IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 9 June 2024, accepted 9 July 2024, date of publication 15 July 2024, date of current version 24 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3428547

RESEARCH ARTICLE

Strongly Secure Identity-Based Authenticated Key Agreement Protocol With Identity Concealment for Secure Communication in 5G Network

HUANHUAN LIAN[1], BURONG KANG[1], AND LIBAO YANG[2]
[1]Research Institute of China Telecom Corporation Ltd., Shanghai 200122, China
[2]China Telecom Corporation Ltd., Beijing 100000, China

Corresponding author: Huanhuan Lian (lianhh@chinatelecom.cn)

ABSTRACT With the rocketing progress of the fifth generation (5G) mobile communication technology, identity-based authenticated key agreement (ID-AKA) protocol performs an increasingly significant part in secure communication. The majority of current efficient and secure ID-AKA protocols need to transmit each participant's identity and public key information in the clear. Moreover, the long-term secret keys of participants are fully handled by the key generate center, which may give rise to new security concerns. To protect the privacy of user's identity and ensure the security of the private keys, we propose a strongly secure identity-based authenticated key agreement scheme with identity concealment for 5G environment. The proposed scheme provides the property of non-full key escrow and eliminates the need of pairing operations. Furthermore, we show a modified security model for our protocol, and demonstrate the security analysis based on the hardness assumptions of the gap Diffie-Hellman problem and authentication encryption security. Finally, with the help of experiments and performance analysis, the detailed comparative results show that our scheme makes improvements in both efficiency and security while compared with recently proposed ID-AKA schemes.

INDEX TERMS Identity-based cryptography, key agreement, identity privacy, non-full key escrow, 5G network.

I. INTRODUCTION

Recent years have witnessed a spurt of progress in Internet technology, the Internet of Things (IoT) is extensively developed in numerous areas [1], [2]. It is estimated that 25 billion IoT devices will be connected [3], which provides enormous convenience for modern society. Concurrently, the advancement of fifth generation (5G) mobile communication technology has progressively become the main driving force for the widespread of IoT. In comparison to the 4G LTE networks, the 5G technology standard for cellular networks offers fast speed, large capacity, very low latency, and an noticeable improvement in consumers' perceived quality of service (QoS). A 5G system is shown in Figure 1. With the advent of 5G technology, the need for secure and reliable communication has become paramount because of the increasing number of linked devices and the growing reliance on wireless networks. Since the information is transferred over a public network in communication of 5G, it is a vital issue to guarantee communication security and users' privacy.

Authentication key agreement (AKA) offering authentication, data integrity and confidentiality for communication, is essential to contemporary cryptography and acts as a link between symmetric and public key cryptography. It also forms the basis of the network security protocol. Public key infrastructure (PKI)-based traditional authenticated key agreement requires complicated certificate and key management. Through the use of identity-based authenticated key agreement (ID-AKA) protocols, two parties can authenticate each other and agree on a shared key via public channels based on their identities. Shamir's pioneered a new concept of identity-based cryptography (IBC) [4], which simplifies the complicated certificate management and gets rid of the

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Huang.

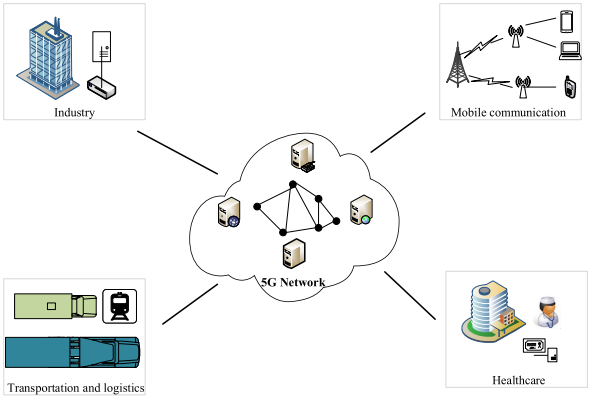

FIGURE 1. Application of 5G system.

PKI building. Boneh and Franklin [5] realized Shamir's idea to introduce a useful and simple identity-based encryption (IBE) scheme, which is under weil pairing. Up to 2002, Smart [6] provided the first ID-AKA scheme, assuming bilinear pairings, based on the IBE technique given in [5]. It was constructed by combining Boneh-Franklin's idea with Joux's Diffie-Hellman (DH) protocol [7]. Afterwards, a number of provably secure ID-AKA protocols were put out [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19].

In the context of 5G communication, ensuring the privacy of user identities is of paramount importance, as emphasized by the 3rd Generation Partnership Project (3GPP) in its determination of fundamental requirements related to user privacy. Therefore, the confidentiality of user identities is crucial in the realm of secure 5G communication. But the identities of users in the aforementioned protocols are transmitted in public network. They fail to address the crucial aspect of user identity privacy. In this paper, we pay attention to building a secure ID-AKA protocol which satisfies identity privacy. In this line, an identity-concealed ID-AKA protocol introduced by Lian et al. at ESORICS 2021 [20] may perform the current state of the art *in security*. For this protocol, it remarkably considers forward identity privacy for all participants. In comparison to the ID-eCK model, the security model presented in this protocol [20] is more stronger. However, pairing operations needed to be executed by each party in key exchange phase. As the pairing operation takes a long time, the efficiency of the ID-AKA protocol is significantly diminished.

In the traditional ID-AKA protocols, the static secret keys are fully computed by the key generator center (KGC). If the AKA scheme adopts full key escrow, it would entail that the trusted third party possesses knowledge of all long-term private keys held by the entities, thereby giving rise to potential security challenges [21]. However, the most of existing ID-AKA protocols suffer from the problem of full key escrow.

Taking the above limitations into consideration, the following question comes up naturally: could we put forth a simple and practical ID-AKA protocol that simultaneously meets: (1) identity concealment, (2)non-full key escrow, and (3) no pairing operation performed by each participant?

Our contributions. We investigate the above interesting question and propose a strongly secure and lightweight ID-based authenticated key agreement protocol. The following contributions are described in this work:

- We design a privacy-enhanced ID-AKA scheme, which supports identity privacy protection. More specifically, the transmitted record does not reveal personal identity information. And the forward identity privacy for all participants is satisfied in the scheme. That is, though participants' static secret keys are revealed, their identity privacy is still protected.
- The proposed scheme is suitable and efficient for devices with limited resource in 5G network. Each user establishes a same session key requiring no pairing operations. In addition, there is no extra operations are needed to generate traditional master public key.
- In order to mitigate the security risks related to full key escrow, the participant's static secret keys are divided into two parts in the proposed scheme. Specifically, one part is generated by the key generation center, while the other is a secret value chosen by the user. As a result, KGC is unable to calculate final session key without knowledge of the user's secret value.

The security of our ID-AKA protocol is proved by the formal security analysis in the random oracle model, under the assumptions of gap Diffie-Hellman problem and authenticated encryption. Moreover, it has been demonstrated by the informal security analysis that the proposed protocol provides ephemeral-secret leakage security, non-full key escrow, perfect forward security and resilience to key compromise impersonation. Finally, we evaluate the performance of our protocol and show its practical feasibility through experiments.

A. RELATED WORK

We first introduce the security models of AKA protocols, and then the ID-AKA protocols are described. The idea of key agreement was first studied by Diffie and Hellman [22]. Bellare, Rogaway and Pointcheval proposed a model for key exchange [23], referred to as BPR00 model, which captures the property of forward secrecy. Canetti and Krawczyk [24] made some changes and showed a modified model which is named as CK01 model. They used session identifier to identity sessions. But the model does not query the session state oracle for the test session. Later, LaMacchia et al. [25] gave an extended CK model, referred to as eCK model, in which the ephemeral secrets of the test session can be get by the attacker. There are several different security models for ID-AKA which are extended by the security model of PKI-based AKA, such as ID-CK framework [9] and ID-eCK framework [26]. However, the existing definition frameworks are not suitable for ID-AKA with identity privacy.

To strengthen the security an efficiency of ID-AKA, different types of ID-AKA protocols have been put forward

in different domains. Chen et al. [10] designed an ID-AKA based on pairings which uses built-in decisional function to deal with the dependence on any oracle. Hölbl et al. [27] introduced a new practical ID-based AKA scheme which employs a variant of signature scheme. There is no computational model used in the protocol's security analysis. Ni et al. [28] designed a highly secure ID-based key exchange scheme in the escrow form. They declared that their protocol is a provable secure scheme in computational model. Ibrahim et al. [29] showed an ID-AKA protocol, which offers the resiliency security but does not offering the property of identity-concealment.

It is discovered that all the aforementioned ID-AKA protocols are unsuitable for 5G environment due to heavy communication and computational costs. Researchers in [14] proposed two eCK secure pairing-free ID-based key agreement schemes including weak perfect forward secrecy, ephemeral secrets reveal resistance and key-compromise impersonation resilience etc. Blazy and Chevalier [8] generically built a no-interactive key exchange scheme using a specific IBE. The protocol needs a central trusted authority to compute the identity-related long-term private keys. But the construction still relies on the public-key infrastructure and don't provide ID-privacy. Wu et al. [30] presented a leakage-resilient ID-AKA protocol which provides the character of unbounded leakage. Authors of [16] proposed an ID-eCK secure ID-AKA protocol which is constructed relied on the relatively less standard difficult problems. Its security definition fail in realizing perfect forward security. What's more, the scheme doesn't satisfy ID-privacy property and its efficiency need to be further improved. Naor et al. [31] constructed an identity-based symmetric password authenticated key exchange without ID-concealment. They present a compiler from PAKE to iPAKE utilizing ID-AKA and then provide a method of siPAKE from any password-based AKE using bilinear groups with "Hash2Curve". Gupta et al. [32] devised an ID-AKA scheme for IoT environment. The protocol only needs a single communication round, but it takes a certain number of bilinear pairings. Authors of [33] proposed a provably secure ID-AKA without using bilinear pairings. The protocol is secure in the strengthened eCK model. For 5G environment, we should also consider protecting the sensitive identity information. However, these ID-AKA protocols lack of the property of identity privacy protection. Lian et al. [20] presented an identity-based identity-concealed AKA protocol, which is the first ID-AKA with identity concealment. But it needs to perform pairing operations. Additionally, the long-term private keys of the mentioned ID-AKA protocols previously are fully escrowed by KGC.

B. ORGANIZATION

The remainder of this paper is organized as follows. In Section II, we briefly review the preliminary knowledge related to this paper. Section III illustrates our system model and security model. In Section IV, we describe the proposed strongly secure ID-AKA protocol. In Section V, the formal security proof and informal security analysis of the proposed protocol are introduced. The evaluation of performance of our construction can be found in Section VI. Finally, Section VII gives a conclusion for this paper.

II. PRELIMINARIES

Define $\mathbb{G}$ by multiplicative group of q order. Let P be a generator of $\mathbb{G}$.

Definition 1 (Computational Diffie-Hellman (CDH)): The Computational Diffie-Hellman (CDH) problem is to calculate $a \cdot b \cdot P \in \mathbb{G}$, given a tuple of $(P, a \cdot P, b \cdot P) \in \mathbb{G}$, where $a, b \in \mathbb{Z}_q^*$. The probability that any PPT adversary can deal with the CDH problem is negligible.

Definition 2 (Decisional Diffie-Hellman (DDH)): The Decisional Diffie-Hellman (DDH) problem is to judge that if $c = ab \bmod q$ is held or not, given a tuple of $(P, a \cdot P, b \cdot P, c \cdot P) \in \mathbb{G}$, where $a, b, c \in \mathbb{Z}_q^*$. The probability that any PPT adversary can deal with the DDH problem is negligible.

Definition 3 (Gap Diffie-Hellman (GDH)): The gap Diffie-Hellman (GDH) problem is to calculate $a \cdot b \cdot P \in \mathbb{G}$ with the aid of a decisional Diffie-Hellman (DDH) oracle for $\mathbb{G}$ and P, given a tuple of $(P, a \cdot P, b \cdot P) \in \mathbb{G}$. And the DDH oracle outputs 1 if and only if $T = a \cdot b \cdot P$, given any input $(A = a \cdot P, B = b \cdot P, T) \in \mathbb{G}^3$. The probability that any PPT adversary can deal with the GDH problem is negligible.

III. SYSTEM MODEL AND SECURITY MODEL

We present the system model and the security model of our identity-based authenticated key agreement scheme with identity concealment in this section.

A. SYSTEM MODEL

In the proposed ID-AKA protocol for 5G network, the system architecture is composed of user, device and a key generation center. The entity can be any intelligent devices that users utilize to obtain services. It requires authenticating the legal identification of the devices that are involved. Smart devices communicate with users to generate authentication message. The key generation center is a trusted third-party entity. It generates public parameters, and involves parties' static secret key generation.

Figure 2 depicts the system model of the proposed protocol. KGC generates the partial static secret keys for users and smart devices. The user and device are allowed to authenticate each other and agree on a same session key through internet or wireless channels, while the exchanged data cannot reveal the identity information.

B. SECURITY MODEL

The ID-eCK model is designed for ID-based authenticated key agreement protocol by Huang and Cao [26], which allows a more powerful adversary with a high capacity for secrecy exposure. However, session matching in the security definition of ID-eCK model depends on user's identity, which is incompatible with the ID-AKA with identity privacy.

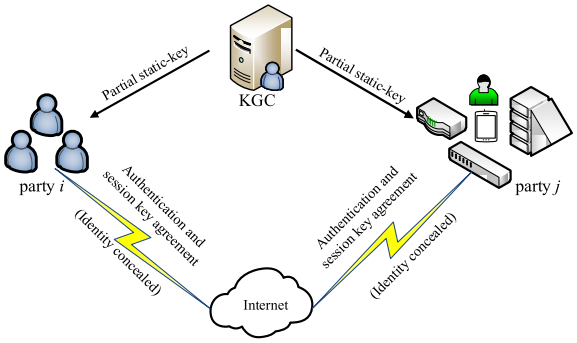

FIGURE 2. System model.

We adopt the security model proposed in [20], with a slight modification in the security definition.

Participants and session: We define a participant by A (resp., B) and the identity of A (resp., B) by ID_A (resp., ID_B). Each participant possesses a pair of ID-based public-private key. Specifically, the public key includes his identity and another value associated partial static secret key. The long-term private key, also called static secret key, includes one value computed by KGC and another value chosen by himself secretly. Each session is assigned a session-identifier *SID* through an incremental counter. The purpose of this counter is solely to different sessions, and it is a tool of the security model.

Adversary abilities: The adversary, $\mathcal{A}$, is considered as probabilistic polynomial-time (PPT) algorithm which rules all participants' communications. Specifically, it can intercept, eavesdrop, forge and inject messages. $\mathcal{A}$ is able to perform a series of queries concurrently, showing as follows, in any order:

- $Send(SID_I, M_I)$: In this oracle, the challenger verifies if the session SID_I exists, if unsuccessful, it ignores; else if M_I is not the final message it returns the subsequent protocol message; otherwise, it runs in accordance with the protocol rule. If the adversary queries this oracle in a not yet started session and M_I being "Start", then the initial protocol message is returned.
- $StaKeyReveal(ID_i)$: In this oracle, it responds with the long-term private keys of entity ID_i, which include one secret key computed by KGC and another key computed by user ID_i.
- $StaKey1Reveal(ID_i)$: In this oracle, it returns the partial long-term private key computed by KGC to the adversary, which is one part of full static secret keys of user (ID_i.
- $StaKey2Reveal(ID_i)$: In this oracle, it returns back another part of full static secret keys of user ID_i, which is generated by user ID_i itself.
- $EpheKeyReveal(SID_I)$: In this oracle, verifies if the session SID_I exists, if unsuccessful, it ignores; else it returns the ephemeral secret key $EheKey_{SID_I}$ associated with the session SID_I to $\mathcal{A}$.
- $MasterKeyReveal()$: In this oracle, the master key of the KGC is brought back to $\mathcal{A}$.
- $Test(ID_{t_0}, ID_{t_1})$: In this oracle, the challenger chooses $b \leftarrow \{0, 1\}$ randomly and defines $ID_t = ID_{t_b}$, and works as receiving the "Start" command. This oracle is only allowed to be queried once.
- $SKReveal(SID_I)$: In this oracle, the session SID_I should be first be checked if it exists, if not, it ignores; else if $SID_I \neq SID_T$, and SID_I has been finished but hasn't been expired, SK_{SID_I} is returned to the adversary; if $SID_I = SID_T$ and $b = 1$, it responds with the true session key SK_{SID_I}, if $SID_I = SID_T$ and $b = 0$, it returns back a random key chosen from $\{0, 1\}^*$.
- $Peer(SID_I)$: In this oracle, if the session SID_I exists, it returns to $\mathcal{A}$ the value stored in $peer_{SID_I}$ where $peer_{SID_I}$ indicates the interacting peer player; otherwise, it ignored.

Note that it is limited that the **Test** oracle can be queried one time by adversary, and the test-session has not been already queried a **SKReveal** oracle.

Definition 4 (Freshness): In this model, define the label of a session by a part of the session transcript. If two sessions have the same session label, we say they are matching. We assume that SID_T is the completed test-session owned by an honest party with identity $ID_t = ID_{t_b}$ and $peer_{SID_T} = ID_k$ for $1 \leq k \leq n$. The peer user with identity ID_k is also honest. Suppose SID'_T be the matching session of SID_T (which may be still on-going) if it exists. We consider the test-session is fresh if none of the following conditions hold.

- *The adversary asked the query $SKReveal(SID_T)$, or made the query $SKReveal(SID'_T)$ if SID'_T exists.*
- *The adversary asked the query $StaKey2Reveal(ID_{t_0})$ or $StaKey2Reveal(ID_{t_1})$, and $EpheKey_{SID_T}$ is revealed through $EpheKeyReveal(SID_T)$ query.*
- *If SID'_T exists, the adversary issued both $StaKeyReveal(ID_k)$ and $EpheKeyReveal(SID'_T)$; else, the adversary issued the query $StaKeyReveal(ID_k)$.*
- *The adversary asked the query $StaKey2Reveal(ID_{t_0})$ or $StaKey2Reveal(ID_{t_1})$, and made the query $MasterKeyReveal()$.*
- *The adversary issued both $StaKey2Reveal(ID_k)$ query and $MasterKeyReveal()$ query.*
- *The adversary asked $Peer(SID'_T)$ if SID'_T exists.*

Note that, for the fresh test-session, there could be the case: the full static-secret keys of ID_{t_0} and ID_{t_1} are revealed via the $StaKeyReveal(ID_t)$ oracle, and the partial static-secret key of ID_k via the $StaKey1Reveal(ID_k)$ oracle and the ephemeral-secret key $EpheKey_{SID'_T}$ are both leaked (if SID'_T exists); or the full static-secret keys of ID_k are exposed via the $StaKeyReveal(ID_k)$ oracle, and the partial static-secret key of ID_t via the $StaKey1Reveal(ID_t)$ oracle and the ephemeral-secret key $EpheKey_{SID_T}$ are both exposed. If SID'_T exists, either revealing the full static keys of ID_{t_0}, ID_{t_1} and ID_k through the $StaKeyReveal(ID_i)$ oracle, or revealing the ephemeral-secret keys of session SID_T and SID'_T via the $EpheKeyReveal(SID)$ oracle, will not reveal the test-session.

Definition 5 (Strong ID-AKA-Security): For any security parameter that is sufficiently large, we say that a two-party ID-based key agreement protocol is strongly secure if it can

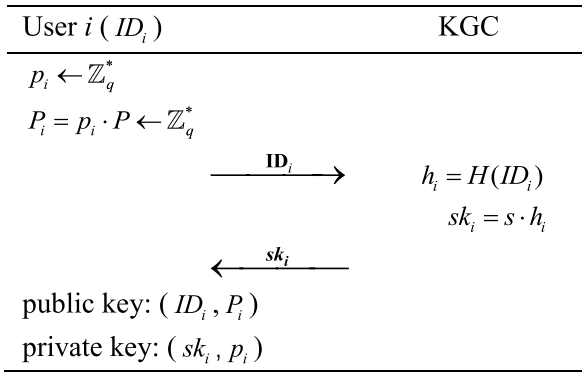

FIGURE 3. Private key extraction of the proposed protocol.

withstand attacks from any PPT adversary $\mathcal{A}$ as specified above and it holds:

- ***Label-security.*** *The probability that any of the following events occurs is negligible.*
  - *More than two sessions have the same session label.*
  - *There are two matching sessions, and we assume they are SID owned by a party with identity ID_i and SID' owned by a party with identity ID_j, such that any of the following events occurs*:
    (1) *Both ID_i and ID_j are initiator or responders.*
    (2) *$SK_{SID} \neq SK_{SID'}$.*
    (3) *$peer_{SID} \neq \perp \wedge peer_{SID} \neq ID_j$, or $peer_{SID'} \neq \perp \wedge peer_{SID'} \neq ID_i$.*
- ***Session-key security with identity-concealment.*** *The adversary asks any order of queries to the above described oracles. If the test-session SID_T of the uncorrupted user ID_t is fresh and completed, then (impersonation security) the probability that the test-session does not have matching session is negligible and (ID-SK indistinguishability) $|Pr[b' = b] - \frac{1}{2}|$ is negligible (or $|Pr[b' = b] - \frac{1}{2}| \leq negl(1^\kappa)$).*

IV. OUR PROPOSED ID-AKA PROTOCOL

In this section, we give the construction of our identity-based authenticated key agreement without pairings, which provides identity concealment. The proposed protocol contains mainly two phases, namely, private key extraction and authenticated key agreement.

Let κ be a secure parameter, $\mathbb{G}$ be a cyclic additive group of order q and P be a generator of $\mathbb{G}$. And the master secret key $msk = s \leftarrow \mathbb{Z}_q^*$ is generated by the key generator center. $SE = (K_{se}, Enc, Dec)$ denotes an authenticated encryption (AE) scheme, where $\mathcal{K} = \{0,1\}^\kappa$ is the key space of K_{se}. Denote $KDF : \{0,1\}^* \rightarrow \{0,1\}^{p(\kappa)}$ as a key derivation function, where $p(\kappa)$ is a polynomial in κ. Assume that $H : \{0,1\}^* \rightarrow \mathbb{G}$ is a cryptographic hash function. We model the hash function and KDF as random oracles.

A. PRIVATE KEY EXTRACTION PHASE

The process of private key generation of participants is given in Figure 3. Denote the identity of user i by ID_i. The users pass their IDs to the KGC, and the KGC produces the partial static secret key for each user.

- The user i randomly picks $p_i \leftarrow \mathbb{Z}_q^*$ as its partial static secret key, and calculates the partial public key $P_i = p_i \cdot P$. Then, it sends the identity ID_i to the KGC.
- The KGC calculates $h_i = H(ID_i)$ and $sk_i = s \cdot h_i$. After that, KGC delivers sk_i to the user i through a safe channel.
- The user's public key and static secret key are respectively set to: (ID_i, P_i), (sk_i, p_i). And the user keeps the static secret key secretly.

The static secret key of a user is comprised of two components: one component is generated by the user itself, and the other is computed by the key generation center. Consequently, the proposed protocol is not key escrowed since the KGC only learns a portion of the static key rather than the full key.

B. AUTHENTICATED KEY AGREEMENT PHASE

The protocol performs the following operations to build a shared session key and realize mutual authentication. Let user A with identity ID_A be a an initiator of a session and user B with identity ID_B be a responder of the session. The high-level overview of our strong ID-AKE protocol is depicted in Figure 4.

- The party A chooses $r_A \leftarrow \mathbb{Z}_q^*$, calculates $x = r_A \cdot p_A$ and $X = x \cdot h_A$. Then X is sent to B by A.
- After receiving X, the party B selects $r_B \leftarrow \mathbb{Z}_q^*$, calculates $y = r_B \cdot p_B$ and $Y = y \cdot h_B$. It verifies whether $X \in \mathbb{G}/1_{\mathbb{G}_1}$ and aborts if not. B computes the primary secret $PS_B = sk_B \cdot r_B \cdot p_B \cdot X + p_B \cdot P_A$, derives keys $(K_1, K_2) \leftarrow KDF(PS_B, X \parallel Y)$. Then the party B generates $C_B = Enc_{K_1}(y, ID_B)$ and delivers (Y, C_B) to A.
- After receiving (Y, C_B), the party A calculates primary secret $PS_A = sk_A \cdot r_A \cdot p_A \cdot Y + p_A \cdot P_B$, generates keys $(K_1, K_2) \leftarrow KDF(PS_A, X \parallel Y)$ and obtains B's identity ID_B and y by decrypting the ciphertext C_B under the secret value K_1. Subsequently, it checks if $y \in Z_q^*$ and $Y = y \cdot (H(ID_B))$, if successful, it calculates $C_A = Enc_{K_1}(x, ID_A)$. Then A sets the session key to be K_2 and delivers C_A to B.
- After receiving C_A, the party B obtains x and ID_A by decrypting the ciphertext C_A. Then B checks if $x \in Z_q^*$ and $X = x \cdot (H(ID_A))$, if successful, the session key is set to be K_2.

Correctness. It suffices to state that the same session key can be obtained by all honest parties in a consistent manner. We have $PS_A = sk_A \cdot r_A \cdot p_A \cdot Y + p_A \cdot P_B = s \cdot H(ID_A) \cdot r_A \cdot p_A \cdot r_B \cdot p_B \cdot H(ID_B) + p_A \cdot p_B \cdot P$ and $PS_B = sk_B \cdot r_B \cdot p_B \cdot X + p_B \cdot P_A = s \cdot H(ID_B) \cdot r_B \cdot p_B \cdot r_A \cdot p_A \cdot H(ID_A) + p_B \cdot p_A \cdot P$. Thus, user A and B both are in possession of the identical session key using the function KDF.

V. SECURITY FOR THE PROPOSED ID-AKA

We give the security analysis to demonstrate that the identity-based authenticated key agreement without pairings is strongly secure.

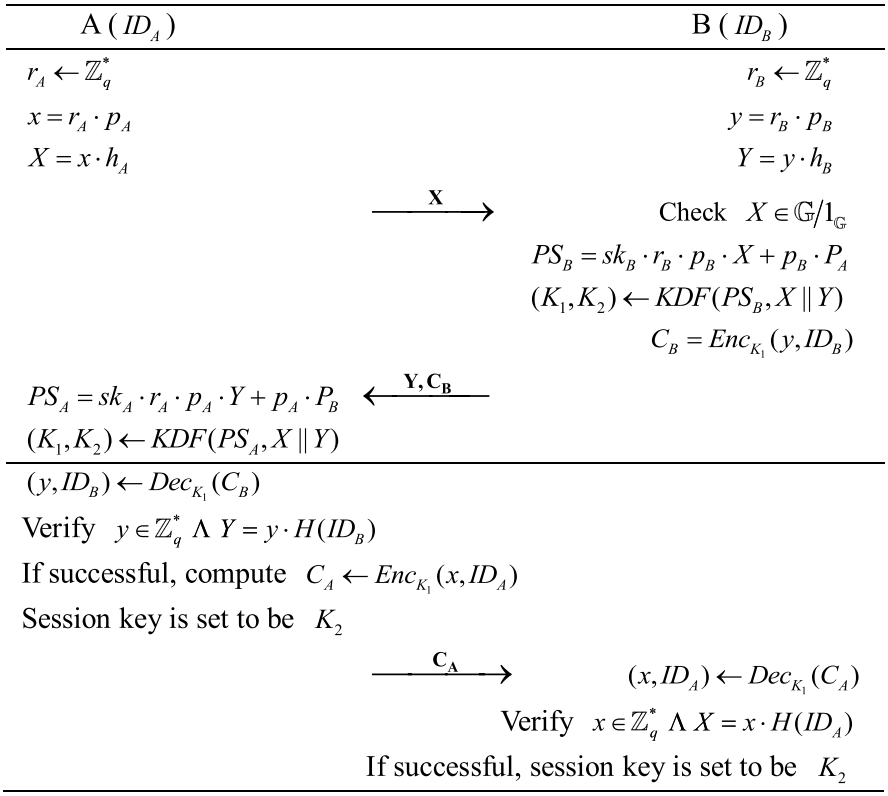

FIGURE 4. Construction of our strong ID-AKA protocol.

A. FORMAL SECURITY PROOF

In the following description of formal security proof, both the hash function H and the KDF are performed as random oracle (RO).

Theorem 1: The proposed scheme ID-AKA in Section IV is strongly secure in the RO model, under the GDH assumption and the AE security.

We need to prove our protocol holds label-security and session-key security with identity privacy defined in Definition 5. We prove the theorem according to Lian's proof process [20]. The first difference is that we replace the Gap-BDH assumption with GDH assumption. The second difference is that the query *MasterKeyReveal* is allowed to the adversary, resulting in a change in the definition and proof of security. In the following, we give a sketch of the proof and the entire proof.

Outline of the proof. For each protocol session, define the session label by "$X||Y$". For the label-security, we must present that (1) the advantage that more than two sessions share the identical session label is negligible; (2) the event that ID_i and ID_j play the same session role or $SK_{SID} = SK_{SID'}$, or $peer_{SID} \neq \perp \wedge peer_{SID} \neq ID_j$, or $peer_{SID'} \neq \perp \wedge peer_{SID'} \neq ID_i$ (SID owned by a party with identity ID_i and SID' owned by a party with identity ID_j) occurs with negligible probability. We prove the label-security in Lemma 1 in Section V-A.1.

For the SK security with identity concealment, we need to present our protocol satisfies impersonation security and ID-SK indistinguishability (in Definition 5). The impersonation security guarantees that the adversary can't impersonate any honest party. This is because that SID_T held at a party with identity ID_t is completed and fresh, and SID_T doesn't have matching session if and only if the adversary could impersonate the honest peer player $ID_k = peer_{SID_T}$. If the impersonation security can be broken by an adversary, then we can construct a simulator to break the GDH assumption under the interaction with this "powerful" adversary. The comprehensive proof of impersonation security is showed in Section V-A.2.1. The ID-SK indistinguishability ensures the privacy of the final exchanged session key and identity information of the participants. If an adversary can break ID-SK indistinguishability, we can construct an algorithm which simulates the protocol to break the GDH assumption. The details of the proof of ID-SK indistinguishability is described in Section V-A.2.2.

1) PROOF OF LABEL-SECURITY

Firstly, we show that the advantage that more than two sessions have the identical session label is negligible. Suppose SID owned by user ID_i and SID' owned by user ID_j are a pair of matching sessions. If both of ID_i and ID_j are initiator or responder, then there exists X and X' such that $X = X'$ (or Y and Y' such that $Y = Y'$), and this events occurs with negligible probability based on the GDH assumption. Thus, ID_i and ID_j plays the same role with negligible probability. The intermediate values $PS_i = sk_i \cdot r_i \cdot p_i \cdot Y + p_i \cdot P_j$ and $PS_j = sk_j \cdot r_j \cdot p_j \cdot X + p_j \cdot P_i$, it is

easy to know that $PS_i = PS_j$. And the key pairs $(K_1, K_2) = KDF(PS_i, X \parallel Y)$ and $(K_1, K_2) = KDF(PS_j, X \parallel Y)$ can be computed by the equal label $X \parallel Y$. Consequently, the two matching sessions possess the same session-key.

Next, we prove that either $peer_{SID} \neq \perp \wedge peer_{SID} \neq ID_j$ or $peer_{SID'} \neq \perp \wedge peer_{SID'} \neq ID_i$ occurs with negligible probability. For our proposed ID-AKA scheme, it may be the case that $peer_{SID} = ID_j$ but $peer_{SID'} = \perp$. In particular, assume that the adversary drops the third-round message that the party ID_i transmitted in SID, and this causes $peer_{SID} = ID_j$ but $peer_{SID'} = \perp$. Let $X = x_i \cdot H(ID_i)$ and $Y = y_j \cdot H(ID_j)$, and consider that $peer_{SID} \neq \perp \wedge peer_{SID} \neq ID_j$, or $peer_{SID'} \neq \perp \wedge peer_{SID'} \neq ID_i$. This means that the adversary can successfully open $Y = y_j \cdot H(ID_j)$ into (ID_k, y_k) for $ID_k \neq ID_j$ in session SID such that $Y = y_j \cdot H(ID_j) = y_k \cdot H(ID_k)$, or open $X = x_i \cdot H(ID_i)$ into (ID_b, x_b) for $ID_b \neq ID_i$ in session SID' such that $X = x_i \cdot H(ID_i) = x_b \cdot H(ID_b)$. According to Lemma 1, we conclude that the advantage of $peer_{SID} \neq \perp \wedge peer_{SID} \neq ID_j$ or $peer_{SID'} \neq \perp \wedge peer_{SID'} \neq ID_i$ is negligible. This finishes the proof of the label-security.

Lemma 1: There isn't a PPT algorithm that generates $\{ID_j, y_j \in \mathbb{Z}_q^*\}$ *and* $\{ID_k, y_k \in \mathbb{Z}_q^*\}$, *where* $\{ID_j, y_j\} \neq \{ID_k, y_k\}$, *with non-negligible probability such that* $y_j \cdot H(ID_j) = y_k \cdot H(ID_k)$, *assuming that* $H : \{0,1\}^* \rightarrow \mathbb{G}$ *is a random oracle.*

Proof: For any pair of different $\{ID_j, y_j\}$ and $\{ID_k, y_j\}$, the probability of $y_j \cdot H(ID_j) = y_k \cdot H(ID_k)$ is that $\Pr[y_j \cdot H(ID_j) = y_k \cdot H(ID_k)] = \frac{1}{q-1} + \frac{1}{(q-1)(q-2)^2}$, assuming that $H : \{0,1\}^* \rightarrow \mathbb{G}$ is a random oracle.

Let E be the event that the adversary outputs a pair of different (ID_j, y_j) and (ID_k, y_k) such that $y_j \cdot H(ID_j) = y_k \cdot H(ID_k)$. Assume the adversary asks the oracle H at most t times, where t is polynomial in $|q|$, then $Pr(E) \leq \frac{t^2}{2(q-1)} + \frac{t^2}{2(q-1)(q-2)^2}$. It can be obviously seen that the probability of the event E occurring is negligible. ■

2) PROOF OF SESSION-KEY SECURITY WITH IDENTITY CONCEALMENT

Let SID_T be the finished fresh test-session owned by the honest party $ID_t = ID_{t_b}$ with $peer_{SID_T} = ID_k$, $1 \leq k \leq n$. Define the session label by "$X \parallel Y$". SID_T's matching session (if it exists) is SID'_T (may be still on-going).

The session-key security with identity concealment is reduced to the GDH assumption and the AE security. Firstly we define two cases to maintain the consistency of the random oracle KDF.

(1) Suppose the test-session holder $ID_t = ID_A$ is an initiator user.

- The input of the simulator $\mathcal{S}$ contains a given value $c \cdot P$, where the master secret key $c \leftarrow \mathbb{Z}_q^*$ is hidden from $\mathcal{S}$. When the adversary queries the long-term private key of ID_A, $\mathcal{S}$ outputs (sk_A, p_A) and answers as follows. If $\mathcal{A}$ has made the hash oracle H query for ID_A, in order to answer the query $H(ID_A)$, the simulator retrieves the value a_A, i.e., $a_A \cdot P = H(ID_A)$; if not, $\mathcal{S}$ chooses a new a_A at random and sets $a_A \cdot P = H(ID_A)$. Then, the simulator computes $sk_A = a_A \cdot c \cdot P$ using a_A and $c \cdot P$ and generates $p_A \leftarrow \mathbb{Z}_q^*$ by itself. $\mathcal{S}$ inputs $X \leftarrow \mathbb{G}_1/1_{\mathbb{G}_1}$, and obtains access to the DDH-oracle $DDH(\cdot,\cdot,\cdot)$. And the responder receives X from $\mathcal{S}$ in the first round of SID_T. When $\mathcal{S}$ receives (Y, C_B) in the second round from responder, then it verifies if $(X \parallel Y, (K_1, K_2))$ has been recorded in the list $\mathcal{L}_{DDH}$: if "yes", $\mathcal{S}$ decrypts C_B using K_1; if "not", $\mathcal{S}$ produces $(K_1, K_2) \leftarrow (0,1)^\kappa \times (0,1)^\kappa$ by itself at random and keeps $(X \parallel Y, (K_1, K_2))$ into the list $\mathcal{L}_{DDH}$. Then, the adversary $\mathcal{A}$ will make RO-query of the type $KDF(PS, X \parallel Y)$, where PS is assumed to be $CDH(X \cdot Y, c \cdot P) + p_A \cdot P_B$. $\mathcal{S}$ computes $Z = CDH(X \cdot Y, c \cdot P) = PS - p_A \cdot P_B$. Then the simulator takes $(sk_A \cdot p_A \cdot Y, r_A \cdot P, Z)$ to ask DDH-oracle: if the oracle outputs "yes", $\mathcal{S}$ responds the previously recorded values (K_1, K_2) to $\mathcal{A}$; otherwise, it returns random values.
- The input of $\mathcal{S}$ includes the master secret key $c \leftarrow \mathbb{Z}_q^*$. When $\mathcal{A}$ asks the private secret key of ID_A, $\mathcal{S}$ computes $sk_A = c \cdot H(ID_A)$, but p_A is unknown to $\mathcal{S}$. If $\mathcal{A}$ has asked H oracle for ID_A, in order to answer the query $H(ID_A)$, the simulator retrieves the value a_A, i.e., $a_A \cdot P = H(ID_A)$; if not, $\mathcal{S}$ chooses a fresh a_A at random and sets $a_A \cdot P = H(ID_A)$. Then the simulator inputs $X \leftarrow \mathbb{G}_1/1_{\mathbb{G}_1}$, and obtains access to the DDH-oracle $DDH(\cdot,\cdot,\cdot)$. Then, the simulator $\mathcal{S}$ does similar to the above situation, but PS is assumed to be $PS = CDH(P_A, P_B) + c \cdot X \cdot Y$. $\mathcal{S}$ can get $CDH(P_A, P_B) = PS - c \cdot X \cdot Y = p_A \cdot p_B \cdot P$, and then computes $Z = CDH(P_A, P_B) \cdot a_A = p_A \cdot p_B \cdot H(ID_A)$. Next, the simulator takes $(X \cdot {r_A}^{-1}, P_B, Z)$ to ask its DDH-oracle, where r_A is generated by the simulator itself because r_A can be exposed to adversary. If the oracle outputs "yes", $\mathcal{S}$ responds the previously recorded values (K_1, K_2) to $\mathcal{A}$; otherwise, it returns random values.

(2) Suppose the test-session holder $ID_t = ID_B$ is a responder user.

- The input of the simulator $\mathcal{S}$ contains a given value $c \cdot P$, where the master secret key $c \leftarrow \mathbb{Z}_q^*$ is hidden from $\mathcal{S}$. When the adversary queries the long-term private key of ID_B, $\mathcal{S}$ outputs (sk_B, p_B) and answers as follows. If $\mathcal{A}$ has made the hash oracle H for ID_B, in order to answer the query $H(ID_B)$, $\mathcal{S}$ retrieves the value a_B, i.e., $a_B \cdot P = H(ID_B)$; if not, $\mathcal{S}$ chooses a new a_B at random and sets $a_B \cdot P = H(ID_B)$. Then, the simulator compute $sk_B = a_B \cdot c \cdot P$ using a_B and $c \cdot P$. Then the simulator inputs $Y \leftarrow \mathbb{G}_1/1_{\mathbb{G}_1}$, and obtains access to the DDH-oracle $DDH(\cdot,\cdot,\cdot)$. $\mathcal{S}$ receives X in the initial round of SID_T, it verifies $X \in \mathbb{G}_1/1_{\mathbb{G}_1}$, and verifies if $(X \parallel Y, (K_1, K_2))$ has been stored in $\mathcal{L}_{DDH}$: if "yes", $\mathcal{S}$ will abort, which happens with at most negligible probability; if "not", $\mathcal{S}$ produces (K_1, K_2) by itself at random and keeps $(X \parallel Y, (K_1, K_2))$ into $\mathcal{L}_{DDH}$. Then, the adversary $\mathcal{A}$ will make the random oracle query of the type $KDF(PS, X \parallel Y)$, where PS is assumed to be $PS = CDH(X \cdot Y, c \cdot P) + p_B \cdot P_A$, and $\mathcal{S}$ computes $Z = CDH(X \cdot Y, c \cdot P) = PS - p_B \cdot P_A$. Then

the simulator $\mathcal{S}$ takes $(sk_B \cdot p_B \cdot X, r_B \cdot P, Z)$ to ask DDH-oracle: if the oracle outputs "yes", $\mathcal{S}$ responds the previously recorded values (K_1, K_2) to $\mathcal{A}$; otherwise, it returns random values.

- The input of $\mathcal{S}$ includes the master secret key $c \leftarrow \mathbb{Z}_q^*$. $\mathcal{A}$ asks the private secret key of ID_B, $\mathcal{S}$ computes $sk_B = c \cdot H(ID_B)$, but p_B is unknown to $\mathcal{S}$. If $\mathcal{A}$ has made the hash oracle H query for ID_B, in order to answer the query $H(ID_B)$, the simulator $\mathcal{S}$ retrieves the value a_B, i.e., $a_B \cdot P = H(ID_B)$; if not, $\mathcal{S}$ chooses a new a_B at random and sets $a_B \cdot P = H(ID_B)$. Then the simulator $\mathcal{S}$ does similar to the above situation, but PS is assumed to be $PS = CDH(P_A, P_B) + c \cdot X \cdot Y$. $\mathcal{S}$ can get $CDH(P_A, P_B) = PS - c \cdot X \cdot Y = p_A \cdot p_B \cdot P$, and then computes $Z = CDH(P_A, P_B) \cdot a_B = p_A \cdot p_B \cdot H(ID_B)$. Next, the simulator queries the DDH-oracle with $(Y \cdot {r_B}^{-1}, P_A, Z)$, where r_A is generated by the simulator itself. If the oracle outputs "yes", $\mathcal{S}$ responds the previously recorded values (K_1, K_2) to $\mathcal{A}$; otherwise, it returns random values.

a: IMPERSONATION SECURITY

We give the proof of the impersonation security in this section. It indicates the probability that there is no matching session about SID_T is negligible. Assume that in the finished and fresh test-session SID_T, the adversary $\mathcal{A}$ might be able to impersonate the honest party ID_k, provided that there does not exist matching session. We separate two scenarios based on whether SID_T is executed at Initiator or Responder, and give proof.

The test-session SID_T executed at Initiator is the first case, which is denoted by Case-1. In this case, we denote $ID_t = ID_A$ as the test-session owner and $ID_k = ID_B$ as the peer party $peer_{SID_T}$. Firstly, according to Lemma 1, we have that Y was never produced with overwhelming probability and delivered by ID_B in any session that is now in existence. The impersonation security can be proved based on the GDH assumption and the AE security.

After receiving the tuple "$(Test, ID_{t_0}, ID_{t_1})$", $\mathcal{S}$ sets $ID_t = ID_{t_b}$ and sends X to $\mathcal{A}$ as the initial message of the test-session. Case-1 supposes that private-key of ID_k is unrevealed and SID'_T does not exists.

- All the other oracles queried by $\mathcal{A}$ regarding $StaKeyReveal$, $Send$ and **Peer** can be answered by $\mathcal{S}$. The experiment is terminated by $\mathcal{S}$ when the **MSKReveal** query is asked. Assuming KDF to be an RO and under the run of $\mathcal{S}$, it is simple to verify that the view of $\mathcal{A}$ is the same as that in its actual attack. In addition, as previously explained in reference to the simulation of $peer_{SID_T}$ and SID_T, $\mathcal{S}$ can well simulate ID_B's behaviors.
  The GDH solver $\mathcal{S}$ inputs $(ID_B, X, c \cdot P)$, where $X \leftarrow \mathbb{G}_1/1_{\mathbb{G}_1}$, and the master secret key is defined to be $msk = c$ (c is unknown to $\mathcal{S}$). The aim of $\mathcal{S}$ is to compute $CDH(X \cdot H(ID_B), c \cdot P)$ with a DDH-oracle. In order to achieve this goal, it takes $k \leftarrow \{1, \ldots, n\}$ and uses a random guessing algorithm to determine the peer user $peer_{SID_T} = ID_k = ID_B$ with probability $\frac{1}{n}$. It then produces the public-keys and the private-keys for all honest parties except ID_B. $\mathcal{S}$ delivers X in the initial round of SID_T. The AE ciphertext transmitted by $\mathcal{A}$ during the second round of SID_T is defined by (Y, C_B). Case-1 implies that Y was not produced by party ID_B, but the decryption of the ciphertext C_B obtains (ID_B, y) such that $y \in \mathbb{Z}_q^*$ and $Y = y \cdot H(ID_B)$, where y may be produced by the adversary and the adversary asked the random oracle $H(ID_B)$. Suppose that $(K_1, K_2) = KDF(PS, X \parallel Y)$. Given that Y was not sent by ID_B and SID_T does not have matching session, with overwhelming probability the AE ciphertext $Enc_{K_1}(y, ID_B)$ was not transmitted in no other session but the test session. According to the security of authenticated encryption, $\mathcal{A}$ must have asked the oracle $KDF(PS, X \parallel Y)$, where $PS = CDH(X \cdot Y, c \cdot P) + p_A \cdot P_B$ which could be verified with the DDH-oracle. As a result, $\mathcal{S}$ gets $CDH(X \cdot Y, c \cdot P) = PS - p_A \cdot P_B = c \cdot H(ID_A) \cdot r_A \cdot p_A \cdot r_B \cdot p_B \cdot H(ID_B)$, and $\mathcal{S}$ uses the key K_1 to decrypt C_B and then gets y, from which it computes $CDH(X \cdot H(ID_B), c \cdot P) = CDH(X \cdot Y, c \cdot P) \cdot y^{-1} = c \cdot r_A \cdot p_A \cdot H(ID_A) \cdot H(ID_B)$, which violates the GDH assumption.
- All the other oracles queried by $\mathcal{A}$ regarding $StaKey1Reveal$, $Send$, **Peer** and **MSKReveal** can be answered by $\mathcal{S}$.
  The GDH solver $\mathcal{S}$ inputs (P_B, X, c), where $X \leftarrow \mathbb{G}_1/1_{\mathbb{G}_1}$, and the master secret key is defined to be $msk = c$. The aim of $\mathcal{S}$ is to calculate $CDH(X, c \cdot P_B)$ with a DDH-oracle. In order to achieve this goal, it takes $k \leftarrow \{1, \ldots, n\}$ and uses a random guessing algorithm to determine the peer user $peer_{SID_T} = ID_k = ID_B$ with probability $\frac{1}{n}$. It then produces the partial private-keys and the public-keys for all honest parties except ID_B. $\mathcal{S}$ delivers X in the first round of SID_T. Given that Y was not sent by ID_B and SID_T does not have matching session, with overwhelming probability the AE ciphertext C_B was not transmitted in no other session but the test-session. According to AE's security, the adversary $\mathcal{A}$ must have asked the oracle $KDF(PS, X \parallel Y)$, where $PS = CDH(P_A, P_B) + c \cdot X \cdot Y$ which could be verified with the DDH-oracle. As a result, $\mathcal{S}$ gets $CDH(P_A, P_B) = PS - c \cdot X \cdot Y = p_A \cdot p_B \cdot P$. As the simulator can generate r_A and also knows a_A, $\mathcal{S}$ computes $Z = CDH(P_A, P_B) \cdot a_A = p_A \cdot p_B \cdot H(ID_A)$. Finally, $\mathcal{S}$ gets $CDH(X, c \cdot P_B) = Z \cdot r_A \cdot c = p_A \cdot p_B \cdot H(ID_A) \cdot r_A \cdot c$, which violates the GDH assumption.

The test session SID_T executed at Responder is the second case, which is denoted as Case-2. In the second case, we denote $ID_t = ID_B$ as the test-session owner and $ID_k = ID_A$ as the peer party $peer_{SID_T}$.

We assume the probability of the event that there exists an adversary $\mathcal{A}$ such that Case-2 occurs is negligible.

- The GDH solver $\mathcal{S}$ inputs $(ID_A, Y, c \cdot P)$, where $Y \leftarrow \mathbb{G}_1/1_{\mathbb{G}_1}$, and the goal of $\mathcal{S}$ is to compute $CDH(Y \cdot H(ID_A), c \cdot P)$ with a DDH oracle.

In order to achieve this goal, it uses a random guessing algorithm to determine the peer user ID_A with probability $\frac{1}{n}$. With the exception of ID_A, it then produces the public-keys and the private-keys for each honest party. Upon receiving X in the initial round of SID_T, and obtaining Y that is input by $\mathcal{S}$, $\mathcal{S}$ then computes the ciphertext C_B, where the keys (K_1, K_2) are set by $\mathcal{S}$ with the DDH-oracle to maintain KDF consistency as previously explained in relation to the $peer_{SID_T}$ simulation. During the second round of SID_T, $\mathcal{S}$ transmits $\{Y, C_B\}$. The ciphertext transmitted by $\mathcal{A}$ in the third-round is set by C_A. The simulator uses the key K_1 to decrypt C_A and gets (ID_A, x) such that $x \in \mathbb{Z}_q^*$ and $X = x \cdot H(ID_A)$ assuming that the test-session SID_T is complete. Note that, before sending X in the first round of SID_T, $\mathcal{A}$ has queried the oracle $H(ID_A)$ with overwhelming probability; otherwise, SID_T will not succeed during the third round with overwhelming probability. And the query $H(ID_A)$ may not be asked by $\mathcal{A}$ itself. For instance, suppose that: the oracle $H(ID_A)$ is computed by ID_A in a non-matching session, where x is revealed to the adversary. Since we suppose SID_T does not have matching session, the key K_1 employed in SID_T has no concern with the AE-keys in each of the remaining sessions. According to the security of authenticated encryption, if $\mathcal{S}$ is able to break the impersonation security in Case-2 with non-negligible probability, $\mathcal{S}$ will obtain $PS = CDH(X \cdot Y, c \cdot P) + p_B \cdot P_A$, from which it calculates $CDH(X \cdot Y, c \cdot P) = PS - p_B \cdot P_A = c \cdot r_A \cdot p_A \cdot H(ID_A) \cdot r_B \cdot p_B \cdot H(ID_B)$ and $\mathcal{S}$ decrypts C_B with the K_1 to learn x. Then the simulator can get $CDH(Y \cdot H(ID_A), c \cdot P) = CDH(X \cdot Y, c \cdot P) \cdot x^{-1} = c \cdot H(ID_A) \cdot r_B \cdot p_B \cdot H(ID_B)$, which violates the GDH assumption.

- The GDH solver $\mathcal{S}$ inputs (P_A, Y, c), where $Y \leftarrow \mathbb{G}_1/1_{\mathbb{G}_1}$. The goal of $\mathcal{S}$ is to calculate $CDH(Y, c \cdot P_A)$ with the help of DDH oracle.

  To achieve this target, it uses a random guessing algorithm to determine the peer party ID_A with successful probability $\frac{1}{n}$. With the exception of ID_A, it then produces the public-keys and the private-keys for each honest party. Upon receiving X in the initial round of SID_T, and obtaining Y which is input by $\mathcal{S}$, $\mathcal{S}$ then computes the ciphertext C_B, where the keys (K_1, K_2) are set by $\mathcal{S}$ with the help of DDH-oracle to maintain the KDF consistency as previously explained in relation to the $peer_{SID_T}$ simulation. During the second round of SID_T, $\mathcal{S}$ transmits $\{Y, C_B\}$. Since we suppose SID_T does not have matching session, K_1 employed in SID_T has no concern with the AE-keys in each of the remaining sessions. According to the security of authenticated encryption, if $\mathcal{S}$ is able to break the impersonation security in Case-2 with non-negligible probability, $\mathcal{S}$ will obtain $PS = c \cdot X \cdot Y + CDH(P_A, P_B)$, from which it calculates $CDH(P_A, P_B) = PS - c \cdot X \cdot Y = p_A \cdot p_B \cdot P$. As the simulator $\mathcal{S}$ can generate r_B and also knows a_B, it can get $Z = CDH(P_A, P_B) \cdot a_B = p_A \cdot p_B \cdot H(ID_B)$. Finally, $\mathcal{S}$ gets $CDH(Y, c \cdot P_A) = Z \cdot r_B \cdot c = p_A \cdot p_B \cdot H(ID_B) \cdot r_B \cdot c$, which violates the GDH assumption.

This completes the proof of impersonation security.

b: ID-SK INDISTINGUISHABILITY

According to the security of authenticated encryption, in purpose of breaking the ID-SK indistinguishability, $\mathcal{A}$ must ask the query $KDF(PS = CDH(X \cdot Y, c \cdot P) + p_A \cdot P_B, X \parallel Y)$ or $KDF(PS = CDH(P_A, P_B) + c \cdot X \cdot Y)$ with non-negligible probability. Then it is reduced to the GDH assumption in the random oracle model. Since the proof strategy is the same as the above proof of impersonation security, the proof here is simplified.

For presentation simplicity, one of $\{peer_{SID_T}, ID_t\}$ represents ID_A and the other represents ID_B. If the GDH solver $\mathcal{S}$ is unaware of the master secret key, and it inputs $(U, V, c \cdot P)$, where $U, V \leftarrow \mathbb{G}_1/1_{\mathbb{G}}$, and the aim of $\mathcal{S}$ is to calculate $CDH(V \cdot U, c \cdot P)$ with DDH-oracle. The full public key and full private key of ID_t are computed by $\mathcal{S}$ itself, while the value of hashing the partial public-key (ID_k) of $peer_{SID_T}$ and the ECDH-component to be transmitted by ID_t in SID_T are set to be (U, V), i.e., the input provided to $\mathcal{S}$. As $\mathcal{S}$ learns (sk_A, p_A, y) or (sk_B, p_B, x) and from $PS = CDH(X \cdot Y, c \cdot P) + p_A \cdot P_B$ or $PS = CDH(X \cdot Y, c \cdot P) + p_B \cdot P_A$, the simulator can get $CDH(V \cdot U, c \cdot P)$ that is either $CDH(X \cdot H(ID_B), c \cdot P)$ or $CDH(Y \cdot H(ID_A), c \cdot P)$, which violates the GDH assumption.

If the GDH solver $\mathcal{S}$ knows the master secret key, and it inputs (U, V, c), where $U, V \leftarrow \mathbb{G}_1/1_{\mathbb{G}}$, and the aim of $\mathcal{S}$ is to calculate $CDH(V, c \cdot U)$ with DDH-oracle. The partial public-key P_{ID_k} of $peer_{SID_T}$ and the ECDH-component to be transmitted by ID_t in SID_T are set to be (U, V), while the full public key and partial private secret key of ID_t are computed by $\mathcal{S}$. As $\mathcal{S}$ knows (sk_A, r_A, a_A) or (sk_B, r_B, a_B) and from $PS = CDH(P_A, P_B) + c \cdot X \cdot Y$, the simulator can get $CDH(V, c \cdot U)$ that is either $CDH(X, c \cdot P_B)$ or $CDH(Y, c \cdot P_A)$, which violates the GDH assumption.

From the full proof above, we can get that the protocol holds the label-security and session-key security with identity-concealment. The proof of Theorem 1 is finished.

B. OTHER SECURITY PROPERTIES

The proposed ID-AKA protocol satisfies other security properties as well. The security properties are explained as follows.

Known-key Security (KKS): Although the current session key and the transmitted records are leaked, the adversary is incapable of calculating any past or future session key. Specifically, the adversary obtain final key $K_2 \leftarrow KDF(PS, X||Y)$, X and Y, where $PS = sk_A \cdot r_A \cdot p_A \cdot Y + p_A \cdot p_B \cdot P$ or $PS = sk_B \cdot r_B \cdot p_B \cdot X + p_A \cdot p_B \cdot P$. However, without the knowledge of static and ephemeral secret keys of participants, previous or future session key can not be influenced even though the session key of present session is learned. Consequently, our protocol can resist against known-key attack.

Key Compromise Impersonation (KCI): The adversary is not able to disguise himself as the other entity to engage in

the key exchange, even though the static secret key of one entity is revealed. Our proposed protocol offers the property of key compromise impersonation resilience.

Perfect Forward Security (PFS): As the adversary does not know ephemeral secret key r, the adversary is unable to recover previous session key although the full static secret keys of all parties are exposed.

Ephemeral Secret Leakage Security: Despite the fact that the session's ephemeral secrets are leaked to adversary, the security of the session key can still be ensured because of without knowledge of long-term private key. Thus, the proposed protocol can defend against ephemeral secret leakage attack.

Non-full Key Escrow: Namely, the static secret keys of entities fully relies on the trusted authority, which will affect the security of protocol upon TA compromise. In our protocol, only one party of private key depends on TA. Thus, our protocol satisfies non-full key escrow.

Mutual Authentication: In our protocol, two parties verified the decryption effectiveness of ciphertext ($C_A.C_B$) after receiving the second and third messages. If the validation process is successful, it implies that the message is being sent by an honest instance. Hence, the mutual authentication is supported by the proposed ID-AKA protocol.

VI. PERFORMANCE EVALUATION

In this section, we first give a functionality comparison among our proposed ID-AKA protocol and several related protocols [20], [21], [32], [33], where the protocols [20], [32], [33] are ID-AKA schemes and the protocol [21] is a certificateless AKA scheme. Then, we present performance analysis and comparison with regard to communication cost and computational efficiency of the proposed protocol and the related protocols [20], [21], [32], [33].

A. FUNCTIONALITY COMPARISON

In this section, we briefly compare the security and functionality features of our scheme with the related schemes [20], [21], [32], [33].

As shown in Table 1, our scheme has stronger security and is the only one that provides all of the following properties: ephemeral-secret leakage security, key compromise impersonation security,non-full key escrow, identity privacy, post-specified ID and mutual authentication. Ephemeral-secret leakage security means that the session key is still preserved although the ephemeral secret is revealed. Identity privacy implies that the transmitted record contains no personal information and does not leak any information about participants' identity. Only our protocol and the protocol in [20] work in the post-specified ID setting, where the initiator isn't aware of the identity information of the peer when the protocol begins. Other security features has been explained in Section V-B. From the comparison results, except for the protocol in [32], other protocols are secure from the attack of the ephemeral-secret leakage. The protocol in [21] and our proposed protocol meets non-full key escrow. What's more, only the protocol in [20] and our protocol fulfill both identity privacy and mutual authentication. In short, our proposed ID-AKA protocol is superior to other four homogeneous protocols concerning security.

B. COMMUNICATION AND COMPUTATION COST COMPARISON AND ANALYSIS

We mainly compare and analyze the communication and computation cost of the proposed protocol and each comparison object [20], [21], [32], [33].

1) THE COMPARISON OF COMMUNICATION COST

Table 2 lists the comparison of the communication overhead between our scheme and the related schemes [20], [21], [32], [33].Suposing that the timestamp size is 32 bits, and the identity size is 160 bits. The size of each point in the ECC is 320 bits. Additionally, let the size of each element in the multiplication bilinear map group is 320 bits. In protocol [20], the exchanged message is $\{X, Y, C_B, C_A\}$. The total communication cost in [20] is $320 + 320 + 256 + 256 = 1152$ bits. In protocol [21], the exchanged message is $\{w'_A = ID||t||X, R_A, T_A, w'_B, R_B, T_B\}$. The total communication cost in [21] is $512 + 320 + 640 + 512 + 320 + 640 = 2944$ bits. In protocol [32], the exchanged message is $\{\psi_1, \sigma_1, \psi_2, \sigma_2\}$. The total communication cost in [32] is $320 + 320 + 320 + 320 = 1280$ bits. In protocol [33], the exchanged record is $\{ID_A, R_{A_1}, R_{A_2}, T_{A_1}, T_{A_2}, ID_B, R_{B_1}, R_{B_2}, T_{B_1}, T_{B_2}\}$. The total communication cost in [33] is $(160+320+320+320+320) \times 2 = 2880$ bits. In our protocol, the exchanged message is $\{X, Y, C_B, C_A\}$. The total communication cost of our protocol is $320+320+256+256 = 1152$ bits. From the analysis above and the results in Table 2, it can be said that our protocol and the protocol in [20] has the same communication overhead, which are more efficient than other two protocols.

2) THE COMPARISON OF COMPUTATION COST

The comparison results among our protocol and other protocols [20], [21], [32], [33] are depicted in Table 2. The computational costs in the key derivation and the authenticated key agreement are listed. Therein, "H" represents general hash operation, "E" represents modular exponentiation in $\mathbb{G}$, "M" represents ECC-based point multiply, "$PM_{\mathbb{G}_1}$" represents pairing-based group multiply in $\mathbb{G}_1$, "MTP" represents map-to-point hash, "Inv" represents modular inverse operation, and "P" represents bilinear pairing operation in $\mathbb{G} \times \mathbb{G}_1$.

For the key derivation phase of our protocol, the user needs to generate a part of private keys and generate the associated public keys. And the key generation center computes the other part of private keys for the user by hashing the identity and then using the master secret-key for point multiplication operation. The total operations for two parties to performed in this phase are $4M + 2MTP$. For the authenticated key agreement phase, the sum of operations performed by both parties is $2H + 10M + 2MTP$. Compared with other protocols in [20], [21], [32] and [33], our protocol's computational expense in the key derivation phase is in the middle level and only slightly higher than protocols [20], [21].

TABLE 1. Comparison of security and functionality features.

Features	Protocol [20]	Protocol [21]	Protocol [32]	Protocol [33]	Ours
Ephemeral-secret leakage security	✓	✓	×	✓	✓
Key compromise impersonation security	✓	✓	✓	✓	✓
Non-full key escrow	×	✓	×	×	✓
Identity privacy	✓	×	×	×	✓
Post-specified ID	✓	×	×	×	✓
Mutual authentication	✓	×	×	×	✓

TABLE 2. Comparison of computation and communication cost.

Protocol	Computation cost		Communication cost (bits)
	Key derivation	Authenticated key agreement	
Protocol [20]	$2MTP + 2E$	$4E + 6E + 2P + 2MTP$	1152
Protocol [21]	$4M + 2H$	$14M + 4H$	2944
Protocol [32]	$2H + 2PM_{\mathbb{G}_1} + 2Inv$	$2H + 8PM_{\mathbb{G}_1} + 4P$	1280
Protocol [33]	$8M + 8H$	$14M + 6H$	2880
Ours	$4M + 2MTP$	$2H + 10M + 2MTP$	1152

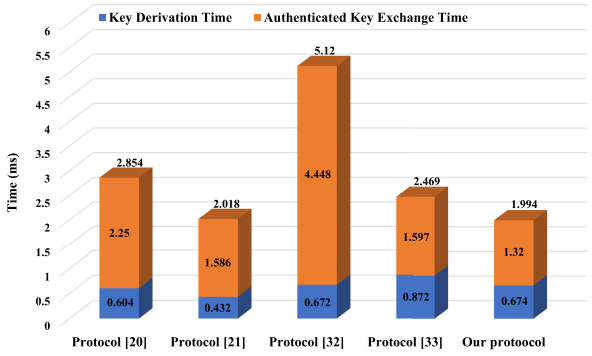

FIGURE 5. Comparison in time consumption.

In the authenticated key agreement phase, the computational expense of our protocol is the least one. Note that it is more important to evaluate the cost of online calculation than that of offline calculation. Therefore, our ID-AKA protocol is the best with regard to total computational cost.

We provide the time consumption of our proposed protocol through experiment. The experiment is carried out by using C++ language and on the platform with 2.9 GHz CPU AMD Ryzen 7 4800H with 8GB of RAM and Ubuntu 20.04 LTS. The protocol is executed using the mcl library and OPENSSL library. Figure 5 presents the comparison about computational performance of our protocol and other related protocols [20], [21], [32], [33] which may represent the state of the art in security or efficiency. The results show that our protocol's time consumption in authenticated key agreement phase and total time consumption in two phases are both the least.

According to the comparison and analysis of functionality features, communication and computation overhead, our proposed ID-AKA protocol for 5G network is more efficient and secure than the current ID-AKA protocols, which has practical applicability.

VII. CONCLUSION

We design a privacy-enhanced identity-based authenticated key agreement protocol for secure communication. It provides the protection of identity privacy and performs no pairing operations, which is suitable for 5G network. Moreover, our scheme enjoys forward identity privacy for all parties and non-full escrow, and works in the post-specified ID setting. The full security proof of our proposed protocol in a modified security model is presented in this work. The security analysis and experiment analysis demonstrate that our proposed protocol realizes desirable security and efficiency.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.
[2] S. Mumtaz, A. Alsohaily, Z. Pang, A. Rayes, K. F. Tsang, and J. Rodriguez, "Massive Internet of Things for industrial applications: Addressing wireless IIoT connectivity challenges and ecosystem fragmentation," *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 28–33, Mar. 2017.
[3] (2020). *GSMA Homepage*. Accessed: Dec. 2020. [Online]. Available: https://www.gsma.com/globalmobiletrends/
[4] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. CRYPTO*, 1984, pp. 47–53.
[5] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Proc. CRYPTO*, 2001, pp. 213–229.

[6] N. P. Smart, "Identity-based authenticated key agreement protocol based on Weil pairing," *Electron. Lett.*, vol. 38, no. 13, pp. 630–632, 2002.
[7] A. Joux, "A one round protocol for tripartite Diffie–Hellman," in *Proc. ANTS*, Jul. 2000, pp. 385–394.
[8] O. Blazy and C. Chevalier, "Non-interactive key exchange from identity-based encryption," in *Proc. ARES*, Aug. 2018, pp. 13:1–13:10.
[9] C. Boyd, Y. Cliff, J. G. Nieto, and K. G. Paterson, "Efficient one-round key exchange in the standard model," in *Proc. ACISP*, 2008, pp. 69–83.
[10] L. Chen, Z. Cheng, and N. P. Smart, "Identity-based key agreement protocols from pairings," *Int. J. Inf. Secur.*, vol. 6, no. 4, pp. 213–241, Jun. 2007.
[11] R. M. Daniel, E. B. Rajsingh, and S. Silas, "An efficient eCK secure identity based two party authenticated key agreement scheme with security against active adversaries," *Inf. Comput.*, vol. 275, Dec. 2020, Art. no. 104630.
[12] O. Ruan, Y. Zhang, M. Zhang, J. Zhou, and L. Harn, "After-the-fact leakage-resilient identity-based authenticated key exchange," *IEEE Syst. J.*, vol. 12, no. 2, pp. 2017–2026, Jun. 2018.
[13] D. Fiore and R. Gennaro, "Making the Diffie–Hellman protocol identity-based," in *Proc. CT-RSA*, 2010, pp. 165–178.
[14] L. Ni, G. Chen, J. Li, and Y. Hao, "Strongly secure identity-based authenticated key agreement protocols without bilinear pairings," *Inf. Sci.*, vols. 367–368, pp. 176–193, Nov. 2016.
[15] K. Shim, "Efficient ID-based authenticated key agreement protocol based on Weil pairing," *Electron. Lett.*, vol. 39, no. 8, pp. 653–654, 2003.
[16] J. Tomida, A. Fujioka, A. Nagai, and K. Suzuki, "Strongly secure identity-based key exchange with single pairing operation," in *Proc. ESORICS*, 2019, pp. 484–503.
[17] Y.-M. Tseng, J.-L. Chen, and S.-S. Huang, "A lightweight leakage-resilient identity-based mutual authentication and key exchange protocol for resource-limited devices," *Comput. Netw.*, vol. 196, Sep. 2021, Art. no. 108246.
[18] M. Xie and L. Wang, "One-round identity-based key exchange with perfect forward security," *Inf. Process. Lett.*, vol. 112, nos. 14–15, pp. 587–591, Aug. 2012.
[19] J. Zhang, X. Huang, W. Wang, and Y. Yue, "Unbalancing pairing-free identity-based authenticated key exchange protocols for disaster scenarios," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 878–890, Feb. 2019.
[20] H. Lian, T. Pan, H. Wang, and Y. Zhao, "Identity-based identity-concealed authenticated key exchange," in *Proc. ESORICS*, 2021, pp. 651–675.
[21] L. Meng, H. Xu, H. Xiong, X. Zhang, X. Zhou, and Z. Han, "An efficient certificateless authenticated key exchange protocol resistant to ephemeral key leakage attack for V2V communication in IoV," *IEEE Trans. Veh. Technol.*, vol. 70, no. 11, pp. 11736–11747, Nov. 2021.
[22] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. IT-22, no. 6, pp. 644–654, Nov. 1976.
[23] M. Bellare and D. P. P. Rogaway, "Authenticated key exchange secure against dictionary attacks," in *Proc. EUROCRYPT*, May 2000, pp. 139–155.
[24] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *Proc. EUROCRYPT*, 2001, pp. 453–474.
[25] B. A. LaMacchia, K. Lauter, and A. Mityagin, "Stronger security of authenticated key exchange," in *Proc. Provable Secur. (ProvSec)*. Wollongong, NSW, Australia: Springer, Nov. 2007, pp. 1–16.
[26] H. Huang and Z. Cao, "An ID-based authenticated key exchange protocol based on bilinear Diffie–Hellman problem," in *Proc. ASIACCS*, Mar. 2009, pp. 333–342.
[27] M. Hölbl, T. Welzer, and B. Brumen, "An improved two-party identity-based authenticated key agreement protocol using pairings," *J. Comput. Syst. Sci.*, vol. 78, no. 1, pp. 142–150, Jan. 2012.
[28] L. Ni, G. Chen, J. Li, and Y. Hao, "Strongly secure identity-based authenticated key agreement protocols in the escrow mode," *Sci. China Inf. Sci.*, vol. 56, no. 8, pp. 1–14, Aug. 2013.
[29] I. Elashry, Y. Mu, and W. Susilo, "A resilient identity-based authenticated key exchange protocol," *Secur. Commun. Netw.*, vol. 8, no. 13, pp. 2279–2290, Sep. 2015.
[30] J.-D. Wu, Y.-M. Tseng, and S.-S. Huang, "An identity-based authenticated key exchange protocol resilient to continuous key leakage," *IEEE Syst. J.*, vol. 13, no. 4, pp. 3968–3979, Dec. 2019.
[31] C. Cremers, M. Naor, S. Paz, and E. Ronen, "CHIP and CRISP: Protecting all parties against compromise through identity-binding PAKEs," in *Proc. CRYPTO*, Aug. 2022, pp. 668–698.
[32] D. S. Gupta, S. H. Islam, M. S. Obaidat, P. Vijayakumar, N. Kumar, and Y. Park, "A provably secure and lightweight identity-based two-party authenticated key agreement protocol for IIoT environments," *IEEE Syst. J.*, vol. 15, no. 2, pp. 1732–1741, Jun. 2021.
[33] R. M. Daniel, A. Thomas, E. B. Rajsingh, and S. Silas, "A strengthened eCK secure identity based authenticated key agreement protocol based on the standard CDH assumption," *Inf. Comput.*, vol. 294, Oct. 2023, Art. no. 105067.

HUANHUAN LIAN received the Ph.D. degree from the College of Computer Science and Technology, Fudan University, Shanghai, China, in 2023. She is currently a Researcher with the Research Institute of China Telecom Corporation Ltd., Shanghai. Her research interests include data security and information security, in particular, applied cryptography, and secure multi-party computation.

BURONG KANG received the Ph.D. degree from the School of Software Engineering, East China Normal University, Shanghai, China. She is currently working with the Research Institute of China Telecom Corporation Ltd., Shanghai. Her research interests include information security, cryptography, cloud security, data privacy, public key cryptography, and network security.

LIBAO YANG received the master's degree in management science and engineering from Beijing Institute of Technology, in 2017. He is currently the Senior Security Project Manager of China Telecom Group Corporation. His research interests include big data security and privacy protection.