Array 15 (2022) 100210

Contents lists available at ScienceDirect

Array

journal homepage: www.sciencedirect.com/journal/array

Terminal independent security token derivation scheme for ultra-dense IoT networks

Vincent Omollo Nyangaresi

Faculty of Biological & Physical Sciences, Tom Mboya University College, Homabay, Kenya

ARTICLE INFO

Keywords:
5G
Authentication
Canetti- Krawczyk
Privacy
ECC
Biometrics
Security
IoT

ABSTRACT

The Fifth Generation (5G) networks deploy base station ultra-densification to boost data rates, capacities, reliability, energy efficiency as well as the reduction of communication latencies. To increase quality of service as well as quality of experience, a large number of Internet of Things (IoT) communications are relayed over 5G networks. For enhanced pervasive computing, most of the devices in 5G-IoT networks are continuously connected to the network, exchanging massive and sensitive data. Therefore, there is need to protect these networks from both privacy and security attacks. As a result, many security protocols have been presented in literature. Unfortunately, IoT devices are heterogeneous in nature with diverse communication and security architectures. These issues render privacy and security protection extremely challenging. Consequently, majority of the conventional protocols fail to fully address privacy and security issues in 5G-IoT networks. Particularly, user collusion, de-synchronization and side-channeling attacks are ignored in most of the security protocols. On the other hand, some of the developed protocols achieve salient security but at extremely high computation, storage and communication complexities. In this paper, an elliptic curve and biometric based security token derivation scheme is presented. Formal security analysis using Burrows–Abadi–Needham (BAN) logic shows the negotiation of a session key between the communicating parties. On the other hand, informal security analysis shows that this scheme is secure under all the Canetti- Krawczyk (CK) threat model assumptions. In terms of efficiency, the comparative performance evaluation carried out shows that this protocol has the least communication and computation complexities among other related protocols.

1. Introduction

The Internet of Things (IoT) comprises of numerous smart devices that are linked together via the internet. The sensors installed in these IoT devices collect data from the environment and transmit the same data to other devices or their operators [1]. As such, these devices have found applications in a wide range of fields such as in intelligent transportation, military, industrial sector, healthcare, smart grids, vehicular communication, smart homes, environmental monitoring and smart cities [2]. To enhance Quality of Service (QoS) and satisfy different user requirements, IoT communications are relayed over Fifth Generation (5G) networks. This is due to the extremely low latencies, high energy efficiency, enhanced capacities, reliability, high speeds as well as flexibility of the 5G networks. As such, 5G- IoT based pervasive connectivity overcomes issues such as network resource management and slow response times. As explained in Ref. [3], 5G-IoT has revolutionized healthcare management through remote patient diagnosis, treatment and monitoring.

Although 5G-IoT networks play critical roles in people's lives, many vulnerabilities lurk in these networks. As such, the collected and transmitted data is exposed to numerous storage and security threats [4]. For instance, remote access of medical data over open wireless channels [5] introduces risks and challenges regarding the preservation of confidentiality, integrity and security [3]. In military and civilian applications, Internet of Drones (IoD) has been deployed to offer reconnaissance, remote training and remote process monitoring. However, during communication with other drones or control rooms in ground stations, security and privacy have been noted to be serious challenges [6]. In smart grids, 5G networks have been crucial in the digitization of power grids to offer higher speeds, low latencies and enhanced reliability [7]. Unfortunately, the deployment of these public 5G networks introduces numerous privacy and security risks to the smart grid infrastructure. As explained in Ref. [8], majority of the security and privacy issues in 5G-IoT can be attributed to pervasive and continuous connection of the smart devices to the network. As such, these devices are susceptible to Denial of Service (DoS), impersonation,

E-mail address: vnyangaresi@tmuc.ac.ke.

https://doi.org/10.1016/j.array.2022.100210
Received 11 January 2022; Received in revised form 16 June 2022; Accepted 21 June 2022
Available online 25 June 2022

packet replays, repudiation, traceability, eavesdropping and Man-in-the-Middle (MitM) attacks.

It is evident that 5G-IoT networks collect massive data that need proper protection. This is critical due to the private and sensitive nature of this data. As explained in Ref. [9], IoT message exchanges lack privacy and security protection. As such, authentication and device location identification are crucial in preventing network invasion by malicious devices [10]. To this end, 5G Authentication and Key Agreement protocol (5G AKA) has been deployed to offer identity privacy. In addition, this AKA protocol helps address fake base station attacks which are common in the Fourth Generation (4G) networks. However, traceability attacks against IoT devices still remains unresolved [11] in 5G networks.

1.1. Research motivation

The need to improve QoS and security in IoT networks has led to the adoption of 5G networks for message relay among the devices. The high capacities offered by 5G networks has facilitated massive private and sensitive data exchanges among the 5G-IoT devices, which can have devastating effects if compromised by adversaries. In addition, base station ultra-densification in 5G implies frequent handovers and hence frequent authentications to curb attacks. Owing to the resource constrained nature of IoT devices, conventional security protocols with high computation, communication and storage complexities are unsuitable in this communication environment. Another challenge in 5G-IoT networks is the device heterogeneity in which communication and security architectures differ from one device to the other. As such, the attainment of high levels of security and privacy protection at optimum levels of QoS is necessary but challenging. As such, there is need for more efficient authentication and key exchange protocols.

1.2. Threat model

Most of the communication in 5G-IoT is over insecure wireless public channels. As such, the exchanged messages are susceptible to a myriad of privacy and security attacks. In this insecure communication environment, the attacker capabilities are better modeled using the Canetti-Krawczyk (CK) threat model. Under this model, an adversary Å is assumed to have the following abilities:

- Can eavesdrop the data exchanged in 5G-IoT environment
- Is capable of intercepting and modifying the exchanged data
- Can insert bogus messages into the communication channel
- Has capabilities of deleting some of the transmitted messages
- Can compromise secret security tokens such as private keys, session states information and session keys
- Is able to physically capture the IoT devices and extract stored secrets through power analysis

Under the above assumptions, adversary Å can launch attacks such as impersonation, packet replays, side-channeling, man-in-the-middle, privileged insider and ephemeral secret leakages.

1.3. Research contributions

Many privacy and security preservation protocols have been developed, based on technologies such as passwords, smart cards, public key infrastructure and blockchains. However, most of these protocols have serious privacy and security vulnerabilities. In addition, some of these schemes have high complexities which are unsuitable for IoT devices. To this end, the contributions of this paper are as follows:

- A scheme that leverages on elliptic curve cryptography and biometrics is developed to directly authenticate the communicating entities devoid of a central authority. This potentially eliminates single point of failure issues.
- An authorization mechanism is implemented using some security tokens for access and membership validation, as well as admittance right groups. This serves to prevent user collusion and privileged insider attacks.
- Formal security analysis is carried out using the widely accepted BAN logic, which demonstrates the existence of a session key between the communicating entities. Informal security analysis is also executed to show that this scheme is secure under all the Canetti-Krawczyk (CK) threat model assumptions.
- In terms of efficiency, comparative performance evaluation is carried out to show that the proposed protocol has the least computation and communication complexities.

The rest of this paper is organized as follows: Section 2 discusses related work, while Section 3 describes the system model of the proposed scheme. On the other hand, Section 4 presents security analysis of the proposed scheme while Section 5 presents the comparative performance evaluation of this protocol. Towards the end of this paper, Section 6 concludes the paper and gives future research directions.

2. Related work

Over the recent past, many security and privacy preservation protocols have been presented in literature. For instance, a cross-layer authentication scheme is developed in Ref. [12] for ultra-dense 5G networks. However, this protocol has scalability issues since it is evaluated in a very limited scenario. Based on Elliptic Curve Cryptography (ECC) and hash functions, a Device to Device (D2D) authentication scheme is introduced in Ref. [13]. Unfortunately, this protocol has extensive communication and computation overheads. In addition, it fails to offer untraceability and device anonymity. The three-factor authentication protocol in Ref. [14] is lightweight and hence can address the issues in Ref. [13]. However, the scheme in Ref. [14] cannot provide backward and forward key secrecy. In addition, its design fails to consider collusion, replay and offline password guessing attacks. Similarly, the anonymous multi-server authentication protocol in Ref. [15] can solve anonymity challenges in Ref. [13]. Unfortunately, the repeated entry of unique identity during login sessions can potentially lead to loss of user untraceability in Ref. [15]. To offer privacy protection, many blockchain-based schemes have been developed. For example, a chameleon hash functions and blockchain based protocol is introduced in Ref. [16], while a homomorphic encryption technique based on blockchains is presented in Ref. [17]. Similarly, a blockchain-based data management scheme is developed in Ref. [6] for IoD communications, while authentication techniques based on blockchain are presented in Refs. [18,19] for 5G ultra-dense networks. Moreover, a consortium blockchain based technique is introduced in Ref. [20], while an authentication scheme using blockchains is developed in Ref. [10] for smart city 5G-IoT communication. However, blockchain technology can potentially lead to high storage and computation complexities [21].

To solve performance issues in blockchain-based protocols, the efficient and secure scheme in Ref. [22] can be deployed. Unfortunately, this protocol cannot withstand side channeling attacks and it fails to offer mutual authentication [6]. To address authentication issues in Ref. [22], the scheme in Ref. [23] has been developed. Although this scheme is secure and resists many attacks, it cannot offer forward key secrecy and is still vulnerable to privileged insider attacks. To solve forward key secrecy challenges in Ref. [23], the ECC and quantum cryptography based scheme in Ref. [24] can be deployed to encipher information exchanged between devices and 5G base stations. Although this protocol attains non-repudiation, confidentiality, availability and integrity in 5G-IoT environment, its real feasibility and performance evaluation are missing. To offer session key negotiation and access

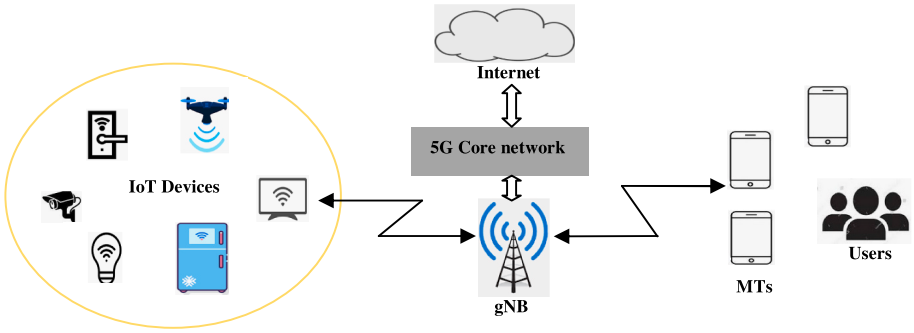

Fig. 1. Network architecture.

control, the protocol in Ref. [25] is introduced. Although this approach is resilient against replay and impersonation attacks, it cannot offer mutual authentication and protection against side-channeling attacks [6]. To address mutual authentication challenges in Ref. [25], the public key based IoT authentication protocol is introduced in Ref. [26]. However, the deployed Public Key Infrastructure (PKI) results in high computation and communication overheads [27]. Similarly, the PKI-based scheme in Ref. [28] has performance issues occasioned by key management challenges.

To address high computation and communication challenges in PKI based protocols, the lightweight ECC based batch authentication protocol in Ref. [29] and chaotic map based protocols such as the one in Ref. [30] can be utilized. Unfortunately, the protocol in Ref. [30] fails to offer user anonymity and cannot withstand privileged insider attacks [4]. On its part, the protocol in Ref. [29] deploys trusted authority which presents a potential single point of failure [31]. Since the protocol in Ref. [32] is based on simple hash function, it is efficient and can therefore solve performance issues in PKI based schemes. Unfortunately, this scheme cannot provide anonymity and protection against impersonation attacks [33]. To curb these attacks, trusted authority based schemes in Refs. [34,35] can be utilized. However, these protocols are vulnerable to single point of failure just like the scheme in Ref. [29]. To protect against message non-repudiation during discovery and transmission phases of D2D communication, an identity and ECC based protocol in Ref. [36] is presented. However, identity based schemes have key escrow issues [37]. Although the protocol in Ref. [38] addresses this challenge, it fails to provide device anonymity and protection against privileged insider attacks. In addition, it has extensive communication and computation overheads [10]. Side–channeling attacks are serious challenges in IoT environments. This is because most of these devices such as drones are deployed in insecure environments and hence prone to physical captures. As such, Physical Unclonable Functions (PUF) based authentication protocols have been introduced. For instance, a PUF-based scheme is introduced in Ref. [39] for IoT devices. However, this scheme fails to provide user and device anonymity as well as untraceability. In addition, PUF-based schemes have stability issues [40]. To address these stability issues, authentication protocols in Refs. [41,42] have been introduced. However, the approach in Ref. [41] has extensive communication costs. It also fails to consider collusion and de-synchronization attacks in its design. On its part, the protocol in Ref. [42] cannot protect against user tracking and privileged insider attacks [43]. Consequently, the authors in Ref. [43] have developed an anonymous three-factor authentication scheme to address these flaws.

The discussions above clearly show that most of the current IoT security protocols fail to provide some crucial security and privacy features, while others have very high communication and computation complexities. To address some of these challenges, this paper presents a lightweight authentication, authorization and key agreement scheme based on ECC and biometrics. This protocol is shown to offer mutual authentication, session key agreement, anonymity, untraceability as well as backward and forward key secrecy. In addition, the proposed scheme is demonstrated to be resilient against majority of the 5G-IoT attacks such as side-channeling, privileged insider, impersonation, de-synchronization, collusion, stolen verifier, MitM, ephemeral secret leakages (ESL), replay, known secret key and offline password guessing. Moreover, this scheme is shown to have the least communication and computation complexities among other related protocols.

3. System model

In this section, the mathematical preliminaries, network model as well as the proposed scheme are described.

3.1. Mathematical preliminaries

This section presents some mathematical formulations of the cryptographic primitives deployed in this paper. These include the elliptic curve discrete logarithmic problem, elliptic curve computational Diffie-Hellman problem, fuzzy extractor probabilistic and deterministic algorithms.

3.1.1. Elliptic curve cryptography

Suppose that a and b are points on an elliptic curve E. Then, the equation of this elliptic curve is written as $y^2 = x^3 + ax + b \bmod P$, where P is a large prime number. The finite field over this elliptic curve is denoted as F_q, where q is another large prime number. As such, $a, b \in F_q$ satisfy the condition that $4a^3 + 27b^2 \neq 0 \bmod q$. Here, the cyclic group of order s is denoted as $G_s = P$, where s is another prime number. Under these conditions, the following definitions hold:

Definition 1. In Elliptic Curve Discrete Logarithmic Problem (ECDLP), an adversary needs to find a when provided with point $M \in G_s$. Here, $a \in Z_s^*$ and $M = aP$.

Definition 2. In Elliptic Curve Computational Diffie-Hellman Problem (ECCDHP), an attacker is required to find abP when provided with both aP and bP, where $a, b \in Z_s^*$.

3.1.2. Fuzzy extraction

Suppose that ε_i is an arbitrary string, β is the user biometric and ν_i is some auxiliary or helper string. Here, string ε_i can be extracted from biometric template β by the fuzzy extractor in an error-free manner. Provided that some other templates of biometric β^* remain close to β, then string ε_i remains constant with the use of helper string ν_i. In essence, each Fuzzy Extractor (FE) has two probabilistic and deterministic algorithms *Rep (.)* and *Gen (.)*. Under this conditions, the following hold:

Gen $(\beta) = (\varepsilon_i, \nu_i)$*:* Given some biometric template β as the input to the fuzzy extractor, the probabilistic algorithm *Gen (.)* outputs some secret biometric key ε_i and auxiliary string ν_i.

Table 1
Notations.

Symbol	Description
P_{FF}	Prime finite field
q	Large prime number
DV_i, DV_j	Device i & device j
ID_U, PW_U	User unique identity and password
P_{VK}	gNB private key
P_{UK}	gNB public key
ID_g	gNB identity
M_{VT}	Membership validation token
A_{VT}	Access validation token
$\mathbb{N}_i$	Random number
h(.)	Hashing operation
M_A	Admittance right mask
ΔFE	Fuzzy extraction error tolerance limit
IDV_i, IDV_j	Unique identity for DV_i & DV_j
SK_{g-D_i}	Shared key between the gNB and DV_i
SK_{g-D_j}	Shared key between the gNB and DV_j
TS_i	Timestamp i
$\mathbb{Z}_K$	Session key
ΔTS	Maximum delay tolerance
Γ	User temporary identity
||	Concatenation operation
⊕	XOR operation

Rep $(\beta^*, \nu_i) = (\varepsilon_i)$: Provided with a noisy biometric template β^* and some auxiliary string ν_i as inputs to fuzzy extractor, then the deterministic algorithm *Rep (.)* serves to reproduce the biometric key ε_i.

Basically, the FE serves to extract biometric information from some biometric template as a random string with some error tolerance ΔFE. In addition, the fuzzy extractor outputs some public string as auxiliary information. With the help of this public string, FE outputs the same random string in the presence of minor change in the input.

3.2. Network model

The network architecture in the proposed scheme comprises of IoT devices DV_i, the 5G base station gNB, the users and the Mobile Terminals (MTs). Fig. 1 presents the interaction model among all these communicating entities, while Table 1 presents the notations used in this paper. In this network, the users deploy their MTs to access the IoT devices. Here, these devices can be drones, smart TVs, bulbs, cameras, doors, fridges among others.

On the other hand, 5G offers the backbone through which the messages are exchanged among the IoT devices as well as the users. As shown in Fig. 1, the 5G gNB base station connects with the 5G core network, consisting of routers and gateways among other components.

On its part, the 5G core network connects to the internet, which comprises of elements such as clouds, data centers and servers. As such, any requested service in this 5G-IoT environment is relayed through the gNB. This base station may need to connected to the data centers and servers located in the internet to obtain some of the services and data. The routers and gateways in the core network are critical for these internet connections.

3.3. The proposed scheme

The four phases that make up the proposed scheme include the initialization phase; user registration; login, authentication and key negotiation (LAK); and parameter update phase. The specific details of these phases are described in the sub-sections below.

3.3.1. Initialization phase

All the IoT devices are required to obtain security tokens from the gNB before they could transmit any packets with other devices. To accomplish, the following five steps are executed.

Step 1: The gNB selects its identity ID_g and elliptic curve E whose prime finite is P_{FF}. In addition, it chooses elliptic curve sub-group S_G whose generator P is of order q. This is followed by the generation of gNB's private key $P_{VK} \in Z_q^*$. Next, it derives its public key as $P_{UK} = (P. P_{VK})$ as shown in Fig. 2.
Step 2: The gNB selects some IoT membership validation token M_{VT}, access rights validation token A_{VT} and some collision-resistant one-way hashing function *h(.)*. It then publishes parameter set {E, S_G, q, P} to all IoT devices.
Step 3: Based on some network sanction policies, the gNB generates indices for various IoT admittance right groups (ARGs) as SPG_1, SPG_q, ….SPG_N, where N is the number of ARGs. It then generates random nonce $\mathbb{N}_1$ and IoT admittance right mask M_A for each IoT group.
Step 4: The gNB generates unique identity IDV_i for DV_i. Next, it derives key SK_{g-D_i} that is shared between the gNB and DV_i. Similarly, it generates unique identity IDV_j for DV_j, followed by the derivation of key SK_{g-D_j} shared between the gNB and DV_j.

The gNB then securely stores parameter set {IDg, P_{UK}, IDV_i, SK_{g-D_i}} in DV_i memory. Similarly, it securely stores parameter set {IDV_i, IDV_j, SK_{g-D_j}} in DV_j memory.

Step 5: The gNB stores parameter set {IDg, P_{UK}, IDV_i, SK_{g-D_i}, IDV_j, SK_{g-D_j}} in its database. Finally, IoT DV_i and DV_j are deployed in the area of interest.

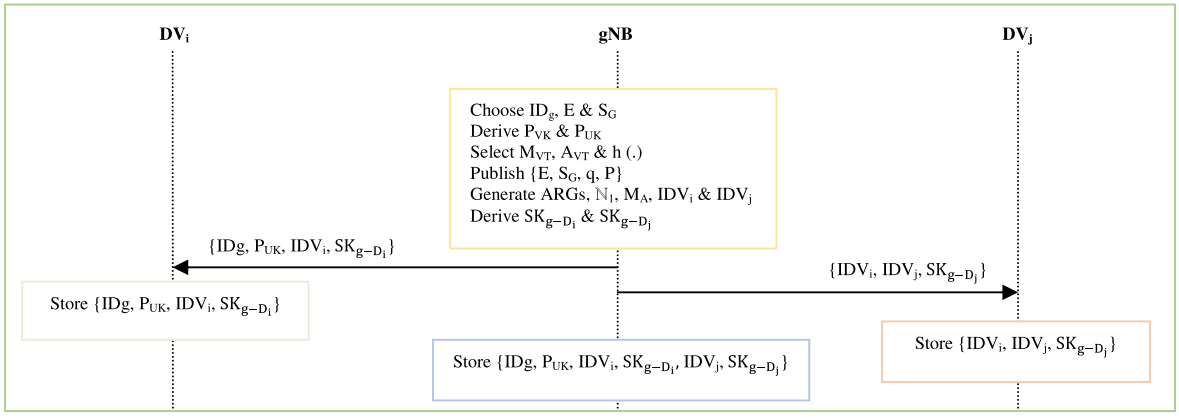

Fig. 2. Initialization phase.

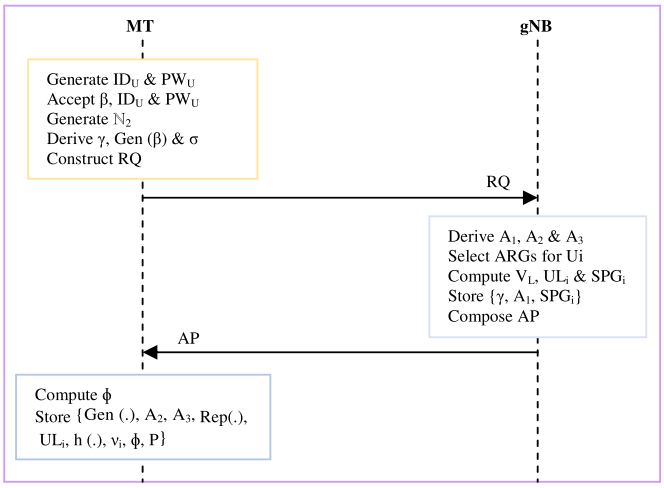

Fig. 3. User registration phase.

3.3.2. *User Registration Phase*

This phase commences by having the user U_i transmit registration request to the gNB over some private channels. This is a three-step process as elaborated below.

Step 1: The user generates unique identity ID_U and password PW_U. Next, user biometric data β, ID_U and PW_U are input to the user's mobile terminal MT. This is followed by the generation of random nonce $\mathbb{N}_2$ and the derivation of temporary identity $\gamma = h(ID_U||\mathbb{N}_2)$, Gen $(\beta) = (\varepsilon_i, \nu_i)$ and security parameter $\sigma = h(PW_U||\varepsilon_i)$. Lastly it transmits registration request $RQ = \{\gamma, \sigma\}$ to the gNB as shown in Fig. 3.

Step 2: Upon receiving message RQ from the user's MT, the gNB derives parameters $A_1 = h(\gamma||ID_g||M_{VT})$, $A_2 = h(\gamma||\sigma) \oplus A_1$ and $A_3 = h(\sigma||A_1)$. The gNB then chooses some ARGs suitable for this user. This is basically the qth and $(q + k)^{th}$ rights for the user.

Afterwards, it uses its A_{VT} to derive association value between γ and SPG_q as $V_L = h(A_1||A_{VT}||\mathbb{N}_1)$, as well as user authorization list $UL_i = \{(SPG_q, V_L), (SPG_{q+k}, B_{L+2})\}$. It then transmits associative parameters $AP = \{A_2, h(.), A_3, UL_i, P, P_{UK}\}$ to the user over some private channels. Lastly, the gNB derives $SPG_i = \{SPG_q, SPG_{q+k}, \ldots\}$ and stores parameter set $\{\gamma, A_1, SPG_i\}$ in its database.

Step 3: After getting AP from the gNB, the user's MT derives parameter $\phi = \mathbb{N}_2 \oplus h(ID_U||\varepsilon_i)$. Finally, the MT store parameter set {Gen (.), A_2, A_3, Rep (.), UL_i, h (.), ν_i, ϕ, P} in its memory.

3.3.3. *Login, authentication and key negotiation phase*

This phase is triggered whenever the user wants some access to the IoT devices, for instance to obtain the collected data. To accomplish this, the following seven steps are executed among the MT, gNB and the IoT devices. After successful mutual authentication, the MT and the IoT devices negotiate a session key for secure packet exchanges. Thereafter, the user can access real-time IoT sensed data that correspond to user access rights.

Step 1: Suppose that the user is interested in accessing real-time data in DV_i. To accomplish this, the user inputs ID_U, PW_U and β to the MT

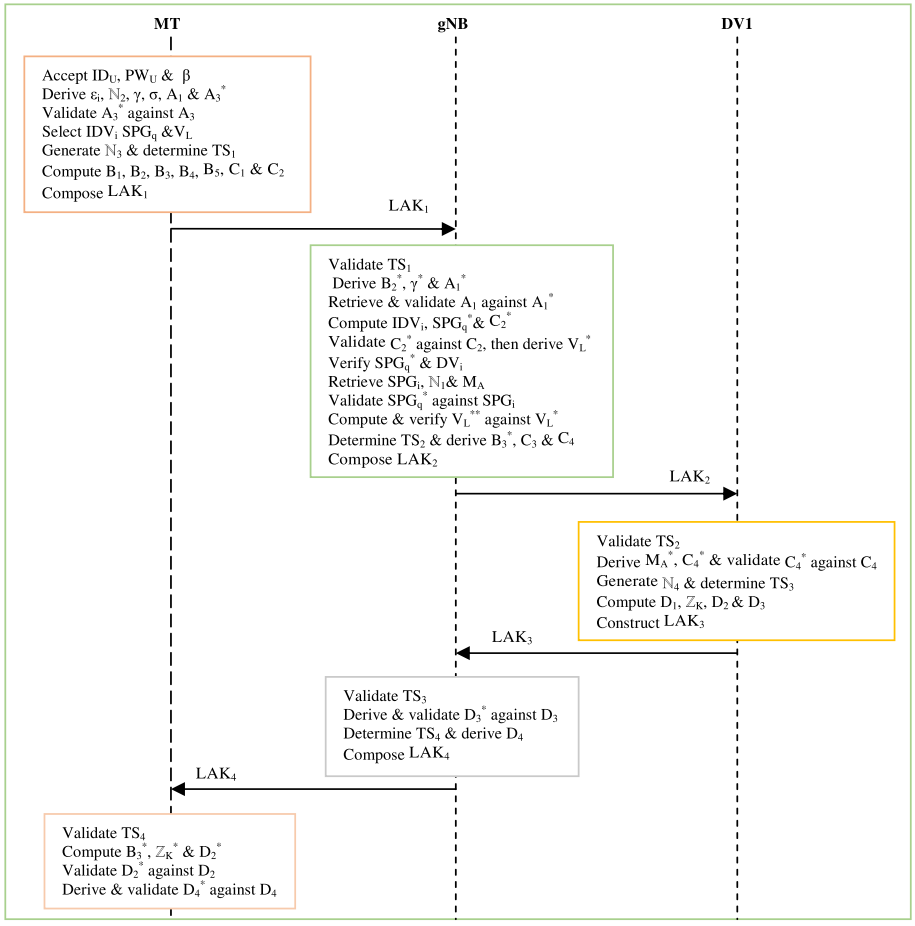

Fig. 4. Login, Authentication and Key negotiation Phase.

which then utilizes stored parameters to derive $\varepsilon_i = Rep(\beta, \nu_i)$, $\mathbb{N}_2 = \phi \oplus h(ID_U||\varepsilon_i)$, $\gamma = h(ID_U||\mathbb{N}_2)$, $\sigma = h(PW_U||\varepsilon_i)$, $A_1 = A_2 \oplus h(\gamma||\sigma)$ and $A_3^* = h(A_1 \oplus h(\gamma||PW_U||\varepsilon_i))$. Next, it checks whether $A_3^* \stackrel{?}{=} A_3$ such that the login request is rejected if the two do not match. Otherwise, the MT chooses IDV_i and retrieves the corresponding SPG_q and V_L from the stored UL_i.

Step 2: The MT generates random nonce $\mathbb{N}_3 \in Z_q^*$ and determines current timestamp TS_1 that it uses to compute parameters $B_1 = \mathbb{N}_3 P$, $B_2 = \mathbb{N}_3 P_{UK}$, $B_3 = \gamma \oplus h(B_1||B_2)$, $B_4 = IDV_i \oplus h(B_2||TS_1)$, $B_5 = SPG_q \oplus h(A_1||TS_1)$, $C_1 = V_L \oplus h(\gamma||TS_1)$ and $C_2 = h(\gamma||IDV_i||SPG_q||A_1||B_1||B_2||TS_1)$. Finally the MT constructs login request $LAK_1 = \{B_1, B_3, B_4, B_5, C_1, C_2, TS_1\}$ that it transmits to the gNB over pubic channels as shown in Fig. 4.

Step 3: After obtaining LAK_1 from the MT, the gNB validates TS_1 against the delay tolerance value ΔTS. Provided that LAK_1 passes the freshness checks, the gNB derives parameters $B_2^* = B_1 P_{VK}$, $\gamma^* = B_3 \oplus h(B_1||B_2^*)$ and $A_1^* = h(\gamma^*||ID_g||M_{VT})$. Next, it uses γ^* to retrieve the user's membership A_1 from its database. This is followed by the confirmation of whether this user is a member of this particular gNB. This is achieved by confirming whether $A_1^* \stackrel{?}{=} A_1$. Provided that these two values are not identical, the gNB rejects the user's login request. Otherwise, the gNB computes $IDV_i = B_4 \oplus h(B_2^*||TS_1)$, $SPG_q^* = B_5 \oplus h(A_1^*||TS_1)$ and $C_2^* = h(\gamma^*||IDV_i||SPG_q^*||A_1^*||B_1||B_2||TS_1)$.

It then checks if $C_2^* \stackrel{?}{=} C_2$ such that the login request is rejected when these two values differ. Otherwise, the gNB derives parameter $V_L^* = C_1 \oplus h(\gamma^*||TS_1)$.

Step 4: The gNB confirms whether SPG_q^* submitted by the user matches the data access rights of IoT devices DV_i. If this is the case, the gNB retrieves SPG_i, random nonce $\mathbb{N}_1$ and M_A for this particular SPG_q^*. Next, it validates that this SPG_q^* belongs to SPG_i. If this is the case, it computes $V_L^{**} = h(A_1^*||A_{VT}||\mathbb{N}_1)$. To confirm that this particular user has rights of SPG_q^*, the gNB checks if $V_L^{**} \stackrel{?}{=} V_L^*$. Provided that this is not the case, the gNB knows that this user is not permitted to access IoT DV_i. As such, it sends access reject response to this user. Otherwise, the gNB determines the current timestamp TS_2 and derives $B_3^* = h(\gamma^*||IDV_i^*||B_2^*)$, $C_3 = M_A \oplus h(IDV_i^*||SK_{g-D_i}||TS_2)$ and $C_4 = h(B_3^*||IDV_i^*||M_A||B_1||SK_{g-D_i}||TS_2)$. Finally, it composes authentication message $LAK_2 = \{B_3^*, C_3, B_1, C_4, TS_2\}$ that it transmits over to DV_i.

Step 5: On receiving message LAK_2 from the gNB, DV_i checks the freshness of timestamp TS_2 using ΔTS and terminates the request if LAK_2 fails the freshness checks. Otherwise, DV_i derives $M_A^* = C_3 \oplus h(IDV_i^*||SK_{g-D_i}||TS_2)$ and $C_4^* = h(B_3^*||IDV_i^*||M_A||B_1||SK_{g-D_i}||TS_2)$. This is followed by the confirmation of whether $C_4^* \stackrel{?}{=} C_4$. Here, the session is terminated if these values are not equivalent. Otherwise, DV_i generates random nonce $\mathbb{N}_4$ and determine current timestamp TS_3. Next, it computes $D_1 = \mathbb{N}_4 P$, session key $\mathbb{Z}_K = h(B_3^*||\mathbb{N}_3 D_1)$, $D_2 = h(ID_g||IDV_i||\mathbb{Z}_K)$ and $D_3 = h(B_3^*||IDV_i||D_1||D_2||SK_{g-D_i}||TS_3)$. Lastly, it composes authentication message $LAK_3 = \{\mathbb{N}_4, D_2, D_3, TS_3\}$ that it forwards to the gNB.

Table 2
BAN Logic notations.

Notation	Meaning
#(G)	G is fresh
K◁ G	K sees G
$\langle G \rangle_R$	G is combined with secret R
K|≡ G	K believes in statement G
K| ~ G	K once said statement G
K $\leftrightarrow^R$ S	*K* and *S* share secret key *R* for communication with each other
K|⇒ G	*K* has jurisdiction over formula *G*

Step 6: On receiving message LAK_3 from the DV_i, the gNB validates the freshness of timestamp TS_3 using delay tolerance threshold ΔTS. Basically, the session is terminated if LAK_3 fails the freshness checks. Otherwise, the gNB computes $D_3^* = h(B_3^*||IDV_i||D_1||D_2||SK_{g-D_i}||TS_3)$ and confirms whether $D_3^* \stackrel{?}{=} D_3$. Here, the session is terminated when these two values are dissimilar. Otherwise, the gNB determines the current timestamp TS_4 and derives $D_4 = h(\gamma^*||IDV_i||A_1^*||D_1||D_2||B_2^*||TS_4)$. Finally, it constructs authentication message $LAK_4 = \{D_1, D_3, D_4, TS_4\}$ that is then transmitted over to the user's MT.

Step 7: After receiving message LAK_4 from the gNB, the MT validates timestamp TS_4 such that the session is terminated if this check fails. Otherwise, the MT derives $B_3^* = h(\gamma||IDV_i||B_2)$, session key $\mathbb{Z}_K^* = h(B_3^*||\mathbb{N}_3 D_1)$ and $D_2^* = h(ID_g||IDV_i||\mathbb{Z}_K)$. It then checks if $D_2^* \stackrel{?}{=} D_2$ such that the session is terminated whenever these values are dissimilar. Otherwise, the user trusts that this session key is shared with the legitimate DV_i. Afterwards, the MT derives $D_4^* = h(\gamma||IDV_i||A_1^*||D_1||D_2||B_2||TS_4)$ and checks if $D_4^* \stackrel{?}{=} D_4$. Basically, the session is terminated when these two parameters are unequal. Otherwise, the gNB, and DV_i are authenticated by the user through the MT and share a session key $\mathbb{Z}_K$ with DV_i. Similar login, authentication and key negotiation procedures are followed for DV_j or any other IoT device that the user may wish to establish a communication session with.

3.3.4. Parameter update phase

This phase is triggered whenever the user wants to update password or biometric data. This may be occasioned by the suspicion that these parameters have been compromised by an adversary. To reduce communication overheads, this update is executed locally devoid of gNB involvement. It may also be important that admission rights accorded to the user be changed when there are changes in policies. This change of admission rights is executed between the user and the gNB over some private channels. This is a five-step process as described below.

Step 1: The user inputs identity ID_U as well as old password PW_U. Next, biometric data β is imprinted on the MT after which it derives $\varepsilon_i = Rep(\beta, \nu_i)$, $\mathbb{N}_2 = \phi \oplus h(ID_U||\varepsilon_i)$, $\gamma = h(ID_U||\mathbb{N}_2)$, $\sigma = h(PW_U||\varepsilon_i)$, $A_1 = A_2 \oplus h(\gamma||\sigma)$ and $A_3^* = h(\sigma||A_1)$.

Step 2: The MT checks if $A_3^* \stackrel{?}{=} A_3$ such that the update is terminated. This is because it implies that at least one of the authentication parameters is invalid. Otherwise, the MT prompts the user to input new password and biometrics PW_U^{New} and β^{New} respectively. Afterwards, it derives $Gen(\beta^{New}) = (\varepsilon_i^{New}, \nu_i^{New})$, $\phi^{New} = \mathbb{N}_2 \oplus h(ID_U||\varepsilon_i^{New})$, $\sigma^{New} = h(PW_U^{New}||\varepsilon_i^{New})$, $A_2^{New} = A_1 \oplus h(\gamma||\sigma^{New})$ and $A_3^{New} = h(\sigma^{New}||A_1)$. Next, the MT substitutes parameter set $\{A_2, A_3, \nu_i, \phi\}$ with its updated equivalent $\{A_2^{New}, A_3^{New}, \nu_i^{New}, \phi^{New}\}$.

Step 3: To execute admission rights change, the gNB transmits an update request UR together with γ and new admission right list UL_i^* to the MT. This request basically prompts the user that the current admission rights need to be refreshed.

Step 4: After getting request UR from the gNB, the user inputs ID_U, PW_U and β to the MT. using the stored values in its memory, the MT validates this login attempt as described in the login, authentication and key negotiation phase. Thereafter, it substitutes stored UL_i with UL_i^* before sending a response message together with parameter A_1 to the gNB. Basically, this message serves to inform the gNB that this update is complete.

Step 5: Upon receiving this message, the gNB validates the received user parameter A_1. Here, the gNB substitutes the stored SPG_i with refreshed the refreshed version SPG_i^{New}. Finally, the current admittance right groups ARGs is also replaced with its refreshed one $ARGs^{New}$ for this particular user.

4. Security analysis

In this section, formal security analysis is executed to show the

Table 3
BAN Logic rules.

Rule	Description
$\frac{K\mid\equiv S\Rightarrow G,\quad K\mid\equiv S\mid\equiv G}{K\mid\equiv G}$	JR: Jurisdiction rule
$\frac{K\mid\equiv K\leftrightarrow^{R}S, K\triangleleft\langle G\rangle_{R}}{K\equiv S\mid\sim G}$	MMR: Message meaning rule
$\frac{K\mid\equiv\#(G)}{K\mid\equiv\#(G,H)}$	FPR: Fresh promotion rule
$\frac{K\mid\equiv\#(G), K\mid\equiv S\mid\sim G}{K\mid\equiv S\mid\equiv G}$	NVR: Nonce verification rule

Table 4
Initial assumptions.

Assumption	Descriptions
IA_1	$gNB\mid\equiv\#\ (TS_1)$
IA_2	$DV_i\mid\equiv\#\ (TS_2)$
IA_3	$gNB\mid\equiv\#\ (TS_3)$
IA_4	$MT\mid\equiv\#\ (TS_4)$
IA_5	$MT\mid\equiv(MT\leftrightarrow^{A_1}gNB)$
IA_6	$gNB\mid\equiv(MT\leftrightarrow^{A_1}gNB)$
IA_7	$DV_i\mid\equiv(DV_i\leftrightarrow^{SK_{g-D_i}}gNB)$
IA_8	$gNB\mid\equiv(DV_i\leftrightarrow^{SK_{g-D_i}}gNB)$
IA_9	$MT\mid\equiv DV_i\mid\equiv(MT\leftrightarrow^{\mathbb{Z}_K}DV_i)$
IA_{10}	$DV_i\mid\equiv MT\mid\equiv(MT\leftrightarrow^{\mathbb{Z}_K}DV_i)$

Table 5
Idealized messages.

Message	Idealized format
$MT\rightarrow gNB$ $LAK_1=\{B_1, B_3, B_4, B_5, C_1, C_2, TS_1\}$	$\langle\gamma, IDV_i, SPG_q, V_L, B_1, TS_1, MT\leftrightarrow^{B_2}gNB\rangle_{A_1}$
$gNB\rightarrow DV_1$ $LAK_2=\{B_3{*}, C_3, B_1, C_4, TS_2\}$	$\langle IDV_i, B_3^*, M_A, B_1, TS_2\rangle_{SK_{g-D_i}}$
$DV_1\rightarrow gNB$ $LAK_3=\{\mathbb{N}_4, D_2, D_3, TS_3\}$	$\langle IDV_i, B_3^*, D_1, D_2, TS_3\rangle_{SK_{g-D_i}}$
$gNB\rightarrow MT$ $LAK_4=\{D_1, D_3, D_4, TS_4\}$	$\langle\gamma, IDV_i, D_1, D_2, TS_4\rangle_{A_1}$

correctness of the proposed scheme. In addition, informal analysis is carried out to show the resilience of the proposed protocol against conventional 5G-IoT attack vectors.

4.1. Formal security analysis

The widely adopted Burrows–Abadi–Needham (BAN) logic is deployed to prove the existence of a session key between the user and DV_i. To achieve this, the notations in Table 2 below are utilized:

Next, the four BAN logic rules in Table 3 are also deployed during this formal proof. In essence, JR implies that if *K* trusts that *S* has control over *G* and *K* trusts that *S* trusts *G*, then *K* also trusts *G*. On the other hand, the MMR implies that if *K* trusts that *R* is shared with *S* and *K* sees *G* combined with *R*, then *K* trusts that *S* said *G*.

In addition, the FPR implies that if *K* trusts that *G* is fresh, then *K* trusts that (G, H) is fresh. Similarly, the implication of NVR is that if *K* trusts that *G* is fresh and *K* believes that *S* said *G*, then *K* trusts that *S* trusts *G*. In addition to these BAN logic rules, the ten Initial Assumptions (IAs) in Table 4 are defined.

For effective proofs, all the exchanged messages are transformed into idealized format as shown in Table 5.

Lastly, four security goals (GLs) are formulated as follows:

G1: $MT\mid\equiv DV_i\mid\equiv(MT\leftrightarrow^{\mathbb{Z}_K}DV_i)$
G2: $MT\mid\equiv(MT\leftrightarrow^{\mathbb{Z}_K}DV_i)$
G3: $DV_i\mid\equiv MT\mid\equiv(MT\leftrightarrow^{\mathbb{Z}_K}DV_i)$
G4: $DV_i\mid\equiv(MT\leftrightarrow^{\mathbb{Z}_K}DV_i)$

Using the idealized messages, initial assumptions and the BAN logic rules, the attainment of the four security goals is proved as follows.

Based on LAK_1, BAN logic proof 1(BLP_1) is obtained:

BLP_1: $gNB\triangleleft\langle\gamma, IDV_i, SPG_q, V_L, B_1, TS_1, MT\leftrightarrow^{B_2}gNB\rangle_{A_1}$

Using IA_6 and MMR on BLP_1, BLP_2 is obtained:

BLP_2: $gNB\mid\equiv DV_i\mid\sim\langle\gamma, IDV_i, SPG_q, V_L, B_1, TS_1, MT\leftrightarrow^{B_2}gNB\rangle$

Based on IA_1 and the FPR.

BLP_3: $gNB\mid\equiv\#\langle\gamma, IDV_i, SPG_q, V_L, B_1, TS_1, MT\leftrightarrow^{B_2}gNB\rangle$

Using NVR on BLP_2 and BLP_3, BLP_4 is yielded:

BLP_4: $gNB\mid\equiv MT\mid\equiv\langle\gamma, IDV_i, SPG_q, V_L, B_1, TS_1, MT\leftrightarrow^{B_2}gNB\rangle$

However, based on LAK_2, BLP_5 is attained:

BLP_5: $DV_i\triangleleft\langle IDV_i, B_3^*, M_A, B_1, TS_2\rangle_{SK_{g-D_i}}$

Applying MMR on IA_7, BLP_6 is obtained as follows.

BLP_6: $DV_i\mid\equiv gNB\mid\sim\langle IDV_i, B_3^*, M_A, B_1, TS_2\rangle$

On the other hand, the application of FPR on IA_2 results in BLP_7:

BLP_7: $DV_i\mid\equiv\#\langle IDV_i, B_3^*, M_A, B_1, TS_2\rangle$

Using NVR on BLP_6 and BLP_7, we obtain.

BLP_8: $DV_i\mid\equiv gNB\mid\equiv\langle IDV_i, B_3^*, M_A, B_1, TS_2\rangle$

Based on LAK_3, BLP_9 is obtained as follows:

BLP_9: $gNB\triangleleft\langle IDV_i, B_3^*, D_1, D_2, TS_3\rangle_{SK_{g-D_i}}$

Using MMR on IA_8, BLP_{10} is yielded:

BLP_{10}: $gNB\mid\equiv\mid\sim\langle IDV_i, B_3^*, D_1, D_2, TS_3\rangle$

According to IA_3 and FPR, the following is attained:

BLP_{11}: $gNB\mid\equiv\#\langle IDV_i, B_3^*, D_1, D_2, TS_3\rangle$

To obtain BLP_{12}, NVR is applied to BLP_{10} and BLP_{11}:

BLP_{12}: $gNB\mid\equiv DV_i\mid\equiv\langle IDV_i, B_3^*, D_1, D_2, TS_3\rangle$

Based on LAK_4, the following is obtained:

BLP_{13}: $MT\triangleleft\langle\gamma, IDV_i, D_1, D_2, TS_4\rangle_{A_1}$

On the other hand, to obtain BLP_{14}, MMR is applied on IA_5:

BLP_{14}: $MT\mid\equiv gNB\mid\sim\langle\gamma, IDV_i, D_1, D_2, TS_4\rangle$

On the other hand, the application of FPR on IA_4 results in BLP_{15}:

BLP_{15}: $MT\mid\equiv\#\langle\gamma, IDV_i, D_1, D_2, TS_4\rangle$

Using NVR on both BLP_{14} and BLP_{15} yields the following.

BLP_{16}: MT $|\equiv$ gNB $|\equiv \langle \gamma, IDV_i, D_1, D_2, TS_4 \rangle$

Since $\mathbb{Z}_K = h(B_3^*||\mathbb{N}_3 D_1)$ and amalgamating BLP_{12} and BLP_{16}, BLP_{17} is obtained:

BLP_{17}: MT $|\equiv DV_i |\equiv (MT \leftrightarrow^{\mathbb{Z}_K} DV_i)$, hence **G1** is attained.

On the other hand, the combination of BLP_4, BLP_8 and the session key results in BLP_{18}:

BLP_{18}: $DV_i |\equiv MT |\equiv (MT \leftrightarrow^{\mathbb{Z}_K} DV_i)$ and as such, **G3** is achieved.

Applying JR on both BLP_{17} and IA_9, BLP_{19} is obtained:

BLP_{19}: MT $|\equiv (MT \leftrightarrow^{\mathbb{Z}_K} DV_i)$, which basically means that **G2** is realized.

Finally, based on JR, IA_{10} and BLP_{18}, the following is obtained:

BLP_{20}: $DV_i |\equiv (MT \leftrightarrow^{\mathbb{Z}_K} DV_i)$, which essentially achieves **G4.**

Since the above BAN logic proofs have successfully attained all the four formulated security goals, it is evident that the proposed protocol realizes mutual authentication as well as session key negotiation between the user's MT and DV_1. The same procedures can be followed to demonstrate the existence of strong mutual authentication and key agreement between the user's MT and any other IoT DV_j.

4.2. Informal security analysis

In this section, various lemmas are formulated and proofed to show that the proposed scheme is robust against many attack vectors under all the assumptions in the Canetti- Krawczyk (CK) threat model. These assumptions are well articulated in Section 1.2 above.

Lemma 1. *Privileged insider and MiTM attacks are prevented in this scheme*

Proof. *During the login, authentication and key negotiation phase, messages LAK_1, LAK_2, LAK_3 and LAK_4 are exchanged. Here, $LAK_1 = \{B_1, B_3, B_4, B_5, C_1, C_2, TS_1\}$, $LAK_2 = \{B_3^*, C_3, B_1, C_4, TS_2\}$, $LAK_3 = \{\mathbb{N}_4, D_2, D_3, TS_3\}$, $LAK_4 = \{D_1, D_3, D_4, TS_4\}$, $B_1 = \mathbb{N}_3 P$, $B_3 = \gamma \oplus h(B_1||B_2)$, $B_4 = IDV_i \oplus h(B_2||TS_1)$, $B_5 = SPG_q \oplus h(A_1||TS_1)$, $C_1 = V_L \oplus h(\gamma||TS_1)$, $C_2 = h(\gamma||IDV_i||SPG_q||A_1||B_1||B_2||TS_1)$, $B_3^* = h(\gamma^*||IDV_i^*||B_2^*)$, $C_3 = M_A \oplus h(IDV_i^*||SK_{g-D_i}||TS_2)$, $C_4 = h(B_3^*||IDV_i^*||M_A||B_1||SK_{g-D_i}||TS_2)$, $D_1 = \mathbb{N}_4 P$, $D_2 = h(ID_g||IDV_i||\mathbb{Z}_K)$, $D_3 = h(B_3^*||IDV_i||D_1||D_2||SK_{g-D_i}||TS_3)$ and $D_4 = h(\gamma^*||IDV_i||A_1^*||D_1||D_2||B_2^*||TS_4)$. Evidently, none of these messages transfers the user password PW_U across the network. Although parameters $\gamma = h(ID_U||\mathbb{N}_2)$ and $\sigma = h(PW_U||\varepsilon_i)$ contain identity ID_U and password PW_U, they are masked in random nonce $\mathbb{N}_2$ and biometric key ε_i. As such, only the user knows ID_U and PW_U and therefore this scheme withstands MitM and privileged insider attacks.*

Lemma 2. *De-synchronization and traceability attacks are thwarted in the proposed scheme*

Proof. *The goal of this attack is to compromise synchronization parameters among the gNB, users and IoT devices such that it becomes difficult for the user to login and authenticate. To curb this attack, the proposed scheme does not require any update of the transient user identity γ, where $\gamma = h(ID_U||\mathbb{N}_2)$. Although this may inadvertently result in traceability attacks against the user, γ is protected by collision-resistant one-way hashing function. In addition, it is masked in high entropy random nonce $\mathbb{N}_2$. Moreover, it is encapsulated in parameter B_3 that is sent as a different value for different messages. For instance, in message LAK_1, it is sent as B_3, where $B_3 = \gamma \oplus h(B_1||B_2)$. However, in message LAK_2, it is sent as B_3^*, where $B_3^* = h(\gamma^*||IDV_i^*||B_2^*)$. Similarly, in message LAK_3, it is transmitted as D_3, where $D_3 = h(B_3^*||IDV_i||D_1||D_2||SK_{g-D_i}||TS_3)$. Further, it is transmitted as D_3 and D_4 in message LAK_4, where $D_4 = h(\gamma^*||IDV_i||A_1^*||D_1||D_2||B_2^*||TS_4)$.*

Lemma 3. *The communicating parties execute strong mutual authentication*

Proof. *In this scheme, the gNB and the user authenticate each other through the validation of membership A_1 as well as parameters C_2 and D_4. Here, it is only the user with valid biometrics data β, password PW_U and gNB issued membership that can derive parameter C_2, where $C_2 = h(\gamma||IDV_i||SPG_q||A_1||B_1||B_2||TS_1)$. To authenticate the user, the gNB checks whether $C_2^* \stackrel{?}{=} C_2$. Upon receiving B_3 in login request LAK_1 from the user's MT, the gNB uses its private key P_{VK} to derive security token $B_2^* = B_1 P_{VK}$. Thereafter, it computes $\gamma^* = B_3 \oplus h(B_1||B_2^*)$ that it utilizes to compute parameter $D_4 = h(\gamma^*||IDV_i||A_1^*||D_1||D_2||B_2^*||TS_4)$. To authenticate the gNB, the user determines whether $D_2^* \stackrel{?}{=} D_2$ and $D_4^* \stackrel{?}{=} D_4$, where $D_2^* = h(ID_g||IDV_i||\mathbb{Z}_K)$ and $D_4^* = h(\gamma||IDV_i||A_1^*||D_1||D_2||B_2||TS_4)$. Similarly, DV_i authenticates the gNB by checking if $C_4^* \stackrel{?}{=} C_4$, while the gNB authenticates DV_i by confirming whether $D_3^* \stackrel{?}{=} D_3$. As such, all the three entities are mutually authenticated. Devoid of SK_{g-D_i} and $\mathbb{Z}_K$, an adversary is unable to derive parameters C_4 and D_2 respectively, and hence its authentication will fail.*

Lemma 4. *Collusion attacks are prevented in this scheme*

Proof. *To grant the user some access rights, the gNB issues the MT with UL_i during the registration phase, where $UL_i = \{(SPG_q, V_L), (SPG_{q+k}, B_{L+2})\}$. During the login, authentication and key negotiation phase, the MT uses message LAK_1 to transmit SPG_q in parameter B_5 which is encapsulated in A_1, where $LAK_1 = \{B_1, B_3, B_4, B_5, C_1, C_2, TS_1\}$, $B_5 = SPG_q \oplus h(A_1||TS_1)$ and $A_1 = h(\gamma||ID_g||M_{VT})$. Similarly, V_L is sent in C_1 which is part of message LAK_1, where $C_1 = V_L \oplus h(\gamma||TS_1)$, $V_L = h(A_1||A_{VT}||\mathbb{N}_1)$ and $\gamma = h(ID_U||\mathbb{N}_2)$. Evidently, V_L is protected through its encapsulation with γ. During the authentication process, the gNB verifies that the user has rights of access SPG_q^* through checking this value in its database as shown in Step 4. Next, it confirm that the user has rights of SPG_q^* by checking if $V_L^{**} \stackrel{?}{=} V_L^*$. Suppose that an adversary has obtained parameters γ, A_1, and SPG_q from some malicious user. The goal here is for the user to assist the adversary escalate his network access rights. However, for this collusion attack to succeed, the adversary needs to derive $V_L^{Adv} = h(A_1||A_{VT}||\mathbb{N}_1)$. Since it is only the gNB that knows nonce $\mathbb{N}_1$ and access rights validation token A_{VT}, the derived V_L^{Adv} will be invalid. In addition, the derivation of parameter A_1 requires knowledge of user temporary identity γ, the gNB unique identity ID_g and membership validation token M_{VT}. Without all these security parameters, the users cannot collude with adversaries to escalate their access rights.*

Lemma 5. *This scheme can withstand ephemeral secret leakages and stolen verifier attacks*

Proof. *The goal of adversaries in this attack is to steal or modify verification tokens such as passwords and biometric data from the gNB's database. To curb this, the user only transmits parameter $\sigma = h(PW_U||\varepsilon_i)$ which is clearly masked in ε_i. In addition, the gNB stores only parameter set $\{\gamma, A_1, SPG_i\}$, where $\gamma = h(ID_U||\mathbb{N}_2)$ and $A_1 = h(\gamma||ID_g||M_{VT})$. Clearly, none of the stored parameters is associated with user password PW_U or biometric data β. Therefore, this scheme is robust against ephemeral secret leakages (ESL) and stolen verifier attacks.*

Lemma 6. *The proposed scheme upholds user and device anonymity*

Proof. *During the registration phase, messages RQ and AP are exchanged, where $RQ = \{\gamma, \sigma\}$ and $AP = \{A_2, h(.), A_3, UL_i, P, P_{UK}\}$. Similarly, messages LAK_1, LAK_2, LAK_3 and LAK_4 are exchanged during the login, authentication and key negotiation phase. Here, $LAK_1 = \{B_1, B_3, B_4, B_5, C_1, C_2, TS_1\}$, $LAK_2 = \{B_3^*, C_3, B_1, C_4, TS_2\}$, $LAK_3 = \{\mathbb{N}_4, D_2, D_3, TS_3\}$ and $LAK_4 = \{D_1, D_3, D_4, TS_4\}$. Clearly, the user's real identity ID_U is never transmitted in these two phases. Consequently, ID_U can never be directly obtained by the eavesdropping of all the exchanged messages. Regarding DV_i's identity IDV_i, its transmission is masked in $C_2 = h(\gamma||IDV_i||SPG_q||A_1||B_1||B_2||TS_1)$, which is part of message LAK_1. Obviously, it is only the gNB that can derive $B_2^* = B_1 P_{VK}$, $\gamma^* = B_3 \oplus h(B_1||B_2^*)$, $A_1^* = h(\gamma^*||ID_g||M_{VT})$ and $IDV_i^* = B_4 \oplus h(B_2^*||$*

TS_1).

Lemma 7. *The proposed scheme upholds user untraceability*

Proof. *In this scheme, the user temporary identity* $\gamma = h\ (ID_U||\mathbb{N}_2)$ *is deployed in all exchanged messages instead of real identity* ID_U. *It is evident that* ID_U *is masked with random nonce* $\mathbb{N}_2$ *and collision-resistant one-way hashing function. In login request* LAK_1, γ *is protected in security parameters* B_3, $C_1 = V_L \oplus h(\gamma||TS_1)$ *and* $C_2 = \mathrm{h}(\gamma||IDV_i||SPG_q||A_1||B_1||B_2||TS_1)$. *To obtain* ID_U *from* γ, *the adversary needs to compute* $\mathbb{N}_2 = \phi \oplus \mathrm{h}(ID_U||\varepsilon_i)$ *and also reverse the one-way hashing function. Here,* $\varepsilon_i = Rep\ (\beta, \nu_i)$ *and hence user biometrics is required to derive* $\mathbb{N}_2$. *Since it is computationally infeasible to reverse h(.) and guess biometrics* β, *an adversary can never recover* $\mathbb{N}_2$. *As such, the attacker can never associate two communication sessions to the same user. Suppose that an adversary has captured the derived session keys* $\mathbb{Z}_K = h\ (B_3{*}||\mathbb{N}_3D_1)$ *and* $\mathbb{Z}_K{*} = h\ (B_3{*}||\mathbb{N}_3D_1)$. *Next, an attempt is made to associate these session keys to some user. Here,* $B_3{*} = h\ (\gamma{*}||IDV_i{*}||B_2{*})$, $\gamma{*} = B_3 \oplus h\ (B_1||B_2{*})$, $B_2{*} = B_1P_{VK}$, $B_1 = \mathbb{N}_3P$, $B_2 = \mathbb{N}_3P_{UK}$, $B_3 = \gamma \oplus \mathrm{h}\ (B_1||B_2)$ *and* $D_1 = \mathbb{N}_4P$. *Clearly, random nonces* $\mathbb{N}_3$ *and* $\mathbb{N}_4$ *are incorporated in these session keys, which are only known to the MT and* DV_i *respectively. Consequently, these session keys are different for each session and can never be associated with a certain user.*

Lemma 8. *The proposed protocol can withstand side-channeling and impersonation attacks*

Proof. *The assumption made in these attacks is that the adversary wants to masquerade as a legitimate user. To accomplish this, side-channeling attack is executed and hence all the security tokens {Gen (.),* A_2, A_3, *Rep (.),* UL_i, *h (.),* ν_i, ϕ, P*} stored in the MT are extracted. In addition, all the exchanged messages {*LAK_1, LAK_2, LAK_3, LAK_4*} in the previous session are captured.*

Case 1: *Suppose that an adversary attempts to derive login request* $LAK_1 = \{B_1, B_3, B_4, B_5, C_1, C_2, TS_1\}$. *Here,* $B_1 = \mathbb{N}_3P$, $B_2 = \mathbb{N}_3P_{UK}$, $B_3 = \gamma \oplus \mathrm{h}\ (B_1||B_2)$, $B_4 = IDV_i \oplus h(B_2||TS_1)$, $B_5 = SPG_q \oplus h(A_1||TS_1)$, $C_1 = V_L \oplus h\ (\gamma||TS_1)$, $\gamma = h\ (ID_U||\mathbb{N}_2)$, $A_1 = \mathrm{A}_2 \oplus h\ (\gamma||\sigma)$, $A_2 = \mathrm{h}\ (\gamma||\sigma) \oplus A_1$, $C_2 = \mathrm{h}\ (\gamma||IDV_i||SPG_q||A_1||B_1||B_2||TS_1)$ *and* $\sigma = h\ (PW_U||\varepsilon_i)$. *Clearly, devoid of valid* ID_U, ID_g, PW_U, P_{UK}, ε_i, A_1, $\mathbb{N}_2$ *and* $\mathbb{N}_3$, *the construction of any valid* LAK_1 *flops. It is evident that the captured messages as well as the MT's memory resident parameters cannot provide the adversary with all the constituents of the login request* LAK_1 *and therefore user impersonation fails.*

Case 2: *Suppose that the adversary has captured previous session messages and wants to impersonate the gNB so as to fool either the user or IoT device* DV_i. *To achieve this, attempts are made to construct messages* LAK_2 *and* LAK_4. *Here,* $LAK_2 = \{B_3{*}, C_3, B_1, C_4, TS_2\}$, $LAK_4 = \{D_1, D_3, D_4, TS_4\}$, $B_1 = \mathbb{N}_3P$, $B_3{*} = h\ (\gamma||IDV_i||B_2)$, $C_3 = \mathrm{M_A} \oplus \mathrm{h}(IDV_i{*}||\ \mathrm{SK_{g-D_i}}||\ TS_2)$, $C_4 = \mathrm{h}(B_3{*}||IDV_i{*}||M_A||B_1||\ \mathrm{SK_{g-D_i}}\ ||TS_2)$, $D_1 = \mathbb{N}_4P$, $D_3 = \mathrm{h}\ (B_3{*}||IDV_i||D_1||D_2||\ \mathrm{SK_{g-D_i}}||\ TS_3)$, $\gamma = h\ (ID_U||\mathbb{N}_2)$, $D_2 = \mathrm{h}\ (ID_g||\ IDV_i||\mathbb{Z}_K)$, $IDV_i = B_4 \oplus h\ (B_2{*}||TS_1)$, $B_2{*} = B_1P_{VK}$, *and* $D_4 = \mathrm{h}(\gamma{*}||\ IDV_i||A_1{*}||D_1||D_2||B_2{*}||TS_4)$. *Evidently, derivation of* D_3 *requires gNB private key* P_{VK} *and shared key between the gNB and* DV_i $\mathrm{SK_{g-D_i}}$, *while the derivation of* $\{C_3\ C_4\}$ *requires shared key between the gNB and* DV_i $\mathrm{SK_{g-D_i}}$. *As such, gNB impersonation flops.*

Case 3: *Suppose that the adversary has captured previous session messages and now wants to impersonate* DV_i. *To accomplish this, message* $LAK_2 = \{B_3{*}, C_3, B_1, C_4, TS_2\}$ *is required. However, devoid of keys* P_{VK} *and* $\mathrm{SK_{g-D_i}}$, *parameters* C_3, $B_3{*}$ *and* C_4 *cannot be computed. In addition, valid timestamps and random nonces are required for these derivations. Devoid of all these security tokens, message* LAK_2 *can never be correctly constructed and hence IoT device impersonation fails.*

Lemma 9. *The communicating entities negotiate session key for traffic protection*

Proof. *During the login, authentication and key negotiation phase,* DV_i *derives session key* $\mathbb{Z}_K = h\ (B_3{*}||\mathbb{N}_3D_1)$. *It then incorporates this session key in security parameters* D_2 *and* D_3, *where* $D_2 = \mathrm{h}\ (ID_g||IDV_i||\mathbb{Z}_K)$ *and* $D_3 = \mathrm{h}\ (B_3{*}||IDV_i||D_1||D_2||\ \mathrm{SK_{g-D_i}}||\ TS_3)$. *Finally, it sends message* $LAK_3 = \{\mathbb{N}_4, D_2, D_3, TS_3\}$ *to the gNB. Here, parameter* $D_3{*}$ *is computed as* $D_3{*} = h\ (B_3{*}||\ IDV_i||D_1||D_2||\ \mathrm{SK_{g-D_i}}||\ TS_3)$. *Thereafter, the derived session key is implicitly validated by checking if* $D_3{*} \stackrel{?}{=} D_3$. *If this verification is successful, it constructs and forwards message* $LAK_4 = \{D_1, D_3, D_4, TS_4\}$ *to the user's MT. Here, session key* $\mathbb{Z}_K{*}$ *and security parameters* $B_3{*}$ *and* $D_2{*}$ *are derived as* $\mathbb{Z}_K{*} = h\ (B_3{*}||\mathbb{N}_3D_1)$, $B_3{*} = h\ (\gamma||IDV_i||B_2)$ *and* $D_2{*} = h\ (ID_g||IDV_i||\mathbb{Z}_K)$. *This is followed by implicit verification of the derived session key* $\mathbb{Z}_K{*}$ *via the checking of whether* $D_2{*} \stackrel{?}{=} D_2$ *as well as* $D_4{*} \stackrel{?}{=} D_4$. *These session keys are then deployed to encipher the exchanged packets to preserve their confidentiality and integrity.*

Lemma 10. *Known secret key attacks are thwarted in the proposed protocol*

Proof. *Suppose that parameters* $B_3{*}$, $\mathbb{N}_3$ *and* D_1 *are captured by an adversary. The aim is to utilize these parameters to compose validation tokens* D_2, D_3, $D_2{*}$ *and* $D_4{*}$. *Here,* $D_1 = \mathbb{N}_4P$, $D_2 = \mathrm{h}\ (ID_g||IDV_i||\mathbb{Z}_K)$, $D_3 = \mathrm{h}\ (B_3{*}||\ IDV_i||D_1||D_2||\ \mathrm{SK_{g-D_i}}||\ TS_3)$, $D_2{*} = h\ (ID_g||IDV_i||\mathbb{Z}_K)$, $D_4{*} = h(\gamma||IDV_i||\ A_1{*}||D_1||D_2||B_2||TS_4)$, $\mathbb{Z}_K = h\ (B_3{*}||\mathbb{N}_3D_1)$ *and* $IDV_i = B_4 \oplus h(B_2{*}||TS_1)$. *It is clear that although* $\mathbb{Z}_K$ *can be computed from the captured parameters, the computation of validation tokens* D_2 *and* $D_2{*}$ *still require the gNB's identity* ID_g *and* DV_i*'s identity* IDV_i. *On its part, token* D_3 *still needs* IDV_i, D_2, $\mathrm{SK_{g-D_i}}$ *and valid timestamp* TS_3. *Similarly, token* $D_4{*}$ *still needs* γ, IDV_i, $A_1{*}$, D_2, B_2 *and valid timestamp* TS_4. *Therefore, the authentication scheme is still secure in the face of active compromise of all the session key components.*

Lemma 11. *This scheme preserves backward and forward key secrecy*

Proof. *Suppose that an attacker has captured the current session keys* $\mathbb{Z}_K$ *and* $\mathbb{Z}_K{*}$. *Thereafter, an attempt is made to derive the session keys for the previous and subsequent communication sessions. Here,* $\mathbb{Z}_K = h\ (B_3{*}||\ \mathbb{N}_3D_1)$, $\mathbb{Z}_K{*} = h\ (B_3{*}||\mathbb{N}_3D_1)$, $B_3{*} = h\ (\gamma||IDV_i||B_2)$, $D_1 = \mathbb{N}_4P$, $IDV_i = B_4 \oplus h\ (B_2{*}||TS_1)$, $\gamma = h\ (ID_U||\mathbb{N}_2)$, $B_2{*} = B_1P_{VK}$, $B_1 = \mathbb{N}_3P$, $B_2 = \mathbb{N}_3P_{UK}$ *and* $B_4 = IDV_i \oplus h\ (B_2||TS_1)$. *Evidently, the computation of any valid session key requires identities* IDV_i *and* ID_U, *random nonces* $\mathbb{N}_2$, $\mathbb{N}_3$ *and* $\mathbb{N}_4$, *timestamp*TS_1, *gNB public key* P_{UK} *and gNB private key* P_{VK}.

Case 1: *Although the current session keys contain parameters* $B_3{*}$, $\mathbb{N}_3$ *and* D_1, *they are protected by collision-resistant one-way hashing function. As such, these parameters cannot be obtained due to the difficulty of reversing h(.). Since parameter* $B_3{*}$ *incorporates* B_2, *it is evident that it is changed after every communication session in accordance with nonce* $\mathbb{N}_3$. *Here, nonce* $\mathbb{N}_2$ *and* $\mathbb{N}_4$ *are only known to the user and* DV_i *respectively. Therefore, the past as well as future session keys are still secure in the face of active capture of the current session keys.*

Case 2: *Suppose that user, gNB and* DV_i *long term secrets are compromised and the attacker obtains* $B_3{*}$, B_1 *and* D_1 *through interception of exchanged message set {*LAK_1, LAK_2, LAK_3, LAK_4*}. However, the adversary needs to solve both ECDL and ECDH problems so as to obtain some of the session key parameters.*

Lemma 12. *Packet replays are prevented in this protocol*

Proof. *During the login, authentication and key negotiation phase, all the exchanged messages* LAK_1, LAK_2, LAK_3 *and* LAK_4 *incorporate timestamps* TS_1, TS_2, TS_3 *and* TS_4. *Here,* $LAK_1 = \{B_1, B_3, B_4, B_5, C_1, C_2, TS_1\}$, $LAK_2 = \{B_3{*}, C_3, B_1, C_4, TS_2\}$, $LAK_3 = \{\mathbb{N}_4, D_2, D_3, TS_3\}$ *and* $LAK_4 = \{D_1, D_3, D_4, TS_4\}$. *Upon receiving message* LAK_1 *from the MT, the gNB validates* TS_1 *against the delay tolerance* ΔTS, *while* DV_i *checks the freshness of timestamp* TS_2 *in message* LAK_2 *using* ΔTS. *Similarly, upon receipt of message* LAK_3 *from the* DV_i, *the gNB validates the freshness of timestamp* TS_3 *using delay tolerance threshold* ΔTS. *On its part, the MT validates timestamp* TS_4 *in message* LAK_4 *such that the session is terminated if this check fails. Consequently, any replayed message will have its timestamp* $TS_i^{Replay} > \Delta TS$. *Therefore, the authentication session using these replayed messages will be*

Table 6
Computation costs.

Scheme	Cost (ms)
[14]	25.78
[22]	48.80
[25]	53.34
[32]	39.09
[41]	27.66
Proposed	6.975

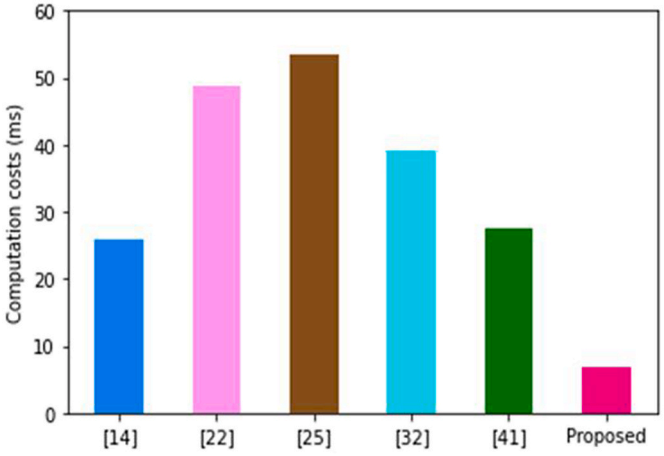

Fig. 5. Computation costs comparisons.

terminated. As such, the proposed scheme can effectively prevent packet replay attacks.

Lemma 13. *This scheme can withstand offline password guessing attacks*

Proof. *Suppose that power analysis is utilized to launch side-channeling attacks upon which all the security tokens {Gen (.), A_2, A_3, Rep (.), UL_i, h (.), ν_i, ϕ, P} in MT's memory are obtained by an adversary. It is further assumed that user identity ID_U has been captured by the attacker.*

Case 1: *The first goal of the adversary is to use the obtained parameters to derive user password PW_U and parameter ε_i needed to launch a user masquerade attack. Among the captured parameters, it is only A_2, A_3 and ϕ that contain PW_U and parameter ε_i. Here, $A_2 = h(\gamma||\sigma) \oplus A_1$, and $A_3 = h(\sigma||A_1)$, $\phi = \mathbb{N}_2 \oplus h(ID_U||\varepsilon_i)$, $\gamma = h(ID_U||\mathbb{N}_2)$, $\sigma = h(PW_U||\varepsilon_i)$ and $A_1 = h(\gamma||ID_g||M_{VT})$. Clearly, to obtaining PW_U and ε_i from parameter σ is cumbersome due to the deployed collision-resistant one-way hashing function. Similarly, deriving ε_i from parameter ϕ involves reversing the one-way hashing function, which is computationally infeasible. In addition, the derivation of ε_i requires not only ν_i but also biometric data β as in Gen $(\beta) = (\varepsilon_i, \nu_i)$.*

Case 2: *Suppose that the attacker has correctly guessed ε_i and is interested in confirming whether the derived password is valid through the $A_3^* \stackrel{?}{=} A_3$ check. Here, $A_3^* = h(A_1 \oplus h(\gamma||PW_U||\varepsilon_i))$ and $A_1 = A_2 \oplus h(\gamma||\sigma)$. As such, this validation still requires parameter $\sigma = h(PW_U||\varepsilon_i)$, which is not stored in the MT's memory. Since the adversary is unable to obtain σ from A_1 owing to the one-way hashing function, offline guessing attack fails.*

5. Performance evaluation

In this section, the proposed scheme is evaluated using computation, communication and the offered security features. These three parameters are chosen since they are the most common metrics for authentication protocol evaluations. Towards the end of this section, experimentations are executed to investigate the variation of end to end delays with the number of authentication requests emanating from IoT devices.

Table 7
Computation costs.

Message	Cost (bits)
MT → gNB	992
$LAK_1 = \{B_1, B_3, B_4, B_5, C_1, C_2, TS_1\}$	
$B_1 = B_3 = B_4 = B_5 = C_1 = C_2 = 160$; $TS_1 = 32$	
gNB → DV_1	672
$LAK_2 = \{B_3^*, C_3, B_1, C_4, TS_2\}$	
$B_1 = B_3^* = C_3 = C_4 = 160$; $TS_2 = 32$	
DV_1 → gNB	480
$LAK_3 = \{\mathbb{N}_4, D_2, D_3, TS_3\}$	
$\mathbb{N}_4 = 128$; $D_2 = D_3 = 160$; $TS_3 = 32$	
gNB → MT	512
$LAK_4 = \{D_1, D_3, D_4, TS_4\}$	
$D_1 = D_3 = D_4 = 160$; $TS_4 = 32$	
Total	2656

Table 8
Communication costs.

Scheme	Cost (bits)
[14]	3104
[22]	3040
[25]	3488
[32]	3088
[41]	4008
Proposed	2656

5.1. Computation costs

In the proposed scheme, the cryptographic operations executed during the login, authentication and key negotiation include fuzzy extraction (T_{FE}), one-way hashing (T_H) and ECC point multiplication (T_{EP}). During the LAK process, $1T_{FE} + 31T_H + 6T_{EP}$ operations are executed. For fair comparisons, the values in Refs. [6,14] are used in which $T_H = 0.055$ ms, $T_{FE} = 1.226$ ms and $T_{EP} = 0.674$ ms. As such, the total computation cost for the proposed scheme is 6.975 ms. Table 6 presents the computation costs of other related schemes.

As shown in Fig. 5, the scheme in Ref. [25] has the highest computation costs of 53.34 ms. This is followed by the protocols in Refs. [14, 22,32,41] in that order. On the other hand, the proposed scheme has the lowest computation overheads of only 6.975 ms. Since most of the IoT devices are limited in terms of processing and battery power, the scheme in Ref. [25] strains these devices and may cause rapid reduction in battery levels.

Consequently, the proposed scheme is the most unsuitable for

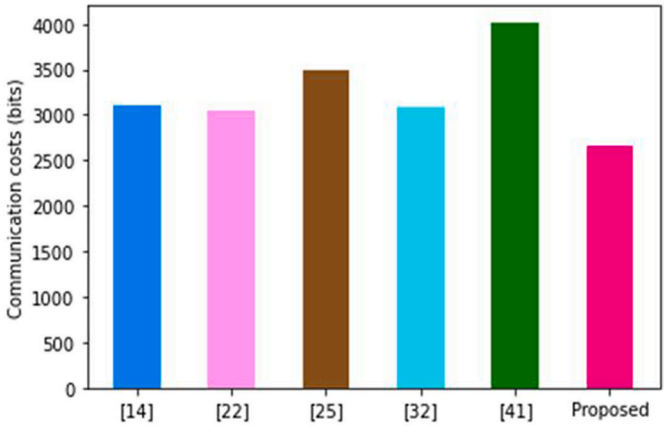

Fig. 6. Communication costs comparisons.

Table 9
Security features comparisons.

	[22]	[32]	[14]	[25]	[41]	Proposed
Security features						
Mutual authentication	×	✓	✓	×	✓	✓
Session key agreement	✓	✓	✓	✓	✓	✓
Anonymity	–	×	✓	–	✓	✓
Forward key secrecy	–	–	×	–	–	✓
Backward key secrecy	–	–	×	–	–	✓
Untraceability	–	–	✓	–	✓	✓
Attacks Resilience						
Side-channeling	×	✓	✓	×	✓	✓
Privileged insider		✓	✓	–	✓	✓
Impersonation	✓	×	✓	✓	✓	✓
De-synchronization	–	–	✓	–	–	✓
Collusion	–	–	–	–	–	✓
Stolen verifier	–	–	✓	–	✓	✓
MitM	✓	×	–	✓	✓	✓
ESL	×	–	✓	×	✓	✓
Replay	✓	✓	–	✓	✓	✓
Known secret key	–	–	✓	–	–	✓
Offline password guessing	–	–	–	–	✓	✓

Key
√ = Supported
× = Not supported
– = Not considered

deployment in an IoT environment. This is justified by its less strain on the processing and battery power levels of the IoT devices.

5.2. Communication costs

During the login, authentication and key negotiation phase, messages LAK_1, LAK_2, LAK_3 and LAK_4 are exchanged. Here, $LAK_1 = \{B_1, B_3, B_4, B_5, C_1, C_2, TS_1\}$, $LAK_2 = \{B_3^*, C_3, B_1, C_4, TS_2\}$, $LAK_3 = \{\mathbb{N}_4, D_2, D_3, TS_3\}$ and $LAK_4 = \{D_1, D_3, D_4, TS_4\}$. Based on the values in Refs. [6,14], the output sizes of the various operations are as follows: elliptic curve point = 320 bits, identity = 32 bits, hash output = 160 bits, random nonce = 128 bits and timestamps = 32 bits. Table 7 gives the derivation of the communication costs of the proposed scheme.

Based on the values in Table 7, the total communication overhead of the proposed scheme is 2656 bits. Table 8 presents the communication costs of other related schemes.

Based on the graphs in Fig. 6, the protocol in Ref. [41] has the highest communication overhead of 4008 bits. This is followed by the schemes in Refs. [14,22,25,32] in that order. On the other hand, the proposed protocol has the least communication overhead of only 2656 bits.

As such, it makes the most efficient use of the network bandwidth. The lower communication costs in the proposed scheme is attributed to the usage of ECC, which provides the same level of security at smaller key sizes compared with techniques such as the Rivest-Shamir-Adleman (RSA). Therefore, it is the most suitable for deployment in 5G-IoT devices since these devices have limited communication capabilities.

5.3. Security features

To show that the proposed scheme offers many salient security and privacy features when compared with other related protocols, the comparisons in Table 9 are presented. It is evident from Table 9 that the protocol in Refs. [22,25] offer only four security features and hence are the most insecure. This is followed by the scheme in Ref. [32] which provides five security features.

On the other hand, the protocols in Refs. [14,41] offer eleven and twelve security features respectively. As such, they are more secure compared with the schemes in Refs. [22,25,32]. On the other hand, the proposed scheme offers seventeen security features and is therefore the most secure among all the other schemes.

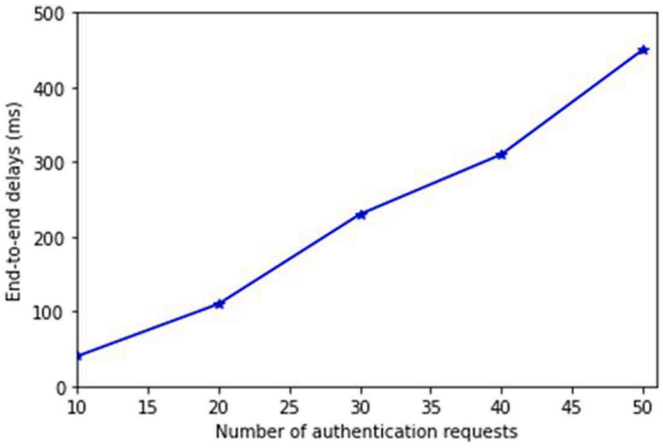

Fig. 7. End-to-end delays.

5.4. Experimentations

In this sub-section, the experimentations that were executed to investigate the variations of End-to-End (E2E) delays with the number of authentication requests are presented. Here, a laptop with a 2.4 GHz Core i5-4210U CPU, 4 GB of RAM and running on Windows 10 Pro-64 bit is utilized. The programming language deployed is Python using PyCrypto library. As shown in Fig. 7, the number of authentication requests is incremented from an initial value of 10 to a maximum of 50 requests.

It can be observed that as the number of authentication requests is increased, there is a corresponding increase in E2E delays. This is attributed to the increased processing that must be executed at the terminals for the surging number of authentication requests. It is clear that the graph of E2E against number of authentication requests is not entirely linear. This is because of other communication impairments such as congestions and packet losses that may necessitate the triggering of error correction techniques that cause further network delays.

6. Conclusion

The 5G-IoT networks convey large amounts of private and sensitive data that can have serious consequences if compromised by attackers. To address this issue, numerous security protocols have been presented over the recent past. However, many security and privacy flaws have been identified in these schemes. In addition, some of the current schemes have poor performance owing to their extremely high communication and computation complexities. As such, the attainment of robust security at low communication and computation overheads is necessary but challenging. In this paper, a lightweight authentication, authorization and session key agreement scheme has been developed to address some of the issues in current security protocols. Its formal and informal security analyzes have shown existence of session key for traffic enciphering, as well as robustness under all the assumptions in the Canetti- Krawczyk (CK) threat model. In terms of performance, it has been shown to have the least communication and computation complexities. As such, the proposed protocol is reliable, efficient and provably secure. Therefore, it is highly applicable in 5G-IoT communication environment where most of the devices are resource constrained. Future work lies in the assessment of the offered security features using other formal as well as informal threat models. There is also need for innovative techniques that can lead to further reduction in the communication overheads of this scheme.

References

[1] Kumari S, Karuppiah M, Das AK, Li X, Wu F, Kumar N. A secure authentication scheme based on elliptic curve cryptography for IoT and cloud servers. J Supercomput 2018;74(12):6428–53.
[2] Cao J, Ma M, Li H, Ma R, Sun Y, Yu P, Xiong L. A survey on security aspects for 3gpp 5g networks. IEEE Commun Surv Tutor 2019;22(1):170–95.
[3] Ayub MF, Mahmood K, Kumari S, Sangaiah AK. Lightweight authentication protocol for e-health clouds in IoT-based applications through 5G technology. Digital Commun Netw 2021;7(2):235–44.
[4] Wu TY, Lee Z, Obaidat MS, Kumari S, Kumar S, Chen CM. An authenticated key exchange protocol for multi-server architecture in 5G networks. IEEE Access 2020; 8:28096–108.
[5] Nyangaresi VO. Lightweight key agreement and authentication protocol for smart homes. In: 2021 IEEE AFRICON, IEEE; 2021. p. 1–6.
[6] Bera B, Saha S, Das AK, Kumar N, Lorenz P, Alazab M. Blockchain-envisioned secure data delivery and collection scheme for 5g-based iot-enabled internet of drones environment. IEEE Trans Veh Technol 2020;69(8):9097–111.
[7] Borgaonkar R, Jaatun MG. 5G as an enabler for secure IoT in the smart grid. In: 2019 first international conference on societal automation (SA). IEEE; 2019. p. 1–7.
[8] Sicari S, Rizzardi A, Coen-Porisini A. 5G in the internet of things era: an overview on security and privacy challenges. Comput Network 2020;179:107345.
[9] Tewari A, Gupta B. Security, privacy and trust of different layers in Internet-of-Things (IoTs) framework. Future Generat Comput Syst 2020;108:909–20.
[10] Vivekanandan M. BIDAPSCA5G: blockchain based Internet of Things (IoT) device to device authentication protocol for smart city applications using 5G technology. Peer-to-Peer Network Appl 2021;14(1):403–19.
[11] Borgaonkar R, Hirschi L, Park S, Shaik A. New privacy threat on 3G, 4G, and upcoming 5G AKA protocols. Proc Priv Enhanc Technol 2019;2019(3):108–27.
[12] Moreira CM, Kaddoum G, Bou-Harb E. Cross-layer authentication protocol design for ultra-dense 5G hetnets. In: IEEE international conference on communications (ICC). IEEE; 2018. p. 1–7.
[13] Das AK, Wazid M, Yannam AR, Rodrigues JJ, Park Y. Provably secure ecc-based device access control and key agreement protocol for iot environment. IEEE Access 2019;7:55382–97.
[14] Wazid M, Das AK, Odelu V, Kumar N, Conti M, Jo M. Design of secure user authenticated key management protocol for generic IoT networks. IEEE Internet Things J 2018;5(1):269–82.
[15] Amin R, Islam SH, Obaidat MS, Biswas G, Hsiao KF. An anonymous and robust multi-server authentication protocol using multiple registration servers. Int J Commun Syst 2017;30(18):e3457.
[16] Zhang Y, Deng R, Bertino E, Zheng D. Robust and universal seamless handover authentication in 5G hetnets. IEEE Trans Dependable Secure Comput 2019;18(2): 858–74.
[17] Wang L, Tian Y, Xiong J. Achieving reliable and anti-collusive outsourcing computation and verification based on blockchain in 5G-enabled IoT. Digital Commun Netw 2022;1–15.
[18] Chen Z, Chen S, Xu H, Hu B. A security authentication scheme of 5G ultra-dense network based on block chain. IEEE Access 2018;6:55372–9.
[19] Fan K, Ren Y, Wang Y, Li H, Yang Y. Blockchain-based efficient privacy preserving and data sharing scheme of content-centric network in 5G. IET Commun 2018;12 (5):527–32.
[20] Messié V, Fromentoux G, Marjou X, Omnes NL. BALAdIN for blockchain-based 5G networks. In: 2019 22nd conference on innovation in clouds, internet and networks and workshops (ICIN). IEEE; 2019. p. 201–5.
[21] Nyangaresi VO, Abduljabbar ZA, Al Sibahee MA, Abduljaleel IQ, Abood EW. Towards security and privacy preservation in 5G networks," in 2021 29th telecommunications forum (TELFOR). IEEE 2021;1–4.
[22] Luo M, Luo Y, Wan Y, Wang Z. Secure and efficient access control scheme for wireless sensor networks in the cross-domain context of the IoT. Secur Commun Network 2018;2018:1–10.
[23] Wu F, Li X, Xu L, Sangaiah AK, Rodrigues JJ. Authentication protocol for distributed cloud computing: an explanation of the security situations for Internet-of-Things-enabled devices. IEEE Consum Electr Mag 2018;7(6):38–44.
[24] Khan A, Abdullah J, Khan N, Julahi A, Tarmizi S. Quantum-elliptic curve cryptography for multihop communication in 5G networks. Int J Comput Sci Network Secur (IJCSNS) 2017;17(5):357–65.
[25] Li F, Han Y, Jin C. Practical access control for sensor networks in the context of the Internet of Things. Comput Commun 2016;89:154–64.
[26] Li N, Liu D, Nepal S. Lightweight mutual authentication for iot and its applications. IEEE Trans Sustain Comput 2017;2(4):359–70.
[27] Nyangaresi VO. Provably secure protocol for 5G HetNets. In: 2021 IEEE international conference on microwaves, antennas, communications and electronic systems (COMCAS). IEEE; 2021. p. 17–22.
[28] Qi M, Chen J, Chen Y. A secure biometrics-based authentication key exchange protocol for multi-server TMIS using ECC. Comput Methods Progr Biomed 2018; 164:101–9.
[29] Cui J, Zhang X, Zhong H, Ying Z, Liu L. RSMA: reputation system-based lightweight message authentication framework and protocol for 5G-enabled vehicular networks. IEEE Internet Things J 2019;6(4):6417–28.
[30] Irshad A, Ahmad HF, Alzahrani BA, Sher M, Chaudhry SA. An efficient and anonymous chaotic map based authenticated key agreement for multi-server architecture. KSII Trans Internet Inf Syst (TIIS) 2016;10(12):5572–95.
[31] Nyangaresi VO. ECC based authentication scheme for smart homes. In: 2021 international symposium ELMAR. IEEE; 2021. p. 5–10.
[32] Singh J, Gimekar A, Venkatesan S. An efficient lightweight authentication scheme for human-centered industrial Internet of Things. Int J Commun Syst 2019;e4189: 1–13.
[33] Jan SU, Qayum F, Khan HU. Design and analysis of lightweight authentication protocol for securing IoD. IEEE Access 2021;9:69287–306.
[34] Tewari A, Gupta B. A lightweight mutual authentication protocol based on elliptic curve cryptography for IoT devices. Int J Adv Intell Paradigms 2017;9(2–3): 111–21.
[35] Adeel A, Ali M, Khan AN, Khalid T, Rehman F, Jararweh Y, Shuja J. A multi-attack resilient lightweight IoT authentication scheme. Trans Emerg Telecommun Technol 2022;33(3):1–15. e3676.
[36] Abd-Elrahman E, Ibn-Khedher H, Afifi H, Toukabri T. Fast group discovery and non-repudiation in D2D communications using IBE. In: 2015 international wireless communications and mobile computing conference (IWCMC). IEEE; 2015. p. 616–21.
[37] Nyangaresi VO. Hardware assisted protocol for attacks prevention in ad hoc networks. In: International conference for emerging technologies in computing. Springer; 2021. p. 3–20.
[38] Malani S, Srinivas J, Das AK, Srinathan K, Jo M. Certificate-based anonymous device access control scheme for IoT environment. IEEE Internet Things J 2019;6 (6):9762–73.
[39] Aman MN, Chua KC, Sikdar B. Mutual authentication in IoT systems using physical unclonable functions. IEEE Internet Things J 2017;4(5):1327–40.
[40] Nyangaresi VO, Petrovic N. Efficient PUF based authentication protocol for internet of drones," in 2021 International Telecommunications Conference (ITC-Egypt). IEEE 2021;1–4.
[41] Wazid M, Das AK, Vasilakos AV. Authenticated key management protocol for cloud-assisted body area sensor networks. J Netw Comput Appl 2018;123:112–26.
[42] Reddy AG, Yoon EJ, Das AK, Odelu V, Yoo KY. Design of mutually authenticated key agreement protocol resistant to impersonation attacks for multi-server environment. IEEE Access 2017;5:3622–39.
[43] Xu D, Chen J, Liu Q. Provably secure anonymous three-factor authentication scheme for multi-server environments. J Ambient Intell Hum Comput 2019;10(2): 611–27.