Robust and Scalable Data Access Control in D2D Communications

QI LI,1,2, LIZHI HUANG1, RUO MO3, HAIPING HUANG1,2, AND HONGBO ZHU 4

1School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, 210023 China (e-mail: liqics@njupt.edu.cn; pluto936@126.com; hhp@njupt.edu.cn)
2Jiangsu Key laboratory of Big Data Security & Intelligent Processing, Nanjing University of Posts and Telecommunications, Nanjing, 210023 China
3School of Cyber Engineering, Xidian University, Xi’an, 710071 China (e-mail: 593430655@qq.com)
4Jiangsu Key Laboratory of Wireless Communications, Nanjing University of Posts and Telecommunications, Nanjing, 210003 China (e-mail: zhuhb@njupt.edu.cn)

Corresponding author: Haiping Huang (e-mail: hhp@njupt.edu.cn).

This research is supported by the National Natural Science Foundation of China under grant No.61502248, 61672297, 61427801, China Postdoctoral Science Foundation (Grant no. 2018M632350), Natural Science Foundation of Jiangsu Province (Grant No. BK20181394).

ABSTRACT As an emerging technique in 5G cellular networks, D2D communication efficiently utilizes the available resources. However, the concerns of data security, identity privacy and system scalability have not been sufficiently addressed. In this paper, we propose a robust and scalable data access control scheme (RSDAC) in D2D communication, where the key build block is a multi-authority ciphertext-policy attribute-based encryption (MA-CP-ABE) with large universe and verifiable outsourced decryption. In RSDAC, the system attribute universe is scalable, which is exponentially large without resource waste. Each base station (BS) governs the whole attribute universe individually. The data owner can define any monotonic access structure to encrypt its data. During key generation phase, each BS can independently verify the user’s legitimacy and then generate intermediate key for the legal user according to its attribute set. A core network server (CNS) acts as the central authority which will generate the final private key for the user basing on his intermediate key. We also design an efficient method to offload the complicated decryption to some devices with adequate computation resource and further check the correctness of decryption result. The security analysis and performance comparison indicate that our scheme is secure, efficient and applicable.

INDEX TERMS D2D, Access Control, CP-ABE, Multi-authority, Large Universe, Efficient Decryption.

I. INTRODUCTION

In recent years, device to device (D2D) communication has emerged as a promising technique to efficiently utilize the spectral resources in 5G cellular networks [1], [2], because of its inherent features, e.g., improving spectral efficiency, delay constrained, improving system capacity, etc. D2D communication enables the user equipments (UEs) to directly communicate with each other without being involved in the fixed network infrastructures, such as bluetooth, base stations (BSs) and access points (APs). By using D2D communication techniques, people can efficiently and rapidly share their data via various UEs. However, despite the above advantages, there are three main issues: data security, identity privacy and system scalability to be addressed before applying D2D communication in practice.

Data Security. As the UE connects with others directly, D2D communication might be vulnerable to many security attacks, such as channel eavesdropping and modification of data [3]. To resist such attacks, a feasible way is to encrypt the data before transmitting it to others. The data owner should also indicate who is allowed to access the encrypted data. Meanwhile, the data should be accessed only by the authorized users and is confidential to the unauthorized users. However, traditional symmetric encryption and public key cryptography are not suitable for D2D communication applications, due to the complexity of key agreement and management.

Identity Privacy. During sharing the data with some users, the data owner may want to hide his or the UE’s identification information. For instance, a data owner shoots a scandal video by his mobile phone and transmits it to someone else, but he do not want to expose any information of himself and his device. If the identity privacy can not be guaranteed, it may result in inferior user experience of D2D communication.
System Scalability. While deploying D2D communication in real applications, the scalability of system is worth considering, due to that plenty of users and UEs are coexisting in the system. Once the system parameter size is set too small, the system may be thoroughly reconstructed in future. If the system parameter size is set too large, it would incur superfluous waste of resource.

To address the issues mentioned above, in this paper, we present a robust and scalable data access control scheme (RSDAC) for D2D communication. We construct a multi-authority ciphertext-policy attribute-based encryption (MA-CP-ABE) scheme with large universe and verifiable outsourced decryption, and take it as the basis of the data access control scheme for D2D communication. In RSDAC, there are multiple base stations (BSs) and a core network server (CNS). Each D2D user equipment (DUE) can link to a BS directly or via the relay of a cellular user equipment (CUE), and is described by some attributes, such as spectrum, brand and trust level. Aiming to improve the efficiency of data encryption, we use key encapsulation mechanism (KEM) to encrypt original data. That is, the original data is first encrypted by a chosen symmetric key (SEK), then SEK is encrypted under a chosen access structure associated with attributes. Only the DUE whose attributes match the access structure can recover SEK and further decrypt the encrypted data. Different from most existing MA-CP-ABE schemes [4]–[7], each BS in our RSDAC manages the whole attribute universe, handles the DUE legitimacy verification and generates the intermediate key for legal DUE according to its attribute set. The CNS is in charge of the registration of BSs and DUEs, and generates the private key for each DUE basing on its intermediate key. In summary, we make the following contributions:

1. To solve the issue of single-point bottleneck, the DUE legitimacy verification is separated from the private key generation. Every BS could independently verify the legitimacy of a DUE. We use an additional randomly chosen parameter to remove the restriction in [8] where the timestamp numbers should be different and not been used before.

2. RSDAC supports exponentially large attribute universe and constant size of system public parameters. We design a method to alleviate the user decryption cost by outsourcing the most complicated decryption operations to a third party (such as the DUE with sufficient computation resource). The correctness of returned partial decryption ciphertext from the third party can also be efficiently checked.

3. RSDAC supports any monotonic access structure. The security analysis and performance results demonstrate that RSDAC is secure, efficient and applicable.

II. RELATED WORK

A. DATA ACCESS CONTROL IN D2D COMMUNICATIONS

Most extensive works [9]–[13] focused on interference management and resource allocation. Aiming to realize confidentiality and integrity for D2D communication in LTE-Advanced, Zhang et al. [3] presented a data sharing protocol by using signature and public key technique. However, the content providing server which administrates the register of all the devices might be a security and performance bottleneck of the system. Kwon et al. [14] showed that how to adopt ciphertext-policy ABE (CP-ABE) to design a D2D authentication protocol, where a group manager should be available. Huang et al. [15] and Yue et al. [16] investigated the fine-grained access control in cellular communication networks, where the connection between only UEs was not considered. Yan et al. [17] realized flexible data access control among lots of devices in D2D communication by employing ABE, where the attributes are described by two-dimensional trust levels. However, their scheme cannot support multiple authorities and the decryption cost is linear with the scale of involved attributes.

B. ATTRIBUTE-BASED ENCRYPTION AND ITS APPLICATIONS

Various approaches [18]–[21] have been employed to preserve user privacy and data security in practice. As one of the most promising cryptographic techniques, ABE has been regarded as an important building block to design fine-grained access control systems.

ABE was first introduced in [22] and further classified to two types: key-policy ABE (KP-ABE) [23]–[25] and CP-ABE [26]–[30]. Different from CP-ABE, KP-ABE associates the ciphertext with attributes and the private key with the access structure. In [31], Yu et al. first adopted ABE to design fine-grained access control scheme for cloud computing. Since then, various data access control schemes based on ABE have been introduced.

Aiming to resolve the problem of single-point bottleneck, Xue et al. [8] proposed a new MA-ABE mechanism where the operation of user legitimacy verification is moved to the attribute authorities (AAs), and every AA can execute the user legitimacy verification by itself and generate intermediate key over the whole attribute universe. The randomness of private keys and collusion resistance rely on the difference of timestamp at that moment. To ensure the timestamp numbers are unique, the CA has to check the timestamp numbers are in the pre-defined time interval. Such method may bring additional computation cost for CA and the delay of key generation.

The large universe problem was first addressed in [22]. On composite order groups, Lewko et al. [32] introduced the first exponentially large universe KP-ABE scheme, whereafter Rouselakis et al. [33] demonstrated how to construct large universe ABE on prime order groups.

To realize efficient user decryption in ABE, Green et al. [34] introduced a decryption outsourcing method to offload most decryption operation to a third-party, which then returns a partial decryption ciphertext (PDC). Only one time of exponential operation on PDC is required by the user to recover the plaintext. However, the correctness of PDC can not be guaranteed. Lai et al. [35] designed a verification method to check the correctness of PDC. The ciphertext length and the
encryption cost are almost twice of that in [34]. Ning et al. [36] presented an auditable CP-ABE scheme without adding any extra encryption overhead or ciphertext element, where the PDC is verified by taking in the system master secret key.

III. SYSTEM MODEL, ADVERSARY MODEL AND SECURITY REQUIREMENTS

A. SYSTEM MODEL

Fig. 1 describes the system model of RSDAC, which consists of four entities: core network server (CNS), base stations (BSs), cellular UEs (CUEs) and D2D UEs (DUEs). In RSDAC, the DUE can connect to the BS which covers it directly or by the relay of a CUE. In particular, we call a DUE the data user (DU) if it is the receiver of some data. In additionally, a DUE or CUE with sufficient computation resource can serve as the outsourced decryption service provider (ODSP) for the DUs. The detailed function of each entity is given as follows:

CNS: CNS is a trusted central authority, which is in charge of initializing the system and generating the corresponding parameters. It also accepts the registration of the BSs and DUs. It labels each BS with a unique \( Bid \) and each DU with a unique \( Uid \). Meanwhile, it creates the public-private key pairs for the BSs and DUs. Additionally, it also creates the final private key for each DU by employing the intermediate key \( IK \) generated by a BS. If necessary, CNS can help DUs check the correctness of PDC.

BS: Every BS is in charge of verifying the legitimacy of a DU. If so, it generates \( IK \) corresponding to the DU’s attribute set. Note that every BS in our system governs the whole attribute universe rather than a disjoint attribute subset which was introduced in prior works [4], [5], [24], [37]. The \( DU_E_4 \) is covered by \( BS_1 \) and \( BS_N \) as in Fig. 1, it can obtain the \( IK \) from either \( BS_1 \) or \( BS_N \).

DO: DO chooses a symmetric encryption key (\( SEK \)) to encrypt its data. Then the DO defines an access structure under which \( SEK \) is encrypted. Finally, the encrypted data along with the ciphertext of \( SEK \) will be shared with the DUs.

DU: Each DU is assigned a unique \( Uid \) by the CNS and issued a public key and a user decryption key (\( UDK \)). Each DU can call for the decryption service from the ODSP by submitting his private key. The DU can also call the CNS to check whether the returned PDC is correctly computed. If so, he can recover \( SEK \) by the \( UDK \) and further decrypt the encrypted data.

ODSP: ODSP could help the DU pre-decrypt a ciphertext according to its private key. If the DU’s attributes match the access structure in \( SEK \) ciphertext, the ODSP will return a PDC.

B. ADVERSARY MODEL AND SECURITY REQUIREMENTS

In RSDAC, CNS is fully trusted. We assume that the BSs could be compromised and they may collude with each other to obtain the \( MSK \). The ODSP is honest-but-curious. That is, it executes its task honestly, but it would try to get as much information as possible of the encrypted data. The DUs might be malicious by colluding with each other to obtain extra access privilege that none of them has.

Concretely, we consider the following security requirements:

1. Fine-grained access control. In order to indicate who is authorized to access its data, the DO should be enabled to define flexible access structure.
2. Data Confidentiality. The data must be confidential to unauthorized access from both unauthorized DUs and ODSP.
3. DUs Collusion Resistance. The malicious DUs may combine their private keys to get access to the ciphertext that none of them is allowed. Such collusion resistance should be resisted.
4. BSs’ Ultra Vires Resistance. The BS can not directly issue private keys for the DUs. That is, the BS could not obtain the \( MSK \) of the system, even if it colludes with the others.
5. Verifiability. Once the ODSP returns a wrong or invalid partial decryption ciphertext, such malicious behavior must be efficiently detected.

IV. PRELIMINARIES

A. BILINEAR MAPS

\( G \) and \( G_1 \) refer to two multiplicative cyclic groups of prime order \( p \). \( \eta \) refers to a generator of \( G \). \( e: G \times G \rightarrow G_1 \) is called a bilinear map if:

1. Bilinearity: \( e(\xi^x, \xi^y) = e(\xi^z) \) \( \forall \xi, \xi \in G \) and \( x, y \in \mathbb{Z}_p \);
2. Non-degeneracy: \( e(\eta, \eta) \neq 1 \) for \( g \).
3. Symmetric: \( e(\eta^x, \eta^y) = e(\eta, \eta^{xy}) = e(\eta^y, \eta^x) \).

B. LINEAR SECRET SHARING SCHEME (LSSS)

Definition 1. A secret sharing scheme II over a set of parties \( P \) is linear (over \( \mathbb{Z}_p \)) if

1. The shares of a secret for each party form a vector over \( \mathbb{Z}_p \).
2. A matrix $A$ with $\ell$ rows and $n$ columns is called the share-generating matrix for $\Pi$. $\rho$ is a function which maps $\{i = 1, \ldots, \ell\}$ to $\mathbb{P}$. While considering the vector $\overrightarrow{v} = (s, r_2, \ldots, r_n)^T$, where $r_2, \ldots, r_n$ are randomly picked from $\mathbb{Z}_p$ and $s \in \mathbb{Z}_p$ is the secret to be shared, then $A\overrightarrow{v}$ is the vector of $\ell$ shares of $s$. The share $(A\overrightarrow{v})_i$ belongs to the party $\rho(i)$.

Every LSSS has the linear reconstruction property [38]. Suppose that $\Pi$ is an LSSS of an access structure $\mathcal{A}(A, \rho)$ and $S \in \mathcal{A}$ is any authorized set. Let $I \subseteq \{1, 2, \ldots, \ell\}$ be $I = \{i : \rho(i) \in S\}$. Then there exist constants $\omega_i \in \mathbb{Z}_p$, such that, if $\lambda_i = (A\overrightarrow{v})_i$ are valid shares of $s$, then $\sum_{i \in I} \omega_i \lambda_i = s$.

### C. h-TYPE ASSUMPTION

Choose a generator $\eta$ from group $\mathbb{G}$ of prime order $p$. Randomly pick $h + 2$ exponents $x, s, y_1, y_2, \ldots, y_n \in \mathbb{Z}_p$. If an adversary is given $(p, G, G_1, e : G \times G \rightarrow G_1)$ and all of the following elements:

$$
\begin{align*}
\mathcal{N} & = \{\eta, \eta^x, \eta^{y_1}, \eta^{y_2}, \eta^{y_3}, \eta^{y_j}, \eta^{y_i/j}, \eta^{y_i/s}, \eta^{2y_i}, i, j \in \{h, h\} + 1 \} \\
& \quad \eta^{x/k}, \eta^{y_j}, i, j \in \{h, h, h\} \text{with } i \neq j + 1 \\
& \quad \eta^{y_i/j}, \eta^{y_j}, i, j \in \{h, h, h\} \text{with } i \neq j + 1 \\
& \quad \eta^{y_i/s}, \eta^{y_j}, i, j \in \{h, h, h\} \text{with } i \neq j + 1
\end{align*}
$$

The advantage with which an algorithm $B$ can solve the above decisional $h$-type problem is defined as $Adv_B(\lambda) = | Pr[B(\mathcal{N}, R = e(\eta, \eta)^{x+k}) = 0] - Pr[B(\mathcal{N}, R = R) = 0] |$, where $e(\eta, \eta)^{x+k} \in G_1$ and $R \in G_1$ is randomly selected.

**Definition 2.** The $h$-type assumption holds if $Adv_B(\lambda)$ is negligible of $\lambda$ for all probabilistic polynomial time (PPT) adversaries.

### D. DEFINITION OF MA-CP-ABE WITH VERIFIABLE OUTSOURCED DECRYPTION

A MA-CP-ABE scheme with verifiable outsourced decryption is comprised of the following nine algorithms:

**Global Setup** $(\lambda, U) \rightarrow (GPK, MSK)$: A CA runs this algorithm by taking in a security parameter $\lambda$ along with the system attribute universe $U$. It outputs the global public parameters $GPK$ and master secret key $MSK$.

**AA Setup** $(GPK, AAid) \rightarrow (PK_{AAid}, SK_{AAid})$: On input the AA’s identifier $(AAid)$ and $GPK$, this AA setup algorithm outputs this AA’s public key $PK_{AAid}$ and private key $SK_{AAid}$.

**User Setup** $(GPK, Uid) \rightarrow (PK_{Uid}, UDK_{Uid})$: On input $GPK$ and a user’s identification information $Uid$, this user setup algorithm outputs this user’s public key $PK_{Uid}$ and the corresponding user decryption key $UDK_{Uid}$.

**AA KeyGen** $(S, GPK, SK_{AAid}) \rightarrow (IK)$: On input an attribute set $S$, $GPK$ and $SK_{AAid}$, this AA key generation algorithm outputs the intermediate key $IK$. of $S$.

**CA KeyGen** $(IK, GPK, MSK, PK_{AAid}) \rightarrow (SK)$: On input $IK$, $GPK$, $MSK$ and $PK_{AAid}$, this CA key generation algorithm outputs the final private key $SK$.

**Encrypt** $(GPK, A, M) \rightarrow (CT)$: On input $GPK$, an access structure $A$ and a message $M$, this encryption algorithm outputs a ciphertext $CT$.

**Transform** $(GPK, SK, CT) \rightarrow (PDC)$: On input $GPK$, $SK$ and $CT$, if $S$ matches $A$, this transformation algorithm outputs a partial decryption ciphertext $PDC$.

**Verify** $(PK_{Uid}, MSK, CT, PDC) \rightarrow 1$ or $0$: The verify algorithm takes in $PK_{Uid}$, $MSK$, $CT$, and $PDC$. If $PDC$ is correctly computed, it outputs $1$. Otherwise, it outputs $0$.

### E. SECURITY MODEL

The security definition of our MA-CP-ABE is given by the following game between an adversary $A$ and a challenger $B$. Identical to the security model in [8], [33], in our game, the challenge access structure chosen by $A$ has to be declared before initializing $GPK$.

**Initialization.** $A$ specifies the challenge access structure $\mathcal{A}^*$. 

**Setup.** By running the **Global Setup**, **AA Setup** and **User Setup** algorithms, $B$ generates the corresponding parameters and transmits the public parameters to $A$.

**Phase 1.** $A$ can make key queries of the attribute sets $S_1, S_2, \ldots, S_{q_1}$, such under restriction that none of the sets can match $\mathcal{A}^*$. 

**Challenge.** $A$ submits two massages $M_0$ and $M_1$ with equal length. $B$ encrypts $M_0$ under $\mathcal{A}^*$ and gets a ciphertext $CT^*$, where $b$ is randomly picked from $\{0, 1\}$. Then $CT^*$ is transmitted to $A$.

**Phase 2.** $A$ acts the same as in Phase 1.

**Guess.** $A$ guesses $b'$ on $b$. The advantage of $A$ in the above scheme is defined as $| Pr[b' = b] - 1/2 |$.

**Definition 3.** A MA-CP-ABE scheme is selectively secure if $| Pr[b' = b] - 1/2 |$ is negligible for any PPT adversary.

### F. VERIFIABILITY

Similarly, the verifiability model of our MA-CP-ABE is defined by the following security game between $A$ and $B$.

**Setup.** $A$ and $B$ act the same as in the above security game.

**Phase 1.** $A$ can request the private keys the same as in the above security game.

**Challenge Phase.** Once receiving the challenge $\mathcal{A}^*$ from $A$, $B$ gets $CT^*$ from the **Encrypt** algorithm and transmits it to $A$.

**Phase 2.** Similar to Phase 1.

**Output.** $A$ outputs an attribute set $S^*$ along with two partial decryption ciphertext $PDC_1^{\ast}$ and $PDC_2^{\ast}$ for $CT^*$. $A$
wins if the entry \((PDC_1^*, PDC_2^*)\) can pass Verify, and Decrypt \((PDC_1^*, UDK_{UId}) \neq \text{Decrypt}(PDC_2^*, UDK_{UId})\).

Definition 4. Our MA-CP-ABE is verifiable if no PPT adversary can get a non-negligible advantage in the above game.

V. PROPOSED SCHEME

This section presents the detailed construction of the proposed RSDAC. Table 1 gives the description of notations employed in this scheme.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNS</td>
<td>core network server</td>
</tr>
<tr>
<td>BS</td>
<td>base station</td>
</tr>
<tr>
<td>DO</td>
<td>data owner</td>
</tr>
<tr>
<td>DU</td>
<td>data user</td>
</tr>
<tr>
<td>ODSP</td>
<td>outsourced decryption service provider</td>
</tr>
<tr>
<td>LSSS</td>
<td>linear secret sharing scheme</td>
</tr>
<tr>
<td>Bid</td>
<td>identifier of a BS</td>
</tr>
<tr>
<td>Uid</td>
<td>identifier of a DU</td>
</tr>
<tr>
<td>GPK</td>
<td>system global public parameters</td>
</tr>
<tr>
<td>MSK</td>
<td>system master secret key</td>
</tr>
<tr>
<td>PK_{Bid}, SK_{Bid}</td>
<td>public key and secret key of Bid</td>
</tr>
<tr>
<td>SEK</td>
<td>a symmetric key used to encrypt original data</td>
</tr>
<tr>
<td>ENKEM</td>
<td>data encrypted by SEK</td>
</tr>
<tr>
<td>CT</td>
<td>ABE ciphertext of SEK</td>
</tr>
<tr>
<td>(h)</td>
<td>access structure</td>
</tr>
<tr>
<td>((A, \rho))</td>
<td>(h) expressed by LSSS matrix (A) and map (\rho)</td>
</tr>
<tr>
<td>UDK_{UId}</td>
<td>user decryption key of Uid</td>
</tr>
<tr>
<td>S_{UId}</td>
<td>attribute set of Uid</td>
</tr>
<tr>
<td>IK_{UId}</td>
<td>intermediate key for S_{UId}</td>
</tr>
<tr>
<td>SK_{UId}</td>
<td>final private key for S_{UId}</td>
</tr>
<tr>
<td>PDC</td>
<td>partial decryption ciphertext</td>
</tr>
</tbody>
</table>

Table 1: Notations employed in RSDAC

Encrypt. The DO performs the data encryption algorithm as follows: DO defines an LSSS access structure \(A(\rho, \lambda)\), where \(A\) refers to a \(\ell \times n\) matrix and \(\rho\) maps each row \(A_i\) to an attribute. DO randomly picks \(s, v_1, \ldots, v_n\) from \(\mathbb{Z}_p\) and sets a vector \(\vec{v} = (s, v_1, \ldots, v_n)^T\). In turn, it computes \(\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_\ell)^T = A \cdot \vec{v}\) and \(C_0 = \eta^s\). For each \(\tau \in \{1, 2, \ldots, \ell\}\), it randomly picks \(\tau_\tau \in \mathbb{Z}_p\) and computes:

\[
C_{\tau,1} = (\lambda_1 \eta^{\tau_1} v_1)^{\tau_\tau},\ C_{\tau,2} = (\eta^{\lambda_2} \psi)^{\tau_\tau} \text{ and } C_{\tau,3} = \eta^{\tau_\tau}.
\]

The SEK in KEM is set as \(e(\eta, \eta)^{\alpha_\tau}\) and the encrypted data is denoted as \(EN_{KEM}\). The ciphertext of SEK is

\[
CT = (C_0, \{C_{\tau,1}, C_{\tau,2}, C_{\tau,3}\}_{\tau \in \{1, 2, \ldots, \ell\}}).
\]

C. USER KEY GENERATION

User Setup. The new joined DU has to register itself from the CNS. For each DU, CNS assigns a unique identification \(UId\) and randomly picks \(c_{UId} \in \mathbb{Z}_p\). It then sets the DU’s public key \(PK_{UId} = \eta^{c_{UId}}\). Finally, CNS gives \(PK_{UId}\) and the corresponding user decryption key \((UDK = c_{UId})\) to the user with identity \(UId\).

BS KeyGen. When receiving the private key request from a DU with identity \(UId\), the BS first checks if the DU’s \(UId\) has the specified attribute set \(S_{UId}\) that it claimed as in [8]. If not, BS_i submits the identity information of \(UId\) to CNS which may subsequently kick this user out. Otherwise, BS_i works as follows:

Firstly, \(BS_i\) queries the current timestamp value \(TSV\) and calculates \(t_1 = H(UId||TSV||0)\) and \(t_2 = H(UId||TSV||1)\).

Secondly, for each \(AT \in S_{UId}\), BS_i randomly picks \(a_j \in \mathbb{Z}_p\) and computes:

\[
\Gamma_{\tau,1} = \eta^{b_{A_t,0} a_j t_1},\ \Gamma_{\tau,2} = \eta^{a_j t_2},\ \Gamma_{\tau,3} = (\eta^{\lambda_2} \psi)^{a_j t_1} v_1^{-k_{Bid}^{a_j t_1}},\ \Gamma_{\tau,4} = (\eta^{\lambda_2} \psi)^{a_j t_2} v_2^{-k_{Bid}^{a_j t_2}}.
\]

The intermediate key of \(S_{UId}\) is set as \(IK_{UId} = \{\Gamma_{1,1}, \Gamma_{1,2}, \Gamma_{2,3}, \Gamma_{2,4}\}_{AT \epsilon S_{UId}}\).

Finally, the terms: \((UId, Bid_i, S_{UId}, TSV, IK_{UId})\) are securely sent to CNS.

CNS KeyGen. After receiving the terms from \(BS_i\), CNS checks if the transmission delay is appropriate. If so, CNS refuses to accept the terms. Otherwise, CNS works as follows:

Firstly, CNS obtains \(PK_{UId}\) and \(PK_{Bid}\) by \(UId\) and \(Bid_i\). Then it sets \(t_1 = H(UId||TSV||0)\) and \(t_2 = H(UId||TSV||1)\).

Secondly, CNS randomly chooses \(d \in \mathbb{Z}_p\) and uses \(MSK\) to create the private key \(SK_{UId}\):

\[
\Gamma_0 = (PK_{UId})^{\alpha_\tau} (w^{k_{Bid}})^{d^{t_1}} u^{d_{t_2}} = \eta^{c_{UId} u^{k_{Bid} t_1} + d_{t_2}}\ \Gamma_1 = (\eta^{k_{Bid}})^{d^{t_1}} \eta^{d_{t_2}} = \eta^{k_{Bid} t_1 + d_{t_2}}
\]

For each \(AT \in S_{UId}\), compute:

\[
\Gamma_{2,1} = (\eta^{k_{Bid} t_1})^{d^{t_1}} \Gamma_{2,2} = (\eta^{k_{Bid} t_2})^{d^{t_2}} = (\eta^{k_{Bid} t_1} + d_{t_2})
\]

\[
\tau_{\tau,3} = (\Gamma_{2,1} \psi)_{\tau,1}^{\tau_{\tau,4}} = (\eta^{k_{Bid} t_1 + d_{t_2}})_{\tau,1}^{\tau_{\tau,4}}
\]

For simplicity, we let \(r = k_{Bid}^{d_{t_1} + d_{t_2}} + r = k_{Bid}^{d_{t_1} + d_{t_2}}\).
Therefore, $SK_{Uid}$ can be denoted as:

$$
\begin{align*}
\mathcal{Y}_0 &= \eta^{\alpha S_{c Uid}} \cdot w^r \\
\mathcal{Y}_1 &= \eta^r \\
\forall A T_r \in S_{Uid},
\mathcal{Y}_{r,2} &= \eta^r \\
\mathcal{Y}_{r,3} &= (g^{AT(r)} \psi)^{r} v^{-r}
\end{align*}
$$

Finally, $SK_{Uid} = (\mathcal{Y}_0, \mathcal{Y}_1, \{\mathcal{Y}_{r,2}, \mathcal{Y}_{r,3}\}_{\forall A T_r \in S_{Uid}})$ is sent to the DU via $BS_i$.

### D. DECRYPTION AND VERIFICATION

**Transform.** After receiving $EN_{K,EM}$ and $CT$ from DO, DU can request the ODSP to decrypt the data that it wants to access by submitting its attribute set $S_{Uid}$, $SK_{Uid}$ and $CT$. If $S_{Uid}$ satisfies $A(\alpha, \beta)$, the ODSP works as follows:

Set $X = \{x : p(x) \in S_{Uid}\}$ and compute such coefficients $\{\mu_x \in \mathbb{Z}_p\}_{x \in X}$ satisfying $\sum_{x \in X} = \mu_x \lambda_x = s$. Then compute

$$
PDC = \prod_{x \in X} (\mathcal{Y}(C_x, \mathcal{Y}_0) = e(\eta, \eta)^{\alpha S_{sec} \cdot \eta^{\alpha S_{sec} Uid}}) = e(\eta, \eta)^{\alpha S_{sec}}
$$

The ODSP sends $PDC$ to the DU.

**User Decryption.** The DU can recover $SEK$ by computing $SEK = PDC^{1/UDK} = e(\eta, \eta)^{\alpha S_{sec} \cdot \eta^{\alpha S_{sec} Uid}} = e(\eta, \eta)^{\alpha S_{sec}}$.

**Verification.** After receiving $(PDC, PK_{Uid}, C_0)$, the CNS checks if the following equation holds: $e((PK_{Uid})^\alpha, C_0) = PDC$. If so, CNS outputs 1 to indicate that the ODSP computes $PDC$ correctly. Otherwise, it outputs 0 to indicate that the ODSP does not correctly compute $PDC$.

### VI. SECURITY ANALYSIS

#### A. FINE-GRANED ACCESS CONTROL

In RSDAC, the attribute universe is exponentially large. The DO can define arbitrary monotonic access structure over descriptive attributes, to indicate who has the access privilege to its data. Moreover, if a DO receives and stores the system public parameters on its device, then it can independently encrypt its data under the access structure, no matter it has connected to a BS or not.

#### B. DATA CONFIDENTIALITY

The data confidentiality of RSDAC is proved by the following theorem:

**Theorem 1.** Assume the $h$-type assumption holds, then our RSDAC is selectively secure.

**Proof.** Recall that the DU’s private key $SK_{Uid}$ is in the form of:

$$
\begin{align*}
\mathcal{Y}_0 &= \eta^{\alpha S_{c Uid}} \cdot w^r \\
\mathcal{Y}_1 &= \eta^r \\
\forall A T_r \in S_{Uid},
\mathcal{Y}_{r,2} &= \eta^r \\
\mathcal{Y}_{r,3} &= (g^{AT(r)} \psi)^{r} v^{-r}
\end{align*}
$$

where $k_{bid}(d + a, d + t_1 + d + t_2)$ and $k_{bid}(d + b d \alpha, d + a \beta)$ are simplified as $r$ and $r$, respectively. Meanwhile, because of the randomly chosen $a_j$ and $d$, and $r$ can be seen as totally random numbers. Thus, this theorem can be proved similarly to that in [36], where the details of proof are given. Theorem 1 holds means that the ciphertext is confidential to the DU if its attributes do not match $h(A, \rho)$ in $CT$.

Moreover, even if the ODSP obtains the user’s private key while providing outsourced decryption service, the encrypted data remains secret since that the user decryption key $UDK$ is not given to the ODSP.

#### C. USER COLLUSION RESISTANCE

By combining their private keys, the malicious DUs may attempt to recover $SEK = e(\eta, \eta)^{\alpha S}$ that none of them can independently do. Unfortunately, they will fail due to the fact that each DU’s private key elements are bounded by a unique chosen number $d$. Since $d$ is chosen by CNS and is unknown to the DUs, it remains impossible for colluding DUs to access unauthorized data.

Different from the scheme [8], the randomness of the DU’s private key in RSDAC not only relies on the timestamp numbers $t_1$ and $t_2$, but also the unique number $d$. Thus, there is no requirement of employing extra master secret key $b$ and computing $\eta^{- (t_1 + t_2)}$ and $\eta^{(t_1 + t_2)}$ as in [8]. Moreover, a malicious DU in our RSDAC can not deduce any useful element from his private key to gain additional access privilege.

#### D. BSS’ ULTRA VIRES RESISTANCE

The BSs may collude with each other to gain the system secret information about $\alpha$ and $\beta$. In [8], the authors showed how the colluded BSs act. Suppose $BS_1$ and $BS_2$ choose the same terms $(t_1, t_2)$, $\eta^a$ can be computed by these two BSs, which then can generate any effective private key and access any encrypted data. Such collusion attacks are resisted by ensuring the terms $(t_1, t_2)$ are different and never used before. However, in RSDAC, we introduce an additional parameter $d$ which is unique for every attribute set. Even if $BS_1$ and $BS_2$ set the same terms $(t_1, t_2)$, they can not cancel $d$ in the exponents because of the different $d_i$ for each attribute set. Thus, the colluded BSs can not obtain any useful information of $\eta^a$.

#### E. VERIFIABILITY

**Theorem 2.** For all PPT adversaries, the advantage is at most negligible in the verifiability security game.

**Proof.** We assume there exists an adversary $A$ which can break the verifiability of our scheme, then a simulator $B$ can be built to interact with $A$ as follows:

**Setup.** $B$ initializes the system and sets the system parameters $GPK$, $MSK$ and $\{(PK_{Bid}, k_{Bid})\}$ as in the real scheme. It then sends $GPK$ and $\{(PK_{Bid}, k_{Bid})\}$ to $A$.

**Phase 1.** $A$ can query the keys of attribute sets $S_1, \ldots, S_{q_1}$. $B$ then generates the corresponding private keys and sends them to $A$. 
Challenger. A declares a challenge LSSS access structure \( A^*(A^*, \rho) \). B obtains a challenge ciphertext \( CT^* \) by running Encryption and transmits it to \( A \).

**Phase 2. Same as Phase 1.**

Output. A has to output two partial decryption ciphertexts \( PDC^*_1 \) and \( PDC^*_2 \) of \( CT^* \).

A wins if the following 3 conditions are fulfilled simultaneously.

1. **Verification** outputs 1 on \( PDC^*_1 \).
2. **Verification** outputs 1 on \( PDC^*_2 \).
3. User Decryption (\( PDC^*_1, UDK_{U(1d)} \)) \( \neq \) User Decryption (\( PDC^*_2, UDK_{U(1d)} \)).

From (1) and (2), we have \( PDC^*_1 = PDC^*_2 \). However, condition (3) means that \( PDC^*_1 \neq PDC^*_2 \). Thus, A has only negligible advantage to win the above game.

**VII. PERFORMANCE ANALYSIS**

**A. FEATURES COMPARISON**

Table 2 compares some features between previous related CP-ABE works and RSDAC, involving multi-authority, robust AA/BS, large universe, access structure, outsourced decryption and verifiability.

From Table 2, we can see that only the scheme in [8] and RSDAC achieve the robust AA/BS. That is, each AA/BS is in charge of governing the whole attribute universe of system. Except the schemes in [6], [8], [17], the other schemes can support large attribute universe. The user decryption overhead in [8], [17], [33] increases with the number of used attributes. On the contrary, the user decryption overhead in [6], [7], [36] and RSDAC is constant size by employing the outsourced decryption technique. Only RSDAC and the scheme in [36] enable the users to check the correctness of PDC. However, the scheme in [36] only supports the single authority, without considering multiple attribute authorities. In general, RSDAC is the only one which simultaneously achieves the promising features mentioned above.

**B. NUMERAL COMPARISON**

In Table 3, we compare the large universe schemes [7], [33], [36] and RSDAC, in terms of the size of system public parameters (PK), user’s private keys (USK), ciphertext (CT) and the entry sent for verifying (ESV). Different form the AA’s parameters \( \{APK_f\} \) in the scheme [7], the BS’s parameters \( \{PK_{B(id)}\} \) in our scheme is only used by the CNS and will not be involved in the encryption phase. Thus, we do not record the size of \( \{PK_{B(id)}\} \) in the size of PK. In Table 3, \( |G| \) and \( |G_1| \) refer to an element in \( G \) and \( G_1 \), respectively. \( S_E \) and \( S_U \) refer to the related attribute sets involved in the CT and SK, respectively. Besides, \( S_A \) denotes the set of attribute authorities.

Table 3 shows that the PK size of RSDAC is the same as in the schemes [33], [36], which is less than that in [7]. Except that in the scheme in [36], the size of USK is the same in other schemes. Specially, the size of CT and ESV in [36] and RSDAC is the same. RSDAC does not add any extra elements while achieving multi-authority and robust BSs. Thus, the proposed RSDAC is considerable and applicable in D2D communications.

**C. IMPLEMENTATION RESULT**

We implement our scheme, the NCDLMW scheme [36] and the LZ scheme [7]. All these three schemes are constructed on the large universe scheme [33]. The implementation is performed on a Ubuntu 18.04 LTS system (with 3.40GHz Inter Core i7 CPU and 8.00GB RAM), based on the JPBC library 2.0.0 [39]. We employ a Type A pairing which is constructed over a symmetric elliptic curve \( \alpha \)-curve with 160-bit prime order \( p \).

In Fig. 2, we evaluate the computation cost during the phase of encryption, key generation and user decryption. Each simulation result is the average of 30 trials.

Fig. 2(a) and Fig. 2(d) demonstrate that the time of encryption and user decryption in RSDAC is almost the same as that in NCDLMW scheme and LZ scheme. More precisely, the encryption overhead of these three schemes is linear with the scale of access structure. The only difference is that the
element $e(\eta, \eta)_{\alpha s}$ in NCDLMW scheme and RSDAC is set as $SEK$ in KEM, while $e(\eta, \eta)_{\alpha s}$ is used to encrypt the data encryption key in LZ scheme. Additionally, the user decryption in these three schemes only costs one time of exponential operation due to the usage of outsourced decryption technique [34].

Fig. 2(b) and Fig. 2(c) show that our scheme requires more computation time in the phase of AA/BS key generation and CA/CNS key generation than that in NCDLMW scheme and LZ scheme. This is due to the fact that we use the method to avoid the single-point bottleneck as in [8], where the computation cost for each attribute is twice of that in the original scheme [40].

VIII. CONCLUSION

In this work, we have addressed three major problems of data security, identity privacy and system scalability in D2D communication, by presenting a data access control scheme for D2D communication with robust multiple authorities, large attribute universe and verifiable outsourced decryption. In particular, each of multiple BSs can complete the task of DU legitimacy verification individually. Different from most prior multi-authority works, each BS can generate intermediate attribute keys according to arbitrary subset of whole system attribute universe. Such keys would be employed by the CNS to create the finally private keys, we also provided an efficient approach to help DUs offload most decryption overhead to a third device and check whether the device has correctly computed. The security analysis, numeral comparison and experimental results showed that RSDAC is secure, efficient and applicable in D2D communication scenario.

Although the verification of DU legitimacy is offloaded to the BSs, the single CNS remains has to create the final private keys for all DUs in the system. It would be interesting to design a more efficient key generation algorithm, where multiple CNSs exist and each of them can independently finish the generation of final private keys.

REFERENCES
