Journal of Information & Optimization Sciences
ISSN 0252-2667 (Print), ISSN 2169-0103 (Online)
Vol. 44 (2023), No. 7, pp. 1347–1364
DOI : 10.47974/JIOS-1321

PUF-based user access control scheme for IoT environment

Devender Kumar *
Sai Kishore Pachigolla †
Shubham Singh Manhas §
Karan Rawat ‡
Division of Information Technology
Netaji Subhas Institute of Technology
New Delhi
India

Abstract

A very important part of any software or hardware associated with the Internet of Things (IoT) is the User Access Control. User Access Control deals with the important security features like authenticating a legitimate user, authorizing a user, etc. A very effective and secure way to ensure the user access control is: three factor user access control. Some three factor user authentication schemes have been developed in the past, brief details regarding them can be found in further sections of the paper. In this paper, we propose a new three factor user access control scheme. Our proposed scheme is based on Mandal et. al.'s user access control scheme published recently. Our scheme involves the use of lightweight cryptographic primitives like Physically Unclonable Function (PUF), one way cryptographic hash function and bitwise exclusive OR (XOR) operations. PUFs make the scheme very lightweight and efficient as compared to other schemes of similar nature. The three factors used in our scheme are: registered device of the user, personal biometrics of the user and password of the user. We present the informal security analysis to show that our scheme is safe from several known attacks.

Subject Classification: *94A62.*

Keywords: *User access control, Internet of Things, Three factor, Physically unclonable function.*

* *E-mail:* **dk_iitm@yahoo.co.in (Corresponding Author)**
† *E-mail:* **psaikishore2798@gmail.com**
§ *E-mail:* **shubhammanhas579@gmail.com**
‡ *E-mail:* **kanurawat128@gmail.com**

I. Introduction

The Internet of Things (IoT) is a network of interconnected devices that interact with each other using the internet and with very minimal to zero human interaction. IoT devices range from small sensors which are responsible for activities like collection of environment data, etc. to very big unmanned ground vehicles. In any IoT network, many smart devices are deployed in a particular area like homes, factories, etc. These smart devices sense the information from the surroundings and process it. Then this information is sent to the appropriate gateway nodes. IoT facilitate automation in factories and thereby reduce the labour cost for the companies. All these advantages of IoT have led to increase in adoption of IoT related technologies in all facets of industries. According to an analysis done by IHS Markit [1], IoT devices will increase from 27 billion in 2017 to 125 billion in 2030.

The increase in the number of IoT devices has resulted in an increase of security concerns posed to these devices and the overall IoT network. The major security concerns include the interception, modification or deletion of communicated messages. This facilitates the need of designing and implementing a secure as well as efficient mechanism for user access control, so that we can ensure that only legitimate users are permitted to use pertinent information and services.

In this paper, we design a new PUF-based user access control scheme for IoT (PUFUAC-IoT). The salient features of PUFUAC-IoT include: providing real-time access to a legitimate registered user U for accessing an IoT device S_i (if the mutual authentication is successful), dynamic addition of a new IoT device in the existing IoT environment, revoking the access of a legitimate user U and updation of his password and biometrics. The mutual authentication in PUFUAC-IoT is done via a trusted gateway node GW. A session key is established after the successful completion of mutual authentication. This session key is then used to ensure secure communication between U and S_i.

In section II, we briefly discuss the background materials like network model, the threat model and our contribution in the paper. Related literature is discussed in section III. Our proposed PUFUAC-IoT scheme is discussed in section IV. Its security analysis is shown in section V. Its performance analysis is provided in section VI. The conclusion is presented in section VII.

II. Background

In this paper, we have made use of Physically Unclonable Fuction (PUF). It can be defined as a function that maps a set of given challenges and produces a bijective arbitrary set of responses. One of the most significant facts which makes it perfect for cryptographic functions is that these functions cannot be replicated. Hence, every PUF gives a unique random value for the same set of challenges.

Our system model is based on the IoT literature given in [2]. It consists of a large number of IoT devices installed as a collection of devices. These collections are known as cells. For any IoT application, each cell makes the use of a gateway node *GW*. The shape and size of IoT devices may vary significantly depending on the application ranging from a small environment sensor to a big IoT-controlled missile. The registration of users, the enrolment of IoT devices and gateways is done by the trusted authority *TA*. The data collected by the IoT devices / sensors can be transferred to appropriate cloud server through gateway nodes for the analysis of the data. The collected data may also be shared with the authorized users depending on their requirement. After authentication between registered user U and the required IoT device S_i, a session key is formed by *GW*. This session key allows further communication between U and S_i, and also allowing U to access data from S_i with ease.

A. *Threat model*

The following assumptions are taken about an attacker A:

- A has the ability to intercept all the messages sent or received over a public channel.
- A cannot modify and intercept the messages over a private channel.
- According to the Dolev-Yao threat model [3], A can modify the intercepted messages.
- The IoT devices may not be physically protected and A can use this opportunity to do physical and impersonation attacks.
- A can recover the credentials stored in the user's trusted device.
- The trusted authority cannot be intruded. But A can get the information kept in the trusted authority.
- A has can access all information available in the device by stealing it.

B. *Our Contribution*

Our contribution is presented as follows:

- We develop a lightweight authentication scheme for user access control phase for IOT environment.
- We show informally that it is resistant to various known attacks.
- We compare our scheme with the other related schemes [4]-[7] in terms of security features, computation cost and communication cost. We show that our scheme is more secure and has less computation cost as compared to the others.

III. Related Literature

For secure access of real-time data from sensors in wireless sensor networks (WSN) many authentication schemes have been proposed where there exists a remote third party trust center that helps in establishment of session keys between the user and the sensor nodes in order to avoid illegal access. As a result, many authentication protocols [11] have been designed for WSNs. Hence, in this section, we revisit the various user access authentication protocols for the Internet of things (IOT) environment.

In 2016, the authors in the paper [12] discussed an access control mechanism for WSNs, supporting node or identity privacy. They used elliptic key cryptography for achieving this wherein a sensor node and coordinator node mutually authenticate to establish a common session key amongst themselves. However, they could not prove message confidentiality and unforgeability at the same time. For the establishment of the above features Yu et. al. [8] and Ma et. al. [13] proposed an Access Control Scheme using Signcryption (ACSC). The main advantage of using this technique is its ability to perform signature and encryption in one logical step, thereby saving cost as opposed to signature and then encryption. Thus it allows simultaneously achieve integrity, non repudiation, confidentiality , and secure authentication with a minimised cost.

In 2017, Li et. al. [9] proposed an identity based access protocol using certificateless signcryption that provided public verifiability , ciphertext authenticity and insider security. This had an added feature of remote user access revocation for registered users. Also, Bala et. al. [14] proposed a novel protocol eliminating the use of certificate management. They proved the scheme to be secure against node impersonation and replay

attack. But, this scheme was impractical due to the use of computationally expensive bilinear pairings.

In 2018, Karati et. al.[15] proposed a Certificate less signature scheme using ECC. They showed that the scheme required low computational cost and is secure against adaptive chosen message and identity attacks. Later on, Pakniat et. al. [16] show that the scheme is vulnerable to forgery of message signatures and modify it to make it unforgeable and retain the benefits of the previous scheme. The authors in the paper [6] developed an access control scheme that provides integrated storage facility, secure data access and privacy. But it does not provide anonymity, untraceability, and key agreement. In the same year, authors in the paper [17] discussed a group key establishment protocol using PUFs.

In 2019, Gope et. al. [18] proposed an anonymous mutual authentication scheme. It is very efficient and has low computation costs. This is achieved by using PUF (physically unclonable function) while developing the scheme. Later, Banerjee et. al. [19] showed that this scheme was vulnerable to Ephemeral Secret Leakage (ESL) attack under Dolev-Yao threat model. Also, in the following year, Zeng et. al. [7] proposed an authentication scheme for multi server environments. Later, Mandal et. al. [4] show that the scheme is not only vulnerable to offline password guessing attack and privileged insider attack but also computationally expensive because it uses bilinear pairings which makes it impractical in real-world scenarios.

IV. Our Proposed PUFUAC-IOT Scheme

Our proposed PUFUAC-IoT scheme consists of four entities such as Users (U) and IoT devices (S_i), Gateway nodes (GW) and Trusted Authority (TA). It is assumed that some IoT devices will be connected with a GW to form a cell. The GW in a cell can connect with other gateway nodes, the TA and cloud servers. The registered users can get real-time data from the designated IoT devices in a particular cell. A summary of notations is described in Table I. Each user's device, the Gateway node and the IoT device consist of a micro controller attached to a PUF. There are four phases in our proposed scheme, namely: enrolment phase, user registration phase, login and access control phase, and physical-tamper checking of IoT devices.

Table I
Notations

Notations	Description
TA	Trusted Authority
U	A User in IOT environment
S_i	IoT Device
GW	A Gateway Node
ID_U	User U's identity
ID_{s_i}	Identity of i^th IoT Device
ID_{GW}	Identity of Gateway Node
$\oplus$	XOR operator
$H(.)$	Hash function
$pass_U$	U ś password
(C_i, R_i)	Challenge-response pair
γ_i	U ś biometric thumb impression
$\|\|$	Concatenation operator
$\phi(.)$	Physically uncloneable function
SK	Session Key
A	Attacker

A. Enrolment phase

In this phase, GW and S_i are enrolled with the TA as follows:

1. For registering (GW), the TA assigns a unique identity ID_{GW}, and also generates a challenge C_{GW} for the interaction with GW. Now to deal with DoS Attacks or the issue of desynchronization, TA also generates some set of challenges $C_{GW}^{syn} = \{c_1, c_2, ..., c_n\}$ for resynchronization with GW and then sends C_{GW}, C_{GW}^{syn} to GW through secure channel.
2. GW extracts PUF outputs $R_{GW} = \phi_{GW}(C_{GW})$, $R_{GW}^{syn} = \phi_{GW}(C_{GW}^{syn})$ and sends $\{R_{GW}, R_{GW}^{syn}\}$ to the TA.
3. A set of IoT Devices are assigned to a GW, collectively called as a cell. Now, to register an IoT device with a GW, the TA assigns a unique identity ID_{S_i}, and also generates a challenge C_{S_i} for the interaction with S_i. Now to deal with DoS Attacks or the issue of desynchronization, TA also generates some set of challenges

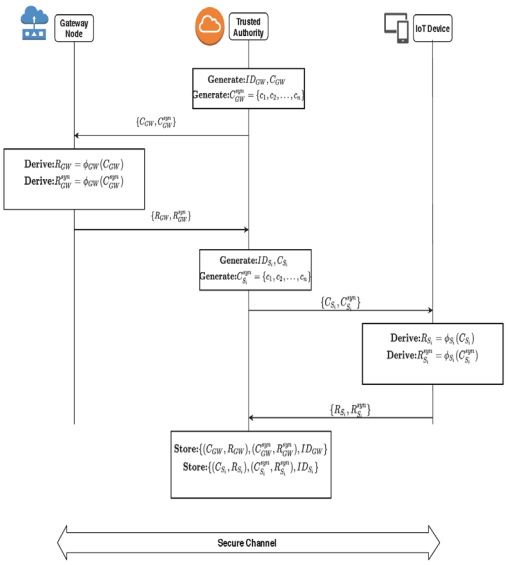

Figure 1

Enrolment phase of gateway and IoT devices

$C_{S_i}^{syn} = \{c_1, c_2, ..., c_n\}$ for resynchronization with S_i and then sends $C_{S_i}, C_{S_i}^{syn}$ to S_i through a secure channel.

4. Now, S_i computes PUF outputs $R_{S_i} = \phi_{S_i}(C_{S_i})$, $R_{S_i}^{syn} = \phi_{S_i}(C_{S_i}^{syn})$ and sends $R_{S_i}, R_{S_i}^{syn}$ to the TA.
5. After getting the Challenge Response Pair for GW and S_i, we store this information in TA along with ID_{GW} and ID_{S_i} respectively, and we also map the information of these IoT devices to the particular Gateway Node to which they belong to form a cell for further communication.

Figure 1 shows enrolment phase of gateway and IoT devices.

B. *User registration phase*

To access IoT device (S_i), U needs to register his/her trusted device with the TA. U follows the below steps for the registration.

1. User U chooses an identity ID_U and sends $\{ID_U, Reg_{req}\}$ to the Trusted Authority through a private channel, where Reg_{req} is the registration request.

2. Upon obtaining the registration message, the TA creates a random challenge C_U for normal authentication process. To address the issue of desynchronization or Denial-of-Service (DoS) attacks, the TA also generates a set of new challenges $C_U^{syn} = \{c_1, c_2, .., c_n\}$ for resynchronization with the user U and transmits the message $\{C_U, C_U^{syn}\}$ to U via a secure channel.
3. After obtaining $\{C_U, C_U^{syn}\}$, the U's trusted device extracts the PUF outputs $R_U = \phi_U(C_U)$, $R_U^{syn} = \phi_U(C_U^{syn})$ and sends $\{R_U, R_U^{syn}\}$ to the TA along with the GW with whom the user U is connected with the designated smart devices S_i.
4. Now, TA stores $\{ID_U, (C_U, R_U)(C_U^{syn}, R_U^{syn})\}$ for further communication with U and will send the challenge response pairs of the IoT devices and the users to the designated gateway nodes GW for further communication.
5. Now TA chooses a fuzzy verifier integer l such that $2^4 \le l \le 2^8$. Then, the gateway transmits l to the user along with the challenge response pair of gateway which consists of the IoT devices to be accessed by the user.

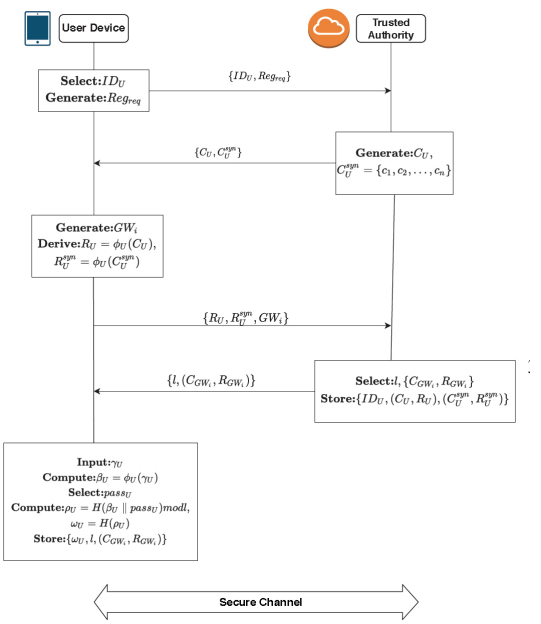

Figure 2

User registration phase

6. After receiving l, U enters his/her biometric thumb impression γ_U into the device. The device retrieves $\beta_U = \phi_U(\gamma_U)$ and then U chooses a password $pass_U$, and inputs $pass_U$ into the device. The device calculates $\rho_U = H(\beta_U \| pass_U) modl$ and stores $\omega_U = H(\rho_U)$ for user verification. Now, the device stores $\{\omega_U, l\}$ for user verification and will also store challenge response pair of Gateway Node for further communication.

User registration phase is summarized in Figure 2.

C. *Login and access control phase*

In this phase, U verifies himself to his registered GW and to S_i for which it belongs to the cell of GW. Then U will share a session key with the smart device S_i for further communications among U and S_i. This phase is described in Figure 3.

1. User U inputs γ_U and $pass_U$ in the device. After that, the device computes $\beta_U = \phi_U(\gamma_U)$, $\rho_U^* = H(\beta_U \| pass_U) modl$ and $\omega_U^* = H(\rho_U^*)$. Now, if $\omega_U^* = \omega_U$, a nonce n_U is created by the device, ID_U and ID_{S_i} for the User and IoT Device respectively. ID_{S_i} is the IoT device that

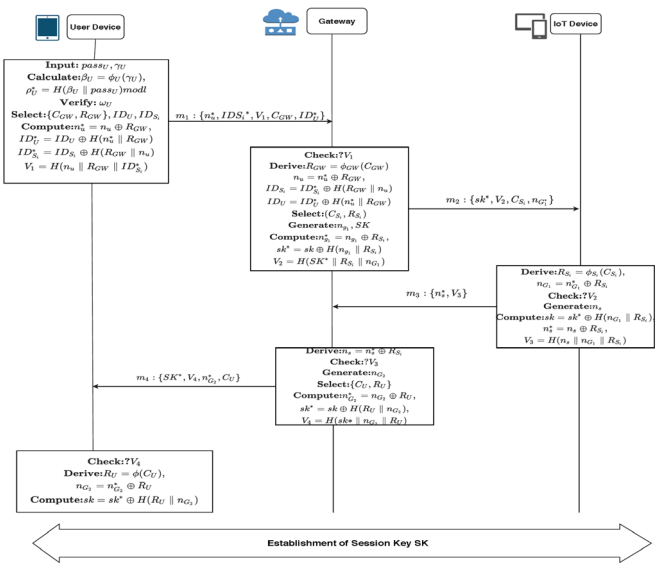

Figure 3

Login and access control phase

the user wants to access. After this U selects a challenge response pair of the gateway node (C_{GW}, R_{GW}). Now,

$$n_U^* = n_U \oplus R_{GW},$$
$$ID_U^* = ID_U \oplus H(n_{U^*} \| R_{GW}),$$
$$ID_{S_i}^* = ID_{S_i} \oplus H(R_{GW} \| n_U).$$

We calculate $V_1 = H(n_U \| R_{GW} \| ID_{S_i}^*)$ for verification purpose and then send this message $m_1 : \{n_U^*, ID_{S_i}^{*'}, V_1, C_{GW}, ID_U^*\}$ via public channel to GW : otherwise, terminate this login session.

2. Now, verify V_1 at the gateway side by first calculating $R_{GW} = \phi_{GW}(C_{GW})$ and then find $n_U = n_U^* \oplus R_{GW}$. If V_1 is verified, then the gateway finds the IoT Device to be accessed $(ID_{S_i}) = ID_{S_i}^* \oplus H(R_{GW} \| n_U)$. Now, gateway choses a challenge response pair for the particular IoT Device and also generates a session key (SK) and also a nonce (n_{G_1}) for establishing a connection between U and ID_{S_i}. . Let us say, the challenge response pair chosen for the IoT Device is (C_{S_i}, R_{S_i}). Now,

$$ID_U = ID_U^* \oplus H(n_{U^*} \| R_{GW}),$$
$$n_{G_1}^* = n_{G_1} \oplus R_{S_i},$$
$$SK^* = SK \oplus H(n_{G_1} \| R_{S_i}),$$
$$V_2 = H(SK^* \| R_{S_i} \| n_{G_1}).$$

V_2 is calculated for verification purpose and then gateway composes a message $m2 : \{SK^*, V_2, C_{S_i}, n_{G_1}^*\}$ and sends to the IoT Device to be accessed via public channel.

3. After receiving $m2$, IoT device starts calculating $R_{S_i} = \phi_{S_i}(C_{S_i})$ and using this R_{S_i} to calculate n_{G_1} by $n_{G_1} = n_{G_1}^* \oplus R_{S_i}$. Now, after calculating these values the sensor device will verify the value with V_2. If the value is verified, Sensor will now extract SK by $SK = SK^* \oplus H(n_{G_1} \| R_{S_i})$. Now, a nonce (n_S) is generated at the sensor side for confirming that device has received the Session Key securely. Now,

$$n_S^* = n_S \oplus R_{S_i},$$
$$V_3 = H(n_S \| n_{G_1} \| R_{S_i}).$$

V_3 is calculated for validating that the key has been received securely by the IoT device and now device composes a message $m3:\{n_S^*, V_3\}$ and sends to the GW.

4. Upon receiving the message $m3$, GW verifies V_3 with the help of R_{S_i} and then computing $n_S = n_S^* \oplus R_{S_i}$. If the verification is successful then it transmits SK to the U. Firstly, GW generates a nonce (n_{G_2}) and chooses CRP pair $\{C_U, R_U\}$ for the user with ID_U extracted at the starting of this phase. The gateway uses this nonce for further computation:

$$n_{G_2}^* = n_{G_2} \oplus R_U$$
$$SK^* = SK \oplus H(R_U \| n_{G_2}),$$
$$V_4 = H(SK^* \| n_{G_2} \| R_U).$$

V_4 is used for verification at the user side for getting the right SK. GW composes the message $m4:\{SK^*, V_4, n_{G_2}^*, C_U\}$..

5. After receiving $m4$, U extracts $R_U = \phi_U(C_U)$ and then calculates $n_{G_2} = n_{G_2}^* \oplus R_U$. Now, comes the verification part, if the verification is successful then using n_{G_2} user can derive $SK = SK^* \oplus H(R_U \| n_{G_2})$. Now, SK is obtained by the User and the session is established between the User and the IoT device successfully.

D. *Physical-tamper checking of IoT devices phase*

GW used to check periodically that the IoT Device S_i has been physically tampered or not. Both GW and S_i mutually authenticate each other as follows.

1. Firstly, Gateway (GW) generates a nonce n_g and then randomly selects a CRP (C_{S_i}, R_{S_i}). GW calculates $n_g^* = R_{S_i} \oplus n_g$, and ver_{GW} for verification purpose at the sensor side $ver_{GW} = H(R_{S_i} \| n_g^*)$.. Now, gateway sends the message $\{n_g^*, C_{S_i}, ver_{GW}\}$ to the S_i via an insecure channel.
2. Upon obtaining $\{n_g^*, C_{S_i}, ver_{GW}\}$, S_i extracts $R_{S_i} = \phi_{S_i}(C_{S_i})$ and then checks the validity of ver_{GW}. If it is valid, S_i creates a nonce n_s and computes n_g as follows:

$$n_s^* = n_s \oplus H(R_{S_i} \| n_g)$$

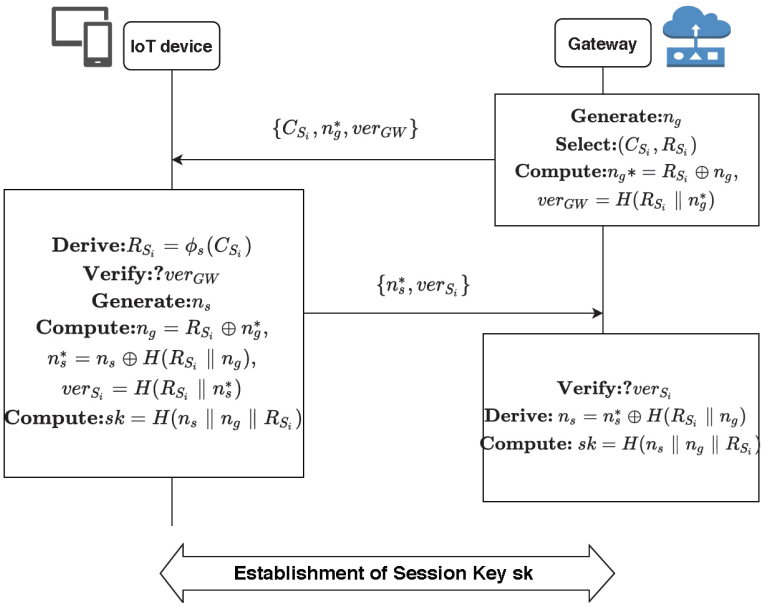

Figure 4

Physical-tamper checking of IoT devices phase

Using this, sensor node calculates session key $sk = H(n_s \| n_g \| R_{S_i})$ and $ver_{S_i} = H(R_{S_i} \| n_s^*)$ then composes a message $\{n_S^*, ver_{S_i}\}$ and transmits it to the GW via an open channel.

3. Now, GW receives a message from S_i and starts verifying the parameter ver_{S_i}. If it is valid, GW computes $n_s = n_s^* \oplus H(R_{S_i} \| n_g)$ and then calculates sk, using all the values. Now, sk is used to send its collected data securely to the gateway as a session is established between GW and S_i.

Figure 4 describes the physical-tamper checking of IoT devices phase.

V. Security analysis

The informal security analysis of our scheme is as follows:

1. **User anonymity**: Any scheme can guarantee user anonymity if it can be proved that any adversary cannot identify or map a particular user to the corresponding activity on the network. Hence in our scheme, we encrypt the user identity with a hash involving random

response from challenge response pair. Hence , the encrypted used id will be different for each login This holds the property of user anonymity in our proposed scheme.

2. **Mutual authentication**: Mutual authentication is done between gateway, user and sensor. First, Gateway GW and the user U mutually authenticate themselves by checking if V_1 matches $H(n_U \| R_{GW} \| ID_{S_i}^*)$. Later, gateway and sensor mutually authenticate themselves by checking if V_2 matches $H(SK^* \| R_{S_i} \| n_{G_1})$.
3. **User untraceability**: In our scheme, the user U_i's identity is assigned randomly and changes with every new session, it is not possible to know the identity of a user from the messages sent over an open channel. If a denial of service attack occurs and there is loss in synchronization, another set of challenge-response pairs are used to generate a new identity , whereby user untraceability is achieved.
4. **Denial of service attack**: In our scheme, some random challenges are generated and their corresponding responses are saved at the GW side. If an attacker changes one of the stored pairs, the gateway can still validate the user with the other available challenge and response pairs. Thus, denial of service attack is impossible in our scheme.
5. **Forward secrecy**: Let's assume that an adversary A gets PUF responses from user and sensor as R_U and R_{S_i}, Still A will not able to get the session key SK because of the updation of challenge response pairs after every session by TA. The updation of session key and challenge response pairs after every session ensure forward secrecy.
6. **User device loss attack**: If the U's device is stolen/lost, then an attacker gets the data stored in the device. Assuming A sends the message $m_1 : \{n_U^*, ID_{S_i}^*, V_1, C_{GW}, ID_U^*\}$ knowing the challenge and response keys of gateway, but he cannot decipher the session key in $m4 : \{SK^*, V_4, n_{G_2}^*, C_U\}$ because it is encrypted with the response given by the user device PUF which he cannot access. Hence, our scheme is resistant against user device loss attack.
7. **Physical attack**: The sensors are present in areas wherein any adversary can have physical access and tamper the devices. We have made the sensor nodes tamper resistant with the help of PUF, since the can be neither cloned nor tampered [20]. Tampering them will lead them to change their ideal behaviour , with which the sensor

node can no longer have access to the network making our protocol resistant to physical attacks.

VI. Performance analysis

Here we present the performance analysis of our PUFUAC-IoT scheme and the schemes [4]-[7]. in terms of communication cost, computation cost and security features. The security features of the proposed scheme and the other related schemes [4]-[7] are shown in Table II. From this table, it is clear that our scheme provides all the security features given in the table while others do not.

J_1: Replay attack, J_2: Man in the middle attack, J_3: Mutual authentication, J_4: Key agreement, J_5: Impersonation attacks, J_6: Mobile device physical capture attack, J_7: Anonymity and untraceability, J_8: ESL attack under the CK-adversary model, J_9: Signcryption-based user access control mechanism, J_{10}: Privileged-insider attack, J_{11}: Offline password guessing attack, J_{12}: Password/biometric change attack, J_{13}: Stolen/lost mobile device attack, No: scheme does not provide a feature, Yes: scheme provides a feature, NA: not applicable in a scheme

It is assumed that we take SHA-256 as a hash function. For the PUF operation, we consider the simulation of an 128-bit arbiter PUF circuit as

Table II
Functionality and Security

Scheme	[7]	[6]	[5]	[4]	PUFUAC-IoT
J_1	Yes	Yes	Yes	Yes	Yes
J_2	Yes	Yes	Yes	Yes	Yes
J_3	Yes	Yes	No	Yes	Yes
J_4	Yes	No	Yes	Yes	Yes
J_5	Yes	Yes	Yes	Yes	Yes
J_6	Yes	Yes	Yes	Yes	Yes
J_7	No	No	No	Yes	Yes
J_8	No	No	No	Yes	Yes
J_9	No	Yes	Yes	Yes	Yes
J_{10}	No	Yes	Yes	Yes	Yes
J_{11}	No	NA	NA	Yes	Yes
J_{12}	No	NA	NA	Yes	Yes
J_{13}	No	NA	NA	Yes	Yes

Table III
Communication Cost

Scheme	No. of messages	Total cost(in bits)
PUFUAC-IoT	4	2432
Mandal et. al. [4]	3	3136
Luo et. al. [5]	2	3040
Xue et. al. [6]	5	9344
Zeng et. al. [7]	2	2080

provided in [18]. The execution time of a hash operation (SHA-256) and PUF operation are taken as 1.37 ms and 0.43 ms, respectively.

For the analysis of communication cost, we calculate the communication cost during the login and access control phase. We take the size of random nonce and identity as 128 bits each. In our PUFUAC-IoT scheme, four messages $m_1 : \{n_U^*, ID_{S_i}^*, V_1, C_{GW}, ID_U^*\}$, $m2 : \{SK^*, V_2, C_{S_i}, n_{G_1}^*\}$, $m3 : \{n_S^*, V_3\}$, $m4 : \{SK^*, V_4, n_{G_2}, C_U\}$ demand $(128+128+256+128+128) = 768bits$, $(128+256+128+128) = 640bits$, $(128+256) = 384bits$, $(128+256+128+128) = 640bits$, which collectively requires $2432bits$. The communication costs of our scheme and the related schemes [4]-[7] are shown in Table III.

Now, we evaluate the computation costs of our scheme PUFUAC-IoT and the schemes [4]-[7] during the login and access control phase in Table IV. Let us assume that C_{SHA-1}, $C_{SHA-256}$, C_{eca}, C_{ecm}, C_{bp}, C_{me}, $C_{senc}/C_{sdec} (\approx C_{SHA-1})$, $C_{fe} (\approx C_{ecm})$ and C_p represent the time complexity of SHA-1, SHA-256, an elliptic curve point addition, an elliptic curve point scalar multiplication, a bilinear pairing, a modular exponentiation operation, a symmetric encryption/decryption, a fuzzy extractor function and a PUF operation, respectively. We take the same execution time (in milliseconds) of these cryptographic operations as in the scheme [4]. So we have $C_{SHA-1} (\approx 0.056ms)$, $C_{SHA-256} (\approx 1.37ms)$, $C_{ecm} (\approx 13.405ms)$, $C_{eca} (\approx 0.081ms)$, $C_{me} (\approx 2.249ms)$, $C_{bp} (\approx 32.713ms)$, $C_{fe} (\approx C_{ecm})$, $C_{senc}/C_{sdec} (\approx C_{SHA-1})$ and $C_P (\approx 0.43ms)$. Our scheme uses hash function and PUF operations, which are lightweight. It is evident from Table IV that the computation cost of our scheme is the least as compared to the related schemes [4]-[7] and also it is more secured than them.

Table IV

Computation Cost

Protocol	Total Cost	Approx time(in ms)
PUFUAC-IoT	$4C_P + 18C_{SHA-256}$	26.380
Mandal et. al. [4]	$C_{fe} + 14C_{ecm}$ $+8C_{eca} + 28C_{SHA-1}$	203.291
Luo et. al. [5]	$3C_{ecm} + 4C_{bp} + 4C_{SHA-1}$ $+C_{eca} + C_{me}$	173.621
Xue et. al. [6]	$6C_{ecm} + 2C_{eca} + 6C_{bp}$ $+3C_{me} + 7C_{SHA-1} + 6C_{senc} / C_{sdec}$	284.345
Zeng et. al. [7]	$9C_{SHA-1} + 2C_{ecm} + 4C_{eca}$ $+2C_{senc} / C_{sdec} + 2C_{bp} + 2C_{me}$	97.674

Conclusion

In this paper, we presented a new scheme for user access control in an IoT environment. Our proposed scheme is based on a recent user access control scheme by Mandal *et. al.*. We have used lightweight cryptographic primitives like PUF, one way cryptographic hash function and bitwise XOR operations. Due to this reason, our scheme is very lightweight and efficient as compared to other schemes of similar nature. We did the informal security analysis of the scheme. We have compared the performance analysis of the proposed scheme in terms of computation cost, communication cost and security features with the related schemes and shown that our scheme is more secured and has less computation cost than the related schemes.

References

[1] "The internet of things: a movement, not a market," tech. rep., HIS Markit (2017).

[2] M. Wazid, A. K. Das, R. Hussain, G. Succi, and J. J. Rodrigues, "Authentication in cloud-driven iot-based big data environment: urvey and outlook," *Journal of Systems Architecture*, vol. 97, pp. 185–196 (2019).

[3] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208 (1983).

[4] S. Mandal, B. Bera, A. K. Sutrala, A. K. Das, K.-K. R. Choo, and Y. Park, "Certificateless-signcryption-based three-factor user access control scheme for iot environment," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3184–3197 (2020).

[5] M. Luo, Y. Luo, Y. Wan, and Z. Wang, "Secure and efficient access control scheme for wireless sensor networks in the cross-domain context of the iot," *Security and Communication Networks*, vol. 2018 (2018).

[6] J. Xue, C. Xu, and Y. Zhang, "Private blockchain-based secure access control for smart home systems.," *KSII Transactions on Internet & Information Systems*, vol. 12, no. 12 (2018).

[7] X. Zeng, G. Xu, X. Zheng, Y. Xiang, and W. Zhou, "E-aua: An efficient anonymous user authentication protocol for mobile iot," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1506–1519 (2018).

[8] H. Yu, J. He, T. Zhang, P. Xiao, and Y. Zhang, "Enabling end-to-end secure communication between wireless sensor networks and the internet," *World Wide Web*, vol. 16, no. 4, pp. 515–540 (2013).

[9] F. Li, J. Hong, and A. A. Omala, "Efficient certificateless access control for industrial internet of things," *Future Generation Computer Systems*, vol. 76, pp. 285–292 (2017).

[10] D. Kumar, "A secure and efficient user authentication protocol for wireless sensor network," *Multimedia Tools and Applications*, vol. 80, no. 18, pp. 27131–27154 (2021).

[11] D. Kaur, K. K. Saini, and D. Kumar, "Cryptanalysis and enhancement of an authentication protocol for secure multimedia communications in iot-enabled wireless sensor networks," *Multimedia Tools and Applications,* pp. 1–19 (2022).

[12] P. Kumar, A. Gurtov, J. Iinatti, M. Sain, and P. Ha, "Access control protocol with node privacy in wireless sensor networks," *IEEE Sensors Journal*, vol. PP, pp. 1–1 (09 2016).

[13] C. Ma, K. Xue, and P. Hong, "Distributed access control with adaptive privacy preserving property for wireless sensor networks," *Security and Communication Networks*, vol. 7, no. 4, pp. 759–773 (2014).

[14] D. Q. Bala, S. Maity, and S. K. Jena, "Mutual authentication for iot smart environment using certificate-less public key cryptography,"

in 2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS), pp. 29–34, IEEE (2017).

[15] A. Karati, S. H. Islam, and G. Biswas, "A pairing-free and provably secure certificateless signature scheme," *Information Sciences,* vol. 450, pp. 378–391 (2018).

[16] N. Pakniat and B. A. Vanda, "Cryptanalysis and improvement of a pairing-free certificateless signature scheme," in 2018 15th International ISC (Iranian Society of Cryptology) Conference on Information Security and Cryptology (ISCISC), pp. 1–5, IEEE (2018).

[17] A. Robinson and R. Steinwandt, "Group key establishment with physical unclonable functions," *Journal of Information and Optimization Sciences*, vol. 40, no. 1, pp. 69–80 (2019).

[18] P. Gope, A. K. Das, N. Kumar, and Y. Cheng, "Lightweight and physically secure anonymous mutual authentication protocol for real-time data access in industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 4957–4968 (2019).

[19] S. Banerjee, V. Odelu, A. K. Das, S. Chattopadhyay, J. J. Rodrigues, and Y. Park, "Physically secure lightweight anonymous user authentication protocol for internet of things using physically unclonable functions," *IEEE Access*, vol. 7, pp. 85627–85644 (2019).

[20] P. Gope, J. Lee, and T. Q. Quek, "Lightweight and practical anonymous authentication protocol for rfid systems using physically unclonable functions," *IEEE Transactions on Information Forensics and Security,* vol. 13, no. 11, pp. 2831–2843 (2018).

Receivd November, 2021

Revised July, 2022