This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2023.3283347

Securing IoT-based Smart Healthcare Systems by using Advanced Lightweight Privacy-Preserving Authentication Scheme

Sangjukta Das, *Member, IEEE,* Suyel Namasudra, *Member, IEEE,* Suman Deb, *Member, IEEE,* Pablo Moreno Ger, and Ruben Gonzalez Crespo, *Senior Member, IEEE*

***Abstract*—In the healthcare network, the Internet of Things (IoT) devices are connected to the network for enabling remote monitoring of patients' health. IoT device security, however, is a serious concern because typical security measures might not be appropriate for IoT devices, making them naturally vulnerable to physical and copying attacks. Therefore, device authentication is a very essential security concern for IoT networks. Additionally, the storage and processing power of these devices are constrained. To address all these requirements, Physically Unclonable Functions (PUFs) for device authentication is a potential strategy. In this paper, an advanced lightweight authentication scheme for IoT devices is proposed by using PUF. This scheme provides robust authentication without storing any sensitive information on the device's memory and establishes the session key exchange process simultaneously. Moreover, this scheme preserves device privacy by including a temporary identity, which is updated at the end of each session. The effectiveness of this novel model is assessed, and results demonstrate that it is more effective and secure than many existing schemes.**

***Index Terms*— Untracebility, Key Agreement, Anonymity, PUF.**

I. INTRODUCTION

THE Internet of Things (IoT) has recently become one of the most popular research topics in both industry and academia. IoT refers to a network of objects, such as sensors, actuators, embedded technology, and smartphones, which are connected by Internet connections. Nowadays, IoT devices are being used by households, workplaces, major corporations, etc., to have network connectivity and to exchange data. One of the numerous applications of IoT is the smart home, which uses IoT along with machine learning techniques to get cost-efficient solutions for energy management with great accuracy [1]. Intelligent transportation system is another application that uses this technology for traffic management and sustainable transportation planning [2]. IoT is also extensively being used in the healthcare domain as well as in the agricultural domain and environmental ecosystems for digital monitoring [3, 4].

However, connecting these devices to the cloud raises data security risks and makes it possible for any unauthorized user to access data available on an IoT network. The implementation of IoT devices on a greater scale also causes many security attacks on both physical and network levels [5]. For example, if an attacker is able to access the devices, s/he can perform a variety of physical attacks to learn the secrets stored in devices and corrupt both devices and the system as a whole. IoT devices are vulnerable to cyberattacks as they lack strong security protocols because of their limited resources to process complex operations. Therefore, while deploying IoT devices on a healthcare network, it is crucial to take security and privacy requirements and problems into account. Different facets of security for IoT applications have been studied by researchers. Yet, IoT devices have significant challenges in maintaining data security and privacy due to device heterogeneity, resource-constrained nature, etc. To solve these security issues in an IoT-enabled healthcare network, numerous studies have designed and deployed effective solutions based on digital certification, access control, and authentication [6]. In [7], one lightweight protocol for user authentication is proposed by using cryptographic operations and biometric information. A mutual authentication scheme is suggested in [8] for a heterogeneous IoT environment. This scheme maintains user privacy by providing anonymity, and it cannot resist security attacks like impersonation attacks. Although conventional authentication methods are considered as secure, an attacker can use a variety of physical attacks to fraudulently capture the data stored on an IoT device [9]. Here, a two-factor authentication system can solve the aforementioned issues by ensuring layered defense, thus, making it more difficult for unauthorized users to access IoT devices [10].

In the above context, the PUF is one of the most dependable and robust security functions used to secure IoT devices [11]. PUFs are designed based on digital logic and Integrated Circuits (ICs) are acknowledged as a promising primitive for hardware security. They are hardware modules that operate as one-way operations. It is difficult to replicate these operations since they provide different outputs for the same inputs [12]. The PUFs are very helpful for enhancing security in devices that cannot support complicated

S. Das is with the Department of Computer Science and Engineering, National Institute of Technology Patna, Bihar, India. Email: sangjukta24@gmail.com

S. Namasudra and S. Deb are with the Department of Computer Science and Engineering, National Institute of Technology Agartala, Tripura, India. Email: suyelnamasudra@gmail.com, sumandeb.cse@nita.ac.in.

P. M. Ger and R. G. Crespo are with the Universidad Internacional de La Rioja, Logroño, Spain. Email: {pablo.moreno, ruben.gonzalez}@unir.net

cryptographic operations. As a result, they are quite beneficial in IoT networks with limited resources. The PUF can be carefully embedded inside the IoT device to make it unclonable and uniquely identifiable. In an area like the healthcare sector, IoT technology is extensively being used for both pre and post-operational monitoring, medication, and medical alerting. Here, PUF-based security can provide privacy and authorized access to patient data. PUF-based authentication can support some factors, such as fingerprints or biometrics along with generic authentication factors like passwords and tokens [13]. In [14], a PUF-based lightweight mutual authentication protocol is designed for IoT applications. The authors have implemented and analyzed the performance of this scheme in terms of energy, memory, and power utilization. Another PUF-based mutual authentication protocol is proposed in [15] for IoT systems. Here, the IoT device only uses PUF and does not use secret keys for the authentication process. However, these schemes cannot ensure the privacy of IoT devices. To solve the privacy issue, one two-factor lightweight authentication scheme using PUF is proposed in [16] that preserves the privacy of IoT devices. However, PUF-based authentication protocols are often vulnerable to many security threats like message tampering, mutual authentication threat, key-agreement attacks, physical and side-channel attack, impersonation attacks, etc. Till date, many PUF-based protocols are proposed and validated using various logical methods to address these issues and to protect IoT devices. As IoT devices in a healthcare system generate critical and sensitive data, it requires security mechanisms in order to maintain the anonymity and privacy of its user or patients.

To address all the above-mentioned issues, a lightweight and privacy-preserving authentication scheme for the IoT-based healthcare environment is proposed in this paper. Here, the pseudonym identity of each device is generated by using random integers and PUF outputs are used for authentication purposes. PUFs provide a distinctive hardware fingerprint to the devices by taking advantage of the natural random changes present in an integrated circuit. As a result, the proposed work is anonymous and safe against user identity profiling attacks. Below are the main contributions of this work:

1) This work presents a lightweight device authentication scheme by using PUFs to provide security to the data in healthcare systems.
2) It provides privacy-preserving, anonymous identity and untraceability to the devices, and enables devices to authenticate themselves without revealing their details.
3) Here, at the end of each communication, this protocol updates its credentials, as well as anonymous identities, which improves data security against cyberattacks.

The rest of the paper is organized as follows. Related works and preliminary studies are presented in sections II and III. Sections IV and V present the overview and the construction of the proposed scheme, respectively. Sections VI and VII discuss the security and performance analysis of the proposed technique, respectively. Finally, the entire work is concluded with future works in section VIII.

II. Related Works

In the literature, some works related to the proposed work are proposed to authenticate and generate a session key for protecting data. In [17], one authentication scheme is proposed for low-power mobile devices. This scheme is susceptible to password brute-force attacks due to the lack of fundamental security requirements. Masud et al. [18] have proposed a lightweight mutual authentication scheme to create a secure channel between the device and the user. Although this secure channel between the user and device prevents unauthorized users from getting access to network data, this scheme cannot resist attacks like device capture attacks. Another authentication scheme for network nodes based on biometric data is proposed by Koya et al. [19]. This gives better security by combining the patient’s electrocardiogram signals with the authentication protocol. However, this scheme faces untraceability and key-escrow issues. This scheme is improved by Gupta et al. [20] by including an anonymous authentication and key agreement technique. Still, scalability issues exist in the scheme of Gupta et al. [20] because of high communication and computation overheads. Also, there exist many schemes based on mutual authentication [21-23].

Many Radio Frequency Identification (RFID) systems and wireless sensor networks use PUFs to achieve secure authentication methods [24]. A double PUF-based RFID identity authentication protocol is proposed by Liang et al. [25]. This scheme is vulnerable to denial-of-service attacks because the messages in this scheme are not authenticated. Alladi et al. [26] have designed a mutual authentication scheme using the Challenge Response Pair (CRP) of the PUF for IoT-enabled healthcare systems. It is a two-phase authentication process to increase physical security against node tampering and node replacement attacks in the healthcare system. However, this scheme is vulnerable to many attacks as the CRPs are stored in the database of the device during the registration process. Another PUF-based authentication for the Internet of Medical Things (IoMT) is proposed by Yanambaka et al. [27]. In this scheme, both the server and the IoMTs are PUF-equipped and the gathered CRPs are stored in a third-party database. However, the messages are not encrypted in this scheme, when they are exchanged between different entities. Due to this reason, this scheme is simple to undertake modeling attacks. Additionally, PUF response noise correction is not considered in this scheme. Another authentication and key agreement technique is suggested by Wang et al. [28] to ensure secure communication between IoT nodes. This is a PUF-based technique that uses lightweight cryptographic operations and the reverse fuzzy extractor to re-produce responses correctly in a noisy environment. However, the higher number of cryptographic processes used in this scheme increases overall computation time. Many PUF-based authentication schemes are also proposed by researchers that use computationally expensive public key cryptography [29-32]. Most of these schemes do not provide the anonymity feature of IoT security protocol. Many other alternative mutual authentication systems based on advanced technologies are found in the literature [33-37].

III. Preliminary Studies

A. Physical Unclonable Function

Physical Unclonable Function is an IC that can output an arbitrary string of bits called the response from a string of bits known as a challenge. PUF provides a unique CRP due to the random variances during the fabrication process of ICs [38]. In a CRP, $R = P(C)$, i.e., PUF P's response R to a challenge C identifies a PUF. Every PUF responds uniquely to the same challenge, indicating that each PUF is unique. However, the output of a PUF may be impacted by environmental conditions, including temperature and voltage. The use of fuzzy extractors may circumvent this issue and provide robust PUF replies suitably for security applications [39].

B. Fuzzy Extractor

The Fuzzy Extractor FE generates probabilistic keys by using two algorithms, namely key Generation (FE.Gen) and key Reconstruction (FE.Rec) [39]. FE.Gen generates a key K and helper data ($hdata$), (K, hd) from an input bit string R as $FE.Gen(R)$. FE.Rec can reconstruct K from a noisy input R' by using $hdata$ as $K = FE.Rec(R', hd$ata).

IV. Overview of the Proposed Scheme

The system model and design goals of the proposed scheme are discussed in this section.

A. System Model

The system model considered in the proposed scheme is similar to the system model used in [40]. Here, the goal is to create mutual authentication and direct communication between any two communicating entities. In Fig. 1, a simplified representation of the system model is shown. The two entities considered in the proposed system model are IoT Device (IoTD) and Central System (CS). Here, it is also assumed that IoT devices have resource limitations, while the server does not have any such limitations and the server is trusted. The role of each entity is defined below:

1) ***IoT Device:*** An IoT device is associated with a patient's body, collects real-time data, and sends them to the server via a gateway device. The device must validate its authenticity before sending it to the CS. Here, the device is a resource-constrained node. Through the Internet, IoT devices interact and send data to the server. It is considered that every device has a PUF. Any effort to tamper with the PUF causes the device to behave functionally differently.
2) ***Central System:*** The device shares the collected data with the CS. The CS initializes the entire system at the beginning and registers all other entities in the system. The CS authenticates each device before creating a secure channel for data exchange.

B. Design Goals

While designing the proposed scheme, a few goals are considered, which are mentioned below:

1) ***Privacy Preservation:*** Sensitive parameters like identity-related information can be used by attackers to carry out impersonation attacks, Man-In-The-Middle (MITM) attacks, or physical attacks. Thus, it is crucial to maintain privacy, when exchanging data over a network.
2) ***Lightweight:*** IoT devices' capacities for computation are constrained. Therefore, to support the processing capacity of devices, any security mechanism must be designed by using lightweight cryptography processes.
3) ***Message Integrity:*** If any malicious user changes the healthcare data, the entire system may be compromised. So, the integrity of healthcare data must be preserved through some advanced and unbreakable procedures.
4) ***Mutual Authentication:*** The healthcare network has many devices, users, and associated equipment. To establish a secure communication channel, each device and user must authenticate themselves with the central server and agree on a session key.

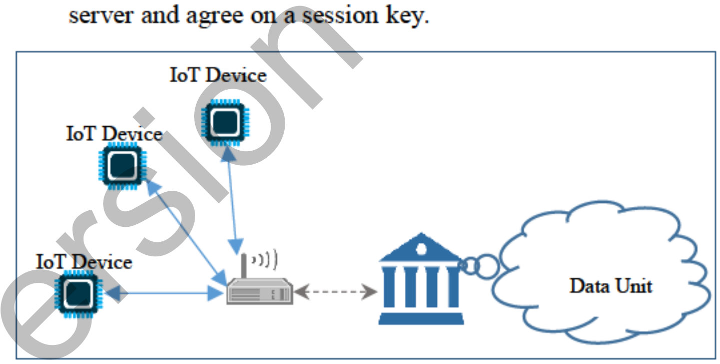


Fig. 1. System model of the proposed scheme

TABLE I
Symbol Description

Symbol	Description
$\mathcal{T}_{id}$	Temporary identity
$\mathcal{D}_{id}$	Device identity
$\{Ch, \mathcal{R}s\}$	Initial CRP
$\{Ch_s, \mathcal{R}s_s\}$	Additional CRP set
$\{x_1, x_2\}$	Random nonce
$P(.)$	PUF operation
$H(.)$	Hash operation
T_Δ	Valid time range
T_C	Current timestamp
$\mathcal{T}_{id}{}^{new}$	New temporary identity
$\mathcal{H}_1$	Intregrity checker for meassage $\mathcal{M}_1$
SK	Secret session key
R_1, R_2, R_3	Messages exchanged during registration
A_1, A_2, A_3	Messages exchanged during Authentication

V. Construction of The Proposed Scheme

The proposed scheme using a PUF is presented in this section for mutual authentication and key exchange. Along with PUF, the fuzzy extractor is also employed in the proposed protocol to reproduce the same keys. The proposed scheme is constructed in four phases, namely system initialization, device registration, authentication, and update phase. The workflow during different phases of the proposed

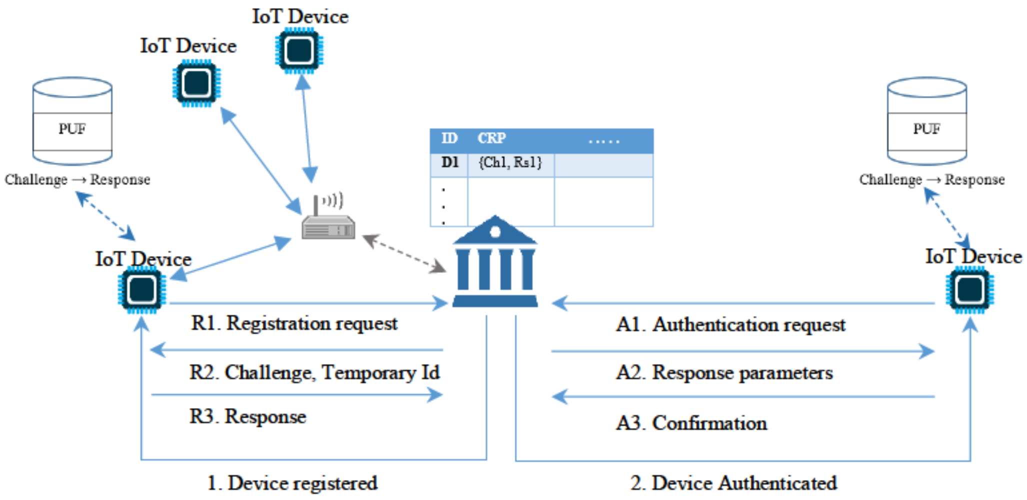


Fig. 2. Proposed scheme's workflow

scheme is illustrated in Fig. 2 and Table I lists the main symbols used in this protocol.

A. System Initialization

The central server initializes the system and selects its key values like a master key, private and public key, and other parameters. The CS uses this master key during the device registration process to generate temporary identities of devices.

B. Device Registration

Initially, each device enrolls itself with the CS by generating CRP, temporary identity, and other random values. At the end of this process, the CS stores the primary CRP, $\mathcal{T}_{id}$, and $\mathcal{D}_{id}$ for each device. Along with this information, the CS stores another set of CRP to prevent an emergency situation like DoS attacks. However, the device stores its $\mathcal{T}_{id}$, $\mathcal{D}_{id}$, primary Challenge Ch, and Challenge set Ch_s. The device does not store its response string corresponding to Ch and Ch_s. One of the most important aspects of this scheme is that the device does not store any secrets, which prevents physical attacks. The registration process initiated by the device is described in the below steps. For simplicity, this process is also shown in Fig. 3, where the messages exchanged during this process are represented by R_x, where $x = \{1,2,\ldots,n\}$.

Step 1: The device sends its identity $\mathcal{D}_{id}$ to the CS through a registration request R_1, which is the first message during the registration process.

Step 2: The CS generates $\mathcal{T}_{id}$ by calculating the hash of the string $(\mathcal{D}_{id}||\boldsymbol{mk})$, where $\boldsymbol{mk}$ is the master key of the CS. The CS selects Ch and Challenge set Ch_s. Then, the CS sends the second message R_2 containing $\mathcal{T}_{id}$, Ch, and Ch_s to the device.

Step 3: The device extracts Ch and Ch_s from R_2 to generate $\mathcal{R}s = P(Ch)$ and $\mathcal{R}s_s = P(Ch_s)$. Finally, the device stores $\mathcal{T}_{id}$, Ch, and Ch_s, and sends the third registration-related message R_3 to the CS.

Step 4: On receiving the response strings corresponding to the challenges sent through R_2, the CS finally stores $\mathcal{D}_{id}$, $\mathcal{T}_{id}$, the primary CRP, and the additional CRP set for the device.

The device registration is an offline process, and the device does not store $\mathcal{R}s$ and $\mathcal{R}s_s$. Both of these factors reduce the possibility for an attacker to obtain crucial information about the device.

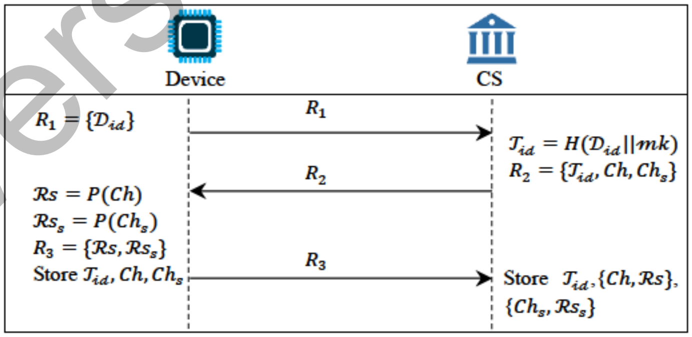

Device
CS
$R_1 = \{\mathcal{D}_{id}\}$
R_1
$\mathcal{T}_{id} = H(\mathcal{D}_{id}||\boldsymbol{mk})$
$R_2 = \{\mathcal{T}_{id}, Ch, Ch_s\}$
R_2
$\mathcal{R}s = P(Ch)$
$\mathcal{R}s_s = P(Ch_s)$
$R_3 = \{\mathcal{R}s, \mathcal{R}s_s\}$
Store $\mathcal{T}_{id}, Ch, Ch_s$
R_3
Store $\mathcal{T}_{id}, \{Ch, \mathcal{R}s\}$, $\{Ch_s, \mathcal{R}s_s\}$

Fig. 3. Device registration process

C. Authentication

In this subsection, the device authentication process is explained in detail. To create a secure communication channel for data transfer, the device initiates the authentication process by sending an authentication request message. The pictorial representation of the interaction between the device and the central system is given in Fig. 4. In Fig. 4, the messages exchanged during the authentication phase are represented by A_x, where $x = \{1,2,\ldots,n\}$. The device authentication process is executed by the following steps.

Step 1: The IoT device takes its $\mathcal{T}_{id}$ and Ch from its secure memory to generate the PUF response corresponding to Ch as $\mathcal{R}s = P(Ch)$. Then, it encrypts $\boldsymbol{x}_1$ by using $\mathcal{R}s$ and forms a message $\mathcal{M}_1 = \{\mathcal{T}_{id}, \{\boldsymbol{x}_1\}_{\mathcal{R}s}\}$. The integrity checker $\mathcal{H}_1$ of $\mathcal{M}_1$ is also computed as $\mathcal{H}_1 = H(\mathcal{M}_1||\mathcal{R}s||T^s)$, where T^s is the current timestamp. The device sends the authentication request A_1 to the server.

Step 2: The CS receives A_1 and checks the validity of T^s. Then, the CS searches for the $\mathcal{T}_{id}$ in its database and takes the

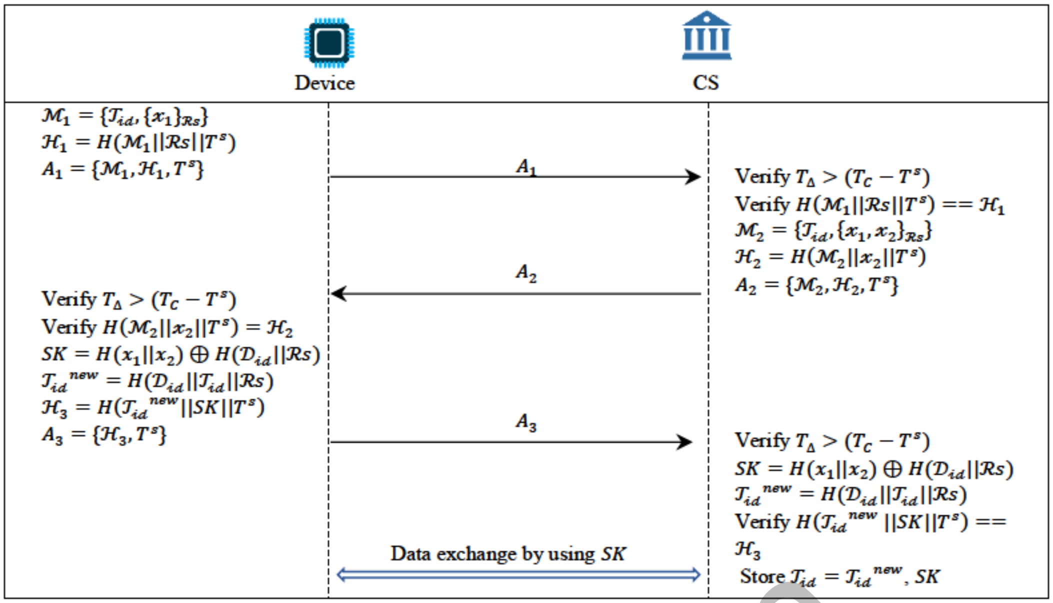

Device
CS
$\mathcal{M}_1 = \{\mathcal{T}_{id}, \{x_1\}_{\mathcal{R}s}\}$
$\mathcal{H}_1 = H(\mathcal{M}_1 || \mathcal{R}s || T^s)$
$A_1 = \{\mathcal{M}_1, \mathcal{H}_1, T^s\}$
A_1
Verify $T_\Delta > (T_C - T^s)$
Verify $H(\mathcal{M}_1 || \mathcal{R}s || T^s) == \mathcal{H}_1$
$\mathcal{M}_2 = \{\mathcal{T}_{id}, \{x_1, x_2\}_{\mathcal{R}s}\}$
$\mathcal{H}_2 = H(\mathcal{M}_2 || x_2 || T^s)$
$A_2 = \{\mathcal{M}_2, \mathcal{H}_2, T^s\}$
A_2
Verify $T_\Delta > (T_C - T^s)$
Verify $H(\mathcal{M}_2 || x_2 || T^s) = \mathcal{H}_2$
$SK = H(x_1 || x_2) \oplus H(\mathcal{D}_{id} || \mathcal{R}s)$
$\mathcal{T}_{id}^{new} = H(\mathcal{D}_{id} || \mathcal{T}_{id} || \mathcal{R}s)$
$\mathcal{H}_3 = H(\mathcal{T}_{id}^{new} || SK || T^s)$
$A_3 = \{\mathcal{H}_3, T^s\}$
A_3
Verify $T_\Delta > (T_C - T^s)$
$SK = H(x_1 || x_2) \oplus H(\mathcal{D}_{id} || \mathcal{R}s)$
$\mathcal{T}_{id}^{new} = H(\mathcal{D}_{id} || \mathcal{T}_{id} || \mathcal{R}s)$
Verify $H(\mathcal{T}_{id}^{new} || SK || T^s) ==$
$\mathcal{H}_3$
Data exchange by using SK
Store $\mathcal{T}_{id} = \mathcal{T}_{id}^{new}$, SK

Fig. 4. Authentication process

primary CRP, i.e., $\{Ch, \mathcal{R}s\}$ pair. Again, the CS checks the integrity of the message by calculating the integrity checker $\mathcal{H}_1$ of $\mathcal{M}_1$. If any of these validation processes are failed, the subsequent steps are not executed and the entire process is terminated. The CS selects random x_2 and calculates the message $\mathcal{M}_2$ after encrypting $\{x_1, x_2\}$ by $\mathcal{R}s$. The CS forms the second message $A_2 = \{\mathcal{M}_2, \mathcal{H}_2, T^s\}$ after computing the integrity checker $\mathcal{H}_2$ as $H(\mathcal{M}_2 || x_2 || T^s)$ and sends it to the device. Here, the T^s used in $\mathcal{H}_2$ and A_2 is a new timestamp selected by the CS.

Step 3: The device verifies the validity of T_S. If T_S is valid, the device obtains the random nonce x_2 from the message $\mathcal{M}_2$ by using $\mathcal{R}s$, and again, checks the integrity message $\mathcal{H}_2$ by calculating $H(\mathcal{M}_2 || x_2 || T^s)$. Then, the device calculates the secret session key SK, $\mathcal{T}_{id}^{new}$, $\mathcal{H}_3$, and A_3. The device sends the third authentication message A_3 to the CS.

Step 4: After receiving A_3, the CS, checks the validity of T_S. Then, it verifies the third integrity $\mathcal{H}_3$ by verifying $SK = H(x_1 || x_2) \oplus H(\mathcal{D}_{id} || \mathcal{R}s)$ and $\mathcal{T}_{id}^{new} = H(\mathcal{D}_{id} || \mathcal{T}_{id} || \mathcal{R}s)$. If $\mathcal{H}_3$ is valid, the CS stores $\mathcal{T}_{id}^{new}$ and SK of the device.

Here, SK is the shared secret session key between the device and the server. This key is valid for only this session. Once the data exchange is completed, the session ends and SK is discarded. Thus, the mutual authentication and key agreement processes are completed, which are also shown in Fig. 4.

D. Credential Update

To maintain the freshness assurance of the authentication protocol [34], the server may update the related CRPs for each device by acquiring new CRPs. The CRP updating process is shown in Fig. 5 and the steps are discussed below:

Step 1: The server sends an update request with the new challenge Ch_1 to the device to change the current CRP. During this phase, $\mathcal{T}_{id}$ of the device can also be updated.

Step 2: On receiving the update request U_1, the device calculates a new response $\mathcal{R}s_1$ corresponding to Ch_1 and second update message U_2. Then, the device sends U_2 and $\mathcal{H}_5$ to the CS. At this stage, the device changes the challenge stored in its memory.

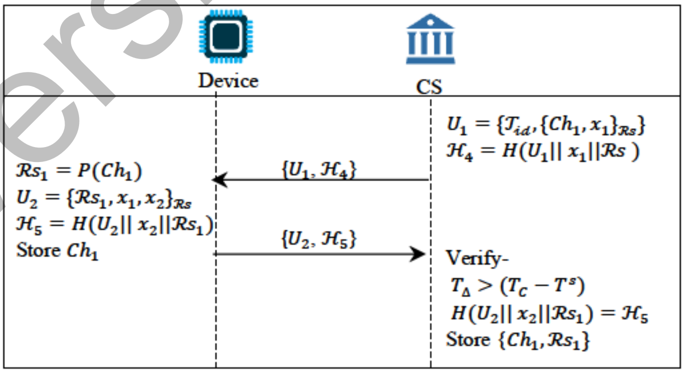

Device
CS
$U_1 = \{\mathcal{T}_{id}, \{Ch_1, x_1\}_{\mathcal{R}s}\}$
$\mathcal{H}_4 = H(U_1 || x_1 || \mathcal{R}s)$
$\{U_1, \mathcal{H}_4\}$
$\mathcal{R}s_1 = P(Ch_1)$
$U_2 = \{\mathcal{R}s_1, x_1, x_2\}_{\mathcal{R}s}$
$\mathcal{H}_5 = H(U_2 || x_2 || \mathcal{R}s_1)$
Store Ch_1
$\{U_2, \mathcal{H}_5\}$
Verify-
$T_\Delta > (T_C - T^s)$
$H(U_2 || x_2 || \mathcal{R}s_1) = \mathcal{H}_5$
Store $\{Ch_1, \mathcal{R}s_1\}$

Fig. 5. Credential update phase

Step 3: On receiving U_2 and $\mathcal{H}_5$, the CS verifies the timestamp and recalculates $\mathcal{H}_5$. If it matches with the received $\mathcal{H}_5$, then, the CS updates its CRP list with $\{Ch_1, \mathcal{R}s_1\}$.

VI. Security Analysis

This section begins with briefly introducing a number of attacks against PUF-based IoT authentication methods. Then, the proposed protocols' informal and formal security analyses are presented. The robustness of the proposed scheme, i.e. Advanced Lightweight Privacy-Preserving Authentication (ALPAS) is assessed against some well-known attacks.

A. Formal Analysis

This section formally analyses the security of ALPAS by using Dolev-Yao (DY) and Canetti-Krawczyk (CK) adversary models, and their assumptions mentioned in [40]. ALPAS has two main entities, namely Device and CS. According to the threat model, an adversary $\mathcal{A}$ has capability to capture, corrupt, alter, delete, or replay all messages sent over the communication channel. $\mathcal{A}$ can perform the following queries:

1) $\boldsymbol{Execute\_Evs}()$**:** $\mathcal{A}$ can eavesdrop over the communication channel between a device and the CS to get all the messages exchanged by executing this query.
2) $\boldsymbol{Send\_Msg}()$**:** By executing this query, $\mathcal{A}$ can send messages to a device and the CS, and can also receive a reply from them.
3) $\boldsymbol{Capture\_Par}()$: $\mathcal{A}$ can execute $Capture\_Par()$ query to capture all the parameters stored in the device and CS's memory. However, $\mathcal{A}$ can execute only a limited number of $Capture\_Par()$ query.
4) $\boldsymbol{Reveal}()$: $\mathcal{A}$ runs this query to reveal the secrets stored in the device's memory using a physical attack.

Considering S is the event, where $\mathcal{A}$ wins a game, then, $\mathcal{A}$'s advantage to break ALPAS is $Adv_{\mathcal{A}} = |2Prob(S) - 1|$. If $Adv_{\mathcal{A}} \leq \varepsilon$, ALPAS is secure, where $\varepsilon > 0$.

Lemma 1. *The output of a PUF cannot be guessed.*

Proof. According to [29], a PUF cannot be replicated and it generates unique responses. A PUF produces a response of n bits for a challenge of m bits such that $\{0,1\}^m \rightarrow \{0,1\}^n$. A security game between an adversary and challenger can be performed as follows. $\mathcal{A}$ can send a query to the PUF using challenge polynomial times. At first, $\mathcal{A}$ sends a challenge Ch^i to $\mathbb{C}$. $\mathbb{C}$ reveals Rs^i as PUF(Ch^i) to $\mathcal{A}$ and sends another challenge Ch^{i+1} to get Rs^{i+1} as PUF(Ch^{i+1}). Then, the adversary $\mathcal{A}$ wins the game, if its guess response $Rs_{\prime}^{i}$ for Ch^i is the same as Rs^i. This indicates $\mathcal{A}$'s advantage in this game is $Adv_{\mathcal{A}}^{puf} = \Pr[Rs^i = Rs_{\prime}^{i}]$. $\mathcal{A}$ can only guess the output of a PUF to a given challenge. Therefore, $Adv_{\mathcal{A}}^{puf} = \frac{1}{2^n}$.

Lemma 2. *The secrets used in ALPAS cannot be revealed by the Reveal oracle. The temporary identities of devices also cannot be correlated by the Reveal oracle.*

Proof: In ALPAS, IoT devices do not store any secret or sensitive data in their local memory. The device stores only its $\mathcal{T}_{id}$, $\mathcal{D}_{id}$, and Ch. According to the assumptions in the threat model [40], $\mathcal{A}$ cannot get Rs by using Ch or invoking the Reveal oracle. Initially, the temporary identity $\mathcal{T}_{id}$ of a device is calculated as $H(\mathcal{D}_{id}||mk)$, which is updated at each new round. Each $\mathcal{T}_{id}$ is therefore only valid for a single session. As a result, $\mathcal{A}$ cannot correlate the temporary identities unless $\mathcal{A}$ is able to receive the secret response, which is impossible. In this case, $\mathcal{A}$'s advantage is $Adv_{\mathcal{A}}^{\mathcal{T}_{id}} = Pr[Corr(\mathcal{T}_{id}, {\mathcal{T}_{id}}^{new}) \neq 0] \approx 0$, where $Corr$ is the correlation coefficient.

Theorem 1. *Mutual Authentication: This protocol can be successfully executed between the device and the CS only if both entities are legitimate.*

Proof: By impersonating an authentic device, $\mathcal{A}$ can try to establish authentication with the server. To simulate this attack, a security game between $\mathcal{A}$ and $\mathbb{C}$ can be performed as follows. Initially, to perform the proposed authentication with the CS, $\mathbb{C}$ selects any legitimate device, namely $\mathcal{D}^*$. $\mathcal{A}$ can send a polynomial number of query requests to the CS and $\mathcal{D}^*$. $\mathcal{A}$ attempts to authenticate itself as a valid device to the CS. If $\mathcal{A}$ successfully completes the authentication step of ALPAS, $\mathcal{A}$ wins the game. If $\mathcal{A}$ is able to produce the third integrity checker $\mathcal{H}_3 = H({\mathcal{T}_{id}}^{new}||SK||T^s)$, only then, it can properly authenticate itself. $\mathcal{A}$ can try to reveal $\mathcal{R}s$ embedded within the SK. Suppose $\mathcal{A}$ is able to reveal n' bits of $\mathcal{R}s$, where $n' < n$. Then, the advantage $Adv_{\mathcal{A}}^{\mathcal{R}s} = Pr[Rs^i = Rs_{\prime}^{i}]$ is $\frac{1}{2^{n-n'}}$. Therefore, $\mathcal{A}$'s advantage of the successful authentication process with the CS is $Adv_{\mathcal{A}}^{Auth} = P\,r[Rs^i = Rs_{\prime}^{i}] - Adv_{\mathcal{A}}^{puf}$. However, $\mathcal{A}$ can only randomly guess $\mathcal{R}s$, i.e., $n' = 0$ by Lemmas 1 and 2 and $P\,r[Rs^i = Rs_{\prime}^{i}] = \frac{1}{2^n}$. So, $Adv_{\mathcal{A}}^{Auth} = P\,r[Rs^i = Rs_{\prime}^{i}] - Adv_{\mathcal{A}}^{puf} = 0$.

Theorem 2. *Privacy: The proposed ALPAS maintains the anonymity of the device.*

Proof: The ALPAS protocol is said to be untraceable if $\mathcal{A}$ cannot correlate two executions of ALPAS by the same $\mathcal{D}^*$ with the CS. The following security game can be used to analyze this attack. Initially, to perform the proposed scheme with the CS, $\mathbb{C}$ selects two valid devices $\mathcal{D}^*$ and $\mathcal{D}^{**}$. $\mathcal{A}$ can send polynomial numbers of query requests to the CS and devices $\mathcal{D}^*$and $\mathcal{D}^{**}$. Then, $\mathbb{C}$ selects one of the device's identity $\mathcal{D}_{id}$ randomly. $\mathcal{A}$ sends a query to the CS and $\mathcal{D}_{id}$ polynomial times. Then, $\mathcal{A}$ guesses the identity $\mathcal{D}_{id*}$. Now, if $\mathcal{D}_{id*}$==$\mathcal{D}_{id}$, $\mathcal{A}$ wins the game.

Here, $\mathcal{A}$'s advantage of guessing $\mathcal{D}_{id*}$ successfully can be presented as $Adv_{\mathcal{A}}^{pri1} = 2 * (\Pr[\mathcal{D}_{id*}{=}\mathcal{D}_{id}] - \frac{1}{2})$. As IoTD's $\mathcal{T}_{id}$s cannot be correlated, $\mathcal{A}$'s advantage of correlating $\mathcal{T}_{id}$ can be presented as $Adv_{\mathcal{A}}^{pri2} = \Pr[corr(\mathcal{T}_{id}, {\mathcal{T}_{id}}^{new}) \neq 0]$. $\mathcal{A}$'s advantage of winning this game can be presented as $Adv_{\mathcal{A}}^{pri} = Adv_{\mathcal{A}}^{pri1} + Adv_{\mathcal{A}}^{pri2}$- $Adv_{\mathcal{A}}^{pri1} \times Adv_{\mathcal{A}}^{pri2}$. If $\mathcal{A}$ guesses $\mathcal{D}_{id*}$ randomly, then, s/he has no advantage. By Lemmas 1 and 2, it can be concluded that $Adv_{\mathcal{A}}^{pri} = 0$.

B. Informal Analysis

To analyze ALPAS informally, a few attack scenarios are considered in this paper.

1) ***Replay Attack:*** Since a valid timestamp is assigned to each transmitted message, even if an attacker replays old messages, it cannot counterfeit the current transmitted message. Furthermore, every parameter, including the temporary identity of devices is updated after every new session. As a result, replay attacks are successfully avoided. In this context, to detect replay attacks, maintaining the freshness of each exchanged message is an important requirement. To fulfill this requirement, the ALPAS uses the timestamp concept, and also, allows the entities to use different credentials during different sessions, which is done via the credential update phase.

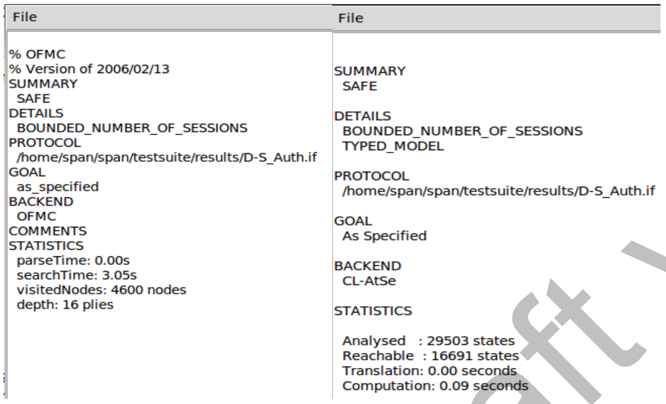
2) ***Message Analysis Attack:*** In Message Analysis attacks, an attacker tries to intercept the transmitted information between the communication entities. Despite the possibility of intercepting authentication communications, in the ALPAS technique, secret keys, session keys, and responses are private and inaccessible to an attacker. This

is accomplished by not keeping the transferred messages locally, but, encrypting and hashing them**.**

3) ***DOS Attack:*** In DOS attacks, a device is targeted by an attacker to temporarily or permanently interrupt its functionality by overloading it with service requests. In ALPAS, only two entities are present, namely CS and IoTD. On the CS's side, DOS attacks are unfeasible because of the server's high computing capabilities. Therefore, only the device is considered for this attack. The device verifies the integrity of each message after receiving any message. Due to the secret key contained in each integrity checker message, the possibility of random guessing of the hash values to pass the verification procedure is very less. The DoS attack is therefore impractical in ALPAS.

4) ***Physical Attack*:** In the proposed model, IoT devices do not keep any secret or sensitive data in their local memory. Additionally, in the system model, one assumption is that the communication between the PUF IC and the device is secure. Therefore, if an adversary gets a device, ALPAS is secure against physical attacks.

C. Formal Verification



```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/D-S_Auth.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 3.05s
  visitedNodes: 4600 nodes
  depth: 16 plies
```

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/D-S_Auth.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 29503 states
  Reachable  : 16691 states
  Translation: 0.00 seconds
  Computation: 0.09 seconds
```

Fig. 6. Summary produced by the AVISPA tool's two back ends (OFMC and CL-AtSe)

In this section, the proposed protocol is simulated using the widely used protocol security analysis and verification tool AVISPA (Automated Validation of Internet Security Protocols and Applications). There are four back-ends (OFMC, CL-AtSe, SATMC, and TA4SP) integrated into AVISPA to create a single platform for protocol verification. To define the roles and goals of the proposed protocol, AVISPA uses the formal language HLPSL (High-Level Protocol Specification Language). The backends cannot directly detect HLPSL, therefore, it is converted into Intermediate Format (IF) using the platform's HLPSL2IF translator. Then, the IF is directly built and executed in backends to verify the protocol's security.Two different entities, namely device and central server, are included in the proposed protocol. As a result, two roles in AVISPA using the HLPSL specification are defined. The role definition codes contain corresponding operations, states, and parameters. The role of the session and environment are also considered. Fig. 6 depicts the output of CL-AtSe and OFMC backends. It indicates that ALPAS is SAFE against many attacks, including MITM attacks, replay attacks, and impersonation attacks, and the confidentiality of the session key is maintained.

VII. Performance Analysis

The performance analysis of ALPAS is shown in this section in comparison to the relevant protocols [15, 16, 27, 28] in the literature. These protocols are based on some features, such as mutual authentication and error correction. Since SHA-2 is currently used in well-known security applications like Transport Layer Security (TLS) and Secure Sockets Layer (SSL), and a number of integrated circuits for commercial security, ALPAS is implemented by using SHA-2. Additionally, SHA-2 is easier and quicker to implement than SHA-3 since a wider range of hardware and software is supported by it. At first, the security properties are compared. Then, the storage requirement, computation, and communication complexities of ALPAS are assessed.

A. Security Feature Comparison

Table II compares ALPAS's security properties to various existing schemes. The schemes proposed in [27, 28] do not provide anonymity and un-traceability properties. Similarly, the scheme [16] does not provide resistance to reply attacks. However, ALPAS has all of the properties mentioned in the table.

B. Storage Requirement Comparison

The overall cost of storage of each entity in ALPAS is determined based on the size of the parameters given in Table III. Table IV lists the cost of storage in ALPAS along with other existing schemes. Each device stores $\{\mathcal{T}_{id}, Ch, Ch_s\}$ and the CS stores $\{\mathcal{D}_{id}, \mathcal{T}_{id}, \text{CRP}\}$ parameters for each device in their memory. In Fig. 7, the total storage cost and the individual storage costs for the CS and the device are shown.

C. Discussion on Experimental Results

In this section, the performance of ALPAS is evaluated in terms of computational cost and communical cost.

1) ***Computation Cost:*** The computational cost of the authentication process of ALPAS is computed by executing a number of operations. The performance of ALPAS is compared with the existing schemes by considering the same conditions and operations. The basic operations that are used in ALPAS and other related schemes are hash, XOR, concatenation (||), PUF, and noise correction. For cost evaluation, only PUF and hash operations are considered since the other operations are comparatively very less time-consuming. Table V represents the time taken to execute each operation by device and server. Then, the number of operations used in the existing protocols and the proposed scheme is compared, which is listed in Table VI. The computation cost comparison is graphically represented in Fig. 8. It can be seen that ALPAS uses comparatively less operations than the other schemes, thus, the computation cost is $10H_f+1P_f$. It is seen that ALPAS takes comparatively less execution time than the other protocols.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2023.3283347

TABLE II
COMPARISONS OF SECURITY FEATURES

Security Feature	Aman et al. [15]	Gope et al. [16]	Yanambaka et al. [27]	Wang et al. [28]	Proposed Scheme
Resistance to replay attack	√	×	×	√	√
Anonymity	√	√	×	×	√
Traceability	√	√	×	×	√
PUF security	×	√	√	×	√
Resistance to physical attack	√	√	√	×	√
Noise consideration in PUF	×	√	×	×	√

TABLE III
SIZE OF THE PARAMETERS

Parameter	Length in Bits	Parameter	Length in Bits
Secret keys	160	Timestamp	32
Hash	256	Challenge	160
Identity parameters	160	Response	160
Random nonce	160	-	-

TABLE IV
STORAGE COST COMPARISON (BITS)

Scheme	Device	CS
[16]	1576	1792
[28]	576	480
Proposed scheme	420	1116

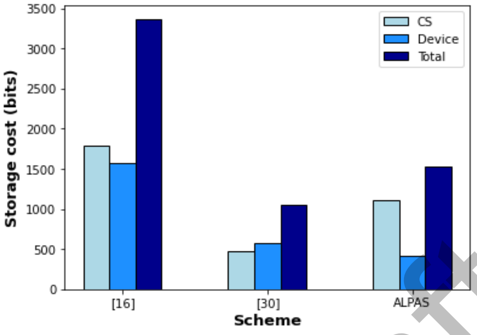


Fig. 7. Storage cost in bits

TABLE V
RUNTIME OF EACH OPERATIONS IN MILLISECONDS

Operation	CS	Device
PUF	-	0.14ms
Hash	0.012ms	0.028ms
XOR	0.003ms	0.005ms
FE.Gen	-	2.7ms
Fe.Rec	-	4.23ms

TABLE VI
COMPARISON OF COMPUTATION COST

Scheme	Device	CS	Total
[16]	$7H_f+2P_f$	$7H_f$	$14H_f+2P_f$
[28]	$7H_f+2P_f$	$7H_f$	$14H_f+2P_f$
Proposed scheme	$5H_f+1P_f$	$5H_f$	$10H_f+1P_f$

2) ***Communication Cost:*** Here, the communication cost means the total number of bits sent and received throughout the authentication procedure. The length of each message being delivered is determined by using Table III, which provides the size of the parameters used in these messages. In Table VII, the communication costs of ALPAS and the existing schemes are compared. This table also shows the total number of messages sent and received by the communicating entities, namely IoT device and CS. From Tables III and VII, it can be seen that the transmitted bits in ALPAS are 1504 bits that are less than [16] and [28].

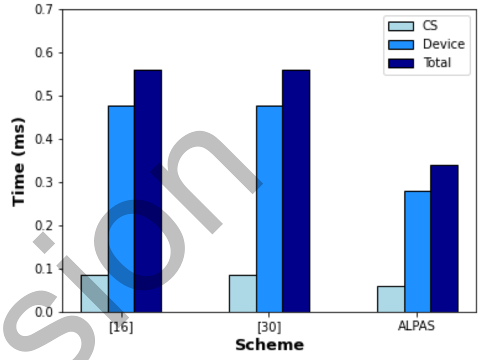


Fig. 8. Computation cost in time (ms)

TABLE VII
COMPARISON OF COMMUNICATION COST

Scheme	Total no. of Messages	Total no. of Bits
[16]	3	1568
[28]	3	1568
Proposed scheme	3	1504

VIII. CONCLUSIONS AND FUTURE WORK

In this paper, a device-to-central server mutual authentication and key exchange protocol has been developed for IoT devices in healthcare systems. The proposed protocol aims to establish a secure channel between the communicating healthcare entities for the secure exchange of sensitive and confidential healthcare data. Here, each device equipped with PUF must register itself with the central server system. This scheme eliminates the requirement to store CRPs in the device's local memory, which not only satisfies the resource limitation of IoT devices, but also reduces the security risk of device node attacks due to the accessibility to these devices. The proposed ALPAS also creates a session key, and securely exchanges it between the device and server at the end of each successful authentication phase. Furthermore, it is demonstrated that ALPAS is secure against many advanced cyber attacks, namely replay, MITM, DoS, and impersonation attacks. Most importantly, it resists physical attacks by using the PUF-based authentication technique. As a future work, it is planned to deploy this PUF-based authentication protocol with a blockchain architecture scheme to provide security and automation to IoT-based healthcare systems.

References

[1] W. Li, T. Logenthiran, V. -T. Phan, and W. L. Woo, "A novel smart energy theft system (SETS) for IoT-based smart home", *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5531-5539, 2019.

[2] Z. Huang et al., "Survey on vehicle map matching techniques", *CAAI Transactions on Intelligence Technology*, vol. 6, no. 1, pp. 55-71, 2021.

[3] P. Gangwani, A. Perez-Pons, T. Bhardwaj, H. Upadhyay, S. Joshi, and L. Lagos, "Securing Environmental IoT Data Using Masked Authentication Messaging Protocol in a DAG-Based Blockchain: IOTA Tangle," *Future Internet*, vol. 13, no. 12, pp. 312, Dec. 2021

[4] S. Das and S. Namasudra, "Multi-authority CP-ABE-based access control model for IoT-enabled healthcare infrastructure", *IEEE Transactions on Industrial Informatics*, vol. 19, no. 1, pp. 821-829, 2023.

[5] A. Gutub, "Boosting image watermarking authenticity spreading secrecy from counting-based secret-sharing", CAAI Transactions on Intelligence Technology, 2022. DOI: 10.1049/cit2.12093

[6] S. Das and S. Namasudra, "A lightweight and anonymous mutual authentication scheme for medical big data in distributed smart healthcare systems", *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2022. DOI: 10.1109/TCBB.2022.3230053.

[7] X. Li, J. W. Niu, J. Ma, W. D. Wang, and C. L. Liu, "Cryptanalysis and improvement of a biometrics-based remote user authentication scheme using smart cards", *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 73–79, 2011.

[8] T. Muhamed, B. Boštjan, and H. Marko, "A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks based on the internet of things notion", *Ad Hoc Networks*, vol. 20, pp. 96–112, 2014.

[9] C. Chang, and H. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks", *IEEE Transactions on Wireless Communications*, vol. 15, no. 1, pp. 357–366, 2016

[10] M. K. Khan and K. Alghathbar, "Cryptanalysis and security improvements of twofactor user authentication in wireless sensor networks", *Sensors*, vol. 10, no. 3, pp. 2450–2459, 2020.

[11] A. Lakhan, M. A. Mohammed, J. Nedoma, R. Martinek, P. Tiwari, and N. Kumar, "Blockchain-enabled cybersecurity efficient IIOHT cyber-physical system for medical applications", *IEEE Transactions on Network Science and Engineering*, 2022, DOI: 10.1109/TNSE.2022.3213651

[12] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, "Physical unclonable functions and applications: A tutorial," Proc. IEEE, vol. 102, no. 8, pp. 1126–1141, Aug. 2014.

[13] O. Günlü, O. İşcan, V. Sidorenko, and G. Kramer, "Code constructions for physical unclonable functions and biometric secrecy systems", *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 11, pp. 2848-2858, 2019.

[14] M. Hossain, S. Noor, R. Hasan, HSC-IoT: A Hardware and software co-verification based authentication scheme for internet of things, in: The Proceedings of 2017 5th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), San Francisco, CA, 2017, pp. 109–116,

[15] M. N. Aman, K. C. Chua, and B. Sikdar, "Mutual authentication in IoT systems using physical unclonable functions," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1327–1340, 2017.

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

[17] X. Li, J. Peng, M. S. Obaidat, F. Wu, M. K. Khan, and C. Chen, "A secure three-factor user authentication protocol with forward secrecy for wireless medical sensor network systems", *IEEE Systems Journal*, vol. 14, pp. 39–50, 2019.

[18] M. Masud, G. S. Gaba, K. Choudhary, M. S. Hossain, M. F. Alhamid, and G. Muhammad, "Lightweight and anonymity-preserving user authentication scheme for IoT-based healthcare", *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2649-2656, 2022.

[19] A. M. Koya, and D. P. P., "Anonymous hybrid mutual authentication and key agreement scheme for wireless body area network", *Computer Networks*, vol. 140, pp. 138–151, 2018.

[20] A. Gupta, M. Tripathi, and A. Sharma, "A provably secure and efficient anonymous mutual authentication and key agreement protocol for wearable devices in WBAN", *Computer Communications*, vol. 160, pp. 311–325, 2018

[21] S. Das and S. Namasudra, "Lightweight and efficient privacy-preserving mutual authentication scheme to secure Internet of Things-based smart healthcare", *Transactions on Emerging Telecommunication Technology*, 2023. DOI: 10.1002/ett.4716.

[22] Y. K. Huang, "Design of a smart cabin lighting system based on internet of things", *Cloud Computing and Data Science*, vol. 4, no. 2, pp. 112-121, 2023.

[23] Z. Chen, "Research on internet security situation awareness prediction technology based on improved RBF neural network algorithm", *Journal of Computational and Cognitive Engineering*, vol. 1, no. 3, pp. 103-108, 2022

[24] S. Karda¸s, S. Çelik, M. Yıldız, and A. Levi, "PUF-enhanced offline RFID security and privacy," *Journal of Network and Computer Applications*, vol. 35, no. 6, pp. 2059–2067, 2012.

[25] W. Liang, S. Xie, J. Long, K. Li, D. Zhang, K. Li, "A double PUF-based RFID identity authentication protocol in service-centric internet of things environments", *Information Sciences*, vol. 503, pp. 129-147,2019.

[26] T. Alladi, V. Chamola and Naren, "HARCI: A Two-Way Authentication Protocol for Three Entity Healthcare IoT Networks", *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 2, pp. 361-369, 2021.

[27] V. P. Yanambaka, S. P. Mohanty, E. Kougianos, and D. Puthal, "Pmsec: Physical unclonable function-based robust and lightweight authentication in the internet of medical things", *IEEE Transactions on Consumer Electronics*, vol. 65, no. 3, pp. 388–397, 2019.

[28] H. Wang, J. Meng, X. Du, T. Cao, and Y. Xie, "Lightweight and anonymous mutual authentication protocol for edge iot nodes with physical unclonable function," Security and Communication Networks, vol. 2022, 2022.

[29] U. Chatterjee, R. S. Chakraborty and D. Mukhopadhyay, "A PUF-based secure communication protocol for IoT", *ACM Transactions on Embedded Computing Systems*, vol. 16, no. 3, pp. 1-25, 2017

[30] H. Jennath, V. S. Anoop, and S. Asharaf, "Blockchain for healthcare: securing patient data and enabling trusted artificial intelligence", *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, pp. 15-23, 2020.

[31] Kumar, and S. Chand, "A provable secure and lightweight smart healthcare cyber-physical system with public verifiability", *IEEE Systems Journal*, 2021. DOI: 10.1109/JSYST.2021.312055.

[32] S. Khasim and S. S. Basha, "An improved fast and secure CAMEL based authenticated key in smart health care system", *Cloud Computing and Data Science*, vol. 3, no. 2, pp. 77-91, 2022.

[33] A. Kishor, C. Chakraborty, and W. Jeberson, "A novel fog computing approach for minimization of latency in healthcare using machine learning", *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, pp. 7-17, 2020.

[34] K. Lam and D. Gollmann, "Freshness assurance of authentication protocols", Proceedings of the European Symposium on Research in ComputerSecurity, pp. 261-272, 1992.

[35] A. Lakhan et al., "Restricted boltzmann machine assisted secure serverless edge system for Internet of Medical Things", *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 2, pp. 673-683, 2023.

[36] Y. Guo, Z. Mustafaoglu, and D. Koundal, "Spam Detection Using Bidirectional Transformers and Machine Learning Classifier Algorithms", *Journal of Computational and Cognitive Engineering*, vol. 2, no. 1, pp. 5–9, Apr. 2022.

[37] A. Lakhan et al., "Federated-learning based privacy preservation and fraud-enabled blockchain IoMT system for healthcare", *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 2, pp. 664-672, 2023.

[38] Dimitrios Schinianakis. Lightweight security for the internet of things: A soft introduction to physical unclonable functions. IEEE Potentials, 38(2):21–28, 2019.

[39] Y. Dodis, J. Katz, L. Reyzin, and A. Smith, "Robust fuzzy extractors and authenticated key agreement from close secrets," in Advances in Cryptology (CRYPTO) (Lecture Notes in Computer Science), vol. 4117. Heidelberg, Germany: Springer, 2006, pp. 232–250

[40] Y. Zheng, W. Liu, C. Gu, and C. H. Chang, "PUF-based Mutual Authentication and Key Exchange Protocol for Peer-to-Peer IoT Applications", *IEEE Transactions on Dependable and Secure Computing*, 2022, DOI:10.1109/TDSC.2022.3193570.