JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 39, 507-524 (2023)
DOI: 10.6688/JISE.202305_39(3).0004

Universally Secure Device-to-Device Communications With Privacy Protection and Fine-Grained Access Control Based on 5G-Enabled Multi-Access Edge Computing

RUEI-HAU HSU, LU-CHIN WANG AND HSIANG-SHIAN FAN
Department of Computer Science and Engineering
National Sun Yat-sen University
Kaohsiung, 804 Taiwan
E-mail: rhhsu@mail.cse.nsysu.edu.tw; {richardwang1134; roi.ss.fan}@gmail.com

Device-to-device (D2D) communications enable new user experiences and low latency in communications among devices for new IoT applications, such as augmented reality (AR), virtual reality (VR), public safety, based on the fifth-generation and beyond (B5G) mobile networks. However, typical D2D communications still rely on the assistance of a centralized component, *i.e.*, proximity service (ProSe) application server, for access control during device discovery procedures in mobile networks. Moreover, D2D communications are mainly launched by certain apps running on user equipment (UE) and need to discover the other UE in the same proximity of a base station (*i.e.*, gNB in 5G) according to the identity or the profile of each UE in an app. This procedure will inevitably disclose the user/application's sensitive information and behaviors to the infrastructures above to assist in establishing the corresponding D2D communications. Moreover, most of related works for secure D2D communication cannot support fine-grained access control and hidden policy during device discovery procedure simultaneously. Thus, this work proposes a new multi-access edge computing (MEC) based secure anonymous D2D communications system, so-called SAD2D, based on our newly proposed cooperative anonymous attribute-based encryption (CoAABE). The security proofs of the proposed fundamental CoAABE scheme and the SAD2D protocol are both provably secure. Additionally, this work conducts the performance evaluation for the SAD2D in the aspect of queueing model, which can reflect the effects of device discovery requests in certain arrival rates regarding the performance. Overall, this work paves the way to achieve fine-grain access controllable security and privacy protection simultaneously for secure D2D communications to B5G MEC-enabled IoT applications.

Keywords: device-to-device communications, proximity service, privacy, multi-access edge computing, 5G, fine-grained access control, attribute based encryption, hidden policy

1. INTRODUCTION

Network-assisted device to device (D2D) communication architecture [1] is a key technology in the future mobile networks, especially the fifth-generation mobile networks and beyond (B5G). Its benefits include traffic offloading, reduced latency, and energy saving [2]. In general network-assisted D2D architecture, the telecommunication service provider (TSP) provides radio resource and session management for quality of ser-

Received January 7, 2022; revised February 10 & March 10, 2022; accepted April 3, 2022.
Communicated by Po-Wen Chi.

vice (QoS), and authentication and key management for security [3–5]. The supports of QoS and security can simplify the procedure of establishing D2D communications and reduce the costs. However, this may result in security and privacy concerns when the infrastructures of TSP are regarded as honest-but-curious ones. The most recent related works for secure D2D communications have considered to support the security features of entity authentication, secure key exchange for the confidentiality of communications, user anonymity for identity protection against location tracking and user behavior analysis, and group-based authentication and key exchange to facilitate the security of multicasting D2D communications [6, 7].

In order to identify the profile of each device during device discovery for D2D communications more precise, there are also some of the works [8,9], considering fine grained access control additionally to recognize the profile, which consisting of several types of attributes, of each user equipment (UE) in advance of the establishment of D2D communications. The fine-grained access control feature can further facilitate the applications that need to identify the UE of each user for more detailed user profile and respond to each D2D communication request rapidly. The above works adopt attribute-based encryption ABE to support secure D2D communications with fine-grained access control. However, the use of ABE may incur additional security issue, where the discovery of devices by ABE can be launched without any limitation as long as the public key of an ABE system is available since a ciphertext of any policy can be produced by using the public key only. Thus, Hsu *et al*. propose a secure network-controllable D2D communication system with fine-grained access control, so-called SGD^2 protocol [10] based on a proposed cooperative attribute-based encryption scheme [11]. Nonetheless, a new security concern is arisen that the policy of each device discovery may be exposed in the above works.

In order to hide policy for device discovery in D2D communications, this work proposes a new design of CoABE by referring the concept of supporting hidden policy in NYO-ABE [12]. The access structure in our proposed CoABE scheme is formulated as several multiple-choice problem, where "partial hidden" to the policy means that the type of each multiple-choice problem is public, but the options of the problem are hidden. Because the fully hidden policy attribute-based encryption methods have restrictions on the flexibility of a policy, our design takes into account both the flexibility and the privacy additionally.

Thus, this work aims at the design of the new cooperative anonymous attribute-based encryption, so-called CoAABE, to support control capability to encryption and anonymity to policy and attributes during encryption/decryption procedures. Based on CoAABE, this work proposes the new design of secure D2D communications to fulfill the security requirements, where not only entity authentication and key exchange security are supported, but also user anonymity, controllable D2D communications, fine-grained access control, and hidden policy. Overall, this work has the following contributions.

1.1 Contributions

1) This work proposes a new cooperative anonymous attribute based encryption, *i.e*., CoAABE, for the fundamental of the proposed secure D2D communications.

2) This work constructs a secure D2D communication scheme that allows UE to dis-

cover the other UEs, which possess the attributes that satisfy the combination of logical predicates. The above device discovery procedure achieves authentication and key exchange security.

3) The device discovery procedure of the proposed secure D2D communication system achieves the privacy protection on user attributes and discovering policy against outsiders and honest-but-curious mobile network operators, including the components involving D2D communications in mobile networks.

4) This paper provides the security proof of the proposed CoAABE and secure D2D communication system under the security definitions capturing attackers' abilities.

5) This work evaluates the performance of this work with the other related works by the implemented prototype system.

1.2 Organization

The organization of this work for the remaining section is shown as follows. Section 2 introduces the background knowledge for the required preliminaries for the proposed CoAABE and SAD2D based on CoAABE. Section 3 introduces the proposed CoAABE scheme and its security proof formally. Section 4 shows the system and the adversary models for the proposed SAD2D protocol. Section 5 introduces the proposed SAD2D protocols. Section 6 evaluates the performance of SAD2D according to the conducted experimental results and compares the security features of SAD2D with the other related works. Section 7 proves the security proof of the proposed SAD2D. Section 8 concludes this work.

2. PRELIMINARIES

2.1 Linear Encryption

A KGC generates a pulic key $pk = \{u, v, h\} \in \mathbb{G}^3$ and a secret key $sk = \{x, y\} \in \mathbb{Z}_p^{*2}$, such that $u^x = v^y = h$. An encryptor first selects two random numbers $\{a, b\} \in \mathbb{Z}_p^{*2}$, select a message M, then compute the ciphertext $C = [\tilde{C}, e_1, e_2] = [M \cdot h^{a+b}, u^a, v^b]$. The decryptor can decrypt the ciphertext $\tilde{C}$ by computes $M = \tilde{C} \cdot (e_1^x \cdot e_2^y)^{-1}$.

2.2 Bilinear Pairing

Let $\mathbb{G}$ be a multiplicative cyclic group. Bilinear pairing refers to an efficient mapping function $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, that can map two elements of $\mathbb{G}$ to another group multiplicative cyclic group, $\mathbb{G}_T$. This function has an important characteristic, which will be used in this paper.

- Let $\{a, b\} \in \mathbb{Z}_p^2$ and $g \in \mathbb{G}$, $e(g^a, g^b) = e(g, g)^{ab}$

3. PROPOSED COAABE FOR SAD2D

3.1 Intuition

The proposed CoAABE is to provide the control on encryption, where only the encryptor, who has the *permission* consisting of granted *attributes* that satisfy the *policy* used for an encryption, can produce a complete ciphertext. Only a complete ciphertext can be decrypted successfully when one decrypts it with the private key of certain attributes that can satisfy the policy of the encryption. In order to do so, the design of CoAABE divides the encryption function into two encryption functions, *i.e.*, pre-encryption function (**PreEnc** in CoAABE) and cooperative encryption function (**CoEnc** in CoAABE). The policy for an encryption is taken as an input for pre-encryption by the encryptor and the set of the granted attributes of each encryptor is taken as an input of cooperative encryption for the *permission* of each encryption. If the permission matches the policy, the output of the cooperative encryption is regarded as a complete ciphertext. Otherwise, the output becomes an invalid ciphertext, which cannot be decrypted correctly. The pre-encryption is conducted by an encryptor and the cooperative encryption is conducted by a cooperator/controller, who will check if the encryptor is permitted for the encryption of a given policy. Based on this design, one can encrypt a message as a complete ciphertext, which can decrypted correctly, in case that the attributes embedded in the private key can satisfy the policy of the encryption.

3.2 Access Structure

CoAABE is a kind of ciphertext-policy attribute-based encryption (CP-ABE). The encryptor selects a policy for encryption to generate the ciphertext. Since CoAABE has an encryption control function, the encryptor's permission is also needed to be defined. The definition of the access structures in the proposed CoAABE is shown as follows:

3.2.1 Attribute type and value

The access structure of CoAABE is similar to a set of multiple-choice questions. An attribute type is like one of the questions in the multiple-choice question set, and an attribute value is like one of the options of a multiple-choice question. The attribute value and the attribute type are abstract concepts and can be in any form in practice, such as strings or integers.

3.2.2 Universal attribute set

A universal attribute set is represented as $\mathbb{A} = \{\mathbb{A}_i\}_{1\leq i\leq n} = \{\{\alpha_{i,t}\}_{1\leq t\leq n_i}\}_{1\leq i\leq n}$, where i is the index of an attribute type, t is the index of attribute value, n_i is the number of the attribute values for the ith attribute type, n is the number of attribute types, $\mathbb{A}_i$ is the sub universal attribute set of ith attribute type and $\alpha_{i,t}$ is the tth attribute value of the ith attribute type.

3.2.3 Permission

A permission is represented as $\mathbb{P} = \{\mathbb{P}_i \subseteq \mathbb{A}_i, \mathbb{P}_i \neq \phi\}_{1\leq i\leq n}$, where i is the index of an attribute type, n is the number of attribute types, $\mathbb{P}_i$ is the sub permission of ith attribute type. Each $\mathbb{P}_i$ should be a subset of $\mathbb{A}_i$ and should not be an empty set.

3.2.4 Policy

A policy is represented as $\mathbb{W} = \{\mathbb{W}_i \subseteq \mathbb{A}_i, \mathbb{W}_i \neq \phi\}_{1 \leq i \leq n}$, where i is the index of an attribute type, n is the number of attribute types, $\mathbb{W}_i$ is the sub policy of ith attribute type. Each $\mathbb{W}_i$ should be a subset of $\mathbb{A}_i$ and should not be an empty set. In this study, the relationship between a permission and a policy is as follows: $\mathbb{W} \models \mathbb{P} \Leftrightarrow \{\mathbb{W}_i \subseteq \mathbb{P}_i\}_{1 \leq i \leq n}$, where "$\models$" means "satisfy" and the formula means that the policy satisfies the permission if and only if all $\mathbb{W}_i$ are the subset of $\mathbb{P}_i$ for $1 \leq i \leq n$.

3.2.5 User attribute list

A user attribute set is represented as $\mathbb{L} = \{\mathbb{L}_i \in \mathbb{A}_i\}_{1 \leq i \leq n}$, where i is the index of an attribute type, n is the number of attribute types, $\mathbb{L}_i$ is the user attribute of ith attribute type. Each $\mathbb{L}_i$ should be an element of $\mathbb{A}_i$. In this study, the relationship between a universal attribute set and a policy is as follows: $\mathbb{L} \models \mathbb{W} \Leftrightarrow \{\mathbb{L}_i \in \mathbb{W}_i\}_{1 \leq i \leq n}$. The formula means that the user attribute list satisfies the policy if and only if all $\mathbb{L}_i$ are in $\mathbb{W}_i$ while $1 \leq i \leq n$.

Table 1. Notations of CoAABE.

Notation	Meaning
$\mathbb{A}$	universal attribute set
$\mathbb{A}_i$	sub-universal attribute set
$\alpha_{i,t}$	the tth attribute value of ith attribute type
$\mathbb{P}$	universal permission set
$\mathbb{P}_i$	sub-permission of ith attribute type
$\mathbb{W}$	universal policy set
$\mathbb{W}_i$	sub-policy of ith attribute type
$\mathbb{L}$	universal user attribute set
$\mathbb{L}_i$	the user attribute of ith attribute type
$\{MSK, MPK\}$	master secret key and master public key
CSK	cooperative secret key
$SK_{\mathbb{L}}$	user secret key with attribute set $\mathbb{L}$
M	a message to be encrypted
$PC_{\mathbb{W}}$	partial ciphertext with a given policy $\mathbb{W}$
$CC_{\mathbb{P}}^{\mathbb{W}}$	complete ciphertext with a given policy $\mathbb{W}$ and permission $\mathbb{P}$

3.3 Construction

- **Setup**$(1^k, \mathbb{A})$

  This function takes the security parameter 1^k to generate following bilinear group parameters $\mathbb{G}, \mathbb{G}_T, e, g \in \mathbb{G}$, and random $\{x, y, a, b, \delta, r_c, w\} \in \mathbb{Z}_p^{*7}$. It then computes x^{-1}, y^{-1}, $u = g^{x^{-1}}$, and $v = g^{y^{-1}}$, such that $u^x = v^y = g$. Next, for each attribute value $\{\{\alpha_{i,t}\}_{1 \leq t \leq n_i}\}_{1 \leq i \leq n}$, this function randomly generates $\{\{a_{i,t}, b_{i,t} \in \mathbb{Z}_p^{*2}, A_{i,t} \in \mathbb{G}\}_{1 \leq t \leq n_i}\}_{1 \leq i \leq n}$. Then, this function computes $Y = e(g,g)^w, U = u^a$, $V = v^b, \Delta = g^{\delta}$. In summary, this function outputs the master public key, MPK, the master secret key, MSK, and the cooperative secret key, CSK as follows:

$MPK = \{U, V, \Delta,\ Y, p, \mathbb{G}, \mathbb{G}_T, g, e, \{\{A_{i,t}^{a_{i,t}}, A_{i,t}^{b_{i,t}}\}_{1\le t\le n_i}\}_{1\le i\le n}\}$, $MSK = \{a, b, \delta, w, \{\{a_{i,t}, b_{i,t}\}_{1\le t\le n_i}\}_{1\le i\le n}\}$, and $CSK = \{x, y\}$.

- **KeyGen**$(MSK, \mathbb{L})$

  The attribute list $\mathbb{L} = \{\mathbb{L}_i \in \mathbb{A}_i\}_{1\le i\le n}$, it means that each $\mathbb{L}_i$ is an attribute value α_{i,t_i} under attribute type i. For each attribute $\mathbb{L}_i$, this function generates random $\{\{s_i, \lambda_i\} \in \mathbb{Z}_p^{*2}\}_{1\le i\le n}$. Next, this function computes $s = \sum_{i=1}^{n} s_i$, $D_0 = g^{w-s}$, $D_e = g^{\delta(a+b)}$. Then, for each $\mathbb{L}_i$, this function generates $\{\{D_{i,0}, D_{i,1}, D_{i,2}\} = \{g^{s_i} A_{i,t_i}^{a_{i,t_i} b_{i,t_i} \lambda_{i,t_i}}, g^{a_{i,t_i}\lambda_{i,t_i}}, g^{b_{i,t_i}\lambda_{i,t_i}}\}\}$ for $1 \le i \le n$. Finally, this function outputs the secret key, $SK_{\mathbb{L}} = \{D_0, D_e, \{D_{i,0}, D_{i,1}, D_{i,2}\}_{1\le i\le n}\}$.

- **PreEnc**$(MPK, \mathbb{W}, M)$

  This function first takes a message $M \in \mathbb{G}_T$ and a policy $\mathbb{W} = \{\mathbb{W}_i \subseteq \mathbb{A}_i\}_{1\le i\le n}$ as input. Next, the function randomly select $r_{i,t} \in \mathbb{Z}_p^*$ for all attribute value $\{\{\alpha_{i,t}\}_{1\le t\le n_i} \in \mathbb{W}_i\}_{1\le i\le n}$ and $r \in \mathbb{Z}_p^*$, and computes $\tilde{C} = M \cdot Y^r, C_0 = g^r, e_1 = U^r, e_2 = V^r$. Finally, for all $\{\{\alpha_{i,t}\}_{1\le t\le n_i} \in \mathbb{W}_i\}_{1\le i\le n}$, this function computes $\{C_{i,t,1}, C_{i,t,2}\} = \{(A_{i,t}^{b_{i,t}})^{r_{i,t}}, (A_{i,t}^{a_{i,t}})^{r-r_{i,t}}\}$, and for all $\{\{\alpha_{i,t}\}_{1\le t\le n_i} \notin \mathbb{W}_i\}_{1\le i\le n}$, this function generate random $\{C_{i,t,1}, C_{i,t,2}\}$. Then, this function outputs the partial ciphertext, $PC_{\mathbb{W}} = \{\widetilde{C}, C_0, e_1, e_2, \{\{C_{i,t,1}, C_{i,t,2}\}_{1\le t\le n_i}\}_{1\le i\le n}\}$,

- **CoEnc**$(PC_{\mathbb{W}}, CSK, MPK, \mathbb{P})$

  This function generates random $\gamma \in \mathbb{Z}_p^*$ and computes $\tilde{C}' = \tilde{C} \cdot Y^{\gamma}, C_0' = C_0 \cdot g^{\gamma}, e_1' = (e_1 \cdot U^{\gamma})^x, e_2' = (e_2 \cdot V^{\gamma})^y$. For all $\{\{\alpha_{i,t}\}_{1\le t\le n_i} \in \mathbb{P}_i\}_{1\le i\le n}$, this function computes $C_{i,t,2}' = C_{i,t,2} \cdot (A_{i,t}^{a_{i,t}})^{\gamma}$. For all $\{\{\alpha_{i,t}\}_{1\le t\le n_i} \notin \mathbb{P}_i\}_{1\le i\le n}$, this function let $C_{i,t,2}'$ be a random number. Finally, this function outputs the complete ciphertext, $CC_{\mathbb{P}}^{\mathbb{W}} = \{\hat{C}', C_0', e_1', e_2', \{\{C_{i,t,1}, C_{i,t,2}'\}_{1\le t\le n_i}\}_{1\le i\le n}\}$.

- **Dec**$(SK_{\mathbb{L}}, CC_{\mathbb{P}}^{\mathbb{W}})$

  This function selects $\{\{C_{i,1}', C_{i,2}'\} = \{C_{i,t_i,1}, C_{i,t_i,2}'\}\}_{1\le i\le n}$, while t_i is the index of $\mathbb{L}_i$. Then computes

$$M = \frac{\tilde{C}' \cdot e(C_0', D_e)}{e(e_1' \cdot e_2', \Delta) e\left(C_0', D_0\right)} \cdot \frac{\prod_{i=1}^{n} e(C_{i,1}', D_{i,1}) e(C_{i,2}', D_{i,2})}{\prod_{i=1}^{n} e\left(C_0', D_{i,0}\right)}. \tag{1}$$

  If $\mathbb{L} \models \mathbb{W}$, this function outputs $\{M\}$. Otherwise, this function outputs a malformed random number, $M' \in \mathbb{G}_T$.

3.4 Representation of Attribute, Policy and Permission

In order to clarify the usage of the data structure of attribute, policy, and permission in CoAABE, this subsection introduces how to map real data into the attribute types and values for access control. Here, we take an email system as an example. An email system takes (***gender, division, position***) as the three attribute types for the access control of the email system. For each attribute type, the real data are assigned as follows: ***gender***=*(male, female)*, ***division***=*(sales, accounting, production)*, and ***position***=*(staff,*

manager). Taking the attribute type *gender* as an example, the variables of the attribute values for the attribute set of *gender* can be represented as $\mathbb{A}_1 = \{\alpha_{1,1}, \alpha_{1,2}\}$. For each $\alpha_{i,t}$, $\{a_{i,t}, b_{i,t}\} \in \mathbb{Z}_p^{*2}$ and $A_{i,t} \in \mathbb{G}$ are generated for the corresponding public and secret parameters in MPK and MSK, respectively. For the example of mapping the attribute value and real data, the administrator of the email system can assign a table for *gender* as $\mathbb{A}_1 = \{\{a_{1,1}, b_{1,1}, A_{1,1}, \text{“male”}\}, \{a_{1,2}, b_{1,2}, A_{1,2}, \text{“female”}\}\}$, *division* as $\mathbb{A}_2 = \{\{a_{2,1}, b_{2,1}, A_{1,1}, \text{“sales”}\}, \{a_{2,2}, b_{2,2}, A_{2,2}, \text{“accounting”}\}, \{a_{2,3}, b_{2,3}, A_{2,3}, \text{“production”}\}\}$, and *position* as $\mathbb{A}_3 = \{\{a_{3,1}, b_{3,1}, A_{3,1}, \text{“staff”}\}, \{a_{3,2}, b_{3,2}, A_{3,2}, \text{“manager”}\}\}$. Based on the above representation of data structure of attribute types, values, and the real data, one can transform the real data to be used for access control into the variables of the proposed CoAABE for using them in the functions.

3.5 Application of CoAABE

This subsection shows the application of CoAABE to a real use case below. Here we take an email system, which allows each user to secretly email messages with specific conditions (*i.e.*, policy, $\mathbb{W}$) to specified receivers, as an example. To construct an email system that provides the fine-grained access control by attribute-based encryption and avoids users to pinpoint receivers by using arbitrary policies for encryption, there are three kinds of servers, *i.e.*, key generation center (KGC), email server (ES), and control server (CS). To establish the email system by CoAABE, the administrator of the email system has to run **Setup** function to obtain (MSK, MPK, CSK) at the beginning. The administrator then publishes MPK, and sends MSK to the KGC and CSK to the CS. Initially, each user is assigned three types of attributes, *i.e.*, gender, division, and position, and is issued a unique user secret key with her/his assigned attribute values (*i.e.*, $\mathbb{L}$) by the KGC running **KeyGen** function. Moreover, each user is assigned a permission, $\mathbb{P}$, consisting of granted attributes to specify which attributes are allowed to compose a policy for an encryption. When a user wants to send an email to specific receivers, she/he needs to determine the policy $\mathbb{W}$ and encrypt an email message M with $\mathbb{W}$ to obtain a partial ciphertext, $PC_{\mathbb{W}}$. The sender then sends $PC_{\mathbb{W}}$ to ES with her/his ID and password. The ES will forward $PC_{\mathbb{W}}$ with the sender's profile to CS. The CS will then retrieve the corresponding permission $\mathbb{P}$ based on her/his profile, *i.e.*, the granted attributes for the permission assigned to the sender. Then, the CS performs **CoEnc** with $PC_{\mathbb{W}}$, its CSK, and $\mathbb{P}$ to obtain $CC_{\mathbb{P}}^{\mathbb{W}}$ and sends back $CC_{\mathbb{P}}^{\mathbb{W}}$ to ES as the complete encrypted email message. If $\mathbb{P}$ satisfies $\mathbb{W}$, $CC_{\mathbb{P}}^{\mathbb{W}}$ can be decrypted correctly by the receivers, who has the secret key $SK_{\mathbb{L}}$, where $\mathbb{L}$ satisfies $\mathbb{W}$. By adopting CoAABE, only the users possess the granted attributes that satisfy the given policy of each encryption can email messages secretly with fine-grained access control ability to filter out intended receivers.

3.6 Security Proof

3.6.1 Semantic security

This work refers to [12–15], for the following security game. In the security game, $\mathbf{A}$ is an adversary, and the simulator of CoAABE, $\mathbf{S}_{CoA}$, is $\mathbf{A}$'s challenger. In addition, $\mathbf{S}_{CoA}$ also acts as an adversary of the simulator of NYO-ABE [12], $\mathbf{S}_{NYO}$.

- **Init:** $\mathbf{A}$ commits $\mathbb{W}_0$ and $\mathbb{W}_1$ to $\mathbf{S}_{CoA}$, then $\mathbf{S}_{CoA}$ commits $\mathbb{W}_0$ and $\mathbb{W}_1$ to $\mathbf{S}_{NYO}$.

- **Setup:** $\mathbf{S}_{NYO}$ generates $MPK' = \{Y, p, \mathbb{G}, \mathbb{G}_T, g, e, \{\{A_{i,t}^{a_{i,t}}, A_{i,t}^{b_{i,t}}\}_{1\le t\le n_i}\}_{1\le i\le n}\}$ and $MSK' = \{w, \{\{a_{i,t}, b_{i,t}\}_{1\le t\le n_i}\}_{1\le i\le n}\}$, and $\mathbf{S}_{NYO}$ gives MPK' to $\mathbf{S}_{CoA}$. After that, $\mathbf{S}_{CoA}$ generates $U, V, \Delta, a, b, \delta$, and CSK. Finally, $\mathbf{S}_{CoA}$ gives $MPK = \{U, V, \Delta, MPK'\}$ to $\mathbf{A}$.
- **Phase 1:** $\mathbf{A}$ submits the user attribute list $\mathbb{L}$ to $\mathbf{S}_{CoA}$, then $\mathbf{S}_{CoA}$ submits the user attribute list $\mathbb{L}$ to $\mathbf{S}_{NYO}$. Next, if $\mathbb{L} \models \mathbb{W}_0 \wedge \mathbb{L} \models \mathbb{W}_1$ or $\mathbb{L} \not\models \mathbb{W}_0 \wedge \mathbb{L} \not\models \mathbb{W}_1$, $\mathbf{S}_{NYO}$ generates $SK'_{\mathbb{L}} = \{D_0, \{D_{i,0}, D_{i,1}, D_{i,2}\}_{1\le i\le n}\}$ and sends to $\mathbf{S}_{CoA}$. Otherwise, reject the query. Finally, $\mathbf{S}_{CoA}$ generates D_e and gives $SK_{\mathbb{L}} = \{D_e, D_0\{D_{i,0}, D_{i,1}, D_{i,2}\}_{1\le i\le n}\}$ to $\mathbf{A}$. The above query can be repeated for polynomial times.
- **Challenge :** The adversary sends messages M_0 and M_1 to $\mathbf{S}_{CoA}$, where $M_0 = M_1$. If the adversary has any $SK_{\mathbb{L}}$ such that $\mathbb{L} \models \mathbb{W}_0 \wedge \mathbb{L} \models \mathbb{W}_1$, then $\mathbf{S}_{CoA}$ submits messages M_0 and M_1 to the challenger $\mathbf{S}_{NYO}$. After that, $\mathbf{S}_{NYO}$ randomly selects $b \in \{0,1\}$ to generate $PC'_{\mathbb{W}_b} = \{\tilde{C}, C_0, \{\{C_{i,t,1}, C_{i,t,2}\}_{1\le t\le n_i}\}_{1\le i\le n}\}$, and sends $PC'_{\mathbb{W}_b}$ to $\mathbf{S}_{CoA}$. Finally, $\mathbf{S}_{CoA}$ generates e_1, e_2, let $PC_{\mathbb{W}_b} = \{PC'_{\mathbb{W}_b}, e_1, e_2\}$, runs $\mathbf{CoEnc}(PC_{\mathbb{W}_b}, CSK, MPK, \mathbb{P} = \mathbb{A})$, and sends $CC^{\mathbb{P}}_{\mathbb{W}_b}$ to $\mathbf{A}$.
- **Phase 2:** Phase 1 is repeated, but if $M_0 \neq M_1$, $\mathbf{A}$ should not submit $\mathbb{L}$ such that $\mathbb{L} \models \mathbb{W}_0 \wedge \mathbb{L} \models \mathbb{W}_1$.
- **Guess:** The adversary guess b' is 0 or 1, and wins the game if $Pr[b' = b] - \frac{1}{2}$ is non-negligible.

Above security game defines that $\mathbf{S}_{CoA}$ is a simulator of CoAABE for $\mathbf{A}$, and $\mathbf{S}_{NYO}$ is a simulator of NYO-ABE. If the advantage of $\mathbf{A}$ breaking CoAABE is ε, the advantage of $\mathbf{S}_{CoA}$ breaking NYO-ABE is ε', then since the above $\mathbf{S}_{CoA}$ acts as a man-in-the-middle between $\mathbf{A}$ and $\mathbf{S}_{NYO}$, it can be inferred that $\varepsilon \le \varepsilon'$. In addition, it has been proved in [12] that ε' is negligible. Therefore, the adversary has only negligible advantage to guess the message, M_b, or policy, $\mathbb{W}_b$.

3.6.2 Complete ciphertext unforgeability against the encryptor

This section will prove that the complete ciphertext of CoAABE is unforgeable against the encryptor. Since the adversary $\mathbf{A}$ can obtain MPK, various $SK_{\mathbb{L}}$, $PC_{\mathbb{W}}$ and $CC^{\mathbb{P}}_{\mathbb{W}}$. Let these parameters be a set AP'. There are two ways for $\mathbf{A}$ to forge a $CC_{\mathbb{W}\not\models\mathbb{P}}$. The first way is to use AP' to generate a $CC_{\mathbb{W}\not\models\mathbb{P}}$ by itself, and the second way is to submit a $PC_{\mathbb{W}\not\models\mathbb{P}}$ to $\mathbf{S}$.

$$
\begin{aligned}
M &= \frac{\tilde{C}\cdot e(C'_0, D_e)\prod_{i=1}^{n} e(C'_{i,1}, D_{i,1}) e(C'_{i,2}, D_{i,2})}{e(e'_1\cdot e'_2, \Delta) e\left(C'_0, D_0\right)\prod_{i=1}^{n} e\left(C'_0, D_{i,0}\right)} \\
&= \frac{M\cdot e(g,g)^{w(r+\gamma)} e(g^{(r+\gamma)}, g^{\delta(a+b)})}{e(g^{(r+\gamma)x}\cdot(g^{(r+\gamma)y}, g^{\delta}) e(g^{(r+\gamma)}, g^{w-s})}\cdot \\
&\quad \frac{\prod_{i=1}^{n} e((A_{i,t_i}^{b_{i,t_i}})^{r_{i,t_i}}, g^{a_{i,t_i}\lambda_{i,t_i}})\cdot e((A_{i,t_i}^{a_{i,t_i}})^{(r+\gamma-r_{i,t_i})}, g^{b_{i,t_i}\lambda_{i,t_i}})}{\prod_{i=1}^{n} e(g^{(r+\gamma)}, g^{s_i} A_{i,t_i}^{a_{i,t_i}b_{i,t_i}\lambda_{i,t_i}})} \qquad (2) \\
&= \frac{M\cdot e(g,g)^{w(r+\gamma)}}{e(g^{(r+\gamma)}, g^{w-s})}\cdot \frac{\prod_{i=1}^{n} e((A_{i,t_i}^{b_{i,t_i}})^{r_{i,t_i}}, g^{a_{i,t_i}\lambda_{i,t_i}})\cdot e((A_{i,t_i}^{a_{i,t_i}})^{(r+\gamma-r_{i,t_i})}, g^{b_{i,t_i}\lambda_{i,t_i}})}{\prod_{i=1}^{n} e(g^{(r+\gamma)}, g^{s_i} A_{i,t_i}^{a_{i,t_i}b_{i,t_i}\lambda_{i,t_i}})}
\end{aligned}
$$

The probability of an adversary generates $CC_{\mathbb{W}\not\models\mathbb{P}}$ by AP' is discussed below. The adversary has to use e'_1 and e'_2 to generate $CC_{\mathbb{W}\not\models\mathbb{P}}$ by AP'. Thus, the adversary has to obtain e'_1 and e'_2 in Phase 1 or Phase 2 before generating a $CC_{\mathbb{W}\not\models\mathbb{P}}$ by AP'.

The probability of an adversary generates $CC_{\mathbb{W}\not\models\mathbb{P}}$ by AP' is discussed below. The adversary has to use e'_1 and e'_2 to generate $CC_{\mathbb{W}\not\models\mathbb{P}}$ by AP'. Thus, the adversary has to obtain e'_1 and e'_2 in Phase 1 or Phase 2 before generating a $CC_{\mathbb{W}\not\models\mathbb{P}}$ by AP'.

To decrypt linear encryption successfully, **A** has to let $e(C'_0, D_e)$ is equal to $e(e'_1 \cdot e'_2, \Delta)$. By the equation 2, $e(e'_1 \cdot e'_2, \Delta) = e(g,g)^{\delta(a+b)(r+\gamma)}$ and $D_e = g^{\delta(a+b)}$. Therefore, **A** chooses $C'_0 = g(r+\gamma)$, such that $e(C'_0, D_e)$ is equal to $e(e'_1 \cdot e'_2, \Delta)$. When C'_0 is $g(r+\gamma)$, the last part of equation 2 is equivalent to NYO-ABE [12], where the random number r is replaced by $r+r'$.

To generate a valid $CC_{\mathbb{W}\not\models\mathbb{P}}$, **A** has to calculate $A_{i,t_i}^{a_{i,t_i}(r+\gamma-r_{i,t_i})}$. Since **A** knows $A_{i,t_i}^{a_{i,t_i}(r-r_{i,t_i})}$, as long as $A_{i,t_i}^{a_{i,t_i}\gamma}$ is calculated, **A** can successfully generate a valid forged $CC_{\mathbb{W}\not\models\mathbb{P}}$. However, **A** can only obtain g^{γ} among all the parameters related to γ. According to the computational Diffie-Hellman assumption (CDH), the probability of **A** successfully calculates $A_{i,t_i}^{a_{i,t_i}\gamma}$ by g^{γ} and $A_{i,t_i}^{a_{i,t_i}r}$ is negligible. Let the event of **A** breaking linear encryption be ε_0, the event of **A** breaking CDH be ε_1, and the event that **A** successfully generates a valid forged $CC_{\mathbb{W}\not\models\mathbb{P}}$ by AP' be ε. From above security game, it can be obtained that $Pr[\varepsilon] = Pr[\varepsilon_0] \vee Pr[\varepsilon_1] \leq Pr[\varepsilon_0] + Pr[\varepsilon_1]$. Because $Pr[\varepsilon_0]$ and $Pr[\varepsilon_1]$ are negligible , $Pr[\varepsilon]$ is negligible.

The probability of an adversary obtain $CC_{\mathbb{W}\not\models\mathbb{P}}$ by submitting $PC_{\mathbb{W}\not\models\mathbb{P}}$ is discussed below. Section 3.3 **CoEnc** shows that for all $\{\{\alpha_{i,t}\}_{1\leq t\leq n_i} \notin \mathbb{P}_i\}_{1\leq i\leq n}$, $C'_{i,t,2}$ is a random number. So even if the encryptor tries to generate a $CC_{\mathbb{W}\not\models\mathbb{P}}$, the unit who runs **CoEnc** turns all $C'_{i,t,2}$ of $\{\{\alpha_{i,t}\}_{1\leq t\leq n_i} \notin \mathbb{P}_i\}_{1\leq i\leq n}$ to malformed. Based on CDH, it is difficult for the encryptor to compute $C'_{i,t,2}$. Therefore, the probability of submitting $PC_{\mathbb{W}\not\models\mathbb{P}}$ to obtain $CC_{\mathbb{W}\not\models\mathbb{P}}$ by the adversary, i.e., $Pr[\varepsilon']$, is negligible.,

In summary, since neither adversary can use AP' to generate a $CC_{\mathbb{W}\not\models\mathbb{P}}$ by itself nor submit a $PC_{\mathbb{W}\not\models\mathbb{P}}$ to obtain $CC_{\mathbb{W}\not\models\mathbb{P}}$. It can be concluded that the complete ciphertext of CoAABE is unforgeable against the encryptor.

4. SYSTEM AND SECURITY MODELS FOR SAD2D

4.1 System Model

The system model is shown in Fig. 1, which consists of three parts. They will be described in detail below.

UEs refer to any device that uses the D2D service. A UE can play two roles, announcing UE (A-UE) and monitor UE (M-UE). A-UE refers to the user who initiates the device discovery, and M-UE refers to the user who responds to the discovery. ProSe application servers (PAS) refer to servers that provide D2D services. In this work, PAS has two components, APMU and KGC. APMU is responsible for managing the attributes and permissions of the user, and KGC is responsible for managing the key. 5G ProSe Function (PSF) is responsible for spectrum allocation. Multi-access Edge Computing (MEC) platform MEC is responsible for assisting PAS to achieve encryption control.

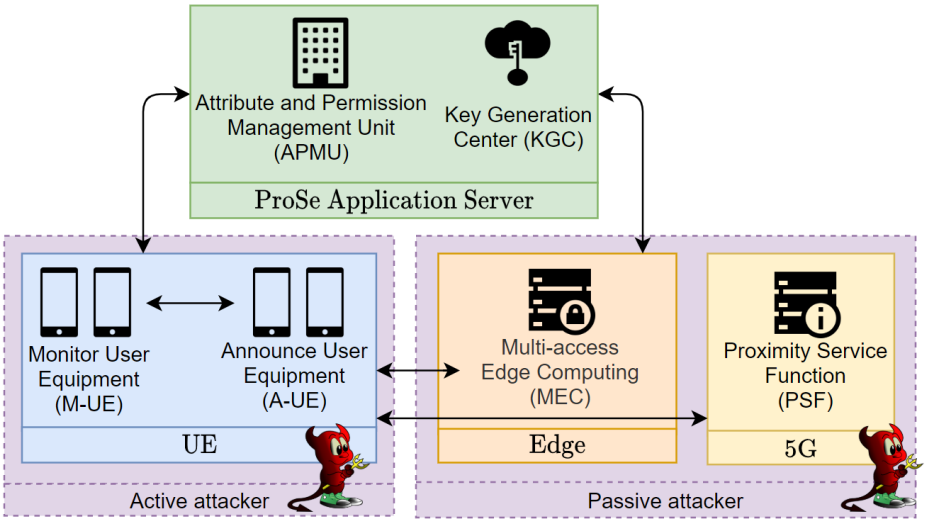

Fig. 1. System and attacker model.

4.2 Attacker Model

As shown in Fig. 1, in this work, it is assumed that the active attacker can monitor, modify, and replay messages between UEs, has background knowledge of cryptography, knows the details of all algorithms that used in this work, and knows all public parameters. Besides, the attacker can run passive attacks on MEC but can not obtain the securely protected CSK.

5. SECURE ANONYMOUS DEVICE-TO-DEVICE PROTOCOLS

5.1 Key Issuance

Fig. 2 shows the key issuance phase of SAD2D. This phase focuses on demonstrating the feasibility of SAD2D rather than providing security features such as mutual authentication. Therefore, it is assumed that the connections between APMU to MEC, APMU to UE, and APMU to KGC are secure. In practice, since APMU is a fixed and trusted server, other components can easily establish a secure channel using HTTPS.

SAD2D allows multiple APMUs to share KGC. Therefore, a new APMU can request to join the system by submitting application name, APP_j, application-related information, Info, security parameters, 1^k, and access structure, $\mathbb{A}_j$. When a KGC receives a new APMU join request, the KGC first audits the APMU information, and if it passes, the KGC runs **Setup**$(1^k, \mathbb{A}_j)$, and sends the CSK_{APP_j} and MPK_{APP_j} back to APMU. After that, APMU forwards CSK_{APP_j} and MPK_{APP_j} to its application server in the MEC and start to accept user join requests.

A UE_i can try to submit UE-related information to any APMU to obtain SAD2D services. When APMU receives UE's information, APMU will first review the information.

If approved, APMU first generates a user attribute list, $\mathbb{L}_{UE_i}$. Next, it asks KGC to use $\mathbb{L}_{UE_i}$ to generate a private key, SK_{UE_i}, a user permission $\mathbb{P}_{UE_i}$, and a pseudonym identity PID_{UE_i}. Finally, APMU delivers PID_{UE_i} and $\mathbb{P}_{UE_i}$ to MEC, and delivers PID_{UE_i}, SK_{UE_i}, and MPK_{APP_j} to UE_i.

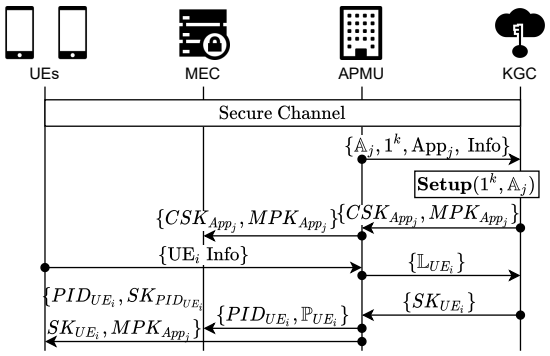

Fig. 2. Key issuance.

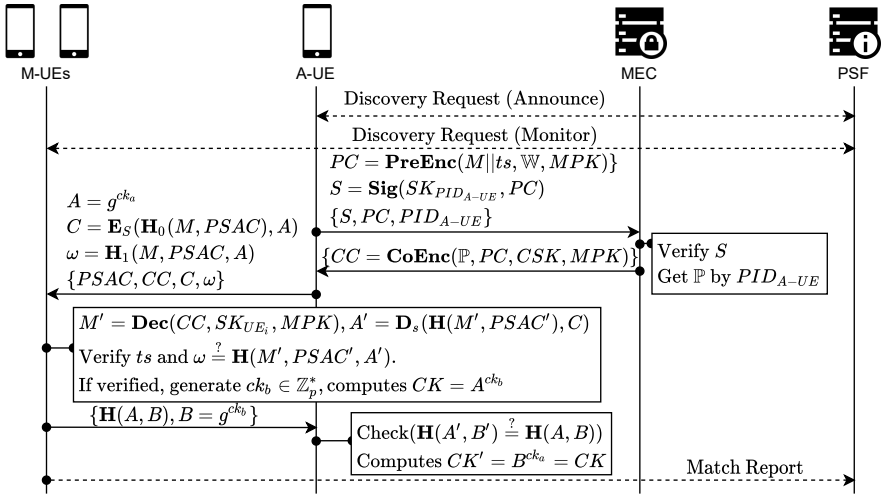

Fig. 3. Discovery.

5.2 Discovery

Fig. 3 shows the discovery phase of SAD2D. At the beginning of the discovery phase, UE will initially establish a communication channel with the assistance of ProSe and share a ProSe Application Code (PSAC). For details of the above steps, please refer to Model A of Direct Discovery in [3]. At this time, any UE that has joined the SAD2D service can communicate with each other.

After the communication channel between UEs established, A-UE starts to initiate a cooperative encryption request. First, A-UE sets a policy, $\mathbb{W}$, and randomly selects a message, $\mathbb{M}$, with a timestamp, ts. Then, A-UE runs PreEnc to generate a partial ciphertext,

PC. Next, A-UE generates a identity-based signature [16], S, with identity-based secret key, $SK_{PID_{A-UE}}$, and PC. Finally, A-UE sends S, PC, and PID_{A-UE} to MEC.

When MEC receives the cooperative encryption request, MEC will check the correctness of the signature and generate a complete ciphertext CC to A-UE. MEC first verifies the signature, S, with PID_{A-UE}, then find permission, $\mathbb{P}$, according to PID_{A-UE}. Next, MEC runs $CC = \mathbf{CoEnc}(\mathbb{P}, PC, CSK, MPK)$, and finally sends CC to A-UE.

After A-UE receives the complete ciphertext, CC, A-UE begins to prepare a message for M-UEs. First, A-UE generates $A = g^{ck_a}$ for Diffie-Hellman key exchange, and computes $C = \mathbf{E}_S(\mathbf{H}_0(M, PSAC), A)$, where $\mathbf{E}_S()$ is a symmetric encryption function. After that, A-UE calculates $\omega = \mathbf{H}_1(M, PSAC, A)$, and finally sends $PSAC, CC, C$, and ω to M-UE.

When an M-UE receives the message, the M-UE first tries to decrypt CC with its CoAABE secret key, SK_{UE_i}. Then, M-UE verifies the timestamp ,ts, and verifies $\omega \stackrel{?}{=} \mathbf{H}(M', PSAC', A')$ to ensure a successful decryption. Among the hashed parameters, the symbol $'$ represents the parameters calculated or held by M-UE. If the M-UE can successfully decrypt CC, it means that the M-UE has access to the SAD2D communication. Therefore, the M-UE can start to run Diffie-Hellman key exchange with the A-UE. First, M-UE chooses a random number, ck_b, calculates $B = g^{ck_b}$, and computes session key, $CK = A^{ck_b}$. Finally, M-UE sends $\mathbf{H}(A, B)$ and $B = g^{ck_b}$ to A-UE.

While the A-UE receives response from the M-UE, A-UE first checks $\mathbf{H}(A', B') \stackrel{?}{=} \mathbf{H}(A, B))$ to ensure that M-UE successfully decrypts CC and retrieves A. After that, A-UE computes $CK' = B^{ck_a}$. Because $CK' = B^{ck_a} = A^{ck_b} = CK$, it can be concluded that the key exchange has been completed, and the A-UE and the M-UE have successfully built a secure anonymous D2D communication channel with a shared session key CK.

6. EVALUATION

6.1 Comparison on Security Features

In terms of security, the biggest achievement of CoAABE is to achieve both encryption control and policy hiding. Besides, this work also adopts a relatively flexible And-Multi access structure. Although access structure $\mathbb{A}$ cannot be expanded at present, referring to NYO-ABE construction [12], there may still be some opportunity for further improvement in this regard.

In addition to CoAABE, the SAD2D proposed in this work utilizes the features of CoAABE encryption control and policy hiding to successfully realize the comprehensive protection of identity, attributes, and privacy. Compared with SGD [10], it also reduces the dependence on the core network. Finally, this work also proposes an application-oriented flexible service framework for distinct D2D application service providers to conduct secure D2D connections with fine-grained access control and privacy protection on private information(*i.e.*, attribute and policy information used for access control in services) against the core network and outsider adversaries.

Table 2. Comparison of SAD2D and related works.

	This work	Hsu *etal*. [10]	Yan *etal*. [8]	Li et al. [17]
Identity privacy	√	√	√	√
Attribute privacy	√	√	×	×
Policy privacy	√	×	×	×
CN* independent	√	×	√	×
App based	√	√	×	√

*CN: core nework

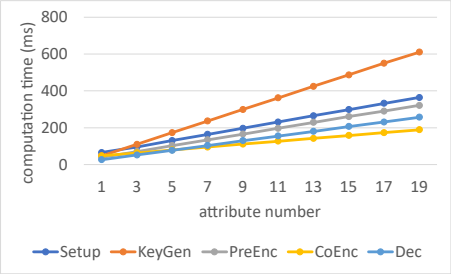

Fig. 4. Attributes of $\mathbb{A}$, $\mathbb{W}$ and $\mathbb{L}$ increased.

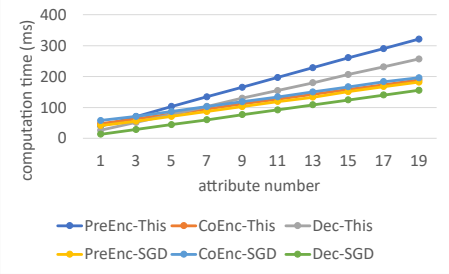

Fig. 5. Comparison with SGD^2.

6.2 Performance Evaluation

This sections introduces the performance evaluation methods and results of this work. The testbed hardware specification of the computer includes CPU of I7-9700 model and memory of 32Gb. The programs in the experiment are based on the jpbc [18] library. Each function is executed 100 times and averaged. After more than 10 tests, it is confirmed that the bias is within 5%.

6.2.1 Performance comparison

Fig. 4 shows that when the number of attributes increases linearly in $\mathbb{A}$, $\mathbb{W}$, and $\mathbb{L}$, the costs of all functions will grow linearly. Fig. 5 shows that the performance of this work on PreEnc and Dec is worse than SGD^2, but this is because of the cost to achieve the protection of the policy.

6.2.2 Queuing theory simulation

This simulation uses the queuing theory to calculate the capacity of MEC in the real world. The experiment includes a fixed parameter T_{stay}, and two sets of variable parameters $T_{process}$ and $T_{interval}$. The following will first introduce how to set each parameter and then introduce the experimental process.

The fixed parameter T_{stay} refers to the time each user stays in the MEC service range. This experiment assumes that the user passes the MEC with a service range of 1KM at a speed of 300 kilometers, so T_{stay} is 12 seconds. This assumption is that the upper limit of the moving speed of ground vehicles is at most 300KM, and the service range of MEC usually is at least one kilometer. Therefore, it is a relatively conservative assumption to

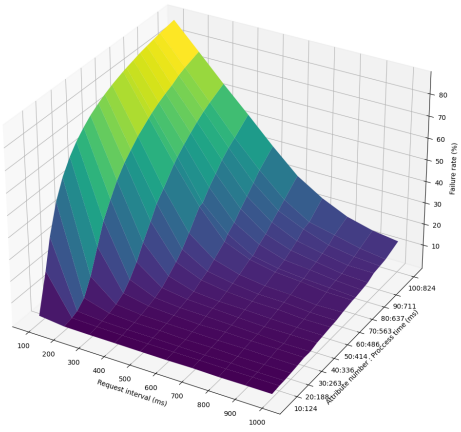

Fig. 6. Queuing test – request fail rate.

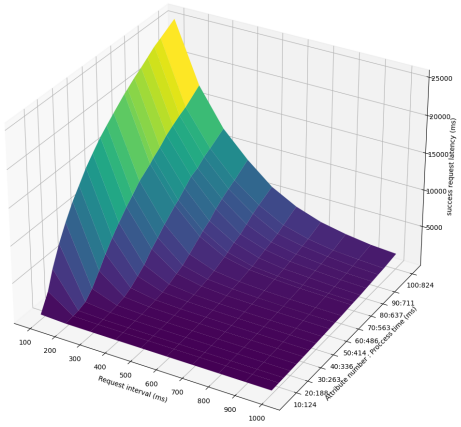

Fig. 7. Queuing test – average success request latency.

assume that T_{stay} is 12 seconds.

The first variable parameter $T_{process}$ refers to the time required for MEC to serve each user, including the transmission delay of 20 seconds [19] plus the calculation time. Due to the characteristics of ABE, the calculation time will be affected by the number of attributes, so this experiment measures the calculation time when the number of attributes is 5 to 100 at intervals of 5. The upper bound of the attribute number is set to 100 because this attribute number is sufficient for most application scenarios.

The second variable parameter $T_{interval}$ refers to the interval between the appearance of two users. Since different user densities, different application scenarios, and even different times will affect this parameter, this experiment selects the most representative range of 100 milliseconds to 1000 milliseconds to display our experimental results after experimenting with various ranges of $T_{interval}$.

At the beginning of the experiment, it sets the duration of the entire experiment to 10,000,000 ms. During this period, the simulation submits service requests continuously at an interval of random.exp ($T_{interval}$), and sets the stay time for each device as random.exp (T_{stay}). The random.exp(β) is a function that draws random numbers from an exponential probability distribution with an average of β. As time progresses, when a device leaves, if the system is processing other requests or the system has not completed the request of the current device, it is determined that the device request has failed. Otherwise, the system determines the device request is successful and records the time interval from the request start to the request complete as the success request latency. Therefore, as shown in Figs. 6 and 7, there will be two experimental results at the end of this experiment, which are failure rate and average successful request time.

In summary, CoAABE inherits the NYO-ABE [12], which is flexible in the design of the access structure. Users can make a trade-off between the number of attributes they need to use and performance considerations. Since the performance of hundreds of milliseconds is very common in attribute encryption methods, the latency is sufficient to meet most non-timeliness requirements.

7. SECURITY ANALYSIS OF SAD2D

7.1 Authenticated Key Exchange

SAD2D uses Diffie-Hellman key exchange (DHKE) technology to exchange the session key, CK, between UEs. If an adversary, $\mathbf{A}$, wants to break the AKE of SAD2D, it can choose to break DHKE or choose to bypass the verification technology of SAD2D. If $\mathbf{A}$ chooses to bypass the verification, there are a total of two attack targets for $\mathbf{A}$ to choose from, A-UE and M-UE.

Observing the SAD2D discovery protocol, it can be found that A-UE decides whether to complete the key exchange by checking $\mathbf{H}(A,B)$. This means that $\mathbf{A}$ must generate a valid $\mathbf{H}(A,B)$ to bypass the authentication of A-UE. Since B can be generated by $\mathbf{A}$, $\mathbf{A}$ can choose to obtain A to generate $\mathbf{H}(A,B)$ or break $\mathbf{H}$. If $\mathbf{A}$ chooses to obtain A, the SAD2D discovery protocol shows that A is protected by $\mathbf{E}_S(\mathbf{H}(M,PSAC),A)$. Since $\mathbf{A}$ can observe $PSAC$, $\mathbf{A}$ can try to obtain M to generate $\mathbf{H}(M,PSAC)$, or choose to break $\mathbf{E}_S$, or choose to break $\mathbf{H}_1$. If $\mathbf{A}$ chooses to obtain M, since M is the ciphertext of CoAABE, $\mathbf{A}$ has to break semantic security of CoAABE to obtain M.

Summarizing the process of $\mathbf{A}$ trying to bypass authentication of A-UE, we assume that the event of $\mathbf{A}$ breaking H_0 is ε_0, breaking H_1 is ε_1, breaking E_S is ε_2, breaking semantic security of CoAABE is ε_3, and breaking authentication of A-UE is ε_{A-UE}. It can be concluded that $Pr[\varepsilon_{A-UE}] = Pr[\varepsilon_0] \vee Pr[\varepsilon_1] \vee Pr[\varepsilon_2] \vee Pr[\varepsilon_3] \leq Pr[\varepsilon_0] + Pr[\varepsilon_1] + Pr[\varepsilon_2] + Pr[\varepsilon_3]$.

Next, it simulates the situation that $\mathbf{A}$ tries to bypass authentication of M-UE. If $\mathbf{A}$ wants to bypass the authentication of M-UE, it means that $\mathbf{A}$ must generate a complete ciphertext, and the corresponding policy is not satisfy encryption permission of $\mathbf{A}$. However, this action is equivalent to breaking the complete ciphertext unforgeability of CoAABE, so we can conclude that the probability $Pr[\varepsilon_{M-UE}]$ of $\mathbf{A}$ bypassing the authentication of M-UE is equivalent to the probability $Pr[\varepsilon_4]$ of breaking the complete ciphertext unforgeability of CoAABE.

In summary, the probability of $\mathbf{A}$ breaking AKE of SAD2D is $Pr[\varepsilon_{SAD2D}] = Pr[\varepsilon_{M-UE}] \vee Pr[\varepsilon_{A-UE}] \vee Pr[\varepsilon_{DHKE}] = Pr[\varepsilon_0] \vee Pr[\varepsilon_1] \vee Pr[\varepsilon_2] \vee Pr[\varepsilon_3] \vee Pr[\varepsilon_4] \vee Pr[\varepsilon_{DHKE}] \leq Pr[\varepsilon_0] + Pr[\varepsilon_1] + Pr[\varepsilon_2] + Pr[\varepsilon_3] + Pr[\varepsilon_4] + Pr[\varepsilon_{DHKE}]$ Since neither ε_0 to ε_4 nor ε_{DHKE} can be broken by a polynomial-time adversary, it can be concluded that $\mathbf{A}$ can not break AKE of SAD2D.

7.2 Privacy

SAD2D guarantees user identity (ID) privacy, policy privacy, and attribute privacy. Regarding ID privacy, since SAD2D never used the UE's ID during the discovery phase, there is no ID privacy issue, so we can consider the event of ID leakage $Pr[\varepsilon_{ID}]$ to be 0.

Regarding policy privacy, the semantic security of CoAABE is not only for ciphertexts but also for policies. Policies can be regarded as ciphertexts that can not be decrypted. $\mathbf{A}$ can only obtain the policy by breaking the semantic security of CoAABE. Therefore, we can conclude that the event of policy privacy leakage $Pr[\varepsilon_W]$ is equal to the probability of CoAABE selects security broken by $\mathbf{A}$.

Regarding attribute privacy, because the attribute is stored in the CoAABE user private keys, the $\mathbf{A}$ cannot obtain the private key of other users. Therefore, the $\mathbf{A}$ can only

explore other user's attributes by continuously sending different policies. However, the **A**'s encryption permission is restricted, so we can conclude that the probability of attribute privacy leakage $Pr[\varepsilon_L]$ is equal to the probability of CoAABE ciphertext unforgeability broken by **A**.

In summary, since **A** cannot successfully obtain ID, policy, or attribute information from other UEs. It can be concluded that SAD2D is a secure anonymous D2D communication protocol on identities, attributes, and policies.

8. CONCLUSION

This work proposes a cooperative attribute-encryption with hidden policy and constructs a security system to enhance the security and privacy for D2D communications in the future mobile networks. The proposed CoAABE and SAD2D are both provably secure according to the security proofs provided. Moreover, it is the first work that supports both fine-grained access control and hidden policy during device discovery procedure in D2D communications. In addition, the proposed SAD2D is efficient based on the experimental results under queueing model. Thus, security, privacy, and efficiency are taken into account in this work, simultaneously.

ACKNOWLEDGMENT

This work was partially supported by the Ministry of Science and Technology of Taiwan under Grant 109-2221-E-110-041-MY3, Grant 109-2923-E-011-006-MY3, and Grant 110-2218-E-110-007-MBK, and in part by the Information Security Research Center, National Sun Yat-sen University, Taiwan.

REFERENCES

1. M. Haus, M. Waqas, A. Y. Ding, Y. Li, S. Tarkoma, and J. Ott, "Security and privacy in device-to-device (D2D) communication: A review," *IEEE Communications Surveys & Tutorials*, Vol. 19, 2017, pp. 1054-1079.
2. R. I. Ansari, C. Chrysostomou, S. A. Hassan *et al.*, "5G D2D networks: Techniques, challenges, and future prospects," *IEEE Systems Journal*, Vol. 12, 2017, pp. 3970-3984.
3. 3rd Generation Partnership Project (3GPP), *Proximity-Based Services (ProSe), TS 23.303 V16.0.0*, 3rd Generation Partnership Project (3GPP), 2020.
4. 3rd Generation Partnership Project (3GPP), *Study on system enhancement for Proximity based Services (ProSe) in the 5G System (5GS), TR 23.752 V0.2.0*, 3rd Generation Partnership Project (3GPP), 2019.
5. 3rd Generation Partnership Project (3GPP), *Proximity-Services (ProSe) Function to ProSe Application Server Aspects (PC2), TS 29.343 V16.0.0*, 3rd Generation Partnership Project (3GPP), 2020.
6. Y. Sun, J. Cao, M. Ma, Y. Zhang, H. Lin, and B. Niu, "EAP-DDBA: Efficient anonymity proxity device discovery and batch authentication mechnism for massive D2D

communication devices in 3GPP 5G HetNet," *IEEE Transactions on Dependable and Secure Computing*, Vol. 67, 2020, pp. 313-323.
7. S. Pizzi, C. Suraci, A. Iera, A. Molinaro, and G. Araniti, "A sidelink-aided approach for secure multicase service delivery: From human-oriented multimedia traffic to machine type communications," *IEEE Transactions on Broadcasting*, Vol. 67, 2021, pp. 313-323.
8. Z. Yan, H. Xie, P. Zhang, and B. B. Gupta, "Flexible data access control in D2D communications," *Future Generation Computer Systems*, Vol. 82, 2018, pp. 738-751.
9. J. Guo, J. Ma, X. Li, J. Zhang, and T. Zhang, "An attribute-based trust negotiation protocol for D2D communication in smart city balancing trust and privacy," *Journal of Information Science and Engineering*, Vol. 33, 2017, pp. 1007-1023.
10. R. H. Hsu, H. S. Fan, and L. C. Wang, "SGD2: Secure group-based device-to-device communications with fine-grained access control for IoT in 5G," in *Proceedings of IEEE Conference on Dependable and Secure Computing*, 2021, pp. 1-8.
11. R. H. Hsu, C. I. Fan, T. Q. Quek, and J. Lee, "CORE: Cooperative encryption with its applications to controllable security services," in *Proceedings of IEEE Conference on Dependable and Secure Computing*, 2018, pp. 1-8.
12. T. Nishide, K. Yoneyama, and K. Ohta, "Attribute-based encryption with partially hidden encryptor-specified access structures," in *Proceedings of International Conference on Applied Cryptography and Network Security*, 2008, pp. 111-129.
13. J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *Proceedings of Annual International Conference on Theory and Applications of Cryptographic Techniques*, 2008, pp. 146-162.
14. D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data," in *Proceedings of Theory of Cryptography Conference*, 2007, pp. 535-554.
15. E. Shi, J. Bethencourt, T. H. Chan, D. Song, and A. Perrig, "Multi-dimensional range query over encrypted data," in *Proceedings of IEEE Symposium on Security and Privacy*, 2007, pp. 350-364.
16. J. C. Choon and J. H. Cheon, "An identity-based signature from gap Diffie-Hellman groups," in *Proceedings of International Workshop on Public Key Cryptography*, 2003, pp. 18-30.
17. Q. Li, L. Huang, R. Mo, H. Huang, and H. Zhu, "Robust and scalable data access control in D2D communications," *IEEE Access*, Vol. 6, 2018, pp. 58 858–58 867.
18. A. De Caro and V. Iovino, "JPBC: Java pairing based cryptography," in *Proceedings of IEEE Symposium on Computers and Communications*, 2011, pp. 850-855.
19. K. Nguyen, M. G. Kibria, J. Hui, K. Ishizu, and F. Kojima, "Minimum latency and optimal traffic partition in 5G small cell networks," in *Proceedings of IEEE 87th Vehicular Technology Conference*, 2018, pp. 1-5.

Ruei-Hau Hsu received the BS and MS degrees in Computer Science from Tunghai University, Taiwan, in 2002 and 2004, respectively. He received the Ph.D. degree in Computer Science and Engineering from National Sun Yat-sen University, Kaohsiung, Taiwan, in 2012. He was the Postdoctoral Research Fellow at the Department of Computer Science, National Chiao Tung University from 2012 to 2014, and at iTrust, Centre for Research in Cyber Security at Singapore University of Technology and Design from 2014 to 2017. He was a Scientist with the Data Storage Institute (DSI) and Institute for Infocomm Research (I2R), Agency for Science, Technology and Research (A*STAR), Singapore. He is currently an Assistant Professor with the Department of Computer Science and Engineering, National Sun Yat-sen University. Dr. Hsu received two Best Doctoral Dissertation Awards from Institute of Information and Computing Machinery and Best Doctoral Dissertation Award from Chinese Cryptology and Information Security in 2012, respectively. In 2012, he has been the member of the Phi Tau Phi Scholastic Honor Society.

Lu-Chin Wang was born in Pingtung, Taiwan. He received the master's degrees in Computer Science and Engineering from National Sun Yat-sen University, Kaohsiung, Taiwan, in 2021. His research interests include cryptography, web security, network security, and communication security.

Hsiang-Shian Fan received the master's degree in Computer Science and Engineering from National Sun Yat-sen University, Kaohsiung, Taiwan. Her research interests include cyber security and cyptography. In 2021, she has been the scholastic honorary member of the Phi Tau Phi scholastic honor society, Taiwan. Additionally, she won the Professor Laih Chi-Sung Thesis Award in 2022.