A Provably Secure and Lightweight Anonymous User Authenticated Session Key Exchange Scheme for Internet of Things Deployment

Soumya Banerjee, Vanga Odelu, Ashok Kumar Das, *Senior Member, IEEE*, Jangirala Srinivas, *Member, IEEE*, Neeraj Kumar, *Senior Member, IEEE*, Samiran Chattopadhyay, *Member, IEEE*, and Kim-Kwang Raymond Choo, *Senior Member, IEEE*

***Abstract*—With the ever increasing adoption rate of Internet-enabled devices [also known as Internet of Things (IoT) devices] in applications such as smart home, smart city, smart grid, and healthcare applications, we need to ensure the security and privacy of data and communications among these IoT devices and the underlying infrastructure. For example, an adversary can easily tamper with the information transmitted over a public channel, in the sense of modification, deletion, and fabrication of data-in-transit and data-in-storage. Time-critical IoT applications such as healthcare may demand the capability to support external parties (users) to securely access IoT data and services in real-time. This necessitates the design of a secure user authentication mechanism, which should also allow the user to achieve security and functionality features such as anonymity and un-traceability. In this paper, we propose a new lightweight anonymous user authenticated session key agreement scheme in the IoT environment. The proposed scheme uses three-factor authentication, namely a user's smart card, password, and personal biometric information. The proposed scheme does not require the storing of user specific information at the gateway node. We then demonstrate the proposed scheme's security using the broadly accepted real-or-random (ROR) model, Burrows–Abadi–Needham (BAN) logic, and automated validation of Internet security protocols and applications (AVISPAs) software simulation tool, as well as presenting an informal security analysis to demonstrate its other features. In addition, through our simulations, we demonstrate that the proposed scheme outperforms existing related user authentication schemes, in terms of its security and functionality features, and computation costs.**

***Index Terms*—Internet of Things (IoT), key agreement, security, session key, user authentication.**

I. Introduction

INTERNET of Things (IoT) has been a trend for the past few years, and it is likely to be so in the foreseeable future, as evidenced by studies such as [1]. Specifically, in an IoT system, data and information are being collected/sensed by IoT devices [e.g., radio frequency identification (RFID) devices, low powered IEEE 802.15.4 devices, embedded systems, and wearable devices] before being sent to another IoT device, intermediary device/node (e.g., edge or fog computing node), or the cloud, via the Internet. IoT applications include Industry 4.0 and those in high risk environments such as disaster relief and battlefields.

Security and privacy are two key concerns in any popular consumer technology deployment [2]. For example, let us consider an IoT healthcare application as shown in Fig. 1. In this scenario, by allowing a medical practitioner (i.e., external user) to have direct access to data sensed by the body sensor devices deployed in his/her patient's body, can enhance the quality of healthcare service. Such information could include current vital readings (blood sugar level, blood pressure, etc.). Based on such current information, necessary remedial actions can be decided upon. Clearly, these information are also private and confidential, and hence both user and accessed sensor node(s) require mutual authentication and session key establishment. Specifically, using the constructed session keys, both user and accessed sensor node(s) can then communicate securely among themselves, in order to facilitate the data/service access.

To achieve this goal, we develop a secure and lightweight user authentication and session key agreement scheme, designed to operate in an IoT environment (see Section IV). We then carry out a formal security analysis of the proposed scheme in the widely adapted real-or-random (ROR) model [4] to prove its session key security. The Burrows–Abadi–Needham (BAN)

Manuscript received March 7, 2019; revised April 20, 2019 and May 15, 2019; accepted June 13, 2019. Date of publication June 17, 2019; date of current version October 8, 2019. This work was supported in part by the Research Initiation Grant under Grant BITS/GAU/RIG/2019/H0626, and in part by the Outstanding Potential for Excellence in Research and Academics (OPERA) through the Birla Institute of Technology and Science (BITS) Pilani (Hyderabad Campus), India, under Award FR/SCM/070518/CSIS. The work of K.-K. R. Choo was supported in part by the Cloud Technology Endowed Professorship and in part by NSF CREST under Grant HRD-1736209. *(Corresponding author: Ashok Kumar Das.)*

S. Banerjee and S. Chattopadhyay are with the Department of Information Technology, Jadavpur University Salt Lake City, Kolkata 700 098, India (e-mail: soumyabanerjee@outlook.in; samirancju@gmail.com).

V. Odelu is with the Department of Computer Science and Information Systems, Birla Institute of Technology and Science, Pilani Hyderabad Campus, Hyderabad 500 078, India (e-mail: odelu.vanga@hyderabad.bits-pilani.ac.in).

A. K. Das is with the Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500 032, India (e-mail: iitkgp.akdas@gmail.com).

J. Srinivas is with the Jindal Global Business School, O. P. Jindal Global University, Haryana 131001, India (e-mail: getsrinunow1@gmail.com).

N. Kumar is with the Department of Computer Science and Engineering, Thapar University, Patiala 147 004, India (e-mail: neeraj.kumar@thapar.edu).

K.-K. R. Choo is with the Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX 78249 USA (e-mail: raymond.choo@fulbrightmail.org).

Digital Object Identifier 10.1109/JIOT.2019.2923373

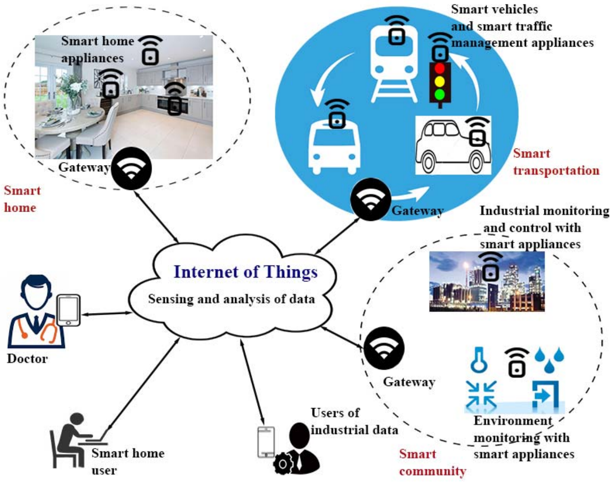

Smart home appliances
Gateway
Smart home
Smart vehicles and smart traffic management appliances
Smart transportation
Gateway
Industrial monitoring and control with smart appliances
Internet of Things
Sensing and analysis of data
Doctor
Gateway
Environment monitoring with smart appliances
Users of industrial data
Smart home user
Smart community

Fig. 1. Generalized IoT architecture (source: [3]).

logic based security proof [5] is also presented to show that the communicating parties achieve mutual authentication. In addition, an informal security analysis is performed to show that the proposed scheme is also secure against other common attacks. The simulation results using the popular formal security verification automated software tool, AVISPA [6], also assure us that replay and man-in-the-middle attacks are protected in the scheme. Both formal and informal security analysis are presented in Section V. A comparative study of the communication and computation costs, as well as the security and functionality features for the proposed scheme and other relevant authentication schemes, is presented in Section VI. Findings from the performance evaluation using NS3 simulator is presented next in Section VII. Section VIII concludes this paper.

II. Basic Preliminaries

The required mathematical background for understanding the proposed scheme is discussed in this section.

A. One-Way Hash Function

One-way hash functions are mathematical functions that have been extensively used in many applications, such as producing message authentication codes (MACs), detecting data integrity during transmission, and digital forensic investigations. Cryptographic one-way hash functions are by design highly sensitive to even small perturbations to the input. A "collision-resistant one-way hash function" is defined as follows [7].

Definition 1: Assume $h : \{0, 1\}^* \rightarrow \{0, 1\}^n$ denotes a one-way hash function, which is by nature deterministic. Specifically, upon receiving a variable length input, the function gives a fixed-size length output of n bits, say. The latter is called a message digest or a hash output. If $\text{Adv}_{\mathcal{A}}^{\text{HASH}}(rt)$ is defined as an adversary $\mathcal{A}$'s advantage in detecting a hash collision in the execution (run) time rt, then $\text{Adv}_{\mathcal{A}}^{\text{HASH}}(rt) = \Pr[(ip_1, ip_2) \in_R \mathcal{A} : ip_1 \neq ip_2 \text{ and } h(ip_1) = h(ip_2)]$, where $\Pr[X]$ means a random event X's probability and $(ip_1, ip_2) \in_R$ implies that both the input strings ip_1 and ip_2 are two randomly picked by $\mathcal{A}$. If an (ϕ, rt)-adversary $\mathcal{A}$ attempts to find a hash collision for $h(\cdot)$, it is understood that $\text{Adv}_{(A)}^{\text{HASH}}(rt) \leq \phi$ with the maximum execution time rt.

B. Fuzzy Extractor for Biometric Verification

For biometric verification, we choose the fuzzy extractor method [8]. Even if there is a slight variation inherent to the biometric capture mechanism, the fuzzy extractor procedure has the ability to identify a user based on his/her noisy biometric. The fuzzy extractor comprises a probabilistic generation procedure $\text{Gen}(\cdot)$, and a deterministic reproduction procedure $\text{Rep}(\cdot)$.

1) Gen: On a user U_i's biometric template, say BIO_i, $\text{Gen}(\cdot)$ outputs a pair having a biometric (secret) key σ_i of l bits, say and its corresponding public (reproduction) parameter τ_i, that is, $\text{Gen}(\text{BIO}_i = (\sigma_i, \tau_i)$.

2) Rep: Given a noisy biometric template BIO_i' of the user U_i, $\text{Rep}(\cdot)$ recovers the original biometric secret key σ_i with the help of public τ_i with the criteria that the Hamming distance between the original biometric template BIO_i and current biometric template BIO_i' does not exceed an error tolerance threshold value t. Thus, $\text{Rep}(\text{BIO}_i', \tau_i) = \sigma_i$.

One of the estimations on error tolerance threshold values provided by Cheon *et al.* [9] is as follows: If the Hamming distance between the original biometric template BIO_i and current biometric template BIO_i' is T and the number of bits in input biometric is n, we then have $t = (T/n)$.

C. Indistinguishably of Encryption Under Chosen Plaintext Attack

Formally, indistinguishably of encryption under chosen plaintext attack (IND-CPA) can be defined as follows [10], [11]. Assume SE/ME denotes the single or multiple intruder (eavesdropper), respectively, EO_{ek_1}, $\text{EO}_{ek_2}, \ldots, \text{EO}_{ek_N}$ are N distinct independent encryption oracles associated with the encryption keys $ek_1, ek_2, \ldots, ek_N$, respectively, and k is the security parameter.

Definition 2: Let $\text{Adv}_{\Omega,\text{SE}}^{\text{IND-CPA}}(k)$ and $\text{Adv}_{\Omega,\text{ME}}^{\text{IND-CPA}}(k)$ be the advantage functions of SE and ME in the security parameter k, respectively. Then, $\text{Adv}_{\Omega,\text{SE}}^{\text{IND-CPA}}(k) = |2\Pr[\text{SE} \leftarrow \text{EO}_{ek_1}; (b_0, b_1 \leftarrow_R \text{SE}); \alpha \leftarrow_R \{0, 1\}; \beta \leftarrow_R \text{EO}_{ek_1}(b_\alpha) : \text{SE}(\beta) = \alpha] - 1|$, and $\text{Adv}_{\Omega,\text{ME}}^{\text{IND-CPA}}(k) = |2\Pr[\text{ME} \leftarrow \text{EO}_{ek_1}, \ldots, \text{EO}_{ek_N}; (b_0, b_1 \leftarrow_R \text{ME}); \alpha \leftarrow_R \{0, 1\};$ and $\beta_1 \leftarrow_R \text{EO}_{ek_1}(b_\alpha), \ldots, \beta_N \leftarrow_R \text{EO}_{ek_N}(b_\alpha) : \text{ME}(\beta_1, \ldots, \beta_N) = \alpha] - 1|$, where Ω is the encryption scheme. Ω is IND-CPA secure in single/multiple intruder setting if $\text{Adv}_{\Omega,\text{SE}}^{\text{IND-CPA}}(k)$ ($\text{Adv}_{\Omega,\text{ME}}^{\text{IND-CPA}}(k)$) is negligible (in k) for any probabilistic polynomial time SE (ME).

The same message, when it is encrypted twice, is produced with the same ciphertext for any deterministic encryption algorithm, and as a result, it is not IND-CPA secure scheme [7], [12]. In this paper, we apply the stateless cipher block chaining (CBC) mode of advanced encryption standard (AES-128) symmetric encryption scheme [13] to achieve our IND-CPA secure user authentication scheme. To incorporate this property, initialization vector (IV) in CBC requires to be made random for each message during the transmission when encryption happens [7].

D. Network and Threat Models

1) Network Model: We adopt the network model presented in [3] for the proposed scheme (see Fig. 1). The distinct scenarios, such as transport, smart home, national and community, consist of multiple IoT smart devices (SDs) operating as sensors and actuators. The SDs are linked to the public Internet via their respective gateway nodes (GWNs). Authorized users, prior to accessing their relevant SD, need to be registered with their corresponding GWN. The registered mobile users (MUs) can mutually authenticate with an accessed SD through GWN, in order to negotiate a session key for accessing the device data.

2) Threat Model: We consider a more realistic threat model that is recently described for IoT security in [14]. In our authentication scheme, the broadly accepted Dolev–Yao (DY) [15] threat model is applied in the proposed scheme in which an adversary $\mathcal{A}$ will have complete control over the communication channel. Consequently, $\mathcal{A}$ can eavesdrop, alter, delete and insert forgery messages during communication. In addition, the end-point entities (IoT sensor nodes and user) cannot be trusted in general.

The "CK-adversary model" [16] is widely regarded as the "current *de facto* standard model in modeling key-exchange protocols." Using the CK-adversary model, the adversary $\mathcal{A}$ can "deliver messages (as in the DY model)," and in addition, $\mathcal{A}$ can also "compromise other information, such as session state, private keys, and session keys." Therefore, it is important that "the leakage of some forms of secret information, such as session ephemeral (short-term) secrets or session key should have the least possible effect on the security of other secret credentials of the communicating entities in an authenticated key-exchange protocol [17]."

It is presumed that $\mathcal{A}$ can physically capture some IoT smart devices (SD_j) and then extract all the sensitive information stored in their memory. Furthermore, $\mathcal{A}$ can extract the sensitive credentials from a lost or stolen smart card of a user through power analysis attacks [18]. In addition, we also presume the GWNs are physically secured by placing them locking system. This will make the physical capture of the GWN much difficult when it is compared with the case of physical capture of the SDs [7]. The GWNs are considered as trusted entities in the IoT environment.

We also use the following assumptions as stated in Amin *et al.*'s [19] scheme. The registered legitimate users always use the words as passwords and identities from the dictionary available to the adversary $\mathcal{A}$ in password-based user authentication protocols. The password and identity of a legitimate user can be individually guessed by $\mathcal{A}$. However, guessing both password and identity of a registered user and then verifying those in polynomial time is a computationally expensive task for $\mathcal{A}$, if the right procedures are adopted (e.g., by not choosing an easy-to-guess password and identity pair). Furthermore, it is also computationally expensive for $\mathcal{A}$ to guess the secret keys and random numbers (nonces) in polynomial time as these are high entropy entities.

In the next section, we will review the related literature.

III. Related Work

The "general security requirements" needed to secure an IoT network are similar to those in other networks, such as "wireless sensor networks (WSNs)," namely authentication, integrity, confidentiality, availability, nonrepudiation, authorization, freshness, and forward and backward secrecy. Based on these security requirements, a user authentication protocol designed for an IoT environment need to be shown to be resilience to attacks such as replay, man-in-the-middle, stolen/lost smart card, online/offline guessing, password change, privileged-insider, and resilience against sensing device capture. The functionality features of a user authentication protocol designed for an IoT network should also include reduced communication and computation costs, password/biometric update phase, user revocation phase, and dynamic sensing device addition phase. The "password/biometric update phase" should allow a user to update his/her password/biometrics locally without further involvement of GWN. The "dynamic sensing device addition phase" is needed as some IoT devices may be physically compromised by an attacker or some devices may be drained of their battery power as they are resource limited, and we need to place additional sensor devices in the network after initial deployment of the nodes.

Assume a scenario where an MU (e.g., a medical practitioner) is roaming in the medical IoT environment. In such a setting, we may wish to safeguard certain information about the user. For example, by achieving anonymity preservation, we prevent other parties from linking the user with the messages to/from him/her or with the sessions in which he/she joins. This is because any unauthorized individuals (e.g., adversary) can attempt to track an MU's current location and location history if the user's identity is disclosed. Clearly, that is privacy implication as well as the potential for physical harm (e.g., physical stalking by patients or their family). In other words, the anonymity of a user is one of several key features in user authentication protocol [20]. For untraceability, an attacker must not follow the trace of a communicating party (e.g., a user) when the user (or device) moves from one communicating party or location to another. This property is also important in the IoT applications so that an attacker cannot trace a user during a session [14]. There have been other studies in the literature, relating to various requirements for remote user authentication in distributed systems, such as user anonymity, privacy, untraceability, liability, and trust [21]–[25]. More recently in 2018, Makhdoom *et al.* [26] identified "user anonymity vis-a-vis id management" as one of the key security and privacy challenges. Thus, it is imperative that the user authentication schemes designed for IoT systems should provide "user anonymity and untraceability" properties.

There have also been a large number of such protocols designed in the literature in the last decade. For example, Zhang *et al.* [27] designed a new authentication protocol, which preserves user privacy and uses only lightweight cryptographic primitives. However, their scheme fails to ensure user anonymity. Chang and Le [28] presented two authentication schemes. The first scheme only utilizes bitwise XOR and hash operations, whereas the second scheme uses additional elliptic curve cryptographic operations. Their first lightweight scheme is subsequently shown to be insecure against session key breach attack. In addition, both schemes are vulnerable to session specific information leakage and offline password guessing attacks [29].

Li *et al.* [30], [31] designed an improved authentication and data encryption mechanism for IoT-based medical care system, as well as an authentication protocol for RFID-based IoT systems. Khalil *et al.* [32] presented a test-bed, where sensors were utilized for controlling devices in a smart building. Porambage *et al.* [33] presented an authentication protocol, where sensors and end-users could mutually authenticate each other to establish a secure connection. Their protocol works in two phases, and it is suitable for deployment on heterogeneous resource limited nodes and it is also scalable with the network size. However, their scheme fails to preserve user anonymity, as demonstrated by Wazid *et al.* [34]. Turkanović *et al.* [35] designed a computationally efficient authentication scheme, but their scheme does not achieve untraceability and fails to safeguard against offline password guessing, privileged inside and impersonation attacks.

Jie *et al.* [36] proposed a multilayer architecture for securing smart homes. Song *et al.* [37], however, observed that the certificate authority in [36] places a large computational overhead on the SDs. They mitigated the limitation by presenting two authentication schemes: 1) the first utilizes hash functions and 2) the other utilizes chaotic systems. In 2017, Challa *et al.* [3] designed an elliptic curve cryptography (ECC) signature-based authentication and key agreement protocol for IoT deployment. However, the protocol has a high computational overhead due to the use of ECC cryptographic operations.

Amin *et al.* [38] designed a user authentication protocol in a distributed cloud computing environment, comprising of IoT devices. However, it was shown that their scheme has several security pitfalls, such as insecurity against privileged-insider attack and impersonation attack [39]. In addition, it was also shown that in their scheme there is neither user anonymity nor forward secrecy [40]. Dhillon and Kalra [41] designed a multifactor remote user authentication scheme for IoT environment, but their scheme fails to preserve untraceability or user anonymity properties. Chuang *et al.* [42] classified continuous authentication protocols into two categories, namely user-to-device models and device-to-device models. Then, they presented a lightweight continuous authentication protocol, but their scheme does not preserve sensing device anonymity or untraceability. A detailed survey on various authentication protocols, including user authentication for IoT setting, is available in [14] and [43].

In summary, most user authentication schemes either fail to satisfy the security requirements for IoT environment or they lack desirable functionality features (e.g., dynamic IoT sensing device addition, biometric and password change procedures, and anonymity and untraceability properties). To address this gap, we focus on designing a new lightweight user authentication protocol suited for IoT architecture, which will also achieve anonymity and untraceability.

TABLE I
SUMMARY OF NOTATIONS

Symbol	Description
U_i, SC_i	i-th user and his/her smart card
SD_j, GWN	j-th IoT smart device and the gateway node in a particular application
ID_X	X's identity
LTK_X	Long term key between an entity X and GWN
$Gen(\cdot), Rep(\cdot)$	Fuzzy extractor probabilistic generation & reproduction functions, respectively
PW_i, BIO_i	U_i's personal biometrics & password, respectively
σ_i, τ_i	Secret biometric key & public reproduction parameter corresponding to BIO_i
t	Error tolerance threshold applied in $Rep(\cdot)$
$h(\cdot)$	Collision-resistant cryptographic one-way hash function
$E[\cdot]_k/D[\cdot]_k$	Symmetric encryption/ decryption under the key k
TS_i, TS_{gwn}, TS_j	Current timestamps of U_i, GWN and SD_j, respectively
ΔTD	Maximum allowable transmission delay
ΔT_L	Lifetime of EID_i
$\Vert, \oplus$	String concatenation and bitwise exclusive (XOR) operations, respectively

IV. PROPOSED SCHEME

We will now discuss our lightweight anonymous user authentication scheme, using the network model presented in Section II-D1. We also remark that the proposed scheme is designed to be sufficiently generic for most IoT applications requiring user authentication.

A summary of the notations used in this paper is presented in Table I. To ensure resilience to replay attacks, current timestamps are utilized. Thus, the clocks of all involved entities are assumed to be synchronized. This is a typical assumption in the literature, such as the schemes presented in [7], [28], and [34].

The proposed scheme has four stages, namely 1) setup; 2) registration; 3) operation; and 4) maintenance (see Sections IV-A–IV-D). In the setup phase, the public parameters of the scheme are chosen by the trusted GWN. Once the setup process is completed, the IoT sensing devices can be enrolled and users can then be registered in the system. Both device enrollment and user registration can be performed dynamically at any time. A registered user can login anonymously in order to authenticate himself/herself in order to securely establish a session key with some designated IoT sensing devices for accessing real-time data. The proposed scheme enables the users to update his/her password and/or biometric information locally with the help of the smart card without further involving GWN. In addition, the proposed scheme also provides a mechanism for smart card revocation.

A. Setup Phase

During the setup phase, the public parameters are selected by the GWN. Specifically, GWN selects a one-way cryptographic hash function $h(\cdot)$, probabilistic generation function $\mathrm{Gen}(\cdot)$ and public reproduction function $\mathrm{Rep}(\cdot)$ for biometric fuzzy extractor, and symmetric cipher Ω containing encryption and decryption algorithms $E[\cdot]_{\mathrm{key}}$ and $D[\cdot]_{\mathrm{key}}$ with the symmetric key, say key, and declares these as public. As in [7], the stateless CBC mode of AES-128 symmetric encryption scheme is applied in order to make the proposed scheme IND-CPA secure. Furthermore, GWN also selects a long-term secret (LTS), which is only known to GWN.

B. IoT Smart Device Enrollment Phase

IoT SDs can be dynamically enrolled into the system at any time after the setup phase. The steps required to enroll an SD_j under the proposed scheme are given below.

Step 1: GWN selects a unique identity ID_j for each SD_j, generates a random number r_j, and then calculates $\mathrm{LTK}_j = h(\mathrm{LTS} \oplus h(\mathrm{ID}_j \| r_j))$.

Step 2: ID_j and LTK_j are loaded into the SD_j's memory before it is deployed in the IoT environment.

Step 3: GWN lists ID_j among its list of available devices.

C. User Registration Phase

In a cloud-based IoT (also referred to as Cloud-of-Things in the literature) system, there are several cloud servers and gateways. A user U_i may be required to register with some specific gateway(s) in order to access the services from the participating IoT devices. In practical applications (e.g., healthcare), the user only needs to register with a particular gateway to access the associated application. For accessing other services from the sensing devices located in other gateways [foreign gateway(s)], the user needs to access the sensing device from its home registered gateway and in that case, the home gateway needs to coordinate with other gateways for forwarding the user request [44]. This is similar to the roaming concept for MUs when they travel internationally. However, in this paper, we assume that U_i can register with its home GWN in order to acquire the services from an SD_j. The steps for U_i's registration under the proposed scheme are described below.

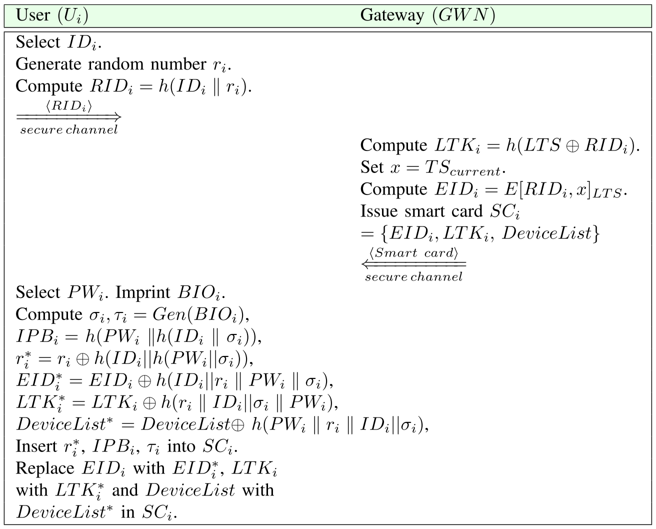
Step 1: U_i picks up an identity ID_i, generates a random number r_i, derives the pseudo-identity $\mathrm{RID}_i = h(\mathrm{ID}_i \| r_i)$, and securely sends it to the GWN.

Step 2: GWN calculates the shared key $\mathrm{LTK}_i = h(\mathrm{LTS} \oplus \mathrm{RID}_i)$, sets x as the current timestamp $\mathrm{TS}_{\mathrm{current}}$, encrypts RID_i and x using the key LTS as $\mathrm{EID}_i = E[\mathrm{RID}_i, x]_{\mathrm{LTS}}$ to be the dynamic identity, and securely issues a smart card SC_i containing the credentials $\{\mathrm{EID}_i, \mathrm{LTK}_i, \mathit{DeviceList}\}$, where *DeviceList* contains the identities of the SDs that U_i is authorized to access.

Step 3: Upon receiving SC_i, U_i chooses a password PW_i and imprints his/her biometric BIO_i into a particular terminal's sensor. U_i then calculates σ_i and τ_i using the fuzzy extractor generator function $\mathrm{Gen}(\cdot)$ as $(\sigma_i, \ \tau_i) = \mathrm{Gen}(\mathrm{BIO}_i)$, and the identity verification token $\mathrm{IPB}_i = h(\mathrm{PW}_i \| h(\mathrm{ID}_i \| \sigma_i))$. U_i also calculates $r_i^* = r_i \oplus h(\mathrm{ID}_i \| h(\mathrm{PW}_i \| \sigma_i))$ and saves r_i^*, IPB_i and τ_i into SC_i.

Step 4: SC_i finally replaces EID_i, LTK_i and *DeviceList* with $\mathrm{EID}_i^* = \mathrm{EID}_i \oplus h(\mathrm{ID}_i \| r_i \| \mathrm{PW}_i \| \sigma_i)$, $\mathrm{LTK}_i^* = \mathrm{LTK}_i \oplus h(r_i \| \mathrm{ID}_i \| \sigma_i \| \mathrm{PW}_i)$ and $\mathit{DeviceList}^* = \mathit{DeviceList} \oplus h(\mathrm{PW}_i \| r_i \| \mathrm{ID}_i \| \sigma_i)$, respectively, in its memory.

All the issued dynamic identities have a fixed lifetime, ΔT_L. The dynamic identities are single use, and these are updated for every successful authentication. If U_i fails to update his/her dynamic identity before it lapses, then his/her access to the system is revoked and a reregistration procedure is required. Under the proposed scheme, GWN is completely stateless with respect to registered users. Due to this property, a large number of users can be simultaneously registered with GWN. Fig. 2 summarizes the steps for user registration.


User (U_i) Gateway (GWN)

Select ID_i.
Generate random number r_i.
Compute $RID_i = h(ID_i \parallel r_i)$.

$\xRightarrow[secure\,channel]{\langle RID_i \rangle}$

Compute $LTK_i = h(LTS \oplus RID_i)$.
Set $x = TS_{current}$.
Compute $EID_i = E[RID_i, x]_{LTS}$.
Issue smart card SC_i
$= \{EID_i, LTK_i, \ DeviceList\}$

$\xLeftarrow[secure\,channel]{\langle Smart \ \ card \rangle}$

Select PW_i. Imprint BIO_i.
Compute $\sigma_i, \tau_i = Gen(BIO_i)$,
$IPB_i = h(PW_i \ \| h(ID_i \parallel \sigma_i))$,
$r_i^* = r_i \oplus h(ID_i || h(PW_i || \sigma_i))$,
$EID_i^* = EID_i \oplus h(ID_i || r_i \parallel PW_i \parallel \sigma_i)$,
$LTK_i^* = LTK_i \oplus h(r_i \parallel ID_i || \sigma_i \parallel PW_i)$,
$DeviceList^* = DeviceList \oplus \ h(PW_i \parallel r_i \parallel ID_i || \sigma_i)$,
Insert r_i^*, IPB_i, τ_i into SC_i.
Replace EID_i with EID_i^*, LTK_i
with LTK_i^* and $DeviceList$ with
$DeviceList^*$ in SC_i.

Fig. 2. User registration phase.

D. Login and User Authentication Phase

In order to access services from the SDs, a registered user U_i must login and authenticate with the accessed SDs. After this phase, both U_i and an accessed SD_j will negotiate a session key for secure communication between them. The following steps are essential under the proposed scheme.

Step 1: U_i supplies his/her identity ID_i and password PW_i, and also imprints BIO_i. SC_i of U_i then calculates $\sigma_i = \mathrm{Rep}(\mathrm{BIO}_i, \ \tau_i)$ with the restriction that the Hamming distance between current biometrics and registered biometrics in Section IV-C does not exceed t (error tolerance threshold value), $\mathrm{IPB}_i' = h(\mathrm{PW}_i \| h(\mathrm{ID}_i \| \sigma_i))$. Only if the calculated IPB_i' matches the stored IPB_i in SC_i will the login be considered successful. This indicates that U_i is valid and U_i has supplied all correct ID_i, PW_i and BIO_i. SC_i then calculates $r_i = r_i^* \oplus h(\mathrm{ID}_i \| h(\mathrm{PW}_i \| \sigma_i))$ and recovers the values of EID_i, LTK_i and *DeviceList* as $\mathrm{EID}_i = \mathrm{EID}_i^* \oplus h(\mathrm{ID}_i \| r_i \| \mathrm{PW}_i \| \sigma_i)$, $\mathrm{LTK}_i = \mathrm{LTK}_i^* \oplus h(r_i \| \mathrm{ID}_i \| \sigma_i \| \mathrm{PW}_i)$ and $\mathit{DeviceList} = \mathit{DeviceList}^* \oplus h(\mathrm{PW}_i \| r_i \| \mathrm{ID}_i \| \sigma_i)$. After selecting the identity ID_j of the accessed IoT SD_j from *DeviceList*, U_i calculates $\mathrm{EID}_j = E[\mathrm{ID}_j \| \mathrm{TS}_i]_{\mathrm{LTK}_i}$ by setting the IV value of CBC mode of AES-128 as $\mathrm{IV} = h(\mathrm{LTK}_i \| \mathrm{TS}_i)$. The login request message $M_1 = \{\mathrm{EID}_i, \mathrm{EID}_j, \mathrm{TS}_i\}$ is then sent to GWN through an open channel.

Step 2: Upon receiving M_1, GWN first inspects the attached timestamp TS_i's freshness by $|\mathrm{TS}_{\mathrm{gwn}} - \mathrm{TS}_i| \le \Delta\mathrm{TD}$, where the received time of the message M_1 is $\mathrm{TS}_{\mathrm{gwn}} = \mathrm{TS}_{\mathrm{current}}$, the current timestamp of GWN and $\Delta\mathrm{TD}$ is the maximum allowable transmission delay. If it is satisfied, then GWN decrypts EID_i with the key LTS to retrieve RID_i and x. If x is equal to $h(\mathrm{LTK} \| \mathrm{RID}_i)$ or $x - \mathrm{TS}_i > \Delta T_L$, then U_i's access has been revoked and the process is aborted here. Otherwise,

LTK_i is calculated as $h(\text{LTS} \oplus \text{RID}_i)$ and EID_j is decrypted with LTK_i and $\text{IV} = h(\text{LTK}_i \| \text{TS}_i)$ to recover ID_j and TS'_i as $(\text{ID}_j, \text{TS}'_i) = D[\text{EID}_j]_{\text{LTK}_i}$. Only when TS'_i matches TS_i will $\text{EID}'_i = E[\text{RID}_i,\ x']_{\text{LTS}}$ be calculated; otherwise, the process is aborted. Under normal operation, $x' = \text{TS}_{\text{gwn}}$. However, if U_i needs to be revoked, $x' = h(\text{LTK} \| \text{RID}_i)$ is calculated. GWN then generates a fresh random nonce x_i, calculates $X_i = h(\text{TS}_{\text{gwn}} \| x_i)$, $\text{auth} = h(\text{LTK}_i \| X_i \| \text{RID}_i)$, and checks if the access to the list of SDs maintained by U_i has changed (in case of dynamic device addition). If it is not so, $\text{Dev}' = \emptyset$ is set; otherwise, the change is saved to Dev'. GWN then looks up LTK_j from its database with ID_j, calculates $D_1 = E[\text{EID}'_i,\ X_i,\ \text{Dev}']_{\text{LTK}_i}$ and $D_2 = E[\text{auth},\ D_1,\ \text{TS}_{\text{gwn}}]_{\text{LTK}_j}$ under the stateless CBC mode of AES-128 by setting the IV values as $h(\text{LTK}_i \| \text{EID}_i \| \text{TS}_i)$ and $h(\text{LTK}_j \| \text{TS}_{\text{gwn}})$, respectively. After these calculations, the authentication request message $M_2 = \{D_2,\ \text{TS}_{\text{gwn}}\}$ is sent to the accessed IoT SD_j through an open channel.

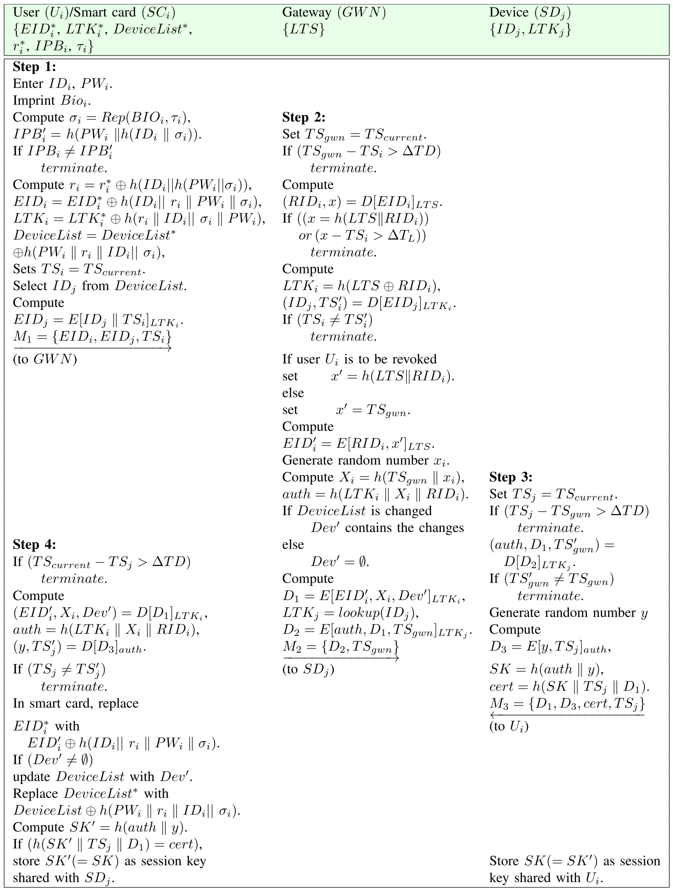
Step 3: When the message M_2 is received, SD_j examines the freshness of TS_{gwn} by the condition $|\text{TS}_j - \text{TS}_{\text{gwn}}| \le \Delta\text{TD}$, where TS_j is SD_j's current timestamp. If it is satisfied, SD_j decrypts D_2 to get auth, D_1 and TS'_{gwn} with key LTK_j and setting the IV in the stateless CBC of AES-128 as $h(\text{LTK}_j \| \text{TS}_{\text{gwn}})$. Only when TS'_{gwn} equates TS_{gwn} will SD_j generate a random number y; otherwise, the process is aborted. SD_j then calculates $D_3 = E[y, \text{TS}_j]_{\text{auth}}$ by setting $h(\text{auth} \| \text{TS}_j)$ as the IV, session key $\text{SK} = h(\text{auth} \| y)$ and $\text{cert} = h(\text{SK} \| \text{TS}_j \| D_1)$. After that, the authentication replies with message $M_3 = \{D_1, D_3, \text{cert}, \text{TS}_j\}$, which is sent to U_i through an open channel.

Step 4: Upon receiving the final message M_3, U_i examines the freshness of TS_j by the condition $|\text{TS}_{\text{current}} - \text{TS}_j| \le \Delta\text{TD}$. If it is satisfied, then U_i decrypts D_1 to obtain EID'_i, X_i and Dev' with the key LTK_i and $\text{IV} = h(\text{LTK}_i \| \text{EID}_i \| \text{TS}_i)$, calculates $\text{auth} = h(\text{LTK}_i \| X_i \| RID_i)$ and decrypts D_3 to retrieve y and TS'_j with the key auth and $h(\text{auth} \| \text{TS}_j)$ as the IV. If TS'_j is not equal to TS_j, the process is then aborted. Otherwise, U_i updates EID^*_i in SC_i with $\text{EID}'_i \ \oplus h(\text{ID}_i \| r_i \| \text{PW}_i \| \sigma_i)$. Moreover, if $\text{Dev}' \neq \emptyset$, then SC_i updates *DeviceList* with Dev' and replaces *DeviceList** with *DeviceList* $\oplus h(\text{PW}_i \| r_i \| \text{ID}_i \| \sigma_i)$. Finally, SC_i calculates $\text{SK}' = h(\text{auth} \| y)$ and checks if $h(\text{SK}' \| \text{TS}_j \| D_1) = \text{cert}$. If the criteria is satisfied, then U_i keeps the session key $\text{SK}'(= \text{SK})$ to establish a secure communication with SD_j. Similarly, SD_j stores the session key $\text{SK}(= \text{SK}')$ to establish a secure communication with U_i.

Fig. 3 summarizes the login and user authentication procedure. The above described steps also incorporate the mechanism for user revocation, as well as notification for the availability of new dynamically added SDs in the IoT environment.

E. Password and Biometric Update Phase

In this section, we describe the process for updating biometric and password of a legitimate registered user U_i in the proposed scheme. This process is executed locally without further communication with GWN, as described in the following steps.



User (U_i)/Smart card (SC_i)
$\{EID^*_i,\ LTK^*_i,\ DeviceList^*,$
$r^*_i,\ IPB_i,\ \tau_i\}$

Gateway (GWN)
$\{LTS\}$

Device (SD_j)
$\{ID_j, LTK_j\}$

Step 1:
Enter $ID_i,\ PW_i$.
Imprint Bio_i.
Compute $\sigma_i = Rep(BIO_i, \tau_i)$,
$IPB'_i = h(PW_i \ \| h(ID_i \ \| \ \sigma_i))$.
If $IPB_i \neq IPB'_i$
    $terminate.$
Compute $r_i = r^*_i \oplus h(ID_i || h(PW_i || \sigma_i))$,
$EID_i = EID^*_i \oplus h(ID_i || \ r_i \ \| \ PW_i \ \| \ \sigma_i)$,
$LTK_i = LTK^*_i \oplus h(r_i \ \| \ ID_i || \ \sigma_i \ \| \ PW_i)$,
$DeviceList = DeviceList^*$
$\oplus h(PW_i \ \| \ r_i \ \| \ ID_i || \ \sigma_i)$,
Sets $TS_i = TS_{current}$.
Select ID_j from $DeviceList$.
Compute
$EID_j = E[ID_j \ \| \ TS_i]_{LTK_i}$.
$\xrightarrow{M_1 = \{EID_i, EID_j, TS_i\}}$
(to GWN)

Step 2:
Set $TS_{gwn} = TS_{current}$.
If $(TS_{gwn} - TS_i > \Delta TD)$
    $terminate.$
Compute
$(RID_i, x) = D[EID_i]_{LTS}$.
If $((x = h(LTS \| RID_i))$
  $or\ (x - TS_i > \Delta T_L))$
    $terminate.$
Compute
$LTK_i = h(LTS \oplus RID_i)$,
$(ID_j, TS'_i) = D[EID_j]_{LTK_i}$.
If $(TS_i \neq TS'_i)$
    $terminate.$
If user U_i is to be revoked
set $x' = h(LTS \| RID_i)$.
else
set $x' = TS_{gwn}$.
Compute
$EID'_i = E[RID_i, x']_{LTS}$.
Generate random number x_i.
Compute $X_i = h(TS_{gwn} \ \| \ x_i)$,
$auth = h(LTK_i \ \| \ X_i \ \| \ RID_i)$.
If $DeviceList$ is changed
    Dev' contains the changes
else
    $Dev' = \emptyset$.
Compute
$D_1 = E[EID'_i, X_i, Dev']_{LTK_i}$,
$LTK_j = lookup(ID_j)$,
$D_2 = E[auth, D_1, TS_{gwn}]_{LTK_j}$.
$\xrightarrow{M_2 = \{D_2, TS_{gwn}\}}$
(to SD_j)

Step 3:
Set $TS_j = TS_{current}$.
If $(TS_j - TS_{gwn} > \Delta TD)$
    $terminate.$
$(auth, D_1, TS'_{gwn}) =$
  $D[D_2]_{LTK_j}$.
If $(TS'_{gwn} \neq TS_{gwn})$
    $terminate.$
Generate random number y
Compute
$D_3 = E[y, TS_j]_{auth}$,
$SK = h(auth \ \| \ y)$,
$cert = h(SK \ \| \ TS_j \ \| \ D_1)$.
$\xleftarrow{M_3 = \{D_1, D_3, cert, TS_j\}}$
(to U_i)

Step 4:
If $(TS_{current} - TS_j > \Delta TD)$
    $terminate.$
Compute
$(EID'_i, X_i, Dev') = D[D_1]_{LTK_i}$,
$auth = h(LTK_i \ \| \ X_i \ \| \ RID_i)$,
$(y, TS'_j) = D[D_3]_{auth}$.
If $(TS_j \neq TS'_j)$
    $terminate.$
In smart card, replace
EID^*_i with
  $EID'_i \oplus h(ID_i || \ r_i \ \| \ PW_i \ \| \ \sigma_i)$.
If $(Dev' \neq \emptyset)$
update $DeviceList$ with Dev'.
Replace $DeviceList^*$ with
$DeviceList \oplus h(PW_i \ \| \ r_i \ \| \ ID_i || \ \sigma_i)$.
Compute $SK' = h(auth \ \| \ y)$.
If $(h(SK' \ \| \ TS_j \ \| \ D_1) = cert)$,
store $SK'(= SK)$ as session key
shared with SD_j.

Store $SK(= SK')$ as session
key shared with U_i.

Fig. 3. Login and authentication phase.

Step 1: U_i provides identity ID_i, presents password PW_i, and imprints current biometrics BIO_i at a particular terminal's sensor. SC_i calculates $\sigma_i = \text{Rep}(\text{BIO}_i, \tau_i)$ and $\text{IPB}'_i = h(\text{PW}_i \| h(\text{ID}_i \| \sigma_i))$. Only when the calculated IPB'_i matches IPB_i stored in SC_i will the login be considered as successful; thus, ID_i, PW_i, and BIO_i are legitimate. SC_i further calculates $r_i = r^*_i \oplus\ h(\text{ID}_i \| h(\text{PW}_i \| \sigma_i))$ and recovers the values $\text{EID}_i = \text{EID}^*_i \oplus\ h(\text{ID}_i \| r_i \|\ \text{PW}_i \| \sigma_i)$, $\text{LTK}_i = \text{LTK}^*_i \oplus\ h(r_i \| \text{ID}_i \| \ \sigma_i \| \text{PW}_i)$, and *DeviceList* $=$ *DeviceList*$^* \oplus\ h(\text{PW}_i \| r_i \| \text{ID}_i \| \ \sigma_i)$, and notifies U_i to provide a new password along with biometrics, if needed.

Step 2: U_i selects a new password PW^{new}_i and imprints new biometrics $\text{Bio}^{\text{new}}_i$. The user can also opt not to update his/her biometrics; thus, the new biometrics $\text{Bio}^{\text{new}}_i$ remains same as the existing biometrics BIO_i. SC_i calculates $(\sigma^{\text{new}}_i, \tau^{\text{new}}_i) = \text{Gen}(\text{BIO}^{\text{new}}_i)$, new identity verification token $\text{IPB}^{\text{new}}_i = h(\text{PW}^{\text{new}}_i \| h(\text{ID}_i \| \sigma^{\text{new}}_i))$, $r^{*\text{new}}_i = r_i \oplus h(\text{ID}_i \| h(\text{PW}_i \| \sigma_i))$, $\text{EID}^{*\text{new}}_i = \text{EID}_i \oplus h(\text{ID}_i \| r_i \| \text{PW}^{\text{new}}_i \| \sigma^{\text{new}}_i)$, $\text{LTK}^{*\text{new}}_i = \text{LTK}_i \oplus h(r_i \| \text{ID}_i \| \sigma^{\text{new}}_i \| \text{PW}^{\text{new}}_i)$, and *DeviceList*$^{*\text{new}} =$ *DeviceList* $\oplus\ h(\text{PW}^{\text{new}}_i \| r_i \| \text{ID}_i \| \sigma^{\text{new}}_i)$.

Step 3: SC_i finally replaces *DeviceList*, EID^*_i, LTK^*_i, R^*_i, IBP_i and τ_i with *DeviceList*$^{*\text{new}}$, $\text{EID}^{*\text{new}}_i$, $\text{LTK}^{*\text{new}}_i$, $r^{*\text{new}}_i$, $\text{IPB}^{\text{new}}_i$, and τ^{new}_i, respectively, in its memory.

F. Smart Card Revocation Phase

In the proposed scheme, the user revocation process is incorporated during the login and user authentication phase (see

Section IV-D). To revoke access of U_i, GWN sets $x' = h(\text{LTS}\|\text{RID}_i)$ instead of generating a new random number. During U_i's subsequent login attempts, the first part of the conditional check will be $(x = h(\text{LTS}\|\text{RID}_i))$ or $(x - \text{TS}_i > \Delta T_L))$, and based on the check the user will either be granted or denied access. Since both EID'_i and X_i contain x', encrypted with the long-term key LTS, U_i cannot subvert this mechanism by refusing to update the value of EID_i because U_i cannot distinguish the valid dynamic identity from one that is a revocation token. Moreover, since all dynamic identities have a fixed lifetime, the continuous usage of the same dynamic identity is not also possible.

Remark 1: Assume that a revoked user may reuse an old EID_i (by not updating EID_i with EID'_i) in order to circumvent his/her revocation. However, in this case the second part $(x - TS_i > \Delta T_L))$ of the conditional check $(x = h(\text{LTS}\|\text{RID}_i))$ or $(x - TS_i > \Delta T_L))$ will be "true," and thus the authentication request will be declined. This is because EID_i has a finite lifetime, ΔT_L, and the revoked user, who may be unaware of being revoked, will use an old EID_i that has expired. Any necessary revocation notice should be postponed and sent after ΔT_L time only. Additionally, when a user is being revoked, there must be some mechanism to identify that the user has been revoked.

Remark 2: If it is operationally challenging to time-synchronize all IoT sensing devices in a large IoT system, we can use only random nonces attached to the messages during the login and authentication phase discussed in Section IV-D. However, to protect against replay attacks, we need to adopt strategies such as those suggested in [45] and [46].

G. Dynamic IoT Device Addition Phase

New IoT SDs can be dynamically enrolled into the system at any time after the setup phase through the steps described in Section IV-B. The list of available SDs is saved in each user's smart card and any change to the list is also reflected via Dev′ during the authentication procedure (see Section IV-D).

V. Security Analysis

We evaluate the security robustness using both formal and informal security analysis in this section. First, we prove that the proposed scheme provides session key security under the popular ROR model [4] (Section V-A) and mutual authentication using BAN logic proof [5] (Section V-B). After that, we demonstrate that the proposed scheme is resilient against other known attacks using informal security analysis (Section V-C). Apart from these, we perform a formal security verification using the popular automated verification tool, AVISPA [6] (Section V-D).

A. ROR Model-Based Formal Security Analysis

We first discuss describe the ROR model [4], prior to presenting the formal security proof.

1) ROR Model: The main participants in the proposed scheme involved during the registration, login and authentication procedures are: 1) user U_i; 2) GWN; and 3) IoT SD_j. The following are associated with the ROR model, which are relevant to the proposed scheme.

a) Participants: We denote $\pi^u_{U_i}$, π^v_{GWN}, and $\pi^w_{\text{SD}_j}$ as the instances u, v, and w corresponding to U_i, GWN, and SD_j, respectively. These are also called oracles.

b) Accepted state: Let π^w be an instance. π^w is in an accepted state, when upon getting the final expected protocol message, it enters into an accept state. If all the sent and received messages by π^w are arranged in succession, it constitutes the session identification sid of π^w for the running session.

c) Partnering: Based on the fulfillment of the following three indicators, two instances, say π^{w_1} and π^{w_2}, are called partners to each other: 1) π^{w_1} and π^{w_2} will be in accept states; 2) π^{w_1} and π^{w_2} will authenticate each other mutually and also have the same sid; and 3) π^{w_1} and π^{w_2} will also be mutual partners of each other.

d) Freshness: We call either the instance $\pi^u_{U_i}$ or $\pi^v_{\text{SD}_j}$ as fresh when the established session key SK among U_i and SD_j can not be disclosed to an adversary $\mathcal{A}$ with the help of the defined Reveal(π^w) query as given below [7].

e) Adversary: According to the threat model (Section II-D2), $\mathcal{A}$ will have complete control over all the communication messages as the ROR model is also based on the DY threat model [15]. This implies that $\mathcal{A}$ may eavesdrop, delete, or adjust the exchanged messages, or even the messages can be also fabricated or injected into the network. Also, the following defined queries are accessible to $\mathcal{A}$ [28].

f) Execute(π^u, π^v, π^w): Execution of this query allows $\mathcal{A}$ to intercept all the transmitted messages among U_i, GWN and SD_j. Due to intercepting nature, an eavesdropping attack is modeled under this query.

g) Send(π^w, m): Upon executing this query by $\mathcal{A}$, a message, say m, can be sent to its participating instance π^w, and a response message is also received in reply. This query is treated as an active attack.

h) Reveal(π^w): Upon executing this query, current session key SK computed by π^w (and its partner) is revealed to $\mathcal{A}$.

i) CorruptSC($\pi^u_{U_i}$): Using this query, the credentials $\{r^*_i$, IPB_i, τ_i, EID^*_i, LTK^*_i, $DeviceList^*\}$ stored in a legal user U_i's stolen or lost smart card SC_i are known to $\mathcal{A}$.

j) CorruptIoTSD($\pi^w_{SD_j}$): By executing this query, $\mathcal{A}$ will have the extracted credentials $\{ID_j$, $\text{LTK}_j\}$ from a captured IoT sensing device SD_j. Based on the observation made in [28], it is also assumed that both the queries *CorruptSC* and *CorruptIoTSD* provide the weak corruption model in which a participant instance's short-term keys and the internal data are not corrupted.

k) Test(π^w): The semantic security of the established session key SK among U_i and SD_j following the indistinguishability in the ROR model [4] is determined using this query. At first, an unbiased coin c needs to be tossed, and then its outcome is only available to $\mathcal{A}$. This outcome decides the result of the *Test* query. Let $\mathcal{A}$ execute this query. If SK is fresh, π^w produces SK upon the satisfaction of the condition

$c = 1$ or a random number for the fulfillment of the condition $c = 0$. In other cases, it returns a null value.

As in [7], a restriction is also imposed on $\mathcal{A}$ for accessing only a limited number of *CorruptSC*$(\pi^u_{U_i})$ and *CorruptIoTSD*$(\pi^w_{\text{SD}_j})$ queries. However, $\mathcal{A}$ is permitted to make as many Test(π^t) queries. Since the GWN is trusted (Section II-D1), $\mathcal{A}$ can not make a corrupt query corresponding to the GWN.

All communicating entities including $\mathcal{A}$ can access a collision resistant hash function $h(\cdot)$ (see Definition 1). $h(\cdot)$ is modeled as a random oracle, say $\mathcal{HO}$.

2) Security Proof: The semantic security of the proposed scheme, say $\mathcal{P}$ under the considered ROR model [4] is demonstrated in Theorem 1. Wang *et al.* [47] investigated that Zipf's law is significantly different from the uniform distribution for user-chosen passwords. In practice, the size of password dictionary is much more constrained in the sense that the users may not use the entire space of passwords, but rather a small space of the allowed characters space [47]. Zipf's law has been applied in proving the session key security of the proposed scheme $\mathcal{P}$ in Theorem 1. Zipf's law is also utilized in recently proposed authentication schemes, such as the scheme in [48].

Theorem 1: If $\mathcal{A}$ is a polynomial time adversary running against the proposed scheme $\mathcal{P}$ under the ROR model, which uses the Zipf's law for the user-chosen passwords, l denotes the number of bits in the biometrics secret key σ_i, and $\text{Adv}^{\text{AKE}}_{\mathcal{P},\mathcal{A}}$ is $\mathcal{A}$'s advantage in breaking $\mathcal{P}$'s semantic security, then

$$\text{Adv}^{\text{AKE}}_{\mathcal{P},\mathcal{A}} \leq \frac{q_h^2}{|\text{Hash}|} + 2\Big(\max\Big\{C'.q_s^{s'}, \frac{q_s}{2^l}\Big\} + \text{Adv}^{\text{IND-CPA}}_{\Omega}(k)\Big)$$

where q_h, q_s, and $|\text{Hash}|$ are the number of $\mathcal{HO}$ queries, *Send* queries and range space of $h(\cdot)$, respectively, $\mathcal{A}$'s advantage in cracking the IND-CPA secure symmetric cipher Ω (see Definition 2) is $\text{Adv}^{\text{IND-CPA}}_{\Omega}(k) = \text{Adv}^{\text{IND-CPA}}_{\Omega,\text{SE}}(k)$ or $\text{Adv}^{\text{IND-CPA}}_{\Omega,\text{ME}}(k)$, and C' and s' are the Zipf's parameters [47].

Proof: We follow the similar proof of this theorem as presented in [48]. We need to define a sequence of five games, namely G_j $(j = 0, 1, 2, 3, 4)$. Let $\text{Succ}^{G_j}_{\mathcal{A}}$ denote an event wherein an $\mathcal{A}$ can guess the random bit c in the game G_j correctly, and the corresponding $\mathcal{A}$'s advantage is given by $\text{Adv}^{G_j}_{\mathcal{P},\mathcal{A}} = \Pr[\text{Succ}^{G_j}_{\mathcal{A}}]$.

Game G₀: This is the initial game, which corresponds to a real attack executed by $\mathcal{A}$ against the proposed scheme $\mathcal{P}$ in the ROR model. Since the bit c is picked up at the beginning of the game G_0, it follows from the definition of the semantic security that

$$\text{Adv}^{\text{AKE}}_{\mathcal{P},\mathcal{A}} = \Big|2.\text{Adv}^{G_0}_{\mathcal{P},\mathcal{A}} - 1\Big|. \quad (1)$$

Game G₁: It corresponds to an eavesdropping attack executed by $\mathcal{A}$. $\mathcal{A}$ can make the *Execute* query, and intercepts all the communicated messages $M_1 = \{\text{EID}_i, \text{EID}_j, \text{TS}_i\}$, $M_2 = \{D_2, \text{TS}_{\text{gwn}}\}$, and $M_3 = \{D_1, D_3, \text{cert}, \text{TS}_j\}$ during the login and authentication phase of the proposed scheme. After the game is finished, the *Test* query is made by $\mathcal{A}$. The outcome of the *Test* query decides if the session key $\text{SK} = h(\text{auth}\|y)$ is a real session key or a random number, where $\text{auth} = h(\text{LTK}_i\|X_i\|\text{RID}_i)$ and $X_i = h(\text{TS}_{\text{gwn}}\|x_i)$, x_i and y are temporal secret keys selected by the GWN and SD_j, respectively, and LTK_i and RID_i are the LTS key and pseudo-identity of U_i, respectively. Therefore, $\mathcal{A}$ needs the secret credentials x_i, y, X_i, LTK_i, and RID_i to compute the session key SK. These secret credentials can not be obtained/derived by eavesdropping the messages M_1–M_3 only. Hence, the winning probability of the game G_1 by $\mathcal{A}$ is not increased. Since the games G_0 and G_1 are indistinguishable, we have

$$\text{Adv}^{G_1}_{\mathcal{P},\mathcal{A}} = \text{Adv}^{G_0}_{\mathcal{P},\mathcal{A}}. \quad (2)$$

Game G₂: The games G_1 and G_2 are indistinguishable except the simulations of the *Send* and $\mathcal{HO}$ queries are included in G_2. The game G_2 is an active attack in which the task of $\mathcal{A}$ is to convince a participant that a modified (fake) message is a legitimate message. Assume that $\mathcal{A}$ executes q_h number of various $\mathcal{HO}$ queries with the help of q_s number of the *Send* queries. It is worth noting that in the proposed scheme, all the transmitted messages M_1–M_3 are constructed in such a manner that all are dynamic in nature and no hash collision occurs. Thus, with the help of the birthday paradox, it follows that:

$$\Big|\text{Adv}^{G_1}_{\mathcal{P},\mathcal{A}} - \text{Adv}^{G_2}_{\mathcal{P},\mathcal{A}}\Big| \leq q_h^2/(2|\text{Hash}|). \quad (3)$$

Game G₃: In this game, the simulation *CorruptSC* and *CorruptIoTSD* are included. In this context, $\mathcal{A}$ can obtain the information $\{r_i^*, \text{IPB}_i, \tau_i, \text{EID}_i^*, \text{LTK}_i^*, \textit{DeviceList}^*\}$ stored in SC_i and also the credentials $\{\text{ID}_j, \text{LTK}_j\}$ from a captured IoT sensing device, say SD'_j. However, for the noncompromised IoT sensing device SD_j, both ID_j and LTK_j are distinct. U_i uses both password PW_i and biometrics BIO_i. However, the probability of guessing the biometric secret key σ_i of l bits (respectively, BIO_i) is approximately $(1/2^l)$ [49]. $\mathcal{A}$ can also try to guess low-entropy passwords using the Zipf's law on passwords [47]. If we only consider the trawling guessing attacks, the actually the advantage of $\mathcal{A}$ will be over 0.5 when $q_s = 10^7$ or 10^8 [47]. If we also consider the targeted guessing attacks (in which $\mathcal{A}$ can make use of the target user's personal information), the advantage of $\mathcal{A}$ will be over 0.5 when $q_s \leq 10^6$ [47]. In practice, only a limited number of wrong password inputs are permitted in the system. Since the games G_3 and G_4 are identical in the absence of guessing attacks, we have the following result [48]:

$$|\text{Adv}^{G_2}_{\mathcal{P},\mathcal{A}} - \text{Adv}^{G_3}_{\mathcal{P},\mathcal{A}}]| \leq \max\Big\{C' \cdot q_s^{s'}, \frac{q_s}{2^l}\Big\}. \quad (4)$$

Game G₄: G_4 is the final game in which $\mathcal{A}$ by intercepting the messages M_1–M_3 tries to derive the session key $\text{SK} = h(\text{auth}\|y)$ with the help of decryption of the information EID_i, EID_j, and D_1–D_3. To derive $\text{auth} = h(\text{LTK}_i\|X_i\|\text{RID}_i)$, it is needed to decrypt EID_i to have RID_i, the LTS LTK_i and also $X_i = h(\text{TS}_{\text{gwn}}\|x_i)$. Also, decryption of D_2 and D_3 requires the secret keys. This task makes computationally expensive due to the usage of the stateless CBC mode of AES-128 encryption/decryption. It is worth noting that for each encryption and decryption, the IV value is set random. Due to the IND-CPA property (see Definition 2), it then follows that:

$$|\text{Adv}^{G_3}_{\mathcal{P},\mathcal{A}} - \text{Adv}^{G_4}_{\mathcal{P},\mathcal{A}}]| \leq \text{Adv}^{\text{IND-CPA}}_{\Omega}(k). \quad (5)$$

Since all the oracles are executed by $\mathcal{A}$, it only remains to guess the bit c for winning the game after querying the *Test* query. Thus, $\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_4} = 1/2$.

From (1) and (2), we obtain, $(1/2).\text{Adv}_{\mathcal{P},\mathcal{A}}^{\text{AKE}} = |\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_0} -(1/2)| = |\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_1} -\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_4}|$. The triangular inequality gives $|\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_1} -\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_4}| \leq |\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_1} - \text{Adv}_{\mathcal{P},\mathcal{A}}^{G_2}| + |\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_2} - \text{Adv}_{\mathcal{P},\mathcal{A}}^{G_4}| \leq |\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_1} - \text{Adv}_{\mathcal{P},\mathcal{A}}^{G_2}| + |\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_2} - \text{Adv}_{\mathcal{P},\mathcal{A}}^{G_3}| +|\text{Adv}_{\mathcal{P},\mathcal{A}}^{G_3} - \text{Adv}_{\mathcal{P},\mathcal{A}}^{G_4}| \leq (q_h^2/2.|\text{Hash}|) + \max\{C' \cdot q_s^{s'}, (q_s/2^l)\}+\text{Adv}_{\Omega}^{\text{IND-CPA}}(k)$. Solving (3)–(5) and rearranging the terms, we have the required result: $\text{Adv}_{\mathcal{P},\mathcal{A}}^{\text{AKE}} \leq q_h^2/|\text{Hash}| +2(\max\{C'.q_s^{s'}, (q_s/2^l)\} + \text{Adv}_{\Omega}^{\text{IND-CPA}}(k))$. ■

B. Mutual Authentication Through BAN Logic

We utilize the widely recognized BAN logic [5] to prove that in the proposed scheme the mutual authentication between a registered legitimate user U_i and an accessed IoT SD_j is achieved in presence of the GWN. The BAN logic uses the following notations.

1) $A \mid\equiv S$*:* Principal A believes a statement S or A is entitled to believe the statement S.
2) *#(S):* Formula S is fresh.
3) $A \mid\Rightarrow S$*:* A has jurisdiction over a statement S.
4) $A \triangleleft S$*:* A sees S.
5) $A \mid\!\sim S$*:* A once said S.
6) (S_1, S_2)*:* Formula S_1 or S_2 is included as a part of the formula (S_1, S_2).
7) $\{M\}_K$*:* Encryption of M using the key K.
8) $\langle S_1\rangle_{S_2}$*:* S_1 and S_2 are combined.
9) $A \overset{K}{\longleftrightarrow} B$*:* A and B apply the shared key K to communicate each other. K is treated as a good key in that sense that it will not be disclosed by any part apart from A and B.
10) $A \overset{K}{\rightleftharpoons} B$*:* K is secret and known only to A and B.

There are four rules which govern the BAN logic, and these are listed as follows.

1) *Rule 1:* $[(A \mid\equiv A \overset{K}{\longleftrightarrow} B, A \triangleleft \{S\}_K)/(A \mid\equiv B \mid\!\sim S)]$ and $[(A \mid\equiv A \overset{S_1}{\rightleftharpoons} B, A \triangleleft \langle S\rangle_{S_1})/(A \mid\equiv B \mid\!\sim S)]$. This is known as the message-meaning rule.
2) *Rule 2:* $[(A \mid\equiv \#(S), A \mid\equiv B \mid\!\sim S)/(A \mid\equiv B \mid\equiv S)]$. This rule is called the nonce-verification rule.
3) *Rule 3:* $[(A \mid\equiv B \mid\Rightarrow S, A \mid\equiv B \mid\equiv S)/(A \mid\equiv S)]$. It is the jurisdiction rule.
4) *Rule 4:* $[(A \mid\equiv \#(S))/(A \mid\equiv \#(S, S_1))]$. This rule is termed as the freshness-conjunction rule.

Using the above rules, we now prove Theorem 2.

Theorem 2: The proposed scheme achieves secure mutual authentication between U_i and SD_j in the presence of the GWN.

Proof: We define the following two goals.

G_1: $SD \mid\equiv U_i \overset{\text{SK}}{\longleftrightarrow} \text{SD}_j$.

G_2: $U_i \mid\equiv U_i \overset{\text{SK}}{\longleftrightarrow} \text{SD}_j$.

The generic forms of the transmitted messages during the login and authentication procedure under the proposed scheme are listed below.

1) Message M_1 gives $U_i \rightarrow \text{GWN}$: $\text{EID}_i = E[\text{RID}_i, x]_{\text{LTS}}$, $\text{EID}_j = \text{ID}_j \oplus h(\text{TS}_i \| \text{LTK}_i)$, TS_i.
2) From message M_2, we have, $\text{GWN} \rightarrow \text{SD}_j$: $D_2 = E[\text{auth}, D_1, \text{TS}_{\text{gwn}}]_{\text{LTK}_j}$, TS_{gwn}.
3) Message M_3 results $\text{SD}_j \rightarrow U_i$: $D_1 = E[\text{EID}'_i, X_i, \text{Dev}']_{\text{LTK}_i}$, $D_3 = E[y, \text{TS}_j]_{\text{auth}}$, cert, TS_j.

The idealized forms of the above messages are also given below.

M_1: $U_i \rightarrow \text{GWN} : \langle[\text{RID}_i, x]_{\text{LTS}}\rangle$.

M_2: $\text{GWN} \rightarrow \text{SD}_j : \langle\langle \text{ auth}, \langle\text{EID}'_i, X_i, \text{Dev}'\rangle_{\text{GWN}\overset{\text{LTK}_i}{\longleftrightarrow}U_i}, \text{TS}_{\text{gwn}} \rangle_{\text{GWN}\overset{\text{LTK}_j}{\longleftrightarrow}\text{SD}_j}\rangle$.

M_3: $\text{SD}_j \rightarrow U_i : \langle\langle \text{ EID}'_i, X_i, \text{Dev}' \rangle_{\text{GWN}\overset{\text{LTK}_i}{\longleftrightarrow}U_i}, \langle y, \text{TS}_j\rangle_{\text{SD}_j\overset{\text{auth}}{\longleftrightarrow}U_i}\rangle$.

The following suppositions regarding the initial states are given below.

H_1: $U_i \mid\equiv \#(X_i)$.

H_2: $U_i \mid\equiv \text{SD}_j \mid\Rightarrow U_i \overset{\text{auth}}{\rightleftharpoons} \text{SD}_j$.

H_3: $U_i \mid\equiv \text{SD}_j \mid\Rightarrow U_i \overset{y}{\rightleftharpoons} \text{SD}_j$.

H_4: $\text{SD}_j \mid\equiv \#(\text{auth})$.

H_5: $\text{SD}_j \mid\equiv U_i \overset{y}{\rightleftharpoons} \text{SD}_j$.

H_6: $\text{SD}_j \mid\equiv \text{GWN} \mid\Rightarrow U_i \overset{\text{auth}}{\rightleftharpoons} \text{SD}_j$.

By analyzing the messages M_1–M_3 and assumptions H_1–H_7 based on the BAN logic rules, the goals (goals G_2 and G_3) are proved as follows. From M_3, we have the following.

S_1: $U_i \mid\equiv \text{SD}_j \mid\!\sim (\text{auth}, y)$.

From Rule 4, H_1, and the fact that auth $= h(\text{LTK}_i, X_i, \text{RID}_i)$, we have the following:

S_2: $U_i \mid\equiv \#(U_i \overset{\text{auth}}{\rightleftharpoons} \text{SD}_j)$.

From Rule 2, S_0, and S_1, we obtain the following result:

S_3: $U_i \mid\equiv \text{SD}_j \mid\equiv U_i \overset{\text{auth}}{\rightleftharpoons} \text{SD}_j$.

From Rule 3, H_2, and S_2, it follows:

S_4: $U_i \mid\equiv U_i \overset{\text{auth}}{\rightleftharpoons} \text{SD}_j$.

Rule 4, and H_4 lead to the following result:

S_5: $U_i \mid\equiv \#(U_i \overset{y}{\rightleftharpoons} \text{SD}_j)$.

From Rule 2, S_4, and S_0, we have the following:

S_6: $U_i \mid\equiv \text{SD}_j \mid\equiv U_i \overset{y}{\rightleftharpoons} \text{SD}_j$.

From Rule 3, H_4 and S_5 lead to the following:

S_7: $U_i \mid\equiv U_i \overset{y}{\rightleftharpoons} \text{SD}_j$.

From S_3, S_6, and since $\text{SK} = h(\text{auth}\|y)$, it follows:

S_8: $U_i \mid\equiv U_i \overset{\text{SK}}{\longleftrightarrow} \text{SD}_j$. (goal G_1) Using Rule 4 and H_5, we obtain the following:

S_9: $\text{SD}_j \mid\equiv \#(U_i \overset{\text{auth}}{\rightleftharpoons} \text{SD}_j)$.

With the help of Rule 2, M_2, and S_8, the following result is obtained.

S_{10}: $\text{SD}_j \mid\equiv \text{GWN} \mid\equiv U_i \overset{\text{auth}}{\rightleftharpoons} \text{SD}_j$.

Rule 3, S_9, and H_7 lead to the following:

S_{11}: $\text{SD}_j \mid\equiv U_i \overset{\text{auth}}{\rightleftharpoons} \text{SD}_j$.

Finally, using S_{10}, H_6, and the fact that $\text{SK} = h(\text{auth}\|y)$, the following goal is obtained:

S_{12}: $\text{SD}_j \mid\equiv U_i \overset{\text{SK}}{\longleftrightarrow} \text{SD}_j$. (goal M_3)

Hence, the goals G_1 and G_2 assure mutual authentication among U_i and SD_j in presence of GWN. ■

C. Informal Security Analysis

Through informal security analysis, we demonstrate that the proposed scheme is resilient against following well-known attacks.

1) Impersonation Attacks: We consider the following scenarios.

a) User impersonation attack: Assume that an adversary $\mathcal{A}$ attempts to impersonate a legitimate user U_i by means of sending a legal login request message M_1 on behalf of U_i to the GWN. To construct a legal message, say $M_1 = \{\mathrm{EID}_i, \mathrm{EID}_j, \mathrm{TS}'_i\}$, $\mathcal{A}$ can generate the current timestamp TS'_i and then proceeds to calculate $\mathrm{EID}_j = E[\mathrm{ID}_j \| \mathrm{TS}'_i]_{\mathrm{LTK}_i}$. However, such an attempt will fail to construct M'_1 because $\mathcal{A}$ does not have the secret credentials ID_j and $\mathrm{LTK}_i = h(\mathrm{LTS} \oplus \mathit{RID}_i)$. Thus, it is computationally expensive for $\mathcal{A}$ to forge M_1 on behalf of the original user U_i and this attack is protected in the proposed scheme.

b) GWN impersonation attack: Suppose $\mathcal{A}$ tries to construct a legal authentication request message M_2 and send it to an accessed sensing device SD_j on behalf of the GWN. To construct the message $M_2 = \{D_2, \mathrm{TS}'_{\mathrm{gwn}}\}$, $\mathcal{A}$ can generate the current timestamp $\mathrm{TS}'_{\mathrm{gwn}}$ and calculate $D_1 = E[\mathrm{EID}'_i, X_i, \mathrm{Dev}']_{\mathrm{LTK}_i}$ and $D_2 = E[\mathrm{auth}, D_1, \mathrm{TS}'_{\mathrm{gwn}}]_{\mathrm{LTK}_j}$. However, this task is computationally expensive as the secret credentials EID_i, X_i, auth, LTK_i and LTK_j are not available to $\mathcal{A}$ (see Definition 2). This shows that the proposed scheme is resilient against the GWN impersonation attack.

c) IoT smart device impersonation attack: Assume that $\mathcal{A}$ also attempts to construct a legal authentication request message M_3 and sent it to U_i on behalf of SD_j. For this motivation, $\mathcal{A}$ can generate the current timestamp TS'_j. $\mathcal{A}$ can not decrypt D_2 to get auth, D_1 and $\mathrm{TS}_{\mathrm{gwn}}$ because it needs the secret key LTK_j. Without auth, it is computationally expensive to compute $D_3 = E[y, \mathrm{TS}'_j]_{\mathrm{auth}}$, session key $\mathrm{SK} = h(\mathrm{auth} \| y)$ and $\mathrm{cert} = h(\mathrm{SK} \| \mathrm{TS}'_j \| D_1)$ in order to send the authentication reply message $M_3 = \{D_1, D_3, \mathrm{cert}, \mathrm{TS}'_j\}$ to U_i on behalf of SD_j (see Definition 2). This clearly shows that the proposed scheme is resilient against this attack.

2) Stolen Smart Card Attacks: Assume that an adversary $\mathcal{A}$ extracts the secret credentials from a lost or stolen SC_i of a registered user U_i through power analysis attacks [18]. Then, $\mathcal{A}$ will have the credentials EID^*_i, LTK^*_i, $\mathit{DeviceList}^*$, r^*_i, IPB_i, and τ_i. Suppose $\mathcal{A}$ guesses a password PW'_i and attempts to verify whether it is a correct password using the knowledge of the information $r^*_i = r_i \oplus h(\mathrm{ID}_i \| h(\mathrm{PW}_i \| \sigma_i))$, $\mathrm{IPB}_i = h(\mathrm{PW}_i \| h(\mathrm{ID}_i \| \sigma_i))$, $\mathrm{EID}^*_i = \mathrm{EID}_i \oplus h(r_i \| \mathrm{PW}_i \| \sigma_i)$, $\mathrm{LTK}^*_i = \mathrm{LTK}_i \oplus h(r_i \| \sigma_i \| \mathrm{PW}_i)$, and $\mathit{DeviceList}^* = \mathit{DeviceList} \oplus h(\mathrm{PW}_i \| r_i \| \sigma_i)$. However, without having the secrets r_i, σ_i, and ID_i, it is computationally expensive to validate PW_i due to $h(\cdot)$'s collision resistant property (see Definition 1). Also, to derive σ_i, $\mathcal{A}$ again requires the secret credentials r_i, PW_i, and ID_i. Therefore, offline (password/biometrics) guessing attacks are protected in the proposed scheme in conjunction with the stolen smart card attack.

3) Privileged-Insider Attack: Though the GWN is trusted, a privileged-insider of the GWN can act as an insider adversary $\mathcal{A}$. Suppose $\mathcal{A}$ knows the registration credential $\mathrm{RID}_i = h(\mathrm{ID}_i \| r_i)$ that was sent during the user registration process to the GWN. Then, to know ID_i from RID_i, $\mathcal{A}$ requires the random secret r_i which is stored in SC_i in the form $r^*_i = r_i \oplus h(\mathrm{PW}_i \| \sigma_i)$. Furthermore, after the user registration is over, suppose $\mathcal{A}$ can have the stolen/lost SC_i of a registered user U_i. However, based on the analysis carried out in Section V-C2, it is computationally expensive to derive other secret credentials r_i, PW_i, and σ_i. This indicates that the proposed scheme is secure against privileged-insider attack.

4) Offline Guessing Attacks: Assume that an adversary $\mathcal{A}$ controls the biometric reader, and he/she has access to U_i's lost/stolen smart card SC_i. Then, $\mathcal{A}$ can compute $(\sigma_i, \tau_i) = \mathrm{Gen}(\mathrm{BIO}_i)$, and have access to EID^*_i and r^*_i from SC_i's memory using power analysis attacks [18]. Assume that $\mathcal{A}$ intercepts the message $M_1 = \{\mathrm{EID}_i, \mathrm{EID}_j, \mathrm{TS}_i\}$ to learn EID_i, and thus $\mathcal{A}$ can construct the following expression:

$$\mathrm{EID}^*_i \oplus \mathrm{EID}_i = h\big(\mathrm{ID}_i \| \big(r^*_i \oplus h(\mathrm{ID}_i \| h(\mathrm{PW}_i \| \sigma_i)) \\ \| \mathrm{PW}_i \| \sigma_i)).$$

This expression contains two unknowns, namely the identity ID_i and password PW_i of the user U_i. According to threat model defined in Section II-D2, guessing both password and identity of a registered user and then verifying those in polynomial time is a computationally expensive task for $\mathcal{A}$, because the registered legitimate users always use the words as passwords and identities from the dictionary [19]. Thus, having the computed σ_i to guess and verify both the password PW_i and identity ID_i of U_i at the same time is a "computationally expensive task." In addition, deriving PW_i and ID_i of U_i from the hash value $\mathrm{EID}^*_i \oplus \mathrm{EID}_i$ is also a computationally expensive task due to the collision resistant property of one-way hash function $h(\cdot)$ (see Definition 1). Therefore, the proposed scheme is secure against the offline (password/identity) guessing attacks when the biometric of a user is compromised.

5) Ephemeral Secret Leakage Attack: In the proposed scheme, both U_i and SD_j establish a common session key $SK = h(auth \| y)$ during the execution of login and authentication phase, where $auth = h(\mathrm{LTK}_i \| X_i \| \mathit{RID}_i)$ and $X_i = h(TS_{\mathrm{gwn}} \| x_i)$, x_i and y are temporal secret keys selected by the GWN and SD_j, respectively, and LTK_i and RID_i are the LTS key and pseudo-identity of U_i, respectively. Based on "the CK-adversary model discussed in the threat model in Section II-D2," the security of SK is then dependent on the following cases.

Case 1: Let $\mathcal{A}$ have the short-term secret credentials x_i and y. Then, it is computationally difficult for $\mathcal{A}$ to calculate correct session key SK without having the permanent (long term) secret credentials LTK_i and RID_i.

Case 2: Let some or all of the LTSs LTK_i and RID_i are revealed to $\mathcal{A}$. Again, it is computationally difficult for $\mathcal{A}$ to calculate SK without short-term secrets x_i and y.

This shows that derivation of a valid session key SK is possible by $\mathcal{A}$ only if the short-term secret and LTS are available at the same time. In addition, compromise of a particular session does not lead to compromise the session keys established in previous/future sessions, because these session keys are entirely different from the compromised session key due to usage of random secrets, current timestamps along with long-terms secrets in calculation of session keys. Hence, the proposed scheme is resilient against ephemeral secret leakage (ESL) attack.

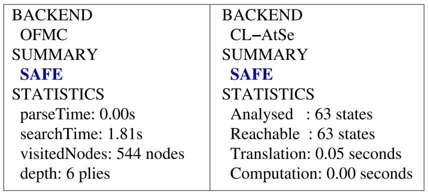
6) Resilience Against Sensing Device Physical Capture Attack: Suppose $\mathcal{A}$ physically captures some IoT SDs. Then, $\mathcal{A}$ can extract all the secret credentials {ID_j, LTK_j} from a captured IoT sensing device, say SD_j's memory. However, it is worth noting that the information ID_j and LTK_j are generated randomly and hence, these are distinct for all deployed sensing devices. Hence, the compromised information {ID_j, LTK_j} do not help in computing the session keys among a user U_i and other noncompromised sensing devices SD'_j. This means that compromise of SD_j does not help to compromise the secure communication between a user U_i and other noncompromised sensing devices SD'_j. Thus, the proposed scheme is resilient against this attack.

7) GWN Bypassing Attack: The GWN bypassing attack is an attack where an attacker $\mathcal{A}$ can create some legitimate messages, in order to gain the trust of other IoT SDs or the authorized users by bypassing GWN in the IoT environment [50]. In the proposed scheme, an SD_j will not be able to create a valid message $M_3 = \{D_1, D_3, \text{cert}, \text{TS}_j\}$, unless it can verify that the received TS_{gwn} is the same as the TS'_{gwn} value retrieved by decrypting D_2 using the key LTK_j. Thus, $\mathcal{A}$ will need to construct $M_2 = \{D_2, \text{TS}_{\text{gwn}}\}$, which is equivalent to the GWN impersonation attack described in Section V-C1. As a result, the proposed scheme is also GWN bypassing attack resilience.

8) Anonymity and Untraceability: Suppose $\mathcal{A}$ eavesdrops and monitors the messages M_1–M_3. However, none of these eavesdropped messages contain any identifying information for user, SD, and the GWN in plaintext formats. Thus, the proposed scheme preserves the anonymity property. Moreover, all these messages are constructed using the temporal random secrets, current timestamps, and LTSs, and these are dynamic in nature from one session to another. This results in tracing a user or a sensing device difficult for $\mathcal{A}$. Therefore, the proposed scheme also preserves untraceability property.

D. Formal Security Verification Using AVISPA Tool

Automated validation of Internet security protocols and applications (AVISPA) [6] is a powerful automated validation tool for security sensitive protocols and applications. Any security protocol to be analyzed in AVISPA requires to be stated under the role-oriented language, high level protocol specification language (HLPSL). There is a translator, called HLPSL2IF, which converts HLPSL to intermediate format (IF). One of the four backends in AVISPA is then given the IF to produce the outcome. The outcome indicates if the tested protocol is safe or unsafe against replay and man-in-the-middle attacks. All the details of AVISPA and HLPSL can be found in [6]. Note that AVISPA implements the DY threat model [15].



```
BACKEND                    | BACKEND
 OFMC                      |  CL-AtSe
SUMMARY                    | SUMMARY
 SAFE                      |  SAFE
STATISTICS                 | STATISTICS
 parseTime: 0.00s          |  Analysed   : 63 states
 searchTime: 1.81s         |  Reachable  : 63 states
 visitedNodes: 544 nodes   |  Translation: 0.05 seconds
 depth: 6 plies            |  Computation: 0.00 seconds
```

Fig. 4. Simulation results under OFMC and CL-AtSe back-ends.

TABLE II
APPROXIMATE TIME FOR CRYPTOGRAPHIC OPERATIONS [52], [53]

Notation	Description (Time to compute)	Rough computation time (in ms)
T_h	One-way hash function	0.5
T_m	ECC point multiplication	63.075
T_a	ECC point addition	16.229
T_{E_s}	Symmetric encryption/decryption	8.7
$T_f \approx T_m$	Fuzzy extractor operation	63.075

The user registration, login, and authentication phases for the proposed scheme are implemented in HLPSL using three basic roles for a user, the GWN, and an SD. The compulsory roles for the session, goal and environment are also defined.

We have then evaluated the proposed scheme against replay and man-in-the-middle attacks under the on-the-fly model checker (OFMC) and constraint logic-based attack searcher (CL-AtSe) back-ends using the security protocol animator (SPAN) for AVISPA [51]. The simulation results provided in Fig. 4 clearly indicate that replay and man-in-the-middle attacks are protected in our scheme.

VI. COMPARATIVE STUDY

The proposed scheme is compared with the recent authentication schemes proposed in IoT environment, such as the schemes of Wazid *et al.* [34], Challa *et al.* [3], Chang and Le [28], and Porambage *et al.* [33].

A. Computation Costs Comparison

For computation cost analysis, we denote T_{E_s}, T_m, T_a, T_f, and T_h as the time needed for computing symmetric encryption/decryption, elliptic curve point multiplication, elliptic curve point addition, fuzzy extractor operation, and hashing operation, respectively. In Table II, we tabulate the approximate time required to perform each operation, which are taken from experimental results performed in [52] and [53].

During the login and authentication phase of the proposed scheme, a user, the GWN and an IoT SD require the computation costs as $12T_h + 3T_{E_s} + T_f$, $5T_h + 5T_{E_s}$, and $2T_h + 2T_{E_s}$, respectively. Hence, the total computation cost in the proposed scheme is $19T_h + 10T_{E_s} + T_f$, which requires approximately 159.58 ms. Table III summarizes the computational overheads of the proposed scheme and the existing schemes in [3], [28], [33], and [34], in terms of atomic operations and an approximate time (in milliseconds) using the values provided in Table II. It can be observed that the proposed scheme requires less overall computation costs, with the exception of the scheme in [34]. However, the computational costs for the resource-limited IoT sensing device in our scheme is less in

TABLE III
COMPUTATION COSTS COMPARISON

Scheme	User	Gateway node	Sensing device	Total cost
Our	$12T_h + 3T_{E_s}$ $+T_f$ (95.18 ms)	$5T_h$ $+5T_{E_s}$ (46 ms)	$2T_h + 2T_{E_s}$ (17.5 ms)	$19T_h+$ $10T_{E_s} + T_f$ (159.58 ms)
[34]	$13T_h + 2T_{E_s}$ $+T_f$ (86.98 ms)	$5T_h$ $+4T_{E_s}$ (37.3 ms)	$4T_h + 2T_{E_s}$ (19.4 ms)	$22T_h+$ $8T_{E_s} + T_f$ (143.68 ms)
[3]	$5T_h + 5T_m$ $+T_f$ (380.95 ms)	$4T_h$ $+5T_m$ (317.38 ms)	$3T_h + 4T_m$ (253.8 ms)	$12T_h$ $+14T_m + T_f$ (952.13 ms)
[28]	$7T_h + 2T_m$ (129.65 ms)	$9T_h$ (4.5 ms)	$5T_h + 2T_m$ (128.65 ms)	$21T_h + 4T_m$ (262.8 ms)
[33]	$3T_h+$ $+2T_m$ (132.05 ms)	$-$	$3T_h + T_a$ $+2T_m$ (132.05 ms)	$6T_h + T_a$ $+4T_m$ (271.53 ms)

TABLE IV
COMMUNICATION COSTS COMPARISON

Scheme	No. of bytes	No. of messages
Our (without revocation)	288	3
Our (with revocation)	320	3
Wazid *et al.* [34]	324	4
Challa *et al.* [3]	316	3
Chang-Le [28]	284	4
Porambage *et al.* [33]	192	4

TABLE V
SECURITY AND FUNCTIONALITY FEATURES COMPARISON

Feature	Our	[34]	[3]	[28]	[33]
$\mathcal{FR}_1$	☑	☑	☑	☑	☒
$\mathcal{FR}_2$	☑	☑	☑	☒	☑
$\mathcal{FR}_3$	☑	☑	☒	☒	☒
$\mathcal{FR}_4$	☑	☑	☒	☒	NA
$\mathcal{FR}_5$	☑	☑	☑	☒	☑
$\mathcal{FR}_6$	☑	☑	☑	☑	☑
$\mathcal{FR}_7$	☑	☑	☑	☑	☑
$\mathcal{FR}_8$	☑	☑	☒	☑	☒
$\mathcal{FR}_9$	☑	☑	☑	☒	☑
$\mathcal{FR}_{10}$	☑	☑	☑	☑	☑
$\mathcal{FR}_{11}$	☑	☑	☒	☒	☒
$\mathcal{FR}_{12}$	☑	☑	☑	☑	☑
$\mathcal{FR}_{13}$	☑	☑	☑	☑	☒
$\mathcal{FR}_{14}$	☑	☑	☑	☑	☒
$\mathcal{FR}_{15}$	☑	☑	☑	☑	☑
$\mathcal{FR}_{16}$	☑	☑	☒	☒	NA
$\mathcal{FR}_{17}$	3	3	3	2	NA
$\mathcal{FR}_{18}$	☑	☑	☑	☒	NA
$\mathcal{FR}_{19}$	☑	☑	☑	☒	NA
$\mathcal{FR}_{20}$	☑	☑	☑	☒	☒
$\mathcal{FR}_{21}$	☑	☑	☑	☒	☒
$\mathcal{FR}_{22}$	☑	☒	☑	☒	☒
$\mathcal{FR}_{23}$	☑	☑	☑	☑	☒
$\mathcal{FR}_{24}$	☑	☑	☑	☑	☒
$\mathcal{FR}_{25}$	☑	☑	☑	☑	☒
$\mathcal{FR}_{26}$	☑	☑	☑	☒	☒

Note: ☑: the scheme supports a feature or it is resilient against an attack; ☒: the scheme does not support a feature or it is not secure against an attack. $\mathcal{FR}_1$: user anonymity; $\mathcal{FR}_2$: smart device anonymity; $\mathcal{FR}_3$: untraceability; $\mathcal{FR}_4$: offline password guessing attack; $\mathcal{FR}_5$: fast detection of erroneous input; $\mathcal{FR}_6$: mutual authentication; $\mathcal{FR}_7$: session key agreement; $\mathcal{FR}_8$: user impersonation attack; $\mathcal{FR}_9$: gateway impersonation attack; $\mathcal{FR}_{10}$: device impersonation attack; $\mathcal{FR}_{11}$: privileged insider attack; $\mathcal{FR}_{12}$: forward secrecy; $\mathcal{FR}_{13}$: replay attack; $\mathcal{FR}_{14}$: man-in-the-middle attack; $\mathcal{FR}_{15}$: stolen verifier attack; $\mathcal{FR}_{16}$: stolen smart card attack; $\mathcal{FR}_{17}$: two/three factor authentication; $\mathcal{FR}_{18}$: local password change; $\mathcal{FR}_{19}$: local biometric update; $\mathcal{FR}_{20}$: dynamic sensor node addition; $\mathcal{FR}_{21}$: resistant to loss of temporary session secrets; $\mathcal{FR}_{22}$: user revocation; $\mathcal{FR}_{23}$: resilience against smart device physical capture attack; $\mathcal{FR}_{24}$: offline registration of IoT smart devices; $\mathcal{FR}_{25}$: formal security analysis; $\mathcal{FR}_{26}$: AVISPA tool-based formal verification.

comparison to that of [34]. This is an important consideration due to the resource limitation of IoT sensing devices. In addition, the computation costs needed for the resource-constrained sensing device is less than the four examined schemes.

B. Communication Costs Comparison

For communication cost computation, it is assumed that the timestamp is 32-bit long, hash digest (assuming SHA-1 hashing algorithm is applied) and identity are 160 bits each, random nonce is 128-bit long, and a ciphertext block (if AES-128 symmetric encryption is applied) is 128 bits. In the proposed scheme, three exchanged messages $M_1 = \{\text{EID}_i, \text{EID}_j, \text{TS}_i\}$, $M_2 = \{D_2, \text{TS}_{\text{gwn}}\}$, and $M_3 = \{D_1, D_3, \text{cert}, \text{TS}_j\}$ require $(\lceil(160+32)/128\rceil * 128 + \lceil(160+32)/128\rceil * 128 + 32) = 544$ bits, $(\lceil(160 + 512 + 32)/128\rceil * 128 + 32) = 800$ bits and $(512 + 256 + 160 + 32) = 960$ bits in the time of the login and authentication phase. The total communication overhead of the proposed scheme is then $(544+800+960) = 2304$ bits (288 bytes). In the proposed scheme, when the user is revoked, the size of D_1 changes to 640 bits, and consequently the total communication overhead becomes $(544+928+1088) = 2560$ bits (320 bytes). Table IV summarizes the communication costs and the number of messages exchanged for all schemes. We observe that the proposed scheme incurs less communication overhead as compared to the schemes in [3] and [34], and incurs similar overhead with the scheme in [28]. The communication overhead is minimal even for user revocation functionality support in our scheme, in comparison to the scheme in [34]. The scheme [33] incurs lower communication costs, but this is at the expense of reduced functionality and security features (see Table V).

C. Security and Functionality Features Comparison

Table V presents a comparative summary of the security and functionality features of the proposed scheme and the four other schemes examined here. It can be observed that the proposed scheme offers improved security and more functionality features, in comparison to the other four schemes. For example, while the scheme in [34] has comparable functionality and security features, our scheme has several advantages. Specifically, the scheme in [34] does not support user revocation, which is a fundamental feature since it is very likely that a smart card will be misplaced or stolen. Such a feature reduces the risk of an adversary compromising the system using a misplaced or stolen card. In addition, a user may need to be revoked due to resignation, change of role/duty, or disciplinary action. Thus, an explicit revocation of the smart card is necessary—a feature offered by our scheme. In the scheme in [34], the identities of the SDs are public, unlike our scheme. This reduces the potential attack vectors.

VII. PRACTICAL PERSPECTIVE: NS3 SIMULATION

We now attempt to quantify the performance of the proposed scheme, in terms of end-to-end delay (EED, in seconds) and network throughput (in bytes per second) using the widely accepted NS3 (3.28) simulator [54].

TABLE VI
SIMULATION PARAMETERS

Parameter	Description	
Platform	NS3(3.28) / Ubuntu 16.04 LTS	
Network scenarios	No. of users	No. of devices
1	5	20
2	8	15
3	8	20
4	5	35
5	8	35
6	10	35
7	8	50
Mobility	random (0-3 m/s)	
Simulation time	1200 sec	

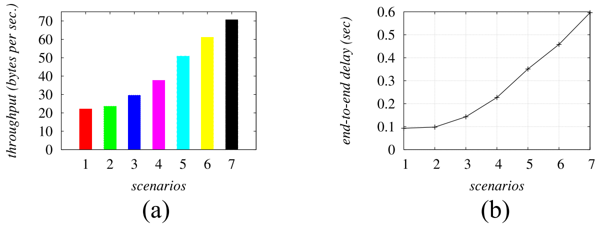


Fig. 5. (a) Throughput (bytes per second). (b) EED (seconds).

The simulation parameters are listed in Table VI. We used the Ubuntu 16.04 LTS platform for simulation. The sensing devices are randomly located in the range between 20 and 100 m away from GWN. The communication range of each sensing device is 50 m and the range of GWN is 200 m. The users are permitted to move randomly within a 150-m^2 area centered around GWN. The users and the devices communicate over the 2.4-GHz Wi-Fi media. We then simulated the IoT environment with different number of users and sensing devices, as listed in Table VI. Other parameters are taken as default parameters under the NS3 environment.

1) Network Throughput: Fig. 5(a) plots the graph of the network throughput for all seven scenarios. The different scenarios are plotted along the horizontal axis. Throughput is calculated as $(\nu_r \times |\rho|)/T_\delta$, where the total time in seconds is T_δ, a packet size is $|\rho|$, and the total received packets are ν_r. The simulation time is 1200 s, which is the same as the actual total time. It is observed that when there is an increase in the number of exchanged messages, there is also an increment in the network throughput in the network.

2) Impact on End-to-End Delay: Fig. 5(b) plots the graph of EED for all seven scenarios. EED can be formulated as $\sum_{i=1}^{\nu_p}(T_{\mathrm{rcv}_i} - T_{\mathrm{snd}_i})/\nu_p$, where T_{rcv_i} and T_{snd_i} are the time needed for receiving and sending a data packet i, respectively, and ν_p denotes the total number of packets. We also observe that the EED increases with the number of transmitted messages. This is primarily due to the increased number of messages, which results in congestion for the network.

VIII. CONCLUSION

In this paper, we presented a new lightweight anonymous user authentication protocol, designed for deployment in an IoT environment. The rigorous formal and informal security analysis on the proposed scheme demonstrated its security robustness. Evaluations using NS3 and a comparative summary also demonstrated its potential to be deployed in a real-world environment, although evaluation in a real-world environment for example implementing the protocol in a test sub-network remains one of our future research agenda.

ACKNOWLEDGMENT

The authors would like to thank the three anonymous reviewers and the Associate Editor for their constructive feedback.

REFERENCES

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.

[2] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, 2010.

[3] S. Challa *et al.*, "Secure signature-based authenticated key establishment scheme for future IoT applications," *IEEE Access*, vol. 5, pp. 3028–3043, 2017.

[4] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Proc. 8th Int. Workshop Theory Pract. Public Key Cryptography (PKC)*, vol. 3386. Les Diablerets, Switzerland, 2005, pp. 65–84.

[5] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Trans. Comput. Syst.*, vol. 8, no. 1, pp. 18–36, 1990.

[6] AVISPA. *Automated Validation of Internet Security Protocols and Applications*. Accessed: Mar. 2018. [Online]. Available: http://www.avispa-project.org/

[7] 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. Depend. Secure Comput.*, to be published. doi: 10.1109/TDSC.2017.2764083.

[8] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Advances in Cryptology (Eurocrypt'04)*. Interlaken, Switzerland: Springer, 2004, pp. 523–540.

[9] J. H. Cheon, J. Jeong, D. Kim, and J. Lee, "A reusable fuzzy extractor with practical storage size: Modifying Canetti *et al.*'s construction," in *Proc. Aust. Conf. Inf. Security Privacy (ACISP)*, vol. 10946. Wollongong, NSW, Australia, 2018, pp. 28–44.

[10] K.-K. R. Choo, *Secure Key Establishment* (Advances in Information Security), vol. 41. New York, NY, USA: Springer, 2009.

[11] S. Wu and K. Chen, "An efficient key-management scheme for hierarchical access control in e-medicine system," *J. Med. Syst.*, vol. 36, no. 4, pp. 2325–2337, 2012.

[12] R. Canetti. (2008). *Introduction to Cryptography. Lecture 9—Symmetric Encryption*. Accessed: Jul. 2018. [Online]. Available: http://www.cs.tau.ac.il/ canetti/f08-materials/scribe9.pdf

[13] S. Frankel, R. Glenn, and S. Kelly. (2013). *The AES-CBC Cipher Algorithm and Its Use With IPsec*. Accessed: Jul. 2018. [Online]. Available: http://tools.ietf.org/html/rfc3602

[14] A. K. Das, S. Zeadally, and D. He, "Taxonomy and analysis of security protocols for Internet of Things," *Future Gener. Comput. Syst.*, vol. 89, pp. 110–125, Dec. 2018.

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

[16] R. Canetti and H. Krawczyk, "Universally composable notions of key exchange and secure channels," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn. Adv. Cryptol. (EUROCRYPT)*, Amsterdam, The Netherlands, 2002, pp. 337–351.

[17] V. Odelu, A. K. Das, M. Wazid, and M. Conti, "Provably secure authenticated key agreement scheme for smart grid," *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 1900–1910, May 2018.

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

[19] R. Amin, S. K. H. Islam, N. Kumar, and K.-K. R. Choo, "An untraceable and anonymous password authentication protocol for heterogeneous wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 104, pp. 133–144, Feb. 2018.

[20] Y. Wei and H. Qiu, "A novel wireless authentication protocol preserving user anonymity and untraceability," in *Proc. Int. Conf. Commun. Technol.*, Guilin, China, 2006, pp. 1–4.

[21] R. Madhusudhan and R. C. Mittal, "Dynamic ID-based remote user password authentication schemes using smart cards: A review," *J. Netw. Comput. Appl.*, vol. 35, no. 4, pp. 1235–1248, 2012.

[22] D. He, N. Kumar, J. Chen, C.-C. Lee, N. Chilamkurti, and S.-S. Yeo, "Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks," *Multimedia Syst.*, vol. 21, no. 1, pp. 49–60, 2015.

[23] P. Gope and T. Hwang, "A realistic lightweight anonymous authentication protocol for securing real-time application data access in wireless sensor networks," *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 7124–7132, Nov. 2016.

[24] X. Li, J. Niu, S. Kumari, F. Wu, A. K. Sangaiah, and K.-K. R. Choo, "A three-factor anonymous authentication scheme for wireless sensor networks in Internet of Things environments," *J. Netw. Comput. Appl.*, vol. 103, pp. 194–204, Feb. 2018.

[25] J. Granjal, E. Monteiro, and J. S. Silva, "Security for the Internet of Things: A survey of existing protocols and open research issues," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 3, pp. 1294–1312, 3rd Quart., 2015.

[26] I. Makhdoom, M. Abolhasan, J. Lipman, R. P. Liu, and W. Ni, "Anatomy of threats to the Internet of Things," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1636–1675, 2nd Quart., 2019. doi: 10.1109/COMST.2018.2874978.

[27] P. Zhang, C. Lin, Y. Jiang, Y. Fan, and X. Shen, "A lightweight encryption scheme for network-coded mobile ad hoc networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 9, pp. 2211–2221, Sep. 2014.

[28] C.-C. Chang and H.-D. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 357–366, Jan. 2016.

[29] A. K. Das, S. Kumari, V. Odelu, X. Li, F. Wu, and X. Huang, "Provably secure user authentication and key agreement scheme for wireless sensor networks," *Security Commun. Netw.*, vol. 9, no. 16, pp. 3670–3687, 2016.

[30] C.-T. Li, T.-Y. Wu, C.-L. Chen, C.-C. Lee, and C.-M. Chen, "An efficient user authentication and user anonymity scheme with provably security for IoT-based medical care system," *Sensors*, vol. 17, no. 7, pp. 1–18, 2017.

[31] C.-T. Li, C.-C. Lee, C.-Y. Weng, and C.-M. Chen, "Towards secure authenticating of cache in the reader for RFID-based IoT systems," *Peer-to-Peer Netw. Appl.*, vol. 11, no. 1, pp. 198–208, 2018.

[32] N. Khalil, M. R. Abid, D. Benhaddou, and M. Gerndt, "Wireless sensors networks for Internet of Things," in *Proc. IEEE 9th Int. Conf. Intell. Sensors Sensor Netw. Inf. Process. (ISSNIP)*, Singapore, 2014, pp. 1–6.

[33] P. Porambage, C. Schmitt, P. Kumar, A. Gurtov, and M. Ylianttila, "Two-phase authentication protocol for wireless sensor networks in distributed IoT applications," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Istanbul, Turkey, 2014, pp. 2728–2733.

[34] 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.

[35] M. Turkanović, B. Brumen, and M. Hölbl, "A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion," *Ad Hoc Netw.*, vol. 20, pp. 96–112, Sep. 2014.

[36] Y. Jie, J. Y. Pei, L. Jun, G. Yun, and X. Wei, "Smart home system based on IoT technologies," in *Proc. 5th Int. Conf. Comput. Inf. Sci. (ICCIS)*, Shiyangzhen, China, 2013, pp. 1789–1791.

[37] T. Song, R. Li, B. Mei, J. Yu, X. Xing, and X. Cheng, "A privacy preserving communication protocol for IoT applications in smart homes," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1844–1852, Dec. 2017.

[38] R. Amin, N. Kumar, G. P. Biswas, R. Iqbal, and V. Chang, "A light weight authentication protocol for IoT-enabled devices in distributed Cloud Computing environment," *Future Gener. Comput. Syst.*, vol. 78, pp. 1005–1019, Jan. 2018.

[39] S. Challa, A. K. Das, P. Gope, N. Kumar, F. Wu, and A. V. Vasilakos, "Design and analysis of authenticated key agreement scheme in cloud-assisted cyber–physical systems," *Future Gener. Comput. Syst.*, to be published. doi: 10.1016/j.future.2018.04.019.

[40] W. Li, B. Li, Y. Zhao, P. Wang, and F. Wei, "Cryptanalysis and security enhancement of three authentication schemes in wireless sensor networks," *Wireless Commun. Mobile Comput.*, vol. 2018, pp. 1–12, Jul. 2018.

[41] P. K. Dhillon and S. Kalra, "Secure multi-factor remote user authentication scheme for Internet of Things environments," *Int. J. Commun. Syst.*, vol. 30, no. 16, pp. 1–20, 2017.

[42] Y. H. Chuang, N. W. Lo, C. Y. Yang, and S. W. Tang, "A lightweight continuous authentication protocol for the Internet of Things," *Sensors*, vol. 18, no. 4, pp. 1–26, 2018.

[43] M. A. Ferrag, L. A. Maglaras, H. Janicke, J. Jiang, and L. Shu, "Authentication protocols for Internet of Things: A comprehensive survey," *Security Commun. Netw.*, vol. 2017, p. 41, Nov. 2017.

[44] A. K. Sutrala, A. K. Das, N. Kumar, A. G. Reddy, A. V. Vasilakos, and J. J. P. C. Rodrigues, "On the design of secure user authenticated key management scheme for multigateway-based wireless sensor networks using ECC," *Int. J. Commun. Syst.*, vol. 31, no. 8, 2018, Art. no. e3514.

[45] A. K. Das, "Analysis and improvement on an efficient biometric-based remote user authentication scheme using smart cards," *IET Inf. Security*, vol. 5, no. 3, pp. 145–151, Sep. 2011.

[46] 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," *J. Netw. Comput. Appl.*, vol. 34, no. 1, pp. 73–79, 2011.

[47] D. Wang, H. Cheng, P. Wang, X. Huang, and G. Jian, "Zipf's law in passwords," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 11, pp. 2776–2791, Nov. 2017.

[48] S. Roy, A. K. Das, S. Chatterjee, N. Kumar, S. Chattopadhyay, and J. J. Rodrigues, "Provably secure fine-grained data access control over multiple cloud servers in mobile cloud computing based healthcare applications," *IEEE Trans. Ind. Informat.*, vol. 15, no. 1, pp. 457–468, Jan. 2019.

[49] V. Odelu, A. K. Das, and A. Goswami, "A secure biometrics-based multi-server authentication protocol using smart cards," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 9, pp. 1953–1966, Sep. 2015.

[50] M. K. Khan and K. Alghathbar, "Cryptanalysis and security improvements of 'two-factor user authentication in wireless sensor networks'", *Sensors*, vol. 10, no. 3, pp. 2450–2459, 2010.

[51] AVISPA. *SPAN, the Security Protocol Animator for AVISPA*. Accessed: Jan. 2018. [Online]. Available: http://www.avispa-project.org

[52] D. He, N. Kumar, M. K. Khan, and J.-H. Lee, "Anonymous two-factor authentication for consumer roaming service in global mobility networks," *IEEE Trans. Consum. Electron.*, vol. 59, no. 4, pp. 811–817, Nov. 2013.

[53] Q. Jiang, J. Ma, G. Li, and L. Yang, "An efficient ticket based authentication protocol with unlinkability for wireless access networks," *Wireless Pers. Commun.*, vol. 77, no. 2, pp. 1489–1506, 2014.

[54] (2018). *NS-3.28*. Accessed: Jul. 2018. [Online]. Available: http://www.nsnam.org/ns-3–28/