_i = ID_i'$, $\langle \text{PT}_i, \perp \rangle = \text{D}_{K_i'}(N_i, A_i, CT_i, \text{TAG}_i)$; Abort if decryption yields $\perp$; $\text{PT}_i = \{PID_i \| SP_i\}$; Select: rn_1, rn_2, TS_1; Compute: $N_1 = rn_1, A_1 = TID_i, K_1 = SP_i, \text{PT}_1 = rn_2$, $\langle CT_1, \text{TAG}_1 \rangle = \text{E}_{K_1}(N_1, A_1, \text{PT}_1)$; $\xrightarrow[(UR_i \rightarrow MS)]{M1_{AKE}:\{TID_i, CT_1, \text{TAG}_1, rn_1, TS_1\}}$.	Check: $\|TS_1 - TS_1'\| < \Delta T$? Retrieve: PID_i, Y_i if TID_i matches either TID_i^o or TID_i^n; Compute: $(SP_i^* \parallel r_i^*) = Y_i \oplus h(PID_i \parallel ID_{MS} \parallel K_{MS} \parallel TS_i)$, $N_2 = rn_1, A_2 = TID_i$, $K_2 = SP_i^*$, $\langle \text{PT}_1', \perp \rangle = \text{D}_{K_2}(N_2, A_2, CT_1, \text{TAG}_1)$; Abort if decryption yields $\perp$; $\text{PT}_1' = rn_2$; Generate: rn_3, rn_4, TS_2; Compute: $X_1 = h(rn_1 \parallel SP_i^* \parallel TS_1 \parallel TS_2)$, $K_3 = X_1^a \oplus X_1^b$, $N_3 = rn_3, A_3 = PID_i, \text{PT}_2 = (rn_2 \parallel rn_4)$, $\langle C_1 \parallel C_2, \text{TAG}_2 \rangle = \text{E}_{K_3}(N_3, A_3, \text{PT}_2)$, $X_2 = (rn_3 \parallel rn_4) \oplus h(PID_i \parallel rn_2 \parallel TS_1 \parallel TS_2)$; Update: $TID_i^o = TID_i, TID_i^n = C_1$; Store: $SK_{MS,UR_i} = C_2$; $\xleftarrow[(MS \rightarrow UR_i)]{M2_{AKE}:\{X_2,\ \text{TAG}_2,\ TS_2\}}$.
Check: $\|TS_2 - TS_2'\| < \Delta T$? Compute: $X_3 = h(rn_1 \parallel SP_i \parallel TS_1 \parallel TS_2)$, $(rn_3^* \parallel rn_4^*) = X_2 \oplus h(PID_i \parallel rn_2 \parallel TS_1 \parallel TS_2)$, $K_4 = X_3^a \oplus X_3^b$, $N_4 = rn_3^*, A_4 = PID_i$, $\text{PT}_3 = (rn_2 \parallel rn_4^*)$, $\langle C_1' \parallel C_2', \text{TAG}_3 \rangle = \text{E}_{K_4}(N_4, A_4, \text{PT}_3)$; Check: $\text{TAG}_2 \stackrel{?}{=} \text{TAG}_3$ holds; Update: $TID_i = C_1'$; Store: $SK_{UR_i,MS} = C_2'$ as SK.	
Both UR_i and MS store $SK_{UR_i,MS} (= SK_{MS,UR_i})$.	

E. Password Update Phase

In order to reinforce the security measures of the protocol, UR_i is required to regularly update its password. This proposed protocol facilitates this functionality, necessitating UR_i to undertake the following crucial steps to ensure password updates.

Step 1. User UR_i inputs ID_i, PW_i^o, and biometric imprint BIO_i^o into the smart device. Then it computes $V_i = h(ID_i \parallel PW_i^o)$, $(r_i \| RP_i) = W_i \oplus V_i$, $BK_i^o = Rep(BIO_i^o, RP_i)$, $K_i^o = (V_i^a \oplus V_i^b) \oplus BK_i^o$, $N_i = r_i$, $A_i = ID_i$, and $\langle \text{PT}_i, \perp \rangle = \text{D}_{K_i^o}(N_i, A_i, CT_i, \text{TAG}_i)$. If the verification of TAG_i is successful, the smart device retrieves $\text{PT}_i = \{PID_i \parallel SP_i\}$ and prompts UR_i to enter new secret credentials.

Step 2. Following that, UR_i inputs their new biometric imprints BIO_i^n into the sensor of a smart device and input new password PW_i^n. Using the fuzzy extractor probabilistic generation function $Gen(\cdot)$, the smart device generates a secret biometric key BK_i^n and a public parameter RP_i^n as $(BK_i^n, RP_i^n) = Gen(BIO_i^n)$. Additionally, smart device computes $V_i^n = h(ID_i \parallel PW_i^n)$, $W_i^n = (r_i \parallel RP_i^n) \oplus V_i^n$, $K_i^n = (V_i^{n\,a} \oplus V_i^{n\,b}) \oplus BK_i^n$, $\text{PT}_i = \{PID_i \parallel SP_i\}$, $N_i = r_i$, $A_i = ID_i$, and $\langle CT_i^n, \text{TAG}_i^n \rangle = \text{E}_{K_i^n}(N_i, A_i, \text{PT}_i)$. Finally, smart device updates the parameters $\{TID_i, W_i^n, CT_i^n, \text{TAG}_i^n\}$ in its memory.

V. Security Analysis

This section showcases the resilience of the proposed protocol against different security threats via an informal analysis, while also establishing the security of session keys through a formal analysis using the ROR model.

A. Informal Security Analysis

This subsection outlines an informal assessment of the proposed protocol's security, emphasizing its robustness against diverse security threats.

1) Stolen Smart Device Attack

When $\mathcal{A}$ gains possession of the smart device or card belonging to UR_i, access to sensitive information becomes possible. This includes $\{TID_i, W_i, CT_i, \text{TAG}_i, Gen(\cdot), Rep(\cdot), h(\cdot)\}$ stored within the memory of the device or card. $\mathcal{A}$ could potentially conduct various attacks on behalf of UR_i. However, the information stored in UR_i's memory remains encrypted, preventing $\mathcal{A}$ from extracting essential data like $\{PW_i, ID_i, BIO_i\}$. Consequently, the proposed protocol effectively thwarts any attempt at launching an attack through a stolen smart device.

2) Password Guessing/Password Update Attack

In this attack scenario, the adversary ($\mathcal{A}$) aims to modify the secret credentials (e.g., $\{PW_i, ID_i, BIO_i\}$) after obtaining critical information, including $\{TID_i, W_i, CT_i, \text{TAG}_i, Gen(\cdot), Rep(\cdot), h(\cdot)\}$. $\mathcal{A}$ selects arbitrary credentials $\{PW_i^{Adv}, ID_i^{Adv}, BIO_i^{Adv}\}$ and computes $V_i^{Adv} = h(ID_i^{Adv} \parallel PW_i^{Adv})$, $(RP_i^{Adv} \parallel rn_i^{Adv}) = W_i \oplus V_i^{Adv}$, $BK_i^{Adv} = Rep(BIO_i^{Adv}, RP_i^{Adv})$, $K_i^{Adv} = (V_i^{Adv\,a} \oplus V_i^{Adv\,b}) \oplus BK_i^{Adv}$, and $\langle \text{PT}_i^{Adv}, \perp \rangle = \text{D}_{K_i^{Adv}}(r_i^{Adv}, ID_i^{Adv}, CT_i, \text{TAG}_i)$. Yet, $\mathcal{A}$ lacks the capability to decrypt without possessing the authentic secret credentials

of UR_i. Furthermore, predicting or generating the biometric keys proves to be challenging. As a result, the proposed protocol demonstrates effective resistance against password guessing/password update attacks.

3) Anonymity and Untraceability

The proposed protocol guarantees the anonymity of entities within the network. The process involves the exchange of two messages: $M1_{AKE} : \{TID_i, CT_1, \text{TAG}_1, rn_1, TS_1\}$ and $M2_{AKE} : \{X_2, \text{TAG}_2, TS_2\}$, necessary to complete the authentication and key exchange process. Even upon intercepting $M1_{AKE}$ and $M2_{AKE}$, it remains practically impossible for $\mathcal{A}$ to decipher the genuine or pseudonymous identity of UR_i from these transmitted messages. Consequently, the proposed protocol effectively prevents identity guessing attacks. Moreover, as both messages are dynamic, created with random numbers and current timestamps, $\mathcal{A}$ can't link messages from different authentication and key exchange sessions. Consequently, the protocol guarantees unlinkability, unobservability, and untraceability features.

4) Replay Attack

As outlined in Section IV-C, the authentication and key exchange process involves message exchanges that include the most recent timestamps. In this phase, entities validate received timestamps to ensure they comply with the acceptable time delay limit, denoted by ΔT. Consequently, the proposed protocol mitigates replay attacks.

5) Man-in-the-Middle Attack

In a man-in-the-middle attack, $\mathcal{A}$ intercepts the message $M1_{AKE} : \{TID_i, CT_1, \text{TAG}_1, rn_1, TS_1\}$ transmitted during the authentication and key exchange process. Subsequently, $\mathcal{A}$ generates modified messages, like $M1'_{AKE} : \{TID_i, CT'_1, \text{TAG}'_1, rn'_1, TS'_1\}$ and transmits $M1'_{AKE}$ to MS. Upon receipt of $M1'_{AKE}$, MS processes the received message from $\mathcal{A}$ while verifying the condition TAG'_1 to authenticate $M1'_{AKE}$. However, MS's verification of TAG1′ fails to authenticate $M1'_{AKE}$ due to $\mathcal{A}$'s inability to generate a valid message on behalf of UR_i without knowledge of its secret credentials, denoted as SP_i. Furthermore, $\mathcal{A}$ cannot create a valid $M2_{AKE} : \{X_2, \text{TAG}_2, TS_2\}$ without possessing the secret credentials. Consequently, the proposed protocol showcases resilience against man-in-the-middle attacks.

6) Denial-of-Service Attack

Within the proposed protocol, UR_i has the authority to dispatch the authentication and key exchange request to MS following local authentication completion. This preliminary local authentication phase serves as a protective measure, thwarting UR_i from inundating MS with an overwhelming volume of requests that might overload MS's message processing capabilities. Therefore, in the proposed protocol, the smart device assesses the successful verification of TAG_i to achieve local authentication, thereby strengthening the system against potential Denial-of-Service attacks.

7) Impersonation Attack

In an attempt to execute a user impersonation attack, $\mathcal{A}$ intercepts the message $M1_{AKE} : \{TID_i, CT_1, \text{TAG}_1, rn_1, TS_1\}$ transmitted during the authentication and key exchange process and crafts a modified message $M1'_{AKE}$. Subsequently, $\mathcal{A}$ disseminates $M1'_{AKE}$ to MS, attempting to deceive MS into believing that $M1'_{AKE}$ originates from a legitimate entity within the network. However, $\mathcal{A}$ fails to generate a valid $M1_{AKE}$ without knowledge of the secret credentials. Furthermore, $\mathcal{A}$ is unable to produce $M2_{AKE} : \{X_2, \text{TAG}_2, TS_2\}$ on behalf of MS without having access to the secret credentials. Consequently, the proposed protocol exhibits resilience against both user impersonation and MS impersonation attacks.

8) Ephemeral Secret Leakage Attack

Within the proposed protocol, the session key is derived as $X_3 = h(rn_1 \parallel SP_i \parallel TS_1 \parallel TS_2)$, $(rn_3^* \parallel rn_4^*) = X_2 \oplus h(PID_i \parallel rn_2 \parallel TS_1 \parallel TS_2)$, $K_4 = X_3^a \oplus X_3^b$, $N_4 = rn_3^*$, $A_4 = PID_i$, $\text{PT}_3 = (rn_2 \parallel rn_4^*)$, and $\langle C'_1 \parallel C'_2, \text{TAG}_3 \rangle = \text{E}_{K_4}(N_4, A_4, \text{PT}_3)$. Next, the smart device checks $\text{TAG}_2 \stackrel{?}{=} \text{TAG}_3$ and if it holds, the smart device updates $TID_i = C'_1$ and stores $SK_{UR_i,MS} = C'_2$ as the session key. It is apparent that $SK_{UR_i,MS} (= SK_{MS,UR_i})$ is formed utilizing both ephemeral secrets $(rn_1, rn_2, rn_3, rn_4, TS_1, TS_2)$ and long-term secrets (SP_i, PID_i). Therefore, compromising the session key requires knowledge of both types of secrets– ephemeral and long-term. Hence, the proposed protocol exhibits resistance against attacks aiming to exploit the leakage of ephemeral secrets.

9) De-Synchronization Attack

In our proposed protocol, during the initialization and user registration phases, unique real identities, pseudo identities, and secret keys are provided to users and the medical server. Specifically, the credentials $\{TID_i, W_i, CT_i, \text{TAG}i\}$ are stored in the memory smart device of user UR_i, while the corresponding credentials $\big\{\{((TID_i^o = null, TID_i^n = TID_i), PID_i, Y_i, TS_i) \mid i = 1, 2, \cdots, n\}, \{ID_{MS}, K_{MS}, h(\cdot)\}\big\}$ are retained on the medical server. By maintaining both the old and new temporal identities, our system enables retrieval of the previous values if the final acknowledgment message-exchanged between UR_i and MS-is blocked by $\mathcal{A}$ or lost due to time delays. Consequently, our protocol demonstrates resilience against de-synchronization attacks.

B. Formal Security Analysis

In this section, the security evaluation of the session key (SK) in the interactions between a user and a medical server within the proposed protocol p is conducted using the ROR model. This assessment addresses possible threats posed by both active and passive adversaries, represented as $\mathcal{A}$. Before delving into the assessment of the semantic security of SK in this interaction, fundamental concepts of the ROR model are introduced.

In this scenario, two primary entities exist: the user is denoted as $\Pi_{UR}^{t_1}$ and the medical server $\Pi_{MS}^{t_2}$, where $\Pi_{UR}^{t_1}$ and $\Pi_{MS}^{t_2}$ represent the t_1^{th} instance of the user and the t_2^{th} instance of the medical server, respectively. For a formal

TABLE IV: Queries and their purposes

Query	Purpose
$Send(\Pi^t, M)$	This query enables $\mathcal{A}$ to deliver a message M to Π^t and retrieve the subsequent response message
$CorruptUR(\Pi_{UR}^{t_1})$	This query allows $\mathcal{A}$ to access the confidential parameters from a compromised user smart device.
$Execute(\Pi_{UR}^{t_1}, \Pi_{MS}^{t_2})$	By executing this query, $\mathcal{A}$ can simulate an eavesdropping attack, intercepting the exchanged messages between participants $\Pi_{UR}^{t_1}$ and $\Pi_{MS}^{t_2}$.
$Test(\Pi^t)$	This query enables $\mathcal{A}$ to request the secret key (SK) from Π^t, and Π^t probabilistically responds with an unbiased flipped coin outcome b.
$Reveal(\Pi^t)$	Through this query, $\mathcal{A}$ discloses the SK generated between $\Pi_{UR}^{t_1}$ and $\Pi_{MS}^{t_2}$.

security analysis, Table IV outlines a range of queries applicable within the ROR model. These queries involve operations like "$Corrupt()$", "$Send()$", "$Execute()$", "$Test()$", and "$Reveal()$". Moreover, a hash function "$h(\cdot)$" acts as a random oracle, denoted as HO, within this context.

Let's introduce some key definitions that form the basis of our formal analysis:

Definition 1. *Let $\mathcal{A}$ run against the AEAD scheme within polynomial-time t, making at most Q_{ue} queries to an encryption/decryption oracle of length L_{en}. The "online chosen ciphertext attack (OCCA3)" advantage of $\mathcal{A}$ is expressed as:*

$$Adv_{p,\mathcal{A}}^{OCCA3}(Q_{ue}, L_{en}, t) \leq Adv_{p}^{OPRP-CPA}(Q_{ue}, L_{en}, t) + Adv_{p}^{INT-CT}(Q_{ue}, L_{en}, t), \quad (1)$$

where $Adv_{p}^{OPRP-CPA}(Q_{ue}, L_{en}, t)$ signifies $\mathcal{A}$'s advantage in the "online pseudo-random permutation chosen-plaintext" attack, and $Adv_{p}^{INT-CT}(Q_{ue}, L_{en}, t)$ represents $\mathcal{A}$'s advantage in the integrity of the ciphertext.

Definition 2. ***(Semantic security).*** *The advantage $Adv^p\mathcal{A}(t)$ of an adversary $\mathcal{A}$ operating in polynomial time t, targeting to compromise the semantic security in the user-medical server interaction to obtain the session key, is determined as $Adv_{\mathcal{A}}^{p}(t) = |2 \cdot Prob[b = b'] - 1|$. Here, b and b' represent the 'correct' and 'guessed' bits, respectively.*

Having established these foundational definitions, we now present the following theorem derived from the authentication and key exchange phase:

Theorem 1. *In the authentication and key exchange phase, $\mathcal{A}$ launches attacks against the interaction between the user $\Pi_{UR}^{t_1}$ and the medical server $\Pi_{MS}^{t_2}$ within polynomial time (t) to retrieve shared session key. The advantage of $\mathcal{A}$ in compromising the session key's security is approximately given by:*

$$Adv_{\mathcal{A}}^{p}(t) \leq \frac{Q_h^2}{|SHA|} + \frac{Q_s}{2^{lbk-1}|PD|} + 2 \cdot Adv_{ASCON,\mathcal{A}}^{OCCA3}(Que, L_{en}, t), \quad (2)$$

where Q_h, Q_s, SHA, PD and $Adv_{ASCON,\mathcal{A}}^{OCCA3}(Q_{ue}, L_{en}, t)$ represent hash queries, send queries, the range space of $h(\cdot)$, a password dictionary, and the advantage of $\mathcal{A}$ in compromising the security of an ASCON *scheme (refer to Definition 1), respectively.*

Proof. Suppose $\mathcal{A}$ engages in a sequence of five games $(Game_i^{\mathcal{A}} | i \in [0, 4])$ aimed at compromising the semantic security of SK. Here, Win_i signifies the probability of $\mathcal{A}$ achieving victory in $Game_i^{\mathcal{A}}$ within a time frame t. Each $Game_i^{\mathcal{A}}$ is detailed as follows:

$Game_0^{\mathcal{A}}$: $Game_0^{\mathcal{A}}$ depicts a situation where $\mathcal{A}$ imitates a real attack against p. The outcome of this game relies on the random flip of an unbiased coin. Subsequently, the semantic security of p is formally defined in *Definition* 2.

$$\text{Adv}_{\mathcal{A}}^{p}(t) = |2 \cdot \text{Prob}[Win_0] - 1|. \quad (3)$$

$Game_1^{\mathcal{A}}$: This particular game scenario involves an assumed eavesdropping attack against p, where $\mathcal{A}$ intercepts all messages exchanged between UR_i and MS throughout the authentication and key exchange process. Subsequently, $\mathcal{A}$ executes a sequence of queries, initiating with $Execute(\Pi_{UR}^{t_1}, \Pi_{MS}^{t_2})$, followed by "Test" and "Reveal" queries to validate the session key, denoted as $SK_{UR_i,MS} (= SK_{MS,UR_i})$. It's crucial to note that the computation of SK between UR_i and MS involves both short-term secrets and long-term secrets, as detailed in Section V-A8. The computational complexity inherent in deriving the session key makes it arduous for $\mathcal{A}$ to calculate. Consequently, the probability of winning $Game_1^{\mathcal{A}}$ remains unchanged compared to $Game_0^{\mathcal{A}}$. Therefore, the indistinguishability between $Game_0^{\mathcal{A}}$ and $Game_1^{\mathcal{A}}$ can be expressed as:

$$\text{Prob}[Win_1] = \text{Prob}[Win_0]. \quad (4)$$

$Game_2^{\mathcal{A}}$: Here, $\mathcal{A}$ aims to initiate an active attack by employing HO and $Send$ queries. $\mathcal{A}$ conducts multiple HO queries to search for collisions within $h(\cdot)$. Given that transmitted messages contain timestamps and random numbers, the likelihood of a collision using the $Send$ query is extremely low. Consequently, retrieving the secret parameters becomes an insurmountable task for $\mathcal{A}$. Utilizing the birthday paradox, we can express this as

$$|\text{Prob}[Win_2] - \text{Prob}[Win_1]| \leq \frac{Q_h^2}{2|SHA|}. \quad (5)$$

$Game_3^{\mathcal{A}}$: In this particular game scenario, $\mathcal{A}$ executes an active attack by employing the $CorruptUR(\Pi_{UR}^{t_1})$ query (as defined in Table IV). Through this method, $\mathcal{A}$ can retrieve specific information, namely $\{TID_i, W_i, CT_i, \text{TAG}i, Gen(\cdot), Rep(\cdot), h(\cdot)\}$, stored within the memory of a user's smart device via a power analysis attack. However, within p, the stored data is encrypted using the credentials $\{PW_i, ID_i, BIO_i\}$, where BIO_i (biometric key) is highly resistant to guessing and generation. Consequently, without knowledge of the valid credentials $\{PW_i, ID_i, BIO_i\}$, it is practically infeasible for $\mathcal{A}$ to extract the confidential credentials utilized in the authentication process. Furthermore, the length of the biometric key is denoted by $\frac{1}{2^{lbk}}$, where lbk represents the length of the biometric key. This effectively renders the probability of guessing BIO_i to be negligible. Additionally, the system allows only a restricted number of incorrect password attempts. Given these conditions, the following deduction can be made:

$$|\text{Prob}[Win_3] - \text{Prob}[Win_2]| \leq \frac{Q_s}{2^{lbk}|PD|}. \quad (6)$$

TABLE V: Functionality features analysis

↓**Feature/ Protocol** →	[30]	[31]	[38]	[39]	[40]	[41]	[42]	[43]	Our
$\mathcal{F}_1$: stolen smart device attack	✓	×	✓	✓	✓	✓	×	✓	✓
$\mathcal{F}_2$: password guessing attack	✓	×	×	✓	✓	✓	✓	✓	✓
$\mathcal{F}_3$: replay attack	✓	✓	✓	×	✓	✓	✓	✓	✓
$\mathcal{F}_4$: man-in-the-middle attack	✓	✓	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_5$: denial-of-service attack	✓	✓	✓	×	✓	✓	✓	✓	✓
$\mathcal{F}_6$: de-synchronization attack	×	×	✓	✓	✓	✓	×	×	✓
$\mathcal{F}_7$: impersonation attack	✓	×	×	×	✓	✓	✓	✓	✓
$\mathcal{F}_8$: anonymity	✓	×	✓	×	×	×	✓	×	✓
$\mathcal{F}_9$: untraceability	✓	✓	✓	×	×	×	✓	×	✓
$\mathcal{F}_{10}$: mutual authentication	✓	✓	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_{11}$: key agreement	✓	✓	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_{12}$: ESL attack	✓	✓	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_{13}$: password change	✓	✓	✓	✓	✓	✓	×	✓	✓
$\mathcal{F}_{14}$: revocation	✓	✓	✓	✓	✓	×	×	×	✓

Note: ✓: indicates the feature is available, ×: indicates the feature is unavailable

$Game_4^{\mathcal{A}}$: In $Game_4^{\mathcal{A}}$, $\mathcal{A}$ launches an active attack by eavesdropping on exchanged messages, $M1_{AKE}$: $\{TID_i, CT_1, \text{TAG}_1, rn_1, TS_1\}$ and $M2_{AKE}$: $\{X_2, \text{TAG}_2, TS_2\}$. Upon capturing these messages, $\mathcal{A}$ aims to extract secret parameters crucial for constructing the session key SK. However, these parameters are encrypted using ASCON, an AEAD scheme, preventing $\mathcal{A}$ from extracting the secret credentials from the encrypted information. Consequently, based on *Definition* 1, we derive:

$$|\text{Prob}[Win_4] - \text{Prob}[Win3]| \leq \text{Adv}^{OCCA3}_{ASCON,\mathcal{A}}(Q_{ue}, L_{en}, t). \tag{7}$$

The query denoted as "$Test$" is conducted by $\mathcal{A}$, involving the flipping of a fair coin, which ultimately determines the semantic security of SK following the completion of all games. Consequently, the probability of success in Win_4 is determined as follows:

$$\text{Prob}[Win_4] = \frac{1}{2}. \tag{8}$$

Thus, from (3) we derive

$$\frac{1}{2}\text{Adv}^{p}_{\mathcal{A}}(t) = \left|\text{Prob}[Win_0] - \frac{1}{2}\right|. \tag{9}$$

Utilizing (8) and (9), while considering (4), we arrive at:

$$\begin{aligned}\frac{1}{2}\text{Adv}^{p}_{\mathcal{A}}(t) &= |\text{Prob}[Win_0] - \text{Prob}[Win_4]| \\ &= |\text{Prob}[Win_1] - \text{Prob}[Win_4]|. \end{aligned} \tag{10}$$

Utilizing the widely recognized triangle inequality on (10), we obtain:

$$\begin{aligned}\frac{1}{2}\text{Adv}^{p}_{\mathcal{A}}(t) \leq\ & |\text{Prob}[Win_1] - \text{Prob}[Win_2]| \\ &+ |\text{Prob}[Win_2] - \text{Prob}[Win_3]| \\ &+ |\text{Prob}[Win_3] - \text{Prob}[Win_4]|. \end{aligned} \tag{11}$$

When (5), (6), and (7) into (11), we get

$$\begin{aligned}\text{Adv}^{p}_{\mathcal{A}}(t) \leq & \frac{Q_h^2}{|SHA|} + \frac{Q_s}{2^{lbk-1}|PD|} \\ &+ 2 \cdot \text{Adv}^{OCCA3}_{ASCON,\mathcal{A}}(Que, L_{en}, t). \end{aligned} \tag{12}$$

This result represents (2), concluding the proof. ■

VI. Comparative Analysis

In this section, we conduct a comprehensive comparison of the proposed protocol, assessing its security and functionality features, and the computation and communication overheads. We benchmark it against notable protocols devised by Tanveer *et al.* [30], Kumari *et al.* [31], Ostad *et al.* [38], Tseng *et al.* [39], Qiu *et al.* [40], Deebak and Hwang [41], Attir *et al.* [42], and Sumithra *et al.* [43]

A. Security and Functionality Features Comparison

When evaluating functionalities, our primary focus lies in mitigating threats such as smart device theft and password guessing, while also preventing replay attacks, man-in-the-middle attacks, denial-of-service attacks, de-synchronization, and impersonation attempts. Our protocol prioritizes ensuring anonymity, unlinkability, unobservability, untraceability, mutual authentication, and a robust key exchange. Moreover, our proposed protocol highlights forward secrecy, facilitates password updates, and integrates a revocation mechanism. For a comprehensive comparison between our proposed protocol and other relevant protocols (Tanveer *et al.* [30], Kumari *et al.* [31], Ostad *et al.* [38], Tseng *et al.* [39], Qiu *et al.* [40], Deebak and Hwang [41], Attir *et al.* [42], and Sumithra *et al.* [43]), Table V presents the comparison results indicating that our proposed protocol excels in simultaneously achieving these essential properties.

B. Computational Overheads Comparison

The proposed protocol's computational overheads, along with other state-of-the-art benchmark protocols, were assessed by computing the execution times for various cryptographic operations, as detailed in Table VII. These operations encompass ASCON encryption/decryption (T_{AS}), ECC point multiplication (T_{EM}), ECC point addition (T_{EA}), fuzzy extractor (T_{FE}), hash function (T_{H}), and symmetric encryption/decryption ($T_{\mathsf{S}}/T_{\mathsf{D}}$). Specifically, the computations were performed across different platforms. The user's smart device operated within constrained resources, using a Raspberry PI-4 with Raspberry Pi OS, 32-bit OS, and 2 GiB of RAM. On the other hand,

TABLE VI: Comparison of Computational Overhead

Protocol	User	Medical server	TE (ms)
Tanveer *et al.* [30]	$3T_{\mathsf{H}} + T_{\mathsf{FE}} + 4T_{\mathsf{AS}} \approx 16.66$	$3T_{\mathsf{H}} + 2T_{\mathsf{AS}} \approx 2.51$	19.17
Kumari *et al.* [31]	$12T_{\mathsf{H}} + 3T_{\mathsf{EM}} + 2T_{\mathsf{S}} \approx 36.91$	$9T_{\mathsf{H}} + 3T_{\mathsf{EM}} + 2T_{\mathsf{S}} \approx 6.868$	43.778
Ostad *et al.* [38]	$11T_{\mathsf{H}} + 2T_{\mathsf{EM}} + 2T_{\mathsf{EA}} \approx 33.97$	$8T_{\mathsf{H}} + 2T_{\mathsf{EM}} + 2T_{\mathsf{EA}} + 2T_{\mathsf{S}} \approx 6.258$	40.228
Tseng *et al.* [39]	$3T_{\mathsf{EM}} + T_{\mathsf{EA}} \approx 23.1$	$2T_{\mathsf{H}} + 5T_{\mathsf{EM}} + 3T_{\mathsf{EA}} \approx 6.1909$	29.29
Qiu *et al.* [40]	$8T_{\mathsf{H}} + 2T_{\mathsf{EM}} \approx 24.34$	$5T_{\mathsf{H}} + 2T_{\mathsf{EM}} \approx 4.03$	28.37
Deebak and Hwang [41]	$7T_{\mathsf{H}} \approx 9.45$	$8T_{\mathsf{H}} \approx 3.76$	13.21
Attir *et al.* [42]	$3T_{\mathsf{H}} \approx 4.05$	$5T_{\mathsf{H}} \approx 2.35$	6.4
Sumithra *et al.* [43]	$6T_{\mathsf{H}} \approx 8.1$	$6T_{\mathsf{H}} \approx 2.82$	10.92
Our Proposed	$3T_{\mathsf{H}} + T_{\mathsf{FE}} + 3T_{\mathsf{AS}} \approx 15.2$	$3T_{\mathsf{H}} + 2T_{\mathsf{AS}} \approx 2.51$	17.71

Note: TE (ms) indicates an estimated total execution time in milliseconds.

TABLE VII: Approximated execution time for various primitives (in milliseconds)

↓Primitive/ Device →	User smart device	Medical server
T_{AS}: ASCON	1.46	0.55
T_{EA}: ECC point addition	2.79	0.3503
T_{EM}: ECC point multiplication	6.77	0.84
$T_{\mathsf{FE}} \approx T_{\mathsf{EM}}$: Fuzzy extractor	6.77	0.84
T_{H}: Hash function	1.35	0.47
$T_{\mathsf{S}}/T_{\mathsf{D}}$: Symmetric encryption/decryption	0.20	0.059

the medical server utilized more abundant resources, employing an Intel® Core,™ i5-8300H CPU@2.30GHz, 8 GiB of RAM, Windows 10.22H2 OS with a 64-bit architecture. These experiments were conducted within the *PyCharm* software environment. Moreover, to ensure accuracy, the experiment was run 100 times, and the average time across these runs was computed for each cryptographic operation on both devices. Utilizing the data from Table VII, we conducted computations to evaluate the computational overheads of eight protocols. The findings are summarized and compared in Table VI. Notably, our protocol requires only 15.2 ms for the user device and 2.51 ms for the medical server, totaling 17.71 ms. This represents a significant reduction in computational overhead compared to existing benchmarks, with improvements of 7.6% over Tanveer *et al.* [30], 59.54% over Kumari *et al.* [31], 55.98% over Ostad *et al.* [38], 39.52% over Tseng *et al.* [39], and 37.61% over Qiu *et al.* [40]. Although our protocol incurs slightly higher computational overhead compared to the protocols proposed by Deebak and Hwang [41], Attir *et al.* [42], and Sumithra *et al.* [43], we believe this trade-off is justified by the substantial benefits it provides in terms of security and functionality features (see Table V).

C. Communication Overheads Comparison

In evaluating the communication overhead, we consider various elements with sizes specified as follows: random numbers, temporal identities, authentication tags, nonce, associated data, keys, hash function output, pseudo-identity, timestamps, and ECC points are $128, 128, 128, 128, 128, 128, 256, 256, 32$, and 320 bits, respectively. During the authentication and key exchange phase of the proposed protocol, two messages, namely

TABLE VIII: Communication Overhead Comparison

Protocol	Messages count	Total overhead (bits)
Tanveer *et al.* [30]	2	1232
Kumari *et al.* [31]	2	1628
Ostad *et al.* [38]	2	1696
Tseng *et al.* [39]	2	1024
Qiu *et al.* [40]	3	1440
Deebak and Hwang [41]	4	1216
Attir *et al.* [42]	2	1312
Sumithra *et al.* [43]	2	800
Our Proposed	2	960

$M1_{AKE} : \{TID_i, CT_1, \text{TAG}_1, rn_1, TS_1\}$ and $M2_{AKE} : \{X_2, \text{TAG}_2, TS_2\}$, have sizes of $\{128 + 128 + 128 + 128 + 32\} = 544$ bits and $\{256 + 128 + 32\} = 416$ bits, respectively. Therefore, the total communication overhead sums up to $\{544 + 416\} = 960$ bits, the lowest among some of the compared protocols, as illustrated in Table VIII. This results in a 22.2% reduction compared to Tanveer *et al.* [30], 40.9% compared to Kumari *et al.* [31], 43.4% compared to Ostad *et al.* [38], 6.3% compared to Tseng *et al.* [39], 33.3% compared to Qiu *et al.* [40], 21.1% compared to Deebak and Hwang [41], and 26.9% compared to Attir *et al.* [42]. Although our protocol incurs slightly higher communication overhead compared to the protocol proposed by Sumithra *et al.* [43], we believe this trade-off is justified by the substantial benefits it provides in terms of security and functionality features (see Table V).

VII. Conclusions

Ensuring robust security and preserving privacy remains paramount, especially within critical domains like telehealthcare information systems, where sensitive data transmission occurs across the public Internet. This paper introduces an authentication and key exchange protocol tailored specifically for telehealthcare, employing ASCON and hash functions. The protocol facilitates efficient user authentication and session key establishment with the medical server. Notably, it is designed for computational efficiency, addressing the resource

constraints of smart devices commonly used in telehealthcare information systems. Furthermore, our protocol enables secure access for doctors and nurses to information stored on the medical server. We formally establish the session key's security using the real-or-random oracle model. Additionally, through informal analysis, we demonstrate the protocol's resilience against various security threats such as de-synchronization attacks, device loss, and password guessing. The protocol also supports several essential properties, including perfect forward secrecy, anonymity, adaptive password changes, and revocation mechanisms. Moreover, a comparative analysis is conducted with existing state-of-the-art protocols, showcasing that our proposed protocol excels in security features, functionality, and minimizes communication and computation overheads. Although this study centered on theoretical and analytical evaluations, we acknowledge the necessity of real-time implementation to gauge practical performance in real-world settings. We have outlined real-time testbed evaluation as a crucial future step to confirm the protocol's effectiveness in actual telehealthcare environments.

ACKNOWLEDGEMENT

This work is partially supported by NSF ECCS-2302469, Toyota. Amazon and Japan Science and Technology Agency (JST) Adopting Sustainable Partnerships for Innovative Research Ecosystem (ASPIRE) JPMJAP2326.

REFERENCES

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surv. Tutor.*, vol. 17, no. 4, pp. 2347-2376, Fourth quarter 2015.

[2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22-32, Feb. 2014.

[3] L. D. Xu, W. He, and S. Li, "Internet of Things in industries: A survey," *IEEE Trans. Ind. Inform.*, vol. 10, no. 4, pp. 2233-2243, Nov. 2014.

[4] S. Tu, A. Badshah, H. Alasmary, and M. Waqas, "EAKE-WC: Efficient and anonymous authenticated key exchange scheme for wearable computing," early access, *IEEE Trans. Mob. Comput.*, Jul. 2023.

[5] Y. A. Qadri, A. Nauman, Y. B. Zikria, A. V. Vasilakos, and S. W. Kim, "The future of healthcare Internet of Things: A survey of emerging technologies," *IEEE Commun. Surv. Tutor.*, vol. 22, no. 2, pp. 1121-1167, Second quarter 2020.

[6] M. M. Salim, L. T. Yang, and J. H. Park, "Lightweight authentication scheme for IoT based e-healthcare service communication," *IEEE J. Biomed. Health Inform.*, early access, Dec. 2023.

[7] L. Zhang, S. Zhu, and S. Tang, "Privacy protection for telecare medicine information systems using a chaotic map-based three-factor authenticated key agreement scheme," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 2, pp. 465-475, Mar. 2017.

[8] M. Wazid, A. K. Das, V. Odelu, N. Kumar, M. Conti, and M. Jo, "Design of secure user authenticated key management protocol for generic IoT networks," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 269-282, Feb. 2018.

[9] J. Zheng, X. Huang, J. Odoom, and Y. Xiang, "A privacy-aware electronic medical record sharing scheme based on blockchain and identity-based cryptography," in *International Conference on Blockchain Technology and Information Security (ICBCTIS)*, Xi'an, China, Jul. 2023, pp. 94-100.

[10] H. Ma, R. Zhang, G. Yang, Z. Song, K. He, and Y. Xiao, "Efficient fine-grained data sharing mechanism for electronic medical record systems with mobile devices," *IEEE Trans. Depend. Secure Comput.*, vol. 17, no. 5, pp. 1026-1038, 1 Sep.-Oct. 2020.

[11] I. Ullah, A. Alkhalifah, S. U. Rehman, N. Kumar, and M. A. Khan, "An anonymous certificateless signcryption scheme for Internet of Health Things," *IEEE Access*, vol. 9, pp. 101207-101216, Jul. 2021.

[12] T.-F. Lee and C.-M. Liu, "A secure smart-card based authentication and key agreement scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 37, Art. no. 9933, pp. 1-11, Mar. 2013.

[13] T.-F. Lee, "Verifier-based three-party authentication schemes using extended chaotic maps for data exchange in telecare medicine information systems," *Comput. Methods Programs Biomed.*, vol. 117, no. 3, pp. 464-472, Oct. 2014.

[14] X. Xu, P. Zhu, Q. Wen, Z. Jin, H. Zhang, and L. He, "A secure and efficient authentication and key agreement scheme based on ECC for telecare medicine information systems," *J. Med. Syst.*, vol. 38, Art. no. 9994, pp. 1-7, Nov. 2014.

[15] F. Wen and D. Guo, "An improved anonymous authentication scheme for telecare medical information systems," *J. Med. Syst.*, vol. 38, Art. no. 26, pp. 1-8, Apr. 2014.

[16] M. Farash and M. Attari, "An efficient and provably secure three-party password-based authenticated key exchange protocol based on Chebyshev chaotic maps," *Nonlinear Dyn.*, vol. 77, pp. 399-411, Feb. 2014.

[17] D. Mishra, "Understanding security failures of two authentication and key agreement schemes for telecare medicine information systems," *J. Med. Syst.*, vol. 39, Art no. 19, pp. 1-8, Feb. 2015.

[18] K. Nandakumar and A. K. Jain, "Biometric template protection: Bridging the performance gap between theory and practice," *IEEE Signal Process. Mag.*, vol. 32, no. 5, pp. 88-100, Sep. 2015.

[19] A. Awasthi and K. Srivastava, "A biometric authentication scheme for telecare medicine information systems with nonce," *J. Med. Syst.*, vol. 37, Art. no. 9964, pp. 1-7, Aug. 2013.

[20] D. Mishra, S. Mukhopadhyay, S. Kumari, M. Khan, and A. Chaturvedi, "Security enhancement of a biometrics based authentication scheme for telecare medicine information systems with nonce," *J. Med. Syst.*, vol. 38, Art. no. 41, pp. 1-11, Apr. 2014.

[21] Z. Tan, "A user anonymity preserving three-factor authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 38, no. 3, pp. 1-9, Mar. 2014.

[22] H. Arshad and M. Nikooghadam, "Three-factor anonymous authentication and key agreement scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 38, Art. no. 136 pp. 1-9, Oct. 2014.

[23] X. Yan, W. Li, P. Li, J. Wang, X. Hao, and P. Gong, "A secure biometrics-based authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 37, no. 5, pp. 1-6, Oct. 2013.

[24] D. Mishra, S. Mukhopadhyay, A. Chaturvedi, S. Kumari, and M. Khan, "Cryptanalysis and improvement of Yan et al.'s biometric-based authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 38, no. 6, pp. 1-12, Jun. 2014.

[25] M. Sarvabhatla, M. Giri, and C. S. Vorugunti, "Cryptanalysis of cryptanalysis and improvement of Yan et al. biometric-based authentication scheme for TMIS," *arXiv preprint arXiv:1406.3943*, pp. 42-45, Jun. 2014.

[26] R. Amin, G. P. Biswas, "A secure three-factor user authentication and key agreement protocol for TMIS with user anonymity," *J. Med. Syst.*, vol. 39, no. 8, pp. 1-19, Aug. 2015.

[27] N. R. Potlapally, S. Ravi, A. Raghunathan, and N. K. Jha, "A study of the energy consumption characteristics of cryptographic algorithms and security protocols," *IEEE Trans. Mob. Comput.*, vol. 5, no. 2, pp. 128-143, Feb. 2006.

[28] A. Badshah, M. Waqas, F. Muhammad, G. Abbas, Z. H. Abbas. S. A. Chaudhry, and S. Chen, "AAKE-BIVT: Anonymous authenticated key exchange scheme for blockchain-enabled Internet of Vehicles in smart transportation," *IEEE Trans. Intelligent Transportation Systems*, vol. 24, no. 2, pp. 1739-1755, Feb. 2023.

[29] A. Badshah, M. Waqas, G. Abbas, F. Muhammad, Z. H. Abbas, S. Vimal, and M. Bilal, "LAKE-BSG: Lightweight authenticated key exchange scheme for blockchain-enabled smart grids," *Sustain. Energy Technol. Assess.*, vol. 52, Art. no. 102248, pp. 1-13, 2022.

[30] M. Tanveer, A. U. Khan, A. Alkhayyat, S. A. Chaudhry, Y. B. Zikria, and S. W. Kim, "REAS-TMIS: Resource-efficient authentication scheme for telecare medical information system," *IEEE Access*, vol. 10, pp. 23008-23021, Feb. 2022.

[31] A. Kumari, S. Jangirala, M. Y. Abbasi, V. Kumar, and M. Alam, "ESEAP: ECC based secure and efficient mutual authentication protocol using smart card", *J. Inf. Secur. Appl.*, vol. 51, Apr. 2020.

[32] S. Khatoon, S. M. M. Rahman, M. Alrubaian and A. Alamri, "Privacy-preserved provable secure mutually authenticated key agreement protocol for healthcare in a smart city environment", *IEEE Access*, vol. 7, pp. 47962-47971, Apr. 2019.

[33] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans. Information Theory*, vol. 29, no. 2, pp. 198-208, Mar. 1983.

[34] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Trans. Computers*, vol. 51, no. 5, pp. 541-552, May 2002.

[35] R. Canetti and H. Krawczyk, "Universally composable notions of key exchange and secure channels," in *Proc. EUROCRYPT*, Amsterdam, The Netherlands, Apr.-May 2002, pp. 337-351.

[36] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Adv. Cryptol.-EUROCRYPT*, Springer Berlin Heidelberg, Interlaken, Switzerland, May 2-6, vol. 23, pp. 523-540, 2004.

[37] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schälffer, "ASCON v1.2: Lightweight authenticated encryption and hashing," *J. Cryptol.*, vol. 34, pp. 1-42, Jul. 2021.

[38] A. Ostad-Sharif, D. Abbasinezhad-Mood, and M. Nikooghadam, "A robust and efficient ECC-based mutual authentication and session key generation scheme for healthcare applications," *J. Med. Syst.*, vol. 43, no. 1, pp. 1-22, Jan. 2019.

[39] C. H. Tseng, S.-H. Wang, and W.-J. Tsaur, "Hierarchical and dynamic elliptic curve cryptosystem based self-certified public key scheme for medical data protection," *IEEE Trans. Reliability*, vol. 64, no. 3, pp. 1078-1085, Sep. 2015.

[40] S. Qiu, G. Xu, H. Ahmad, and L. Wang, "A robust mutual authentication scheme based on elliptic curve cryptography for telecare medical information systems," *IEEE Access*, vol. 6, pp. 7452-7463, Dec. 2018.

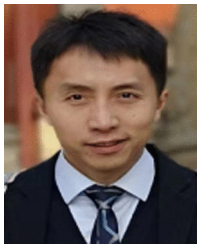
[41] B. D. Deebak and S. O. Hwang, "Privacy preserving based on seamless authentication with provable key verification using mIoMT for B5G-enabled healthcare systems," *IEEE Trans. Services Comput.*, vol. 17, no. 3, pp. 1097–1113, May-Jun. 2024.

[42] A. Attir, F. Naït-Abdesselam, and K. M. Faraoun, "Lightweight anonymous and mutual authentication scheme for wireless body area networks,""*Comput. Netw.*, vol. 224, pp. 109625, Apr. 2023.

[43] V. Sumithra, R. Shashidhara, and D. Mukhopadhyay, "Design of a secure and privacy preserving authentication protocol for telecare medical information systems," *Secur. Privacy*, vol. 5, no. 4, pp. e228, Jul. 2022.

DAKE ZENG received the B.Sc. degree from the Department of Computer Science, Hangzhou Dianzi University, in 2022. Currently, he is pursuing the M.Sc. degree in Computer Science and Technology at the Department of Computer Science, Beijing University of Technology. His research interests include edge computing, IoT security, and information security techniques.

AKHTAR BADSHAH received the B.Sc. degree in computer software engineering from the University of Engineering and Technology, Peshawar, Pakistan, in 2011, the M.Sc. degree in software engineering from the University of Engineering and Technology, Taxila, Pakistan, in 2017, and the Ph.D. degree in computer engineering from the Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Pakistan, in 2023. Currently, he is a Lecturer in the Department of Software Engineering at the University of Malakand, Dir Lower, Pakistan. His research interests include information security and privacy, applied cryptography, Internet of Things (IoT) security, blockchain applications and technologies, and digital twins security. He has made notable contributions to his field, with multiple research findings published in renowned journals.



SHANSHAN TU (Senior Member, IEEE) received the PhD degree from Computer Science Department, Beijing University of Posts and Telecommunications, in 2014. From 2013 to 2014, he visited University of Essex for National Joint Doctoral Training. He worked with the Department of Electronic Engineering, Tsinghua University as a postdoctoral researcher from 2014 to 2016. He is currently an associate professor and deputy dean with the Faculty of Information Technology, Beijing University of Technology, China. His research interests are in the areas of cloud computing, MEC, and information security techniques.

MUHAMMAD WAQAS (Senior Member, IEEE) received the Ph.D. degree from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2019. From October 2019 to March 2022, he was a Research Associate with the Faculty of Information Technology, Beijing University of Technology, Beijing. Since April 2022, he has been an Assistant Professor with the Computer Engineering Department, College of Information Technology, University of Bahrain, Zallaq, Bahrain. He is currently a Senior Lecturer with the School of Computing and Mathematical Sciences, Faculty of Engineering and Science, University of Greenwich, London, U.K. He has also been an Adjunct Senior Lecturer with the School of Engineering, Edith Cowan University, Perth, WA, Australia, since November 2021. His current research interests are in the areas of wireless communications, vehicular networks, cybersecurity, and machine learning. He is recognized as a Global Talent in the area of Wireless Communications by U.K. Research and Innovation.

ZHU HAN (Sâ€™01â€“Mâ€™04-SMâ€™09-Fâ€™14) received the B.S. degree in electronic engineering from Tsinghua University, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, in 1999 and 2003, respectively. From 2000 to 2002, he was an R&D Engineer of JDSU, Germantown, Maryland. From 2003 to 2006, he was a Research Associate at the University of Maryland. From 2006 to 2008, he was an assistant professor at Boise State University, Idaho. Currently, he is a John and Rebecca Moores Professor in the Electrical and Computer Engineering Department as well as in the Computer Science Department at the University of Houston, Texas. Dr. Hanâ€™s main research targets on the novel game-theory related concepts critical to enabling efficient and distributive use of wireless networks with limited resources. His other research interests include wireless resource allocation and management, wireless communications and networking, quantum computing, data science, smart grid, carbon neutralization, security and privacy. Dr. Han received an NSF Career Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011,Â the EURASIP Best Paper Award for the Journal on Advances in Signal Processing in 2015, IEEE Leonard G. Abraham Prize in the field of Communications Systems (best paper award in IEEE JSAC) in 2016, IEEE Vehicular Technology Society 2022 Best Land Transportation Paper Award, and several best paper awards in IEEE conferences. Dr. Han was an IEEE Communications Society Distinguished Lecturer from 2015 to 2018 and ACM Distinguished Speaker from 2022 to 2025, AAAS fellow since 2019, and ACM Fellow since 2024. Dr. Han is a 1% highly cited researcher since 2017 according to Web of Science. Dr. Han is also the winner of the 2021 IEEE Kiyo Tomiyasu Award (an IEEE Field Award), for outstanding early to mid-career contributions to technologies holding the promise of innovative applications, with the following citation: for contributions to game theory and distributed management of autonomous communication networks.