Peer-to-Peer Networking and Applications (2024) 17:665–684
https://doi.org/10.1007/s12083-023-01611-9

PGASH: Provable group-based authentication scheme for Internet of Healthcare Things

Chandan Trivedi[1,2] · Keyur Parmar[1] · Udai Pratap Rao[3]

Received: 19 June 2023 / Accepted: 13 December 2023 / Published online: 12 January 2024

Abstract

Electronic healthcare based on medical sensors is now developing to incorporate a significant amount of the Internet of Things (IoT) to communicate between sensors and intended recipients. The key requirements in this domain are to exchange messages safely and to provide confidentiality during communication. Designing and implementing an authentication strategy is essential for resolving security concerns, but it is also challenging to work with constrained computing and processing resources during group communication. Standard one-to-one authentication models do not consider the scalability of resource-limited nodes, which is a vital factor to deal with. However, group authentication presents a unique concept for IoT nodes that verify group members concurrently. The conventional group authentication methods based on the IoT are vulnerable to security risks and cannot defend against attacks like replay attacks, forgery attacks, or unauthorized key distribution by the group manager. In this paper, we propose a dynamic and provable group authentication scheme (GAS) based on a secret sharing scheme that can withstand the dishonest behavior of group managers. We introduced a key updating scenario with a provable group authentication model for dynamic node leaving and joining. Our system complies with the requirements for secrecy and accuracy, and based on security analysis, it is resistant to attacks, as mentioned earlier. Performance analysis and security proof show that our approach performs well in terms of computation cost for group members while maintaining security.

Keywords Healthcare · Security · Group authentication · Secret sharing · IoHT

1 Introduction

The proliferation of wearable technology and devices within the Internet of Things (IoT) has brought about significant progress in medical sensors, particularly in the realm of health monitoring for advanced e-health applications. Wireless physiological sensors are essential in measuring patients' body indicators such as weight, oxygen, temperature, pulse rate, and blood sugar under healthcare monitoring. Additional smart devices can serve as actuators, delivering automated treatments in response to the readings provided by the sensors [1]. At a specific period or the users' request, these wearable body devices transmit readings to medical practitioners, as well as to gateway nodes or controller devices (such as wearable smart things) [2].

Figure 1 presents an overview of healthcare monitoring enabled by IoT, which can be viewed as the Internet of Healthcare Things (IoHT) environment. In this environment, the data collected from various sensors can be processed and consolidated as integrated healthcare data, which can then be utilized for subsequent actions. The operations within the IoHT environment adhere to the three-layer architecture of IoT. The base of architecture is the sensing layer that senses data from sensor devices deployed on patients' bodies. The network layer communicates and transmits data to its upper layer, and an application layer is responsible for processing

✉ Udai Pratap Rao
udai.cs@nitp.ac.in

Chandan Trivedi
chandan.trivedi@nirmauni.ac.in

Keyur Parmar
keyur@coed.svnit.ac.in

[1] Computer Science and Engineering, Sardar Vallabhbhai National Institute of Technology, Surat 395007, Gujarat, India

[2] Computer Science and Engineering, Institute of Technology, Nirma University, Ahemdabad 382481, Gujarat, India

[3] Computer Science and Engineering, National Institute of Technology, Patna 800005, Bihar, India

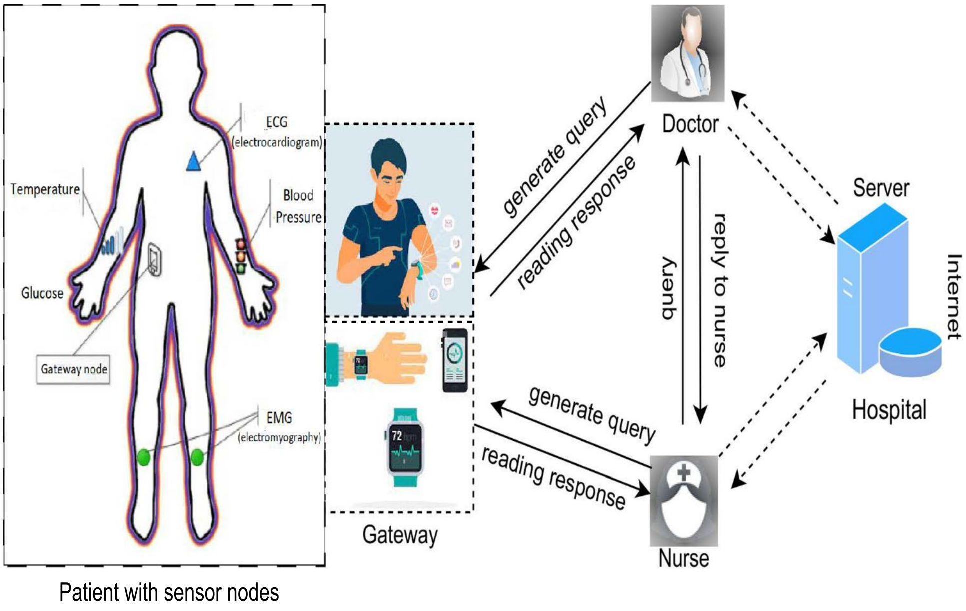


Fig. 1 Overview of healthcare monitoring with IoT devices [3]

data gathered from sensors to specific applications. However, the health details of an individual are a potential target for attackers, so there is a convincing reason to authenticate and securely transmit confidential health information and preserve privacy [4].

With the expansion of healthcare monitoring through medical IoT sensors, it becomes crucial to address concerns regarding the efficient and secure transmission of health information. It is essential to acknowledge the sensitivity of personal medical readings and ensure they are not disregarded [5]. The IoHT environment is deployed with resource-limited medical equipment and wearable sensors having embedded communication and computation capabilities. Connecting healthcare devices for implementing the concept of monitoring in hospitals and personal care clinics encounters several issues, such as securing communication during multicast messages.

One of the most challenging tasks is establishing and managing secure group communications [6] and authentication [7] of participating IoT nodes. However, installing cryptographic keys within a group of nodes is also essential to protect information transmission. A key is necessary to encrypt data for secrecy, generate Message Authentication Codes (MAC) for integrity, or verify and sign messages for authenticity and confidentiality. It signifies that a non-member device can't read a group message [8]. The property of integrity is attained when all end nodes within a group receive the transmitted information accurately. Authenticity, on the other hand, shows that devices or nodes ascertain the identity of the sender [6].

Group-oriented authentication has numerous applications in the IoT, which can be applied to healthcare. Fouda et al. [9] came up with a smart grid-oriented scheme that uses hash functions and joint session keys to execute mutual authentication between distributed smart devices and meters. However, their scheme only provides mutual authentication and does not consider attack scenarios in the group authentication process if used in the IoHT environment. Shun et al. [10] presented a wearable sensor-based scheme leveraging ECC to accomplish user and mutual authentication in healthcare centers. Their scheme does not consider group-based authentication for devices which is more efficient than one-to-one authentication and it reduces the computation cost for the IoHT scenario when massive requests are handled by the group manager. Fang et al. [11] used a biometric identification scheme to establish mutual authentication between users and sensing devices in the wireless sensor network (WSN) scenario. However, the proposed biometric scheme considers three-factor authentication, which is suitable for one-to-one authentication of devices. If it is used in the IoHT scenario with group authentication, it may have more computation cost due to more operations in their scheme and does not provide any mechanism for the dishonest behavior of the group manager. IoT devices in healthcare often operate in environments where simplicity and ease of use are crucial. Introducing a three-factor authentication system may increase complexity and potentially impact user-friendliness. In healthcare settings, where quick and efficient access to information is vital, overly complex authentication processes may hinder workflow. Zhang et al. [12] has introduced the idea of a key agreement scheme based on certificate-less and aggregated signatures. Their scheme if used in IoHT-based group authentication, may not handle massive requests on the group manager, and the authentication key is not generated after leaving the group vulnerable to key compromisation. Li and Liu [13] put forward an authentication scheme based on wireless radio frequency, utilizing a dual physical unclonable function (PUF) identity, specifically in the context of Radio Frequency Identification (RFID). The scheme aims to enable two-way authentication between tags and readers. The scalability of PUF-based solutions may be a concern in large-scale IoT healthcare deployments. As the number of devices increases, managing and securing a large number of unique PUFs may become a complex task. Also, implementing PUF-based security solutions may add to the overall cost of IoT devices. In healthcare, where cost-effectiveness is often a critical factor, the expense associated with PUF implementation may be a limiting factor.

One-to-one communication only has a small number of receivers and can interact with two parties. However, group communication allows unicasting, multicasting, and broadcasting in a distributed healthcare environment where many IoT devices act as a single group unit and disseminate information to several stakeholders. Therefore, secure group communication is motivation and a key requirement for working in the healthcare environment. The majority of IoT authentication schemes follow a node-to-node authentication approach, where smart devices engage in interactions to verify identity information. However, these authentication schemes are unsuitable for the IoHT and future IoT applications, where many devices will join and leave the network. Furthermore, IoT nodes are characterized by their limited resources, including small memory and power consumption capabilities. As a result, they face constraints in supporting higher communication overhead and complex computing tasks. As the use of IoT nodes and sensor devices is growing exponentially in the healthcare environment, the authentication server faces difficulty in dealing with massive authentication requests, while trending intelligent applications need to authenticate participating nodes effectively [14]. In conclusion, there is a need for a novel authentication scheme including an identity factor that enables the verification of the legitimacy within the participating nodes of the group simultaneously. Such a scheme would help reduce costs and enhance efficiency in IoT systems. Identity-based authentication is crucial in the context of group authentication for several reasons, particularly when it comes to managing and securing access to resources, systems, or data for groups of users. Dynamic group membership, access control, and individual accountability are some key reasons why identity-based authentication is required in group authentication scenarios. Therefore, it is necessary to introduce a lightweight group authentication scheme (GAS) based on provable security for the IoHT.

In a GAS, the prover is responsible for generating authentication proofs, and the verifier verifies these proofs to ensure the authenticity of the group members. The prover interacts with the verifier by presenting proofs that demonstrate the possession of certain credentials or secret keys by each member of the group. By authenticating multiple nodes at once, GAS reduces the overhead associated with individual authentication in traditional schemes. It can enhance efficiency and scalability in scenarios where a large number of nodes need to be authenticated within a group. IoHT, with the condition of massive authentication requests, can adequately untangle the problem of limited node resources and much computation during authentication. Authenticating the identity of groups in the Internet of Things (IoT) poses several challenges, such as resource constraint, dynamic membership, resilience to attacks, key distribution and management, and trust establishment. Many IoT devices have limited computational power, memory, and energy resources. Robust authentication mechanisms may be resource-intensive, making it challenging to implement on resource-constrained devices while ensuring the security of the authentication process. In IoT environments, group memberships can be dynamic and change frequently. Devices may join or leave a group at any time, making it challenging to maintain accurate and up-to-date group membership information. IoT environments are susceptible to various cyber-attacks, including replay attacks, man-in-the-middle attacks, and impersonation attacks. Group authentication mechanisms must be designed to be resilient against these threats. Establishing trust among group members is essential for secure communication. However, in dynamic IoT environments, trust relationships need to be established quickly and reliably, even among devices that have not interacted before. Hence, conducting research on group authentication in the context of IoT devices holds the utmost importance as it offers a viable solution to the existing challenges associated with authenticating the identity of groups in IoT. This research not only addresses the specific problem of group identity authentication but also provides practical and substantial value to the field.

1.1 Contributions

Below, we present a summary of our contributions:

- In our work, we applied the private key provability concept in a novel way to the GAS in the IoHT scenario. We validate the distribution of private keys for deceptive group managers.
- The solution we propose allows for dynamic change in the group, along with the manager modifying the private key at run time. This modification facilitates the granting or revocation of group member privileges and leaves out the need to reissue the private key to every individual member.
- Our work incorporates confidentiality and correctness along with the prevention of malicious attacks like impersonation, replay, and forgery attacks, ensuring security requirements.

- We propose a scheme with lower computational and communication costs than the current typical GAS that satisfies the requirements of resource-constrained nodes for lightweight group authentication. Our scheme is better suited to IoHT applications with a high volume of authentication requests, such as healthcare group communication.

1.2 Paper organization

Section 2 of this paper presents related work. Section 3 covers the mathematical and theoretical concepts required to develop a scheme. Section 4 illustrates the proposed scheme, encompassing its key components. Moving forward to Section 5, a comprehensive examination of the scheme's security and performance analysis is presented. Lastly, Section 6 serves as the concluding section, summarizing the key findings and concluding remarks of the paper.

2 Related work

In this section, we present related work on group authentication schemes. We also highlight research gaps and the need for a lightweight authentication scheme.

Among the methods for realizing group authentication, the SSS scheme [15] is an efficient and low-cost method. This security method divides the secret into shares, which should include no information about the secret. A minimum quantity of shares is required to access the secret via SSS. This threshold-based process represents the bare minimum of shares required to discover the secret. It is known as complete secrecy, as it prevents an adversary from learning more about the secured secret if they find any number of shares below the threshold. Harn's [16] scheme applies Shamir's secret sharing to group authentication, which is more efficient than traditional authentication. However, Ahmadian and Jamshidpour [17] highlighted the insecurity of Harn's scheme, demonstrating that attackers can forge a legitimate authentication token without the need to recover any polynomials. It shows that an attacker can impersonate a group member without detection, effectively bypassing the identity authentication process. Following that, Chien [18] suggested a GAS based on bilinear pairings. The security of this scheme relies on the difficulty of the ECDLP. In each cycle of group authentication, a distinct primitive meta-encrypted private key is utilized, effectively preventing replay attacks throughout the authentication process.

In contrast, Xia et al. [19] used an anonymous veto network algorithm to overcome the problem of token modification. However, their scheme GAS ignores the problem of group manager deception or dishonesty. During the authentication stage, there is a possibility that the group manager might mistakenly distribute an incorrect private key, failing group members to pass the authentication process. They also highlighted that the scheme lacks resistance against forgery attacks due to the publicly computable nature of the lagrangian coefficient. This vulnerability enables attackers to create a fresh, legitimate token by manipulating the lagrangian coefficient within the authentication token.

Aydin et al. [20] came up with an idea based on the *GAS* of Harn and Chien and presented a lightweight scheme within a group. Despite using computationally efficient simple accumulation operations by group members in their scheme, it fails to provide resistance against forgery attacks. However, the lagrangian coefficient modification process remains the same as defined in Xia et al. [19]. Additionally, the scheme does not offer protection against replay attacks, allowing attackers to replay the authentication token of valid group members to bypass the subsequent cycle of group authentication.

Park and Park [21] has introduced a selective group authentication for medical IoT scenarios using a secret sharing scheme assuming a selection of things based on user request. However, they have not considered registration authority and authentication server as a medium of authentication for each thing when added or removed during group updates, resulting in computation overhead and low scalability. Work by Lee and Lee [22] shows the group authentication in the smart metering environment under the smart home application. Their scheme utilizes the secret sharing scheme for intra-group and mutual authentication with the server. However, their scheme only resists replay attacks.

The framework proposed by Wang et al. [23] shows the authentication based on the central service provider and PDA devices. They used the secret sharing scheme with elliptic curve cryptography to resist multiple attacks, but forgery attacks and impersonation were not considered under provable security. Further, Tan and Chung [24] introduced the group key distribution scheme for WBANs based on smartphone ECG sensors and key management for validated sensors. They have slightly modified the dynamic key changing mechanism at the sensor side and use certificateless biometric authentication. However, their scheme does not discuss group manager dishonesty and private key verification concerning the healthcare scenario. A method proposed by Khatoon et al. [25] shows a mutual authentication scheme in a telemedicine system and uses an ECDLP-based security mechanism. The main focus of their approach is the registration, authentication, and password-changing phase. However, they have not included the concept of private key verification during the dynamic setting of group members.

Table 1 shows a brief idea of related group authentication schemes; based on the same the existing group authentication schemes have the following limitations:

- According to our literature study, group authentication schemes that rely on secret sharing technology cannot withstand forgery attacks. In such attacks, an attacker

Table 1 Related work and their significance in IoT

Paper	Contributions and Benefits	Limitations
Ahmadian and Jamshidpour [17]	Secret sharing based group authentication scheme and present how an attacker can forge authentication token and resolve the same	Scheme discussed the only about bypassing authentication phase but verification of private key is left unattended
Chien [18]	Authentication scheme based on bilinear pairing and ECDLP, Prevents replay attack by utilizing private key for each cycle of authentication	Dynamic behavior of nodes are not covered with respect to change in the private key. However, the scheme does not support massive authentication requests.
Xia et al. [19]	Introduced token modification algorithm based on secret sharing scheme and defend against replay and impersonation attack	Possibility that the group manager might mistakenly distribute an incorrect private key, failing group members to pass the authentication process and lack in resistance to forgery attack
Aydin et al. [20]	Efficient computation using accumulation operations and lagrangian coefficient based solution for authentication	Fails to resist against forgery attack and allowing attackers to replay the authentication token of valid group members to bypass the subsequent cycle of group authentication
Park and Park [21]	Authentication based on selecting specific things from a group and uses ECC-based operations for security	Scheme does not consider registration authority and authentication server as a medium of authentication for each dynamic addition and withdrawal
Lee and Lee [22]	Present group authentication considering the smart home application and ‘SSS’ for mutual authentication in intra-group communication	Scheme can only resist replay attack and does not provide resistance to other security attacks.
Wang et al. [23]	Discuss authentication based on ECC and ‘SSS’ for PDA and service provider	Provable security and forgery attack was not a part the scheme
Tan and Chung [24]	Scheme is for group key management and biometric authentication with sensor’s dynamic key using certificate-less method	Scheme not discuss group manager dishonesty and private key verification concerning the healthcare scenario
Khatoon et al. [25]	A mutual authentication scheme in a telemedicine system that uses ECDLP and covers registration, authentication, and password-changing phase	Not included the concept of private key verification during the dynamic setting of group members

can manipulate the coefficient within the authentication token to create a counterfeit token, enabling them to pass the group authentication process successfully.

- According to our literature survey, group authentication schemes cannot defend against replay attacks. In IoT, the latency of communication between nodes can lead to a failure in group authentication. In this case, an attacker can intercept and replay the authentication token from a previous cycle, thereby successfully passing the group authentication process.
- Existing group authentication schemes do not take into account the possibility of deception by the group manager, as group members blindly accept the private key provided by them. In the context of IoHT, the group manager, a gateway or other high-capacity device with computing power and storage capacity, may intentionally provide incorrect private keys to certain legitimate group members. As a result, the group authentication process consistently fails to succeed.
- The group authentication schemes examined in existing literature are computationally demanding and are unsuitable for resource-constrained IoT scenarios. Consequently, minimizing the computational burden on IoT nodes during the authentication process is essential. Group authentication, in comparison to one authentication, is efficient and scalable with less burden on sensor nodes.

Within the Internet of Healthcare Things (IoHT) scenario, the nodes exhibit dynamic behavior, where new nodes can join the network while existing nodes may depart. A prime illustration of this is observed in the healthcare domain, where medical sensors placed on a patient's body act as a group of IoT nodes. These nodes undergo dynamic changes based on the patient's requirements [26]. However, in the *GAS* of the literature [18–20], the joining and withdrawal of nodes have not been taken into consideration. As the nodes undergo dynamic changes, a new node can successfully pass group authentication by utilizing the assigned private key. Conversely, a leaving node cannot pass group authentication using its original private key. Thus, the manager needs to update the node's private key to fulfill the revocation/grant of permissions. Considering the literature, it is essential to design a provable, lightweight, and secure *GAS* under IoHT group communication scenarios that support dynamic changes of group members concerning healthcare applications and their deployment in the real world.

3 Background study and preliminaries

The primary focus of this section is to provide an overview of the theoretical foundation utilized in the proposed scheme. Additionally, it presents the security-related definitions crucial for understanding and evaluating the scheme's security aspects. Table 2 shows various terms used in the paper.

Table 2 List of abbreviations and symbols used in the report and proposed scheme

Abbreviations	Meaning	Abbreviations	Meaning
SSS	Shair's Secret Sharing	T_{EA}	the time for an ECC point addition
IoHT	Internet of Healthcare Things	$T_{(mul,q)}$	time of the last multiplication operation
IoT	Internet of Things	T_{pair}	time of a bilinear pairing operation
PGASH	Provable Group Authentication Scheme in IoHT	T_a	run time for an algorithm
GAS	Group Authentication Scheme	$T_{(inv,q)}$	the time of the last inverse operation
n	number of members in group	r	random number generate by members
M_i	i^{th} member of group	T	
S_O	secret owner	v	verification value during node leaving
k	threshold in number of members in *SSS*	R, S, F	real, simulator and forging respectively
Z_q	finite field with q as prime number	h	random number by group manager during private key updating
$f(x)$	polynomial function of coefficient a_i and variable x_i	λ	initialization input
s_i	secret/private key	ϕ	random session number
H	reconstruction function under *SSS*	ε	non-negligible function for comparing probability
G_1, G_2	additive cyclic group of q	G_M	group manager
G_T	multiplicative cyclic group of order q	q	prime number
P_1, P_2, Q	$P_1, P_2 \in G_1$, and $Q \in G_2$	u	non-member of group
a, b	parameter $\in Z_q$	*Prob*	probability
$e(\cdot)$	function of bilinear pairing	c_i	a part of authentication token
$h(\cdot)$	hash function of $(0, 1)^* \in Z_q$	I_A, Ex_A	internal and external adversary
T_{EM}	time of an elliptic curve point multiplication	T_{htp}	time of one hash mapping to a point operation
Sys_{param}	System parameter during initialization	pkG_M, skG_M	public-private key pair of group manager

3.1 Privacy and security essentials in IoHT

This subsection emphasizes the key security requirements and their significance for IoHT architecture.

- **Authentication of Users**: The susceptibility of wireless communications to access by unauthorized users is the key issue in wireless healthcare environments. So, it becomes imperative to incorporate a resilient authentication process where each user must authenticate their identity before accessing the patient's physiological information [27]. The user and the healthcare nodes must mutually authenticate in real-time to ensure communication is secure [28].
- **Confidentiality**: Personal health data is highly susceptible, and healthcare devices are wireless, so patient physiological information needs confidentiality and should be protected from passive threats such as eavesdropping or traffic monitoring [29]. The clinical data of patients are also only seen or accessed by authorized practitioners.
- **Low computation and communication cost**: As wireless medical sensors are constrained devices and IoT-based healthcare often needs space for the execution of their operations, the protocol must be accurate in terms of connectivity and computational costs [27]
- **Data freshness**: Professionals in the healthcare field rely on regular access to up-to-date patient physiological data to maintain fresh and live medical records. The freshness of this data is vital as it prevents adversaries from replaying outdated observations or treatments [2].
- **Session key establishment**: In order to facilitate secure communication between a healthcare user and a sensor node, it is necessary to establish a session key that enables subsequent interactions to occur safely and securely [8, 30].
- **Resistance to common attacks**: In legitimate healthcare environments, the framework should be protective against numerous common threats, such as theft attacks, information leakage attacks, password guessing attacks, replay attacks, and impersonation attacks [29, 31].
- **Convenience**: The architecture of IoT-enabled healthcare systems should prioritize user-friendliness and ease of deployment. For instance, users should be able to securely change their credentials whenever desired, ensuring a seamless and convenient experience [32].

In this paper, we focus on security requirements such as confidentiality, authentication of users, low computation and communication costs, and security against replay and forgery attacks.

3.2 Preliminary for proposed scheme

This subsection discusses fundamental concepts and mathematical terminology used for the proposed scheme.

- **Shamir's secret sharing scheme (SSS)** [15]**:** The 'SSS' scheme contains n group members $M_1, M_2, \ldots M_n$ and a secret owner S_O. Here, in this case, S_O distributes secrets to at least k $(k < n)$ group members who can reconstruct secrets, where k represents the threshold for secret recovery. The initialization, secret distribution, and reconstruction are major steps involved in this scheme [20]
- **Bi-linear pairing** [33]**:** The bi-linear pairing establishes a mapping denoted as $e(\cdot)$ from $G_1 \times G_2$ to G_T, where G_1 and G_2 represent the additive cyclic group with order q, and G_T represents the multiplicative cyclic group with order q. In our scheme, it can be used during ECC-based multiplication and provide robust security for operations. The bi-linear pairings adhere to the following three properties:
  - **Bi-linear:** For $P_1, P_2 \in G_1, Q \in G_2$ and $a, b \in Z_q$, there exist properties such as $e(aP_1, bQ) = e(P_1, Q)^{ab}$ and $e(P_1 + P_2, Q) = e(P_1, Q)e(P_2, Q)$
  - **Non-degenerate:** For any point $P \in G_1, Q \in G_2$, have $e(P, Q) \neq 1$, and vice versa.
  - **Computability:** Efficient computation of $e(P, Q)$ for all $(P, Q) \in G_1 \times G_2$ can be achieved within polynomial time.
- **Elliptic curve discrete logarithmic problem (ECDLP)** [34]**:** Suppose E is an elliptic curve with a finite field over Z_q and Q, P is a point on E that satisfies $Q = kP$, where $k \in Z_q$; then the ECDLP is a challenging problem means: given Q, P, it is hard to find or solve it for k.

3.3 Security models and its requirements

This subsection defines the applicable terminologies of the security proofs, models, and attack scenarios for a provable group authentication scheme.

3.3.1 Adversary model

This paper assumes that the adversary is divided into external attackers (Ex_A) and internal attackers (I_A). The precise definition of the aforementioned is outlined as follows:

- **Internal Attackers** (I_A)**:** Suppose the m members are participating in group authentication within a range of k to n. Here, k is the limit value of the group's authenticated members, and n is the number of group members. An internal attacker (I_A) controls at most $k - 1$ group members so that he can obtain the private keys and secret information. I_A intends to obtain the private keys of unrestrained group members and take out authentication tokens for completing group authentication. Here, we consider the dishonest behavior of group members that obtain private keys

- **External Attackers** (Ex_A)**:** An external attacker (Ex_A) should not get a valid private key for any group member. The objective of an Ex_A is to assume the identity of a group member and carry out actions without raising suspicion or detection. This type of attack attempts to gain unauthorized access to the targeted organization's network. External attackers are showing dishonest behavior of group members that is impersonating.

3.3.2 Communication model

We assume a secure link between the Group Manager (G_M) and every group member is in place. It ensures the secure distribution of private keys without any risk of leakage. Additionally, all group members can connect to a broadcast channel, allowing them to receive any message sent within a designated time frame.

3.3.3 Provable group authentication (PGA) framework

A provable group authentication framework [19] is defined with five stages of algorithms:

1. **Initialization** ($Init(\lambda)$) **:** The initialization algorithm is executed by the G_M with parameters λ, which produce the system parameters Sys_{param} for further stages.
2. **Private key distribution** ($prkeyDist(\{x_i\}_{i\in[1,n]}$)**:** The G_M uses the private key distribution algorithm and the input to the system consists of the set of public identity information for each group member $\{x_i\}_{i\in[1,n]}$. The output, on the other hand, comprises the set of private keys for each group member $\{s_i\}_{i\in[1,n]}$. These private keys are subsequently transmitted to the respective group members over a secure communication link.
3. **Private key verification** ($prkeyVeri(s_i, Sys_{param})$)**:** Every group member executes the verification algorithm, employing their key s_i (i.e., private key) and the Sys_{param} as inputs. The algorithm yields an output of 1 if the verification succeeds for a private key; otherwise, it produces an output of 0.
4. **Authentication token generation** ($AuthtokenGen$ $(Sys_{param}, \phi, x_i, s_i)$**:** Each group member actively engaged in group authentication executes the authentication token generation algorithm. Consider a scenario where $M_1, M_2, \dots, M_m$ represents the group members actively involved in group authentication. In this case, the session index (ϕ), system parameters Sys_{param}, private key s_i, and the collection of public identity information of group members $\{x_i\}_{i\in[1,m]}$ serve as inputs. As a result, the authentication token output is obtained in the form of $\{c_i, r_iP\}$.
5. **Group authentication** ($GAS(Sys_{param}, \{c_i, r_iP\}_{i\in[1,m]}$)**:** Every group member involved in authentication executes the group authentication algorithm. The set of authentication tokens $\{c_i, r_iP\}_{i\in[1,m]}$ and system parameters Sys_{param} serve as inputs. If the authentication process fails, the output is 0; else, the output is 1.

Our proposed scheme uses the defined group authentication to achieve the security requirements. Specific steps such as verification and token generation are used to resist the dishonesty of the group manager and dynamic node joining or leaving scenarios.

3.3.4 Security essentials

To effectively counter security attacks, authenticate participating nodes, and ensure confidentiality, the previously defined PGA framework must adhere to the following security requirements.

- **Correctness:** Assuming the valid group members $(M_1, M_2, \dots, M_m)$ taking part in GAS, we can formally declare the group authentication as successful if the below expression holds.

$$\begin{aligned}&Prob[Sys_{param} \leftarrow Init(\lambda); \{s_i\}_{i\in[1,n]} \leftarrow prkeyDist(\{x_i\}_{i\in[1,n]}),\\&prkeyVeri(s_i, Sys_{param}) = 1|_{i\in[1,n]},\\&c_i, r_iP \leftarrow AuthtokenGen(Sys_{param}, \phi, \{x_i\}_{i\in[1,m]}, s_i),\\&GAS(Sys_{param}, \{c_i, r_iP\}_{i\in[1,m]}) = 1] = 1\end{aligned}$$

  Consequently, we can affirm that the group authentication scheme is correct. The above expression incorporates the term $Prob(X)$, which signifies the probability of event X.
- **Confidentiality:** The Internal attackers or adversaries (I_A) can not obtain any secret information of unrestrained group members during the group authentication phase. Formally, if the difference of $Prob[ViewI_A(Real_R(\lambda, Sys_{param}))]$ and $Prob[ViewI_A(SIM_S(\lambda, Sys_{param}))]$ is less than $\epsilon(\lambda)$, then the group authentication scheme can achieve confidentiality. Here, $ViewI_A(Real_R(\lambda, Sys_{param}))$ indicates the view of the actual/real operating scheme as (R) for internal attackers (I_A). While $ViewI_A(SIM_S(\lambda, Sys_{param}))$ represents a view of a simulator (S) that takes public attributed as input for I_A, and the term $\epsilon(\lambda)$ refers to a negligible function having the attribute λ".
- **Unforgeability:** It is infeasible for internal attackers (I_A) to fabricate an authentication token to bypass group authentication successfully. The GAS is unforgeable if it satisfies the following expression:

$$\begin{aligned}&Prob[Sys_{param} \leftarrow Init(\lambda), \{s_i\}_{i\in[1,n]} \leftarrow prkeyDist(\{x_i\}_{i\in[1,n]}),\\&prkeyVeri(Sys_{param}, s_i) = 1|_{i\in[1,n]},\\&C \leftarrow A_I^R(Sys_{param}, Z, \{x_i\}_{i\in[1,m]}, \{s_i\}_{i\in M_A}),\\&(\phi \notin Z) \wedge GAS(Sys_{param}, C) = 1] < \epsilon(\lambda)\end{aligned}$$

Here, M_A represents the group members controlled by I_A that satisfy $M_A \subset M$ and $M_A \leq k - 1$. S represents a simulator used for requesting group authentication services, Z is an indexed collection representing the requested session, and $\varepsilon(\lambda)$ indicates a negligible function with the input parameter.

- **No impersonation:** External attackers (Ex_A) cannot assume a group member's identity and thus cannot provide security. The group authentication scheme is not impersonating if the following expression is satisfied.

$$Prob[Sys_{param} \leftarrow Init(\lambda)); prkeyVeri(s_i, Sys_{param}) = 1|_{i\in[1,n]},$$
$$c_i, r_iP \leftarrow AuthtokenGen(Sys_{param}, \phi, \{x_i\}_{i\in[1,m]} \cup x_u, \{s_i\}_{i\in[1,m]}),$$
$$c_u, r_uP \leftarrow Ex_A(Sys_{param}, \phi, \{x_i\}_{i\in[1,m]} \cup x_u, c_i, r_iP_{i\in[1,m]}),$$
$$GAS(Sys_{param}, \{c_i, r_iP\}_{i\in[1,m]\cup M_u}) = 1] < \varepsilon(\lambda)$$

Here, we assumed that Ex_A impersonates or pretends to be a group member M_u, where $u \notin [1, m]$.

4 Proposed scheme

In order to address the issue of identity authentication failures among group members, resulting from the distribution of unauthorized private keys by the group manager, as well as to cater to the group authentication needs of resource-limited IoT nodes, we introduce a solution called *PGASH* (Provable Group-based Authentication Scheme for IoHT). This proposed scheme is specifically designed to be lightweight and tailored for the IoHT context, emphasizing efficient group communication while ensuring the provability of private keys.

4.1 System model

Figure 2 shows the group communication of the proposed scheme under the IoHT, and further, this requires group authentication for confidentiality and identifying the legitimacy of group members. The system model depicted in Fig. 3 comprises two distinct member types: the Group Manager (G_M) and the group members. The role of G_M involves establishing and updating system parameters and distributing personal data to group members through secure links. Group members, on the other hand, communicate via wireless links to verify the identities of fellow group members. In the context of the IoHT, G_M can represent utilities such as smart gadgets and wearable gateways, while group members can encompass IoT nodes like sensors, implantable devices, or smart wearables.

In this paper, the PGASH scheme operates under the assumption that the identity of the Group Manager (G_M) is legitimate. However, it acknowledges the possibility of the G_M engaging in fraudulent or dishonest behavior towards certain group members by distributing incorrect private keys. Consequently, the group members must verify the authenticity of the private key issued by G_M subsequent to its distribution. Group members need to register themselves on the server for their identification purposes, and the G_M will distribute private key to valid group members only.

Our scheme considers a total number of individuals, denoted as m (i.e., $k \leq m \leq n$), where k represents the threshold for the required number of members for successful group authentication and n is the number of group members. Throughout the group authentication phase, the PGASH can withstand collusion between a maximum of $k - 1$ internal group members.

4.2 Proposed PGASH scheme

The *PGASH* scheme is structured into six distinct stages to facilitate its operation. The initial three stages encompass

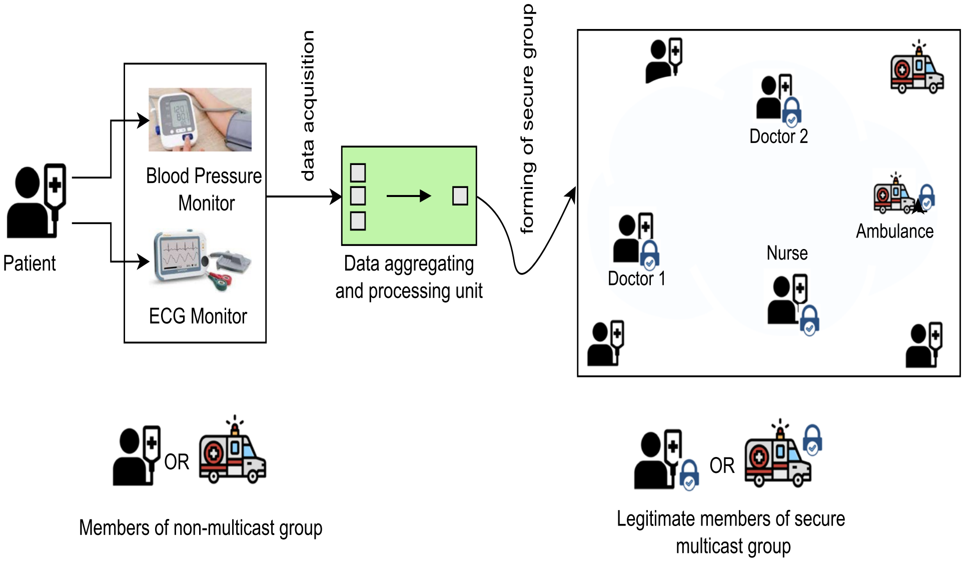


Fig. 2 IoHT group communication environment of proposed system model

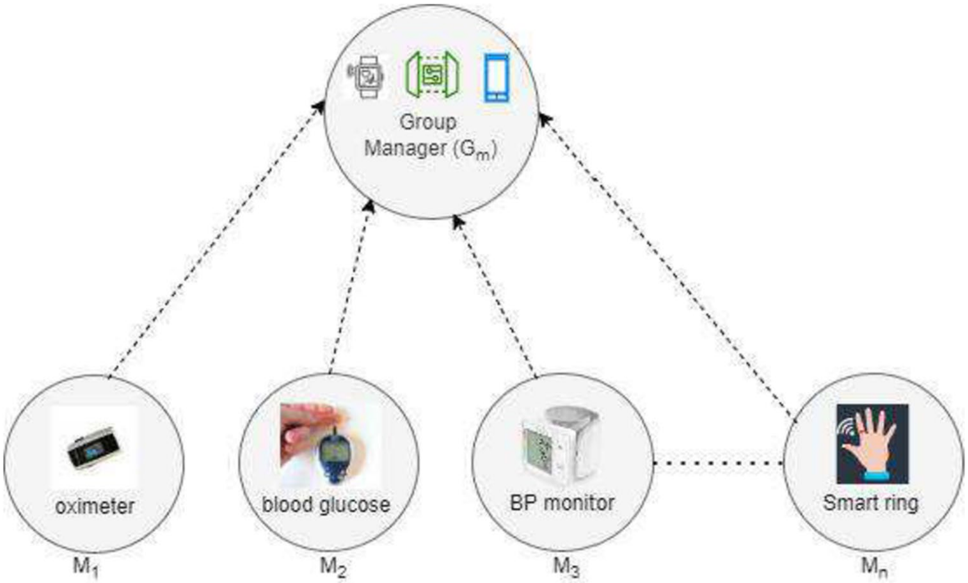


Fig. 3 System Model

the initialization, distribution, and verification of private keys. The remaining three phases involve token generation, group authentication, and addressing dynamic changes in the IoHT environment. Figures 4 and 5 illustrate the flow and process of the PGASH scheme, respectively, providing comprehensive details. The following sections outline the specifics of each stage within the scheme.

1. **Initialization** ($Init(\lambda)$)**:** Enter security parameters λ during initialization, and then G_M chooses two additive cyclic groups, G_1 and G_2, which have a large prime order denoted as q. Furthermore, G_M chooses a multiplicative cyclic group denoted as G_T with an order of q. The generators of G_1 and G_2 are represented by P and R respectively, which are selected to establish the bi-linear pairing $e(\cdot) : G_1 \times G_2 \rightarrow G_T$. Further choose one $k-1$ secret polynomial of degree $f(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1} (modq)$ and let the secret information $s = a_0$, and calculate $Q = sP, v_j = e(a_j P, R)$, where $j \in [0, k-1]$. Now select the hash by using $h(\cdot) : \{0,1\}^* \rightarrow Z_q$, and generate the key pair of $G_M\{pkG_M, skG_M\}$. Finally, all the public parameters of the system are viewed as

$$Sys_{param} = (G_1, G_2, G_T, Q, R, P, \{v_j\}_{j \in [0,k-1]}, h(\cdot), e(\cdot), pkG_M)$$

2. **Distribution of private keys** ($prkeyDist(\{x_i\}_{i \in [1,n]})$**:** Assume group M have n group members $M_1, M_2, \ldots, M_n$, the group manager G_M based on the public identity of the group members x_i calculates $s_i = f(x_i)$, and it is known as the private key of them. Distribute this key to the group members M_i $(i = 1, 2, \ldots, n)$ over a secret channel/link.

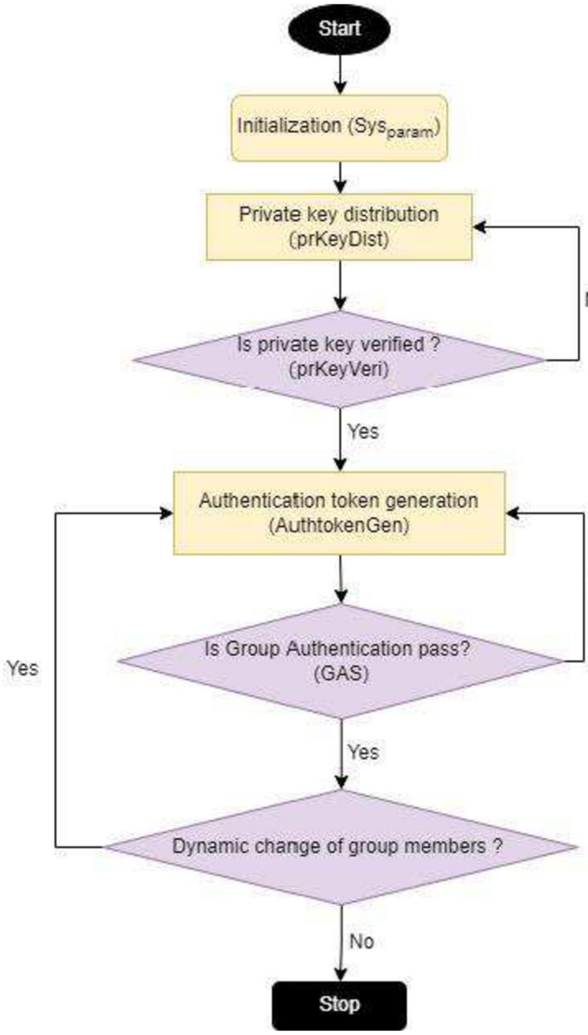
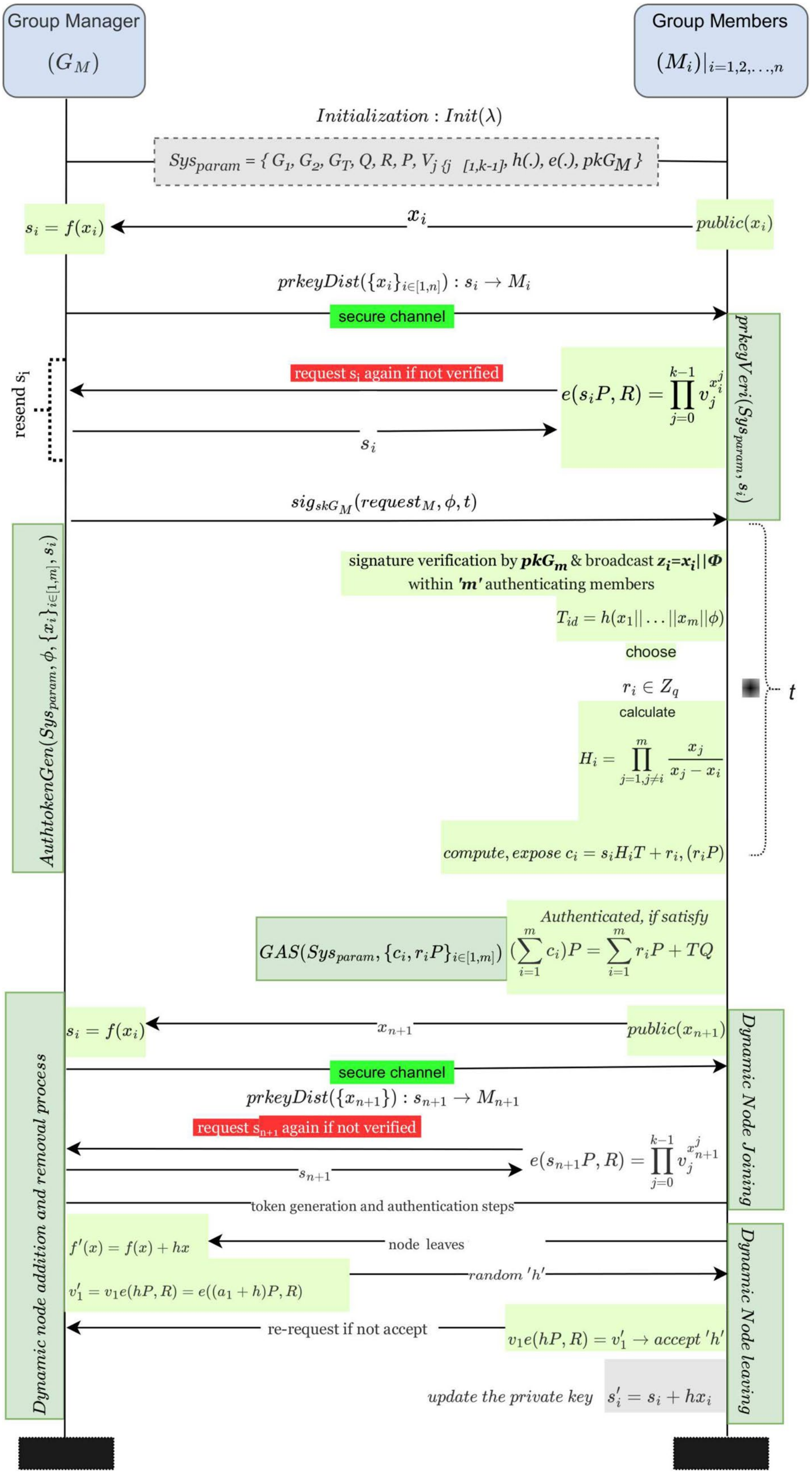

Start
Initialization (Sys_{param})
Private key distribution
(prKeyDist)
Is private key verified ?
(prKeyVeri)
No
Yes
Authentication token generation
(AuthtokenGen)
Is Group Authentication pass?
(GAS)
No
Yes
Yes
Dynamic change of group members ?
No
Stop

Fig. 4 Flow of proposed PGASH scheme

Fig. 5 Detailed illustration of Proposed PGASH scheme


Group Manager
(G_M)
Group Members
$(M_i)|_{i=1,2,...,n}$
$Initialization : Init(\lambda)$
$Sys_{param} = \{ G_1, G_2, G_T, Q, R, P, V_{j\{j\ [1,k-1]}, h(.), e(.), pkG_M \}$
$s_i = f(x_i)$
x_i
$public(x_i)$
$prkeyDist(\{x_i\}_{i\in[1,n]}) : s_i \rightarrow M_i$
secure channel
$prkeyVeri(Sys_{param}, s_i)$
resend s_i
request s_i again if not verified
$e(s_iP, R) = \prod_{j=0}^{k-1} v_j^{x_i^j}$
s_i
$sig_{skG_M}(request_M, \phi, t)$
$AuthtokenGen(Sys_{param}, \phi, \{x_i\}_{i\in[1,m]}, s_i)$
signature verification by pkG_m & broadcast $z_i = x_i || \Phi$ within 'm' authenticating members
$T_{id} = h(x_1||...||x_m||\phi)$
choose
$r_i \in Z_q$
calculate
$H_i = \prod_{j=1,j\neq i}^{m} \frac{x_j}{x_j - x_i}$
$compute, expose\ c_i = s_iH_iT + r_i, (r_iP)$
t
$GAS(Sys_{param}, \{c_i, r_iP\}_{i\in[1,m]})$
Authenticated, if satisfy
$(\sum_{i=1}^{m} c_i)P = \sum_{i=1}^{m} r_iP + TQ$
Dynamic node addition and removal process
$s_i = f(x_i)$
x_{n+1}
$public(x_{n+1})$
Dynamic Node Joining
secure channel
$prkeyDist(\{x_{n+1}\}) : s_{n+1} \rightarrow M_{n+1}$
request s_{n+1} again if not verified
$e(s_{n+1}P, R) = \prod_{j=0}^{k-1} v_j^{x_{n+1}^j}$
s_{n+1}
token generation and authentication steps
$f'(x) = f(x) + hx$
node leaves
Dynamic Node leaving
$v_1' = v_1e(hP, R) = e((a_1 + h)P, R)$
$random\ 'h'$
re-request if not accept
$v_1e(hP, R) = v_1' \rightarrow accept\ 'h'$
$update\ the\ private\ key\ \ s_i' = s_i + hx_i$

3. **Verification of private keys** ($prkeyVeri(s_i, Sys_{param})$)**:** Upon collecting the private key assigned or shared by G_M, the members M_i verify relation $e(s_iP, R) = \prod_{j=0}^{k-1} v_j^{x_i^j}$ to validate the correctness of the private key s_i. If the relation holds, the private key s_i is accepted. However, if it does not hold, the group members request the private key s_i from G_M again. So, the group member who is not registered can not participate in the verification process, and the relationship does not hold if any member tries to impersonate.
4. **Authentication token generation** ($AuthtokenGen$ $(Sys_{param}, \phi, x_i, s_i)$)**:**

   (a) G_M sends a message to the group as sig_{skG_M} $(request_M, \phi, t)$, where $request_M$ is the GM-oriented authentication request, t is the limit for response duration, ϕ is the present session sequence number, and $sig_{skG_M}(\cdot)$ is the signature of the message by the group manager. Subsequently, the G_M updates the session sequence number.

   (b) Group members take public key (pkG_M) and verify the signature by G_M. If the verification is successful, proceed to the first ϕ session. Suppose the members in group authentication are $M_1, M_2, \dots, M_m$, and each broadcast $z_i = x_i || \phi$ within the group as a reply message.

   (c) Upon the reply within the designated duration t, each participating group member M_i calculates the present session id $T_{id} = h(x_1|| \dots ||x_m||\phi)$, choose a random number $r_i \in Z_q$, and calculate the lagrangian coefficient $H_i = \prod_{j=1, j\neq i}^{m} \frac{x_j}{x_j - x_i}$. Now authentication key is computed by using $c_i = s_iH_iT + r_i$ and r_iP, and expose the authentication token $\{c_i, r_iP\}$

5. **Group authentication** ($GAS(Sys_{param}, \{c_i, r_iP\}_{i\in[1,m]})$)**:** Within the designated period t, when group member M_i (where i ranges from 1 to m) receives the authentication token disclosed by other group members, each member proceeds to verify the relation $(\sum_{i=1}^{m} c_i)P = \sum_{i=1}^{m} r_iP + TQ$. This relation performs validation for the identity of group members. If it is successfully established, the group authentication process is considered successful. However, if it fails to be established, indicating a mismatch in identities, the group authentication is deemed unsuccessful, and another round/cycle of group authentication is initiated.
6. **Dynamic change stage :** During changes in group membership, group members possess the capability to join or leave the group dynamically. The role of the Group Manager (G_M) involves updating the private keys of group members and adjusting their permissions, granting or revoking them as necessary. In the subsequent cycle of authentication, recently joined members can utilize their private keys to complete the group authentication process successfully. Conversely, members who have left the group cannot utilize their real private keys to bypass the group authentication. Further, we specify the procedure for joining and leaving group members with a change in authentication key based on the private key as follows:

   (a) **Joining a group**

   - If M_{n+1} want to join the group $M = M_1, M_2 \dots, M_n$, then G_M calculates $s_{n+1} = f(x_{n+1})$ according to publicly identifiable information x_{n+1} of M_{n+1}. The resulting value is regarded as the private key of the group member and is subsequently distributed over a secure channel to M_{n+1}.
   - After collecting the private key s_{n+1} assigned by G_M, M_{n+1} verifies the relation $e(s_{n+1}P, R) = \prod_{j=0}^{k-1} v_j^{x_{n+1}^j}$ to validate the authenticity and private key. s_{n+1} is accepted if the relation holds. However, if it does not hold, M_{n+1} requests the s_{n+1} from G_M once again.
   - If the private key is authentic, then the authentication token generation will generate a new token which is further passed to the authentication stage, which is different from the previous authentication because of the private key change.

   (b) **Depart/leave a group**

   - Assume M_n want to leave the group $G_M = M_1, M_2, \dots, M_n$, then G_M chooses random number $h \in Z_q$, and update the secret polynomial $f(x)$ and v_1 for $f'(x) = f(x) + hx = a_0 + (a_1 + h)x + \dots + a_{k-1}x^{k-1}(mod\ q)$ and get $v_1' = v_1e(hP, R) = e((a_1 + h)P, R)$
   - The group manager transmits the value hash to the members M_i, where i ranges from 1 to $n - 1$. Each member, M_i, verifies the relation $v_1e(hP, R) = v_1'$ to determine its validity. If it holds, the value of h is accepted. However, if it does not hold, the group members request h from G_M again. Subsequently, the group members M_i (where i ranges from 1 to $n - 1$) update their private keys as $s_i' = s_i + hx_i$.
   - As per the update in private of remaining group members, the authentication token will be generated again, and the updated token will go to the group authentication stage for generating a new authentication key and discard the previous authentication key.

Once dynamic changes occur within the group, such as members joining or departing, group authentication can be initiated.

5 Security and performance analysis

In this section, we provide mathematical proof for the confidentiality, correctness, and unforgeability of the PGASH. Additionally, we analyze its resilience against group manager deception and replay attacks. The security of the PGASH relies on the ECDLP, forming its foundation for protection.

5.1 Security proofs

- **The PGASH scheme adheres to correctness** Prove that the members participating in GAS are among the 'n' group members:
  - Let's assume the set $M_1, M_2, \ldots, M_m$, where m varies from k to n, and k represents the minimum number of authenticated groups. By applying the Lagrange interpolation theorem, the secret value can be expressed as $s = f(0) = \sum_{i=1}^{m} f(x_i)H_i = \sum_{i=1}^{m} s_iH_i$. In this, H_i denotes the lagrangian coefficient, defined as $H_i = \prod_{j=1,j\neq i}^{m} \frac{x_j}{x_j - x_i}$, and the group membership token is represented as $c_i = s_iH_iT + r_i$. With this information, we can establish the following.

$$\sum_{i=1}^{m} c_i = T\sum_{i=1}^{m} s_iH_i + \sum_{i=1}^{m} r_i = sT + \sum_{i=1}^{m} r_i,$$
$$\left(\sum_{i=1}^{m} c_i\right)P = (sT + \sum_{i=1}^{m} r_i)P = \sum_{i=1}^{m} r_iP + TsP$$
$$= \sum_{i=1}^{m} r_iP + TQ$$

  As a result, the successful group authentication is confirmed by the establishment of the verification formula $(\sum_{i=1}^{m} c_i)P = \sum_{i=1}^{m} r_iP + TQ$
  - Let's assume that M_{n+1} has joined the group $M = M_1, M_2, \ldots, M_n$. In the scenario where the group members participating in GAS are $M_1, M_2, \ldots, M_m, M_{n+1}$, where m ranges from $k-1$ to n, the secret value can be determined using the Lagrange interpolation theorem as $s = f(0) = \sum_{i=1}^{m} s_iH_i + s_{n+1}H_{n+1}$. Here, $H_i = \prod_{j=1,j\neq i}^{m} \frac{x_j}{x_j - x_i} \frac{x_{n+1}}{x_{n+1} - x_i}, |i = 1, \ldots, m$, and $H_{n+1} = \prod_{j=1}^{m} \frac{x_j}{x_j - x_{n+1}}$ represent the lagrangian coefficients. The group membership token is calculated as $c_i = s_iH_iT + r_i$. Now, the formula can be verified through relation $(\sum_{i=1}^{m} c_i + c_{n+1})P = \sum_{i=1}^{m} r_iP + r_{n+1}P + TQ$.
  - Let's consider the scenario where M_n withdraws/leaves from the group $M = M_1, M_2, \ldots, M_n$. In this case, the group members participating in GAS are $M_1, M_2, \ldots, M_m$, where $k \leq m \leq n-1$. The key s'_i of members $M_i (i = 1, 2, \ldots, m)$ is defined as $s'_i = f'(x_i) = f(x_i) + hx_i$, based on the Lagrange interpolation theorem. According to this theorem, the secret value can be calculated as $s = f'(0) = \sum_{i=1}^{m} s'_iH_i$, where $H_i = \prod_{j=1,j\neq i}^{m} \frac{x_j}{x_j - x_i}$ represents the Lagrangian coefficient. The group membership token is given by $c'_i = s'_iH_iT + r_i$. Similarly, the formula can be verified by establishing $(\sum_{i=1}^{m} c'_i)P = \sum_{i=1}^{m} r_iP + TQ$, thereby completing the proof.
- **The PGASH scheme satisfies confidentiality:** To demonstrate that $Real_R(\lambda, Sys_{param})$ represents the actual running scheme on R, while $SIM_S(\lambda, Sys_{param})$ is a scheme simulated by a simulator S utilizing public attributes as input, we can derive the proof from *Lemma 1.* The details of the two operational schemes are presented below.

1. $Real_R(\lambda, Sys_{param})$
   (a) **Initialization stage:** G_M creates parameters: $Sys_{param} = (G_1, G_2, G_T, Q, R, P, \{v_j\}_{j\in[0,k-1]}, h(\cdot), e(\cdot), pkG_M)$
   (b) **'prkey' distribution:** G_M compute the key $s_i = f(x_i)$ and send it to members over a secure channel. Assuming an I_A know at most $k-1$ private keys of group members $s_1, s_2, \ldots, s_{k-1}$.
   (c) **'prkey' verification:** Each group member verifies the relation $e(s_iP, R) = \prod_{j=0}^{k-1} v_j^{x_i^j}$ is established.
   (d) **Authentication token generation stage:** Let's assume that the members participating in GAS are a subset of the n group members, specifically denoted as $M_1, M_2, \ldots, M_m$, where m ranges from k to $n \in \phi$. During this session, every group member M_i participating in GAS selects a random number $r_i \in Z_q$ and generates the session id as $T = h(x_1 || \ldots ||x_m||\phi)$. In this session, $H_i = \prod_{j=1,j\neq i}^{m} \frac{x_j}{x_j - x_i}$ computes and exposes the authentication token by using $c_i = s_iH_iT + r_i$ and riP. At this stage, the I_A knows $k-1$ private keys randomly from the group members $s_1, s_2, \ldots, s_{k-1}$ and all public parameters.
   (e) **Group authentication:** Each group member verifies the below relation for the authentication result

$$\left(\sum_{i=1}^{m} c_i\right)P = \sum_{i=1}^{m} riP + TQ$$

2. $SIM_S(\lambda, Sys_{param})$
   (a) **Initialization stage:** Consider public parameters of the output of simulator S as $Sys_{param} = (G_1, G_2, G_T, Q, R, P, h(\cdot), \{v_j\}_{j\in}$ $[0, k-1], e(\cdot), pkG_M)$.

(b) **‘prKey’ distribution:** S send private keys $(s_1, s_2, \dots, s_{k-1})$ of $k-1$ group members to I_A.

(c) **‘prKey’ verification:** Members verifies the relation $e(s_iP, R) = \prod_{j=0}^{k-1} v_j^{x_i^j}$ for using private key in next stage.

(d) **Authentication token generation stage:** It is assumed that among the n group members, the members $M_1, M_2, \dots, M_m$ are participating in authentication. Here, $k \le m \le n \in \phi$. During this session, Simulator S send $r_1, r_2, \dots, r_{k-1}$ to I_A, Then S randomly chooses $m-k$ from Z_q. Now value for $c'_k, c'_{k+1}, \dots, c'_{m-1}$, is calculated for c'_m that satisfy the formula $c'_m = \sum_{i=k}^{m} c_i - \sum_{i=k}^{m-1} c - i'$. Here public authentication tokens are $c_1, \dots, c_{k-1}, c'_k, \dots, c'_m$ and $\{r_iP\}_{i \in [1,m]}$ are available as public parameters.

(e) **Group authentication:** Each member verifies the below relation for the authentication result

$$\Big(\sum_{i=1}^{k-1} c_i \sum_{i=k}^{m} c'_i P = \sum_{i=1}^{m} r_i P + TQ$$

–**Lemma 1:** If an I_A is limited to making a maximum of Q attempts and comparing between two operating modes with a non-negligible probability ε, then it is possible to construct an algorithm that has the potential to solve the elliptic curve with $\varepsilon' \ge \varepsilon \frac{q}{Q}$. Now prove that if there is an I_A who can differentiate between the two operating schemes, then he can compare between $c_k, \dots, c_m$ and $c'_k, \dots, c'_m$ using $c_i = s_i H_i T + r_i$, where $i = k, \dots, m$. Now, we need to show that the probability of distinguishing the two operating scenarios equals the probability that I_A can get s_i and r_i.

- Based on the Lagrange interpolation theorem, the reconstruction of private keys for other group members can only be achieved if the number of available private keys is greater or equal to the threshold k. The adversary, I_A, is aware of maximum $k-1$ private keys, denoted as $s_1, s_2, \dots, s_{k-1}$, and to proceed, they would need to estimate at least one point on the polynomial. Since point values in Z_q are randomly distributed, the probability of I_A correctly estimating a point is approximately $\frac{1}{q}$. Consequently, the probability of I_A obtaining s_i is at most $\frac{1}{q}$.
- Assume for I_A the probability of solving DLP on elliptic curves ε' is given by (P, r_iP) for the group G_1, now I_A able to finding value r_i, is ε'.
- In summary, probability that I_A can get s_i and r_i is $\varepsilon \le \varepsilon' \frac{Q}{q}$, where Q is number of attempts by I_A. So, the probability $\varepsilon \le \varepsilon' \frac{Q}{q}$ that the two operating scenarios can be distinguished. It can be deduced that the probability $\varepsilon \ge \varepsilon' \frac{Q}{q}$ of I_A that can solve DLP on elliptic curves where ε with a non-negligible probability opposes the assumption that the DLP is hard on elliptic curves.
- The hardness assumptions is contradictory, so I_A can not distinguish between $Real_R(\lambda, Sys_{param})$ and $SIM_S(\lambda, Sys_{param})$. We get difference less than $\varepsilon(\lambda)$ for $Prob[ViewI_A(Real_R(\lambda, Sys_{param}))]$ and $Prob[ViewI_A(SIM_S(\lambda, Sys_{param}))]$. Therefore, the PGASH scheme satisfies confidentiality as defined in the security requirements.
- **The PGASH scheme satisfies non-forgery:** The proof assumes that the X event refers to Ex_A. It can be predicted from the public parameters that s and $r_i, i = 1, \dots, m$, the F event means Ex_A successfully impersonated a group member $M_u, u \ne [1, m]$ and it is undetected, we can get $Prob[F] = Prob[\bar{X}] \cdot Prob[F|\bar{X}] + Prob[X] \cdot Prob$ $[F|X] \le Prob[X] + Prob[F|\bar{X}]$. First, based on the discrete log-hard problem assumption, one can get $Prob[X] < \varepsilon_1(\lambda)$, where, $\varepsilon_1(\lambda)$ represents the negligible function having attribute λ. Then, analyze the probability of $Prob[F|X\bar{]}$. In this case, Ex_A generate a token c_u that satisfy $(\sum_{i=1}^{m} c_i + c_u) = (sT + \sum_{i=1}^{m} r_i + r_u), \{c_i\}_{i \in [1,m]}$. A known set of authentication tokens for other participants, since s, $r_i \in Z_q$ is randomly distributed, so Ex_A guess $(sT + \sum_{i=1}^{m} r_i + r_u)$ is likely to be $1/q$, and Ex_A try the polynomial degree, and get $Prob[F|\bar{X}] = Q/q$, (Here, Q=number of attempts by Ex_A). So as per above discussion, we derive $Prob[F] < \varepsilon_1(\lambda) + Q/q < \varepsilon(\lambda)$, where, $\varepsilon(\lambda)$ is negligible function with respect to attribute λ. Therefore, the security requirements of non-forgery are satisfied by the PGASH scheme.

- **The PGASH scheme satisfies unforgeability** In the proof, we define the event X as the situation where the adversary I_A can predict the value of s using public parameters. The event Y represents the scenario where I_A can obtain certain secret information by interacting with the simulator. Lastly, the event F signifies the successful forging of previously unforgeable information. We deduce the following information by utilizing the restrained group member’s authentication token.

$$\begin{aligned} Prob[F] &= Prob[F|X \vee Y] \cdot Prob[X \vee Y] \\ &\quad + Prob[F|\bar{X} \wedge \bar{Y}] \cdot Prob[\bar{X} \wedge \bar{Y}] \\ &\le Prob[X \vee Y] + Prob[F|\bar{X} \wedge \bar{Y}] \\ &\le Prob[F|\bar{X} \wedge \bar{Y}] + Prob[X] + Prob[Y] \end{aligned}$$

First, based on the DLP problem, assume that we can get $Prob[X] < \varepsilon_1(\lambda)$, where $\varepsilon_1(\lambda)$ is a negligible function of attribute λ. Second, proof of confidentiality claims the scheme satisfies confidentiality and does not reveal any secret data, even after I_A requests for the polynomial degree of the simulator. Therefore, it can be inferred that, $Prob[Y] < \varepsilon_2(\lambda)$, here $\varepsilon_2(\lambda)$ represents a negligible function value with respect to λ. Let's examine $Prob[F|\bar{X} \wedge \bar{Y}]$ in this scenario. To create an authentication token that does not belong to the restrained group, individual I_A must guess the randomly distributed value of $s \in Z_q$. Consequently, the likelihood of I_A correctly guessing s is approximately $1/q$. I_A can make multiple attempts using different polynomials, resulting in $Prob[F|\bar{X} \wedge \bar{Y}] = \frac{Q}{q}$, where Q represents the number of attempts made by I_A. Based on that, we get $Prob[F] = Prob[X] + Prob[Y] + Prob[F|\bar{X} \wedge \bar{Y}] < \varepsilon_1(\lambda) + \varepsilon_2(\lambda) + Qq < \varepsilon(\lambda)$. Therefore, the PGASH scheme satisfies unforgeability as per the security requirements

5.2 Security analysis

In this subsection, we show the security analysis based on attack scenarios such as anti-replay attacks and dishonesty of group managers.

5.2.1 Anti-group manager deception

In PGASH, group members employ a verification process to ensure the validity of their private keys, thus preventing any potential cheating by group managers. Each group member, denoted as M_i (where $i = 1, \ldots, m$), utilizes public values $v_j = e(a_jP, R)$ (where $j = 0, 1, \ldots, k-1$) and verifies the relation $e(s_iP, R) = \prod_{j=0}^{k-1} v_j^{x_i^j}$.

When there is a dynamic change in group membership, such as the addition of M_{n+1} to the group M_i (where $i \in [1, n]$), M_{n+1} verifies the equation $e(s_{n+1}P, R) = \prod_{j=0}^{k-1} v_j^{x_{n+1}^j}$. On the other hand, if M_n withdraws from the group $M = M_1, M_2, \ldots, M_n$, the remaining group members M_i (where $i \in [1, n-1]$) verify the relation $v_1 e(hP, R) = v_1'$. If this relation does not hold, it indicates that the private key is invalid, and there may be a deception by the G_M. In such cases, the group members request a re-issuance of private keys from the G_M. The above discussion demonstrates that the PGASH scheme effectively prevents the group manager's deception.

5.2.2 Anti-replay attack

In the PGASH, the group manager (G_M) launches a group authentication request to the group M. This request includes the session number ϕ, which G_M updates in each cycle. Consequently, the session IDs generated in different cycles denoted as $T = h(x_1|| \ldots ||x_m||\phi)$ are distinct. Even if the same group members participate in two consecutive cycles of group authentication, adversaries cannot bypass the subsequent cycle by replaying messages from a previous cycle.

Let's assume that in the previous cycle, the session number is represented as ϕ_1, and the members participating in group authentication are denoted as $M_1, M_2, \ldots, M_m$. The session ID is calculated as $T_1 = h(x_1|| \ldots ||x_m||\phi_1)$, and the session token is given by $c_i = s_iH_iT_1 + r_i \cdot P$. Now, considering the next cycle with a different session sequence represented as ϕ_2 (where $\phi_2 \neq \phi_1$), the same members $M_1, M_2, \ldots, M_m$ participate in group authentication. The session ID for this cycle is calculated as $T_2 = h(x_1|| \ldots ||x_m||\phi_2)$, and the session token is given by $c_i' = s_iH_iT_2 + r_i' \cdot P$. Importantly, since $T_1 \neq T_2$, the attacker is unable to bypass the replay protection mechanism for the group members (M_i). The session token of the previous cycle passes the group authentication of the next cycle. As a result, the PGASH scheme effectively mitigates replay attacks and enhances the security of GAS.

5.3 Performance analysis

In this subsection, the performance of the PGASH is evaluated based on computational overhead, communication cost, and security. A comparison is made with existing schemes found in the literature [18–20, 35]. Computational overhead is considered a crucial performance metric for group authentication schemes [36]. Assume $T_{(mul,q)}\forall Z_q$, $T_{(inv,q)}\forall Z_q$, T_{EA} and T_{EM} are the time of the last multiplication, inverse, ECC point addition, and ECC point multiplication respectively. Moreover, T_{pair} and T_{htp} are the time of a bi-linear pairing and one hash mapping to a point operation, respectively. We ignore the time of hash function, addition, and subtraction in the performance evaluation because these operations are negligible compared with operations such as point multiplication. According to the literature [18] the computational cost of$T_{EM} \cong 1189T_{(mul,q)}$, $T_{EA} \cong 4.92T_{(mul,q)}$, $T_{pair} \cong 5356T_{(mul,q)}$ and $T_{(inv,q)} \cong 240T_{(mul,q)}$. Let the '$m$' be the number of members participating in PGASH and other schemes presented in [18–20, 35, 37]. The computational cost of group members in all schemes is as follows:

- In the GAS proposed by Xia et al. [19], the generation of authentication tokens for each group member incurs a cost of $(2m-2)T_{(mul,q)} + T_{(inv,q)} + 3T_{EM} + (m-1)T_{EA}$. Additionally, the verification of group membership requires a cost of $(m-1)T_{EA} + 2T_{pair}$ per group member. Hence, the total cost for each group member is given by $(2m-2)(T_{(mul,q)} + T_{EA}) + T_{(inv,q)} + 3T_{EM} + 2T_{pair}$. In order to provide a comprehensive assessment of the computational cost in the scheme [19], it is determined that the cost of a single group authentication for each group member is $(12m + 14507)T_{(mul,q)}$.

- In the GAS proposed by Chien [18], the computational cost for each group member was calculated. Upon analysis, it was determined that the algorithm requires $(7m + 12134)T_{(mul,q)}$ operations to generate an authentication token and verify group membership. Additionally, during the initial stage of group authentication, there is a requirement to negotiate a point w among the group members. While the author does not provide the specific algorithm for this negotiation, the time allocated for this process is T_a. Therefore, the overall cost of a single group authentication for each group member can be expressed as $(7m + 12134)T_{(mul,q)} + T_a$.
- In the GAS presented by Aydin et al. [20], the generation of authentication tokens for each group member requires $T_{(inv,q)} + (2m - 3)T_{(mul,q)} + 2T_{EM}$ operations. Additionally, the verification of group membership incurs a cost of $(m - 1)T_{EA}$ per group member. Therefore, the total cost for each group member is calculated as $T_{(inv,q)} + (2m - 3)T_{(mul,q)} + 2T_{EM} + (m - 1)T_{EA}$. To further assess the computational cost of the scheme [20], it is determined that the cost of a single group authentication for each group member is $(7m + 2610)T_{(mul,q)}$.
- In the GAS introduced by Mahmood et al. [37], the generation of authentication tokens for each group member requires $T_{EM} + T_{(mul,q)}$ operations. Moreover, the verification of group membership incurs a cost of $3T_{EM} + 2T_{EA}$ per group member. Since this scheme involves one-to-one authentication among group members, each member is associated with the remaining $m - 1$ group members, resulting in a total cost of $(m - 1)(4T_{EM} + 2T_{EA} + T_{(mul,q)})$ for each group member. To further assess the computational cost of the scheme [37], it is determined that the cost of a single group authentication per group member is $(4767(m - 1))T_{(mul,q)}$.
- In the GAS proposed by Wang et al. [35], the generation of authentication tokens for each group member requires $T_{EA} + T_{htp} + 2T_{EM}$ operations. Additionally, the verification of group membership incurs a cost of $3(m - 1)T_{EA} + 3T_{pair} + T_{(mul,q)} + (m - 1)T_{htp}$ per group member. Consequently, the total cost for each group member is calculated as $(3m - 2)T_{EA} + 2T_{EM} + 3T_{pair} + T_{(mul,q)} + mT_{htp}$. In order to further assess the computational cost of the scheme [35], it is determined that the cost of a single group authentication for each group member is $(15m + 18437)T_{(mul,q)} + mT_{htp}$.
- In the proposed PGASH scheme, each group member incurs a cost of $(2m - 1)T_{(mul,q)} + T_{(inv,q)} + T_{EM}$ to generate an authentication token represented as $c_i = s_i H_i T + r_i$ and $r_i P$. Additionally, the verification of group membership requires a cost of $2T_{EM} + mT_{EA}$ per group member. Consequently, the total cost for each group member is calculated as $(2m - 1)T_{(mul,q)} + T_{(inv,q)} + 3T_{EM} + mT_{EA}$. In order to further assess the computational cost of the PGASH scheme, it is determined that the cost of a single group authentication for each group member is $(7m + 3806)T_{(mul,q)}$.

Table 3 shows the comparison of the computational cost and security of group members between our scheme PGASH and the literature of [18–20, 35] schemes. From the table 3, it is evident that the PGASH scheme outperforms other schemes regarding private key verification and security. It effectively handles dynamic membership changes, offering resistance against replay and forgery attacks. Additionally, the PGASH scheme demonstrates lower computational overhead during the group authentication phase than other schemes.

Figure 6 illustrates a comparison between the PGASH scheme and existing schemes [18–20, 35] for single-group authentication of each group member. The graph depicts the number of authenticated entities within the group and the associated cost for group members. It is important to highlight that the scheme presented in [18] does not provide a precise method for negotiating a shared secret value. Consequently, the actual duration of execution may surpass the estimated time. The computational time is similar for both [20] and PGASH schemes. As per the computation cost with different member nodes, the cost is less compared to [18, 19, 35], which shows the scheme is achieving scalability in terms of computation time. However, the scheme presented in [20] is vulnerable to replay and forgery attacks, compromising its security.

The communication cost of an algorithm refers to the amount of data that needs to be transmitted or exchanged between different entities in the system. In the proposed

Table 3 Parameters of PGASH scheme in comparison to related schemes

	PKV	DJLG	FA	RA	Computational cost (for each group member for a single group authentication)	Communication cost: GM to each group member (Approximate Bytes)
Our Scheme: PGASH	×	×	×	×	$(7m + 3806)T_{(mul,q)}$	272
Chien [18]	✓	✓	✓	×	$(7m + 12134)T_{(mul,q)} + T_a$	309
Xia et al. [19]	✓	✓	×	×	$(12m + 14507)T_{(mul,q)}$	285
Aydin et al. [20]	✓	✓	✓	✓	$(7m + 2610)T_{(mul,q)}$	258
Wang et al. [35]	✓	×	×	×	$(15m + 18437)T_{(mul,q)} + mT_{htp}$	298

PKV: Private key verification, DJLG: Dynamic join and leave of group members, FA: resist forgery attack, RA: resist replay attack

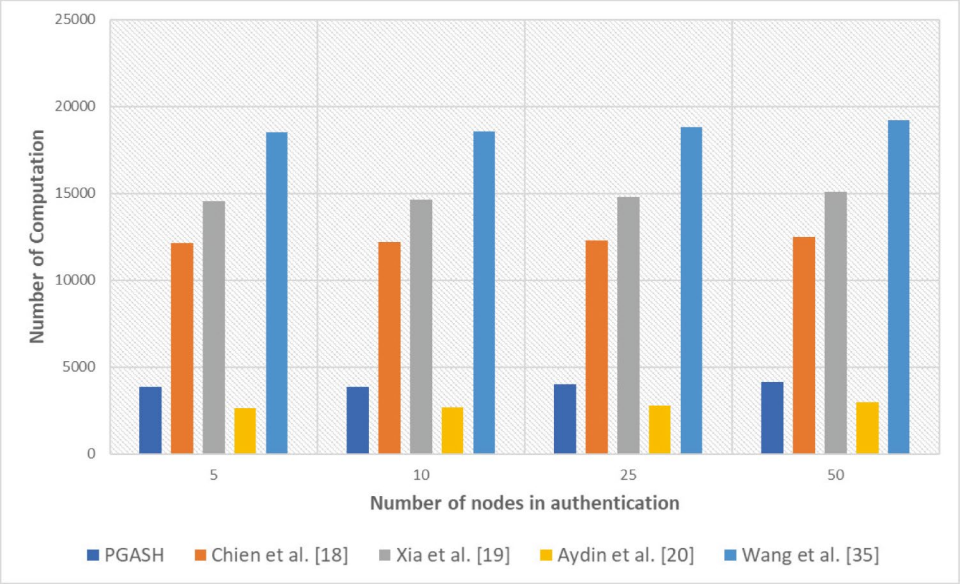


Fig. 6 Comparison of computations cost with the number of group members

scheme, communication cost can be analyzed in terms of the messages exchanged between the group manager (GM) and group members, as well as the data shared among group members during various stages of the protocol. GM initializes the system parameters and generates public parameters. The communication cost here involves transmitting the public parameters to the group members. The size of the public parameters will depend on the size of the groups and the cryptographic parameters. In the next step, GM distributes private keys to individual group members over a secret channel. The communication cost involves the transmission of private keys to each member. The cryptographic parameters determine the size of each private key. After receiving the private keys, group members perform verification by checking the validity of the private keys. The communication cost here involves the possible exchange of messages between group members and the GM in case the verification fails. Further, GM initiates the authentication token generation process by sending a message to the group. Group members respond with their computed authentication tokens. The communication cost involves the transmission of messages between GM and group members. Group members exchange authentication tokens for group authentication. The communication cost involves the transmission of authentication tokens between group members. The size of each token and the number of members affect the communication cost.

The communication cost in initialization and sending parameters to members is 828 bits. In private key generation, for a single member, the communication cost is 288 bits. The following

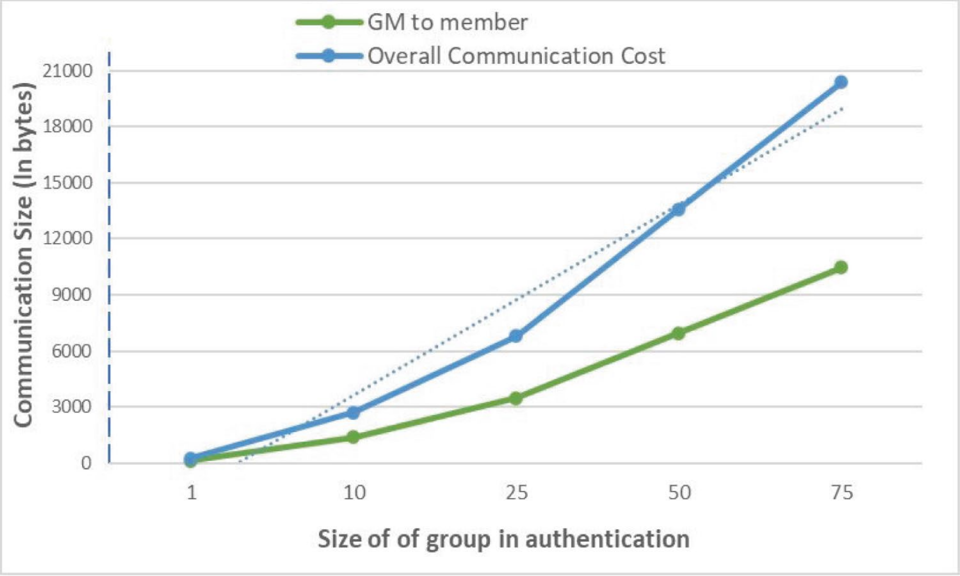


Fig. 7 Comparison of communication cost with the number of group members

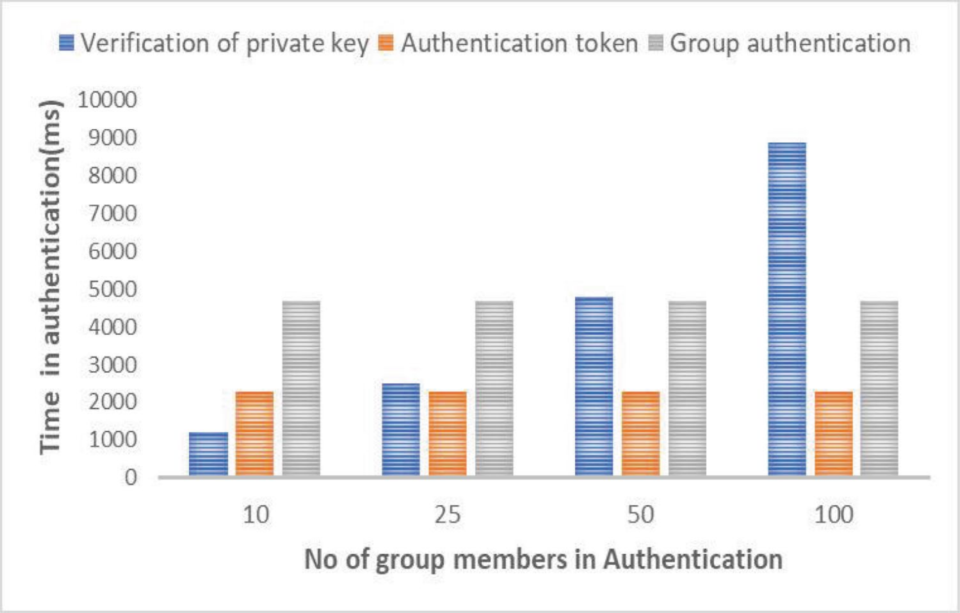


Fig. 8 Total time spent by group member in authentication

verification process takes place among group members and takes 160 bits if verification fails. During authentication token generation, including failing verification once exchange, a total of 1056 bits for a single group member. Finally, group authentication may send a 1-bit message of whether authentication is successful or not. The overall communication cost of the proposed scheme is approximately 2174 bits. This cost includes the message exchanges between group members during the verification and authentication token phases. The comparison of communication cost considering single member and group manager is presented Table 3 that shows our scheme provides less communication cost. However, the scheme in [20] shows better communication cost than our scheme but does not consider the security aspects covered in our scheme. Further, the cost comparison between the group manager and group member does not include the communication among group members and results in a communication cost of 1116 bits. Figure 7 shows the communication cost with respect to varying sizes of group members, which grow linearly with respect to participating members. Figure 8 describes the total time spent by all group members in the proposed scheme under different group authentication numbers and different stages.

The proposed scheme appears to be scalable due to several characteristics that support efficient operation as the size of the group increases. Our scheme supports asynchronous communication, allowing group members to proceed with their tasks independently within designated time limits. This asynchronous nature contributes to scalability by reducing waiting times. Group members can independently verify private keys or generate authentication tokens, allowing for efficient parallel processing. Our scheme appears to handle dynamic changes in group size well. Group members can join or leave, and the group authentication process adapts to these changes without requiring a complete reconfiguration. Private key distribution is performed over a secret channel/link. While the specific details of the secret channel/link are not provided, this mechanism can enhance the security and efficiency of key distribution. The verification process involves a bilinear pairing operation $(e(s_i.P, R))$ and a polynomial evaluation $\prod_{j=0}^{k-1} v_j^{x_i^j}$. These operations can be efficiently computed, and the scheme structure suggests that the verification process can scale with the group size. However, using optimized cryptographic operations and efficient hash functions during different stages may add value to the scalability of the overall system and can be addressed in the future by researchers.

6 Conclusion

IoHT-oriented provable group authentication is an effective solution to overcome concurrent authentication among multiple devices in a group communication scenario. We presented a PGASH scheme for a healthcare scenario in which the private key distributed by the group manager is verified and provable, preventing the fraudulent behavior of the group manager and supporting the dynamic joining and leaving of group members. The proposed scheme ensures confidentiality and correctness while providing resistance against malicious attacks, including impersonation, replay, and forgery attacks. In contrast to prevailing group authentication schemes based on the Internet of Things (IoT), the proposed scheme showcases enhanced security measures and reduced computational overhead. It is particularly well-suited for the Internet of Health Things (IoHT) scenarios, with numerous authentication requests. The scheme effectively addresses the requirements of provable group authentication for resource-limited nodes engaged in group communication.

Author contributions Chandan Trivedi wrote the main manuscript and carried out a literature survey, design scheme, mathematical proof, and security analysis. Keyur Parmar and Udai Pratap Rao have verified and edited mathematical proof, proposed scheme and performance analysis, and reviewed the paper.

Funding No funds, grants, or other support was received.

Data availibility statement Not Applicable.

Declarations

Ethics approval Yes.

Consent to participate Yes.

Consent to publication Yes.

Competing interests The authors declare no competing interests.

References

1. Rouhani S, Butterworth L, Simmons AD, Humphery DG, Deters R (2018) Medichain tm: a secure decentralized medical data asset management system. In: 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp 1533–1538. IEEE
2. Darshan K, Anandakumar K (2015) A comprehensive review on usage of internet of things (iot) in healthcare system. In: 2015 International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT), pp 132–136. IEEE
3. Trivedi C, Rao UP (2023) Secrecy aware key management scheme for internet of healthcare things. J Supercomput pp 1–31
4. Sengupta J, Ruj S, Bit SD (2020) A comprehensive survey on attacks, security issues and blockchain solutions for iot and iiot. J Netw Comput App 149
5. Yue X, Wang H, Jin D, Li M, Jiang W (2016) Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control. J Med Syst 40(10):218
6. Feroz Khan AB, Anandharaj G (2021) Ahkm: An improved class of hash based key management mechanism with combined solution for single hop and multi hop nodes in iot. Egypt Inform J 22(2):119–124. https://doi.org/10.1016/j.eij.2020.05.004
7. Gautam A, Kumar R (2021) A comprehensive study on key management, authentication and trust management techniques in wireless sensor networks. SN Appl Sci 3. https://doi.org/10.1007/s42452-020-04089-9
8. Lara E, Aguilar L, Sanchez MA, García JA (2020) Lightweight authentication protocol for m2m communications of resource-constrained devices in industrial internet of things. Sensors 20(2). https://doi.org/10.3390/s20020501
9. Fouda MM, Fadlullah ZM, Kato N, Lu R, Shen XS (2011) A lightweight message authentication scheme for smart grid communications. IEEE Trans Smart Grid 2(4):675–685
10. Shun Z, Hongli F, Hong Z, Miaomiao T (2018) Efficient and revocable certificateless remote anonymous authentication protocol in wireless body area network. J Commun 39(4):100–111
11. Fang W-D, Zhang W-X, Yang Y, Zhang C-L, Chen W (2018) Btheuap: Biometric-based three-factor user authentication protocol for wireless sensor network. Acta Electon Sin 46(3):702
12. Zhang W-F, Lei L-T, Wang X-M, Wang Y (2020) Secure and efficient authentication and key agreement protocol using certificateless aggregate signature for cloud service oriented vanet. Acta Electon Sin 48(9):1814
13. Li T, Liu Y (2021) A double puf-based rfid authentication protocol. J Comput Res Dev 58(8):1801–1810
14. Bang AO, Rao UP, Visconti A, Brighente A, Conti M (2022) An iot inventory before deployment: A survey on iot protocols, communication technologies, vulnerabilities, attacks, and future research directions. Comput Sec 102914
15. Shamir A (1979) How to share a secret. Commun ACM 22(11):612–613
16. Harn L (2012) Group authentication. IEEE Trans Comput 62(9):1893–1898
17. Ahmadian Z, Jamshidpour S (2017) Linear subspace cryptanalysis of harn's secret sharing-based group authentication scheme. IEEE Trans Inf Forensics Secur 13(2):502–510
18. Chien H-Y (2017) Group authentication with multiple trials and multiple authentications. Secur Commun Netw 2017
19. Xia Z, Liu Y, Hsu C-F, Chang C-C (2020) Cryptanalysis and improvement of a group authentication scheme with multiple trials and multiple authentications. Secur Commun Netw 2020
20. Aydin Y, Kurt GK, Ozdemir E, Yanikomeroglu H (2020) A flexible and lightweight group authentication scheme. IEEE Internet Things J 7(10):10277–10287
21. Park Y, Park Y (2017) A selective group authentication scheme for iot-based medical information system. J Med Syst 41(4):1–8. https://doi.org/10.1007/s10916-017-0692-9
22. Lee D-H, Lee I-Y (2018) Dynamic group authentication and key exchange scheme based on threshold secret sharing for iot smart metering environments. Sensors 18(10). https://doi.org/10.3390/s18103534
23. Wang A, Shen J, Yan L, Ren Y, Liu Q (2018) A practical group authentication scheme for smart devices in iot. EAI Endorsed Trans Internet Things 4(15). https://doi.org/10.4108/eai.5-3-2019.156719
24. Tan H, Chung I (2019) Secure authentication and group key distribution scheme for wbans based on smartphone ecg sensor. IEEE Access 7:151459–151474. https://doi.org/10.1109/ACCESS.2019.2948207
25. Khatoon S, Rahman SMM, Alrubaian M, Alamri A (2019) Privacy-preserved, provable secure, mutually authenticated key agreement protocol for healthcare in a smart city environment. IEEE Access 7:47962–47971. https://doi.org/10.1109/ACCESS.2019.2909556
26. Xiong H, Qin Z (2015) Revocable and scalable certificateless remote authentication protocol with anonymity for wireless body area networks. IEEE Trans Inf Forensics Secur 10(7):1442–1455
27. Trnka M, Cerny T, Stickney N (2018) Survey of authentication and authorization for the internet of things. Secur Commun Netw 2018
28. Forouzan BA (2007) Cryptography and Network Security. McGraw-Hill, Inc
29. Neshenko N, Bou-Harb E, Crichigno J, Kaddoum G, Ghani N (2019) Demystifying iot security: an exhaustive survey on iot vulnerabilities and a first empirical look on internet-scale iot exploitations. IEEE Commun Surv Tutorials
30. Kabra A, Kumar S, Kasbekar GS (2020) Efficient, flexible and secure group key management protocol for dynamic iot settings. arXiv preprint arXiv:2008.06890
31. Hassan WH et al (2019) Current research on internet of things (iot) security: A survey. Comput Netw 148:283–294

32. Ahanger TA, Aljumah A (2018) Internet of things: A comprehensive study of security issues and defense mechanisms. IEEE Access 7:11020–11028
33. Zhang L, Zhang F, Huang X (2009) A secure and effcient certificateless signature scheme using bilinear pairing. Chin J Electron 18(1):145–148
34. Koblitz N (1987) Elliptic curve cryptosystems. Math Comput 48(177):203–209
35. Wang F, Chang C-C, Chou Y-C (2015) Group authentication and group key distribution for ad hoc networks. Int J Netw Secur 17(2):199–207
36. Choksy P, Chaurasia A, Rao UP, Kumar S (2023) Attribute based access control (abac) scheme with a fully flexible delegation mechanism for iot healthcare. Peer-to-Peer Netw App pp 1–23
37. Mahmood K, Chaudhry SA, Naqvi H, Kumari S, Li X, Sangaiah AK (2018) An elliptic curve cryptography based lightweight authentication scheme for smart grid communication. Futur Gener Comput Syst 81:557–565

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Chandan Trivedi is working as an Assistant Professor in Computer Science and Engineering Department. Mr. Trivedi has more than 7 years of teaching experience. He received his BTech degree in Computer Science Engineering from Rajasthan Technical University, Kota and MTech in Computer Engineering from Sardar Vallabhbhai National Institute of Technology (SVNIT), Surat. He received Gold Medal for securing the highest CGPA among all graduating 2013 to 2015 batch of MTech students of Computer Engineering Branch at SVNIT. He has worked in Image processing domain with Space Applications Center, ISRO, Ahmedabad, India as an Intern during the tenure of MTech dissertation. He is also a PhD scholar at SVNIT, Surat under the supervision of Dr. Keyur Parmat and Dr. Udai Pratap Rao. His area of interest in research are Security and Privacy in IoT, Blockchain Technology, Image Processing, Algorithms, Computer Networks and Data Communications. He has publications of research articles in journals like Security and Privacy (Wiley), Journal of Supercomputing (Springer), and IEEE Access. He has edited one book, titled as 'Blockchain for Information Security and Privacy' published by CRC Press, Taylor and Francis Group.

Dr. Keyur Parmar is an assistant professor at Computer Science and Engineering Department of S. V. National Institute of Technology (NIT), Surat, India (an Institute of National Importance). His research interests broadly include Information Security and Blockchain Technology. He is also interested in other research areas where there is a need for security and/or decentralization such as Security in Wireless Sensor Networks (WSNs) or Security in Internet of Things (IoTs). A number of reputed national and international journals and conferences have published his research articles.

Udai Pratap Rao (Senior Member, IEEE) received a PhD in Computer Engineering from Sardar Vallabhbhai National Institute of Technology, Surat (Gujarat), India, in 2014. He has been working as Associate Professor in the Department of Computer Science and Engineering at National Institute of Technology Patna (Bihar) India, since Nov 2022. Prior to joining National Institute of Technology Patna, he worked as Assistant Professor at Sardar Vallabhbhai National Institute of Technology, Surat (Gujarat) India, in the Department of Computer Science and Engineering for more than 15 Years. His research interests include Information Security & Privacy, Privacy in Location-Based Service, Big Data Privacy, Security and Trust Management in Online Social Networks (OSNs), Security and Privacy in Internet of Things (IoT), and Distributed Computing. He has published about 100 papers extensively in journals, book chapters, and refereed conference proceedings. He has supervised 05 PhD thesis in the fields of data privacy, IoT security, and Security and Trust Management in OSN. He is currently the PI of the research project entitled "Design and Implementation of Secure Service and Attribute-based Authorization Model in Dynamic and Constrained-specific IoT Environment" funded by IHUB NTIHAC Foundation, IITK, under the aegis of the National Mission on Interdisciplinary Cyber-Physical System (NM-ICPS), DST, Government of India. He was the Principal Investigator (PI) of the Micro Research project entitled "Investigating Light-Weight Cryptography Algorithms and Its Application to Various IoT Devices" funded by TEQIP-III from July 2019 to Jan 2021. He also acted as the Chief Investigator of the "Information Security Education and Awareness Project Phase II" project from July 2018 to Dec 2019, funded by the Ministry of Electronics and Information Technology (MeitY) Govt. of India. He has edited three books, i) Blockchain for Information Security and Privacy published by CRC Press, Taylor and Francis Group & ii) Security, Privacy and Data Analytics published in year 2022 and 2023 by Springer in their Lecture Notes in Electrical Engineering book series (LNEE). He organised International Conferences on Security, Privacy, and Data Analytics (ISPDA 2021 & ISPDA 2022). He serves as a reviewer of many peer-reviewed journals.