EAKE-WC: Efficient and Anonymous Authenticated Key Exchange Scheme for Wearable Computing

Shanshan Tu, *Member, IEEE*, Akhtar Badshah, Hisham Alasmary, Muhammad Waqas, *Senior Member, IEEE*

***Abstract*—Wearable computing has shown tremendous potential to revolutionize and uplift the standard of our lives. However, researchers and field experts have often noted several privacy and security vulnerabilities in the field of wearable computing. In order to tackle these problems, various schemes have been proposed in the literature to improve the efficiency of authentication and key establishment procedure. However, the existing schemes have relatively high computation and communication overheads and are not resilient to various potential security attacks, which reduces their significance for applicability in constrained wearable devices. In this work, we propose an efficient and anonymous authenticated key exchange scheme for wearable computing (EAKE-WC), which performs mutual authentication between the user and the wearable device, and between the cloud server and the user. It also establishes secret session keys for each session to secure communication among the communicating entities. Additionally, the proposed EAKE-WC scheme is designed using authenticated encryption with associated data (AEAD) primitives like ASCON, bitwise XOR, and hash functions. Our results from the security analysis depict compliance of the proposed EAKE-WC with wearable computing's security criteria. In addition, we also demonstrate through a comprehensive comparative analysis that the proposed scheme, EAKE-WC, outperforms the existing benchmark schemes in various key performance areas, including lower communication and computational overheads, enhanced security, and added functionality.**

***Index Terms*—Key exchange, authentication, authenticated encryption with associated data, wearable computing, security.**

I. Introduction

Wearable devices are specific communication devices that are typically integrated into clothing items or any other wearable accessory or worn on the body directly. By connecting it to the Internet and integrating it with various software applications, users can perceive and monitor their physiological and surrounding environmental conditions. They may immediately view, respond to, and transmit information without a manual procedure. Its features cover various industries, including mobile payment, social networking, casual games, audio-visual entertainment, positioning and navigation, and health management [1]. Presently, wearable devices mainly include smart watches and bracelets, smart shoes, smart clothing, smart glasses (mostly VR/AR headsets), smart accessories, and ear-mounted devices. In healthcare, wearable devices are employed to track various physiological data about patients, including several blood-related readings (glucose level, blood pressure, and oxygen level), heart rate, and body temperature [2]. Generally, the acquired data is first transmitted from a wearable device to the mobile terminal of the user through a wireless protocol for additional processing and then sent by the mobile to the cloud for further analysis. Further, the doctor monitors the patient's health condition remotely while connected to the cloud server and makes treatment recommendations.

With the advancement of hardware and software technology, smart wearable devices will be seen everywhere in our daily life, bringing great convenience. However, the installed applications and wearable devices in the wearable computing environment face security and privacy issues. Further, the utilization of wireless transmission technology in the wearable computing environment, where data collecting, storage, and transmission are more susceptible to numerous assaults, constitutes a more serious security concern [3], [4].

An authenticated key exchange (AKE) scheme is a widely-used security solution that offers protection against various potential security attacks, including "de-synchronization", "ephemeral secret leakage (ESL)", "replay", "impersonation", and others. AKE enables two legitimate parties to mutually authenticate each other, establishing a secure communication channel. Once successfully authenticated, the two entities derive a shared session key that can be used to encrypt and decrypt data transmitted over the open channel for future secure communication. AKE schemes are an effective way to ensure secure and private communication between two entities [5]. Numerous cryptographic primitives have been employed in the various authentication schemes used in wireless communication environments [6], [7]. These schemes utilize various cryptographic primitives, such as public key cryptography schemes [8], [9], hybrid schemes based on the cryptographic hash function and symmetric technology [10], [11], and schemes that only use the cryptographic hash function [12], [13]. As wearable devices are resource-limited, most wearable computing environments utilize authentication schemes that employ lightweight cryptographic primitives, like symmetric cryptography and cryptographic hash functions [14]. Although many potential security attacks can be thwarted to some extent by the AKE schemes currently in use in wireless environments, several of them have been proven to fall short of their stated security objectives. In order to facilitate secure communication in wearable computing environments, we devise a more reliable, robust, and ultralightweight AKE scheme.

Following are some of the major contributions of this work:

S. Tu is with the Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China (e-mail: sstu@bjut.edu.cn).

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

H. Alasmary is with the Department of Computer Science, College of Computer Science, King Khalid University, Abha 61421, Kingdom of Saudi Arabia, (e-mail: alasmary@kku.edu.sa).

M. Waqas is with the Computer Engineering Department, College of Information Technology, University of Bahrain, 32038, Bahrain and School of Engineering, Edith Cowan University, Perth WA 6027, Australia. (e-mail: engr.waqas2079@gmail.com).

- Firstly, we proposed an efficient and anonymous authenticated key exchange scheme for wearable computing, called EAKE-WC. In EAKE-WC, after mutual authentication between the user and the cloud server as well as between the user and the wearable device, session keys are created with the aid of a cloud server.
- Secondly, for a resource-limited wearable computing environment, the devised EAKE-WC scheme utilizes lightweight cryptographic primitives composed of authenticated encryption with associated data (AEAD) primitive, known as ASCON [15], bitwise XOR operations, and hash functions.
- Thirdly, the devised EAKE-WC scheme preserves the session key security according to a rigorous security analysis using the widely used "Real-Or-Random (ROR) model". Furthermore, the informal security analyses prove that our devised EAKE-WC meets the security objectives of wearable computing.
- Finally, a rigorous comparative analysis is presented to reveal that the performance of the EAKE-WC is better than the benchmark schemes by analyzing computation and communication overheads along with security and functionality traits.

The rest of this article is divided into the following sections. We first briefly present the relevant prior work about AKE schemes in wearable computing in Section II. The relevant background, including assumptions, basics of networks, design goals, threat models, and preliminary knowledge, is discussed in Section III. Then, the proposed scheme is presented in Section IV followed by security analysis in Section V and by comparative analyses among the proposed and other competing schemes in Section VI. Finally, in Section VII, we draw some concluding remarks.

II. Related Work

Researchers have recently proposed numerous AKE schemes for resource-limited devices in wearable computing environments. The ultimate aim is to maximize performance while reducing overheads under the notion that mutual authentication and key agreement are accomplished as securely as possible. In [16], Wang *et al.* briefly pointed out the evaluation metric for anonymous two-factor authentication schemes considering the impressive list of security requirements (for instance, resistance to impersonation attacks) and desirable characteristics (for instance, user anonymity). Then, they offer a counterargument to the unresolved problem left by Huang *et al.* [17].

A lightweight anonymous user AKE scheme, proposed by Gupta *et al.* [18], was based on bitwise XOR operations as well as on hash functions, and the mutual authentication between the device and the mobile terminal of the user is accomplished via the assistance of the cloud server. However, Hajian *et al.* [19] investigated Gupta *et al.*'s scheme and illustrated how it is vulnerable to the privileged insider, de-synchronization, and offline password guessing attacks.

Alternatively, Liu *et al.* [20] proposed an asymmetric three-party-based AKE scheme making use of the visual out-of-band channel and quick response code to provide mutual authentication between mobile terminals and wearable devices. Similarly, for enhanced security in healthcare systems, Wu *et al.* [21] developed an anonymous two-factor AKE scheme. However, their proposed scheme is vulnerable to privileged insider and de-synchronization attacks.

Another novel bio-hashing-based AKE scheme was proposed by Li *et al.* [22]. Their proposed scheme, however, cannot render user anonymity and is vulnerable to privileged insider and sensor node capture attacks. Ali *et al.* [24] investigated the security flaws present in the work of Amin *et al.* [23] and suggested an improved three-factor-based secure remote user authentication scheme.

Das *et al.* [25] proposed a mutual authentication scheme for wearable devices. However, Das *et al.*'s scheme is not resilient against de-synchronization attacks [26]. In 2020, Srinivas *et al.* [27] presented a cloud-based authentication scheme employing hash functions and the Chinese Remainder Theorem, which accomplish mutual authentication between remote users and wearable sensor nodes and can resist potential security attacks.

In [28], Guo *et al.* designed an anonymous and efficient AKE scheme for wearable computing. The proposed scheme reduced the computation overhead by employing bitwise XOR operations and a large number of hash functions. Fotouhi *et al.* [29] designed a lightweight two-factor-based AKE scheme for the Internet of medical things. However, Li *et al.* [30] pointed out that Fotouhi *et al.* [29] cannot resist sensor physical capture and stolen verifier attacks.

Wazid *et al.* [31] designed and performed testbed experiments of the user AKE scheme for smart healthcare. However, the proposed scheme cannot resist de-synchronization attacks.

In [32], Wang *et al.* proposed a physical unclonable function (PUF) and blockchain-based AKE scheme for wireless medical sensor networks. However, Yu and Park [33] indicated that Wang *et al.*'s scheme [32] does not provide mutual authentication and is not secured against session key disclosure and MitM attacks. To address the limitations of Wang *et al.*'s scheme [32], Yu and Park [33] devised an improved version of their scheme.

Some of the relevant aforementioned AKE schemes are summarized in Table I, where the cryptographic primitives used in the study, as well as their limitations and/or shortcomings, are also listed.

III. Background

This section briefly introduces the network model and assumptions, threat models, design goals, and preliminary knowledge. In Table II, we listed various notations utilized in this paper.

A. Network Model and Assumptions

As depicted in Fig. 1, our proposed network model consists of four entities:

1) A trusted registration authority (RA) which is responsible for securely registering wearable devices and users offline.

TABLE I: Summary of the Existing Recent User AKE Schemes in the Wearable Computing Environment

Scheme	Cryptographic Primitives	Shortcomings/ Limitations
Wu *et al.* 2017 [21]	Symmetric encryption/ decryption and hash functions	• Cannot resist de-synchronization and privileged insider attacks • Unable to provide anonymity and untraceability features
Amin *et al.* 2018 [23]	Hash functions	• Vulnerable to de-synchronization and stolen mobile device attacks • Unable to provide anonymity and untraceability features
Das *et al.* 2018 [25]	Biometric fuzzy extractor and hash functions	• Cannot resist de-synchronization attack
Gupta *et al.* 2019 [18]	Hash functions	• Vulnerable to de-synchronization, offline password-guessing, and privileged insider attacks • Unable to provide anonymity and untraceability features
Fotouhi *et al.* 2020 [29]	Hash functions	• Cannot resist sensor physical capture and stolen verifier attacks
Wang *et al.* 2022 [32]	Biometric fuzzy extractor, PUF, and hash functions	• Cannot resist session key disclosure and MitM attacks • Unable to provide mutual authentication
Guo *et al.* 2022 [28]	Hash functions	• Requires high computational and communication overheads
Wazid *et al.* 2022 [31]	Biometric fuzzy extractor and hash functions	• Cannot resist de-synchronization attack • Requires high computational and communication overheads

Note: man-in-the-middle (MitM), physical unclonable function (PUF).

TABLE II: List of Notations

Notation	Description
U_j, WD_i	jth user and ith wearable device of U_j
MT_j	jth mobile terminal
rn_j	random secret used in the registration of MT_j
ID_j, IDS_j	jth user unique identity and pseudo-identity
PW_j	Password of U_j
ID_i, IDS_i	ith wearable device unique identity and pseudo-identity
$E(\cdot)/D(\cdot)$	ASCON encryption/decryption function
k	ASCON encryption/decryption key
MAC_i	Message authentication code generated by ASCON
C_i	ith cipher text generated by ASCON encryption function
TS_i	Timestamp
ΔT	Message time delay limit
rn_n	nth random number
$\Vert$, $\oplus$, $h(\cdot)$	concatenation, XOR, and hash-function, respectively
$\mathcal{A}$	Adversary
SK	Session key
RA	Trusted registration authority
CS, K	Cloud server and master secret key of CS.

2) Wearable devices which gather various physiological data from the patient, such as glucose level, blood pressure, oxygen level, heart rate, and body temperature.
3) Mobile/user terminals that are connected to wearable devices and receive the collected data wirelessly. The mobile terminal then establishes an Internet connection and sends the data to the cloud server for further analysis.
4) Cloud server which stores the physiological data and allows authorized remote users, such as physicians, to access and analyze the data.

The mobile terminal plays a critical role in our proposed network model. It acts as an intermediary between the wearable devices and the cloud server, receiving the collected physiological data from the wearable devices and forwarding it to the cloud server. We primarily concentrate on the AKE schemes in the given network model. Session keys are formed with the assistance of the cloud server after mutual authentication (made possible by the AKE scheme) between both groups (the user and the wearable device, and the user and the cloud server).

B. Threat Model

The current study establishes the widely-used "Dolev-Yao threat model" [34] as well as the "Canetti and Krawczyk (CK) adversary model" [35], and makes the following presumptions about the capabilities of the adversary.

- Communication between any two participants occurs across an insecure open channel, where an adversary $\mathcal{A}$ can eavesdrop, modify, compromise, delay, replay, and delete part of the message or the entire message.
- The cloud server and the RA are considered trusted entities and cannot be compromised. However, mobile/user terminals and wearable devices could be stolen/lost, so both are not considered trustworthy entities. Furthermore, $\mathcal{A}$ can access the sensitive credentials stored in their memory by employing power analysis attacks.
- The adversary $\mathcal{A}$ can compromise secret keys, session keys, other session states, and other ephemeral information under the "CK-adversary model".
- Similarly, the adversary $\mathcal{A}$ might guess the user's password or look over their shoulder. However, $\mathcal{A}$ cannot obtain the mobile terminal as well as access the user's password simultaneously. Furthermore, the user's mobile terminal and the wearable devices are both prone to being stolen by adversary $\mathcal{A}$. However, it is assumed that the adversary cannot capture both simultaneously.

C. Design Goals

The proposed EAKE-WC seeks to achieve the following design goals:

1) **Mutual authentication:** The user and the wearable device, likewise, the user and the cloud server, should be able to independently check each other's authenticity to guarantee the participants' reliability.
2) **Session key agreement:** The user and the wearable device, likewise, the user and the cloud server, can negotiate secret session keys to encrypt and decrypt the subsequent communications.
3) **Forward security:** Ensures that the previous secret session key is not affected if the current secret session key is compromised.
4) **Anonymity:** Protection and anonymity for the real identities of all the communicating entities, i.e., wearable devices, mobile/user terminals, and the cloud server.
5) **Untraceability:** The AKE message exchanged by the various parties, i.e., wearables, mobile/user terminals, and the cloud server, should not be traceable from the adversary's perspective.
6) **Un-linkability:** The scheme should ensure that several communications from the same source cannot be coupled or that there is no connection between various contacts of the same entity, preventing the adversary from acquiring important parameters from many interactions of the same entity.
7) **Resilience against potential attacks:** The scheme should withstand various attacks, such as replay, impersonation, modification, and MitM attacks, to ensure the security of wearable computing.

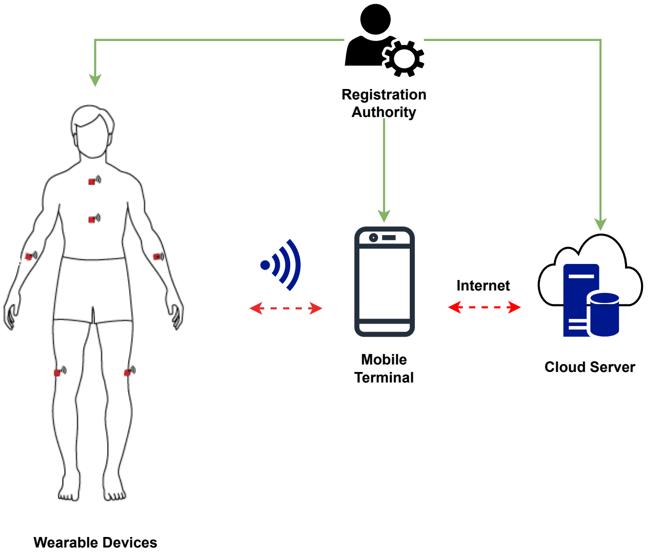

Fig. 1: Structure of cloud-assisted wearable computing environment.

D. Preliminary Knowledge

1) ASCON

A well-known AEAD algorithm, ASCON, provides integrity, authenticity, and confidentiality of the data at the same time without utilizing a message authentication code, and is a nonce-based, inverse-free, online (encryption and decryption), and single-pass block cipher devised for resource-constrained devices. The ASCON encryption procedure can be described as follows (Equation 1):

$$(CTxt, MAC) = E_K(N,\ AD,\ PTxt) \tag{1}$$

where E signifies the encryption procedure, which takes shared secret key K, nonce N, associated data AD, and plaintext $PTxt$ of arbitrary length as input. It provides ciphertext $CTxt$ of the same length as $PTxt$ and a message authentication code MAC as output, simultaneously authenticating AD and $PTxt$. Whereas ASCON decryption procedure can be described as follows (Equation 2):

$$(PTxt,\ \perp) = D_K(N,\ AD,\ CTxt,\ MAC) \tag{2}$$

where D signifies the decryption procedure, which takes the shared secret key K, nonce N, associated data AD, ciphertext $CTxt$, and message authentication code MAC produced by the ASCON encryption procedure E as input and outputs either the $PTxt$ if message authentication code MAC verified correctly or trigger an error $\perp$ if the message authentication code verification fails.

IV. The Proposed Scheme

This section presents the proposed EAKE-WC scheme for wearable computing environments to ensure that only authorized users can access the deployed wearable devices. The proposed EAKE-WC is based on the preloaded key method and utilizes the secure hash algorithm (SHA-256) with ASCON AEAD primitive. Furthermore, it is assumed that the time of every entity in the wearable computing environment is synchronized. The EAKE-WC is divided into five phases, each of which is described in further detail below:

A. Setup Phase

The RA establishes secret credentials during this phase using the steps enlisted below for individual wearable devices.

Step-1: A master secret key K is selected by the RA for the cloud server.

Step-2: For each wearable device WD_i, the RA picks a unique identity ID_i, a pseudo-random identity IDS_i, and an encryption key k_W.

Step-3: The RA stores the credentials $\{ID_i,\ IDS_i,\ k_W\}$ in the memory of wearable device WD_i and also keeps the credentials $\{(ID_i,\ IDS_i^{old} = null,\ IDS_i^{new} = IDS_i,\ k_W^{old} = null,\ k_W^{new} = k_W)\}$ in the database of the cloud server.

B. Registration Phase

The user U_j registers with the RA securely via a private channel, as described in the User Registration Procedure (URP) below:

URP1: U_j picks an identity ID_j and a random number rn_j and forwards both parameters to the RA through a secure private channel.

URP2: After obtaining the parameters $\{ID_j, rn_j\}$, the RA selects a pseudo-random identity IDS_j and and an encryption key k_U and computes the parameter $X_1 = h(ID_j \parallel K)$ and the parameter P_j as $P_j = X_1^1 \oplus X_1^2$.

URP3: The RA stores the parameters $\{(ID_j,\ IDS_j^{old} = null,\ IDS_j^{new} = IDS_j,\ k_U^{old} = null,\ k_U^{new} = k_U)\}$ in the database of the cloud server, and transmits $\{P_j,\ IDS_j,\ k_U\}$ to U_j via a secure channel.

URP4: After receiving $\{P_j,\ IDS_j,\ k_U\}$, U_j selects a password PW_j and then computes $X_2 = h(ID_i \parallel PW_j)$, $k = X_2^1 \oplus X_2^2$, $\{CT, MAC\} = E_k(rn_j,\ X_2^2, (P_j))$. Next, store the credentials $\{(rn_j,\ CT,\ MAC,\ IDS_j^{old} = null,\ IDS_j^{new} = IDS_j,\ k_U^{old} = null,\ k_U^{new} = k_U)\}$ in the memory of MT_j.

C. User login Phase

LG-1: User U_j insert his/her identity ID_j and password PW_j^l into MT_j.

LG-2: MT_j retrieves the parameters rn_j, CT, and MAC and then computes $X_3 = h(ID_j \parallel PW_j^l)$, $k^l = X_3^1 \oplus X_3^2$, and $\{PT_j,\ \perp\} = D_{k^l}(rn_j,\ X_3^2,\ CT,\ MAC)$, if verification of MAC fails, abort the login process. Else, U_j successfully login into MT_j and retrieves the secret parameter P_j from the plaintext as $PT_j = P_j$.

D. Authenticated Key Exchange Phase

AKE-1: After U_j successfully login procedure. MT_j then generates a random number rn_1 and current timestamp TS_1. Next, it constructs a message M_1 with parameters $M_1 =< rn_1,\ IDS_j,\ TS_1 >$ and dispatches it to WD_i using the available public channel.

AKE-2: Upon receiving M_1 from U_j, WD_i verifies the validity of TS_1 by evaluating the condition $|TS_2 - TS_1| \overset{?}{\leq} \Delta T$, where TS_2 refers to the current timestamp and ΔT is the maximum permitted time interval (notice that the subsequent steps all utilize the timestamp checking method). The message is rejected if the condition does not hold, and WD_i cancels the request. Else the WD_i keeps M_1, generates a random number rn_2, and picks the current timestamp TS_3, and retrieve $ID_i,\ IDS_i$, and k_W from its own memory. Next, WD_i computes $\{(C_1 \parallel C_2 \parallel C_3 \parallel C_4), MAC_1\} = E_{k_W}(rn_2 \oplus TS_3,\ IDS_i,\ (rn_1 \parallel IDS_j \parallel ID_i \parallel rn_2))$. *In the proposed scheme, we are using ASCON as the encryption algorithm, which takes the key, nonce, and associative data to generate the ciphertext and message authentication code. Here, associative data is IDS_i, nonce is $rn_2 \oplus TS_3$ and plaintext is $(rn_1 \parallel IDS_j \parallel ID_i \parallel rn_2)$*. Finally, WD_i fabricates the message M_2 with credentials $M_2 =< rn_2,\ IDS_i,\ MAC_1,\ TS_3 >$ and transmits it to U_j via the public communication channel.

AKE-3: U_j checks $|TS_4 - TS_3| \overset{?}{\leq} \Delta T$ to ensure the timeliness of the received message. If the message is not replayed, U_j picks rn_3 and TS_5 and derives $\{(C_5 \parallel C_6 \parallel C_7 \parallel C_8), MAC_2\} = E_{k_U}(rn_1 \oplus rn_3 \oplus TS_5,\ IDS_j, (rn_3 \parallel IDS_i \parallel P_j \parallel IDS_j))$. Finally, U_j sends the message $M_3 =< M_2,\ rn_1,\ rn_3,\ IDS_j,\ MAC_2,\ TS_5 >$ to CS via open communication channel.

AKE-4: CS checks $|TS_6 - TS_5| \overset{?}{\leq} \Delta T$ to validate whether the received message is replayed. If the message is not replayed, CS searches IDS_j in its database and retrieves the corresponding ID_j and k_U. In addition, CS computes $X_4 = h(ID_j \parallel K)$, $P_j = X_4^1 \oplus X_4^2$, $\{(C_9 \parallel C_{10} \parallel C_{11} \parallel C_{12}), MAC_3\} = E_{k_U}(rn_1 \oplus rn_3 \oplus TS_5,\ IDS_j,\ (rn_3 \parallel IDS_i \parallel P_j \parallel IDS_j))$. Finally, to check the received message's authenticity and integrity, CS validates $MAC_2 \overset{?}{=} MAC_3$. Abort the AKE process if the condition does not hold. Otherwise, CS continues the AKE procedure.

AKE-5: CS Searches IDS_i and retrieves ID_i and k_W and computes $\{(C_{13} \parallel C_{14} \parallel C_{15} \parallel C_{16}), MAC_4\} = E_{k_W}(rn_2 \oplus TS_3,\ IDS_i,\ (rn_1 \parallel IDS_j \parallel ID_i \parallel rn_2))$. In addition, to validate the integrity and authenticity, CS checks $MAC_1 \overset{?}{=} MAC_4$. If it holds, CS continues the AKE procedure.

AKE-6: Finally, CS stores $(IDS_j^{old} = IDS_j, IDS_j^{new} = C_{10},\ k_U^{old} = k_U,\ k_U^{new} = C_{11})$ and $(IDS_i^{old} = IDS_i,\ IDS_i^{new} = C_{14},\ k_W^{old} = k_W,\ k_W^{new} = C_{15})$ in its own database. In addition, CS computes the session key between U_j and WD_i as $SK_{UD} = C_{13}$ and U_j and CS as $SK_{CU} = C_9$. Furthermore, CS generates TS_7 and computes $X_5 = SK_{CU} \oplus SK_{UD}$, $MAC_5 = h(X_5 \parallel P_j \parallel TS_7)$ and sends the message $M_4 =< C_{12},\ C_{16},\ X_5,\ MAC_5,\ TS_7 >$ to U_j via open communication channel.

AKE-7: U_j checks the timeliness of the received message as $|TS_8 - TS_7| \overset{?}{\leq} \Delta T$. In addition, U_j determines $C_8 \overset{?}{=} C_{12}$ and validates $MAC_5 \overset{?}{=} h(X_5 \parallel P_j \parallel TS_7)$. If it holds, U_j continues with the AKE process. Further, U_j stores $(SK_{UC} = C_5, IDS_j^{old} = IDS_j,\ IDS_j^{new} = C_6,\ k_U^{old} = k_U,\ k_U^{new} = C_7$ and determines the session key between U_j and CS as $SK_{UD} = X_5 \oplus SK_{UC})$. Finally, U_j picks TS_9 and sends $M_5 =< C_{16},\ TS_9 >$ to WD_i using the open channel.

AKE-8: WD_i checks, if the received message is fresh through $|TS_{10} - TS_9| \overset{?}{\leq} \Delta T$. Moreover, WD_i checks $C_4 \overset{?}{=} C_{16}$. If it holds, WD_i derives the session key as $(SK_{DU} = C_1$ to achieve the indecipherable communication between WD_i and U_j. Furthermore, WD_i updates the parameters $IDS_i = C_2,\ k_W = C_3)$ in its memory.

Wearable device WD_i	**User** U_j **/ Mobile terminal** MT_j	**Cloud server** CS
$\{ID_i,\ IDS_i,\ k_W\}$	$\{rn_j,\ CT,\ MAC,\ IDS_j^{old},\ IDS_j^{new},\ k_U^{old},\ k_U^{new}\}$	$\{(ID_j,\ IDS_j^{old},\ IDS_j^{new},\ k_U^{old},\ k_U^{new}),$ $(ID_i,\ IDS_i^{old},\ IDS_i^{new},\ k_W^{old},\ k_W^{new})\}$
	LG-1: U_j inputs $ID_j,\ PW_j^l$, **LG-2:** Computes $X_3 = h(ID_j \parallel PW_j^l)$, $k_j^l = X_3^1 \oplus X_3^2$, $\{PT_j,\ \perp\} = D_{k_j^l}(rn_j,\ X_3^2,\ CT,\ MAC)$. Error, if verification of MAC fails, Else, retrieve $PT_j = \{P_j\}$. **AKE-1:** Pick rn_1, TS_1. $\xleftarrow[\text{(Via public channel)}]{M_1:\{rn_1,\ IDS_j,\ TS_1\}}$	
AKE-2: Check $\|TS_2 - TS_1\| \overset{?}{\le} \Delta T$ If not, abort. Pick rn_2, TS_3. Retrieve $ID_i,\ IDS_i,\ k_W$. Compute $\{(C_1 \parallel C_2 \parallel C_3 \parallel C_4), MAC_1\} = E_{k_W}(rn_2 \oplus TS_3,\ IDS_i,\ (rn_1 \parallel IDS_j \parallel ID_i \parallel rn_2))$. $\xrightarrow[\text{(Via public channel)}]{M_2:\{rn_2,\ IDS_i,\ MAC_1,\ TS_3\}}$		
	AKE-3: Check if $\|TS_4 - TS_3\| \overset{?}{\le} \Delta T$, If not, abort. Pick rn_3, TS_5. Computes $\{(C_5 \parallel C_6 \parallel C_7 \parallel C_8), MAC_2\} = E_{k_U}(rn_1 \oplus rn_3 \oplus TS_5,\ IDS_j,\ (rn_3 \parallel IDS_i \parallel P_j \parallel IDS_j))$. $\xrightarrow[\text{(Via public channel)}]{M_3:\{M_2,\ rn_1,\ rn_3,\ IDS_j,\ MAC_2,\ TS_5\}}$	
		AKE-4: Check if $\|TS_6 - TS_5\| \overset{?}{\le} \Delta T$, If not, abort. Search IDS_j and retrieve its corresponding ID_j and k_U. Computes $X_4 = h(ID_j \parallel K)$, $P_j = X_4^1 \oplus X_4^2$, $\{(C_9 \parallel C_{10} \parallel C_{11} \parallel C_{12}), MAC_3\} = E_{k_U}(rn_1 \oplus rn_3 \oplus TS_5,\ IDS_j,\ (rn_3 \parallel IDS_i \parallel P_j \parallel IDS_j))$. Check if $MAC_2 \overset{?}{=} MAC_3$ If not, abort. **AKE-5:** Search IDS_i and retrieve its corresponding ID_i and k_W. Compute $\{(C_{13} \parallel C_{14} \parallel C_{15} \parallel C_{16}), MAC_4\} = E_{k_W}(rn_2 \oplus TS_3,\ IDS_i, (rn_1 \parallel IDS_j \parallel ID_i \parallel rn_2))$. Check if $MAC_1 \overset{?}{=} MAC_4$ If not, abort. **AKE-6:** Store $\{(SK_{CU} = C_9,\ IDS_j^{old} = IDS_j,\ IDS_j^{new} = C_{10},\ k_U^{old} = k_U,\ k_U^{new} = C_{11}),$ $(SK_{UD} = C_{13}, IDS_i^{old} = IDS_i,\ IDS_i^{new} = C_{14},\ k_W^{old} = k_W,\ k_W^{new} = C_{15})\}$. Generates TS_7. Compute $X_5 = SK_{CU} \oplus SK_{UD}$, $MAC_5 = h(X_5 \parallel P_j \parallel TS_7)$. $\xleftarrow[\text{(Via public channel)}]{M_4:\{C_{12},\ C_{16},\ X_5,\ MAC_5,\ TS_7\}}$
	AKE-7: Check if $\|TS_8 - TS_7\| \overset{?}{\le} \Delta T$, $C_8 \overset{?}{=} C_{12}$, and $MAC_5 \overset{?}{=} h(X_5 \parallel P_j \parallel TS_7)$ If not, abort. Store $(SK_{UC} = C_5, IDS_j^{old} = IDS_j, IDS_j^{new} = C_6,\ k_U^{old} = k_U,\ k_U^{new} = C_7,\ SK_{UD} = X_5 \oplus SK_{UC})$. Pick TS_9. $\xleftarrow[\text{(Via public channel)}]{M_5:\{C_{16},\ TS_9\}}$	
AKE-8: Check if $\|TS_{10} - TS_9\| \overset{?}{\le} \Delta T$ and $C_4 \overset{?}{=} C_{16}$, If not, abort. Store $(SK_{DU} = C_1,\ IDS_i = C_2,\ k_W = C_3)$.		

Fig. 2: Login and AKE procedures of the proposed EAKE-WC scheme.

E. Password Reset Phase

If U_j needs to update the password, the following actions should be performed.

Step 1: First, the identity (ID_j) and the old password (PW_j^o) should be entered into MT_j.

Step 2: Secondly, MT_j retrieves the parameters rn_j, CT, MAC from its memory and computes $X_3 = h(ID_j \parallel PW_j^o)$, $k^l = X_3^1 \oplus X_3^2$, and $\{PT_j,\ \perp\} = D_{k^l}(rn_j,\ X_3^2,\ CT,\ MAC)$, if verification of MAC fails, abort the password reset process. Else, MT_j prompts for a new password PW_j^n and retrieves the secret parameter P_j from the plaintext as $PT_j = P_j$. Further, U_j enters a new password PW_j^n to reset the password.

Step 3: Finally, MT_j computes $X_2 = h(ID_i \parallel PW_j^n)$, $k^n = X_2^1 \oplus X_2^2$, $\{CT^n, MAC^n\} = E_{k^n}(rn_j,\ X_2^2, (P_j))$. Next, updates the credentials list $\{(rn_j,\ CT^n,\ MAC^n,\ IDS_j^{old} = null,\ IDS_j^{new} = IDS_j,\ k_U^{old} = null,\ k_U^{new} = k_U)\}$ in the memory of MT_j.

V. Security Analysis

The security aspect of the proposed EAKE-WC scheme is examined in this section utilizing informal and formal (mathematical) security analysis techniques.

A. Informal Security Analysis

The following security aspects of our EAKE-WC scheme protect it from the most critical vulnerabilities in wearable computing environments.

1) Replay Attack

Suppose $\mathscr{A}$ eavesdrops on all transmitted messages i.e., $M_1 : \{rn_1,\ IDS_j,\ TS_1\}$, $M_2 : \{rn_2,\ IDS_i,\ MAC_1,\ TS_3\}$, $M_3 : \{M_2,\ rn_1,\ rn_3,\ IDS_j,\ MAC_2,\ TS_5\}$, $M_4 :$

$\{C_{12},\ C_{16},\ X_5,\ MAC_5,\ TS_7\}$, and $M_5 : \{C_{16},\ TS_9\}$. These transmitted messages are formed through random numbers and current timestamps, indicating their validity for a single session only. Therefore, our proposed scheme resists replay attacks.

2) *Password Guessing Attack*

Consider the scenario that a stolen or lost mobile terminal is acquired by $\mathcal{A}$, then $\mathcal{A}$ can retrieve all the parameters stored in the mobile terminal, i.e., $\{rn_j,\ CT,\ MAC,\ IDS_j^{old},\ IDS_j^{new},\ k_U^{old},\ k_U^{new}\}$. To retrieve the password $\mathcal{A}$ needs to computes $X_3 = h(ID_j^l \parallel PW_j^l)$, $k_j^l = X_3^1 \oplus X_3^2$, $\{P_j, MAC^l\} = D_{k_j^l}(rn_j,\ X_3^2, (CT))$. It is infeasible for $\mathcal{A}$ to retrieve the password of the user without the knowledge of ID_j.

3) *Mutual Authentication*

In the proposed EAKE-WC scheme, the mutual authentication among the participating entities is accomplished as follows: 1) in M_3, CS verifies $MAC_2 \overset{?}{=} MAC_3$ to authenticate U_j ; 2) in M_3, CS then verifies $MAC_1 \overset{?}{=} MAC_4$ to authenticate WD_i ; 3) in M_4, U_j verifies $MAC_5 \overset{?}{=} h(X_5 \parallel P_j \parallel TS_7)$ to authenticate CS; 4) in M_5, WD_i verifies $C_4 \overset{?}{=} C_{16}$ to authenticate U_j.

4) *MitM Attack*

Consider a scenario in which $\mathcal{A}$ intercepts all the transmitted messages $M_1 \sim M_5$ during the login and AKE procedure and tries to alter them so that the recipient believes they are authentic messages. Assuming $\mathcal{A}$ alters the message M_1, it is clear from the analysis of the user impersonation attack that $\mathcal{A}$ is unable to send a legitimate message to CS. As a result, the authentication is ended. $\mathcal{A}$ must be familiar with ID_i, k_W, P_j, k_U in order to correctly modify M_2. Similar to this, $\mathcal{A}$ cannot produce legitimate random numbers and timestamps to edit M_3, M_4, and M_5 without knowing ID_i, k_W, P_j, k_U, and K. Thus, MitM attacks cannot succeed against our scheme.

5) *Anonymity and Untraceability*

During the login and AKE phases, assume $\mathcal{A}$ intercepts all messages $M_1 \sim M_5$. Though every message includes timestamps and random numbers, as well as parameter MAC, which is generated using the ASCON encryption function using secret keys, without knowing these secret parameters, $\mathcal{A}$ is unable to compute the user's identity ID_j from $M_1 \sim M_5$, nor to compute the identity ID_i of WD_i. Thus, our proposed EAKE-WC can provide anonymity for wearable devices and users. Additionally, every session has distinct random numbers, timestamps, and ASCON encryption keys, signifying that every message is dynamic and distinct. As a result, $\mathcal{A}$ is unable to follow a user across all the sessions. From the intercepted communications $M_1 \sim M_5$, $\mathcal{A}$ can learn the pseudo-identities of U_j and WD_i, but in every session, U_j's pseudo-identity IDS_j and k_U, and WD_i's pseudo-identity IDS_i and k_W are changed to their respective new values. In other words, because of their transient identities and keys, users and wearable devices cannot be tracked by $\mathcal{A}$. So, we also accomplish untraceability with our scheme.

6) *De-Synchronization Attack*

In our proposed EAKE-WC scheme, the unique real identities, pseudo-random identities, and encryption keys are provided to wearable devices, users, and cloud server during the setup and user registration phases, respectively, i.e., the credentials $\{ID_i,\ IDS_i,\ k_W\}$ are stored in the memory of wearable device WD_i and also the respective credentials $\{(ID_i,\ IDS_i^{old} = null,\ IDS_i^{new} = IDS_i,\ k_W^{old} = null,\ k_W^{new} = k_W)\}$ are kept on CS. Likewise, MT_j store the credentials $\{(ID_j,\ rn_j,\ CT,\ MAC,\ IDS_j^{old} = null,\ IDS_j^{new} = IDS_j,\ k_U^{old} = null,\ k_U^{new} = k_U)\}$ in its own memory and also the respective credentials $\{(ID_j,\ IDS_j^{old} = null,\ IDS_j^{new} = IDS_j,\ k_U^{old} = null,\ k_U^{new} = k_U)\}$ are kept on CS. Since both the old and new pseudo-random identities and encryption keys are maintained, when the last acknowledgment message, for instance, between CS and U_j, is blocked by $\mathcal{A}$ or lost due to time delay, the old values can be retrieved. Similarly, between U_j and WD_i, the old values can be retrieved. Hence, our scheme resists de-synchronization attacks.

7) *Impersonation Attack*

An adversary $\mathcal{A}$ seeking to impersonate a wearable device WD_i would need to generate a correct MAC_1, or perform a replay attack as discussed in Subsection V-A1. However, since $\mathcal{A}$ lacks knowledge of ID_i and k_W, generating a valid MAC_1 is not possible. Hence, $\mathcal{A}$ is unable to impersonate wearable devices.

Similarly, suppose during the AKE phase, $\mathcal{A}$ intercepts M_1, which is transmitted to WD_i over a public channel. Using the obtained message as a basis, $\mathcal{A}$ attempts to impersonate the MT_j and construct a convincing message to persuade the other party that it is a legitimate mobile terminal. $\mathcal{A}$ generates a nonce rn_1^* and a timestamp TS_1^* and transmits $M_1 : \{rn_1^*,\ IDS_j,\ TS_1^*\}$ to WD_i. WD_i calculates $\{(C_1 \parallel C_2 \parallel C_3 \parallel C_4), MAC_1\} = E_{k_W}(rn_2 \oplus TS_3, IDS_i, (rn_1^* \parallel IDS_j \parallel ID_i \parallel rn_2))$ and transmits $M_2 : \{rn_2,\ IDS_i,\ MAC_1,\ TS_3\}$ to $\mathcal{A}$. Without knowing ID_i and k_W, $\mathcal{A}$ cannot generate a valid MAC_1. Next, upon obtaining M_2, MT_j computes M_3. Again, without knowing P_j and k_U, $\mathcal{A}$ cannot generate a valid MAC_2. As a result, $\mathcal{A}$ is unable to send a valid message to CS, which means that it cannot successfully impersonate the MT_j. Similarly, the impersonation of the cloud server can be proven with the same justification.

8) *Known key security*

Within our scheme, the concept of known key security entails that the security of other session keys should not be compromised if one session key is exposed. For instance, let's assume that an adversary $\mathcal{A}$ obtains session keys $SK_{CU} = C_9$ and $SK_{UD} = C_{13}$ for a particular session. It's important to note that these session keys are generated using unique temporary identities, nonces, timestamps, and dynamic encryption keys for each session. Therefore, the session keys for each session are distinct, ensuring that our scheme is capable of achieving known key security.

TABLE III: Queries and their Significance

Query	Significance
$Send(\Pi^t,\, msg)$	Using this query, $\mathcal{A}$ can forward a message msg to Π^t and acquire the response message.
$CorruptMT(\Pi_U^{t_1})$	Using this query, $\mathcal{A}$ can retrieve the secret parameters from the stolen mobile terminal.
$CorruptWD(\Pi_{WD}^{t_2})$	Using this query, $\mathcal{A}$ can retrieve the secret parameters from the stolen wearable device.
$Execute(\Pi_U^{t_1},\, \Pi_S^{t_2},\, \Pi_{WD}^{t_3})$	This query simulates an eavesdropping attack, allowing $\mathcal{A}$ to acquire the communicated messages among participants $\Pi_U^{t_1}$, $\Pi_S^{t_2}$, and $\Pi_{WD}^{t_3}$.
$Test(\Pi^t)$	Using this query, $\mathcal{A}$ requests Π^t for the SK, and Π^t responds probabilistically with an outcome of unbiased flipped coin b.
$Reveal(\Pi^t)$	Using this query, $\mathcal{A}$ reveals the SKs generated between $\Pi_U^{t_1}$ and $\Pi_{WD}^{t_3}$ and between $\Pi_U^{t_1}$ and $\Pi_S^{t_2}$.

9) ESL Attack

In the proposed EAKE-WC, CS computes $\{(C_9 \parallel C_{10} \parallel C_{11} \parallel C_{12}), MAC_3\} = E_{k_U}(rn_1 \oplus rn_3 \oplus TS_5,\ IDS_j, (rn_3 \parallel IDS_i \parallel P_j \parallel IDS_j))$ and $\{(C_{13} \parallel C_{14} \parallel C_{15} \parallel C_{16}), MAC_4\} = E_{k_W}(rn_2 \oplus TS_3,\ IDS_i, (rn_1 \parallel IDS_j \parallel ID_i \parallel rn_2))$. Then the acquired $MACs$ are verified, the ephemeral session key is set to be $SK_{CU} = C_9$ and $SK_{UD} = C_{13}$, between the user and CS, and between user and wearable device, respectively. Given that these session keys are generated employing long-term secret parameters ID_i, ID_j, and K, both session-dependent random numbers, i.e., rn_1, rn_2, and rn_3, and semi-permanent values, i.e., k_W and k_U, the adversary is incapable of retrieving the current or the previous ephemeral parameters, i.e., rn_1, rn_2, and rn_3. Furthermore, if SKs are leaked, the adversary benefits equal the benefits retrieved from the public parameter C_{12} and C_{16}. The keys k_W and k_U and other ephemeral session keys are, therefore, unaffected by the disclosure of an ephemeral session key. Hence, the proposed EAKE-WC can resist the ESL attack.

10) Non-Linkability

Our proposed EAKE-WC scheme is designed to ensure non-linkability among multiple messages originating from the same source. Specifically, the AKE phase, which involves the user, wearable device, and cloud server, is tailored to support dynamic messaging by utilizing unique random numbers, timestamps, and encryption keys for each session. This dynamic approach guarantees that the interaction information of the user, wearable device, and cloud server are uncorrelated, thereby preventing adversaries from extracting sensitive credentials from any of these sources. Thus, the AKE scheme maintains the non-linkability feature and ensures information security for all parties involved in the messaging process.

B. Security Analysis Using ROR Model

In this subsection, we utilize the ROR model to evaluate the session key (SK) security of the proposed EAKE-WC in Theorem 1 against active/passive adversary $\mathcal{A}$. Before revealing the semantic security of SK for EAKE-WC, this section put forward the ROR concepts.

In the proposed EAKE-WC, there are three participants, i.e., the user $\Pi_U^{t_1}$, the cloud server $\Pi_{CS}^{t_2}$, and the wearable device $\Pi_{WD}^{t_3}$, where $\Pi_U^{t_1}$, $\Pi_{CS}^{t_2}$, and $\Pi_{WD}^{t_3}$ are instances t_1^{th} of U_j, t_2^{th} of CS, and t_3^{th} of WD_i, respectively. To perform the formal (mathematical) security analysis, Table III lists different queries to the ROR model, including "$Execute()$", "$Send()$", "$Corrupt()$", "$Test()$", and "$Reveal()$". Additionally, the "collision-resistant one-way hash function $h(\cdot)$" is employed as a random oracle.

Definition 1. ***(Semantic security).*** *Let $Adv_{\mathcal{A}}^{EAKE-WC}(pl_t)$ be an adversary $\mathcal{A}$'s advantage executing in polynomial-time pl_t to compromise the semantic security of the proposed EAKE-WC scheme to retrieve the session keys among the communicating entities. Then, $Adv_{\mathcal{A}}^{EAKE-WC}(pl_t) = |2 \cdot Prob[b = b'] - 1|$, where b and b' signify the "correct" and "guessed" bits, respectively.*

Definition 2. *Let $\mathcal{A}$ is running against the AEAD scheme in polynomial-time pl_t and transmitting 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 as follows:*

$$Adv_{\phi,\mathcal{A}}^{OCCA3}(Q_{ue},\, L_{en},\, pl_t) \leq Adv_{\phi}^{OPRP-CPA}(Q_{ue},\, L_{en},\, pl_t) + Adv_{\phi}^{INT-CT}(Q_{ue},\, L_{en},\, pl_t), \tag{3}$$

where $Adv_{\phi}^{OPRP-CPA}(Q_{ue},\, L_{en},\, pl_t)$ represents $\mathcal{A}$'s advantage on "online pseudo-random permutation chosen-plaintext" attack and $Adv_{\phi}^{INT-CT}(Q_{ue},\, L_{en},\, pl_t)$ denotes $\mathcal{A}$'s advantage on the integrity of the ciphertext.

Theorem 1. *During the AKE phase, $\mathcal{A}$ launches attacks against the proposed EAKE-WC in polynomial time (pl_t) in an attempt to retrieve the shared session keys among the user U_j, server S, and wearable device WD_i. The advantage of $\mathcal{A}$ in compromising the security of the session key is then approximately as follows.*

$$Adv_{\mathcal{A}}^{EAKE-WC}(pl_t) \leq \frac{HQ_u^2}{|SHA|} + 2 \cdot C' Q_s^{s'} + 2 \cdot Adv_{ASCON,\mathcal{A}}^{OCCA3}(Q_{ue},\, L_{en},\, pl_t), \tag{4}$$

*where HQ_u, Q_s, SHA, and $Adv_{ASCON,\mathcal{A}}^{OCCA3}(QR,\, L_{ED},\, pl_t)$ signify hash queries, send queries, range space of $h(\cdot)$, and advantage of $\mathcal{A}$ in compromising the security of an online AEAD scheme (ASCON) (**Definition** 2), respectively, and C' and s' signify the Zipf's parameters [36].*

Proof. Let $\mathcal{A}$ play a series of five games ($Game_i^{\mathcal{A}} | i \in [0, 4]$) to breach the semantic security of SK and $Succ_i$ denote the success probability in which $\mathcal{A}$ win game $Game_i^{\mathcal{A}}$ in pl_t. The description of each game $Game_i^{\mathcal{A}}$ is given as follows:

$Game_0^{\mathcal{A}}$: This game represents a real attack by $\mathcal{A}$ against the proposed EAKE-WC scheme. The decision is made by flipping an unbiased coin, and it follows that the proposed scheme's semantic security is given in **Definition** 1 that

$$\mathrm{Adv}_{\mathcal{A}}^{EAKE-WC}(t_p) = |2 \cdot \mathrm{Prob}[Succ_0] - 1|. \tag{5}$$

$Game_1^{\mathcal{A}}$: In this game, we assume an eavesdropping attack against the proposed EAKE-WC in which $\mathcal{A}$ eavesdrop on all the transmitted messages among U_j, WD_i, and S during the AKE procedure. Then $\mathcal{A}$ runs $Execute(\Pi_U^{t_1}, \Pi_S^{t_2}, \Pi_{WD}^{t_3})$ query, followed by "$Test$" and "$Reveal$" queries to check that the session keys i.e., $SK_{U_jD_i} = SK_{D_iU_j}$ and $SK_{U_jC} = SK_{CU_j}$ are valid. It is worth mentioning that the SK between the U_j and WD_i and between CS and U_j are computed using both long-term and short-term secrets, as discussed in Section IV. It is computationally hard for $\mathcal{A}$ to compute both session keys, and the probability of winning $Game_1^{\mathcal{A}}$ is not changed from that of $Game_0^{\mathcal{A}}$. Therefore, the indistinguishability of $Game_0^{\mathcal{A}}$ and $Game_1^{\mathcal{A}}$ gives:

$$\text{Prob}[Succ_1] = \text{Prob}[Succ_0]. \tag{6}$$

$Game_2^{\mathcal{A}}$: In this game, $\mathcal{A}$ tries to lunch an active attack by executing the SHA and $Send$ queries. $\mathcal{A}$ runs numerous SHA queries in order to find $h(\cdot)$ (i.e., SHA-256) collisions. Given that the transmitted messages include timestamps and random integers, there is nearly no chance of a collision when using the $Send$ query. As a result, retrieving the secret parameters is impossible for the $\mathcal{A}$. Thus, by birthday paradox, we get

$$|\text{Prob}[Succ_2] - \text{Prob}[Succ_1]| \leq \frac{HQ_u^2}{2|SHA|}. \tag{7}$$

$Game_3^{\mathcal{A}}$: This game simulates stolen/lost mobile terminal and password-guessing attacks. By utilizing $CorruptMT(\Pi_U^{t_1})$ query, $\mathcal{A}$ gets $\{rn_j,\ CT,\ MAC,\ IDS_j^{old},\ IDS_j^{new},\ k_U^{old},\ k_U^{new}\}$ from a stolen/lost MT_j. $\mathcal{A}$ then tries to extract the encrypted secret parameter P_j. To succeed in this game, $\mathcal{A}$ needs to know both ID_i and PW_i. If we limit the system's allowable number of incorrect password inputs. Therefore, based on the findings of the birthday paradox and Zipf's rule on passwords, we have

$$|\text{Prob}[Succ_3] - \text{Prob}[Succ_2]| \leq C' Q_s^{s'}. \tag{8}$$

$Game_4^{\mathcal{A}}$: Finally, in $Game_4^{\mathcal{A}}$, $\mathcal{A}$ simulates an active attack by eavesdropping on the transmitted messages, such as $M_1 : \{rn_1,\ IDS_j,\ TS_1\}$, $M_2 : \{rn_2,\ IDS_i,\ MAC_1,\ TS_3\}$, $M_3 : \{M_2,\ rn_1,\ rn_3,\ IDS_j,\ MAC_2,\ TS_5\}$, $M_4 : \{C_{12},\ C_{16},\ X_5,\ MAC_5,\ TS_7\}$, and $M_5 : \{C_{16},\ TS_9\}$. $\mathcal{A}$ attempt to obtain the secret parameters needed to create the SK after capturing these messages. However, an AEAD primitive ASCON is employed to encrypt the secret credentials. As a result, $\mathcal{A}$ cannot decrypt the encrypted data. Therefore, based on Definition 2, we can conclude

$$|\text{Prob}[Succ_4] - \text{Prob}[Succ_3]| \leq \text{Adv}_{ASCON,\mathcal{A}}^{OCCA3}(Q_{ue},\ L_{en},\ pl_t). \tag{9}$$

A "$Test$" query is executed by $\mathcal{A}$, where a fair coin is flipped and SK's semantic security is decided after playing all the games. As a result

$$\text{Prob}[Succ_4] = \frac{1}{2}. \tag{10}$$

Therefore, from (5) we have

$$\frac{1}{2}\text{Adv}_{\mathcal{A}}^{EAKE-WC}(pl_t) = \left|\text{Prob}[Succ_0] - \frac{1}{2}\right|. \tag{11}$$

Using (10) and (11) as well as noting (6), we obtain

$$\begin{aligned}\frac{1}{2}\text{Adv}_{\mathcal{A}}^{EAKE-WC}(pl_t) &= |\text{Prob}[Succ_0] - \text{Prob}[Succ_4]| \\ &= |\text{Prob}[Succ_1] - \text{Prob}[Succ_4]|. \end{aligned} \tag{12}$$

By employing the well-known triangle inequality to (12), we have

$$\begin{aligned}\frac{1}{2}\text{Adv}_{\mathcal{A}}^{EAKE-WC}(pl_t) \leq\ & |\text{Prob}[Succ_1] - \text{Prob}[Succ_2]| \\ &+ |\text{Prob}[Succ_2] - \text{Prob}[Succ_3]| \\ &+ |\text{Prob}[Succ_3] - \text{Prob}[Succ_4]|. \end{aligned} \tag{13}$$

When (7), (8), and (9) into (13), we get

$$\begin{aligned}\text{Adv}_{\mathcal{A}}^{EAKE-WC}(pl_t) \leq\ & \frac{HQ_u^2}{|SHA|} + 2 \cdot C' Q_s^{s'} \\ &+ 2 \cdot \text{Adv}_{ASCON,\mathcal{A}}^{OCCA3}(Q_{ue},\ L_{en},\ pl_t). \end{aligned} \tag{14}$$

i.e., (4). This completes the proof. ■

VI. Comparative Analysis

In this section, we compare the proposed EAKE-WC in terms of security and functionality characteristics and computation and communication overheads with eminent benchmark schemes, such as Amin *et al.* [23], Gupta *et al.* [18], Wang *et al.* [32], Guo *et al.* [28], and Wazid *et al.* [31]. Further, as wearable devices are resource-limited, we also focus on comparing the computation and communication overheads of wearable devices in these eminent benchmark schemes.

A. Computation Overhead Comparison

Reducing the computational overhead is a design objective of an AKE scheme while preserving security and functionality traits. To compute the computational overheads of the proposed EAKE-WC and the related eminent benchmark schemes, we considered the experimental results reported in [37]. The approximated execution time of various cryptographic primitives is given in Table IV. Let T_a, T_{em}, T_{fe}, T_h, and T_{se}/T_{sd} denote the time required for ASCON encryption/ decryption, ECC point multiplication, fuzzy extractor, hash function, and symmetric encryption/ decryption function, respectively. The computational overhead of the proposed EAKE-WC and eminent benchmark schemes, such as Amin *et al.* [23], Gupta *et al.* [18], Wang *et al.* [32], Guo *et al.* [28], and Wazid *et al.* [31], are compared in Table V and Fig. 3. In our proposed EAKE-WC, the computation overhead needed for a resource-limited wearable device is $T_a \approx 0.309$, lower than all of the benchmark schemes, which reveals that our proposed EAKE-WC is more appropriate

TABLE IV: Execution Time for Various Primitives [37]

Primitive	EXT (ms)
T_a: ASCON	0.309
T_{em}: ECC point multiplication	2.21
$T_{fe} \approx T_{em}$: Fuzzy extractor	2.21
T_h: Hash function	0.292
T_{se}/T_{sd}: Symmetric encryption/ decryption	0.325

Note: EXT (ms): Approximate execution time (ms)

TABLE V: Computational Overhead Comparison

Scheme	User	Cloud server	Wearable device	TE (ms)
Amin *et al.* [23]	$12T_h \approx 3.504$	$16T_h \approx 4.672$	$6T_h \approx 1.752$	$34T_h \approx 9.928$
Gupta *et al.* [18]	$7T_h \approx 2.044$	$5T_h \approx 1.46$	$4T_h \approx 1.168$	$16T_h \approx 4.672$
Wang *et al.* [32]	$6T_h \approx 1.752$	$4T_h \approx 1.168$	$3T_h \approx 0.876$	$13T_h \approx 3.796$
Guo *et al.* [28]	$21T_h \approx 6.132$	$18T_h \approx 5.256$	$7T_h \approx 2.044$	$46T_h \approx 13.432$
Wazid *et al.* [31]	$T_{fe} + 15T_h \approx 6.59$	$8T_h \approx 2.336$	$7T_h \approx 2.044$	$T_{fe} + 30T_h \approx 10.97$
EAKE-WC	$2T_h + 2T_a \approx 1.202$	$2T_h + 2T_a \approx 1.202$	$T_a \approx 0.309$	$4T_h + 5T_a \approx 2.713$

Note: TE (ms): Approximate total execution time in milliseconds.

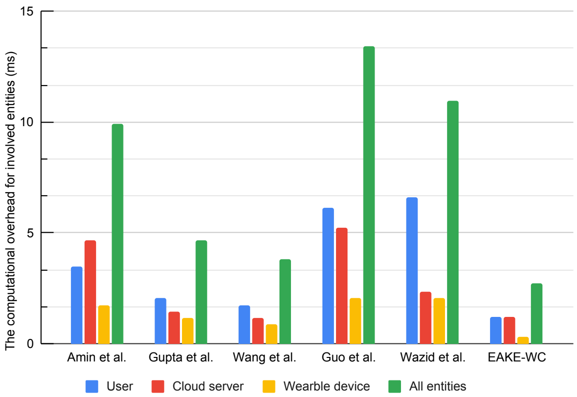

Fig. 3: Computational overhead of EAKE-WC and related benchmark schemes.

for resource-limited environments. The total computational overhead of the proposed EAKE-WC during the login and AKE procedure is $4T_h + 5T_a \approx 2.713$, which is lower than all of the benchmark schemes. Therefore, our proposed EAKE-WC is more efficient and suitable for resource-limited environments.

B. Communication Overhead Comparison

One of the essential design objectives of an AKE scheme in a resource-limited environment is to reduce communication overhead while maintaining security. For comparison purposes, we make the following assumptions: timestamps, message authentication codes, i.e., $MACs$ generated by ASCON functions, random numbers, all identities, and hash digest of sizes 32 bits, 128 bits, 128 bits, 128 bits, and 256 bits, respectively. Table VI illustrates the communication overhead of our proposed EAKE-WC and the benchmark schemes in the login and authentication procedures. The communication overheads of wearable devices are 512 bits, 800 bits, 448 bits, 672 bits, and 544 bits, respectively, in the schemes of Amin *et al.* [23], Gupta *et al.* [18], Wang *et al.* [32], Guo *et al.* [28], and Wazid *et al.* [31]. In our proposed EAKE-WC, the wearable device transmits only message M_2, which requires 416 bits. Therefore, our proposed EAKE-WC is more appropriate for resource-limited environments. Furthermore, our proposed EAKE-WC requires sending five messages, i.e., $M_1 \sim M_5$, during the login and AKE procedure, which demands 288 bits, 416 bits, 960 bits, 672 bits, and 160 bits, respectively. Thus, the total communication overhead of the proposed EAKE-WC scheme is 2496 bits. Table VI and Fig. 4 reveal that our proposed EAKE-WC has less communication overhead than the Amin *et al.* [23], Gupta *et al.* [18], Guo *et al.* [28], and Wazid *et al.* [31]. Although the total communication overhead of the proposed EAKE-WC as compared to the scheme of Wang *et al.* [32] is slightly higher. However, only the proposed EAKE-WC sustains more security and functionality traits (see Table 4).

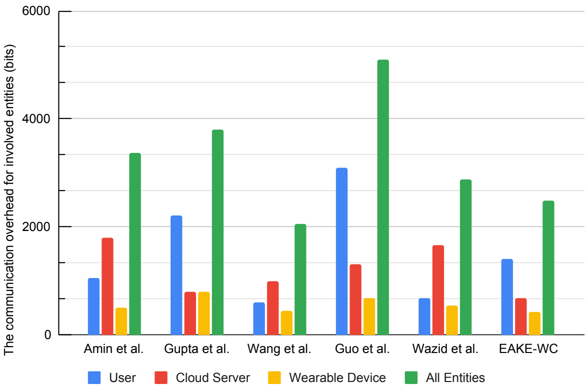

Fig. 4: Communication overhead of EAKE-WC and related benchmark schemes.

TABLE VI: Communication Overhead Comparison

Scheme	No. of Transmitted Messages	CW (bits)	TO (bits)
Amin *et al.* [23]	4	512	3360
Gupta *et al.* [18]	5	800	3808
Wang *et al.* [32]	5	448	2048
Guo *et al.* [28]	5	672	5088
Wazid *et al.* [31]	4	544	2880
Proposed EAKE-WC	5	416	2496

Note: TO (bits): Total communication overhead (bits), CW (bits): communication overhead of WD_i (bits)

C. Security and Functionality Features Comparison

Table VII outlines the comparison of the proposed EAKE-WC and other competing benchmarks based on the set of fourteen security and functionality attributes, namely, $\mathcal{SF}_1$: mutual authentication; $\mathcal{SF}_2$: replay attack; $\mathcal{SF}_3$: impersonation attacks; $\mathcal{SF}_4$: untraceability; $\mathcal{SF}_5$: mobile terminal theft attack; $\mathcal{SF}_6$: wearable device capture/theft attack; $\mathcal{SF}_7$: MitM attack; $\mathcal{SF}_8$: anonymity; $\mathcal{SF}_9$: non-linkability; $\mathcal{SF}_{10}$: password guessing attack; $\mathcal{SF}_{11}$: known key security; $\mathcal{SF}_{12}$: ESL attack; $\mathcal{SF}_{13}$:

TABLE VII: Security and Functionality Features Analysis

↓Feature/ Scheme→	[23]	[18]	[32]	[28]	[31]	EAKE-WC
$\mathcal{SF}_1$: Mutual authentication	✓	✓	×	✓	✓	✓
$\mathcal{SF}_2$: Replay attack	✓	✓	✓	✓	✓	✓
$\mathcal{SF}_3$: Impersonation attacks	✓	✓	✓	✓	✓	✓
$\mathcal{SF}_4$: Untraceability	×	×	✓	✓	✓	✓
$\mathcal{SF}_5$: MTC attack	✓	✓	✓	✓	✓	✓
$\mathcal{SF}_6$: WDC attack	✓	✓	✓	✓	✓	✓
$\mathcal{SF}_7$: MitM attack	✓	✓	×	✓	✓	✓
$\mathcal{SF}_8$: Anonymity	×	×	✓	✓	✓	✓
$\mathcal{SF}_9$: Non-linkability	×	×	✓	✓	✓	✓
$\mathcal{SF}_{10}$: Password guessing attack	✓	✓	✓	✓	✓	✓
$\mathcal{SF}_{11}$: Known key security	✓	✓	✓	✓	✓	✓
$\mathcal{SF}_{12}$: ESL attack	✓	✓	×	✓	✓	✓
$\mathcal{SF}_{13}$: Validated through FM	✓	✓	×	✓	×	✓
$\mathcal{SF}_{14}$: De-synchronization attack	×	×	✓	✓	×	✓

Note: ✓: signifies the feature's availability, ×: denotes the feature's unavailability; FM: formal (mathematical) model, WDC attack: wearable device capture/theft attack, MTC attack: Mobile terminal theft attack.

validated through formal (mathematical) model; and $\mathcal{SF}_{14}$: desynchronization attack.

Our proposed EAKE-WC and Guo *et al.* [28] renders more functionality features and enhance security. However, the computation and communication overheads of Guo *et al.* [28] are higher than our proposed EAKE-WC scheme.

D. *Critical Discussion*

The wearable computing environment involves multiple entities communicating through insecure public channels, making them vulnerable to various security threats and attacks. To address these concerns, we developed the EAKE-WC scheme, which facilitates authentication between users and wearable devices with the assistance of a cloud server. Our proposed scheme allows users, wearable devices, and cloud server to authenticate and establish secret session keys for secure communication. We conducted a rigorous security analysis, demonstrating that our EAKE-WC is robust against potential security attacks and meets session key security requirements. Our proposed EAKE-WC scheme offers improved performance compared to most benchmarks, achieved by reducing computational and communication overheads while introducing additional security and functional features. This improved performance is due to the use of ultra-lightweight cryptography technology, including XOR, ASCON, and hash functions, to minimize the computational and communication overheads of AKE procedures. Our proposed EAKE-WC scheme lacks a credential revocation mechanism, which prevents it from achieving conditional privacy protection. As a result, our aim is to develop an enhanced authentication scheme in the future that can incorporate conditional privacy while also supporting the revocation mechanism.

VII. Conclusion

The present study introduced EAKE-WC, an effective and anonymous authenticated key exchange scheme suitable for wearable computing. By utilizing lightweight cryptography primitives like XOR, ASCON, and the hash function, we established two secret session keys, ensuring secure communication between the wearable device, the user, and the cloud server. The security analysis demonstrated that EAKE-WC complied with the security requirements of wearable computing. Additionally, the comparative analysis showed that our proposed scheme surpassed state-of-the-art benchmark schemes regarding communication and computational overhead. As a result, EAKE-WC delivered improved security and performance in the wearable computing environment.

Despite offering security against common attacks in wearable computing, we acknowledge the potential for physical tampering attacks on the wearable device. Therefore, we propose future work to incorporate Physical Unclonable Functions (PUFs) into the design of EAKE-WC to address this limitation. Additionally, we intend to evaluate EAKE-WC in a real-world setting to refine the scheme further, if required, to provide enhanced security and better performance.

Acknowledgement

The authors extend their appreciation to the Ministry of Education in KSA for funding this work through the project number KKU-IFP2-H-15.

References

[1] S. Seneviratne *et al.*, "A survey of wearable devices and challenges," *IEEE Commun. Surv. Tutor.,* vol. 19, no. 4, pp. 2573-2620, Jul. 2017.

[2] A. Pantelopoulos and N. G. Bourbakis, "A survey on wearable sensor-based systems for health monitoring and prognosis," *IEEE Trans. Syst. Man Cybern. Part C (Applications and Reviews)*, vol. 40, no. 1, pp. 1-12, Jan. 2010.

[3] O. Arias, J. Wurm, K. Hoang and Y. Jin, "Privacy and security in Internet of things and wearable devices," *IEEE Trans. Multi-Scale Comput. Syst.*, vol. 1, no. 2, pp. 99-109, April-June 2015.

[4] J. Zhou, Z. Cao, X. Dong, and X. Lin, "Security and privacy in cloud-assisted wireless wearable communications: Challenges, solutions, and future directions," *IEEE Wirel. Commun.*, vol. 22, no. 2, pp. 136-144, Apr. 2015.

[5] Y. K. Ever, "Secure-anonymous user authentication scheme for e-Healthcare application using wireless medical sensor networks," *IEEE Syst. J.*, vol. 13, no. 1, pp. 456-467, Mar. 2019.

[6] 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," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 1739-1755, Feb. 2023.

[7] A. Badshah, M. Waqas, F. Muhammad, G. Abbas and Z. H. Abbas, "A Novel Framework for Smart Systems Using Blockchain-Enabled Internet of Things," in *IT Professional*, vol. 24, no. 3, pp. 73-80, May 2022.

[8] D. He, S. Zeadally, N. Kumar, and J. H. Lee, "Anonymous authentication for wireless body area networks with provable security," *IEEE Syst. J.*, vol. 11, no. 4, pp. 2590-2601, 2017.

[9] J. Shen, Z. Gui, S. Ji, J. Shen, H. Tan, and Y. Tang, "Cloud-aided lightweight certificateless authentication protocol with anonymity for wireless body area networks," *J. Netw. Comput. Appl.,* vol. 106, pp. 117-123, Mar. 2018.

[10] J. Liu and K. S. Kwak, "Hybrid security mechanisms for wireless body area networks," *ICUFN 2010 - 2nd International Conference on Ubiquitous and Future Networks,* (Jeju, South Korea), 16-18 June 2010, pp. 98-103.

[11] Q. Xie, D. S. Wong, G. Wang, X. Tan, K. Chen, and L. Fang, "Provably secure dynamic ID-based anonymous two-factor authenticated key exchange protocol with extended security model," *IEEE Trans. Inf. Forensics Secur.,* vol. 12, no. 6, pp. 1382-1392, Jan. 2017.

[12] L. Zhang, Y. Zhang, S. Tang, and H. Luo, "Privacy protection for e-health systems by means of dynamic authentication and three-factor key agreement," *IEEE Trans. Ind. Electron.,* vol. 65, no. 3, pp. 2795-2805, Mar. 2018.

[13] M. Wazid, A. K. Das, V. Odelu, N. Kumar, and W. Susilo, "Secure remote user authenticated key establishment protocol for smart home environment," *IEEE Trans. Dependable Secure Comput.,* vol. 17, no. 2, pp. 391-406, 1 Mar.-Apr. 2020.

[14] B. Gong, G. Zheng, M. Waqas, S. Tu and S. Chen, "LCDMA: Lightweight Cross-domain Mutual Identity Authentication Scheme for Internet of Things," in *IEEE Internet of Things Journal*, Mar. 2023.

[15] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schläffer, “Ascon v1.2. Submission to NIST (2019),” https://ascon.iaik.tugraz.at, accessed: 2022-11-14.

[16] D. Wang, D. He, P. Wang, and C. -H. Chu, “Anonymous Two-Factor Authentication in Distributed Systems: Certain goals are beyond attainment,” *IEEE Trans. Dependable Secure Comput.*, vol. 12, no. 4, pp. 428-442, July-Aug. 2015.

[17] X. Huang, X. Chen, J. Li, Y. Xiang, and L. Xu, “Further observations on smart-card-based password-authenticated key agreement in distributed systems,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 7, pp. 1767-1775, Jul. 2014.

[18] A. Gupta, M. Tripathi, T. J. Shaikh, and A. Sharma, “A lightweight anonymous user authentication and key establishment scheme for wearable devices,” *Comput. Netw.*, vol. 149, pp. 29-42, Feb. 2019.

[19] R. Hajian, S. ZakeriKia, S. H. Erfani, and M. Mirabi, “SHAPARAK: Scalable healthcare authentication protocol with attack-resilience and anonymous key-agreement,” *Comput. Netw.*, vol. 183, article no. 107567, Dec. 2020.

[20] S. Liu, S. Hu, J. Weng, S. Zhu, and Z. Chen, “A novel asymmetric three-party based authentication scheme in wearable devices environment,” *J. Netw. Comput. Appl.*, vol. 60, pp. 144-154, Jan. 2016.

[21] F. Wu, L. Xu, S. Kumari, and X. Li, “An improved and anonymous two-factor authentication protocol for healthcare applications with wireless medical sensor networks,” *Multimedia Syst.*, vol. 23, no. 2, pp. 195-205, Mar. 2017.

[22] X. Li, J. Niu, S. Kumari, J. Liao, W. Liang, and M. K. Khan, “A new authentication protocol for healthcare applications using wireless medical sensor networks with user anonymity,” *Secur. Commun. Netw.*, vol. 9, no. 15, pp. 2643-2655, Oct. 2016.

[23] R. Amin, S. H. Islam, G. P. Biswas, M. K. Khan, and N. Kumar, “A robust and anonymous patient monitoring system using wireless medical sensor networks,” *Future Gener. Comput. Syst.*, vol. 80, pp. 483-495, Mar. 2018.

[24] R. Ali, A. K. Pal, S. Kumari, A. K. Sangaiah, X. Li, and F. Wu, “An enhanced three-factor based authentication protocol using wireless medical sensor networks for healthcare monitoring,” *J. Ambient Intell. Humanized Comput.*, vol. 269, pp. 1-22, Sep. 2018.

[25] A. K. Das, M. Wazid, N. Kumar, M. K. Khan, K. K. R. Choo, and Y. H. Park, “Design of secure and lightweight authentication protocol for wearable devices environment,” *IEEE J. Biomed. Health Inform.*, vol. 22, no. 4, pp. 1310-1322, Jul. 2018.

[26] Q. Jiang, Y. Qian, J. Ma, X. Ma, Q. Cheng, and F. Wei, “User centric three-factor authentication protocol for cloud-assisted wearable devices,” *Int. J. Commun. Syst.*, vol. 32, no. 6, pp. 1-20, Apr. 2019.

[27] J. Srinivas, A. K. Das, N. Kumar, and J. Rodrigues, “Cloud centric authentication for wearable healthcare monitoring system,” *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 5, pp. 942-956, Sep./Oct. 2020.

[28] Y. Guo, Z. Zhang, and Y. Guo, “Anonymous authenticated key agreement and group proof protocol for wearable computing,” *IEEE Trans. Mobile Comput.*, vol. 21, no. 8, pp. 2718-2731, Aug. 2022.

[29] M. Fotouhi, M. Bayat, A. K. Das, H. A. N. Far, S. M. Pournaghi, and M. A. Doostari, “A lightweight and secure two-factor authentication scheme for wireless body area networks in health-care IoT,” *Comput. Netw.,* vol. 177, pp. 1-16, Aug. 2020.

[30] J. Li, Z. Su, D. Guo, K.-K. R. Choo, and Y. Ji, “PSL-MAAKA: Provably secure and lightweight mutual authentication and key agreement protocol for fully public channels in Internet of Medical Things,” *IEEE Internet Things J.,* vol. 8, no. 17, pp. 13183-13195, Sep. 2021.

[31] M. Wazid, S. Thapliyal, D. P. Singh, A. K. Das, and S. Shetty, “Design and testbed experiments of user authentication and key establishment mechanism for smart healthcare cyber-physical systems,” *IEEE Trans. Netw. Sci. Eng.,* early access, Mar. 2022.

[32] W. Wang *et al.*, “Blockchain and PUF-based lightweight authentication protocol for wireless medical sensor networks,” *IEEE Internet Things J.,* vol. 9, no. 11, pp. 8883-8891, June 2022.

[33] S. Yu and Y. Park, “A robust authentication protocol for wireless medical sensor networks using blockchain and physically unclonable functions,” *IEEE Internet Things J.,* vol. 9, no. 20, pp. 20214-20228, Oct. 2022.

[34] 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.

[35] R. Canetti and H. Krawczyk, “Universally composable notions of key exchange and secure channels,” in *Proc. EUROCRYPT 2002* (Amsterdam, The Netherlands), Apr. 28-May 2, 2002, pp. 337–351.

[36] D. Wang, H. Cheng, P. Wang, X. Huang, and G. Jian, “Zipf’s Law in Passwords,” *IEEE Trans. Inf. Forensics Secur.,* vol. 12, no. 11, pp. 2776-2791, Nov. 2017.

[37] M. Tanveer, G. Abbas, Z. H. Abbas, M. Bilal, A. Mukherjee, and K. S. Kwak, “LAKE-6SH: Lightweight user authenticated key exchange for 6LoWPAN-based smart homes,” *IEEE Internet of Things J.,* vol. 9, no. 4, pp. 2578-2591, Feb. 2022.

Shanshan Tu received his PhD degree from Computer Science Department at Beijing University of Posts and Telecommunications in 2014. From 2013 to 2014, he visited University of Essex for National Joint Doctoral Training. He worked in the Department of Electronic Engineering at Tsinghua University as a postdoctoral researcher from 2014 to 2016. He is currently an Associate Professor and Deputy Dean at Faculty of Information Technology, Beijing University of Technology, China. His research interests are in the areas of cloud computing, MEC and information security techniques.

Akhtar Badshah received PhD degree with the Faculty of Computer Science and Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, Pakistan. Currently, he is a Lecturer with the Department of Software Engineering, University of Science and Technology, Bannu, Pakistan. Since 2014, he has been with the Department of Software Engineering, University of Malakand, Pakistan, as a Lecturer. His research interests include cryptography, blockchain, and IoT security.

Hisham Alasmary is an Assistant Professor at King Khalid University. He obtained his Ph.D. from the Department of Computer Science at the University of Central Florida in 2020, and his M.Sc. degree in Computer Science from The George Washington University, in Washington, D.C., USA, in 2016. His research interests include Software Security, IoT Security and Privacy, ML/DL Applications in Information Security, and Adversarial Machine Learning.

MUHAMMAD WAQAS (M'18, SM'22) received his B.Sc. and M.Sc. degrees from the Department of Electrical Engineering, University of Engineering and Technology Peshawar, Pakistan, in 2009 and 2014, respectively. From 2012 to 2015, he served Sarhad University of Science and Information Technology, Peshawar, Pakistan, as a Lecturer and program coordinator. Dr Muhammad Waqas pursued his PhD degree (Sept. 2015 - Jun. 2019) with the Department of Electronic Engineering, Tsinghua University, Beijing, China. From Aug. 2019 to Mar.. 2022, he served Faculty of Computer Science and Engineering, GIK Insitute of Engineering Sciences and Technology, Pakistan as an Assistant Professor. He was also associated with the Faculty of Information Technology, Beijing University of Technology, Beijing, China as a Research Associate from Oct. 2019 to Sept. 2022. Currently, he is an Assistant Professor at the Computer Engineering Department, College of Information Technology, University of Bahrain, Bahrain. He is also an Adjunct Senior Lecturer at the School of Engineering, Edith Cowan University, Perth, Australia. He has several research publications in reputed Journals and Conferences. He is co-chair, TPC member and reviewer of several international conferences and journals. He is also an Associate Editor of the International Journal of Computing and Digital Systems, and received the best paper award at ASSP in 2021. His current research interests are in the areas of physical layer security, vehicular networks, mobile edge computing and the internet of things.