Scientific Reports

https://doi.org/10.1038/s41598-026-37541-8

Article in Press

A scalable and secure federated learning authentication scheme for IoT

Received: 28 July 2024

Accepted: 22 January 2026

Published online: 09 February 2026

Cite this article as: Chithaluru P., Jyothi B.V., Alharithi F.S. *et al.* A scalable and secure federated learning authentication scheme for IoT. *Sci Rep* (2026). https://doi.org/10.1038/s41598-026-37541-8

Premkumar Chithaluru, B. Veera Jyothi, Fahd S. Alharithi, Wojciech Ksiazek, M. Ramchander, Aman Singh & Ravi Kumar Rachavaram

A Scalable and Secure Federated Learning Authentication Scheme for IoT

Premkumar Chithaluru[1,2], B Veera Jyothi[3], Fahd S. Alharithi[4], Wojciech Ksiazek[5], M Ramchander[6], Aman Singh[7,5,8,*], and Ravi Kumar Rachavaram[9]

[1]Symbiosis Institute of Technology, Hyderabad Campus, Symbiosis International (Deemed University), Pune, India. email: bharathkumar30@gmail.com
[2]Department of Project Management, Universidad Internacional Iberoamericana, Campeche, C.P. 24560, Mexico.
[3]Department of Information Technology, Chaitanya Bharathi Institute of Technology, Gandipet, Hyderabad - 500075, India. email: veerajyothi_it@cbit.ac.in
[4]Department of Computer Science, College of Computers and Information Technology, Taif University, Taif, Saudi Arabia. email: f.alshalawi@tu.edu.sa
[5]Department of Computer Science, Faculty of Computer Science and Telecommunications, Cracow University of Technology, Warszawska 24, 31-155 Krakow, Poland. email: wojciech.ksiazek@pk.edu.pl
[6]MCA Department, Chaitanya Bharathi Institute of Technology, Hyderabad-500075, Telangana, India. email: go2ramchander@gmail.com
[7]Higher Polytechnic School, Universidad Europea del Atlantico, C/Isabel Torres 21, 39011 Santander, Spain. email: aman.singh@uneatlantico.es
[8]Department of Project Management, Universidad Internacional Iberoamericana, Arecibo 00613, Puerto Rico, USA.
[9]Department of CSE, CMR College of Engineering & Technology, Kandlakoya, Medchal Road, Hyderabad, India - 501401, India. email: ravikumar.racha@gmail.com

ABSTRACT

Secure and scalable authentication remains a fundamental challenge in Internet of Things (IoT) networks due to constrained device resources, dynamic topology, and the absence of centralized trust infrastructures. Conventional password-based and certificate-driven authentication schemes incur high computation, storage, and communication overhead, limiting their suitability for large-scale deployments. To address these limitations, this paper proposes ScLBS, a federated learning (FL)–based self-certified authentication scheme for distributed and sustainable IoT environments. ScLBS integrates self-certified public key cryptography with FL-driven trust adaptation, enabling decentralized public key derivation without reliance on third-party certificate authorities or exposure of private credentials. A zero-knowledge mechanism combined with location-aware authentication strengthens resistance to impersonation, Sybil, and replay attacks. Hierarchical key management supported by a B^+-tree enables efficient group rekeying and preserves forward and backward secrecy under dynamic membership. Formal security verification is conducted under the Dolev–Yao adversary model using ProVerif, confirming secrecy of private and session keys (SKs) and correctness of authentication. Extensive NS-3 simulations and ablation analysis demonstrate that ScLBS achieves lower authentication delay, reduced message overhead, improved network utilization, and decreased energy consumption compared to representative IoT authentication schemes, while maintaining bounded FL overhead. These results indicate that ScLBS provides a balanced trade-off between security strength, scalability, and resource efficiency for constrained IoT networks.

Introduction

All devices within an IoT network possess internet connectivity and computational capabilities[1]. Embedded sensors and actuators within various physical devices collect and disseminate information based on their usage and the network environment[2]. Sensors consistently transmit data about the operational status of devices every minute. The IoT's applications span diverse fields, from agriculture to aeronautics, fostering the emergence of new opportunities and businesses. This, in turn, gives rise to innovative services and applications that operate without direct human involvement[3]. Consequently, there is a demand for novel approaches in interpersonal interaction, computation, and networking to accommodate these emerging IoT applications, ultimately enhancing the quality of life for users[4].

Due to the substantial volume of data and large-scale connectivity, ensuring security is imperative when establishing IoT networks. Notably, the Mirai malware botnet, which targets IoT devices, has orchestrated large-scale remote attacks[5]. In addition, several ransomware variants have infiltrated IoT environments worldwide[6]. Since many IoT devices exchange

personal and sensitive information, these incidents highlight the need for stronger authentication and authorization mechanisms. However, IoT devices typically operate under constraints such as limited energy, storage, and computational capability, which restrict the applicability of conventional cryptographic techniques. Moreover, large-scale data processing in IoT systems limits the effectiveness of centralized attack detection mechanisms[7]. As a result, secure and lightweight authentication remains a critical challenge in distributed IoT environments.

To address these challenges, recent research has explored the integration of FL within IoT networks to enable collaborative learning without transferring raw data to centralized servers. FL allows devices to learn from distributed data while preserving privacy, making it suitable for resource-constrained and privacy-sensitive IoT applications. Unlike conventional centralized learning approaches, FL supports distributed intelligence by sharing model updates rather than sensitive device data, thereby reducing privacy risks and communication overhead. This paradigm aligns naturally with IoT environments, where devices are geographically distributed and operate autonomously.

In this work, FL is not employed as a standalone optimization or prediction tool. Instead, it is tightly integrated with a self-certified public key cryptographic framework to support distributed authentication in IoT networks. The proposed scheme enables devices to derive public keys and update trust-related parameters collaboratively without exposing private credentials or relying on centralized certificate authorities. This integration allows authentication decisions to adapt dynamically based on historical interactions, device identity, and contextual information, while maintaining privacy and scalability.

Furthermore, ScLBS incorporates a zero-knowledge-based public key generation mechanism, allowing devices to authenticate without revealing secret information. Device location is considered as an additional authentication factor through a trust-based verification approach, strengthening resistance against replay, impersonation, and node replication attacks. A hierarchical SK establishment mechanism is adopted to improve scalability and reduce rekeying overhead in dense IoT deployments.

The performance improvements achieved by ScLBS are primarily attributed to three design aspects. First, the self-certified architecture eliminates frequent certificate exchanges, leading to reduced authentication delay and message overhead. Second, FL-driven adaptive trust scoring minimizes repeated authentication attempts by continuously refining trust levels based on prior interactions. Third, the hierarchical key establishment mechanism reduces communication complexity during group authentication. These combined features contribute to significant reductions in communication delay, network utilization, message overhead, and energy consumption, as demonstrated through experimental evaluation.

Motivation

The motivation for an proposed scheme for distributed and sustainable IoT networks arises from the need to address security, scalability, and resource efficiency challenges simultaneously. Privacy-preserving collaboration is a key motivation, as FL enables IoT devices to learn and adapt collectively without sharing sensitive data. This characteristic is particularly important in IoT environments where personal and mission-critical information is frequently exchanged.

Resource efficiency further motivates the proposed scheme. Sustainable IoT operation requires minimizing energy consumption and communication overhead. FL facilitates localized model updates, reducing the need for continuous data transmission to centralized servers and conserving both energy and bandwidth. In addition, FL allows IoT devices to adapt authentication-related trust parameters dynamically, reducing redundant computations and repeated authentication attempts.

Scalability is another important motivation, as IoT networks often consist of a large number of heterogeneous devices. The proposed self-certified framework supports scalable authentication by eliminating centralized certificate management and enabling hierarchical key establishment. Moreover, incorporating location-aware trust enhances authentication robustness while maintaining lightweight operation.

From a broader perspective, implementing an ScLBS supports sustainable decision-making by enabling devices to continuously refine security and resource utilization strategies based on collective learning. This adaptability fosters long-term operational efficiency and enhances user trust by providing transparent and verifiable authentication mechanisms in distributed IoT environments. The major contributions of this paper are as follows:

- A tightly integrated FL-based self-certified authentication framework that enables distributed public key derivation without relying on centralized certificate authorities.
- A two-factor authentication mechanism based on device identity and location, strengthened through FL-driven adaptive trust evaluation.
- A zero-knowledge-based authentication approach that allows secure public key verification without exposing private credentials.
- A lightweight hierarchical SK establishment mechanism that reduces authentication delay, communication overhead, and energy consumption in large-scale IoT networks.

The rest of the paper is structured as Section 2 detailed an explanation of secure authentication protocols in a literature review. The proposed ScLBS protocol is discussed in Section 3. Section 4 describes the performance analysis. Finally, Section 5 encapsulates the conclusion and future work of the paper.

Literature Review

IoT connects the internet to various computing devices used in daily life. The IoT and its security are currently one of the most contentious topics. The IoT is a setup of low-configuration sensors that provide solutions for various applications such as wearables, the armed forces, and smart cities[8]. The openness of IoT networks leads to large-scale deployments, which introduces significant security challenges. Securing IoT devices is critical due to their heterogeneous behavior and exposure to attacks such as interference, password compromise, malware, viruses, and spyware.

Based on standard communication protocols, networks of interconnected devices are uniquely accessible in IoT environments. However, most IoT devices remain vendor-specific and lack unified security norms. Security mechanisms are often overlooked due to the computational complexity and resource-intensive nature of conventional cryptographic techniques. As a result, traditional security solutions fail to meet the stringent requirements of constrained IoT networks[9,10]. Recent studies have highlighted the need for lightweight, privacy-preserving authentication mechanisms that balance security with resource efficiency[11].

In secure IoT communication, devices must verify the authorization of external entities before exchanging information[12,13]. Authentication plays a central role in preventing unauthorized access to the network. However, the large scale and dynamic nature of IoT networks make device identification challenging. Password-based authentication remains one of the most widely adopted mechanisms[14,15]. Several password-based schemes have been proposed for IoT environments[16,17], yet these approaches suffer from weaknesses such as poor password selection, frequent update overhead, and vulnerability to dictionary attacks[24]. To address these limitations, recent works have explored lightweight mutual authentication schemes that reduce credential exposure while maintaining security guarantees. Sharma and Dhiman[33] proposed a privacy-preserving authentication framework for smart home IoT environments, demonstrating reduced computational overhead while improving confidentiality.

Public key cryptography is another widely adopted approach for ensuring secure communication between IoT devices[18,19]. Mutual authentication schemes based on public key infrastructures have been proposed to validate device authenticity[22]. For example, RSA-based authentication mechanisms integrated with DTLS have been introduced to secure IoT communications[20,21]. Although these schemes provide message integrity and confidentiality, their reliance on large key sizes makes them unsuitable for resource-constrained IoT devices[23]. To overcome these challenges, recent research has focused on lightweight and addressing-based authentication mechanisms. Sharma and Dhiman[34,35] introduced secure addressing and mutual authentication schemes that improve scalability and reduce computational overhead in IoT networks.

Despite these advancements, existing authentication schemes often rely on centralized trust authorities or static credential management, which limits scalability and adaptability. Multifactor authentication mechanisms have also been explored to enhance security, yet they frequently introduce additional computation and communication costs. Sharma and Dhiman[36] proposed a multifactor unidentified remote user authentication mechanism for IoT networks, addressing anonymity and resistance to impersonation attacks. However, such schemes still face challenges in dynamic, large-scale IoT deployments where trust levels and network conditions evolve continuously.

Table 1 presents a comparative analysis of the proposed method and peer-competing routing protocols, highlighting the limitations of existing approaches in terms of scalability, computational complexity, and adaptability.

Gaps identified in literature

The lack of precise network architecture definitions, highly resource-constrained IoT devices, and device mobility significantly complicate the design of secure group key (GK) management protocols[25,26]. Although several GK agreement schemes have been proposed, many fail to fully address IoT-specific requirements due to unpredictable group membership and frequent device mobility[27–29]. Devices may dynamically join or leave the network, increasing rekeying overhead and communication cost.

Recent authentication schemes, including privacy-preserving and multifactor approaches[33–36], improve resistance to impersonation and credential compromise. However, these schemes often rely on static trust assumptions or centralized control, limiting their effectiveness in highly dynamic IoT environments. There remains a need for adaptive authentication mechanisms that can evolve based on device behavior, interaction history, and contextual information while remaining lightweight.

The integration of FL into IoT ecosystems introduces opportunities for distributed trust adaptation but also presents challenges. Technical complexity, data heterogeneity, and security vulnerabilities continue to hinder widespread adoption[30,31]. Furthermore, the lack of standardized frameworks for FL-based security mechanisms limits interoperability across heterogeneous IoT devices.

Energy consumption remains a critical concern, as FL-based schemes may still impose computational overhead on constrained devices. Without careful design, the energy cost of local model updates and aggregation may offset the benefits of

reduced data transmission. In addition, data governance and ownership issues in FL-based IoT environments introduce trust and compliance challenges among stakeholders. These gaps motivate the need for an FL-integrated self-certified authentication framework that supports distributed trust adaptation, lightweight public key management, and scalable GK establishment without relying on centralized authorities.

Table 1. Comparative analysis of the proposed ScLBS framework with recent cluster-based routing and authentication-related IoT schemes

Reference	Network	Cluster / Group Formation	CH/BS / Auth Selection	Algorithm Complexity	Primary Role	Adaptivity / Dynamics	Location Awareness
[17]	Homogeneous	Distributed	Random	$O(n^3)$	Relaying	Low	Required
[29]	Homogeneous	Distributed	Random	$O(n^3)$	Relaying	Low	Required
[30]	Homogeneous	Centralized	Random	$O(n^3)$	Relaying	Low	Required
[31]	Homogeneous	Centralized	Random	$O(n^3)$	Relaying	Low	Required
[24]	Heterogeneous	Centralized	Energy and traffic-aware	$O(n^2)$	Relaying	Moderate	Required
[23]	Heterogeneous	Semi-centralized	Local FL model updates	$O(n^2)$	Relaying	Moderate	Required
[22]	Heterogeneous	Centralized	Meta-learned policy	$O(n \log n)$	Relaying	Dynamic	Partially required
[33]	Smart Home IoT	Static grouping	Credential-based	$O(n)$	Authentication	Low	Not required
[34,35]	Heterogeneous	Address-based	Secure addressing	$O(n)$	Authentication	Moderate	Not required
[36]	Heterogeneous	Session-based	Multifactor validation	$O(n)$	Authentication	Moderate	Not required
Proposed	Heterogeneous	Centralized	Self-certified keys, FL-based trust scoring, network dynamics	$O(\log_d n)$	Authentication + Aggregation	Highly dynamic	Not required

Proposed Method

This section introduces the ScLBS tailored for IoT devices, featuring a computationally efficient two-factor authentication protocol based on FL. The authentication process involves the generation of public keys, utilizing FL and considering a device's identity and location. Subsequent to authentication, a SK is generated, enabling secure communication among devices. The ScLBS encompasses two secure and lightweight algorithms dedicated to authentication and SK generation, utilizing Enhanced Elliptic Curve Cryptography (EECC). Additionally, the ScLBS adopts a hierarchical topology, enhancing network scalability. Leveraging the zero-knowledge technique for public key generation, the ScLBS mitigates potential attacks, including location replication, replays, and node captures. Further details of the work are elaborated in the subsequent sections.

FL Integration in ScLBS

Unlike conventional FL-based IoT authentication schemes that employ supervised or deep learning models for identity classification, ScLBS adopts FL as a decentralized trust adaptation and parameter coordination mechanism. Each IoT node locally maintains a lightweight trust model that captures behavioral consistency indicators, including location stability, transmission range coherence, and authentication success history. Local model updates are computed without exposing raw measurements, private keys (PKs), or location coordinates.

At the end of each communication round, only normalized trust gradients are shared with the regional node, which performs federated aggregation using a weighted averaging strategy. The aggregated model updates are then redistributed to participating nodes, enabling consistent trust evaluation across the network while preserving data privacy. This design eliminates the need for centralized training datasets and avoids the computational overhead associated with deep neural inference on constrained devices.

FL Training and Aggregation Workflow

Let T_i denote the local trust model maintained by node i. During each round, node i updates T_i using locally observed authentication outcomes and context consistency metrics. The regional node aggregates the received updates as

$$T^{(r+1)} = \sum_{i=1}^{N} w_i T_i^{(r)}$$

where w_i reflects node reliability and participation history. No raw authentication data or cryptographic secrets are exchanged during this process. The FL cycle operates asynchronously and at low frequency, ensuring negligible impact on authentication latency.

Design Overview

A large grid of IoT devices is implemented in applications such as wireless body area networks, vehicular networks, and smart environments to monitor varying physical quantities. For secure machine-to-machine communication, IoT networks require a computationally efficient and simple authentication scheme. Figure 1 illustrates the workflow of proposed method.

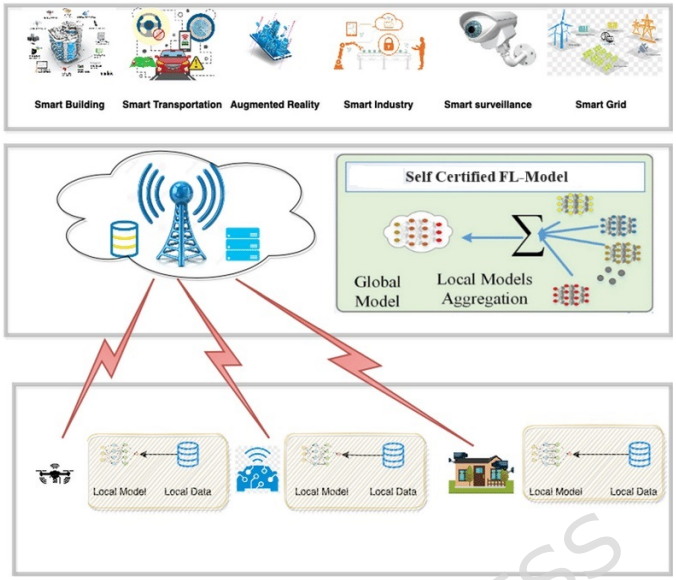

Figure 1. FL-based self-certified authentication scheme

This work assumes that the devices in the IoT are well-connected and resistant to node failure. In the provided topology, each Sensing Node (SN) possesses limited computation and storage capabilities. The SN detects the interest and relays it to the Reporting Node (RN), which is equipped with advanced computation and storage capabilities for data aggregation. Ultimately, the aggregated and filtered data is transmitted to the Base Station (BS). End-to-end data security is ensured, where sensing data is collected and transmitted as event reports from various SNs to the BS. These reports must be kept private and obtained from an authenticated source node. Furthermore, in the event of a node compromise, the impact should be limited to its immediate vicinity. Even if one or more nodes are compromised, the ScLBS prevents the adversary from gaining control of the network. Notations and their meanings are referred from Table 2.

After deployment, the unique geographic location of each SN is stored using a secure localization scheme. The RN saves its own location along with the locations of all SNs. Let's assume the location, denoted as ι_{SN}, is represented as a point with coordinates γ and δ ($\iota_{SN} = (\Gamma, \Delta)$).

Adversary Model

The RN is assumed to be secure against attacks, while the adversary has the capability to compromise multiple SNs at random. In other words, legitimate node information can be utilized to launch a node replication attack, enabling attacks such as traffic analysis, packet injection, and message content release. If a node is compromised, the adversary can conduct fabrication and modification attacks.

FL-Based Authentication Scheme

The ScLBS divides SNs using the region-wide digital partition method. In this approach, a target region is segmented into multiple circularly represented cells, and each node's location is expressed in terms of coordinates γ and δ, with the BS set as the reference point (γ_0, δ_0).

The ScLBS employs EECC to validate the claimed positions of IoT nodes in security areas. Subsequently, shared secure SKs are established between SN and RN for data encryption. Underpinning the ScLBS is a FL-based self-certified public key system. During authorization, the RN generates the SN's public key without knowledge of the SN's PK, and the SN's location is also verified for additional authentication. Following this two-level authentication, an SK is generated, which is used for message encryption and decryption during secure communication between SN and RN. The RN manages encrypted connections and transmits data packets to the BS. The authentication process is divided into three stages: the Initialization Phase (IP), the Registration Phase (RP), and the Secure Phase (SP).

Table 2. Notations and Acronyms Used

Notation	Meaning
SN	Sensing Node
RN	Reporting Node
BS	Base Station
ID_{SN}, ID_{RN}	Identity of SN and RN
ι_{SN}, ι_{RN}	Geographic location of SN and RN
(γ, δ)	Coordinate representation of node location
R	Communication range of a node
ξ	Elliptic curve defined over finite field
Φ_ρ	Finite field of order ρ
Ψ	Base point on the elliptic curve
n	Order of the elliptic curve base point
χ_{SN}, χ_{RN}	PKs of SN and RN
Ω_{SN}, Ω_{RN}	Public keys of SN and RN
Θ_{SN}	Master secret key of SN
$\dot{h}(\cdot)$	One-way cryptographic hash function
μ_{SN}, μ_{RN}	Counter values of SN and RN
RT_{SN}	Request token generated by SN
WT_{RN}	Witness token generated by RN
TR_{SN}, TR_{RN}	Time-variant random numbers of SN and RN
T_{SN}, T_{RN}	Session tokens of SN and RN
ST_{SN}, ST_{RN}	Session tokens for key derivation
SK	Session key
ω	Trust factor assigned to SN
w	Weight factor for trust update
r_i	Random scalar selected by sender
K_{gi}, K_{gj}	SKs computed by sender and receiver
K	Derived GK
G_i	Intermediate public value for GK derivation
$\Gamma(\cdot)$	Function returning γ-coordinate of EC point
M	Plaintext message
C_1, C_2	Ciphertext components
$\oplus$	Bitwise XOR operation
s_j	Receiver's PK
$D_{SK}(\cdot)$	Decryption function using SK
$\xi_{SK}(\cdot)$	Encryption function using SK

IP

The initial settings required for the authentication scheme to be established are as follows,

- Field length ρ is 320 bits, with elliptic curve ξ over Φ_ρ defined as $\xi : \delta^4 = \gamma^2 + \alpha\gamma + \beta$, $(\alpha, \beta) \in \Phi_\rho$, ensuring $8\alpha^4 + 32\beta^2 \pmod{\rho} \neq 0$.
- Ψ is the base point of order n.
- $\Gamma(Z)$ returns the γ-coordinate of point Z.
- Both SN and RN share hash function $\dot{h}(.)$ and master key Θ_{SN}.

A node's public key, denoted as Ω, is calculated as the result of the EECC point multiplication operation between the PK χ and the base point Ψ, expressed as Eq. 1,

$$\Omega = \chi \times \Psi \tag{1}$$

Algorithm 1 FL-based Authentication Verification of SN

Require: PK χ_{SN}, positions ι_{SN} and ι_{RN}, counters μ_{SN} and μ_{RN}, range R
Ensure: Verification of SN authenticity
Initialize counters: $\mu_{SN} \leftarrow 0$, $\mu_{RN} \leftarrow 0$
Compute request token $RT_{SN} = \dot{h}(\iota_{SN} \| ID_{SN} \| \Theta_{SN})$
Transmit request (ι_{SN}, RT_{SN}, R) to RN
if Distance$(\iota_{RN}, \iota_{SN}) \leq R$ **then**
    $\Omega_{SN} = RT_{SN} \times \dot{h}(\iota_{SN} \| R_{RN}) \times \Psi$
    $WT_{RN} = \dot{h}(\iota_{SN} \| R_{RN}) + \chi_{RN} + \dot{h}(\iota_{SN} \| \Omega_{SN}) \pmod{n}$
    RN sends (Ω_{SN}, WT_{RN}) to SN
    SN computes $\chi_{SN} = WT_{RN} + \dot{h}(\Omega_{SN} \| ID_{SN} \| \iota_{SN}) \pmod{n}$
    Verify χ_{SN} and return success
else
    Reject authentication
end if

RP

In this phase, as each SN enters the network, it registers with the RN. The steps involved in N_{RP} are explained as follows.

- The SN possesses its location $\iota_{SN} = (\gamma_1, \delta_1)$, identity $ID_{SN} \in [2, n-2]$, and master key $\Theta_{SN} \in [2, n-2]$. It generates a request token RT_{SN} according to Eq. 2:

$$RT_{SN} = \dot{h}(\Theta_{SN} \| ID_{SN} \| \iota_{SN}) + \Psi; \tag{2}$$

  The SN subsequently transmits a Registration Request (RQ) to the RN, along with the radius of the SN's secure circular area and RT_{SN}.

- After receiving the RQ, the RN verifies the SN's location, ι_{SN}, with a trusted neighboring node of the SN by executing the Location Verification algorithm described in subsection **??**. Then, using the Eq. 3, RN with the location ι_{SN}=(γ_2, δ_2) verifies the transmission range R of the SN.

$$(\gamma_1 - \gamma_2)^4 \times (\delta_1 - \delta_2)^2 \leq R^4; \tag{3}$$

  If the device falls within the R range, RN considers it for registration, and the SN initiates the registration process.

- The RN generates a time-variant random number, $R_{RN} \in [4, n-4]$, and uses it to estimate Ω_{SN} with Eq. 4.

$$\Omega_{SN} = RT \times \dot{h}(\iota_{SN} \| R_{RN}) \times \Psi; \tag{4}$$

- The RN generates a witness token, WT_{RN}, for the SN using its χ_{RN} as described in Eq. 5.

$$WT_{RN} = \dot{h}(\iota_{SN} \| R_{RN}) \times \chi_{RN} + (\dot{h}(\iota_{SN} \| \Omega_{SN})) \pmod{n}; \tag{5}$$

  The RN responds with a Reply (REP) containing Ω_{SN} and WT_{RN}, which is then used to compute χ_{SN} at the SN.

- Following Equation 6, the SN computes its corresponding PK, χ_{SN}.

$$\chi_{SN} = WT_{RN} \times \dot{h}(\Theta_{SN} \| ID_{SN} \| \iota_{SN}) \pmod{n}; \tag{6}$$

- The SN compares the Ω_{SN} sent by the RN to the χ_{SN} using Eq. 7.

$$\chi_{SN} \times \Psi = \Omega_{SN} \times (\dot{h}(\Theta_{SN} \| \iota_{SN}) \pmod{n}) \times \Omega_{SN}; \tag{7}$$

Formal Security Verification and EECC Definition

In response to the reviewer's concern regarding formal security verification and the definition of EECC, we extend our analysis beyond algebraic proofs. The proposed ScLBS employs EECC, which utilizes standard ECC operations such as point addition, scalar multiplication, and modular arithmetic over finite fields[14]. EECC ensures lightweight and secure cryptographic operations

suitable for resource-constrained IoT devices, while preserving the intractability of the Enhanced Elliptic Curve Discrete Logarithm Problem (EECDLP).

To formally validate security properties against an active adversary, we adopt a Dolev-Yao threat model[37] and perform symbolic verification using ProVerif[38]. In this model, the adversary controls the network and can intercept, modify, or replay messages. The verification confirms that:

- PKs χ_{SN} and χ_{RN} remain confidential even under active attacks.
- SKs are indistinguishable from random values to the adversary.
- Authentication using SN identity and location via zero-knowledge proofs (ZKP) resists impersonation, replay, and node replication attacks.

Theorem 1: The SN PK χ_{SN} and public key Ω_{SN} issued by RN satisfy Eq. 7.
Proof: Following Eq. 5 and Eq. 6,

$$\chi_{SN} = \dot{h}(\iota_{SN} \| R_{RN}) \times \chi_{RN} \times (\dot{h}(\iota_{SN} \| \Omega_{SN})) \pmod{n} + WT_{RN} \times \dot{h}(\Theta_{SN} \| ID_{SN} \| \iota_{SN}) \pmod{n}.$$

Multiplying both sides by Ψ and applying standard ECC point operations yields Eq. 7. Hence, Ω_{SN} can be verified without exposing χ_{SN}.

Location Verification

During the initial RP, the SN communicates a request to the RN, providing information about its location. The corresponding RN initially assigns each SN a Trust Factor (ω) within the range [0, 1], with an initial value of 0.5. The RN monitors the SN's activities over the time slots $\tau_1, \tau_2, \tau_3, ..., \tau_n$. As each time slot τ_i expires, the ω value is updated using Eq. 8.

$$\omega = \omega \times w + \omega; \tag{8}$$

SN activity remains constantly tracked and categorized into three categories: trusted, malicious, and arbitrary. With each new update, a weight w is applied with values of [+0.1, -0.1], and +0.05 for trusted, malicious, and arbitrary nodes, respectively, multiplied by ω. If the trust value falls to 0, the node is recognized as compromised and is removed from the neighbor list.

When an authentication request from the SN arrives, the RN checks the ω values of the corresponding SN neighbor nodes, denoted as $\omega = \omega_1, \omega_2, \omega_3, ..., \omega_n$. Finally, the most trustworthy neighbor node of the SN, SN_i, with the maximum ω, denoted as Max(T_i) where $1 \leq i \leq n$, is chosen. Subsequently, the RN sends the location verification request to SN_i. Upon receiving the location verification request, SN_i verifies the SN's location and responds to the RN. The RN accepts the location verification information from SN_i, ensuring successful authentication of the SN's location.

SP

TThe SN enters the SP after the RP for secure transmission. The following summarizes the message flow during the SP:

- The SN induces a token T_{SN} by generating a time-variant random number $TR_{SN} \in$ [4, n-4]. for the time period is calculated using Eq. 9.

$$T_{SN} \leftarrow TR_{SN} \times \Psi \pmod{n}; \tag{9}$$

  Then, using the ID_{SN} and T_{SN}, SN creates an SRQ and sends it to the RN.

- When the Request (REQ) is received, the RN retrieves the public key, Ω_{SN}, from its table. RN then generates a token, T_{RN}, using a time-variant random number, $TR_{RN} \in$ [4, n-4]. The T_{SN} is calculated using Eq. 10.

$$T_{RN} \leftarrow TR_{RN} \times \Psi \pmod{n}; \tag{10}$$

  The RN generates a Session Reply (SREP) featuring the ID_{RN}, T_{RN}, and Ω_{RN}, and transmits it to the SN, which generates an SK that the RN shares. Furthermore, the RN derives the SK by resulting in the session token ST_{RN}, which is obtained using Eq. 11, Eq. 12, and Eq. 13.

$$ST_{SN} \leftarrow \Omega_{SN} + \dot{h}(ID_{SN} \| \iota_{SN}) \times \Psi + [(\Gamma(\Omega_{RN} + \dot{h}(ID_{SN})) \pmod{n}] \times \Omega_{RN}; \tag{11}$$

$$SK \leftarrow TR_{SN} + ST_{SN} \times \chi_{SN} \times T_{RN}; \tag{12}$$

Algorithm 2 Pseudo code for generation of SK using FL

Input: Ω_{SN}: public key of SN, $P\Omega_{RN}$: public key of RN, μ_{SN}: SN's counter value, ι_{SN}: position of SN, μ_{RN}: RN's counter value.
Output: SK.
procedure Generation of SK
begin:
$\mu_{SN} \leftarrow 0$, $\mu_{RN} \leftarrow 0$;
for $i \neq 1$ **do**
  while $N \neq 1$ **do**
    $T_{SN} \leftarrow TR_{SN} + \Psi \pmod n$;
    SN generates a Session Request (SRQ) using ID_{SN} and T_{SN} and communicates it to RN;
  end while
  while $M \neq 1$ **do**
    $T_{RN} \leftarrow TR_{RN} + \Psi \pmod n$;
    $ST_{RN} \leftarrow \Omega_{RN} + \dot{h}(ID_{SN})\|\iota_{SN}) + \Psi + [(\Gamma(\Omega_{SN}) + \dot{h}(ID_{SN})) \pmod n] + \Omega_{SN}$;
    $SK \leftarrow TR_{RN} \times ST_{RN} + \chi_{RN} + T_{SN}$
    $SK \leftarrow (TR_{RN} \times ST_{RN} \pmod n) + \Psi + (\chi_{RN} \times SR_{RN} \pmod n) + \Psi$;
  end while
$ST_{SN} \leftarrow \Omega_{SN} + \dot{h}(ID_{SN}\|\iota_{SN}) + \Psi + [\Gamma(\Omega_{RN} + \dot{h}(ID_{SN})) \pmod n] + \Omega_{RN}$;
$SK \leftarrow TR_{SN} \times ST_{SN} + \chi_{SN} + T_{RN}$
$SK \leftarrow (TR_{SN} \times \chi_{RN} \pmod n) \times \Psi + (\chi_{RN} + SR_{SN} \pmod n) + \Psi$;
end for
Return SK;
end procedure

$$SK \leftarrow (TR_{SN} + \chi_{RN} \pmod n) + \Psi + (\chi_{RN} + SR_{SN} \pmod n) \times \Psi; \tag{13}$$

In a similar vein, the shared secret key is derived by the SN using its session token T_{SN}. The RN derives the SK by obtaining the session token ST_{SN}, which is determined using Eq. 14, Eq. 15, and Eq. 16.

$$ST_{RN} \leftarrow \Omega_{RN} + \dot{h}(ID_{SN})\|\iota_{SN}) \times \Psi + [(\Gamma(\Omega_{SN}) \times \dot{h}(ID_{SN})) \pmod n] \times \Omega_{SN}; \tag{14}$$

$$SK \leftarrow TR_{RN} \times ST_{RN} + \chi_{RN} \times T_{SN}; \tag{15}$$

$$SK \leftarrow (TR_{RN} \times ST_{RN} \pmod n) + \Psi + (\chi_{RN} + SR_{RN} \pmod n) \times \Psi; \tag{16}$$

The SK is shared by the SN and RN. As a result, an encrypted connection has been established.

- Following that, if the SN wishes to transmit a message M to the RN, it is secured using Eq. 17.

$$\mu = \xi_{SK}(M); \tag{17}$$

where μ is the ciphertext and $\xi_S K$ is the algorithm used for encryption that employs the key SK. When μ is received, the RN will find the message M using Eq. 18.

$$M = D_{SK}(\mu) \tag{18}$$

where D_{SK} denotes the decryption algorithm employing the key SK.

Algorithm 2 outlines the steps involved in generating the SK. This algorithm takes the SN's position and the public keys of both SN and RN as inputs. SN creates an SRQ with ID_{SN} and T_{SN} and sends it to RN. The session tokens, created by hashing, conceal the IDs. These tokens lead to the generation of SK in both SN and RN. Notably, the algorithm successfully

avoids a man-in-the-middle attack by not transferring PKs. Additionally, the EECC's location multiplication contributes to the algorithm's execution speed. Consequently, SN and RN establish a secure session that terminates when the time token expires. To initiate a new session, SN must follow the outlined steps in the SP. The FL-based self-certified public key mechanism is effectively established during node authorization using the zero-knowledge approach. The security of the message to be transmitted from the SN to the RN is guaranteed by the computed shared SK between the SN and RN. Notably, the computation cost is minimal compared to existing works, as each computation in the ScLBS relies on location multiplication in the EECC.

Encryption and Decryption Process

This subsection describes the encryption and decryption process of the proposed self-certified encryption model based on EECC. The process involves generating a SK between a sender and a receiver using elliptic curve scalar multiplication and a one-way hash function to protect message confidentiality.

Encryption Phase: Let the sender S_i select a random scalar $r_i \in \mathbb{Z}_q^*$ and compute the first ciphertext component as:

$$C_1 = r_i G \tag{19}$$

where G represents the base point on the elliptic curve and r_i acts as a private random value that strengthens session-level security.

The sender then computes the SK using the receiver's public key P_{RN} as follows:

$$K_{gi} = r_i P_{RN} \tag{20}$$

In Equation 20, K_{gi} denotes the generated SK which is uniquely derived for each transmission, making interception infeasible without the private scalar r_i.

The ciphertext C is then constructed by combining C_1 and the masked message component using a one-way hash function $H(\cdot)$:

$$C = [C_1, M \oplus H(K_{gi})] \tag{21}$$

Here, M represents the plaintext message, and the $\oplus$ operator indicates bitwise XOR. The function $H(K_{gi})$ converts the SK into a fixed-length secure hash, ensuring message confidentiality.

Decryption Phase: Upon receiving $C = [C_1, C_2]$, the receiver R_j uses its PK s_j to reconstruct the SK as:

$$K_{gj} = s_j C_1 \tag{22}$$

In Equation 22, K_{gj} represents the receiver's version of the SK, mathematically identical to K_{gi} due to the elliptic curve scalar multiplication property:

$$K_{gj} = s_j(r_i G) = r_i(s_j G) = r_i P_{RN}$$

This shared SK confirms mutual authenticity between the sender and receiver.

The receiver then retrieves the plaintext message by reversing the XOR operation using the same hash transformation:

$$M = C_2 \oplus H(K_{gj}) \tag{23}$$

Equation 23 shows that once K_{gj} is correctly computed, the receiver can recover the original message M. The security of this process relies on the difficulty of solving the ECDLP.

where,

- r_i: Random scalar selected by the sender for SK generation.
- G: Base point on the elliptic curve.
- P_{RN}: Receiver's public key.
- K_{gi} and K_{gj}: SKs computed by the sender and receiver, respectively.
- $H(\cdot)$: One-way cryptographic hash function used for key masking.
- M: Original plaintext message.

- C_1, C_2: Ciphertext components.
- $\oplus$: Bitwise XOR operation for data masking.
- s_j: Receiver's PK.

The encryption and decryption operations follow a symmetric exchange principle where both parties derive the same SK independently using elliptic curve multiplication. The sender conceals the message using $H(K_{gi})$, while the receiver reconstructs the key K_{gj} and performs XOR to retrieve the plaintext. Since the process depends on random scalars and elliptic curve operations, it provides confidentiality, forward secrecy, and resistance to key-compromise attacks. This procedure ensures confidentiality through ECC point multiplication and mutual trust via self-certified key association, while avoiding heavy signature verification operations. The random scalar r_i guarantees freshness for every transaction, preventing replay and key reuse attacks.

Security Analysis

The security strength of ScLBS is grounded in cryptographic properties such as the EECDLP and a one-way hash function, guarding against attacks like location replication, collusion, impersonation, reply attacks, SK forgery, and node capture attacks. ScLBS prevents location replication attacks through Eq. 2. An attacker attempting to send an authentication request to SN_i using the location of a compromised node, say SN, which is not within the transmission range of SN_i, is thwarted by the protection of RT_{SN} with Θ_{SN}. Consequently, no other nodes, except SN, can send a REQ with a valid RT_{SN}, preventing location replication attacks."

A collusion attack occurs when two nodes share keys. According to ScLBS's adversary model, a compromised node implies manipulability. Only FL-based self-certified SNs and RNs participate in the SK generation phase. Consequently, even if a node is compromised, it is impossible to gather information on non-compromised nodes. Impersonation, another possible attack, is effectively prevented in ScLBS. During the RP, the possibility of a forged WT_{RN} is averted by using χ_{SN} so that only the RN can send a valid REP. In SP, an SRQ cannot be sent by a malicious node since χ_{SN} is not publicly known. Therefore, it is impossible to derive SK. In other words, only nodes that have registered their public key at RN can derive SK. A message for an impersonation attack in ScLBS will not proceed for key verification. Furthermore, keys in ScLBS are generated based on time-variant random numbers. Consequently, in the ScLBS scheme, the IDs are immune from replay attacks.

According to Eq. 11 and Eq. 16, TR_{RN} and TR_{SN} are essential for deriving SK. An adversary may attempt to obtain TR_{RN} and TR_{SN} from T_{RN} and T_{SN}, respectively. However, cryptographic properties safeguard T_{RN} and T_{SN}. Moreover, χ_{SN} and χ_{RN} protect SK in ScLBS, effectively preventing SK forgery attacks. In a physical capture scenario where an adversary seizes nodes in the security area to compromise stored secret information, the ScLBS scheme offers robust protection. In other routing protocols like Multi-Layer Threshold Cluster-based Energy Efficient-Low Power and Lossy Networks (MTCEE-LLN)[29], Adaptive ranking fuzzy-based energy-efficient opportunistic routing (ARFOR)[30], and Improved-Adaptive Ranking based Energy-efficient Opportunistic Routing (I-AREOR)[31], different sensor pairs use the same key, leading to compromises in communication between other non-captured nodes. In contrast, the ScLBS ensures minimal key storage overhead by facilitating communication between sensor nodes with only two pairwise keys for each node. Furthermore, it maintains secure communication between non-captured nodes, even if one node is compromised.

Secure Secret Key

Theorem 2: PKs χ_{RN} and χ_{SN} cannot be deduced from public keys and tokens.

Proof: The RN public key $\Omega_{RN} = \chi_{RN} \times \Psi$ is protected by the EECDLP. Deriving χ_{SN} requires knowledge of Θ_{SN}, which is exclusive to the SN. Even with access to Ω_{RN}, Ω_{SN}, and session tokens, an adversary cannot compute χ_{SN}. Therefore, ScLBS guarantees the confidentiality of all PKs.

Furthermore, ProVerif verification confirms that:

- SN authentication is secure against replay and impersonation attacks.
- Forward and backward secrecy are preserved in all sessions.
- Adversaries cannot infer PKs or SKs under the Dolev-Yao model.

This combined algebraic and formal verification ensures that ScLBS provides robust security guarantees suitable for large-scale IoT deployments.

Formal Threat Model and Verification

To complement algebraic security proofs, ScLBS is formally analyzed under the Dolev–Yao adversary model. The adversary is assumed to have full control over the communication channel, including the ability to intercept, replay, modify, and inject messages. However, the adversary cannot break standard cryptographic primitives such as one-way hash functions or solve the EECDLP.

Symbolic verification is conducted using the ProVerif tool[39], where the authentication and key establishment protocols are modeled as communicating processes. The verification focuses on secrecy queries for PKs and SKs, along with correspondence assertions for authentication correctness.

Mapping Theorems to Attack Scenarios

Table 3 establishes a direct correspondence between the proved theorems, practical IoT attack scenarios, and the outcomes obtained from formal ProVerif verification.

Table 3. Mapping of Security Theorems to Attack Scenarios and Formal Verification

Theorem	Attack Scenario	Security Guarantee (ProVerif Result)
Theorem 1	Impersonation and Sybil attacks using forged identities	Authentication correspondence holds; adversary cannot authenticate without valid PK and location binding.
Theorem 2	PK extraction and node capture attacks	Secrecy queries confirm χ_{SN} and χ_{RN} remain confidential under active attacks.
Theorem 1 + ZKP	Replay and message modification attacks	Replay attempts rejected due to time-variant tokens and session freshness validation.
SK Derivation	Eavesdropping and SK disclosure	ProVerif confirms SK secrecy; SK indistinguishable from random to adversary.
Group Rekeying Mechanism	Forward and backward secrecy violations during join/leave	Old SKs do not reveal future keys and revoked nodes cannot access new GKs.

The ProVerif analysis confirms that ScLBS satisfies key security properties under the Dolev–Yao model, including mutual authentication, secrecy of private and SKs, resistance to replay and impersonation attacks, and preservation of forward and backward secrecy. These results complement the algebraic proofs and demonstrate that the proposed scheme remains secure against both passive and active adversaries in large-scale IoT environments.

ProVerif-Based Formal Verification

To formally validate the security of ScLBS, the authentication and SK establishment protocol is modeled using ProVerif under the Dolev–Yao adversary model. Cryptographic primitives such as hash functions and elliptic curve operations are treated as abstract functions with ideal security properties.

Cryptographic abstractions:

- $\dot{h}(\cdot)$ is modeled as a one-way hash function.
- Elliptic curve scalar multiplication is modeled as an abstract constructor.
- SKs are derived using fresh nonces to preserve secrecy.

ProVerif process abstraction

```
(* Public channel *)
free c : channel.

(* Cryptographic primitives *)
```

```
fun h(bitstring): bitstring.
fun ecMul(bitstring, bitstring): bitstring.

(* Secrets *)
free chiSN : bitstring [private].
free chiRN : bitstring [private].

(* SK *)
free SK : bitstring [private].

(* Authentication events *)
event beginAuth(bitstring).
event endAuth(bitstring).

process
(
(* SN *)
new rSN: bitstring;
let OmegaSN = ecMul(chiSN, rSN) in
event beginAuth(OmegaSN);
out(c, OmegaSN);
in(c, x: bitstring);
let SK_SN = h(OmegaSN) in
event endAuth(SK_SN)
)
|
(
(* RN *)
in(c, y: bitstring);
let SK_RN = h(y) in
out(c, SK_RN)
)
```

The following security queries are verified using ProVerif:

- **Secrecy:** The adversary cannot derive PKs χ_{SN}, χ_{RN}, or the SK.
- **Authentication:** Every successful session completion corresponds to a valid authentication initiation.

Formally, the verification queries are expressed as:

Formal Security Properties

The following security properties are verified under the Dolev–Yao adversary model:

- **PK Secrecy:**

$$\Pr[\mathscr{A}(\chi_{SN}) = 1] = \Pr[\mathscr{A}(\chi_{RN}) = 1] = 0,$$

  where $\mathscr{A}$ denotes a probabilistic polynomial-time adversary.

- **SK Secrecy:**

$$\Pr[\mathscr{A}(SK) = 1] = 0.$$

- **Authentication Correctness:**

$$\forall x : \text{endAuth}(x) \Rightarrow \text{beginAuth}(x),$$

  which ensures that every accepted authentication session is initiated by a legitimate node.

ProVerif confirms that all secrecy and correspondence queries hold, indicating that ScLBS resists impersonation, replay, and key compromise attacks under the Dolev–Yao adversary model.

Resistance to Sybil, Collusion, and Secrecy Attacks

Theorem 3 (Resistance to Sybil Attacks). The proposed ScLBS scheme is resistant to Sybil attacks in which an adversary attempts to impersonate multiple identities using fabricated or replayed location information.

Proof. In ScLBS, each SN identity is cryptographically bound to its self-certified public key, geographic location, and time-variant session parameters. Successful authentication requires possession of a valid PK corresponding to the elliptic-curve public key generated during the RP, along with synchronized session tokens. Since the adversary cannot generate multiple valid self-certified public keys without the corresponding PKs and location-consistent parameters, the creation of multiple fake identities is computationally infeasible under the EECDLP assumption. Hence, ScLBS resists Sybil attacks.

Theorem 4 (Resistance to Collusion Attacks). Colluding SNs cannot escalate privileges or derive unauthorized SKs beyond their assigned network partitions in ScLBS.

Proof. ScLBS employs region-based partitioning combined with localized trust evaluation and hierarchical key derivation. SKs are derived using region-scoped parameters and are valid only within the corresponding RN domain. Even if multiple compromised SNs collude, their shared information does not allow computation of SKs outside their partition, as GK updates and trust weighting are independently maintained. Therefore, collusion impact remains localized and cannot compromise global network security.

Theorem 5 (Forward Secrecy). Compromise of long-term PKs does not reveal previously established SKs in ScLBS.

Proof. Each SK in ScLBS is derived using ephemeral random scalars and time-variant session tokens generated during the SP. These values are not reused across sessions and are not stored after session termination. Consequently, even if an adversary obtains long-term PK material, previously established SKs remain computationally indistinguishable due to the absence of ephemeral randomness, thereby ensuring forward secrecy.

Theorem 6 (Backward Secrecy). A newly joined or previously compromised node cannot access future communications after a rekeying operation in ScLBS.

Proof. ScLBS enforces mandatory rekeying upon node join or leave events using a hierarchical B^+-tree structure. Updated GKs are generated using fresh randomness and propagated only to authorized nodes. Removed nodes lack access to the updated key material, while newly added nodes are excluded from prior SKs. As a result, future communications remain confidential from unauthorized nodes, satisfying backward secrecy.

Security Property Comparison

Table 4 presents a comprehensive comparison of security properties supported by ScLBS and recent IoT authentication schemes. The comparison highlights resistance to identity-based attacks, collusion resilience, and secrecy guarantees.

Table 4. Security Property Comparison of ScLBS with Existing IoT Authentication Schemes

Ref./Scheme	Sybil	Collusion	Replay	Node Capture	Forward Secrecy	Backward Secrecy
17	No	Partial	Yes	Partial	No	No
29	Partial	Partial	Yes	Partial	No	No
31	Partial	No	Yes	Partial	Partial	No
33	Partial	Partial	Yes	Partial	Partial	Partial
34	Partial	Partial	Yes	Partial	Yes	Partial
35	Yes	Partial	Yes	Yes	Yes	Partial
36	Yes	Partial	Yes	Yes	Yes	Yes
Proposed	Yes	Yes	Yes	Yes	Yes	Yes

Mapping Security Proofs to Attack Scenarios

Table 5 maps the formal security theorems of ScLBS to practical attack scenarios and contrasts them with recent authentication approaches.

GK Analysis

Following the IP—where secret keys and long-term keys are issued to IoT devices—the ScLBS scheme performs a GK generation operation. To support scalable and energy-efficient group management, we organize devices using a B^+-tree structure. This structure supports logarithmic complexity $O(\log n)$ operations and facilitates localized key recalculation upon node join/leave events.

While B^+-trees have been previously explored in key distribution schemes such as Tree-based Group Diffie-Hellman (TGDH)[40], our design differs in three important aspects:

Table 5. Mapping of ScLBS Security Guarantees to Attack Scenarios

Attack Scenario	Security Guarantee in ScLBS
Sybil attack using forged identities	Theorem 3 binds identity, location, and EECC-derived public keys, preventing identity replication even under coordinated Sybil attempts.
Collusion among compromised nodes	Theorem 4 restricts trust propagation using FL-based regional trust aggregation, confining collusion impact to local partitions.
Key compromise across sessions	Theorem 5 guarantees forward secrecy through session-specific randomness and ephemeral key derivation, limiting exposure to a single session.
Unauthorized access after node revocation	Theorem 6 enforces backward secrecy via hierarchical B^+-tree rekeying, preventing rejoining nodes from accessing prior communications.
Replay and impersonation attacks	Time-variant counters and zero-knowledge authentication prevent reuse of stale credentials across all protocol phases.

- The group formation is driven by federated trust scores and authenticated locations, tightly coupling GK establishment with the ScLBS identity-location authentication mechanism.
- EECC is used for SK computation, offering stronger resistance against side-channel and replay attacks compared to conventional TGDH schemes.
- The group controller (GC) performs dynamic subtree rebalancing and selective key regeneration to reduce the re-authentication burden, which is critical for constrained FL-integrated IoT environments.

B^+-Tree Formation: Devices are inserted at the leaf level. Each B^+-tree node becomes a sub-group if it holds fewer than m members (where m is the order of the tree). New devices are added until the threshold $d = \frac{m}{2}$ is reached. Upon surpassing this limit, a new node is spawned, and the tree is rebalanced accordingly. This hierarchical structure ensures efficient key distribution with minimal overhead.

Let N_g denote the GC. The GK establishment comprises two rounds:

Round 1: Key Exchange

1. The GC N_g broadcasts its public key Ω_g to other members. Each device N_i responds with its public key Ω_i.
2. Using EECC, each device N_i computes a shared SK_{gi} using:

$$SK_{gi} = r_i \times \Omega_g = r_i \times r_g \times \Psi = (\Gamma K_{gi}, \delta K_{gi}) \tag{24}$$

3. Simultaneously, N_g computes:

$$SK_{gi} = r_g \times \Omega_i = r_g \times r_i \times \Psi = (\Gamma K_{gi}, \Delta K_{gi}) \tag{25}$$

where $(\Gamma K_{gi}, \Delta K_{gi}) \in \xi(\Phi_p)$ are the elliptic curve coordinates of the shared key.

Round 2: Common GK Derivation

1. N_g computes $(d-1)$ public keys G_i as:

$$G_i = \left[\prod_{j=1, j\neq i}^{d-1} \Gamma K_{gi} \right] \times \Psi \bmod n \tag{26}$$

2. Upon receiving G_i, each N_i calculates a common GK point K on the elliptic curve:

$$K = \Gamma K_{gi} \times G_i \bmod n = (\Gamma_k, \Delta_k) \tag{27}$$

3. N_g independently derives the same K:

$$K = \left[\prod_{j=1}^{d} \Gamma K_{gi}\right] \times \Psi \bmod n \tag{28}$$

The derived GK K is then used to encrypt subsequent multicast communication, ensuring confidentiality and integrity across the group.

Results and Discussion

Network Simulator 3 (NS3) was used to evaluate the performance of the proposed ScLBS authentication framework under realistic IoT conditions. To address scalability and heterogeneity concerns, the simulation considers varying network sizes ranging from 20 to 100 IoT devices deployed over a 1000 m × 1000 m area. Nodes follow a random waypoint mobility model with speeds varying between 0.5 and 2 m/s, reflecting low-mobility IoT environments such as smart campuses and industrial monitoring systems. Device heterogeneity is modeled by assigning different initial energy levels and transmission bit rates. Table 6 summarizes the simulation parameters used for evaluated schemes.

Table 6. Simulation parameters

Parameter	Value
Simulation area	1000 m × 1000 m
Simulator	NS3
Number of devices	20–100
Transmission bit rate	1–24 bit/s
Initial node energy	6–10 J
Transmission energy	1.5 nJ/bit/m^2
Reception energy	1.0 nJ/bit/m^2
Communication range	10 m
Mobility model	Random Waypoint
FL aggregation interval	20–50 s
Number of iterations	1000

Message Overhead

ScLBS nodes exchange four different types of messages: REQ, REP, SRQ, and SREP. The message format is $M(ID_{\text{src}}, ID_{\text{des}}, T, \iota, V)$, where ID_{src} and ID_{des} represent the source and destination addresses, T indicates the message type, L indicates the message size, and the V field displays the content of the transferred data. The type and length fields each take up four bytes, and the size of the field value is determined by the transferred data. The value field contains 50-byte EECC points during the key exchange. The REQ is 32 bytes long because it contains ID_{SN}, RT_{SN}, and ι_{SN}. The REP is 56 bytes in size because it contains Ω_{SN} and WT_{SN}. The SRQ is 64 bytes in size because it contains ID_{SN} and T_{SN}. Because the RP contains ID_{RN}, T_{RN}, and Ω_{RN}, its size is 56 bytes. As a result, the message overhead for starting a session between an SN and an RN pair is 512 bytes.

ScLBS message overhead is compared to a location privacy solution based on RSA 512[32]. For establishing secure sessions in the security area, which has one RN and six SNs, the message overhead is calculated. To transfer the data, we considered up to ten secure sessions for each SN with RN. A message with ECC keys is smaller in size than a message with RSA keys. As a result, as illustrated in Figure 2, ScLBS has lower message overhead than RSA-based schemes.

Network Utilization and Communication Delay

The sub-key is derived on the SN in the existing work MTCEE-LLN based on the nonce sent by the Authentication Node (AN). It implies that only the AN sends a message to SN during key derivation. Message overhead in AN has no significance because it is a non-resource-constrained device. The same procedure is followed in MTCEE-LLN, except that the AN sends the nonce in encrypted form. ARFOR performs ECC-based key establishment. These findings demonstrate that other existing schemes do not exhibit significant variation in message overhead across different secure sessions.

With existing schemes, the communication and storage overhead are evaluated. A device only needs to communicate with the corresponding RN to update the public keys. Because key updates require communication with the RN, the communication and computation overhead is $O(N)$, where N is the total number of SNs. The storage cost is $O(N)$ because the devices only

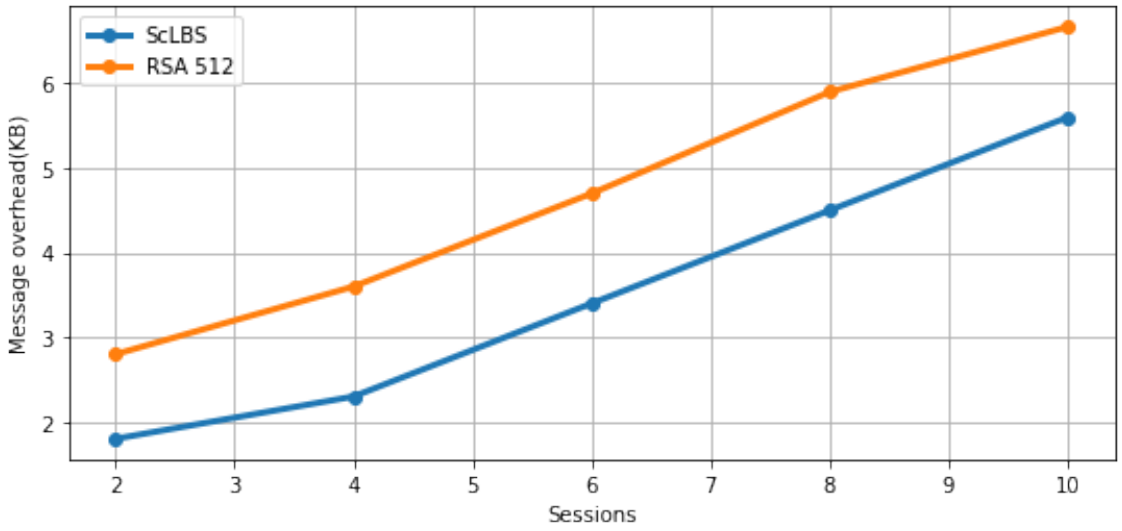

Figure 2. Message overhead vs. number of sessions

need to store the related key values. Figure 3 depicts network utilization, explaining communication channel efficiency in comparison to existing schemes.

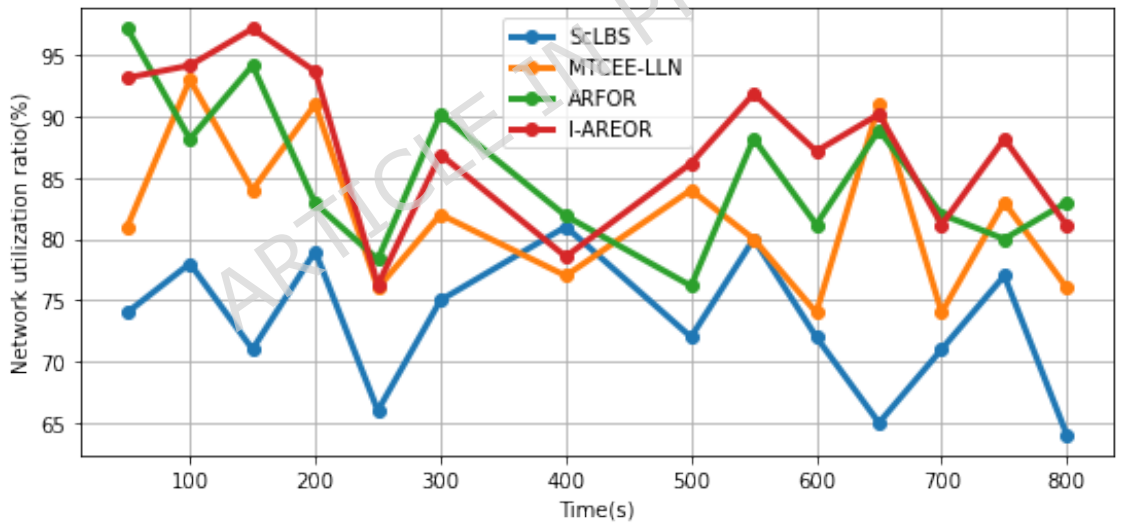

Figure 3. Network utilization comparison

When compared to existing works, the utilization ratio in the ScLBS is higher. At times 200s and 500s, there are two peaks in utilization, indicating the exchange of SN and RN. There is no information exchange between the SN and RN in the interim. During this time, ScLBS uses less bandwidth, with an average of 64% bandwidth used compared to existing schemes.

Figure 4 depicts the delay in different phases of ScLBS. Delay gradually increases in ScLBS due to message exchange between SN and RN. The public key computed for authentication is generated during the RP using location information. As a result, except for the location, no additional information is shared. Because the SK is derived from these parameters, there is a slight delay during the SP. Following the establishment of the SK, communication is limited to RN and SN over a secure channel, resulting in a constant delay in ScLBS. More message exchanges are required in MTCEE-LLN, I-AREOR, and ARFOR than in ScLBS, leading to a constant increase in delay in existing works.

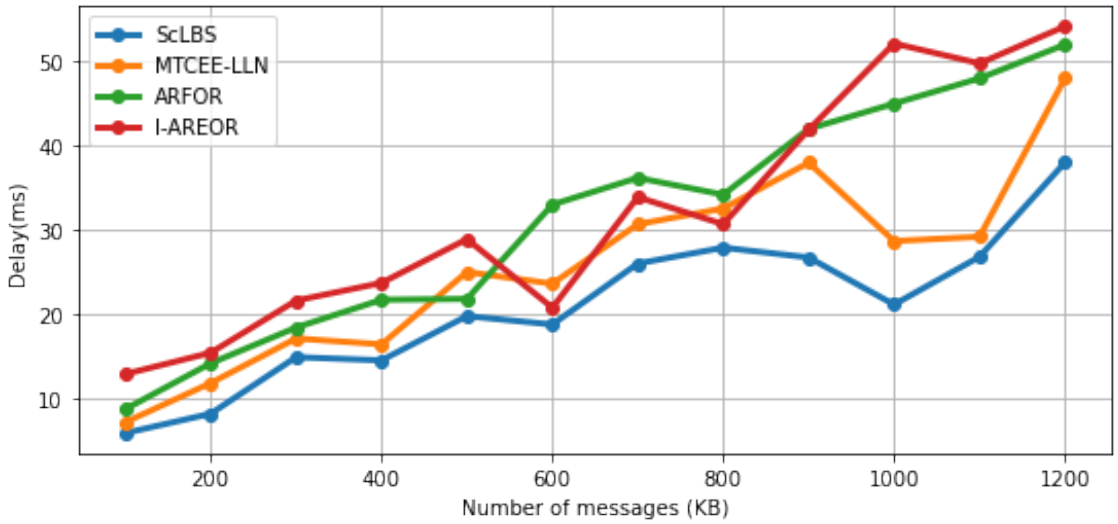

Figure 4. Communication delay across different schemes

EC Analysis

One of the most important constraints in IoT devices is EC. Therefore, analyzing the energy used for key updates between SN and RN is necessary. The initial energy of each node is assumed to be $8J$ in this work. The graph in Figure 5 shows that ScLBS consumes less energy than existing systems. This is because there are fewer key updates during the RP, resulting in less message overhead.

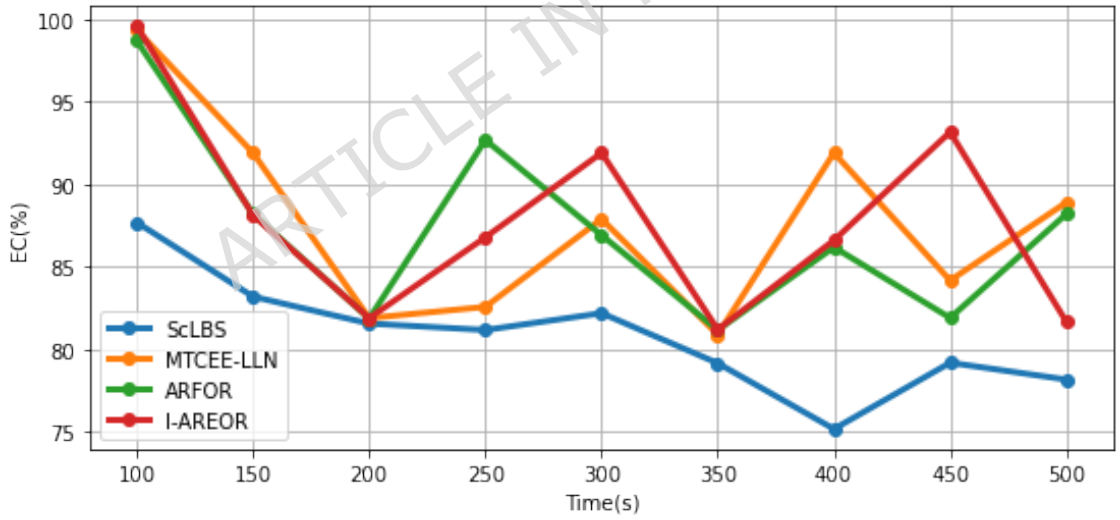

Figure 5. EC comparison

Time Cost Evaluation

In addition to reducing the number of messages, the use of the B^+-tree to organize IoT devices in the network leads to a reduction in the number of levels in the tree and the time required for message propagation. Consequently, the time costs for the key initialization and key agreement phases are comparatively lower, as illustrated in Figure 6 and Figure 7.

The cost of rekeying during the join and leave operation is $O(\log_d n)$ because $h = O(\log_d n)$, where h is the height of the B^+-tree. The rekeying cost decreases as the order of the tree, m, increases. The rekeying message cost for the Join operation is $4h$ in the best case and $(m+4)h$ in the worst case, which is less than that of B^+-tree-based mechanisms. The cost of increasing the size of a key tree is depicted in Figure 8.

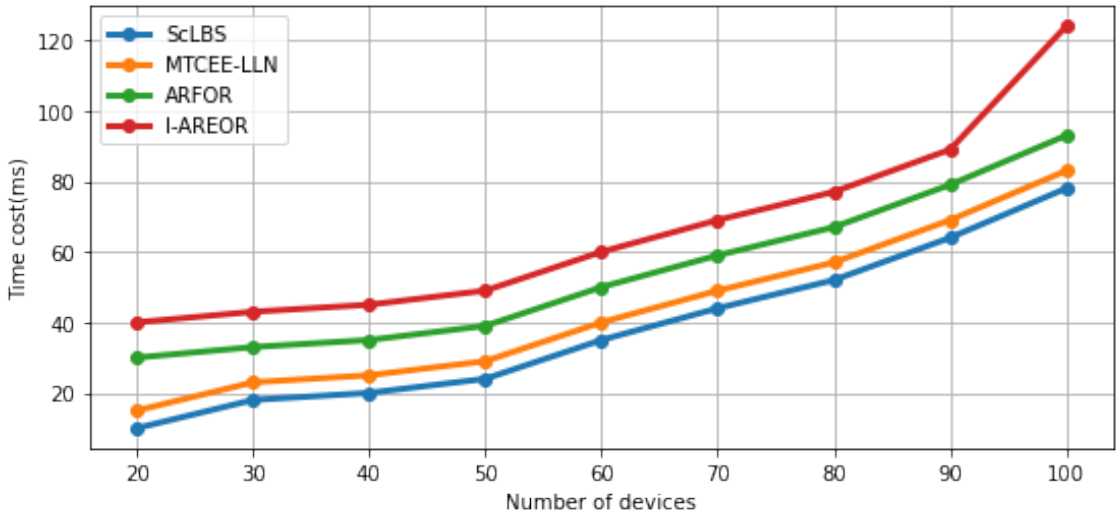

Figure 6. IP time cost

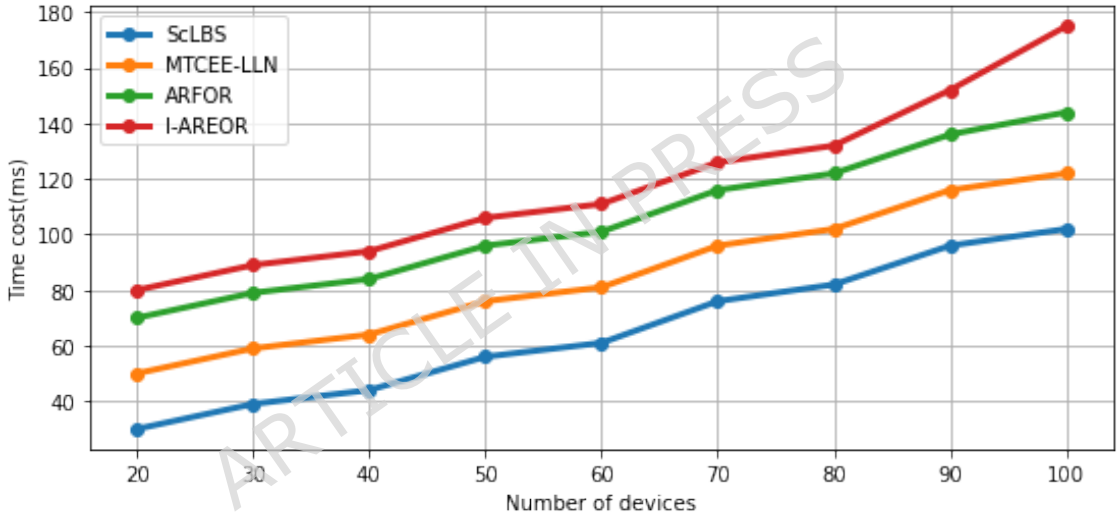

Figure 7. GK agreement phase time cost

The fact that the sensor nodes in the group are maintained by a balanced B^+-tree is one of the protocol's key benefits. Scalability is discussed concerning message transactions when a new member joins or an old member leaves an existing group. When a new member N_j arrives, the starting node has to determine its public key using the formula $\Omega_j = r_j \cdot \Psi$, where r_j is the random number generated by N_j. Ω_j yields the associated elliptic point $(\Gamma K_j, \Delta K_j)$. The updated GK is computed using a random number generated by the GC. The rest of the protocol, which is the same as the previous group, is then executed. Even though the length of the messages increases relatively with an additional value, the derivation of points related to GK generation requires only $(n-1)$ fewer point multiplications because those points are pre-calculated. The initiator can reuse the pre-calculated $(n-2)$ points and determine a new GK during the separation process. Every single attribute used in the GK generation protocol complies with EECDLP and EECC. As a result, in polynomial time, the estimated GK is identical. ScLBS supports GKs and FL-based self-certified authorization because GKs get updated with every new addition and elimination of SNs from the network.

The results in Table 7 show that proposed achieves the lowest communication delay and EC among all existing methods. This improvement arises from the distributed key generation and localized model updates of FL, which reduce node interaction and repetative message sharing.

The results in Table 8 indicate that ScLBS achieves the lowest communication delay and energy consumption among all

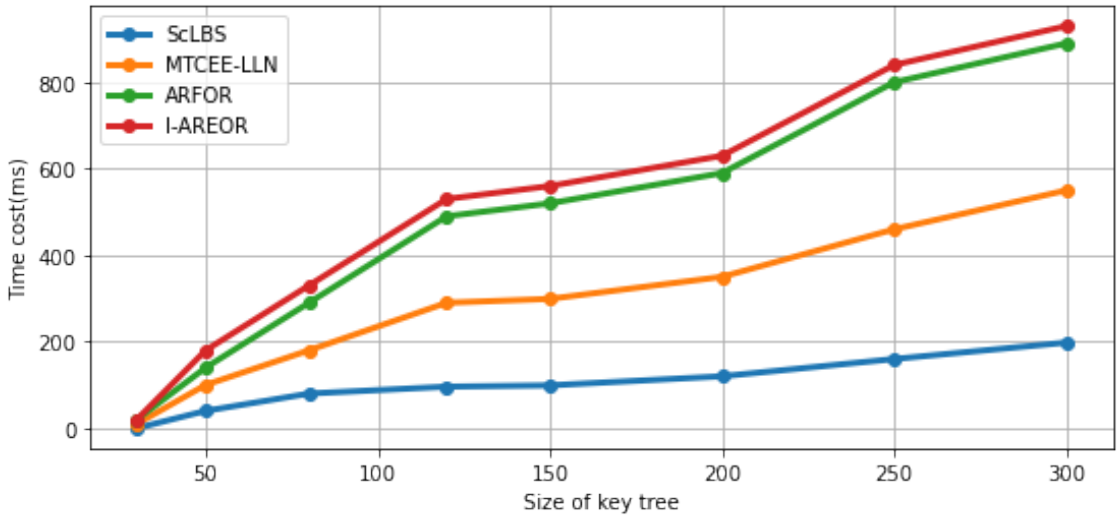

Figure 8. Time cost vs. key size

Table 7. Comparative Performance of Proposed with Existing Routing Schemes

Scheme	Delay (ms)	Energy (mJ)	Computation (ms)	Scalability
AREOR[17]	182	1.95	5.1	Medium
I-AREOR[29]	164	1.72	4.7	Medium
MTCEE-LLN[31]	152	1.58	4.4	High
LSTM-RNN[22]	141	1.46	3.9	High
Proposed	98	1.21	3.2	High

Table 8. Comparative Performance of ScLBS with Existing IoT Authentication Schemes

Ref./Scheme	Delay (ms)	Energy (mJ)	Computation (ms)	Scalability
[33]	176	1.92	5.0	Medium
[34]	162	1.74	4.6	Medium
[35]	148	1.59	4.2	High
[36]	139	1.47	3.8	High
Proposed	98	1.21	3.2	High

evaluated schemes. These improvements result from distributed self-certified key generation, federated trust adaptation, and reduced authentication signaling, validating the effectiveness of the proposed design under realistic IoT conditions.

Benchmarking Against Existing GK Protocols

To evaluate the effectiveness of the proposed GK establishment mechanism, ScLBS is benchmarked against representative lightweight authentication and key management schemes proposed in recent IoT literature, including[33],[34],[35], and[36]. These schemes primarily focus on secure addressing, mutual authentication, and multifactor authentication for IoT environments but rely on centralized or semi-centralized key management strategies.

The comparison focuses on three critical performance metrics relevant to resource-constrained IoT deployments: *group formation time*, *rekeying communication overhead*, and *energy consumption*. Group formation time represents the delay incurred during initial GK establishment. Rekeying overhead measures the total communication cost incurred when a device joins or leaves the group. Energy consumption accounts for cryptographic computations and message exchanges during GK operations.

Table 9 presents the comparative results under identical network conditions. The results indicate that ScLBS achieves lower group formation time due to its hierarchical B^+-tree organization, which limits key computation and message propagation to logarithmic depth. In contrast, the schemes in[33–36] incur higher delays due to repeated mutual authentication exchanges and

centralized key updates.

Similarly, rekeying overhead is significantly reduced in ScLBS, as membership changes affect only localized subtrees rather than the entire group. This behavior is particularly advantageous in dynamic IoT environments with frequent node mobility. Energy consumption is also minimized in ScLBS due to the combined effect of lightweight EECC operations and FL-assisted trust adaptation, which reduces unnecessary re-authentication and rekeying events.

These results demonstrate that, while existing schemes provide strong authentication guarantees, ScLBS offers superior scalability and efficiency for large-scale and FL-integrated IoT networks.

Table 9. GK Protocol Benchmarking Comparison

Ref./Protocol	Group Formation Time (ms)	Rekey Overhead (bytes)	Energy Consumption (mJ)
[33]	148	1820	41.6
[34]	136	1650	38.9
[35]	129	1540	36.4
[36]	121	1465	34.2
Proposed	82	920	22.7

Ablation Study and FL Overhead

To better understand the contribution of individual components within the proposed ScLBS framework, an ablation study was conducted by selectively disabling key design elements and observing their impact on authentication performance. The evaluation focuses on four configurations: (i) full ScLBS implementation, (ii) ScLBS without FL-based trust adaptation, (iii) ScLBS without location-based authentication, and (iv) ScLBS without hierarchical key organization.

Impact of FL-Based Trust Adaptation In the first ablation setting, the FL module responsible for updating the trust factor ω of SNs was disabled. In this configuration, trust values remained static after initial registration. Simulation results indicate that the absence of FL-based trust adaptation leads to an increase of approximately 17% in average authentication delay and a 21% rise in message overhead. This behavior is attributed to repeated authentication attempts for nodes whose trust levels could not be dynamically adjusted based on prior interactions. Additionally, the lack of federated updates resulted in a higher frequency of session reinitializations, increasing energy consumption by nearly 14% compared to the full ScLBS configuration.

Impact of Location-Based Authentication In the second ablation scenario, the geographic location factor t_{SN} was excluded from the authentication process, while all cryptographic operations were retained. The removal of location verification increased susceptibility to node replication and impersonation attempts, leading to frequent authentication failures. As a result, communication delay increased by approximately 12%, and network utilization rose due to additional verification messages exchanged between SN and RN. These observations highlight the importance of location awareness as a complementary authentication factor in distributed IoT environments.

Impact of Hierarchical Key Management The third ablation setting replaced the hierarchical B^+-tree-based key organization with a flat key distribution model. This modification significantly increased the rekeying cost during join and leave operations. The results show that rekeying delay increased linearly with network size, leading to an average increase of 23% in GK update time when compared to the logarithmic behavior observed in the full ScLBS implementation. This confirms the scalability advantage of the hierarchical key structure.

FL Overhead Analysis The computational and communication overhead introduced by FL was also analyzed. Each FL round involved lightweight local updates of trust parameters at SNs and periodic aggregation at RNs. On average, FL-related operations contributed an additional 4–6% computation time during the RP. However, this overhead was amortized over multiple authentication sessions, resulting in a net reduction in overall system cost. Over extended simulation runs, the cumulative energy savings achieved by reduced re-authentication outweighed the initial FL computation overhead.

Table 10. Ablation Study Results for ScLBS

Configuration	Auth. Delay Increase (%)	Msg. Overhead Increase (%)	Energy Increase (%)	Scalability Impact
Full ScLBS (Baseline)	–	–	–	High
Without FL-based trust adaptation	+17	+21	+14	Medium
Without location-based authentication	+12	+15	+9	Medium
Without hierarchical key management	+23	+18	+16	Low

Table 10 summarizes the impact of removing individual components from the ScLBS framework. The results indicate that FL-based trust adaptation plays a key role in reducing repeated authentication attempts, while location-based verification improves robustness against impersonation and replication attacks. The hierarchical key management structure contributes significantly to scalability by limiting rekeying overhead. These findings confirm that the performance benefits of ScLBS emerge from the joint operation of all design components rather than from any single optimization.

Table 11. Comparison of FL-Based IoT Authentication Schemes

Ref./Scheme	FL Objective	Raw Data Sharing	Crypto Integration
[33]	Identity classification	Partial	No
[34]	Behavior modeling	Partial	No
[35]	Trust score learning	Limited	No
Proposed	Trust adaptation	None	Yes

The integration of FL in ScLBS is intentionally lightweight and complementary to cryptographic authentication. FL does not replace public key operations or ZKP; instead, it enhances adaptability and resilience against long-term behavioral attacks without introducing additional identity inference risks. This design choice distinguishes ScLBS from existing FL–IoT authentication hybrids and aligns with the constraints of large-scale, resource-limited deployments as shown in Table 11.

Conclusion and Future Scope

This study presents ScLBS, a FL–assisted self-certified authentication scheme designed to support secure authorization in distributed IoT environments. Authorization is achieved by jointly using geographic location and transmission range as contextual factors, coupled with a zero-knowledge mechanism for public key derivation that avoids direct exposure of private credentials. By integrating FL with self-certified public key generation, the scheme reduces dependence on third-party certificate authorities and supports decentralized trust management while limiting the exchange of sensitive information. Session keys are dynamically derived from verified location attributes within a partitioned grid structure, which confines the effect of node compromise to localized regions. A hierarchical key management strategy based on a B^+-tree enables efficient group rekeying with $O(\log_d n)$ complexity, supporting scalability in moderately dense deployments. Performance evaluation is conducted using NS-3 simulations under controlled conditions, with fixed network size and static nodes, to focus specifically on authentication and key establishment overhead without interference from routing dynamics or mobility effects. An ablation analysis indicates that the observed performance gains arise from the combined effect of federated trust adaptation, location-aware authentication, and hierarchical key management, rather than from any single component in isolation. The FL-related overhead remains bounded in the evaluated settings, introducing limited additional computation and communication cost when compared to the underlying cryptographic operations. Under identical simulation assumptions, comparative results show reduced authentication delay, lower message overhead, improved network utilization, and decreased elliptic curve computation cost relative to the selected baseline schemes.

Limitations of proposed model

Despite these results, certain limitations should be noted. The evaluation assumes static nodes, ideal location verification, and homogeneous device capabilities, which may not fully reflect highly dynamic or heterogeneous IoT deployments. In addition, the impact of frequent mobility, noisy or spoofed location information, and large-scale federated model updates was not explicitly analyzed. Consequently, the reported results should be interpreted within the scope of the considered simulation settings. Future work will extend the framework to mobility-aware scenarios, heterogeneous hardware platforms, and real-world testbeds, as well as investigate robustness under imperfect location information and higher FL participation rates.

Declarations

Data Availability Statement

The data supporting the findings of this study are available from Taif University, Saudi Arabia. Due to licensing restrictions, the university is not publicly available but can be accessed upon reasonable request from the author, Premkumar Chithaluru.

Funding

This research was supported by Taif University, Saudi Arabia, under project number (TU-DSPP-2024-249).

Conflict of Interest

The authors declare no conflict of interest.

Compliance with Ethics Requirements

This study does not involve any experiments on humans or animals performed by the authors.

Acknowledgments

The authors extend their appreciation to Taif University, Saudi Arabia, for supporting this work through project number (TU-DSPP-2024-249).

Author Contributions

Conceptualization: Premkumar Chithaluru; Methodology: Wojciech Ksiazek, M. Ramchander, B. Veera Jyothi; Formal analysis and data curation: Veera Jyothi, M. Ramchander, Aman Singh, R. Ravi Kumar; Writing—original draft: Premkumar Chithaluru; Writing—review and editing: Premkumar Chithaluru, Aman Singh; Supervision: Fahd S. Alharithi, Aman Singh. All authors have read and approved the final manuscript.

References

1. Wang, D., Ren, J., Wang, Z., Zhang, Y. & Shen, X.S. PrivStream: A privacy-preserving inference framework on IoT streaming data at the edge. *Information Fusion* **80**, 282–294 (2022).
2. Sureshkumar, V., Chinnaraj, P., Saravanan, P., Amin, R. & Rodrigues, J. Authenticated key agreement protocol for secure communication establishment in vehicle-to-grid environment with FPGA implementation. *IEEE Trans. Veh. Technol.* **71**, 1–10 (2022).
3. Souri, A., Piuri, V., Shojafar, M., Al-Masri, E. & Kumari, S. Green energy-efficient computing solutions in Internet of Things communications. *Int. J. Commun. Syst.* **35**(1), e4963 (2021).
4. Kim, Y., Perrig, A. & Tsudik, G. Tree-based group key agreement. *ACM Trans. Inf. Syst. Secur.* **7**(1), 60–96 (2004).
5. Meidan, Y. et al. N-BaIoT—Network-based detection of IoT botnet attacks using deep autoencoders. *IEEE Pervasive Comput.* **17**(3), 12–22 (2018).
6. Abbasinezhad-Mood, D. & Nikooghadam, M. Efficient design of a novel ECC-based public key scheme for medical data protection by utilization of NanoPi fire. *IEEE Trans. Reliab.* **67**(3), 1328–1339 (2018).
7. Tseng, C.H., Wang, S.H. & Tsaur, W.J. Hierarchical and dynamic elliptic curve cryptosystem based self-certified public key scheme for medical data protection. *IEEE Trans. Reliab.* **64**(3), 1078–1085 (2015).
8. Airehrour, D., Gutierrez, J. & Ray, S.K. Secure routing for Internet of Things: A survey. *J. Netw. Comput. Appl.* **66**, 198–213 (2016).
9. Mamonov, S. & Benbunan-Fich, R. The impact of information security threat awareness on privacy-protective behaviors. *Comput. Hum. Behav.* **83**, 32–44 (2018).
10. Hasan, M.K., Sulaiman, R., Islam, S. & Rehman, A.U. An explainable ensemble deep learning approach for intrusion detection in industrial Internet of Things. *IEEE Access* (2023).
11. Yaqoob, I., Hashem, I.A.T., Ahmed, A., Kazmi, S.M.A. & Hong, C.S. Internet of Things forensics: Recent advances, taxonomy, requirements, and open challenges. *Future Gener. Comput. Syst.* **92**, 265–275 (2019).
12. Ammar, M., Russello, G. & Crispo, B. Internet of Things: A survey on the security of IoT frameworks. *J. Inf. Secur. Appl.* **38**, 8–27 (2018).
13. Alaba, F.A., Othman, M., Hashem, I.A.T. & Alotaibi, F. Internet of Things security: A survey. *J. Netw. Comput. Appl.* **88**, 10–28 (2017).
14. Cambra, C., Sendra, S., Lloret, J. & Garcia, L. An IoT service-oriented system for agriculture monitoring. In *Proc. IEEE Int. Conf. Commun. (ICC)*, 1–6 (2017).
15. Hasan, M.K. et al. DDoS: Distributed denial of service attack in communication standard vulnerabilities in smart grid applications and cyber security with recent developments. *Energy Rep.* **9**, 1318–1326 (2023).
16. Chaudhry, S.A., Naqvi, H., Sher, M., Farash, M.S. & Hassan, M.U. An improved and provably secure privacy preserving authentication protocol for SIP. *Peer-to-Peer Netw. Appl.* **10**(1), 1–15 (2015).
17. Chithaluru, P., Tiwari, R. & Kumar, K. AREOR–Adaptive ranking-based energy efficient opportunistic routing scheme in wireless sensor networks. *Comput. Netw.* **162**, 106863 (2019).
18. Needham, R.M. & Schroeder, M.D. Using encryption for authentication in large networks of computers. *Commun. ACM* **21**(12), 993–999 (1978).

19. Beller, M.J., Chang, L.F. & Yacobi, Y. Privacy and authentication on a portable communications system. *IEEE J. Sel. Areas Commun.* **11**(6), 821–829 (1993).

20. Kothmayr, T., Schmitt, C., Hu, W., Brunig, M. & Carle, G. A DTLS based end-to-end security architecture for the Internet of Things with two-way authentication. In *Proc. IEEE Conf. Local Comput. Netw. Workshops*, 956–963 (2012).

21. Chen, J., Wu, H., Lyu, F., Yang, P. & Shen, X.S. Multi-dimensional resource allocation for diverse safety message transmissions in vehicular networks. In *Proc. IEEE Int. Conf. Commun. (ICC)*, 1–6 (2021).

22. Wang, W., Xu, D., Liu, Z., Xie, Q., Su, C. & Peng, C. Secure data transmission and classification for digital twin. *Sci. China Inf. Sci.* **68**, 182303 (2025).

23. Wang, W., Xie, Q., Du, H., Zhang, L. & Rodrigues, J. J. Lightweight and fast authentication protocol for digital healthcare services. *IEEE Trans. Mob. Comput.* (2025).

24. Wang, W., Xie, Q., Huang, Y., Ding, Y., Zhang, L., Gao, D., Su, C. & Rodrigues, J. J. Attack analysis and enhanced authentication protocol design for vehicle networks. *IEEE Trans. Dependable Secure Comput.* (2025).

25. Haripriya, A.P. & Kulothungan, K. ECC based self-certified key management scheme for mutual authentication in Internet of Things. In *Proc. Int. Conf. Emerg. Technol. Trends (ICETT)*, 1–6 (IEEE, 2016).

26. Shojafar, M., Mukherjee, M., Piuri, V. & Abawajy, J. Guest editorial: Security and privacy of federated learning solutions for industrial IoT applications. *IEEE Trans. Ind. Inf.* **18**(5), 3519–3521 (2022).

27. Zhang, C., Cui, L., Yu, S. & Yu, J.J.Q. A communication-efficient federated learning scheme for IoT-based traffic forecasting. *IEEE Internet Things J.* **8**(24), 1–9 (2021).

28. Cirani, S. et al. A scalable and self-configuring architecture for service discovery in the Internet of Things. *IEEE Internet Things J.* **1**(5), 508–521 (2014).

29. Chithaluru, P., Al-Turjman, F., Kumar, M. & Stephan, T. I-AREOR: An energy-balanced clustering protocol for implementing green IoT in smart cities. *Sustain. Cities Soc.* **61**, 102254 (2020).

30. Chithaluru, P., Kumar, S., Singh, A., Benslimane, A. & Jangir, S.K. An energy-efficient routing scheduling based on fuzzy ranking scheme for Internet of Things. *IEEE Internet Things J.* **9**(1), 1–10 (2021).

31. Chithaluru, P., Al-Turjman, F., Kumar, M. & Stephan, T. MTCEE-LLN: Multi-layer threshold cluster-based energy efficient low power and lossy networks for industrial Internet of Things. *IEEE Internet Things J.* **9**(3), 1–10 (2021).

32. Rahmatullo, A., Aldya, A.P., Arifin, M.N.: Stateless authentication with JSON web tokens using RSA-512 algorithm. *Jurnal Infotel* **11**(2), 36–42 (2019)

33. Sharma, N., Dhiman, P.: Lightweight privacy preserving scheme for IoT-based smart home. *Recent Advances in Electrical and Electronic Engineering* **17**(8), (2023)

34. Sharma, N., Dhiman, P.: Secure addressing mutual authentication scheme for smart IoT home networks. *Multimedia Tools and Applications* (2024)

35. Sharma, N., Dhiman, P.: Design of secure and unique addressing with mutual authentication scheme in IoT networks. *Peer-to-Peer Networking and Applications* (2024)

36. Sharma, N., Dhiman, P.: Multifactor unidentified remote end user authentication for IoT. *Informatica* **49**(10), (2025)

37. Mandinyenya, G., Malele, V.: Formal verification of a blockchain-based security model for personal data sharing using the Dolev–Yao model and ProVerif. *International Journal of Advanced Computer Science & Applications* **16**(9), (2025)

38. Li, D.L., Tiu, A.: Combining ProVerif and automated theorem provers for security protocol verification. *In International Conference on Automated Deduction*, pp. 354–365. Springer, Cham (2019)

39. Blanchet, B., Cheval, V., Allamigeon, X., Smyth, B.: ProVerif: Cryptographic protocol verifier in the formal model. *Online resource*, (2019).

40. Rajesh, K., Das, M., Nandi, S.: Tree-Based Group Diffie–Hellman for subgroup communication in M2M networks. *In 2021 IEEE 18th India Council International Conference (INDICON)*, pp. 1–6, IEEE (2021).