Deniable-based Privacy-preserving Authentication against Location Leakage in Edge Computing

Shengke Zeng, Hongjie Zhang, Fei Hao, and Hongwei Li

***Abstract*—Edge computing provides cloud services at the edge of the network for Internet of Things (IoT) devices. It aims to address low latency of the network and alleviates the data processing of the cloud. This "cloud-edge-device" paradigm brings convenience as well as the challenges for location-privacy protection of IoT. In the edge computing environment, the fixed edge equipments supply the computing services for the adjacent IoT devices. Therefore, edge computing suffers location leakage as the connection and authentication records imply the location of IoT devices. This work focuses on the location awareness in edge computing environment. We adopt the "deniability" of authentication to prevent location leakage while IoT devices connecting to the edge nodes. In our solution, an efficient deniable authentication based on 2-user ring signature is constructed. The robustness of authentication makes the fixed edge equipments to accept the legal end device. Besides, the deniability of authentication cannot convince any third party that the fact of this authentication occurred as communication transcript is no longer an evidence for this connection. Therefore, it handles the inherent location risk in edge computing. Compared to efficient deniable authentications, our protocol saves 10.728% and 14.728% computational cost, respectively.**

***Index Terms*—Edge Computing, Location Privacy, Privacy-preserving Authentication, Deniability.**

I. Introduction

EDGE computing is a distributed computing paradigm which brings the cloud resources closer to IoT devices or local edge servers. Compared to the traditional cloud computing, edge computing improves response times and gains better bandwidth availability. In edge computing, the edge equipment location is known and fixed as the beacon node. In order to reduce communication costs, the IoT devices usually select the nearest edge equipments to conduct the tasks. Obviously, edge computing paradigm discloses the location of IoT devices to the public.

On the other hand, authentication is necessary for IoT devices to connect and communicate with edge equipments. When communication occurs, it ensures that IoT devices are the legal counterparties and the message delivered from the source is intact. Digital signature seems to be alternative to realize identity authentication and message authentication. However, the public verifiability of digital signature convinces anyone that this authentication is made by the source. Therefore, the location of IoT device is revealed naturally from the communication to the edge equipment.

Location is a kind of very important privacy information for clients as the same as the identity of users. Some research indicates the location information is closely related to the individual habits, activities and relationships [1]. Therefore, the location privacy receives great concerns [2], [3], [4]. Many solutions for protecting user's location privacy are proposed. The straight approach regarding to location privacy is to use pseudonyms to make the user identity and location information irrelevant [5]. It requires to carry great number of certificates for user to achieve the strong privacy thus it is inefficient in terms of storage. The second one is to enlarge the user's location into a region in which the accurate location information is replaced by a coarse-grained position to prevent the attacker from learning the exact location. Such as the spatial cloaking technique [6], the user's location is hidden in a large cloaked area such that it cannot be pinpointed by the attacker. This kind of approaches always degrade the quality of location-based services. Mix-zone technique is also widely employed for location privacy [7], [8], which allows the users to exchange their pseudonyms in a dedicated area. Therefore, dummy users have to be created for privacy if there is short of sufficient neighboring users in the area (mix zone). K-anonymity as another kind of approach to handle privacy [9], [10], is to utilize other k-1 positions to cover user location. Obviously, it requires a mass of location information to be involved in the privacy preservation. Since the location information is confidential in location-aware applications, the cryptographic algorithms are employed to encrypt the location data. Homomorphic encryptions are used to protect the location privacy during the localization [11], [12], [13], [14]. Undoubtedly, cryptographic primitives provide the high-level privacy but also lead to the heavy computation and storage. In addition, differential privacy is also an important tool to prevent the disclosing sensitive information [15], [16]. Recently, differential privacy technology is used to protect user's location privacy by a geo-indistinguishable task allocation [17].

In summary, the existing location privacy studies mainly focus on preventing the location leakage to attacker or the server which provides the location-based service, therefore, the encryptions or obfuscated location are necessary. Thus, both efficiency and service quality are always the victims of privacy protection. While in the edge computing environment, encrypting position or coarse position does not take effect

S. Zeng and H. Zhang are with the School of Computer and Software Engineering, Xihua University, Chengdu 610039, China; S. Zeng is also with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China.

F. Hao is with the School of Computer Science, Shaanxi Normal University, Xi'an 710119, China and also with the College of Engineering, Mathematics, and Physical Sciences, University of Exeter, Exeter, EX4 4QF, U.K.

H. Li is with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

*Corresponding author: Fei Hao ; Email: feehao@gmail.com:

Manuscript received****, ****; revised ****, ****.

as the edge equipment location (which is known) implies IoT device location from connection. Intuitively, the identity anonymity technique enables IoT device location is irrelevant to its identity. However, it cannot defend against the attackers with prior knowledge. We observe that if the communication between IoT device and the fixed edge equipment is "off-the-record", the location privacy is preserved although the communication peer captures the accurate location. In other words, we make the conversation peer has no evidence to convince any third party the fact that this interaction has occurred. This off-the-record communication can be viewed as the *deniability* capability to the protocol participants.

Indeed, deniability is an important privacy-preserving feature of cryptographic protocols. Since the authentication during the connection between IoT device and edge equipment is necessary. If the location privacy preservation is also required, the authentication protocol with deniability should be considered without degrading the service quality. With deniable authentication, the edge equipment guarantees the legality of the IoT device. However, the edge equipment cannot prove to any third party that the IoT device was ever involved in this connection. In other words, this connection is "off-the-record", which is a critical clue to provide the client privacy. Obviously, there is no evidence to reveal the connection to edge equipment.

The edge equipment is fixed and its location is public, the communication records between the IoT devices and edge equipments imply user's location privacy except that user can deny the fact. In this paper, we aim to design a deniable-based privacy-preserving authentication scheme to provide both authentication and location privacy. Latency is the critical evaluation metric in edge computing, therefore the low communication capacity should be concerned for the practicality. Intuitively, the non-interactive deniable authentication seems to be candidate. However, the non-interactive deniable authentications can reach the *partial* deniability only. It implies that the simulation can be made only by the receiver not anyone. Therefore, the sender can deny its involvement as the receiver may produce the indistinguishable communication transcript. Unfortunately, *partial* deniability is not enough for edge computing scenario since the receiver is an edge equipment which might be a trusted party by the public. Therefore, the communication transcript points to the client (the IoT device). Thus, we focus on the *full* deniability which requires interactive steps inevitably. In order to make authentication protocol applicable to the Internet-based service practically, we should consider concurrent environment. For the black-box simulation in case of malicious verifier, the challenge-response mechanism has to be added in the traditional approaches of *deniability*. Hence, it causes extra communication round and rewinding steps in simulation. Thus, the concurrent deniability does not hold and the heavy communication cost also reduces the quality of service in the edge computing. In addition, the encryption algorithm is a common primitive to construct deniable authentication protocols [18], [19]. The underlying cost is that the encryption must be CCA2-secure which is inefficient for practice. In order to be suitable for practicable applications, we adopt different primitives to avoid CCA-paradigm and any strong number-theoretic assumptions. The major contributions of this work are summarized as follows.

1) We present a privacy-preserving authentication scheme with *full* deniability to avoid the location leakage of IoT devices in edge computing. The fact on deniability of the communication is favorable for IoT devices as it does not have any evidence to show that IoT devices have ever been involved in some connection to the edge equipment. It preserves location privacy of the IoT devices naturally.
2) We focus on both latency and concurrency of the edge computing environment. Most of fully deniable authentications are built on the zero-knowledge argument against malicious verifier, thus they require at least 4 rounds. This security is strong but not practical in communication applications. We observe that the communication round can be reduced to 2 if the verifier follows protocol honestly. This assumption is feasible as the receiver in edge computing is the edge equipment which is accepted by the public and its behavior would be honest for its reputation. Therefore, this simplified 2-round authentication protocol with full deniability is optimal for the IoT devices connection to edge equipments. In addition, our protocol does not require extra challenge-response mechanism to extract the witness. Therefore, the deniability does not fail in concurrent setting such as Internet environment.
3) We avoid the need for encryption algorithms in our construction to realize authentication, otherwise the underlying encryption must be CCA2-secure which is inefficient. Moreover, we are not dependent on the strong and inefficient assumptions. The existed works to achieve the fully concurrent deniability by the timing constraints, plaintext awareness (PA) of the underlying encryption, knowledge of exponent assumption (KEA) assumption or the public random oracles. It results in the inefficiency or impracticable. Instead, we adopt the 2-user ring signature to simulate the deniability and the full deniability is met by the unconditional anonymity of the ring signature. Therefore, we avoid the underlying CCA2-secure primitive, the strong number-theoretic assumptions and public random oracles.

The rest of this paper is organized as follows. Section II introduces the related work of deniable authentications. Section III provides some preliminaries that are the building blocks in our protocol. Section IV describes the system model of edge computing and its security requirements. In Section V, we propose a privacy-preserving authentication scheme with full deniability and apply it to edge computing environment to protect the location privacy for IoT devices. The security of our scheme is proven and the performance is analyzed in Section VI. Section VII concludes this work.

II. Related Work

Deniable authentication was first introduced by Dolev *et al*. [20] and formally studied by Dwork *et al.* [18]. It follows "simulation paradigm" to realize deniability. The authentication is deniable if the conversation transcript can be simulated without any secretes. Therefore, the participants can deny as

someone else would produce this indistinguishable communication transcript. We call it is *fully* deniable if this simulation can be run by anyone not only the verifier. The generic technique to realize the deniable authentication is to revoke the secret which is used to authenticate in an appropriate phase. Thus, the early works [18], [21] require more rounds to reveal the witness upon the receipt of the committed secret against malicious verifiers. In this way, the simulation is perfect as anyone with the revoked secret can simulate the statistically indistinguishable communication transcript.

However, this kind of approach leads to heavy communication rounds inevitably. It is not suitable for the Internet-based applications. In addition, the Internet is a fully concurrent environment. However, the simulation in the constructions relying on revoked witness requires rewinding steps. Obviously, this deniability cannot hold in the concurrent scenario.

Some related works have been proposed to overcome this barrier. There exists some approaches to reveal the witness without rewinding steps. Di Raimondo *et al.* [21] demonstrated that the assumption of plaintext-awareness [22] can be used to extract the witness. Inspired by Raimondo's idea, Zeng *et al.* [19] presented a deniable authentication with source hiding based on PA secure multi-receiver encryption. However, the underlying assumption is strong. Stinson *et al.* [23] proposed a 2-round deniable identification protocol. Its deniability against dishonest verifier is based on KEA assumption and random oracles. Their scheme does not rely on any signatures or encryptions thus it can be used in identity authentication only. Jiang [24] used the public random oracle to extract the witness to avoid the rewinding steps. Yao *et al.* [25] utilized KEA assumption to extract the witness to ensure the deniability in the concurrent interactive setting. Tian *et al.* [26] made use of the selectively unforgeable but existentially forgeable signature to simulate the transcript. Jiang [27] proposed a moderate encryption to realize deniability without rewinding by virtue of timed commitment. However, these works suffer the limitations such as the strong number-theoretical assumptions, inefficiency or public random oracles.

The direct application of deniable authentication is to design the deniable key exchange (DKE) protocols [25], [28], [29]. As Yao *et al.* claimed that if the key exchange is deniable then all the transactions using the session key generated by the key exchange can be deniable for both the participants [25]. There are some other applications for deniable authentication. Some researches concern E-mail privacy [30], [31]. Li *et al.* [32] applied the deniable authentication to pervasive computing and Zeng *et al.* [33] utilized the deniability to construct the privacy-preserving LBS.

III. Preliminaries

We present the building blocks including the deniable authentication and ring signature of our scheme in this section.

A. Deniable Authentication (DA)

Generally, DA protocols enable a sender to speak to the receiver privately. In other words, the receiver can guarantee the legality of the sender but cannot convince others that the sender has participated in the authentication. It is realized by without leaving the "paper trail" of the conversation. Formally, the communication record can be simulated by someone else and therefore, there is no evidence to show the sender involvement. In other words, the participates (not only the sender even the receiver) can deny the fact of an authentication conversation.

1) Security Properties of DA Protocol: There are two roles in authentication protocol, namely the sender (prover, denoted as $\mathtt{P}$) and the receiver (verifier, denoted as $\mathtt{V}$). $\mathtt{P}$ authenticates a message $\mathtt{m}$ to $\mathtt{V}$. The fundamental security requirements of the deniable authentication protocol are the *Completeness*, the *Authentication (Unforgeability)* and the *Deniability*.

- **Completeness**. $\mathtt{V}$ accepts the authentication for the message $\mathtt{m}$ with overwhelming probability if $\mathtt{P}$ and $\mathtt{V}$ follow the authentication protocol honestly.
- **Authentication (Unforgeability)**. This property states that an attacker $\mathcal{A}$ can not pretend to be the sender $\mathtt{P}$ to complete authentication. Consider the probabilistic polynomial time attacker $\mathcal{A}$ trying to forge a message. It adaptively chooses a sequence of arbitrary messages $\mathtt{m}_1$, $\mathtt{m}_2, \cdots$ and asks some good participant $\mathtt{P}_\mathtt{i}$ to authenticate $\mathtt{m}_\mathtt{i}$. We say that $\mathcal{A}$ succeeds if $\mathtt{V}$ accepts $\mathcal{A}$'s authentication to message $\mathtt{m} \notin \{\mathtt{m}_\mathtt{i}\}_{i=1,2,\cdots}$ as $\mathtt{P}_\mathtt{i}$ and $\mathcal{A}$ does not have $\mathtt{P}_\mathtt{i}$'s secret. The authentication (unforgeability) requirement is that the probability of success of $\mathcal{A}$ is negligible.
- **Deniability**. This property states that $\mathtt{P}$ and $\mathtt{V}$ can deny the involvement of authentication. Formally, the deniability can be captured by the *simulation* paradigm. The adversary $\mathcal{A}$'s view of this conversation can be simulated by a simulator $\mathcal{M}$ without the secret of the sender $\mathtt{P}$ and the two transcripts (the real one and the simulated one) have the same distribution. Therefore, the real authentication transcript can not be convinced by others as it can be performed by running $\mathcal{M}$. In addition, the concurrent deniability should be considered in the interactive fashion. In the Internet-based environment, the attacker $\mathcal{A}$ may launch a concurrent interaction with $\mathtt{P}$ by arbitrary interleaved steps. The concurrent deniability should hold even in such setting. We denote the interaction between $\mathcal{A}$ and real sender $\mathtt{P}$ by $\Gamma^{\mathtt{rea}}$, denote the interaction between $\mathcal{A}$ and the simulator $\mathcal{M}$ by $\Gamma^{\mathtt{sim}}$. The authentication is deniable if a distinguisher $\mathcal{D}$'s view in $\Gamma^{\mathtt{rea}}$ and $\Gamma^{\mathtt{sim}}$ are indistinguishable. Formally, $|\Pr[\mathcal{D}(\mathtt{view}(\mathcal{A}, \Gamma^{\mathtt{rea}})) = 1] - \Pr[\mathcal{D}(\mathtt{view}(\mathcal{A}, \Gamma^{\mathtt{sim}})) = 1]| \approx \mathtt{negl}(\kappa)$, where $\mathtt{negl}(\kappa)$ is a negligible function for the security parameter κ.

2) Review of DA Protocol : Let us review the traditional deniable authentication protocol proposed by Dwork *et al* [18] to explain the *Authentication (Unforgeability)* and the *Deniability*. In this protocol, $\mathtt{P}$ has a public-private keypair $(v_\mathtt{P}, s_\mathtt{P})$ of a non-malleable encryption algorithm $E(\cdot)$. $\mathtt{P}$ and $\mathtt{V}$ perform the interactive deniable authentication protocol as shown in Fig. 1.

Only the legal sender $\mathtt{P}$ can get the right r with the decryption key $s_\mathtt{P}$. Therefore, the correct r implies $\mathtt{P}$'s authentication. On the other hand, the exposure of r in **Round 3** is the vital

Round 1. $\mathtt{V} \rightarrow \mathtt{P}$: $\mathtt{V}$ randomly chooses value r and computes $c = E_{v_\mathtt{P}}(\mathtt{m} \parallel r)$.
Round 2. $\mathtt{P} \rightarrow \mathtt{V}$: Upon receiving message c, $\mathtt{P}$ decrypts c by its private key $s_\mathtt{P}$ and obtains the suffix of plaintext, namely r. Technically, $\mathtt{P}$ returns $d = E_{v_\mathtt{P}}(r)$ instead of r. d can be viewed as the commitment of r as $E(\cdot)$ is non-malleable.
Round 3. $\mathtt{V} \rightarrow \mathtt{P}$: Upon receiving message d, $\mathtt{V}$ opens c using the random encryption coin ρ used in the encryption in the Round 1 and returns (r, ρ) to $\mathtt{P}$.
Round 4. $\mathtt{P} \rightarrow \mathtt{V}$: Upon receiving message (r, ρ), $\mathtt{P}$ checks the correct r and opens d by revoking the random encryption coin σ used in the encryption in the Round 2.
Finally, $\mathtt{V}$ accepts $\mathtt{P}$'s authentication if d is opened correctly.

Fig. 1. Dwork *et al.*'s Deniable Authentication Protocol

step to realize the deniability. With r, anyone can produce d even without the secret $s_\mathtt{P}$. We can see that this simulation is perfect after seeing r. Therefore, we say this perfect simulation run by anyone is realized by rewinding steps. Note that, this kind of deniability does not hold in concurrent environment due to its rewinding steps. Dwork *et al.* [18] handled this problem by timing assumption.

3) Full Deniability v.s. *Partial Deniability:* The advantage of non-interactivity is reflected in the communication overhead. Non-interactivity handles concurrency problem naturally as it requires 1 round. However, the non-interactive deniable authentication achieves partial deniability only. Obviously, the transcripts in the non-interactive deniable authentication protocol cannot be simulated by anyone otherwise it conflicts the unforgeability. The generic construction of non-interactive deniable authentication is to calculate the authentication tag with the sender's secret and the receiver's public key [32]. In this way, the receiver is assured that a message originated from the sender but cannot prove this to any third party, just like the designated-verifier signature. However, the transcript can only be simulated by using receiver's secret. In other words, it realizes only the partial deniability.

The full deniability states that the conversation transcript can be simulated by anyone not only the receiver. It is proven to be simulatable with challenge-response sub-protocol. Therefore, it must be realized by the interactive steps. The full deniability is more practical in terms of strong privacy compared to partial deniability. Since only the receiver can simulate the transcript in the partially deniable authentication. If the receiver (*i.e.*, service provider) is accepted by the public, it is unfair for the sender (*i.e.*, client). Therefore, we focus on the full deniability in our application.

B. Ring Signature

The ring signature scheme is used to sign a message anonymously. Given a valid ring signature σ w.r.t. a message $\mathtt{m}$ and a set of public keys $\mathcal{PK} = \{\mathtt{PK}_1, \ldots, \mathtt{PK}_\mathtt{n}\}$, any verifier cannot decide which user in the ring $\mathcal{PK}$ is the actual signer even the secret keys of all the users in $\mathcal{PK}$ are exposed.

The syntax of the ring signature is as follows.

Definition 1 (Ring Signature)**.** *A ring signature scheme is a triple of algorithms (*$\mathtt{KGen}, \mathtt{RSig}, \mathtt{RVer}$*). Formally:*

1) *A probabilistic key generation algorithm* $\mathtt{KGen}(1^\kappa)$*. Given the security parameter* κ*, output the keypair* $(\mathtt{PK}_\mathtt{i}, \mathtt{SK}_\mathtt{i})$ *for user* $\mathtt{i}$*. That is* $(\mathtt{PK}_\mathtt{i}, \mathtt{SK}_\mathtt{i}) \leftarrow \mathtt{KGen}(1^\kappa)$.
2) *A probabilistic ring signing algorithm* $\mathtt{RSig}(\mathtt{m}, \mathcal{PK}; \mathtt{SK}_\mathtt{s})$*. Given a message* $\mathtt{m}$*, a ring* $\mathcal{PK}$ *and the private (signing) key* $\mathtt{SK}_\mathtt{s}$ *of the signer* $\mathtt{s}$ *(*$\mathtt{PK}_\mathtt{s} \in \mathcal{PK}$*), output the ring signature* σ*. That is* $\sigma \leftarrow \mathtt{RSig}(\mathtt{m}, \mathcal{PK}; \mathtt{SK}_\mathtt{s})$.
3) *A deterministic verification algorithm* $\mathtt{RVer}(\mathtt{m}, \sigma, \mathcal{PK})$*. Given the ring signature* σ*, the message* $\mathtt{m}$ *with respect to the ring of public keys* $\mathcal{PK}$*, determine whether* σ *is valid w.r.t.* $(\mathtt{m}, \mathcal{PK})$*. That is to check* $\mathtt{RVer}(\mathtt{m}, \sigma, \mathcal{PK}) \stackrel{?}{=} 1$.

The properties of a secure ring signature contain the *Completeness*, *Unconditional Anonymity* and *Unforgeability*. Formally:

Completeness. For any $\{\mathtt{PK}_\mathtt{i}, \mathtt{SK}_\mathtt{i}\}_{i=1}^n$ output by $\mathtt{KGen}(1^\kappa)$, any $\mathtt{s} \in \{1, 2, \cdots, n\}$ and any message $\mathtt{m}$, we have $\mathtt{RVer}(\mathtt{m}, \mathtt{RSig}(\mathtt{m}, \mathcal{PK}; \mathtt{SK}_\mathtt{s}), \mathcal{PK}) = 1$, where $\mathcal{PK} = \{\mathtt{PK}_1, \ldots, \mathtt{PK}_\mathtt{n}\}$.

Unconditional Anonymity. This property states that an attacker is not able to tell which user in a ring $\mathcal{PK}$ produced the signature σ although it obtains the signing keys of all the users in $\mathcal{PK}$. Formally, we consider the anonymity game:

Given a ring signature scheme $(\mathtt{KGen}, \mathtt{RSig}, \mathtt{RVer})$ and an adversary $\mathcal{A}$, the **anonymity game** is as follows:

1) For $\mathtt{i} = 1, 2, \ldots$, generate $(\mathtt{PK}_\mathtt{i}, \mathtt{SK}_\mathtt{i}) \leftarrow \mathtt{KGen}(1^\kappa)$, $\mathcal{A}$ is given $\mathcal{R} = \{\mathtt{PK}_1, \mathtt{PK}_2, \ldots\}$.
2) $\mathcal{A}$ is given access to an oracle $\mathtt{Osign}(s, \mathtt{m}, \mathcal{PK})$ which returns $\mathtt{RSig}(\mathtt{m}, \mathcal{PK}; \mathtt{SK}_\mathtt{s})$. We require $\mathcal{PK} \subseteq \mathcal{R}$ and $\mathtt{PK}_\mathtt{s} \in \mathcal{PK}$.
3) $\mathcal{A}$ is given $\mathtt{SK}_1, \mathtt{SK}_2, \ldots$, $\mathcal{A}$ outputs a message $\mathtt{m}$, distinct indices $\mathtt{s}_\mathtt{o}, \mathtt{s}_\mathtt{1}$, and a ring $\mathcal{PK}$ for $\mathtt{PK}_{\mathtt{s}_0}, \mathtt{PK}_{\mathtt{s}_1} \in \mathcal{PK}$. Furthermore, $\mathcal{A}$ is given $\sigma \leftarrow \mathtt{RSig}(\mathtt{m}, \mathcal{PK}; \mathtt{SK}_{\mathtt{s}_\mathtt{b}})$
4) $\mathcal{A}$ outputs a bit b' and succeeds if $b' = b$.

Definition 2 (Anonymity against full key exposure)**.** *A ring signature scheme is unconditionally anonymous if the success probability of* $\mathcal{A}$ *in the above* ***anonymity game*** *is negligibly close to 1/2.*

Remark 1. *This is a weak definition for the anonymity except that full key exposure is required. Indeed, this model does not consider such attack that the adversary would generate public keys in arbitrary manner (i.e., possibly depend on the public keys of the honest users). Therefore, Bender et al. [34] defined a stronger model, and they considered the adversarially-chosen keys attack in which the adversary must know the actual signer. However, it is not necessary for our scheme to depend on such stronger model since we require the adversary should be unable to prove to a third party the*

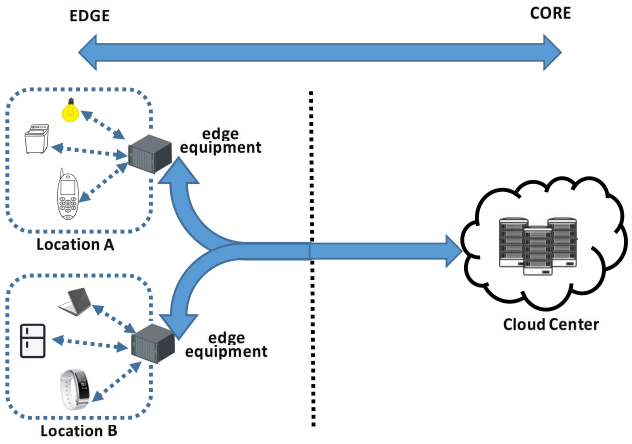

Fig. 2. The Illustration of Edge Computing System

actual signer even though it knows someone. On the other hand, we require the unconditional anonymity w.r.t. full key exposure such that the transcript can be simulated by anyone to reach the full deniability.

Unforgeability. The intuitive notion of unforgeability is that a forger should be unable to output $(\mathcal{PK}, \mathtt{m}, \sigma)$ such that $\mathtt{RVer}(\sigma, \mathtt{m}, \mathcal{PK}) = 1$ where the corresponding signing keys in $\mathcal{PK}$ are unknown to the forger. Formally, we consider the unforgeability game:

Given a ring signature scheme $(\mathtt{KGen}, \mathtt{RSig}, \mathtt{RVer})$ and a forger $\mathcal{F}$, the **unforgeability game** is working as follows:

1) Generate $(\mathtt{PK_i}, \mathtt{SK_i}) \leftarrow \mathtt{KGen}(1^\kappa)$ for $\mathtt{i} = 1, 2, \ldots$, $\mathcal{F}$ is given $\mathcal{R} = \{\mathtt{PK}_1, \mathtt{PK}_2, \ldots\}$.
2) $\mathcal{F}$ is given access to a signing oracle $\mathtt{Osign}(s, \mathtt{m}, \mathcal{PK})$ which returns $\mathtt{RSig}(\mathtt{m}, \mathcal{PK}; \mathtt{SK_s})$. We require $\mathcal{PK} \subseteq \mathcal{R}$ and $\mathtt{PK}_s \in \mathcal{PK}$.
3) $\mathcal{F}$ outputs $(\mathcal{PK}^*, \mathtt{m}^*, \sigma^*)$, succeeds if $\mathcal{PK}^* \subseteq \mathcal{R}$, $\mathtt{RVer}(\sigma^*, \mathtt{m}^*, \mathcal{PK}^*) = 1$ and $\mathcal{F}$ never queried $\mathtt{Osign}(\cdot, \mathtt{m}^*, \mathcal{PK}^*)$.

Definition 3 (Unforgeability)**.** *A ring signature scheme is unforgeable if the success probability of $\mathcal{F}$ in the above **unforgeability game** is negligible.*

IV. Edge Computing Model and Its Security Goals

A. Edge Computing System Overview

The centralization of resources in cloud computing increases the average network latency and jitter. While edge computing immigrates the tasks from the central cloud to the distributed edge equipments to provide convenience for the local IoT devices. As shown in Fig. 2, the edge equipments are deployed close to IoT devices. The edge equipments' location are public and fixed, the communication record between IoT device and edge equipment implies IoT device location. Therefore, the location privacy of IoT devices should be considered in the edge computing.

B. Security Requirement

During the access procedure, there are two aspects should be concerned.

Authentication. Authentication is necessary during the communication between IoT devices and edge equipments, which includes identity authentication and message authentication. When the IoT devices connect to the edge equipments, the identity of IoT device must be confirmed. When the communication occurs between the two entities, the message integrity also should be considered. Clearly, password-based authentications suffers insecurity even though it is efficient and practical. And, key-based authentication avoids non-randomness of the passwords and provides the stronger security. In such kind of authentication, user should prove its knowledge of the public key or knowing the shared secret.

Location Privacy Leakage. As shown in Fig. 2, the edge equipments are close to IoT devices. In addition, the edge equipments are fixed and public, its location implies the location of end devices. Therefore, it leaks the client location inherently since the authentication transcript and the connection record are the evidence to validate the fact of IoT device involvement.

Therefore, a secure communication scheme in the edge computing with location awareness should meet the following security requirements.

- *Authentication*. The edge equipment authenticates the IoT device to identify the client and to assure the message integrity. Formally, a forger $\mathcal{F}$ can query the authentication transcripts for its adaptive chosen messages $\mathtt{m}_1, \mathtt{m}_2, \cdots$ from its challenger. Finally, $\mathcal{F}$ forges an accepted authentication on $(\mathtt{m}^*, \mathtt{PK_i})$ without knowing the secret of $\mathtt{PK_i}$. The success probability of $\mathcal{F}$ in the authentication game is denoted by $\Pr[\mathtt{Succ}^{\mathtt{auth}}_{\mathcal{F}}]$. We require that *authentication* is satisfied if $\Pr[\mathtt{Succ}^{\mathtt{auth}}_{\mathcal{F}}]$ is negligible.
- *Location Privacy*. The client location privacy is preserved in the authentication when it accesses to the fixed edge equipment. It seems paradoxical and challengeable. A feasible way is to make the authentication transcript simulatable. In other words, we require that a distinguisher $\mathcal{D}$'s views in a real conversation transcript $\Gamma^{\mathtt{rea}}$ and a simulated $\Gamma^{\mathtt{sim}}$ are indistinguishable. Therefore, the real client can deny that it communicated with the edge equipment before as the conversation transcript may be simulated by others. Thus, there is no evidence to show IoT device location.

V. Privacy-preserving Authentication Protocol against Location Leakage

We present a simplified privacy-preserving authentication protocol with full deniability against location leakage for edge computing environment in this section. Authentication transcript leaks the IoT device location while it connecting to the fixed edge equipment. Therefore, we adopt the *full deniability* to make it is confident for IoT device to deny the fact of communication record. This kind of authentication does not expose IoT device location even the edge equipment accepts IoT device's communication. We observe that the verifier in this scenario can be assumed honest and thus we

The sender denoted as $\mathtt{P}$ authenticates a message $\mathtt{m}$ to the receiver denoted as $\mathtt{V}$. Let $(\mathtt{SK_P}, \mathtt{PK_P})$ denote the private/public key pair for $\mathtt{P}$ and $(\mathtt{RSig}(\cdot), \mathtt{RVer}(\cdot))$ denote the ring signature scheme.
Round 1. $\mathtt{V} \rightarrow \mathtt{P}$: $\mathtt{V}$ randomly chooses a value $\mathtt{PK}$ and sends $\mathtt{PK}$ to $\mathtt{P}$.
Round 2. $\mathtt{P} \rightarrow \mathtt{V}$: Upon receiving $\mathtt{PK}$, $\mathtt{P}$ generates a 2-user ring signature σ on $\mathtt{m}$, that is $\sigma = \mathtt{RSig}(\mathtt{m}, (\mathtt{PK_P}, \mathtt{PK}); \mathtt{SK_P})$ and sends σ to $\mathtt{V}$.
Finally, $\mathtt{V}$ accepts $\mathtt{P}$'s authentication if σ is a valid ring signature w.r.t. $(\mathtt{m}, \mathtt{PK_P}, \mathtt{PK})$. That is $\mathtt{RVer}(\sigma, \mathtt{m}, \mathtt{PK_P}, \mathtt{PK}) = 1$.

Fig. 3. Our Deniable Authentication Protocol

can simplify and optimize this fully deniable authentication protocol with 2 rounds to adapt to IoT. Besides, the concurrent communication must be considered for Internet-based applications. Under this setting, the interaction executions in the fully deniable authentication protocol can be arbitrarily interleaved by the attacker. Thus the deniability may fail. If the rewinding steps in the traditional deniable authentication protocol are not necessary, the concurrent deniability can be reached naturally. Moreover, we avoid CCA-paradigm for constructing authentication. Therefore, it is practical and suitable for the Internet environment. In this section, we first introduce our generic construction of the underlying deniable authentication protocol and then we instantiate this deniable authentication with a concrete 2-member ring signature scheme to implement a privacy-preserving authentication protocol for edge computing against client location leakage.

A. Deniable Authentication Protocol

Most deniable authentications are constructed by CCA2-secure encryptions as shown in Fig. 1. The underlying building block is impractical. It is more significant to construct the efficient protocols based on primitives with looser requirements. Moreover, the deniability in CCA-paradigm deniable authentications is proven to be black-box simulatable and hence has to add the challenge-response sub-protocol with the secret revocation. It incurs rewinding steps in the simulation. Therefore, the deniability property in CCA-paradigm deniable authentication holds only if copies of the protocol are performed sequentially. It is impractical in the Internet-based service which is under the concurrent environment. Stinson-Wu scheme [23] does not rely on any signatures or encryptions but only to realize identity authentication. Recently, Zeng *et al.* [35] made use of projective hash functions to construct authentication protocol with deniability. In this section, we adopt another building block to avoid CCA-paradigm encryption.

We propose an authentication with full deniability. Different with the traditional approaches (*i.e.*, CCA paradigm) to reach fully deniable authentication as shown in Fig. 1, we construct it by employing the ring signature with 2 members. Our communication round is only 2 which is optimal round in the fully deniable authentication and it reaches the concurrent deniability. Indeed, the simulation in our protocol does not require the rewinding steps and is therefore perfect although there exists adversaries who may schedule the executions or delay messages in arbitrary ways. While most interactive deniable authentications involve rewinding steps to be black-box simulatable, hence the timing assumption is necessary to handle concurrency problem. Our instantiation presented in Section V-B shows it is efficient and practical.

Intuitively, 2-user ring signature scheme solves the authentication and privacy naturally. However, we do not adopt the traditional way (*i.e.*, the 2-user ring containing the sender and the receiver) to construct the ring signature. Otherwise, either the sender or receiver must be convinced to be involved in this conversation due to the publicly verification of ring signature. Our inspiration is that: the one is the sender $\mathtt{P}$'s real public key $\mathtt{PK_P}$, the other one is a logic "public key" $\mathtt{PK}$ which is a random value challenged by the receiver. The sender responses it by generating a ring signature σ with $\mathtt{m}$ and the 2-size ring $\mathcal{PK} = \{\mathtt{PK_P}, \mathtt{PK}\}$. That is $\sigma \leftarrow \mathtt{RSig}(\mathtt{m}, \mathcal{PK}; \mathtt{SK_P})$. The corresponding private key of the logic public key $\mathtt{PK}$ is only known by the receiver. Thus, a valid ring signature implies that the receiver is assured that $\mathtt{m}$ originated from the sender $\mathtt{P}$. The authentication is achieved. On the other hand, the *full* deniability is realized since the authentication tag σ can be simulated by anyone. Indeed, the simulator randomly chooses a value r to simulate the logic public key R which has the same distribution with $\mathtt{PK}$. Note that $\mathtt{PK}$ is just a random value chosen by the receiver $\mathtt{V}$ (not its pubic key) if $\mathtt{V}$ performs protocol honestly. Therefore, the simulated ring signature $\sigma_{\mathtt{sim}} = \mathtt{RSig}(\mathtt{m}, (\mathtt{PK_P}, R); r)$ produced by the "private key" r is indistinguishable from σ due to the unconditional anonymity property of ring signature, see Section III-B. Therefore, the full deniability is achieved without rewinding steps and it can hold in concurrent setting also. Our 2-round deniable authentication is shown in Fig. 3.

We briefly present the authentication and full deniability of our generic 2-round deniable authentication as shown in Fig. 3. The formal proof will be elaborated in the next section.

Our generic 2-round deniable authentication shown in Fig. 3 satisfies authentication property of deniable authentication (DA) protocol. Actually, an adversary $\mathcal{A}$ violates this property if it forges a ring signature σ which passes the verification algorithm $\mathtt{RVer}(\cdot)$. Note that, the ring signature σ is generated by $\mathtt{SK_P}$ or the secret of $\mathtt{PK}$ if the ring signature algorithm is sound. Obviously, $\mathtt{SK_P}$ and the secret of $\mathtt{PK}$ are not known to $\mathcal{A}$. Therefore, our DA protocol meets authentication if the underlying ring signature algorithm is unforgeable.

Our generic 2-round deniable authentication shown in Fig. 3 satisfies full deniability property of DA protocol. This property follows the unconditional anonymity of the underlying ring signature algorithm. Due to the unconditional anonymity of the ring signature scheme $(\mathtt{RSig}(\cdot), \mathtt{RVer}(\cdot))$, the ring signature σ generated by $\mathtt{SK_P}$ has the indistinguishable distribution to that generated by the secret of $\mathtt{PK}$. Since $\mathtt{PK}$ is a random

value chosen by the honest verifier $\mathtt{V}$, anyone can pick random value to generate the same distributed $\mathtt{PK}'$. Obviously, this ring signature σ' produced by the secret of $\mathtt{PK}'$ is valid and has the same distribution as σ. In other words, anyone can simulate a indistinguishable conversation transcript. The full deniability follows.

B. Application to Edge Computing against Location Leakage

We instantiate our deniable authentication described above with a concrete 2-user ring signature scheme and implement it on edge computing environment to preserve client location privacy.

Setup. The edge computing system runs this algorithm to publish the parameters $\mathtt{Para}$ as follows: Choose a safe prime q, let $\mathbb{G}$ and $\mathbb{G}_1$ be two multiplicative cyclic groups of order q that are associated to an efficiently computable bilinear pairing $\hat{e}: \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$. g is the generator of $\mathbb{G}$. Choose a collision-free hash function $\mathtt{H}: \{0,1\}^* \to \mathbb{G}$. $\mathtt{Para} = (q, \mathbb{G}, \mathbb{G}_1, \hat{e}, g, \mathtt{H})$.

KeyGen. The client $\mathtt{U_i}$ (e.g. IoT device) runs this algorithm to generate its keypair $(\mathtt{PK_i}, \mathtt{SK_i})$ as follows: Choose $x_i \leftarrow \mathbb{Z}_q$ and compute $y_i = g^{x_i}$. Set $\mathtt{SK_i} = x_i$ and $\mathtt{PK_i} = y_i$. The public key $\mathtt{PK_i}$ of $\mathtt{U_i}$ is authenticated by the certificate system.

Access Authentication. $\mathtt{U_P}$ submits the connection request to the nearest edge equipment, say EE, with its certificate $\mathtt{Cert_P}$ which includes $\mathtt{PK_P}$. If $\mathtt{Cert_P}$ is valid, EE starts this authentication as follows:

1) EE $\to$ $\mathtt{U_P}$: EE randomly chooses a value $h \leftarrow \mathbb{G}$ and sends h to $\mathtt{U_P}$.
2) $\mathtt{U_P}$ $\to$ EE: Upon receiving h, $\mathtt{U_P}$ generates a 2-user ring signature σ as follows:
   a) Choose $r \leftarrow \mathbb{Z}_q$;
   b) Compute $\mathtt{H}(\mathtt{m})$:
      - If this is an identity authentication, $\mathtt{m} = \mathtt{ID_P} \| \mathtt{PK_P}$ where $\mathtt{ID_P}$ is the identity of $\mathtt{U_P}$.
      - If this is a message authentication, $\mathtt{m}$ is the message delivered from $\mathtt{U_P}$ to EE.
   c) Output $\sigma = \left(h^{x_\mathtt{P}} \cdot \mathtt{H}(\mathtt{m})^r, g^r\right)$.

   σ is the ring signature w.r.t. $(\mathtt{m}, \{\mathtt{PK_P}, h\})$. $\mathtt{U_P}$ sends σ to EE to complete its authentication.

Finally, EE accepts $\mathtt{U_P}$'s authentication if $\sigma = (A, B)$ is a valid ring signature w.r.t. $(\mathtt{m}, \{\mathtt{PK_P}, h\})$. That is to check whether $\hat{e}(y_\mathtt{P}, h) \cdot \hat{e}(B, \mathtt{H}(\mathtt{m})) \stackrel{?}{=} \hat{e}(A, g)$.

VI. Security Analysis and Performance Evaluation

In this section, we analyze the security and performance of our protocol. Since the verification of *Correctness* of our protocol is straightforward, in what follows we will prove that our protocol meets other two properties: *Authentication* and *Location Privacy*, which have been presented in security model. Then we provide the performance evaluation for our instantiation and the comparisons of the underlying deniable authentication to the related constructions regarding to computational cost.

A. Security Analysis

We focus on the security of the instantiation for our deniable authentication protocol. As presented in the security model in Section IV, this privacy-preserving authentication protocol should concern *Authentication* and *Full Deniability* if it is applied to be against location leakage in edge computing environment.

1) Authentication: The communication between the client (IoT device) and the server (edge equipment) should concern the identity authentication and message authentication. This authentication is provided by the soundness (unforgeability) of the underlying deniable authentication protocol. We apply 2-user ring signature scheme to preserve the authentication. Indeed, the generated ring signature σ is bounded to two public keys $\mathtt{PK_P}$ and h. Due to the unforgeability of the ring signature, only the member who knows the secret of $\mathtt{PK_P}$ or h can generate a valid signature. The receiver (i.e., EE) is assured that σ originated from the sender $\mathtt{U_P}$ (who knows the secret of $\mathtt{PK_P}$) since h's secret is unknown to anyone. Intuitively, our protocol meets authentication due to the unforgeability of the underlying ring signature scheme. Theorem 1 formally proves this property.

Theorem 1. *Our protocol described in Section V-B satisfies authentication if Computational Diffie-Hellman (CDH) assumption holds and the hash function* $\mathtt{H}$ *is a random oracle.*

Proof. Suppose $\mathcal{A}$ is the adversary against the soundness of our protocol presented in Section V-B. We construct $\mathcal{A}'$ to break the unforgeability of underlying 2-user ring signature scheme. $\mathcal{C}$ is the challenger of $\mathcal{A}'$ whose goal is to solve CDH problem. $\mathcal{C}$ controls the random oracle. Given a CDH problem instance (g, g^a, g^b) over the pairing group $(\mathbb{G}, \mathbb{G}_1, g, q, \hat{e})$, $\mathcal{C}$ runs $\mathcal{A}$ and $\mathcal{A}'$ as follows.

$\mathcal{C}$ sets the sender public key as $\mathtt{PK_P} = g^a$. When $\mathcal{A}$ queries authentication to arbitrary messages $\mathtt{m}_1, \mathtt{m}_2, \ldots$ adaptively, $\mathcal{A}'$ acts as sender $\mathtt{P}$ and $\mathcal{A}$ acts as receiver $\mathtt{V}$ by given g^b. $\mathcal{A}'$ performs this authentications to $\mathcal{A}$ as follows.

When $\mathtt{m_i}$ is queried by $\mathcal{A}$ by sending $h = g^b$ to $\mathcal{A}'$ according to our protocol, $\mathcal{A}'$ makes hash query $\mathtt{H}$-query $\mathtt{H}(\mathtt{m_i})$ to $\mathcal{C}$. Note that, before any hash queries are made, $\mathcal{C}$ chooses $i^* \in [1, q_H]$ where q_H is the number of hash queries. If $\mathtt{m_i}$ is already in $\mathcal{C}$'s hash list (which is empty at the beginning), $\mathcal{C}$ responds following its hash list. Otherwise, $\mathcal{C}$ chooses $w_i \leftarrow \mathbb{Z}_q$ randomly and sets $\mathtt{H}(\mathtt{m_i}) = g^{w_i}$ if $i = i^*$ and sets $\mathtt{H}(\mathtt{m_i}) = g^{b+w_i}$ if $i \neq i^*$. Then $\mathcal{A}'$ makes ring signature queries to $\mathcal{C}$ on $\mathtt{m_i}$. If $i = i^*$, $\mathcal{C}$ aborts this query. Otherwise, $\mathcal{C}$ chooses $r_i' \leftarrow \mathbb{Z}_q$ randomly and returns $\sigma = \left((g^a)^{-w_i} \mathtt{H}(\mathtt{m_i})^{r_i'}, (g^a)^{-1} g^{r_i'}\right)$ to $\mathcal{A}'$. Finally, $\mathcal{A}'$ responds σ to $\mathcal{A}$ to complete its authentication to $\mathtt{m_i}$. Obviously, the simulation of $\mathcal{A}'$ is perfect and $\mathcal{A}$ accepts $\mathcal{A}'$'s authentication since the returned $\sigma = (A, B)$ satisfies the verification $\hat{e}(y_\mathtt{P}, h) \cdot \hat{e}(B, \mathtt{H}(\mathtt{m})) = \hat{e}(A, g)$, where $\mathtt{H}(\mathtt{m}) = g^{b+w_i}$.

$\mathcal{A}$ pretends to $\mathtt{P}$ to make fake authentication to $\mathtt{m}^*$ after its authentication queries to $\mathtt{m}_1, \mathtt{m}_2, \ldots$. Obviously, $\mathcal{A}$ succeeds if and only if its production $\sigma_{\mathtt{m}^*} = (A^*, B^*)$ satisfies the verification equation $\hat{e}(y_\mathtt{P}, h) \cdot \hat{e}(B^*, \mathtt{H}(\mathtt{m}^*)) = \hat{e}(A^*, g)$. It helps $\mathcal{A}'$ to break the unforgeability of the underlying ring

signature undoubtedly. If m^* is the i^*-th queried message in the hash list, it helps $\mathcal{C}$ to solve CDH problem indeed. In this case, $\mathtt{H}(\mathtt{m}^*) = g^{w_{i^*}}$ and $(A^*, B^*) = \left(g^{ab}(g^{w_{i^*}})^r, g^r\right)$. Therefore, $\mathcal{C}$ obtains g^{ab} by $A^*/(B^*)^{w_{i^*}}$.

Therefore, we have that if $\mathcal{A}$ breaks the authentication property of our protocol with probability ε after making q_H queries to the random oracle, $\mathcal{C}$ solves the CDH problem with ε/q_H. □

2) Location Privacy: In the edge computing environment, the location of edge equipment reveals client location during the connection and communication. Our protocol makes use of "deniability" to handle this problem. If the IoT device can deny its involvement in this authentication, there is no evidence for its connection. The location privacy is preserved naturally. Thus, the deniability of our protocol indicates location privacy of IoT devices.

Following the security model of deniable authentication protocol described in Section III, we use *simulation* fashion to prove the deniability of our protocol. If a simulator (run by anyone) can simulate the authentication transcript without participant's secret, the full deniability is satisfied. The formal proof is presented as follows.

Theorem 2. *Our protocol described in Section V-B satisfies the* full deniability *if the receiver follows our protocol honestly.*

Proof. For the underlying ring signature algorithm, we argue it is unconditionally anonymous against full key exposure. Indeed, the underlying ring signature algorithm outputs the signature $\sigma = (A, B) = (h^{x_{\mathtt{P}}}\mathtt{H}(\mathtt{m})^r, g^r)$, which can be rewritten as $(y_{\mathtt{P}}^{\omega}\mathtt{H}(\mathtt{m})^r, g^r)$ where $\omega = \log_g h$.

Note that, the value $h \in \mathbb{G}$ is randomly chosen by verifier $\mathtt{V}$ (EE) in our protocol. Therefore, we require $\mathtt{V}$ honestly to return a random value from $\mathbb{G}$ in the first flow. In the simulation, the simulator $\mathcal{S}$ chooses $\bar{\omega} \leftarrow \mathbb{Z}_q$ randomly to simulate h, i.e., $\bar{h} = g^{\bar{\omega}}$. Obviously, $\bar{h}$ has the same distribution as h and the simulated h is perfect. With the secret $\bar{\omega}$, the simulator $\mathcal{S}$ generates $\bar{\sigma} = (\bar{A}, \bar{B}) = (y_{\mathtt{P}}^{\bar{\omega}}\mathtt{H}(\mathtt{m})^{\bar{r}}, g^{\bar{r}})$. Obviously, $\bar{\sigma}$ is "identical" to σ as $\hat{e}(y_{\mathtt{P}}, \bar{h}) \cdot \hat{e}(\bar{B}, \mathtt{H}(\mathtt{m})) = \hat{e}(\bar{A}, g)$ holds.

Clearly, this simulated transcript $(\bar{h}, \bar{\sigma})$ (produced by anyone) is indistinguishable from the real one. Therefore, the actual sender $\mathtt{U}_{\mathtt{P}}$ can fully deny its involvement as this authentication transcript for connection to EE may be "fabricated" by anyone and the location privacy is preserved. □

B. Performance Evaluation

We make use of deniable authentication (DA) protocol to preserve authentication and privacy while IoT devices accessing and communicating with the edge equipment. The performance of the underlying DA protocol mainly affects the efficiency of our protocol in Section V-B. Therefore, we analyze the performance of our work from two sides: we first analyze our underlying DA protocol theoretically and give the performance comparisons among related constructions; then we implement our protocol on the specific edge computing environment to show its efficiency.

1) Efficiency of Underlying DA Protocol: Our underlying DA protocol as described in Fig. 3 employs 2-user ring signature algorithm and challenge-response phase to realize authentication and full deniability. With this design and under the assumption of honest verifier, our DA protocol is only 2 rounds, which is the optimal communication round in the fully deniable authentication protocols. Besides that, our DA protocol realizes concurrent deniability even. Since we do not require "rewinding" steps to simulate the authentication transcript, the copies of the protocol are not necessarily performed sequentially. Therefore, our DA protocol with both optimal round and concurrent denibaility can be applied to Internet-based service practically.

There are some other related DA protocols to be compared in this qualitative research. Dwork's scheme [18] is constructed by encryptions and the underlying encryption algorithm is required to be CCA2 secure. Their work requires 4 rounds to realize full deniability and the deniability does not hold in concurrent setting. From authentication type point of view, this work supports both message authentication and identity authentication. Jiang's scheme [24] relies on public random oracle to realize 3-round mutual authentication (and 2 rounds if one-way authentication). The communication transcript is simulated without rewinding steps, thus it realizes concurrent deniability. Yao *et al.*'s work [25] depends on non-malleable zero-knowledge proofs, thus the underlying communication round is heavy. The communication round of their scheme is 2+ means that the round of one-way authentication is 2 whereas its NMZK may incur extra communication round. Stinson-Wu scheme [23] depends on KEA assumption to realize a 2-round full deniable identification protocol. It does not rely on any underlying signatures or encryptions. However, KEA is not the standard assumption. The above three works reach identity authentication only. Li *et al*'s protocols [32] are 1 round which seem communication optimal. However, it achieves partial deniability only. In order to show these protocols clearly, we conclude their features in Table I and the communication round is in one-way authentication fashion.

2) Efficiency of Our Instantiation: Qualitatively, Table I reports that our scheme has superior properties in both communication and security. Our communication round is optimal for the full deniability, therefore we have lower latency undoubtedly. In this subsection, we consider the computation performance quantitatively in the specific edge computing environment. The efficient non-interactive deniable authentication protocols from [32] are chosen as the comparison objects. Although their protocols are 1 round, their deniability is weak. Our experiment shows the computational cost of these protocols.

The end devices equipped with processor of Intel(R) Pentium(R) CPU G4500 3.50GHz and RAM of 4.00GB are used to simulate the IoT devices. We choose 80-bit level to re-implement their protocols and we choose Type A pairing from free C library PBC following their implementation. Fig. 4 reports the experimental results.

We run our protocol, HDA-1 and HDA-2 in [32] 20 times to get 60 records totally. The average cost is also calculated and we make 3 lines of every 21 records from each protocols. And

TABLE I
COMPARISONS OF DENIABLE AUTHENTICATION PROTOCOLS

Scheme	Deniability	Round	Concurrency	pRO	Assumption	CCA-paradigm	Authentication Type
Dwork's scheme [18]	Full	4	×	-	Standard	Yes	Message & Identity
Jiang's scheme [24]	Full	2	✓	✓	Standard	No	Identity
Yao's scheme [25]	Full	2+	✓	-	Standard	No	Identity
Stinson-Wu scheme [23]	Full	2	✓	-	KEA	No	Identity
Li's scheme [32]	Partial	1	✓	-	Standard	No	Message & Identity
Our scheme	Full	2	✓	-	Standard	No	Message & Identity

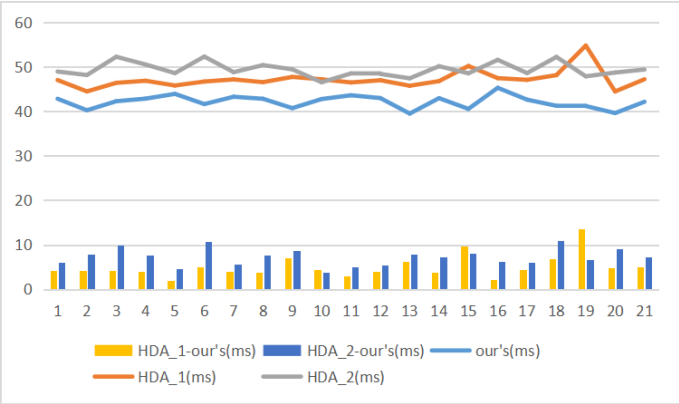

Fig. 4. Experimental Time Cost

TABLE II
EXPERIMENTAL RESULT ANALYSIS

Protocol	Level	Cost	Difference	Rate
Ours	80-bit	42.15335ms	-	-
HDA-1	80-bit	47.2193ms	5.06595	10.728%
HDA-2	80-bit	49.41545ms	7.2621	14.696%

under the 3 lines, we compute 42 records of difference cost, difference between HDA-1 and our protocol is yellow bar, difference between HDA-2 and our protocols is blue bar. These bars graphically illusrate the cost saving between different protocols.

Table II shows our experimental result with average cost. We find that our protocol saves 10.728% cost comparing with HDA-1 and 14.728% cost comparing to HDA-2.

VII. CONCLUSION

Authentication in edge computing environment incurs location leakage, we propose a privacy-preserving authentication scheme with full deniability to protect the location of IoT devices. Our underlying protocol has only 2 communication rounds which achieves the optimal communication latency for the full deniability. Our scheme does not rely on CCA-paradigm encryptions, rewinding steps and any strong number-theoretical assumptions, thus it is practical in the concurrent Internet-based environment. Compared to the existing efficient deniable authentications, our scheme has better performance for end devices in terms of computational cost. Therefore, it adapts to the resource-constrained IoT devices.

ACKNOWLEDGEMENT

This work is supported by the Ministry of Education "chunhui plan" (Z2016150), the National Natural Science Foundation of China (61872087) and the European Unions Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie (840922).

REFERENCES

[1] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying location privacy," in *2011 IEEE symposium on security and privacy*. IEEE, 2011, pp. 247–262.
[2] X. He, R. Jin, and H. Dai, "Leveraging spatial diversity for privacy-aware location-based services in mobile networks," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 6, pp. 1524–1534, 2018.
[3] N. Wang, J. Fu, J. Li, and B. K. Bhargava, "Source-location privacy protection based on anonymity cloud in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 100–114, 2019.
[4] C. Ma, Z. Yan, and C. W. Chen, "Sspa-lbs: Scalable and social-friendly privacy-aware location-based services," *IEEE Transactions on Multimedia*, vol. 21, no. 8, pp. 2146–2156, 2019.
[5] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *Journal of computer security*, vol. 15, no. 1, pp. 39–68, 2007.
[6] C.-Y. Chow, M. F. Mokbel, and X. Liu, "Spatial cloaking for anonymous location-based services in mobile peer-to-peer environments," *GeoInformatica*, vol. 15, no. 2, pp. 351–380, 2011.
[7] A. R. Beresford and F. Stajano, "Mix zones: User privacy in location-aware services," in *IEEE Annual Conference on Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second*. IEEE, 2004, pp. 127–131.
[8] B. Palanisamy and L. Liu, "Effective mix-zone anonymization techniques for mobile travelers," *Geoinformatica*, vol. 18, no. 1, pp. 135–164, 2014.
[9] L. Sweeney, "K-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.
[10] S. Zhang, X. Li, Z. Tan, T. Peng, and G. Wang, "A caching and spatial k-anonymity driven privacy enhancement scheme in continuous location-based services," *Future Generation Computer Systems*, vol. 94, pp. 40–50, 2019.
[11] T. Shu, Y. Chen, and J. Yang, "Protecting multi-lateral localization privacy in pervasive environments," *IEEE/ACM Transactions on Networking*, vol. 23, no. 5, pp. 1688–1701, 2015.
[12] S. Li, H. Li, and L. Sun, "Privacy-preserving crowdsourced site survey in wifi fingerprint-based localization," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, p. 123, 2016.
[13] Z. Yang and K. Järvinen, "The death and rebirth of privacy-preserving wifi fingerprint localization with paillier encryption," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 1223–1231.
[14] J. Han, H. Wang, Z. Zheng, and Q. Xu, "Privacy preserved wireless sensor location protocols based on mobile edge computing," *Computers & Security*, vol. 84, pp. 393–401, 2019.
[15] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential privacy for location-based systems," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, 2013, pp. 901–914.

[16] T. Li, J. Li, Z. Liu, P. Li, and C. Jia, "Differentially private naive bayes learning over multiple data sources," *Information Sciences*, vol. 444, pp. 89–104, 2018.
[17] Y. Qian, Y. Jiang, M. S. Hossain, L. Hu, M. Ghulam, and S. U. Amin, "Privacy-preserving based task allocation with mobile edge clouds," *Informaiton Sciences*, vol. 507, pp. 288–297, 2020.
[18] C. Dwork, M. Naor, and A. Sahai, "Concurrent zero-knowledge," *Journal of the ACM (JACM)*, vol. 51, no. 6, pp. 851–898, 2004.
[19] S. Zeng, Y. Chen, S. Tan, and M. He, "Concurrently deniable ring authentication and its application to lbs in vanets," *Peer-to-Peer Networking and Applications*, vol. 10, no. 4, pp. 844–856, 2017.
[20] D. Dolev, C. Dwork, and M. Naor, "Non-malleable cryptography," *SIAM review*, vol. 45, no. 4, pp. 727–784, 2003.
[21] M. Di Raimondo and R. Gennaro, "New approaches for deniable authentication," *Journal of cryptology*, vol. 22, no. 4, pp. 572–615, 2009.
[22] M. Bellare and A. Palacio, "Towards plaintext-aware public-key encryption without random oracles," in *International Conference on the Theory and Application of Cryptology and Information Security*. Springer, 2004, pp. 48–62.
[23] D. R. Stinson and J. Wu, "An efficient and secure two-flow zero-knowledge identification protocol," *J. Math. Crypt*, vol. 1, no. 2007, pp. 201–220, 2007.
[24] S. Jiang and R. Safavi-Naini, "An efficient deniable key exchange protocol (extended abstract)," in *Financial Cryptography and Data Security, 12th International Conference, FC 2008, Cozumel, Mexico, January 28-31, 2008, Revised Selected Papers*, ser. Lecture Notes in Computer Science, vol. 5143. Springer, 2008, pp. 47–52.
[25] A. C.-C. Yao and Y. Zhao, "Privacy-preserving authenticated key-exchange over internet," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 1, pp. 125–140, 2013.
[26] H. Tian, X. Chen, and W. Susilo, "Deniability and forward secrecy of one-round authenticated key exchange," *The Journal of Supercomputing*, vol. 67, no. 3, pp. 671–690, 2014.
[27] S. Jiang, "Timed encryption with application to deniable key exchange," *Theoretical Computer Science*, vol. 560, pp. 172–189, 2014.
[28] S. Jiang and R. Safavi-Naini, "An efficient deniable key exchange protocol," in *International Conference on Financial Cryptography and Data Security*. Springer, 2008, pp. 47–52.
[29] S. Zeng and Y. Chen, "Concurrently deniable group key agreement and its application to privacy-preserving vanets," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.
[30] L. Harn and J. Ren, "Design of fully deniable authentication service for e-mail applications," *IEEE Communications letters*, vol. 12, no. 3, pp. 219–221, 2008.
[31] F. Li, D. Zhong, and T. Takagi, "Efficient deniably authenticated encryption and its application to e-mail," *IEEE Trans. Information Forensics and Security*, vol. 11, no. 11, pp. 2477–2486, 2016.
[32] F. Li, J. Hong, and A. A. Omala, "Practical deniable authentication for pervasive computing environments," *Wireless Networks*, vol. 24, no. 1, pp. 139–149, 2018.
[33] S. Zeng, Y. Mu, M. He, and Y. Chen, "New approach for privacy-aware location-based service communications," *Wireless Personal Communications*, vol. 101, no. 2, pp. 1057–1073, 2018.
[34] A. Bender, J. Katz, and R. Morselli, "Ring signatures: Stronger definitions, and constructions without random oracles," *Journal of Cryptology*, vol. 22, no. 1, pp. 114–138, 2009.
[35] S. Zeng, Z. H. Mu, Yi, and M. He, "A practical and communication-efficient deniable authentication with source-hiding and its application on wi-fi privacy," *Informaiton Sciences*, vol. 516, pp. 331–345, 2020.