IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 28 April 2024, accepted 24 May 2024, date of publication 30 May 2024, date of current version 10 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3407353

RESEARCH ARTICLE

PSAF-IoT: Physically Secure Authentication Framework for the Internet of Things

OMAR ALRUWAILI[1], FAISAL MOHAMMED ALOTAIBI[2], MUHAMMAD TANVEER[3], SLIM CHAOUI[1], AND AMMAR ARMGHAN[4], (Senior Member, IEEE)

[1]Department of Computer Engineering and Networks, College of Computer and Information Sciences, Jouf University, Sakakah, Al-Jouf 72388, Saudi Arabia
[2]Department of Computer Science, Prince Sattam Bin Abdulaziz University, Al-Kharj, Riyadh 16278, Saudi Arabia
[3]School of Systems and Technology (SST), University of Management and Technology, Lahore 54770, Pakistan
[4]Department of Electrical Engineering, College of Engineering, Jouf University, Sakaka 72388, Saudi Arabia

Corresponding author: Ammar Armghan (aarmghan@ju.edu.sa)

This work was funded by the Deanship of Scientific Research at Jouf University under Grant DSR2022-NF-15.

ABSTRACT The Internet of Things (IoT) revolutionizes connectivity, as IoT devices grow exponentially, vulnerabilities emerge, ranging from data breaches to device hijacking. Thus there is the critical need for robust security measures, including encryption protocols and authentication mechanisms, to safeguard against cyber threats. Despite advancements in the authentication mechanism, still IoT security remains an ongoing concern. Because most of the authentications are vulnerable to diverse security attacks and other resource intensive. Thus protecting user data in this increasingly interconnected IoT-enabled world requires a secure and lightweight authentication mechanism. In this direction, in this paper, we propose a physically secure authentication framework for the IoT (PSAF-IoT). PSAF-IoT utilizes a combination of physical unclonable functions, secure hash algorithm, and elliptic curve cryptography to establish robust security measures. It guarantees the creation of a secure channel (session key) following user authentication at the gateway node, allowing the user to use the established secure channel for future communication. The secure channel establishment procedure is validated for security by employing formal methods such as the random oracle model and Scyther-based simulations. Additionally, PSAF-IoT undergoes informal validation to demonstrate resilience against node capture, replay attacks, impersonation, and other common security threats. Notably, PSAF-IoT demonstrates efficiency in terms of execution time, energy consumption, and communication costs, as evidenced by comparative analyses with related authentication frameworks, all while enhancing information security functionalities.

INDEX TERMS Session key, authentication, elliptic curve cryptography, security, Internet of Things, PUF.

I. INTRODUCTION

The Internet of Things (IoT) has appeared as a transformative force, revolutionizing the way we interact with technology and the physical world [1]. By interconnecting everyday devices and objects, IoT promises remarkable convenience, efficiency, and innovation across various domains, including healthcare, transportation, agriculture, and manufacturing. However, this instantaneous expansion of interconnected IoT devices also brings forth noteworthy challenges, particularly in the realm of information security [2]. As IoT ecosystems continue to grow, concerns encompassing data privacy, integrity, and security have become increasingly prominent [3]. The inherent vulnerabilities associated with IoT devices, such as limited computational resources, lack of standardized security protocols, and susceptibility to cyber attacks, pose serious risks to both individuals and organizations. Unauthorized access, data breaches, and malicious exploitation of IoT devices can lead to severe consequences, ranging from compromised personal information to widespread disruption of critical infrastructure [4].

A specific example of critical infrastructure is a smart factory, where various resource-constrained IoT devices are deployed to perform actuation, sensing, and monitoring tasks.

The associate editor coordinating the review of this manuscript and approving it for publication was Parul Garg.

The administrator of the smart factory monitors these IoT devices remotely from an office by sending various control commands [5]. Since these commands are sensitive and transmitted over a public communication channel susceptible to security attacks, they can be compromised by attackers. An attacker could modify control commands and other sensitive information exchanged between the IoT devices and the administrator, disrupting the smart factory's production process [6]. Therefore, it is crucial to ensure that only authenticated users can communicate with the IoT-enabled smart factory network, and the information exchanged between the administrator and IoT devices must be encrypted [7]. Addressing these security concerns necessitates a comprehensive approach that ensures information security within the IoT environment. One notably promising solution that has garnered attention in recent years is authentication and key agreement (AKA), a mechanism to enhance information security in IoT environments. In this context, numerous AKA frameworks have been proposed for the IoT environment [8]. However, many of them are vulnerable to various security attacks, while others are computationally intensive. Additionally, most of these frameworks do not offer physical security measures. This paper endeavors to delve into the intricacies of AKA and design a framework tailored to bolster information security within the IoT ecosystem. Through an in-depth exploration of AKA principles, key features, and real-world implementations, our objective is to develop an AKA framework that enables secure communication within the IoT environment. Additionally, the proposed framework must offer resistance to various security attacks and ensure physical security.

II. RELATED WORK

AKA frameworks play a paramount role in improving information security within the IoT ecosystem. However, a comprehensive examination of existing literature reveals that many AKA frameworks have been proposed for IoT environments, and many fall short of providing satisfactory physical security measures. In [9] the authors introduced a new AKA framework. Informally assessed against a range of security attacks, the proposed AKA framework demonstrates logical correctness by employing BAN logic. However, it is significant to mention that the AKA framework is weak against desynchronization attacks and demonstrates computational inefficiencies. In [10] the authors presented a secure AKA framework tailored for the IoT, specifically developed to enable remote access to sensing devices by legitimate or administrator users. The proposed AKA framework presented in [10] is susceptible to different attacks, including denial of service (DoS), sensor node capture, desynchronization, and replay attacks during the fourth phase. In [11] the author devised a two-factor user AKA framework for smart homes established on elliptic curve cryptography (ECC). The AKA framework experienced a formal security analysis using the random oracle model and was also validated through the Proverif tool. The AKA framework demonstrates resilience against various security threats, including replay, DoS, session key, forward secrecy, and password-guessing attacks. In [12] the authors proposed a secure AKA framework tailored for future IoT environments. The authors of [12] conducted both formal and informal security analyses, substantiating the protocol's robustness against noteworthy security attacks while functioning within constrained communication, computation, and storage resources. In [13] the authors devised a secure 3-factor AKA framework tailored for IoT-enabled smart homes, striving to give authorized users access to secure home services. Their AKA framework boasts resilience against diverse security threats, including impersonation and session key disclosure attacks. Despite its resilience, a crucial design flaw affected their AKA framework incapable of achieving mutual authentication.

In [14] the authors presented an AKA technique, which proffers perfect forward secrecy (PFS) and robust resistance against impersonation and modification attacks. However, it incurs a slightly elevated execution time and is susceptible to clock desynchronization and traceability attacks. In [15] the authors proposed an AKA technique, which eliminates the clock desynchronization issue and exhibits resistance against impersonation, man-in-the-middle (MITM), and replay attacks while guaranteeing PFS. Nonetheless, it stays powerless against modification and traceability attacks. In [16] the authors presented an authentication scheme, which handles clock desynchronization issues and presents resistance against impersonation, replay, password-guessing, and MITM attacks while guaranteeing PFS. However, it stays susceptible to modification attacks. In [17] the author presented an AKA mechanism for IoT devices and servers employing secure vaults. However, the designed AKA mechanism confronts challenges with keys of equal size, which can lead to offline password-guessing attacks or cloning attempts. In [18] the author suggested a PUF-based AKA scheme for the IoT, integrating ECC to lessen execution time and storage overhead during the AKA phase. In [19] the authors developed a PUF-based AKA scheme for IoT environments, leveraging wireless signal characteristics to secure devices from spoofing, cloning, tampering, and other attacks. In [20] the authors offered an AKA scheme for IoT devices, yet it lacks efficiency against offline password-guessing attacks. In [28] the authors delivered an AKA scheme for smart cities, facilitating AKA between IoT sensors and receivers employing ECC. In [21] the authors proposed a protected AKA scheme to establish a session key between IoT devices and receivers. Despite its efficiency, the technique involves complex mathematical operations that require robustness for high-speed wireless environments.

In [22], an AKA method for IoT-enabled WSNs established on temporal credentials is presented, striving for robustness. Despite its strengths, vulnerabilities are found in the AKA strategy introduced in [23], which utilizes ECC and a hash algorithm. These vulnerabilities incorporate defenselessness

to DoS, session key compromise, and impersonation attacks. In [24], an AKA mechanism is suggested, but it falls short of supplying a satisfactory user anonymity shield and lacks an efficient technique for password update. An anonymous AKA procedure employing chaotic maps and the hash algorithm is outlined in [25], with its security corroborated operating the BAN logic model. However, vulnerabilities are determined in an AKA procedure conceived for the cloud-enabled IoT environment employing ECC and a hash function, as provided in [26]. This technique involves four participants during the AKA phase and is corroborated for security employing BAN logic. Additionally, [27] presents an AKA procedure based on AEAD and a hash function, with security corroborated accomplished operating ROM and Scyther. Another AKA procedure relying on a hash function is examined in [28], which is encountered to be powerless to diverse security attacks as documented in [26]. The security of this technique is demonstrated by employing Scyther. Lastly, the security framework offered in [29], which operates ECC and hash algorithm, is susceptible to stolen smart card attacks. Similarly, the security framework in [30] does not effectively thwart DoS attacks. The scheme introduced in [31] is also discovered to be powerless against DoS attacks.

A. MOTIVATION AND RESEARCH CONTRIBUTION

In Section II, we reviewed various authentication schemes specifically designed for IoT networks. Many of these schemes are not suitable for the resource-constrained nature of IoT due to their high computational requirements, execution time, and communication costs. Additionally, most of them have been proven vulnerable to various security attacks and do not provide physical security. To address these challenges, this paper proposes a physically secure authentication framework for IoT, named PSAF-IoT. The main contributions of this paper are as follows:

- In this paper, we introduce an AKA framework leveraging hash functions, ECC, and PUF. The proposed PSAF-IoT allows only authorized users to access the IoT network and establish a session key, enabling encrypted communication between users/administrators and IoT devices deployed in the IoT-enabled environment. PSAF-IoT mitigates insider attacks by not storing the gateway's permanent secret key in its database. Instead, the PUF is utilized to derive the gateway's permanent secret key.
- Formal analysis is conducted using the Scyther tool, demonstrating PSAF-IoT's resilience against various attacks. Additionally, informal security analysis indicates that PSAF-IoT is resistant to MITM, DoS, replay, impersonation, and password-guessing attacks.
- The efficiency of the proposed PSAF-IoT is evaluated in terms of communication cost, energy consumption, and execution time as compared to [32], [33], and [34]. It is observed that PSAF-IoT requires [24.76% to 70.52%] lower communication cost,, and [35.5% to 78.66%] shorter execution time compared to related security schemes.

B. PAPER OUTLINE

The layout of the paper is outlined as follows. In Section III, we delve into the details of authentication and attack models, as well as provide background knowledge. The phases involved in developing PSAF-IoT are elaborated in Section IV. Security evaluation of PSAF-IoT is discussed in Section V. Efficiency and effectiveness comparisons between PSAF-IoT and other related schemes are presented in Section VI. Finally, we conclude the paper in Section VII by outlining the findings.

III. SYSTEM MODELS AND PRELIMINARIES

A. NETWORK MODEL

We utilize the network model for user authentication and session key generation, depicted in Figure 1. The model consists of an IoT devices (ITD_k), a gateway (GW_j), and a user (U_i).

In an IoT-enabled smart factory or similar environments, ITD_k collect information from the production plant and transmit this gathered data to either U_i or the GW_j. ITD_k communicate the collected information to GW_j using wireless communication channels, such as Wi-Fi.

A GW_j serves as a central element in an IoT environment, providing essential functionalities such as data aggregation, protocol translation, security enforcement, local processing, and connectivity management. GW_j also handles access control within IoT networks deployed in smart factories or other IoT-enabled environments. The GW_j keeps records of sensitive information associated with both users and IoT devices deployed within the environment. This sensitive information is employed to grant access only to specific users, ensuring reliable connectivity to the IoT network. The IoT GW_j connects to IoT networks using Wi-Fi and LoWPAN communication technologies, while it connects to users through cellular and WAN technologies.

In the IoT network, U_i acts as the operator or administrator of the smart factory. The administrator controls the smart factory production remotely, exchanging sensitive control commands with the deployed ITD_k to manage various functions of the production plant. To ensure reliable and accurate information exchange between ITD_k and U_i, a secure channel is essential. Therefore, this paper proposes an AKA framework to guarantee encrypted communication between the user and IoT devices. Table 1 lists the notations used in the paper to facilitate smooth reading and comprehension.

B. ATTACK MODEL

The proposed AKA framework utilized the well-established Dolev-Yao [35], [36] adversary model to illustrate the capabilities of potential adversaries within the system. In this scenario, communication between entities occurs over an insecure channel with untrustworthy end devices.

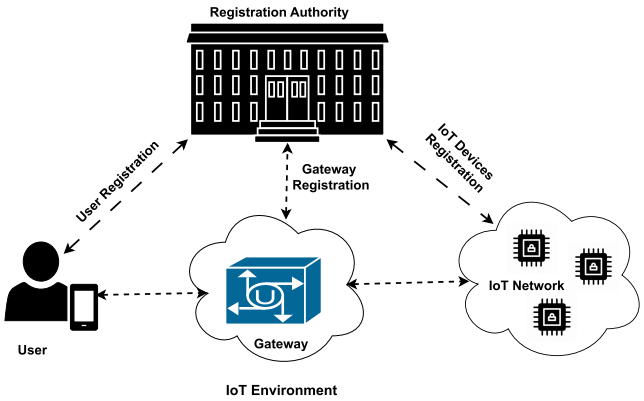

FIGURE 1. Network model for user authentication.

TABLE 1. Notations.

Notation	Description
U_i	User
RA	Registration authority
ITD_k	IoT device
PW_i	password of U_i
$\oplus$	XOR function
C	Challenge
RP	Response
$PUF()$	PUF function
$\mathcal{A}$	Adversary or attacker
UD_i	User device
GW_j	IoT gateway
ID_i	Identity ID_i
Bio_i	Biometric information of U_i
$\|$	Concatenation operation
$E(a,b)$	Epileptic curve
PK_g	Public key of the gateway
PK_k	Public key of the IoT device
P	Base or generation point
H	Helper data
$Gen(\cdot)$	FE based key generation algorithm
$Rep(\cdot)$	FE based key regeneration algorithm
$H(\cdot)$	Hash function

According to this model, the adversary is assumed to have complete control over the communication channel. Essentially, the adversary possesses the ability to intercept, modify, retransmit, fabricate, and delete any data transmitted over the insecure network. It is crucial to consider scenarios where a smart card is stolen or lost, as this could potentially lead to the leakage of stored secret credentials due to sophisticated power analysis attacks by adversaries.

Canetti and Krawczyk's adversary model (CK-adversary model) [37] offers a framework for modeling AKA protocols. This model considers not only the exposure of secrets but also the leakage of session keys, session states, and session ephemeral secrets. Even if certain confidential information, such as session-specific temporal data or session keys, is compromised by an adversary, the adversary should not gain any additional advantage to access other related credentials. Simultaneously, the compromise should minimally impact the secrecy of other sessions.

C. PRELIMINARIES

1) ELLIPTIC CURVE CRYPTOGRAPHY

Suppose that p is a large prime number. The description of an elliptic curve $E(a, b)$ over a finite field $\mathbb{F}_p$ is $E(a, b)$: $y^2 = x^3 + ax + b$, where $(a, b) \in \mathbb{F}_p$, $(x, y) \in \mathbb{F}_p$, $\mathbb{F}_p$ and $4a^3 + 27b^2 \neq 0$ are the "algebraic closure" of $\mathbb{F}_p$ [27], [38], [39]. P is the "generator" or "base point" of E. The two challenging issues in ECC are the "ECDLP" and the "ECDHP".

Definition 1: The "ECDLP" is the challenge for determining the integer x provided an elliptic curve $E(a, b)$ over a finite field $\mathbb{F}_p$, a point P on that curve, and another point $PK = xP$, where x is an integer chosen at random. Mathematically,

$$PK = xP \text{ where } k \in \mathbb{Z},$$

and the goal of the attacker is to find x given $E(a, b)$, P, and PK.

Definition 2: The "ECDHP" is the challenge of figuring out the secret that is shared S given an elliptic curve $E(a, b)$ over a finite field $\mathbb{F}_p$, a base point P on that curve, and public keys $PK_1 = aP$ and $PK_2 = bP$, where a and b are integers selected at random. The mathematical formula for the shared secret is

$$S = aPK_2 = bPK_1 \text{ where } a, b \in \mathbb{Z},$$

The objective of the attacker is to calculate S given $E(a, b)$, P, PK_1, and PK_2.

2) PHYSICAL UNCLONABLE FUNCTION

PUFs employ the intrinsic physical attributes of a device, such as manufacturing discrepancies like delay variations or impedance fluctuations [40]. PUFs serve as the "fingerprint" of hardware, with their responses containing inherent identity information. This quality makes them particularly valuable in the domain of IoT, particularly for tasks like secure key generation and identity verification, where robust security measures are essential [40]. Mathematically, we imply a PUF as $R = PUF(C)$, where C describes the challenge and R symbolizes the response. To preserve constant PUF output despite temperature fluctuations, a fuzzy extractor (FE) is utilized. This technique improves the dependability and applicability of PUFs across diverse strategies and applications.

3) FUZZY EXTRACTOR

The utilization of FE can effectively mitigate the impact of environmental noise on PUF responses. A FE comprises two integral functions: a probabilistic key generation function denoted as $Gen(\cdot)$, and a deterministic reconstruction function represented by $Rep(\cdot)$ [41], [42]. These functions are described below:

- $Gen(\cdot)$ function after acquiring the string R as an input and returns two outputs, a secret key $k \in \{0, 1\}^n$ and

reproduction data $h \in \{0,1\}^*$, which may be stated as follows: $Gen(R). = (k, h)$.
- If the Hamming distance between R and R' is insignificant, $Rep(\cdot)$ function will recover the secret key k using R' and the reproduction data $h \in \{0,1\}^*$ as inputs: $(k) = Rep(h, R')$.

IV. THE PROPOSED PSAF-IoT FRAMEWORK

The construction of the proposed PSAF-IoT is detailed in the following subsections.

A. GW_J REGISTRATION

RA selects the system parameters, such as elliptic curve $E(a, b)$, base point P and distributes these parameters in IoT environment. In addition, for the registration of GW_j selects the parameters such as C_1 and sends $\{E(a, b), P, C_1\}$ to GW_j securely. GW_j after getting $\{E(a, b), P, C_1\}$ selects a random number R_1 and computes $RP_1 = PUF(C_1)$ and acquires the stable key $(K_1, H_1) = Gen(RP_1)$, $Q_1 = H(K_1 \parallel R_1)$ and public key $PK_g = (Q_1 \cdot P)$. GW_j makes PK_g public and stores $\{C_1, H_1, R_1, PK_g\}$ in its own database.

B. ITD_K REGISTRATION

RA assigns a distinctive challenge C_2 and a private key K_2, sending $\{C_2, K_2\}$ to ITD_k via a secure communication channel. Additionally, ITD_k is equipped with a PUF. Upon receiving $\{C_2, K_2\}$, ITD_k calculates $RP_2 = PUF(C_2)$ and acquires the stable key $(K_3, H_2) = Gen(RP_2)$. Furthermore, ITD_k computes the public key $PK_k = (K_2 \cdot P)$, $Q_2 = H(PK_k)$, the identity of the IoT device $ID_k = (Q_2^a \oplus Q_2^b)$ where Q_2^a and Q_2^b are two equal chunks of Q_2, and $Q_3 = K_2 \oplus H(K_3||H_2)$. Finally, ITD_k stores $\{C_2, H_2, Q_3, PK_k, Q_{33}\}$ in its memory and securely sends PK_k to the RA.

C. USER DEVICE REGISTRATION

U_i acts as the remote user requiring access to devices within the IoT network. Its preliminary goals include acquiring real-time data from IoT devices and issuing control commands to them. To accomplish this, U_i must register with the RA before starting any communication with the IoT network.

U_i possesses a user device (UD_i) equipped with a biometric information (Bio_i) reader and an interface capable of accepting user identity (ID_i) and password (PW_i) inputs. Upon receiving these inputs, UD_i computes $(\sigma_i, H_3) = Gen(Bio_i)$, $Q_4 = H(\sigma_i \parallel ID_i)$, and $Q_5 = H(\sigma_i \parallel ID_i \parallel PW_i)$ and transmits the parameters $\{ID_i, Q_4, Q_5\}$ to the RA. RA after getting these parameters $\{ID_i, Q_4, Q_5\}$ computes $SID_i = H(ID_i \parallel R_1)$, $Q_6 = (Q_4 \cdot PK_g)$, $Q_7 = (Q_5 \oplus H(K_1 \parallel R_1))$, and $Q_8 = H(Q_6 \parallel Q_5)$. Furthermore, the RA assigns the public key PK_k of the IoT device, allowing U_i to access real-time information. The RA then sends the parameters $\{Q_8, PK_k, PK_g\}$. Upon receiving the parameters, UD_i calculates $Q_9 = H(Q_8 \parallel PK_k \parallel PK_g \parallel Q_5)$ and stores the parameters $\{Q_8, Q_9, PK_k, PK_g, H_3\}$ in its database.

D. AKA PHASE

U_i must first authenticate with GW_j before connecting to the IoT device within the IoT network. Once mutual authentication with GW_j is achieved, U_i establishes a session key with the IoT device in the network. This session key is then utilized for encrypting future communications between the IoT device and U_i. The following algorithms outline the process for establishing the session key.

Algorithm 1 initiates the execution of the AKA phase, taking the parameters $\{ID_i, PW_i, Bio_i, Q_8, Q_9, PK_k, PK_g, H_3\}$ as inputs. At UD_i, the biometric key is computed using the biometric information Bio_i and helper data. Furthermore, UD_i, using PW_i and ID_i as input parameters, calculates Q_4^*, Q_5^*, and Q_6^*, culminating in the computation of Q_8. To corroborate the authenticity of U_i, UD_i checks the condition $Q_8^* = Q_8$. Upon satisfying this condition, UD_i then verifies the integrity of the stored parameters in its memory through the condition $Q_9^* = Q_9$. If both conditions are met, UD_i completes local authentication and proceeds to generate the AKA message M_1.

After generating the random numbers R_2, R_3, and R_4, along with timestamps T_i, UD_i computes the public key Q_{10} and secret shares Q_{11} and Q_{12}. Additionally, UD_i calculates the parameters for message M_1, including Q_{14}, Q_{15}, and Q_{16}. Here, Q_{16} serves as the parameter to corroborate the integrity of message M_1 at the receiving end. Finally, UD_i constructs the message M_1 with the parameters $\{T_i, Q_{10}, Q_{14}, Q_{15}, Q_{16}\}$ and transmits M_1 to GW_j via a public communication channel.

Algorithm 2 begins execution by taking the parameters $\{C_1, H_1, R_1, PK_g, Q_7, T_i, Q_{10}, Q_{14}, Q_{15}, Q_{16}\}$ as inputs. Initially, GW_j verifies the freshness of the received M_1 and then computes the stable secret key using the PUF function and Upon computing the stable secret key, GW_j derives the secret share Q_{17}. Using this secret share, UD_i calculates $(R_3 \parallel (ID_i \oplus R_3))$ and SID_i. GW_j obtains Q_5 from Q_7. Finally, GW_j computes Q_{16} and verifies the condition $Q_{16}^* = Q_{16}$. If this condition is satisfied, GW_j trusts the authenticity of the received message M_1. After validating the authenticity of M_1, GW_j computes SID_k and verifies its presence in its database. If SID_k is found, GW_j generates T_j and calculates the parameter Q_{18}. This Q_{18} acts as the authenticity verification parameter. Finally, GW_j constructs the message M_2 with parameters $\{T_j, Q_{10}, Q_{15}, Q_{18}\}$ and sends M_2 to the specific IoT device for session key establishment.

By executing Algorithm 3 and taking the input parameters $\{T_j, Q_{10}, Q_{15}, Q_{18}, C_2, H_2, Q_3, Q_{33}, PK_k\}$, ITD_k verifies the freshness of the received message. Subsequently, it computes the private key using the PUF and Upon computing its own secret key, ITD_k calculates K_2 and the secret share Q_{12}^*. ITD_k obtains the parameters $(R_4 \parallel (ID_i \oplus R_3))$ and computes Q_{18}^*. Subsequently, ITD_k checks the condition $Q^*18 = Q18$. If this condition holds, ITD_k believes that M_2 is a valid message and proceeds with the AKA phase. Otherwise, it halts the AKA phase.

Algorithm 1 Generation of AKA Message M_1

Input: $\{ID_i, PW_i, Bio_i, Q_8, Q_9, PK_k, PK_g, H_3\}$
Output: $\{T_i, Q_{10}, Q_{14}, Q_{15}, Q_{16}\}$

procedure Algo-1($\{ID_i, PW_i, Bio_i, Q_8, Q_9, PK_k, PK_g, H_3\}$)
  $(\sigma_i^*) \leftarrow Rep(Bio_i, H_3)$,
  $Q_4^* \leftarrow H(\sigma_i^* \parallel ID_i)$
  $Q_5^* \leftarrow H(\sigma_i^* \parallel ID_i \parallel PW_i)$
  $Q_6^* \leftarrow (Q_4^* \cdot PK_g)$
  $Q_8^* \leftarrow H(Q_6^* \parallel Q_5^*)$
  if Q_8^*=Q_8 **then**
    $Q_9^* \leftarrow H(Q_8^* \parallel PK_k \parallel PK_g \parallel Q_5^*)$
    if Q_9^*=Q_9 **then**
      generates R_2, R_3, R_4, and T_i
      $Q_{10} \leftarrow H(R_2 \parallel Q_4^*) \cdot P$
      $Q_{11} \leftarrow Q_{10} \cdot PK_g$
      $Q_{12} \leftarrow Q_{10} \cdot PK_k$
      $Q_{13} \leftarrow H(PK_k)$
      $ID_k \leftarrow (Q_{13}^a \oplus Q_{13}^b)$
      $Q_{14} \leftarrow (ID_k \parallel (ID_i \oplus R_3)) \oplus H(Q_{11} \parallel T_i)$
      $Q_{15} \leftarrow (R_4 \parallel (ID_i \oplus R_3) \oplus ID_k) \oplus H(Q_{12} \parallel T_i)$
      $Q_{16} \leftarrow H(R_3 \parallel ID_i \parallel Q_{10} \parallel Q_5^* \parallel Q_{15} \parallel Q_{14})$
    else
      Ends AKA phase
    end if
  else
    Ends AKA phase
  end if
end procedure

Algorithm 2 Generation of AKA Message M_2

Input: $\{C_1, H_1, R_1, PK_g, Q_7, SID_k, T_i, Q_{10}, Q_{14}, Q_{15}, Q_{16}\}$
Output: $\{T_j, Q_{10}, Q_{15}, Q_{18}\}$

procedure Algo-2($\{C_1, H_1, R_1, PK_g, SID_k, T_i, Q_{10}, Q_{14}, Q_{15}, Q_{16}\}$)
  if $T_d \geq |T_r - T_i|$ **then**
    $RP_1 \leftarrow PUF(C_1)$
    $K_1 \leftarrow Gen(RP_1, H_1)$
    $Q_1 \leftarrow H(K_1 \parallel R_1)$
    $ID_g \leftarrow H(Q_1)$
    $Q_{17} \leftarrow Q_1 \cdot Q_{10}$
    $(ID_k \parallel (ID_i \oplus R_3)) \leftarrow (Q_{14}) \oplus H(Q_{17} \parallel T_i)$
    $SID_i \leftarrow H(ID_i \parallel R_1)$
    if SID_i found in database of GW_j **then**
      retrieves Q_5
      $Q_{16}^* \leftarrow H(R_3 \parallel ID_i \parallel Q_{10} \parallel Q_5 \parallel Q_{15} \parallel Q_{14})$
      if Q_{16}^*=Q_{16} **then**
        message M_1 is valid
        $SID_k \leftarrow H(ID_k)$
        if SID_k found in database of GW_j **then**
          selects T_j
          $Q_{18} \leftarrow H(Q_{10} \parallel T_j \parallel SID_k \parallel ID_g \parallel Q_{15} \parallel T_i)$
        else
          Ends AKA phase
        end if
      else
        Ends AKA phase
      end if
    else
      Ends AKA phase
    end if
  else
    Ends AKA phase
  end if
end procedure

ITD_K selects R_5 and T_k, computes Q_{19}, and derives the session key SK_k. This session key will be utilized for future communications in encrypted form. Additionally, ITD_k calculates Q_{20}, which serves as the authentication parameter for M_3. Finally, ITD_k constructs the message M_3 with the parameters $\{T_k, Q_{19}, Q_{20}\}$ and sends it to UD_i through a single channel of communication.

To complete the AKA phase, UD_i receives the message M_3 and checks its timeliness. It then computes SID_k and derives H($R_5 \parallel K_2^* \parallel ID_g$). Additionally, UD_i calculates the session key SK_i to facilitate communication in an encrypted form. Finally, UD_i computes Q_{20}^* and verifies the condition $Q_{20}^* = Q_{20}$ to ensure the integrity and authenticity of the message M_3. If this condition holds, UD_i considers the authentication successful and establishes the session key.

E. PASSWORD CHANGE MECHANISM

When designing an AKA framework, it is essential to incorporate mechanisms for password or biometric updates or changes. In the proposed framework, PSAF-IoT, a mechanism for password and biometric changes is provided, utilizing Algorithm 5.

V. SECURITY ANALYSIS OF PSAF-IoT

The security resilience of PSAF-IoT against various security vulnerabilities is assessed through both formal and informal security analyses.

A. INFORMAL SECURITY ANALYSIS

In this section, we validate the resilience of the proposed PSAF-IoT through informal, non-mathematical discussions.

1) PASSWORD GUESSING ATTACK

The proposed PSAF-IoT presents a robust defense against password-guessing attacks. Even if attackers manage to obtain credentials like $\{Q_8, Q_9, PK_k, PK_g, H_3\}$ from UD_i, launching a successful password-guessing attack proves challenging. The attacker is required to accomplish the conditions $Q_8^* = Q_8$ and $Q_9^* = Q_9$, which is a challenging

Algorithm 3 Generation of AKA Message M_3

Input: $\{T_j, Q_{10}, Q_{15}, Q_{18}, C_2, H_2, Q_3, Q_{33}, PK_k\}$
Output: $\{T_k, Q_{19}, Q_{20}\}$

procedure Algo-3($\{T_j, Q_{10}, Q_{15}, Q_{18}, C_2, H_2, Q_3\}$)
  if $T_d \geq |T_r - T_j|$ **then**
    $RP_2^* \leftarrow PUF(C_2)$
    $K_3^* \leftarrow Gen(RP_2^*, H_2)$
    $Q_2 \leftarrow H(PK_k)$
    $ID_k \leftarrow (Q_2^a \oplus Q_2^b)$
    $SID_k \leftarrow H(ID_k)$
    $K_2^* \leftarrow Q_3 \oplus H(K_3^* || H_2)$
    $ID_g \leftarrow Q_{33} \oplus H(H_2 || K_3^*)$
    $Q_{12}^* \leftarrow Q_{10} \cdot K_2^*$
    $(R_4 \parallel (ID_i \oplus R_3) \oplus ID_k) \leftarrow Q_{15} \oplus H(Q_{12}^* \parallel T_i \parallel ID_k)$
    $Q_{18}^* \leftarrow H(Q_{10} \parallel T_j \parallel SID_k \parallel ID_g \parallel Q_{15} \parallel T_i)$
    if Q_{18}^*=Q_{18} **then**
      message M_2 is valid
      selects R_5, T_k
      $Q_{19} \leftarrow H(R_5 \parallel K_2^* \parallel ID_g) \oplus H(Q_{12}^* \parallel (R_4 \parallel (ID_i \oplus R_3) \oplus ID_k))$
      $SK_k \leftarrow H(R_4 \parallel (ID_i \oplus R_3) \oplus ID_k) \parallel H(R_5 \parallel K_2^* \parallel ID_g))$
      $Q_{20} \leftarrow H(T_k \parallel SK_k \parallel H(R_5 \parallel K_2^* \parallel ID_g) \parallel SID_k)$
    else
      Ends AKA phase
    end if
  else
    Ends AKA phase
  end if
end procedure

task without comprehending the user's secret credentials, including ID_i, PW_i, Bio_i, and σ_i. As acquiring access to both the biometric information and the key shows considerable hurdles for attackers, given the difficulty in guessing and replicating biometric keys. Therefore, PSAF-IoT mitigates the chance of password-guessing attacks.

2) REPLAY ATTACK

In the PSAF-IoT framework, the AKA phase involves the exchange of three messages: M_1, M_2, and M_3. Each message is timestamped with the latest timestamps: T_i, T_j, and T_k, respectively, to safeguard against replay attacks. To guarantee the freshness of these messages, conditions are set for T_d to be greater than or equal to the absolute differences between the received timestamp and each message's timestamp. Specifically, for M_1, M_2, and M_3, the conditions are $T_d \geq |T_r - T_i|$, $T_d \geq |T_r - T_j|$, and $T_d \geq |T_r - T_k|$, respectively. If these conditions are met, the receiving participant considers the message as fresh. Otherwise, if the conditions are not met, the messages M_1, M_2, and M_3 are treated as delayed. This method thwarts replay attacks in the PSAF-IoT framework.

Algorithm 4 Authentication Successfully Completed

Input: $\{T_k, Q_{19}, Q_{20}\}$
Output: {AKA Phase Completed and Session Key is Established}

procedure Algo-4($\{T_k, Q_{19}, Q_{20}\}$)
  if $T_d \geq |T_r - T_k|$ **then**
    $SID_k \leftarrow H(ID_k)$
    $H(R_5 \parallel K_2^* \parallel ID_g) \leftarrow Q_{19} \oplus H(Q_{12}^* \parallel (R_4 \parallel (ID_i \oplus R_3) \oplus ID_k))$
    $SK_i \leftarrow H(R_4 \parallel (ID_i \oplus R_3) \oplus ID_k) \parallel H(R_5 \parallel K_2^* \parallel ID_g))$
    $Q_{20}^* \leftarrow H(T_k \parallel SK_k \parallel H(R_5 \parallel K_2^* \parallel ID_g) \parallel SID_k)$
    if Q_{20}^*=Q_{20} **then**
      message M_3 is valid
      session key established and verified
      authentication successfully completed
    else
      Ends AKA phase
    end if
  else
    Ends AKA phase
  end if
end procedure

3) PRIVILEGED INSIDER ATTACK

The susceptibility to a privileged insider attack materializes when GW_j stores permanent secret keys in plaintext within its database. In contrast, PSAF-IoT assumes a different method by abstaining from keeping such keys in the GW_j database. Instead, the secret key K_1 is generated through the PUF function, with only the challenge parameter C_1 maintained in the database. Additionally, the secret parameter Q_1 is derived using the secret key K_1 and R_1. While R_1 is stored in plaintext, the absence of plaintext for the permanent secret key K_1 complicates attackers' efforts to carry out any attacks, even if they gain access to parameters like $\{C_1, H_1, R_1, PK_g, SID_k\}$. The method adopted by PSAF-IoT effectively mitigates the risk of privileged insider attacks.

4) MITM ATTACK

To carry out a MITM attack in PSAF-IoT, the attacker needs to intercept all exchanged messages, especially M_1, M_2, and M_3, during the AKA phase. Following the DY model, which allows intercepted messages to be captured and altered, PSAF-IoT involves the exchange of three messages: M_1, M_2, and M_3. For the attacker to modify M_1, M_2, and M_3, they need access to specific secret parameters shared exclusively between UD_i, GW_j, and ITD_k. These parameters are $\{R_3, R_4, R_5, Q_{11}, ID_i\}$, $\{ID_g, ID_i\}$, and $\{R_5, SID_k, ID_i, Q_{12}\}$, respectively. Without these credentials, the attacker cannot alter the intercepted message. In PSAF-IoT, the integrity and authenticity of M_1, M_2, and M_3 are verified using the conditions $Q_{20}^* = Q_{20}$, $Q_{18}^* = Q_{18}$, and $Q_{16}^* = Q_{16}$ at UD_i,

Algorithm 5 Password Change Mechanism

Input: $\{ID_i, PW_i^o, Bio_i^o, Q_8^o, Q_9^o, PK_k, PK_g, H_3^o, Gen(\cdot), Rep(\cdot)\}$
Output: $\{Q_8^n, Q_9^n, PK_k, PK_g, H_3^n\}$
procedure Algo-p($\{ID_i, PW_i^o, Bio_i^o, Q_8^o, Q_9^o, PK_k, PK_g, H_3^o, PW_i^n, Bio_i^n\}$)
  $(\sigma_i^o) \leftarrow Rep(Bio_i^o, H_3^o)$,
  $Q_4^o \leftarrow H(\sigma_i^o \parallel ID_i)$
  $Q_5^o \leftarrow H(\sigma_i^o \parallel ID_i \parallel PW_i^o)$
  $Q_6^o \leftarrow (Q_4^o \cdot PK_g)$
  $Q_8^o \leftarrow H(Q_6^o \parallel Q_5^o)$
  if Q_8^o=Q_8 **then**
    $Q_9^o \leftarrow H(Q_8^o \parallel PK_k \parallel PK_g \parallel Q_5^o)$
    if Q_9^o=Q_9 **then**
      enter PW_i^n, Bio_i^n
      $(\sigma_i^n, H_3^n) \leftarrow Gen(Bio_i^n)$,
      $Q_4^n \leftarrow H(\sigma_i^n \parallel ID_i)$
      $Q_5^o \leftarrow H(\sigma_i^n \parallel ID_i \parallel PW_i^o)$
      $Q_6^n \leftarrow (Q_4^n \cdot PK_g)$
      $Q_8^n \leftarrow H(Q_6^n \parallel Q_5^n)$
      $Q_9^n \leftarrow H(Q_8^n \parallel PK_k \parallel PK_g \parallel Q_5^n)$
    else
      Ends password change mechanism
    end if
  else
    Ends password change mechanism
  end if
end procedure

ITD_k, and GW_j, respectively. If any of these conditions are not met, the corresponding network entity will reject the message. This mechanism prevents modified messages from being accepted by the receiving entity, making PSAF-IoT resistant to MITM attacks.

5) DoS ATTACKS

The PSAF-IoT protects against DoS attacks. UD_i generates the AKA message M_1 for GW_j only after validating the conditions $Q_8^* = Q_8$ and $Q_9^* = Q_9$. If these conditions are met, UD_i generates M_1. Otherwise, UD_i refrains from sending AKA messages to GW_j. This approach ensures that UD_i cannot flood GW_j with excessive AKA messages, thereby preventing potential DoS attacks in the proposed PSAF-IoT system.

6) TEMPORARY SECRET COMPROMISED ATTACK

The session key in PSAF-IoT protocol is implied as $SK_k (= SK_i) = H(R_4 \parallel (ID_i \oplus R_3) \oplus ID_k) \parallel H(R_5 \parallel K_2^* \parallel ID_g))$. This key combines various parameters, both direct and indirect, encompassing both permanent and temporary values. To compromise the security of the session key established during the AKA phase in PSAF-IoT, an attacker would need knowledge of both the permanent and temporary parameters. Thus, having knowledge of only one type of parameter, either permanent or temporary, is insufficient for an adversary to compromise the session key's security.

7) IMPERSONATION ATTACK

There are two potential types of impersonation attacks that could target the PSAF-IoT framework. In one scenario, the attacker attempts to impersonate a legitimate user in the IoT environment by crafting a message, M_1. However, generating M_1 requires knowledge of specific parameters like $\{R_3, R_4, R_5, Q_{11}, ID_i\}$. Without access to these credentials, constructing a valid M_1 becomes challenging for the attacker. Additionally, the authenticity of M_1 is verified at GW_j through conditions like $Q_{16}^* = Q_{16}$. Likewise, generating M_2 is challenging without knowing the parameters used in generating M_1 and $\{ID_g, ID_i\}$. The authenticity of M_2 is confirmed at ITD_k using conditions such as $Q_{18}^* = Q_{18}$. Similarly, creating M_3 without knowledge of $\{R_5, SID_k, ID_i, Q_{12}\}$ is also difficult. The authenticity of M_3 is validated at UD_i through conditions like $Q_{20}^* = Q_{20}$. Therefore, meeting these conditions at the UD_i, GW_j, and ITD_k ends is challenging without permanent and temporary credentials. As a result, the proposed PSAF-IoT protocol offers robust protection against impersonation attacks.

B. ROM-BASED SECURITY VALIDATION

Through ROM, PSAF-IoT is thoroughly examined, and $\mathcal{A}$ is given permission to construct a variety of queries that allow for the execution of legitimate attacks. Various component of the ROM model are described as follows.

1) PARTICIPANTS

Within the PSAF-IoT framework, three key entities/ participants are involved: UD_i, GW_j, and ITD_k. The instances I_1, I_2,and I_3 representing UD_i, GW_j, and ITD_k are denoted as $\pi_{UD_i}^{I_1}$, $\pi_{GW_j}^{I_2}$, and $\pi_{ITD_k}^{I_3}$, respectively, functioning as oracles.

2) PARTNERSHIP

If instances $\pi_{UD_i}^{I_1}$ and $\pi_{GW_j}^{I_2}$ have a common session key, they establish a partnership at the acceptance state.

3) FRESHNESS

By $\mathcal{A}$, the SK generated during the AKA phase between $\pi_{UD_i}^{I_1}$ and $\pi_{GW_j}^{I_2}$ cannot be revealed or made public.

Table 2 contains a list of these queries. We evaluate every potential query in order to formally corroborate the security of PSAF-IoT. The subsequent variety of queries are used to simulate various attack scenarios against PSAF-IoT.

Theorem 1: Let $\mathcal{A}$ be an adversary bounded by polynomial time (plt), challenging the security of PSAF-IoT. We denote hash queries as H^2q, send queries as Q_{se}, and PUF queries as H_{puf}^2. The password dictionary space is represented by PSD, and $|PUF|$ indicates the length of the PUF query. $|HSL|$ indicates the length of the PUF query The length of the biometric key is denoted by 2^{le}. Additionally, $Adv^{ECDLP}\mathcal{A}(plt)$

TABLE 2. ROM based queries.

Query	Explanation of the Query
$Execute(\pi_{UD_i}^{I_1}, \pi_{GW_j}^{I_2}, \pi_{ITD_K}^{I_2})$	This query represents an eavesdropping attack, allowing the interception of messages between two honest participants.
$Reveal(\pi^{I_1})$	This query exposes the current session key generated by $\mathcal{A}$ and its partner to an adversary.
$Send(\pi^{I_1}, msg)$	In this query, $\mathcal{A}$ sends a message, msg, to an instance and receives a response message, modeling an active attack.
$CorruptUD(\pi^{I_1})$	This query simulates a lost or stolen UD_i attack, where $\mathcal{A}$ can extract all information stored in UD_i after executing this query.
$Test(\pi^{I_1})$	Under this query, an impartial coin, "b", is flipped by $\mathcal{A}$. Based on the result, $\mathcal{A}$ establishes the session key between itself and another entity. The coin flip results in either 0 or 1. If $b = 1$, $\mathcal{A}$ returns the session key, SK; if $b = 0$, $\mathcal{A}$ returns a random binary string; otherwise, it returns a null value.

signifies the advantage of $\mathcal{A}$ in compromising and solving the ECDLP in plt. The estimation of $\mathcal{A}$'s advantage in compromising the security of the secret key (SK) can be expressed as follows.

$$Adv_{\mathcal{A}}^{PSAF-IoT}(plt) \leq \frac{H_q^2}{|HSL|} + \frac{H_{puf}^2}{|PUF|} + \frac{S_q}{2^{le-1} \cdot |PSD|} + 2 \cdot Adv_{\mathcal{A}}^{ECDLP}(plt). \quad (1)$$

Proof: We prove Theorem 1 by examining the trailing five games ($Game_k \mid k = 0, 1, 2, 3, 4$) [43]. The adversary $\mathcal{A}$'s advantage in compromising the security of the secret session key is represented as $Adv_{\mathcal{A}}^{PSAF-IoT}(plt) = \mid 2 \cdot Adv^{Game} - 1 \mid$. Here, Adv^{Game} denotes $\mathcal{A}$'s probability of winning by accurately predicting the bit "b" in each $Game_k$.

$Game_0$: In this scenario, $\mathcal{A}$ executes a real attack on PSAF-IoT. By the definition of success, we have achieved the desired outcome.

$$Adv_{\mathcal{A}}^{PSAF-IoT}(plt) = \mid 2 \cdot Adv^{Game_0} - 1 \mid . \quad (2)$$

$Game_1$: the security of the cryptographic scheme against eavesdropping attacks is robust. In this scenario, $\mathcal{A}$ attempts to intercept M_1, M_2, and M_3 using the query $Execute(\pi_{UD_i}^{I_1}, \pi_{GW_j}^{I_2}, \pi_{ITD_K}^{I_2})$ and derive the session key $SK_k (= SK_i) = H(R_4 \parallel (ID_i \oplus R_3) \oplus ID_k) \parallel H(R_5 \parallel K_2^* \parallel ID_g))$ using intercepted M_1, M_2, and M_3. However, due to the combination of permanent and temporary parameters in the session key generation process, $\mathcal{A}$ needs both these parameters to successfully compute a valid session key. In the concluding phase of $Game_1$, $\mathcal{A}$ attempts to reveal the presumed secret key through the operation $Reveal(\pi^{g_1})$ and evaluates its correctness with $Test(\pi^{g_1})$ by comparing it to a random bit. However, lacking both the permanent and temporary parameters, $\mathcal{A}$ is incapable of producing a legitimate session key. As a result, the likelihood of $\mathcal{A}$ succeeding in this endeavor is minimal. Therefore, we can conclude that the cryptographic scheme provides strong security against eavesdropping attacks, as an adversary cannot derive the session key without having both the permanent and temporary parameters. This leads to $Game_0$ and $Game_1$ becoming indistinguishable. From this, we can deduce that.

$$Adv^{Game_1} = Adv^{Game_0} \quad (3)$$

$Game_2$: In this scenario, $\mathcal{A}$ initiates an active attack using hash (HSL) queries. Within the context of PSAF-IoT, the session key is generated using the SHA algorithm on the sides of both UD_i. $\mathcal{A}$ tries to identify a collision through HSL queries with the goal of compromising the security of the session key, SK. Additionally, all messages exchanged during the AKA phase are safeguarded by the hash function. Therefore, to compromise the security of the proposed PSAF-IoT, an attacker would attempt to find a collision in the hash function's output.However, the likelihood of successfully identifying a collision is extremely low. Additionally, $\mathcal{A}$ utilizes PUF queries in this attack. To find the collision in the PUF is highly challenging, if not impossible. As a result, $\mathcal{A}$'s advantage does not increase in $Game_2$. Consequently, we conclude that.

$$Adv^{Game_2} - Adv^{Game_1} \leq \frac{H_q^2}{2|HSL|} + \frac{H_{puf}^2}{2|PUF|}. \quad (4)$$

$Game_3$: During the game, $\mathcal{A}$ executed the $CorruptUD(\pi^{I_1})$ query to start an active attack. After UD_i is successfully taken over, the attacker will be able to obtain the set of credentials $\{Q_8, Q_9, PK_k, PK_g, H_3, Gen(.), Rep(.)\}$ that are kept in UD_i's memory. $\mathcal{A}$ seeks to determine the password, biometric key, and true identity of the user. From the obtained parameter through the power analysis, it is hard for the attacker to get the any sensitive information. The likelihood of accurately estimating the biometric key is $\frac{1}{2^{le}}$, which is almost negligible. In addition, the system limits the quantity of failed password attempts that may be made. In view of these facts, the following deductions can be made:

$$Adv^{Game_3} - Adv^{Game_2} \leq \frac{Q_{se}}{2^{le} \cdot \mid PSD \mid}. \quad (5)$$

$Game_4$: In this gaming scenario, $\mathcal{A}$ executes $Execute(\pi_{UD_i}^{I_1}, \pi_{GW_j}^{I_2}, \pi_{ITD_K}^{I_2})$ to capture M_1, M_2, and M_3. After acquiring all messages, $\mathcal{A}$ endeavors to disclose the undercover data that was encrypted and communicated among all entities in the IoT network. It's worth mentioning that both M_1 and M_2 contain the parameter Q_{10}, which is the public key of UD_i. To access the sensitive credentials utilized in generating the session key, compromising the security of ECC is essential. To achieve this, $\mathcal{A}$ needs to solve the ECDLP in polynomial time. Definitions 1 clarify that these objectives are achievable.

$$Adv^{Game_4} - Adv^{Game_3} \leq Adv_{\mathcal{A}}^{ECDLP}(plt). \quad (6)$$

Upon concluding all games ($Game_k \mid k \in [0, 3]$), $\mathcal{A}$ does not accumulate a substantial advantage in accurately forecasting the bit "b". Therefore, we extrapolate that

$$Adv^{Game_4} = 1/2 \quad (7)$$

From (2) and (3), we get

$$Adv_{\mathcal{A}}^{PSAF-IoT}(plt) = \mid 2 \cdot Adv^{Game_0} - \frac{1}{2} \mid . \quad (8)$$

From (8), we get

$$\frac{1}{2}.Adv_{\mathcal{A}}^{PSAF-IoT}(plt) = \mid Adv^{Game_0} - Adv^{Game_4} \mid . \quad (9)$$

By using (7) and (9), we obtain

$$\frac{1}{2}.Adv_{\mathcal{A}}^{PSAF-IoT}(plt) = \mid Adv^{Game_1} - Adv^{Game_4} \mid \quad (10)$$

Upon considering the triangular inequality, we have

$$\begin{aligned} &\mid Adv^{Game_1} - Adv^{Game_4} \mid \\ &\leq \mid Adv^{Game_1} - Adv^{Game_2} \mid + \mid Adv^{Game_2} - Adv^{Game_4} \mid \\ &\leq \mid Adv^{Game_1} - Adv^{Game_2} \mid + \mid Adv^{Game_2} - Adv^{Game_3} \mid \\ &\quad + \mid Adv^{Game_3} - Adv^{Game_4} \mid . \end{aligned} \quad (11)$$

By using (4), (6), and (11), we get

$$\begin{aligned} Adv_{\mathcal{A}}^{PSAF-IoT}(plt) \leq \frac{H_q^2}{\mid Hh \mid} + \frac{H_{puf}^2}{\mid PUF \mid} + \frac{Q_{se}}{2^{le-1} \cdot \mid PSD \mid} \\ + 2.Adv_{\mathcal{A}}^{ECDLP}(plt). \end{aligned} \quad (12)$$

□

C. SECURITY VALIDATION USING SCYTHER

The Scyther tool [44], [45] functions as an automated and highly efficacious formal verification tool particularly developed to assess protocol security against identity attacks. Functioning on the basis of the DY model [46], it investigates protocol security through the utilization of security claims. Additionally, Scyther ascertains all security claim categories within the protocol, visually depicting any breaches per claim. Utilizing the security protocol description language (SPDL), Scyther expresses the protocol model under scrutiny. Furthermore, it presents a user-friendly graphical interface. Once security claims, roles, and protocols are specified, security validation commences with the execution of authentication commands. This approach assumes a black-box cryptographic model. Protocols are modeled based on role definition, permitting Scyther to investigate both correctness and authenticity. Therefore, we modeled the proposed PSAF-IoT according to SPDL to corroborate protocol claims.

In implementing PSAF-IoT, SPDL serves as the framework. This SPDL script defines the roles of three pivotal entities: UD_i (user), GW_j (Gateway node), and ITD_k (IoT device). Each role within the SPDL script is associated with specific claims. Testing with Scyther confirms that there have been no identified attacks targeting the session key disclosure, aliveness, non-injective agreement, weak agreement, and non-synchronization facets of the proposed PSAF-Io. Figure 2 shows the PSAF-IoT is secure as no attack is found against the PSAF-IoT.

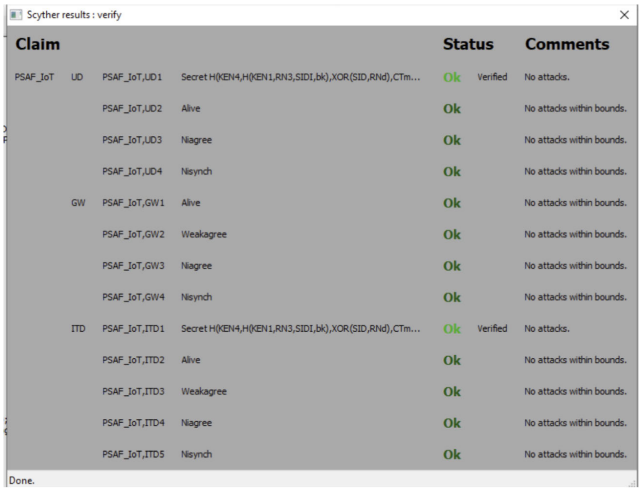

FIGURE 2. Secrity analysis using the Scyther tool.

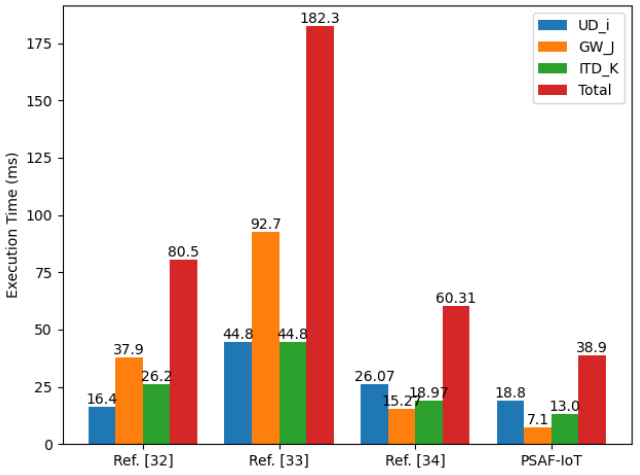

FIGURE 3. Execution time to accomplish the AKA phase.

VI. PERFORMANCE ANALYSIS

We assess the efficiency of PSAF-IoT: in comparison with [32], [33], and [34], focusing on their security features, communication costs, and execution time. We employ a Raspberry Pi 3 micro-controller as GW_j and (UD_i) with the specification, CPU: quad-core 1.2 GHz, 1GB RAM, and Ubuntu OS. Computational time for various cryptographic operations is represented as follows: T_{hs} for hash function, T_{epm} for ECC multiplication, T_{epa} for ECC addition, T_{bp} for Bi-linear pairing, T_{fe} for fuzzy extractor, and T_{pf} for PUF. We conduct cryptographic operations using "Pycrypto". Each cryptographic primitive experiment is repeated 300 times. Time required by $T_{hs} \approx 0.6$ ms, $T_{epm} \approx 2.9$ ms, $T_{epa} \approx 0.17$ ms, $T_{bp} \approx 7.6$ ms, $T_{fe} \approx 2.9$, ms and $T_{pf} \approx 0.00054$ ms.

A. EVALUATION OF EXECUTION TIME

In this section, we evaluate the execution time of the proposed PSAF-IoT during the AKA phase. It turns out that PSAF-IoT

TABLE 3. AKA phase execution time comparison.

Framework	User/UD_i	Gateway GW_j	IoT device ITD_k	Aggregated Execution Time
Ref. [32]	$8T_{hs} + 4T_{epm} \approx 16.4$ ms	$10T_{hs} + 11T_{epm} \approx 37.9$ ms	$5T_{hs} + 8T_{epm} \approx 26.2$ ms	$23T_{hs} + 23T_{epm} \approx 80.5$ ms
Ref. [33]	$7T_{hs} + 14T_{epm} \approx 44.8$ ms	$12T_{hs} + 19T_{epm} + 4T_{bp} \approx 92.6$ ms	$7T_{hs} + 14T_{epm} \approx 44.8$ ms	$26T_{hs} + 47T_{epm} + 4T_{bp} \approx 182.3$ ms
Ref. [34]	$19T_h + 4T_{epm} + T_{epa} + T_{fe} \approx 26.0$ ms	$T_h + 5T_{epm} + T_{epa} \approx 15.27$ ms	$12T_h + 4T_{epm} + T_{epa} \approx 18.97$ ms	$32T_h + 13T_{epm} + 3T_{epa} + T_{fe} \approx 60.31$ ms
PSAF-IoT	$12T_{hs} + 3T_{epm} + T_{fe} \approx 18.79$ ms	$7T_{hs} + T_{fe} + T_{pf} \approx 7.1$ ms	$12T_{hs} + T_{fe} + 1T_{epm} + T_{pf} \approx 13.0$ ms	$31T_{hs} + 4T_{epm} + 3T_{fe} + 2T_{pf} \approx 38.9$ ms

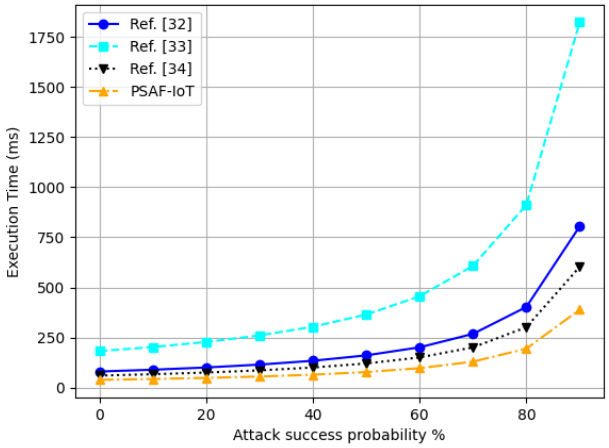

FIGURE 4. Execution time of AKA phase of PSAF-IoT under jamming and eavesdropping attack.

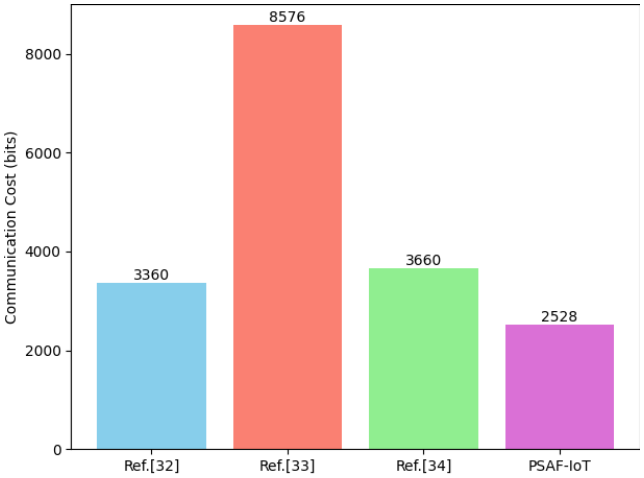

FIGURE 6. Communication cost require to complete the AKA phase.

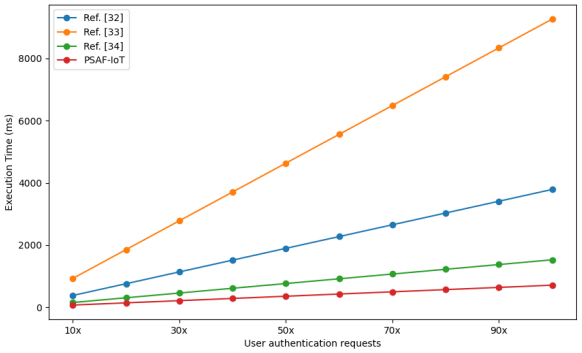

FIGURE 5. Execution time required at GW_j with increasing user AKA requests.

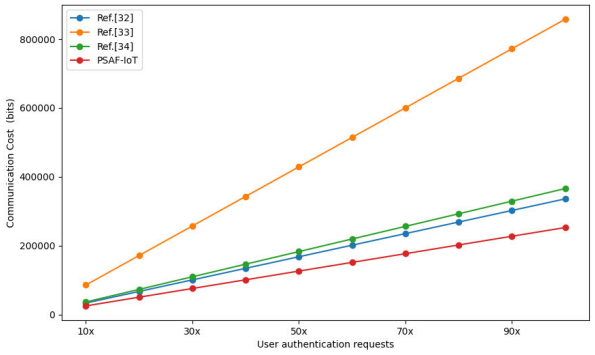

FIGURE 7. Communication with increasing the number of AKA requests.

requires an execution time of 23 ms to complete the AKA phase, which is less than that of related AKA frameworks. This comparison is provided in Table 3 and Figure 3. Additionally, it is imperative to assess the execution time requirement of GW_j when multiple users send concurrent AKA requests to GW_j, as GW_j is responsible for validating users in the IoT environment. This comparison is presented in Figure 5. The proposed PSAF-IoT requires 51.68%, 78.66%, and 35.5% lower execution time than [32], [33], and [34], respectively. When the adversary interrupts the execution of PSAF-IoT or drops the messages communicated to complete the AKA phase, the proposed PSAF-IoT demonstrates lower execution time under jamming and eavesdropping attacks compared to related security schemes. This is depicted in Figure 4.

B. EVALUATION OF COMMUNICATION COST

In this section, we investigate the communication cost associated with the proposed PSAF-IoT during the AKA phase. It is found that PSAF-IoT requires a communication cost of 2528 bits to conduct the AKA phase, which is lower than that of comparable AKA frameworks. This comparison is detailed in Table 4 and Figure 6. Furthermore, it is crucial to evaluate the bandwidth requirements, when multiple users commence simultaneous AKA requests to different ITD_k deployed in an IoT environment. This assessment is illustrated in Figure 7. The proposed PSAF-IoT requires 24.76%, 70.52%, and 30.93% lower communication cost than [32], [33], and [34], respectively. In scenarios where an adversary disrupts the execution of PSAF-IoT or intercepts messages intended for conducting the AKA phase, the proposed PSAF-IoT exhibits reduced communication costs

TABLE 4. AKA phase communication cost comparison.

Framework	No. Messages Exchange	Total Communication Cost(bits)
Ref. [32]	4	3360
Ref. [33]	8	8576
Ref. [34]	3	3660
PSAF-IoT	3	2528

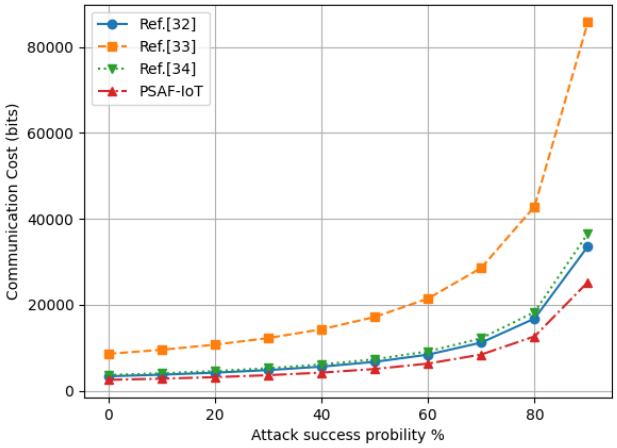

FIGURE 8. Communication cost under the jamming and eavesdropping attacks.

under jamming and eavesdropping attacks compared to other security schemes. This is shown in Figure 8.

VII. CONCLUSION

Ensuring information security within the IoT infrastructure poses significant challenges. In response to this issue, we introduce a new AKA framework named PSAF-IoT. This framework enables users to establish secure communication channels with specific devices deployed in the IoT environment. Leveraging PUF, PSAF-IoT achieves physical security and ensures resistance against privileged insider and node capture. The security of the secure channel establishment process is validated through formal methods such as the random oracle model and Scyther-based implementation. Comparative analysis revealed that PSAF-IoT demonstrates efficiency in terms of execution time, and communication costs, requiring [35.5% to 78.66%] lower execution time, and [24.76% to 70.52%] lower communication costs compared to alternatives. These findings underscore the significance of adopting PSAF-IoT to safeguard user communication in increasingly interconnected IoT-enabled environments.

REFERENCES

[1] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A survey on IoT security: Application areas, security threats, and solution architectures," *IEEE Access*, vol. 7, pp. 82721–82743, 2019.

[2] N. Neshenko, E. Bou-Harb, J. Crichigno, G. Kaddoum, and N. Ghani, "Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on internet-scale IoT exploitations," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2702–2733, 3rd Quart., 2019.

[3] M. Tanveer, A. Aldosary, A. K. Das, S. A. Aldossari, and S. A. Chaudhry, "PAF-IoD: PUF-enabled authentication framework for the Internet of Drones," *IEEE Trans. Veh. Technol.*, to be published.

[4] S. Subashini, G. Kamalam, and P. Vanitha, "A survey of IoT in healthcare: Technologies, applications, and challenges," in *Proc. Artif. Intell. Mach. Learn.*, 2024, pp. 136–144.

[5] A. Schlemitz and V. Mezhuyev, "Approaches for data collection and process standardization in smart manufacturing: Systematic literature review," *J. Ind. Inf. Integr.*, vol. 38, Mar. 2024, Art. no. 100578.

[6] Y. Hu, Q. Jia, Y. Yao, Y. Lee, M. Lee, C. Wang, X. Zhou, R. Xie, and F. R. Yu, "Industrial Internet of Things intelligence empowering smart manufacturing: A literature review," *IEEE Internet Things J.*, vol. 11, no. 11, pp. 19143–19167, Jun. 2024.

[7] M. Soori, B. Arezoo, and R. Dastres, "Internet of Things for smart factories in industry 4.0, a review," *Internet Things Cyber-Physical Syst.*, vol. 3, pp. 192–204, 2023.

[8] C.-C. Lin, C.-T. Tsai, Y.-L. Liu, T.-T. Chang, and Y.-S. Chang, "Security and privacy in 5G-IIoT smart factories: Novel approaches, trends, and challenges," *Mobile Netw. Appl.*, vol. 28, pp. 1043–1058, Jul. 2023.

[9] V. Sureshkumar, R. Amin, M. S. Obaidat, and I. Karthikeyan, "An enhanced mutual authentication and key establishment protocol for tmis using chaotic map," *J. Inf. Secur. Appl.*, vol. 53, Aug. 2020, Art. no. 102539.

[10] R. Vinoth, L. J. Deborah, P. Vijayakumar, and N. Kumar, "Secure multifactor authenticated key agreement scheme for industrial IoT," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3801–3811, Mar. 2021.

[11] D. Kaur and D. Kumar, "Cryptanalysis and improvement of a two-factor user authentication scheme for smart home," *J. Inf. Secur. Appl.*, vol. 58, May 2021, Art. no. 102787.

[12] M. Rana, A. Shafiq, I. Altaf, M. Alazab, K. Mahmood, S. A. Chaudhry, and Y. B. Zikria, "A secure and lightweight authentication scheme for next generation IoT infrastructure," *Comput. Commun.*, vol. 165, pp. 85–96, Jan. 2021.

[13] S. Yu, N. Jho, and Y. Park, "Lightweight three-factor-based privacy-preserving authentication scheme for IoT-enabled smart homes," *IEEE Access*, vol. 9, pp. 126186–126197, 2021.

[14] S. Rostampour, M. Safkhani, Y. Bendavid, and N. Bagheri, "ECCbAP: A secure ECC based authentication protocol for IoT edge devices," *Pervas. Mobile Comput.*, vol. 67, Jun. 2020, Art. no. 101194.

[15] X. Jia, D. He, N. Kumar, and K. R. Choo, "A provably secure and efficient identity-based anonymous authentication scheme for mobile edge computing," *IEEE Syst. J.*, vol. 14, no. 1, pp. 560–571, Mar. 2020.

[16] P. K. Panda and S. Chattopadhyay, "A secure mutual authentication protocol for IoT environment," *J. Rel. Intell. Environ.*, vol. 6, pp. 79–94, Jan. 2020.

[17] T. Shah and S. Venkatesan, "Authentication of IoT device and IoT server using secure vaults," in *Proc. 17th IEEE Int. Conf. Trust, Secur. Privacy Comput. Communications/12th IEEE Int. Conf. Big Data Sci. Eng. (TrustCom/BigDataSE)*, Aug. 2018, pp. 819–824.

[18] J. R. Wallrabenstein, "Practical and secure IoT device authentication using physical unclonable functions," in *Proc. IEEE 4th Int. Conf. Future Internet Things Cloud (FiCloud)*, Aug. 2016, pp. 99–106.

[19] M. N. Aman, M. H. Basheer, and B. Sikdar, "Two-factor authentication for IoT with location information," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3335–3351, Apr. 2019.

[20] P. Gope and B. Sikdar, "Lightweight and privacy-preserving two-factor authentication scheme for IoT devices," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 580–589, Feb. 2019.

[21] R. Sharma and R. Arya, "A secure authentication technique for connecting different IoT devices in the smart city infrastructure," *Cluster Comput.*, vol. 25, no. 4, pp. 2333–2349, 2022.

[22] C.-T. Chen, C.-C. Lee, and I.-C. Lin, "Efficient and secure three-party mutual authentication key agreement protocol for WSNs in IoT environments," *PLoS ONE*, vol. 15, no. 4, 2020, Art. no. e0232277.

[23] D. Kumar, H. K. Singh, and C. Ahlawat, "A secure three-factor authentication scheme for wireless sensor networks using ECC," *J. Discr. Math. Sci. Cryptography*, vol. 23, no. 4, pp. 879–900, 2020.

[24] A. Ostad-Sharif, H. Arshad, M. Nikooghadam, and D. Abbasinezhad-Mood, "Three party secure data transmission in IoT networks through design of a lightweight authenticated key agreement scheme," *Future Gener. Comput. Syst.*, vol. 100, pp. 882–892, Nov. 2019.

[25] H. Qiao, X. Dong, Q. Jiang, S. Ma, C. Liu, N. Xi, and Y. Shen, "Anonymous lightweight authenticated key agreement protocol for fog-assisted healthcare IoT system," *IEEE Internet Things J.*, vol. 10, no. 19, pp. 16715–16726, Oct. 2023.

[26] C. Wang, D. Wang, Y. Duan, and X. Tao, "Secure and lightweight user authentication scheme for cloud-assisted Internet of Things," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 2961–2976, 2023.
[27] M. Tanveer, A. U. Khan, N. Kumar, and M. M. Hassan, "RAMP-IoD: A robust authenticated key management protocol for the Internet of Drones," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 1339–1353, Jan. 2022.
[28] 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, Mar. 2020.
[29] Y. Choi, D. Lee, J. Kim, J. Jung, J. Nam, and D. Won, "Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors*, vol. 14, no. 6, pp. 10081–10106, 2014.
[30] 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.
[31] S. Challa, M. Wazid, A. Kumar Das, N. Kumar, A. Goutham Reddy, E.-J. Yoon, and K.-Y. Yoo, "Secure signature-based authenticated key establishment scheme for future IoT applications," *IEEE Access*, vol. 5, pp. 3028–3043, 2017.
[32] P. Porambage, A. Braeken, C. Schmitt, A. Gurtov, M. Ylianttila, and B. Stiller, "Group key establishment for enabling secure multicast communication in wireless sensor networks deployed for IoT applications," *IEEE Access*, vol. 3, pp. 1503–1511, 2015.
[33] B. Yu and H. Li, "Anonymous authentication key agreement scheme with pairing-based cryptography for home-based multi-sensor Internet of Things," *Int. J. Distrib. Sensor Netw.*, vol. 15, no. 9, 2019, Art. no. 1550147719879379.
[34] M. Wazid, A. K. Das, N. Kumar, and M. Alazab, "Designing authenticated key management scheme in 6G-enabled network in a box deployed for industrial applications," *IEEE Trans. Ind. Informat.*, vol. 17, no. 10, pp. 7174–7184, Oct. 2021.
[35] G. Abbas, M. Tanveer, Z. H. Abbas, M. Waqas, T. Baker, and D. Al-Jumeily OBE, "A secure remote user authentication scheme for 6LoWPAN-based Internet of Things," *PLoS ONE*, vol. 16, no. 11, 2021, Art. no. e0258279.
[36] D. Dolev and A. C. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.
[37] M. Tanveer, A. Badshah, H. Alasmary, and S. A. Chaudhry, "CMAF-IIoT: Chaotic map-based authentication framework for industrial Internet of Things," *Internet Things*, vol. 23, Oct. 2023, Art. no. 100902.
[38] X. Li, J. Niu, M. Z. A. Bhuiyan, F. Wu, M. Karuppiah, and S. Kumari, "A robust ECC-based provable secure authentication protocol with privacy preserving for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3599–3609, Aug. 2018.
[39] W. Huang, "ECC-based three-factor authentication and key agreement scheme for wireless sensor networks," *Sci. Rep.*, vol. 14, no. 1, p. 1787, 2024.
[40] S. Yu and K. Park, "PUF-based robust and anonymous authentication and key establishment scheme for V2G networks," *IEEE Internet Things J.*, vol. 11, no. 9, pp. 15450–15464, May 2024.
[41] M. Tanveer, A. Alkhayyat, A. U. Khan, N. Kumar, and A. G. Alharbi, "REAP-IIoT: Resource-efficient authentication protocol for the industrial Internet of Things," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 24453–24465, Dec. 2022.
[42] K. Mahmood, Z. Ghaffar, M. Farooq, K. Yahya, A. K. Das, and S. A. Chaudhry, "A security enhanced chaotic-map based authentication protocol for Internet of Drones," *IEEE Internet Things J.*, early access, Mar. 20, 2024, doi: 10.1109/JIOT.2024.3379930.
[43] 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 Things J.*, vol. 9, no. 4, pp. 2578–2591, Feb. 2022.
[44] A. Aldosary and M. Tanveer, "PAAF-SHS: PUF and authenticated encryption based authentication framework for the IoT-enabled smart healthcare system," *Internet Things*, vol. 26, Mar. 2024, Art. no. 101159.
[45] C.-M. Chen, Z. Li, A. K. Das, S. A. Chaudhry, and P. Lorenz, "Provably secure authentication scheme for fog computing-enabled intelligent social Internet of Vehicles," *IEEE Trans. Veh. Technol.*, to be published.
[46] H. Alasmary and M. Tanveer, "ESCI-AKA: Enabling secure communication in an IoT-enabled smart home environment using authenticated key agreement framework," *Mathematics*, vol. 11, no. 16, p. 3450, 2023.

OMAR ALRUWAILI received the Ph.D. degree in computer engineering from Florida Institute of Technology. He is currently an Assistant Professor and the Chair of the Computer Engineering and Networks Department, College of Computer and Information Science, Jouf University. He is also actively teaching and researching to publish articles in field on computer engineering and networks, wireless sensor networks, and the Internet of Things.

FAISAL MOHAMMED ALOTAIBI received the M.S. degree in computer science from Liverpool John Moores University, and the Ph.D. degree in computer science from the University of Liverpool in 2022. He is currently an Assistant Professor with Prince Sattam Bin Abdulaziz University. His research interest includes security and privacy in the Internet of Things. He is actively engaged in artificial intelligence research, and his work also includes designing AI/ML-based intrusion detection systems. Additionally, he serves as a reviewer for several esteemed journals.

MUHAMMAD TANVEER received the M.S. degree in computer science from the Institute of Management Sciences, Lahore, in 2017, and the Ph.D. degree in computer science from the GIK Institute of Engineering Sciences and Technology, in 2022. He is currently an Assistant Professor with the University of Management and Technology, Lahore. His research interests include security and privacy in the Internet of Things. He is involved in authentication mechanisms using quantum cryptography and AEAD. Additionally, his research encompasses designing AI/ML-based intrusion detection systems. He also serves as a reviewer for several reputable journals.

SLIM CHAOUI received the Dipl.-Ing. degree in electrical engineering from the Technical University of Braunschweig, Germany, in 1997, and the Ph.D. degree in communications from the Technical University of Darmstadt, Germany, in 2003. He is currently an Associate Professor of communications and networks with the College of Computer and Information Sciences, Jouf University, Saudi Arabia. His research interests include channel coding, digital communications, coded cooperative networks, lossless image compression, and IA applied to communication systems.

AMMAR ARMGHAN (Senior Member, IEEE) received the bachelor's degree from COMSATS University, in 2006, the M.S. degree in electronics and communication engineering from the University of Nottingham, in 2010, and the Ph.D. degree from Wuhan National Laboratory of Optoelectronic, Huazhong University of Science and Technology, Wuhan, China. He is currently an Associate Professor of electrical engineering with Jouf University. He has published more than 150 high-impact factor articles in the last three years. His research interests include machine learning, complementary metamaterial-based microwave and terahertz devices, the Internet of Things, optics, and photonics. He is a reviewer of several reputed journals.