SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks (2016)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.1446

SPECIAL ISSUE PAPER

Verifiable attribute-based proxy re-encryption for secure public cloud data sharing

Suqing Lin[1,2], Rui Zhang[1,2] and Mingsheng Wang[1,2]*

[1]State Key Lab of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
[2]University of Chinese Academy of Sciences, Beijing 100049, China

ABSTRACT

For secure data sharing in the public cloud, attribute-based encryption was introduced to simultaneously achieve data confidentiality and fine-grained access control. In order to update access control of the attribute-based encrypted data from delegation, attribute-based proxy re-encryption (AB-PRE) was proposed accordingly. Most previous AB-PRE schemes require that the proxy executes the re-encryption honestly. However, the public cloud as a proxy may not meet the requirement because the encrypted data are delegated to the public cloud and out of control for data owners. In this paper, we introduce verifiability for AB-PRE to check the correctness of the re-encryption executed by the proxy. By introducing a commitment scheme and a key derivation function, we propose a generic construction of unidirectional single-hop AB-PRE with verifiable re-encryption (AB-VPRE) for both key-policy and ciphertext-policy settings, and the access structure can be monotonic and non-monotonic. We prove the security and the verification soundness of our constructed AB-VPRE scheme in the standard model and provide three instantiations. Compared with previous work on AB-PRE, our proposed AB-VPRE schemes require less computation and can efficiently detect the malicious behaviors of the proxy.

KEYWORDS
attribute-based proxy re-encryption; verifiability; master key security

***Correspondence**
Mingsheng Wang, State Key Lab of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China.
E-mail: wangmingsheng@iie.ac.cn

1. INTRODUCTION

With the rapid development of cloud applications, more and more users require data storage and computation services in the public cloud. Data owners outsource their data to the public cloud for sharing with others, and security of the data sharing is a problem of extreme concern. Attribute-based encryption (ABE) [1] was then introduced to achieve data privacy and fine-grained access control simultaneously.

If the attribute-based encrypted data in the public cloud is in urgent need while all the qualified users are unavailable, proxy re-encryption is a good help to re-encrypt the data from delegation without revealing any confidential information. Traditional proxy re-encryption in the public key cryptosystem allows a proxy to translate one ciphertext into another with a different public key. Employing proxy re-encryption into attribute-based cryptosystem, Liang *et al.* [2] proposed attribute-based proxy re-encryption (AB-PRE). According to which ciphertexts or private keys that access policies are associated with, AB-PRE is divided into ciphertext-policy (CP) AB-PRE and key-policy (KP) AB-PRE. CP (or KP) AB-PRE delegates a third party to translate the data encrypted under one access policy (or set of attributes) to another one such that access control of the encrypted data will be changed. Hence, AB-PRE is a promising technique to achieve access control update from delegation. Specifically, qualified users can delegate the public cloud to re-encrypt the ciphertext under the guarantee of the security of private information.

An application of CP-AB-PRE is the electronic medical record system where patients' medical records are encrypted associated with access policies (e.g., "Orthopedics" **AND** "Director" **AND** "Union Hospital") before outsourcing to the cloud. Doctors obtain private keys according to their attributes (e.g., "Department" and "Title"). If a patient is in an emergency and no qualified doctors are available, CP-AB-PRE is needed to translate the encrypted records into that under another access policy such that the records can be obtained in time.

An application of KP-AB-PRE is secure forensic analysis for the audit log that includes detailed accounts of activities on the system or network to be protected [3]. If a company intends to upload some audit log entries to the cloud for forensic analysis without leaking secret information, it could encrypt these entries with annotated attributes (e.g., "User Name", "Date", and "Action"). A forensic analyst would be issued a secret key associated with an access policy (e.g., "User Name: Alice" **OR** ("Date: Between 2012-1-1 and 2014-1-1" **AND** "Action: Access the data pertained to designing the new product")). If the specified forensic analyst is absent, he or she can delegate the public cloud to update access control by applying KP-AB-PRE such that the work will not be delayed.

The existing AB-PRE schemes (e.g., [2,4–6]) always assume that the proxy is semi-trusted and executes the re-encryption honestly; however, the public cloud as a proxy is not necessarily honest and may perform unreliable computation due to malfunction, malicious attack, and so on. Previous work on AB-PRE mainly focuses on data privacy and access control rather than taking the re-encryption verification into consideration together. Data privacy and access control guarantee that any adversary is not able to learn any confidential information about the plaintext and the private key, but lack of checking on the correctness of the re-encryption could make the user receive incorrect results from the proxy. Recall the application scenarios described before. Incorrect information about the medical records could cause a serious medical accident, and invalid results analyzed from incorrectly re-encrypted entries may lead to significant economic losses for the company.

Therefore, how to guarantee the correctness of the re-encrypted ciphertexts and detect the re-encryption performed by the proxy for AB-PRE is worthy of attention. In this paper, we investigate the network model and propose a novel technique to solve this issue.

1.1. Our techniques

By combining a commitment scheme and a key derivation function during the encryption, we provide the verification mechanism for the re-encryption executed by an untrusted proxy. Our verification mechanism is divided into two steps: invariant test and correct detection. Invariant test checks for the invariant components of the original ciphertext during the re-encryption. Correct detection that occurs during the decryption helps a user to examine the calculation correctness of the re-encrypted ciphertexts.

As the network model shown in Figure 1, the encrypted data are stored in the public cloud server denoting cloud storage service providers (CSS) for data sharing and the re-encryption is executed by the public cloud server denoting re-encryption cloud service providers (RCS). We note that CSS is assumed to be semi-trusted and to honestly perform the invariant test. If a user requires updating access control of the data, it submits the re-encryption key (re-key) to RCS and CSS sends the original ciphertext to RCS. Next, the re-encrypted ciphertext is returned to CSS and experience the invariant test. If the invariant test fails, an error symbol will be given for storage. Otherwise, the re-encrypted ciphertext will be stored instead. When the qualified users decrypt the re-encrypted ciphertext, the commitment as a component of the ciphertext can be used to verify the re-encryption.

1.2. Our contributions

For AB-PRE applied to update access control of the encrypted data with an untrusted proxy (e.g., the public cloud), the correctness of the re-encrypted ciphertexts returned from the proxy requires to be checked. We

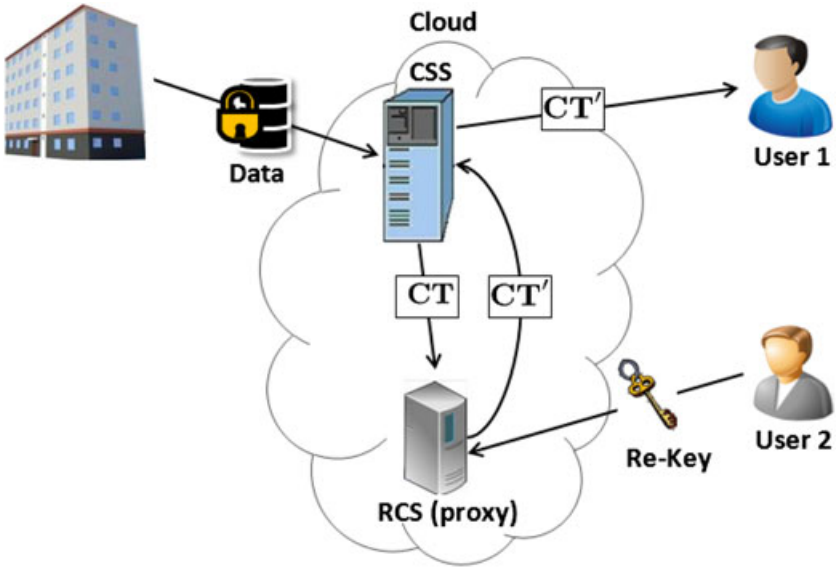

Figure 1. The network model.

DOI: 10.1002/sec

propose a technique to introduce verifiability for AB-PRE to detect the re-encryption executed by the proxy and provide a generic construction of unidirectional single-hop AB-PRE with verifiable re-encryption (AB-VPRE).

We present the model of AB-VPRE and describe definitions for data privacy, the verification soundness, and the weak master key security. The verification algorithms are introduced according to the two steps of invariant test and correct detection. Only if the verification is passed can the decryption algorithm be performed successfully to recover the data. We prove that our constructed AB-VPRE scheme is secure against chosen plaintext attack (CPA secure) and meets the verification soundness in the standard model.

Finally, we instantiate our construction with three AB-KEMs in KP and CP settings, supporting linear-secret-sharing-realizable and non-monotonic access structures. The resulting AB-VPRE schemes inherit the security property of the underlying AB-KEMs and have the weak master key security under the computational Diffie–Hellman (CDH) assumption. Compared with previous AB-PRE schemes, our instantiations of AB-VPRE schemes are relatively more efficient, and the re-encryption executed by an untrusted proxy can be verified.

1.3. Related work

After the concept of ABE [1] was introduced, KP-ABE and CP-ABE schemes [3,7–9] with monotonic and non-monotonic access structures were developed, respectively. For reducing the decryption overhead for resource-constrained users, Green *et al.* [10] proposed outsourcing the decryption of ABE. Then Lai *et al.* [11] presented an ABE scheme with verifiable outsourced decryption.

Proxy re-encryption scheme was first formalized by Blaze, Bleumer, and Strauss [12], and the first unidirectional single-hop PRE scheme was proposed by Ateniese *et al.* [13]. Although several PRE schemes were presented after that, the first generic construction of a secure against chosen-ciphertext attacks (CCA-secure) PRE scheme was proposed by Hanaoka *et al.* [14]. Recently, Ohata *et al.* [15] introduced re-encryption verifiability to the PRE scheme to detect illegal activities of the proxy and showed its CCA security. After the concept of ABE emerged, Liang *et al.* [2] proposed the first CP-AB-PRE scheme based on the CP-ABE scheme [8] supporting non-monotonic access structures. Then Luo *et al.* [16] proposed another CP-AB-PRE scheme with multi-value positive attributes. Apart from this, Yu *et al.* proposed two AB-PRE schemes of KP [17] and CP [18], respectively. Seo *et al.* [19] presented a CP-AB-PRE scheme with constant paring operation latency. Liang *et al.* constructed CCA-secure CP-AB-PRE schemes [4–6] and some other models based on AB-PRE [20].

2. PRELIMINARIES

In this section, we review some useful notations and definitions.

2.1. Notations

Let $\mathbf{A}(u, v, \cdots) \rightarrow w$ ($w \leftarrow \mathbf{A}(u, v, \cdots)$) denote the operation of running an algorithm $\mathbf{A}$ with inputs $(u, v, \cdots)$ and output w. Denote $x \parallel y$ as the concatenation of two strings x and y and $|x|$ as the size of the string x. Let $\ell_{\vec{z}}$ denote the number of elements in the vector $\vec{z}$. $b \in_R S$ denotes the operation of selecting an element b uniformly at random from a set S. Denote $\vec{1}$ as the vector $(1, 0, \cdots, 0)$. Let $\mathbb{N}$ be the set of natural numbers. 1^λ ($\lambda \in \mathbb{N}$) denotes the string of λ ones. Let $\mathbb{R}$ be the set of real numbers. A function $\mathrm{negl} : \mathbb{N} \rightarrow \mathbb{R}$ is negligible if $\forall c > 0, \exists \lambda_0 \in \mathbb{N}$, s.t., $\mathrm{negl}(\lambda) < 1/\lambda^c, \forall \lambda > \lambda_0$.

2.2. Bilinear maps

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplicative cyclic groups of prime order p, and g is a generator of $\mathbb{G}$. Let $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ be an efficiently computable map satisfying that (i) non-degeneracy: $e(g, g) \neq 1$; and (ii) bilinearity: $\forall u, v \in \mathbb{G}$, $\forall a, b \in \mathbb{Z}_p^*$, $e(u^a, v^b) = e(u, v)^{ab}$. We say $(\mathbb{G}, \mathbb{G}_T)$ is a bilinear group pair and e is a bilinear map from $\mathbb{G}$ into $\mathbb{G}_T$.

3. MODEL OF AB-VPRE

We set I_{key} and I_{enc} as the inputs to the key generation and the encryption algorithms, respectively, and define $f(I_{key}, I_{enc}) = 1$ if and only if I_{enc} and I_{key} are matched. An AB-VPRE scheme is defined by the following polynomial-time algorithms:

- Setup($1^\lambda, U$): The setup algorithm takes as input the security parameter λ and the attribute universe U and then outputs the public parameters PP and the master secret key MSK.
- KGen(PP, MSK, I_{key}): The key generation algorithm takes as input PP, MSK, and I_{key} and then outputs a private key SK_I.
- Enc(PP, I_{enc}, M): The encryption algorithm takes as input PP, I_{enc}, and a message M and then outputs a ciphertext CT_I.
- Dec(PP, SK_I, CT_I): The decryption algorithm at original ciphertexts takes as input PP, SK_I, and CT_I and then outputs M if $f(I_{key}, I_{enc}) = 1$, and an error symbol $\perp$, otherwise.
- RKGen(PP, SK_I, I'_{enc}): The re-encryption key generation algorithm takes as input PP, SK_I, and I'_{enc} and then outputs a re-encryption key $RK_{I \rightarrow I'}$.
- REnc($PP, RK_{I \rightarrow I'}, CT_I$): The re-encryption algorithm takes as input PP, $RK_{I \rightarrow I'}$, and CT_I and then outputs a re-encrypted ciphertext $CT_{I'}$ if $f(I_{key}, I_{enc}) = 1$, and an error symbol $\perp$, otherwise.
- REVer.Test($CT_I, CT_{I'}$): The re-encryption test algorithm takes as input CT_I and $CT_{I'}$ and then outputs a bit b.
- REVer.Dec($PP, SK_{I'}, CT_{I'}, \mathsf{b}$): The re-encryption verification and decryption algorithm takes as input PP, $SK_{I'}$, $CT_{I'}$, and a bit b and then outputs M if $\mathsf{b} = 1$, and an error symbol $\perp$, otherwise.

Correctness. For all $(PP, MSK) \leftarrow \text{Setup}(1^\lambda, U)$, $SK_I \leftarrow \text{KGen}(PP, MSK, I_{key})$, $SK_{I'} \leftarrow \text{KGen}(PP, MSK, I'_{key})$, $RK_{I\rightarrow I'} \leftarrow \text{RKGen}(PP, SK_I, I'_{enc})$, $CT_I \leftarrow \text{Enc}(PP, I_{enc}, M)$,

(1) $\text{Dec}(PP, SK_I, CT_I)$ outputs M if $f(I_{key}, I_{enc}) = 1$, and $\perp$ otherwise.
(2) For all $CT_{I'} \leftarrow \text{REnc}(PP, RK_{I\rightarrow I'}, CT_I)$, if $f(I'_{key}, I'_{enc}) \neq 1$, $\text{REVer.Dec}(PP, SK_{I'}, CT_{I'}, \mathsf{b})$ returns $\perp$; otherwise, $\text{REVer.Dec}(PP, SK_{I'}, CT_{I'}, \mathsf{b})$ outputs M if $\text{REVer.Test}(CT_I, CT_{I'}) = 1$.

Before providing the definitions of data privacy, the weak master key security, and the verification soundness for AB-VPRE, we define the key generation oracle $\mathcal{O}_{sk}$, the re-encryption key generation oracle $\mathcal{O}_{rk}$, the re-encryption oracle $\mathcal{O}_{re}$, the decryption oracle $\mathcal{O}_{de}$, and the re-encryption verification and decryption oracle $\mathcal{O}_{rvd}$, respectively, as described in Table I.

CPA security at original ciphertexts. An AB-VPRE scheme is CPA secure at original ciphertexts if for any probabilistic polynomial-time (PPT) adversary $\mathcal{A}$, assisted by the oracle $\mathcal{O} \in \{\mathcal{O}_{sk}, \mathcal{O}_{rk}\}$, the advantage in the following security game is negligible:

$$\begin{aligned}|\Pr[&(PP, MSK) \leftarrow \text{Setup}(1^\lambda, U);\\ &(M_0, M_1, I^*_{enc}, st) \leftarrow \mathcal{A}^{\mathcal{O}}(PP);\\ &b \in_R \{0,1\}; CT^* \leftarrow \text{Enc}(PP, M_b, I^*_{enc});\\ &b' \leftarrow \mathcal{A}^{\mathcal{O}}(CT^*, st) : b = b'] - \tfrac{1}{2}| \leq \text{negl}(\lambda),\end{aligned}$$

where st is the state information and all I_{key} issued to $\mathcal{O}_{sk}$, $\mathcal{O}_{rk}$ satisfies $f(I_{key}, I^*_{enc}) \neq 1$.

CPA security at re-encrypted ciphertexts. An AB-VPRE scheme is CPA secure at re-encrypted ciphertexts if for any PPT adversary $\mathcal{A}$, assisted by the oracle $\mathcal{O} \in \{\mathcal{O}_{sk}, \mathcal{O}_{rk}\}$, the advantage in the following security game is negligible:

$$\begin{aligned}|\Pr[&(PP, MSK) \leftarrow \text{Setup}(1^\lambda, U);\\ &(M_0, M_1, I^*_{enc}, st) \leftarrow \mathcal{A}^{\mathcal{O}}(PP);\\ &SK_I \leftarrow \text{KGen}(PP, MSK, I_{key});\\ &RK_{I\rightarrow I^*} \leftarrow \text{RKGen}(PP, SK_I, I^*_{enc});\\ &b \in_R \{0,1\}; CT_b \leftarrow \text{Enc}(PP, M_b, I_{enc});\\ &CT^* \leftarrow \text{ReEnc}(PP, CT_b, RK_{I\rightarrow I^*});\\ &b' \leftarrow \mathcal{A}^{\mathcal{O}}(CT^*, st) : b = b'] - \tfrac{1}{2}| \leq \text{negl}(\lambda),\end{aligned}$$

where st is the state information, $f(I_{key}, I_{enc}) = 1$ and all I'_{key} issued to $\mathcal{O}_{sk}$, $\mathcal{O}_{rk}$ satisfies $f(I'_{key}, I^*_{enc}) \neq 1$.

Table I. Definitions of oracles.

Oracle	Input	Output
$\mathcal{O}_{sk}$	I_{key}	SK_I
$\mathcal{O}_{rk}$	I_{key}, I'_{enc}	$RK_{I\rightarrow I'}$
$\mathcal{O}_{re}$	CT_I, I_{key}, I'_{enc}	$CT_{I'}$
$\mathcal{O}_{de}$	I_{key}, CT_I	M or an error symbol $\perp$
$\mathcal{O}_{rvd}$	$I'_{key}, CT_I, CT_{I'}$	M or an error symbol $\perp$

Selective CPA security. An AB-VPRE scheme is selectively CPA secure at original ciphertexts (re-encrypted ciphertexts) if we add an Init stage before Setup where the adversary submits the challenge I^*_{enc}.

Weak master key security. An AB-VPRE scheme has the weak master key security if for any PPT adversary $\mathcal{A}$, given access to the oracle $\mathcal{O} \in \{\mathcal{O}_{sk}, \mathcal{O}_{rk}, \mathcal{O}_{de}, \mathcal{O}_{rvd}\}$, the advantage in the following security game is negligible:

$$\begin{aligned}\Pr[&(PP, MSK) \leftarrow \text{Setup}(1^\lambda, U);\\ &(I^*_{key}, SK_{I^*}, RK_{I^*\rightarrow I'}) \leftarrow \mathcal{A}^{\mathcal{O}}(PP);\\ &CT_I \leftarrow \text{Enc}(PP, M, I_{enc}) :\\ &M \leftarrow \text{Dec}(PP, SK_{I^*}, CT_I)] \leq \text{negl}(\lambda),\end{aligned}$$

where $f(I^*_{key}, I_{enc}) = 1$, and any I_{key} issued to $\mathcal{O}_{sk}$ satisfies $I_{key} \neq I^*_{key}$. $RK_{I^*\rightarrow I'}$ is a re-encryption key generated from SK_{I^*} where (I^*_{key}, I'_{enc}) has been issued to $\mathcal{O}_{rk}$.

Selective weak master key security. An AB-VPRE scheme has selective weak master key security if we add an Init stage before Setup where the adversary commits to I^*_{key}.

Verification soundness. An AB-VPRE scheme meets verification soundness if for any PPT adversary $\mathcal{A}$, given access to the oracle $\mathcal{O} \in \{\mathcal{O}_{sk}, \mathcal{O}_{rk}, \mathcal{O}_{re}, \mathcal{O}_{de}, \mathcal{O}_{rvd}\}$, the advantage in the following security game is negligible:

$$\begin{aligned}\Pr[&(PP, MSK) \leftarrow \text{Setup}(1^\lambda, U);\\ &(M^*, I^*_{enc}, st) \leftarrow \mathcal{A}^{\mathcal{O}}(PP);\\ &CT_{I^*} \leftarrow \text{Enc}(PP, M^*, I^*_{enc});\\ &(I'_{key}, I'^*_{enc}, CT_{I'^*}) \leftarrow \mathcal{A}^{\mathcal{O}}(CT_{I^*}, st);\\ &SK_{I'} \leftarrow \text{KGen}(PP, MSK, I'_{key});\\ &\text{REVer.Test}(CT_{I^*}, CT_{I'^*}) = \mathsf{b} :\\ &\mathsf{b} \wedge \text{REVer.Dec}(PP, SK_{I'}, CT_{I'^*}, \mathsf{b})\\ &\qquad \notin \{M^*, \perp\}] \leq \text{negl}(\lambda),\end{aligned}$$

where st is the state information and $f(I'_{key}, I'^*_{enc}) = 1$.

4. GENERIC CONSTRUCTION

In this section, we present our construction of an AB-VPRE scheme based on a pairing-based AB-KEM scheme described in a similar form of bilinear predicate encoding scheme [21–23] and analyze its security. Here, we just consider the AB-KEM scheme built on the bilinear group system of prime order for practical applications.

4.1. Bilinear encoding AB-KEM

Let $\Pi_{\text{KM}} = (\text{KM.Setup}, \text{KM.KGen}, \text{KM.Enc}, \text{KM.Dec})$ denote an AB-KEM scheme of which the algorithms are described as follows:

- $\text{KM.Setup}(1^\lambda, U)$: The setup algorithm chooses a bilinear group system $(\mathbb{G}, \mathbb{G}_T, e)$ of prime order p ($p \in \Theta(2^\lambda)$) and then selects a generator $g \in_R \mathbb{G}$,

a vector $\vec{a} = (x, a_1, a_2, \cdots) \in_R (\mathbb{Z}_p^*)^{\ell_{\vec{a}}}$ and a value $\alpha \in_R \mathbb{Z}_p^*$. It outputs the public parameters $PK = (e, g, g^{\vec{a}}, e(g,g)^{\alpha})$ and the master secret key $MSK = (\alpha, \vec{a})$.

- KM.KGen(PK, MSK, I_{key}): The key generation algorithm chooses a vector $\vec{r} \in_R (\mathbb{Z}_p^*)^{\ell_{\vec{r}}}$ and then outputs the private key $SK = (SK_0, SK_1)$, where $SK_0 = g^{\alpha} \cdot g^{x\mathrm{KE}_0(\vec{r},\vec{a})}$, $SK_1 = g^{\mathrm{KE}(I_{key},\vec{r},\vec{a})}$.
- KM.Enc(PK, I_{enc}, M): The encryption algorithm selects $s \in_R \mathbb{Z}_p^*$, and a vector $\vec{u} \in_R (\mathbb{Z}_p^*)^{\ell_{\vec{u}}}$, and sets $C_0 = g^s$, $C_1 = g^{s\mathrm{CE}(I_{enc},\vec{u},\vec{a})}$, $C_2 = h^s$, then outputs the ciphertext $C = (C_0, C_1, C_2)$. The encapsulated key is $\mathsf{DK}_0 = e(g,g)^{\alpha s}$.
- KM.Dec(SK, C): If $f(I_{key}, I_{enc}) \neq 1$, return an error symbol $\perp$; otherwise, compute $\frac{e(SK_0,C_0)}{F(SK_1,C_1)}$ to obtain $\mathsf{DK}_0 = e(g,g)^{\alpha s}$.

Here, KE_0, KE, CE are polynomial-time computable functions, where the outputs are vectors over $\mathbb{Z}_p^*$ and $F : \mathbb{G}^{\ell_{SK_1}} \times \mathbb{G}^{\ell_{C_1}} \to \mathbb{G}_T$ is a polynomial-time computable bilinear function satisfying the following property:

Decryptability. For all I_{key}, I_{enc}, such that $f(I_{key}, I_{enc}) = 1$, there exists an efficiently computable bilinear function F such that

$$F\left(g^{\mathrm{KE}(I_{key},\vec{r},\vec{a})}, g^{s\mathrm{CE}(I_{enc},\vec{u},\vec{a})}\right) = e\left(g^{x\mathrm{KE}_0(\vec{r},\vec{a})}, g^s\right).$$

The property of decryptability is similar to the property of α–reconstruction proposed in [22,23], and the correctness of the aforementioned scheme can be verified easily. We require that the functions KE_0, KE, and CE have linear properties described as follows:

Linearity. (i) For all $\vec{a} \in (\mathbb{Z}_p^*)^{\ell_{\vec{a}}}$, $\mathrm{KE}_0(\vec{r},\vec{a})$ is linear in $\vec{r}$. (ii) For all I_{key} and $\vec{a} \in (\mathbb{Z}_p^*)^{\ell_{\vec{a}}}$, $\mathrm{KE}(I_{key},\vec{r},\vec{a})$ is linear in $\vec{r}$. (iii) For all I_{key} and $\vec{a} \in (\mathbb{Z}_p^*)^{\ell_{\vec{a}}}$, $\mathrm{CE}(I_{enc},\vec{u},\vec{a})$ is affine in $\vec{a}$ and also affine in $\vec{u}$ if $\vec{u}$ is not null.

Remark 1. The linearity of KE_0, KE, and CE imply that we can compute $g^{x\mathrm{KE}_0(\vec{r},\vec{a})}$, $g^{\mathrm{KE}(I_{key},\vec{r},\vec{a})}$, and $g^{s\mathrm{CE}(I_{enc},\vec{u},\vec{a})}$ given I_{key}, I_{enc} along with $g, g^{\vec{a}}$ (but not $\vec{a}$).

4.2. Construction of AB-VPRE

- Setup($1^{\lambda}, U$): Choose an AB-KEM described as in Section 4.1, and a symmetric key encryption scheme $\Pi_{\mathrm{SE}} = (\mathrm{SE.Gen}, \mathrm{SE.Enc}, \mathrm{SE.Dec})$, where ℓ_{SE} denotes the length of the private key, and then select a key derivation function KDF with the output length ℓ and a commitment scheme $(\mathsf{Commit}, \mathsf{Decom})$. Call KM.Setup($1^{\lambda}$) $\to$ (PK, MSK), then output the public parameters $PP = (PK, \mathsf{KDF}, \ell, \Pi_{\mathrm{SE}}, \ell_{\mathrm{SE}}, (\mathsf{Commit}, \mathsf{Decom}))$ and the master secret key MSK.
- KGen(PP, MSK, I_{key}): Call KM.KGen(PK, MSK, I_{key}) $\to SK$ and output the private key $SK_I = (I_{key}, SK)$.
- Enc(PP, I_{enc}, M): Call KM.Enc(PK, I_{enc}) $\to (C_I, \mathsf{DK}_0)$, where $C_I = (C_0, C_1, C_2)$. Compute $\mathsf{KDF}(\mathsf{DK}_0, \ell) = \mathsf{DK} \parallel d$, where $\ell = \ell_{\mathrm{SE}} + |d|$, then run SE.Enc($\mathsf{DK}, M$) $\to \overline{C}$. Calculate $\mathsf{Commit}_d(\mathsf{DK}) = \widehat{C}$ and output the ciphertext $CT_I = (I_{enc}, C_I, \overline{C}, \widehat{C})$.
- Dec(PP, SK_I, CT_I): If $f(I_{key}, I_{enc}) \neq 1$, return $\perp$. If $f(I_{key}, I_{enc}) = 1$, call KM.Dec(SK, C_I) $\to \mathsf{DK}_0$ and compute $\mathsf{KDF}(\mathsf{DK}_0, \ell) = \mathsf{DK} \parallel d$, where $\ell = \ell_{\mathrm{SE}} + |d|$. If $\mathsf{Commit}_d(\mathsf{DK}) = \widehat{C}$, run SE.Dec($\mathsf{DK}, \overline{C}$) $\to M$ and output M; otherwise, output $\perp$.
- RKGen(PP, SK_I, I'_{enc}): Parse $SK_I = (I_{key}, SK)$ where $SK = (SK_0, SK_1)$, and $SK_0 = g^{\alpha} \cdot g^{x\mathrm{KE}_0(\vec{r},\vec{a})}$, $SK_1 = g^{\mathrm{KE}(I_{key},\vec{r},\vec{a})}$. Choose $\vec{\xi}, \vec{\delta} \in_R (\mathbb{Z}_p^*)^{\ell_{\vec{r}}}$ and set $D = g^{\mathrm{KE}_0(\vec{\delta},\vec{a})}$, $rk = (rk_0, rk_1)$, where $rk_0 = SK_0 \cdot g^{x\mathrm{KE}_0(\vec{\xi},\vec{a})}$, $rk_1 = SK_1 \cdot \widetilde{rk}_1 = SK_1 \cdot g^{\widetilde{\mathrm{KE}(I_{key},\vec{\xi}+\vec{\delta},\vec{a})}}$. Call KM.Enc($PK, I'_{enc}$) $\to$ $(\widetilde{C}_{I'}, \widetilde{\mathsf{DK}})$, then run SE.Enc($\mathsf{KDF}(\widetilde{\mathsf{DK}}, \ell_{\mathrm{SE}}), D$) $\to \widetilde{C}$, and set $\mathbb{C}_{I'} = (\widetilde{C}_{I'}, \widetilde{C})$. Output the re-encryption key $RK_{I \to I'} = (I_{key}, I'_{enc}, rk, \mathbb{C}_{I'})$.
- REnc($RK_{I \to I'}, CT_I$): If $f(I_{key}, I_{enc}) = 1$, Call KM.Dec(rk, C_I) $\to C'_1$ and output the re-encrypted ciphertext $CT_{I'} = (I'_{enc}, C_0, C'_1, C_2, \overline{C}, \widehat{C}, \mathbb{C}_{I'})$; otherwise, output an error symbol $\perp$.
- REVer.Test($CT_I, CT_{I'}$): Parse $CT_I = (I_{enc}, C_0, C_1, C_2, \overline{C}, \widehat{C})$, $CT_{I'} = (I'_{enc}, C'_0, C'_1, C'_2, \overline{C}', \widehat{C}', \mathbb{C}_{I'})$. If $C_0 = C'_0$ and $C_2 = C'_2$ and $\overline{C} = \overline{C}'$ and $\widehat{C} = \widehat{C}'$, output $\mathsf{b} = 1$; otherwise, return $\mathsf{b} = 0$.
- REVer.Dec($PP, SK_{I'}, CT_{I'}, \mathsf{b}$): If $f(I'_{enc}, I'_{key}) \neq 1$ or $\mathsf{b} = 0$, return $\perp$; otherwise, call KM.Dec($SK_{I'}, \widetilde{C}_{I'}$) $\to \widetilde{\mathsf{DK}}$, and run SE.Dec($\mathsf{KDF}(\widetilde{\mathsf{DK}}, \ell_{\mathrm{SE}}), \widetilde{C}$) $\to D$, then compute $\mathsf{DK}_0 = C'_1 \cdot e(D, C'_2)$ and $\mathsf{KDF}(\mathsf{DK}_0, \ell) = \mathsf{DK} \parallel d$. If $\mathsf{Commit}_d(\mathsf{DK}) = \widehat{C}'$, run SE.Dec($\mathsf{DK}, \overline{C}'$) $\to M$, and output M; otherwise, return a special symbol $\perp$ indicating that the verification fails.

We note that the correctness for the original ciphertext holds naturally, and the correctness for the re-encrypted ciphertext can be verified as follows. If $f(I'_{enc}, I'_{key}) = 1$,

$$\begin{aligned} C'_1 \cdot e(D, C'_2) &= \frac{e(rk_0, C_0)}{F(rk_1, C_1)} \cdot e(D, C_2) \\ &= \frac{e\left(SK_0 \cdot g^{x\mathrm{KE}_0(\vec{\xi},\vec{a})}, C_0\right)}{F\left(SK_1 \cdot g^{\mathrm{KE}\left(I_{key},\vec{\xi}+\vec{\delta},\vec{a}\right)}, C_1\right)} \cdot e(D, C_2) \\ &= e(g,g)^{\alpha s} \cdot \frac{e\left(g^{x\mathrm{KE}_0\left(\vec{r}+\vec{\xi}+\vec{\delta},\vec{a}\right)}, C_0\right)}{F\left(g^{\mathrm{KE}\left(I_{key},\vec{r}+\vec{\xi}+\vec{\delta},\vec{a}\right)}, C_1\right)} \\ &= e(g,g)^{\alpha s} = \mathsf{DK}_0. \end{aligned}$$

According to the linearity of the functions KE, KE_0, and the decryptability property, the third and the fourth

equalities hold respectively. Then $\mathsf{KDF}(\mathsf{DK}_0, \ell) = \mathsf{DK} \parallel d$. If REVer.Test$(CT_I, CT_{I'}) = 1$ and $\mathsf{Commit}_d(\mathsf{DK}) = \widehat{C}$, REVer.Dec$(PP, SK_{I'}, CT_{I'}, \mathsf{b})$ outputs M.

4.3. Data privacy and re-encryption verifiability

Now we discuss data privacy for original and re-encrypted ciphertexts, and re-encryption verifiability for the constructed AB-VPRE scheme. Formally, we have the following:

Theorem 1. *Suppose the AB-KEM Π_{KM} is CPA secure, the symmetric key encryption scheme Π_{SE} is semantically secure, the key derivation function* KDF *is secure, and the commitment scheme* (Commit, Decom) *is computationally hiding and binding, then the constructed AB-VPRE scheme is CPA secure at original ciphertexts and meets verification soundness.*

Proof. We consider four games (**Game**$_0$, **Game**$_1$, **Game**$_2$, and **Game**$_3$) between a challenger and a probabilistic polynomial-time (PPT) adversary $\mathcal{A}$ as follows.

Game$_0$: The real CPA game at original ciphertexts where the challenge ciphertext $CT_{I^*} = (I^*_{enc}, C_{I^*}, \overline{C}^*, \widehat{C}^*)$ is generated from $(C_{I^*}, \mathsf{DK}_0) \leftarrow \mathrm{KM.Enc}(PK, I^*_{enc})$, $\mathsf{KDF}(\mathsf{DK}_0, \ell) = \mathsf{DK} \parallel d$, $\overline{C}^* \leftarrow \mathrm{SE.Enc}(\mathsf{DK}, M_b)$, $b \in_R \{0,1\}$, and $\widehat{C}^* = \mathsf{Commit}_d(\mathsf{DK})$.

Game$_1$: The game is identical to **Game**$_0$ except that $\overline{C}^* \leftarrow \mathrm{SE.Enc}(\mathsf{DR}, M_b)$, $b \in_R \{0,1\}$, and $\widehat{C}^* = \mathsf{Commit}_d(\mathsf{DR})$, where $\mathsf{DR} \leftarrow \mathrm{SE.Gen}(1^\lambda)$ and d is a random coin.

Game$_2$: Identical to **Game**$_1$ except that $\widehat{C} = \mathsf{Commit}_d(\mathsf{R})$, where R is a random string of length ℓ_{SE} and d is a random coin.

Game$_3$: Identical to **Game**$_2$ except that $\overline{C}^* \leftarrow \mathrm{SE.Enc}(\mathsf{DR}, \tilde{M})$, where $\mathsf{DR} \leftarrow \mathrm{SE.Gen}(1^\lambda)$ and $\tilde{M}$ is a random message.

Hereafter, **Game**$_i \overset{c}{\approx}$ **Game**$_j$ $(i, j = 0, 1, 2, 3, i \neq j)$ denotes that **Game**$_i$ and **Game**$_j$ are computationally indistinguishable from each other. We will prove the indistinguishability in the following lemmas.

Lemma 1. *Assume the AB-KEM Π_{KM} is CPA secure, the key derivation function* KDF *is secure, and the commitment scheme* (Commit, Decom) *is computationally hiding, then* **Game**$_0 \overset{c}{\approx}$ **Game**$_2$.

Proof. Consider the following game:

Game$'$: Same as **Game**$_0$ except that $\overline{C}^* \leftarrow \mathrm{SE.Enc}(\mathsf{DR}', M_b)$, $b \in_R \{0,1\}$, and $\widehat{C}^* = \mathsf{Commit}_d(\mathsf{DR}')$, where $\mathsf{DR}' \parallel d = \mathsf{KDF}(\mathsf{DR}_0, \ell)$ and DR_0 is a random session key.

We first show that CPA security of the AB-KEM implies **Game**$_0 \overset{c}{\approx}$ **Game**$'$. Suppose there exists a PPT adversary $\mathcal{A}$ that can distinguish **Game**$_0$ and **Game**$'$ with non-negligible probability. We construct an algorithm $\mathcal{B}$ to break CPA security of the AB-KEM.

Actually, $\mathcal{B}$ receives PK from the challenger and then chooses KDF with the output length ℓ, Π_{SE} with the private-key length ℓ_{SE}, and (Commit, Decom), then sends $PP = (PK, \mathsf{KDF}, \ell, \Pi_{\mathrm{SE}}, \ell_{\mathrm{SE}}, (\mathsf{Commit}, \mathsf{Decom}))$ to $\mathcal{A}$. $\mathcal{A}$ adaptively issues private key and re-encryption key queries. For any private key query for I_{key}, $\mathcal{B}$ forwards it to its own oracle and returns the answer to $\mathcal{A}$. For any re-encryption key query for (I_{key}, I'_{enc}), $\mathcal{B}$ obtains the private key SK_I from its own oracle and sends $RK_{I \to I'}$ to A by running $\mathrm{RKGen}(PP, SK_I, I'_{enc}) \to RK_{I \to I'}$. After receiving two equal-length messages M_0, M_1 and I^*_{enc} from $\mathcal{A}$ with the restriction that $f(I_{key}, I^*_{enc}) \neq 1$ for any queried I_{key}, $\mathcal{B}$ sends I^*_{enc} to the challenger. The challenger runs $\mathrm{Enc}(PP, I^*_{enc}) \to (C_{I^*}, \mathsf{DK}_0)$, and sets $\mathsf{K}_0 = \mathsf{DK}_0$, $\mathsf{K}_1 = \mathsf{DR}_0$, where DR_0 is a random session key, then picks $\beta \in_R \{0,1\}$ and returns $(C_{I^*}, \mathsf{K}_\beta)$ to $\mathcal{B}$. $\mathcal{B}$ computes $\mathsf{KDF}(\mathsf{K}_\beta, \ell) = \mathsf{K} \parallel d$ and $\mathsf{Commit}_d(\mathsf{K}) = \widehat{C}^*$and then selects $b \in_R \{0,1\}$, runs $\mathrm{SE.Enc}(\mathsf{K}, M_b) \to \overline{C}^*$, and sends $CT_{I^*} = (I^*_{enc}, C_{I^*}, \overline{C}^*, \widehat{C}^*)$ to $\mathcal{A}$ as the challenge ciphertext. After the second query phase with the same restriction as before, $\mathcal{A}$ outputs its guess $b' \in \{0,1\}$. If $b' = b$, $\mathcal{B}$ outputs 0; otherwise, $\mathcal{B}$ outputs 1.

We can see that $\mathcal{B}$ has properly simulated **Game**$_0$ and **Game**$'$ for the case of $\beta = 0$ and $\beta = 1$, respectively. If the adversary $\mathcal{A}$ can distinguish **Game**$_0$ and **Game**$'$ with non-negligible probability, $\mathcal{B}$ can attack CPA security of the AB-KEM with non-negligible advantage. Thus **Game**$_0 \overset{c}{\approx}$ **Game**$'$.

The security of the KDF implies that $\mathsf{KDF}(\mathsf{DR}_0, \ell)$ is indistinguishable from a random string. That is, DR' and d are both indistinguishable from a random string, which implies that **Game**$' \overset{c}{\approx}$ **Game**$_1$. Then **Game**$_0 \overset{c}{\approx}$ **Game**$_1$. Because the commitment scheme is computationally hiding, **Game**$_1 \overset{c}{\approx}$ **Game**$_2$. Hence, **Game**$_0 \overset{c}{\approx}$ **Game**$_2$. □

Lemma 2. *Assume Π_{SE} is semantically secure, then* **Game**$_2 \overset{c}{\approx}$ **Game**$_3$.

Proof. Suppose there exists a PPT adversary $\mathcal{A}$ that can distinguish **Game**$_2$ and **Game**$_3$ with non-negligible probability. We can construct an algorithm $\mathcal{B}$ to attack the semantical security of Π_{SE} with non-negligible probability.

Actually, $\mathcal{B}$ runs $\mathrm{Setup}(1^\lambda, U) \to (PP, MSK)$ and sends PP to $\mathcal{A}$ and then simulates the oracles $\{\mathcal{O}_{sk}, \mathcal{O}_{rk}\}$ for $\mathcal{A}$. When $\mathcal{A}$ submits M_0, M_1, and I^*_{enc} to $\mathcal{B}$, $\mathcal{B}$ computes $C_{I^*}, \widehat{C}^*$ as described in **Game**$_2$ and then selects a random message $\tilde{M}$, picks $b \in_R \{0,1\}$, and sends $\overline{M}_0 = M_b, \overline{M}_1 = \tilde{M}$ to the challenger. The challenger picks $\beta \in_R \{0,1\}$ and computes $\overline{C}^* \leftarrow \mathrm{SE.Enc}(\mathsf{DR}, \overline{M}_\beta)$, where $\mathsf{DR} \leftarrow \mathrm{SE.Gen}(1^\lambda)$, and returns $\overline{C}^*$ to $\mathcal{B}$. Then $\mathcal{B}$ sends $CT_{I^*} = (I^*_{enc}, C_{I^*}, \overline{C}^*, \widehat{C}^*)$ to $\mathcal{A}$. After given access to the oracles again, $\mathcal{A}$ outputs its guess $b' \in \{0,1\}$. If $b' = b$, $\mathcal{B}$ returns 0 and outputs 1 otherwise.

We can see that $\mathcal{B}$ has properly simulated **Game**$_2$ and **Game**$_3$, respectively. If $\mathcal{A}$ can distinguish **Game**$_2$ and **Game**$_3$ with non-negligible probability, $\mathcal{B}$ can attack the

semantical security of Π_{SE} with non-negligible probability. Thus $\mathbf{Game}_2 \overset{c}{\approx} \mathbf{Game}_3$. □

By transitivity of computational indistinguishability, we obtain $\mathbf{Game}_0 \overset{c}{\approx} \mathbf{Game}_3$. Because $\mathbf{Game}_3$ contains no information about the messages submitted by the adversary, the advantage of the adversary in $\mathbf{Game}_3$ is negligible. Thus, the advantage of the adversary in the real game is negligible. It follows that the constructed AB-VPRE scheme is CPA secure.

Lemma 3. *Suppose that the commitment scheme* (Commit, Decom) *is computationally binding, then the constructed AB-VPRE scheme meets verification soundness.*

Proof. Suppose there exists an adversary $\mathcal{A}$ that can attack the verification soundness with non-negligible probability. We can build an algorithm $\mathcal{B}$ to attack the computational binding of the commitment scheme with non-negligible probability. Let $\mathcal{B}$ be the sender in the commitment scheme and executes $\mathcal{A}$ as follows:

$\mathcal{B}$ calls $\mathrm{Setup}(1^\lambda, U) \rightarrow (PP, MSK)$ and sends PP to $\mathcal{A}$, then simulates the oracles $\mathcal{O}_{sk}, \mathcal{O}_{rk}, \mathcal{O}_{re}, \mathcal{O}_{de}, \mathcal{O}_{rvd}$ for $\mathcal{A}$. After receiving I^*_{enc}, M^* from $\mathcal{A}$, $\mathcal{B}$ runs $\mathrm{Enc}(PP, I^*_{enc}, M^*) \rightarrow CT_{I^*}$, where $CT_{I^*} = (I^*_{enc}, C_0, C_1, C_2, \overline{C}^*, \widehat{C}^*)$ and $\widehat{C}^* = \mathsf{Commit}_d(\mathsf{DK})$, then sends CT_{I^*} to $\mathcal{A}$, and sends $\widehat{C}^*$ to the receiver. After given access to the oracles again, $\mathcal{A}$ outputs a re-encrypted ciphertext $CT_{I'^*} = (I'^*_{enc}, C'_0, C'_1, C'_2, \overline{C}'^*, \widehat{C}'^*, \mathbb{C}_{I'^*})$, and I'_{key} satisfying $f(I'_{key}, I'^*_{enc}) = 1$. $\mathcal{B}$ calls $\mathrm{KGen}(PP, MSK, I'_{key}) \rightarrow SK_{I'}$, then runs $\mathrm{REVer.Test}(CT_{I^*}, CT_{I'^*}) \rightarrow \mathsf{b}$ and $\mathrm{REVer.Dec}(PP, SK_{I'}, CT_{I'^*}, \mathsf{b}) \rightarrow M'$. $\mathcal{A}$ wins the game if $\mathsf{b} = 1$ and $M' \notin \{M^*, \perp\}$, which implies that there exists a tuple (DK', d') satisfying $\mathsf{DK}' \neq \mathsf{DK}$ such that $\mathrm{SE.Dec}(\mathsf{DK}', \overline{C}^*) \rightarrow M'$, and $\mathsf{Commit}_{d'}(\mathsf{DK}') = \mathsf{Commit}_d(\mathsf{DK})$. That contradicts to the computational binding of the commitment scheme.

Suppose that $\mathcal{A}$ wins the game with non-negligible probability, then $\mathcal{B}$ breaks the computational binding of the commitment scheme with non-negligible probability. □

Combining all the preceding discussions, we complete the proof for Theorem 1. □

The security property of re-encrypted ciphertexts is provided in Theorem 2. Because its proof is similar to that for Theorem 1, we omit the proof here for the limit of space.

Theorem 2. *Suppose the AB-KEM Π_{KM} is CPA secure, the symmetric key encryption scheme Π_{SE} is semantically secure, the key derivation function* KDF *is secure, and the commitment scheme* (Commit, Decom) *is computationally hiding, then the constructed AB-VPRE scheme is CPA secure at re-encrypted ciphertexts.*

Remark 2. We remark that the constructed AB-VPRE scheme inherits the security of the underlying AB-KEM scheme. If the AB-KEM scheme is selectively CPA secure, the resulting AB-VPRE scheme is selectively CPA secure as well.

5. INSTANTIATIONS

In this section, we instantiate our generic construction with three AB-KEMs. For comparisons with the existing AB-PRE schemes [2,6,20] that are based on similar AB-KEMs as ours, we just provide three selectively secure AB-VPRE schemes. That is, one KP-AB-VPRE scheme and two CP-AB-VPRE schemes with monotonic and non-monotonic access structures, respectively. We note that one can achieve adaptively secure AB-KEMs in prime-order groups by properly instantiating the bilinear encoding AB-KEM [23] and thus obtain adaptively secure AB-VPRE schemes.

5.1. AB-VPRE-1

We first describe the CP-AB-KEM scheme [2] as in the subsequent discussion ($U = \{1, 2, \cdots, n\}$).

- KM.Setup$(1^\lambda, U)$: Choose a bilinear group system $(\mathbb{G}, \mathbb{G}_T, e)$ of prime order p ($p \in \Theta(2^\lambda)$). Select a generator $g \in_R \mathbb{G}$, $x, y, \{t_i\}_{1 \le i \le 3n} \in_R \mathbb{Z}_p^*$. Set $\vec{a} = \left(x, \{t_i, \frac{x}{t_i}\}_{1 \le i \le 3n}\right)$, $\alpha = xy$, $h = g^x$. Then $e(g,g)^\alpha = e(g,h)^y$, $g^{\vec{a}} = \left(g^x, \{g^{t_i}, h^{\frac{1}{t_i}}\}_{1 \le i \le 3n}\right)$. The public parameters $PK = (e, g, g^{\vec{a}}, e(g,g)^\alpha)$ and the master secret key $MSK = (\alpha, \vec{a})$.
- KM.KGen(PK, MSK, S): Choose $\vec{r} = (r_1, \cdots, r_n) \in_R (\mathbb{Z}_p^*)^n$ and implicitly set $\mathrm{KE}_0(\vec{r}, \vec{a}) = \Sigma_{i=1}^n r_i = r_0$, and $\mathrm{KE}(S, \vec{r}, \vec{a}) = (\{\frac{x}{t_i} r_i\}_{i \in S}; \{\frac{x}{t_{n+i}} r_i\}_{i \in U \setminus S}; \{\frac{x}{t_{2n+i}} r_i\}_{i \in U})$. Then $SK = (SK_0, SK_1)$ where $SK_0 = g^\alpha \cdot g^{x\mathrm{KE}_0(\vec{r},\vec{a})} = h^{y+r_0}$, $SK_1 = g^{\mathrm{KE}(S,\vec{r},\vec{a})} = (\{h^{\frac{r_i}{t_i}}\}_{i \in S}; \{h^{\frac{r_i}{t_{n+i}}}\}_{i \in U \setminus S}; \{h^{\frac{r_i}{t_{2n+i}}}\}_{i \in U})$.
- KM.Enc$(PK, \mathbb{A})$: $\mathbb{A}$ is represented as $\bigwedge \underline{i}$, $i \in \mathcal{I} \subseteq U$, where $\underline{i}$ denotes an attribute (positive i^+ or negative i^-). Select $s \in_R \mathbb{Z}_p^*$, and implicitly set $\mathrm{CE}(\mathbb{A}, \vec{u}, \vec{a}) = (\{t_i\}_{i \in I \wedge \underline{i} = i^+}, \{t_{n+i}\}_{i \in I \wedge \underline{i} = i^-}, \{t_{2n+i}\}_{i \in U \setminus I})$, where $\vec{u}$ is null. Then the ciphertext $C = (C_0, C_1, C_2)$, where $C_0 = g^s$, $C_2 = h^s$, $C_1 = g^{s\mathrm{CE}(\mathbb{A},\vec{u},\vec{a})} = (\{g^{t_i s}\}_{i \in I \wedge \underline{i} = i^+}, \{g^{t_{n+i} s}\}_{i \in I \wedge \underline{i} = i^-}, \{g^{t_{2n+i} s}\}_{i \in U \setminus I})$. The encapsulated key $\mathsf{DK}_0 = e(g,g)^{\alpha s}$.
- KM.Dec(SK, C): If $f(S, \mathbb{A}) \neq 1$, output $\perp$; otherwise, define bilinear function F as follows:

$$F(SK_1, C_1) = F\left(g^{\mathrm{KE}(S,\vec{r},\vec{a})}, g^{s\mathrm{CE}(\mathbb{A},\vec{u},\vec{a})}\right) = \prod_{i \in I \wedge \underline{i} = i^+ \wedge i \in S} e\left(g^{t_i s}, h^{\frac{r_i}{t_i}}\right) \prod_{i \in U \setminus I} e\left(g^{t_{2n+i} s}, h^{\frac{r_i}{t_{2n+i}}}\right) \prod_{i \in I \wedge \underline{i} = i^- \wedge i \in U \setminus S} \left(g^{t_{n+i} s}, h^{\frac{r_i}{t_{n+i}}}\right) = e(g,h)^{sr_0}$$

and compute $\frac{e(SK_0, C_0)}{F(SK_1, C_1)} = e(g,g)^{\alpha s}$.

Based on the aforementioned AB-KEM, the one-time pad, and the Pedersen commitment scheme [24], we construct the CP-AB-VPRE scheme named AB-VPRE-1. The algorithms of AB-VPRE-1 are described according to the construction presented in Section 4. For the sake of understanding, we describe the algorithms Setup and RKGen here.

- Setup($1^\lambda, U$): Call KM.Setup(1^λ)$\rightarrow$ (PK, MSK). Select $w_1, w_2 \in_R \mathbb{G}$ and a collision-resistant hash function $H : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ used for commitment and choose KDF with the output length ℓ. Return $PP = (PK, w_1, w_2, H, \mathsf{KDF}, \ell)$ and MSK.
- RKGen($PP, SK_S, \mathbb{A}'$): Choose $\vec{\xi} = (\xi_1, \cdots, \xi_n)$, $\vec{\delta} = (\delta_1, \cdots, \delta_n)$ from $(\mathbb{Z}_p^*)^n$ randomly and set $D = g^{\Sigma_{i=1}^n \delta_i}$, $rk = (rk_0, rk_1)$, where $rk_0 = h^{y+r_0+\Sigma_{i=1}^n \xi_i}$, $rk_1 = (\{h^{\frac{r_i'}{t_i}}\}_{i\in S}; \{h^{\frac{r_i'}{t_{n+i}}}\}_{i\in U\setminus S}; \{h^{\frac{r_i'}{t_{2n+i}}}\}_{i\in U})$ with $r_i' = r_i + \xi_i + \delta_i$, $i \in U$. Call KM.Enc($PK, \mathbb{A}'$) $\rightarrow$ $(\tilde{C}_{\mathbb{A}'}, \widetilde{\mathsf{DK}})$ and run $\mathsf{KDF}(\widetilde{\mathsf{DK}}, |D|) \oplus D \rightarrow \tilde{C}$, then set $\mathbb{C}_{\mathbb{A}'} = (\tilde{C}_{\mathbb{A}'}, \tilde{C})$, and finally output the re-encryption key $RK_{S\rightarrow\mathbb{A}'} = (S, \mathbb{A}', rk, \mathbb{C}_{\mathbb{A}'})$.

5.2. AB-VPRE-2

The CP-AB-KEM scheme [25] can be described as follows.

- KM.Setup($1^\lambda, U$): Choose a bilinear group system $(\mathbb{G}, \mathbb{G}_T, e)$ of prime order p ($p \in \Theta(2^\lambda)$). Select a generator $g \in_R \mathbb{G}$, $x, \alpha, \{t_i\}_{i\in U} \in_R \mathbb{Z}_p^*$ and set $\vec{a} = (x, \{t_i\}_{i\in U})$. Then $g^{\vec{a}} = (h = g^x, \{T_i = g^{t_i}\}_{i\in U})$. The public parameters $PK = (e, g, g^{\vec{a}}, e(g,g)^\alpha)$ and the master secret key $MSK = (\alpha, \vec{a})$.
- KM.KGen(PK, MSK, S): Choose $r \in_R \mathbb{Z}_p^*$ and implicitly set $\mathrm{KE}_0(r, \vec{a}) = r$, and $\mathrm{KE}(S, r, \vec{a}) = (r, \{rt_i\}_{i\in S})$. Then $SK = (SK_0, SK_1)$ where $SK_0 = g^\alpha g^{x\mathrm{KE}_0(r,\vec{a})} = g^\alpha g^{xr}$, $SK_1 = g^{\mathrm{KE}(S,r,\vec{a})} = (g^r, \{g^{rt_i}\}_{i\in S}) = (g^r, \{T_i^r\}_{i\in S})$.
- KM.Enc($PK, \mathbb{A}, M$): $\mathbb{A} = (\mathbf{A}_{m\times n}, \rho)$, where $\mathbf{A}_{m\times n}$ is an $m \times n$ matrix and ρ maps row A_i to an attribute. Select $s, \{s_i\}_{i\in[m]}, \{v_{i+1}\}_{i\in[n-1]} \in_R \mathbb{Z}_p^*$. Let $\vec{u} = (\{s_i\}_{i\in[m]}, \{v_{i+1}\}_{i\in[n-1]})$, $\vec{v} = (1, v_2, \cdots, v_n)$, and set $\mathrm{CE}(\mathbb{A}, \vec{u}, \vec{a}) = (\{xA_i \cdot \vec{v} - s_i t_{\rho(i)}, s_i\}_{i\in[m]})$. Then $C = (C_0, C_1, C_2)$, where $C_0 = g^s = C_2$, $C_1 = g^{s\mathrm{CE}(\mathbb{A},\vec{u},\vec{a})} = (\{g^{xsA_i\cdot\vec{v}} T_{\rho(i)}^{-ss_i}, g^{ss_i}\}_{i\in[m]})$. The encapsulated key $\mathsf{DK}_0 = e(g,g)^{\alpha s}$.
- KM.Dec(SK, C): If $f(S, \mathbb{A}) \neq 1$, output $\perp$; otherwise, $\exists\ \omega_i \in \mathbb{Z}_p^*$ s.t. $\sum_{\rho(i)\in S} \omega_i A_i = \vec{1}$. Define $F(SK_1, C_1) = F(g^{\mathrm{KE}(S,r,\vec{a})}, g^{s\mathrm{CE}(\mathbb{A},\vec{u},\vec{a})}) = \prod_{\rho(i)\in S}(e(g^r, g^{xsA_i\cdot\vec{v}} T_{\rho(i)}^{-ss_i}) e(T_{\rho(i)}^r, g^{ss_i}))^{\omega_i} = e(g,g)^{xsr}$ and compute $\frac{e(SK_0,C_0)}{F(SK_1,C_1)} = e(g,g)^{\alpha s}$.

Similar to the construction of AB-VPRE-1, we can obtain a CP-AB-VPRE scheme based on the aforementioned CP-AB-KEM, named AB-VPRE-2, and provide the algorithm RKGen in the following.

- RKGen($PP, SK_S, \mathbb{A}'$): Choose $\xi, \delta \in_R \mathbb{Z}_p^*$ and set $D = h^\delta$, $rk = (rk_0, rk_1)$, where $rk_0 = g^\alpha h^{r+\xi}$, $rk_1 = (g^{r+\xi+\delta}, \{T_i^{r+\xi+\delta}\}_{i\in S})$. The following descriptions are the same as AB-VPRE-1.

5.3. AB-VPRE-3

The KP-AB-KEM [3] has the same setup algorithm as the CP-AB-KEM [25], and the remaining algorithms are described as follows ($\mathbb{A} = (\mathbf{A}_{m\times n}, \rho)$).

- KM.KGen($PK, MSK, \mathbb{A}$): Select $\vec{r} = (\{r_i\}_{i\in[m]}, r_0, \{v_{i+1}\}_{i\in[n-1]}) \in_R (\mathbb{Z}_p^*)^{m+n}$ and set $\vec{v} = (r_0, v_2, \cdots, v_n)$. Implicitly set $\mathrm{KE}_0(\vec{r}, \vec{a}) = r_0$, $\mathrm{KE}(\mathbb{A}, \vec{r}, \vec{a}) = (\{xA_i \cdot \vec{v} - r_i t_{\rho(i)}, r_i\}_{i\in[m]})$, then $SK = (SK_0, SK_1)$, where $SK_0 = g^\alpha g^{x\mathrm{KE}_0(\vec{r},\vec{a})} = g^\alpha g^{xr_0}$, $SK_1 = g^{\mathrm{KE}(\mathbb{A},\vec{r},\vec{a})} = (\{g^{xA_i\cdot\vec{v}} T_{\rho(i)}^{-r_i}, g^{r_i}\}_{i\in[m]})$.
- KM.Enc(PK, S, M): Choose $s \in_R \mathbb{Z}_p^*$ and implicitly set $\mathrm{CE}(S, \vec{u}, \vec{a}) = (1, \{t_i\}_{i\in S})$, where $\vec{u}$ is null. Then the ciphertext $C = (C_0, \breve{C}_1, C_2)$, where $C_0 = g^s$, $\breve{C}_1 = \{T_i^s\}_{i\in S}$, $C_2 = h^s$. Let $C_1 = g^{s\mathrm{CE}(S,\vec{u},\vec{a})} = (C_0, \breve{C}_1)$. The encapsulated key is $\mathsf{DK}_0 = e(g,g)^{\alpha s}$.
- KM.Dec(SK, C): If $f(S, \mathbb{A}) \neq 1$, return $\perp$; otherwise, $\exists\ \omega_i \in \mathbb{Z}_p^*$ s.t. $\sum_{\rho(i)\in S} \omega_i A_i = \vec{1}$. Define $F(SK_1, C_1) = \prod_{\rho(i)\in S} e(g^{xA_i\cdot\vec{v}} T_{\rho(i)}^{-r_i}, g^s)^{\omega_i} \cdot e(g^{r_i}, T_{\rho(i)}^s)^{\omega_i} = e(g,g)^{xsr_0}$ and compute $\frac{e(SK_0,C_0)}{F(SK_1,C_1)} = e(g,g)^{\alpha s}$.

Similarly, we build the KP-AB-VPRE scheme named AB-VPRE-3 based on this AB-KEM and provide the algorithm RKGen subsequently.

- RKGen($PP, SK_{\mathbb{A}}, S'$): Choose $\vec{\xi}, \vec{\delta} \in_R (\mathbb{Z}_p^*)^{m+n}$, where $\vec{\xi} = (\xi_0, \cdots, \xi_{m+n-1})$, $\vec{\delta} = (\delta_0, \cdots, \delta_{m+n-1})$. Let $\vec{v}' = \vec{v} + \vec{\xi} + \vec{\delta}$. Set $D = g^{\delta_0}$, $rk = (rk_0, rk_1)$, where $rk_0 = g^\alpha g^{x(r_0+\xi_0)}$, $rk_1 = (\{g^{xA_i\cdot\vec{v}} T_{\rho(i)}^{-r_i'}, g^{r_i'}\}_{i\in[m]})$ with $r_i' = r_i + \xi_i + \delta_i$, $i \in [m]$. The following descriptions are the same as the generic construction.

Remark 3. We based that remark on the KP-AB-KEM [26] adapted from the preceding one for reducing decryption costs; we can build the KP-AB-VPRE scheme with fast decryption, named AB-VPRE-3a.

5.4. Security analysis

Because the Pedersen commitment scheme is perfectly hiding and computationally binding under the discrete-logarithm assumption, the instantiations of AB-VPRE inherit the security of the underlying AB-KEMs if the key derivation function is secure. By similar analysis, we can obtain the following security results.

Theorem 3. *Suppose the ADBDH assumption* [2] *holds and the key derivation function* KDF *is secure, then the constructed scheme AB-VPRE-1 is selectively CPA secure and meets verification soundness.*

Theorem 4. *Suppose the decisional q-parallel BDHE assumption* [25] *holds and the key derivation function* KDF *is secure, then the scheme AB-VPRE-2 is selectively CPA secure and meets verification soundness.*

Theorem 5. *Suppose the decisional q-BDHE assumption* [26] *holds and the key derivation function* KDF *is secure, then the scheme AB-VPRE-3 is selectively CPA secure and meets verification soundness.*

Note that the master key security of AB-VPRE-1 can be proved under the same assumption as described in [2] by similar analysis. Now we prove the weak master key security of AB-VPRE-2 and AB-VPRE-3 under CDH Assumption.

Theorem 6. *The schemes AB-VPRE-2 and AB-VPRE-3 have the selective weak master key security under the CDH assumption.*

Proof. We first prove that AB-VPRE-2 has the selective weak master key security under the CDH assumption. The simulator $\mathcal{B}$ is given $(p, \mathbb{G}, \mathbb{G}_T, e, g, g^a, g^b)$, where g is a generator of $\mathbb{G}$, $a, b \in_R \mathbb{Z}_p^*$ and intends to compute g^{ab}. $\mathcal{B}$ simulates the security game for the adversary $\mathcal{A}$ as follows:

$\mathcal{A}$ delivers $\hat{S}$ to $\mathcal{B}$. $\mathcal{B}$ chooses $\mathbb{A} = (\mathbf{A}_{m\times n}, \rho)$ satisfying $f(\hat{S}, \mathbb{A}) = 1$, where $A_i = (a_{i1}, \cdots, a_{in})$ is the i-th row of $\mathbf{A}_{m\times n}$. Let $\Lambda = \{i : \exists \omega_i, \text{s.t.} \sum_{\rho(i)\in\hat{S}} \omega_i A_i = \vec{1}\}$. $\mathcal{B}$ chooses $\omega_j \neq 0, j \in \Lambda$ and sets $g^{t_{\rho(j)}} = g^{at'_{\rho(j)}}$, then selects $\alpha, x \in_R \mathbb{Z}_p^*, t'_{\rho(j)}, t_i \in_R \mathbb{Z}_p^*, i \in U \setminus \{\rho(j)\}$, w_1, $w_2 \in_R \mathbb{G}$, a collision-resistant hash function $H : \{0,1\}^* \to \mathbb{Z}_p^*$ and KDF with the output length ℓ. It sends $PP = (g, g^x, e(g,g)^\alpha, \{T_i = g^{t_i}\}_{i\in U}, w_1, w_2, H, \mathsf{KDF}, \ell)$ to $\mathcal{A}$. $\mathcal{A}$ is given access to the oracles $\mathcal{O}_{sk}, \mathcal{O}_{rk}, \mathcal{O}_{de}, \mathcal{O}_{rvd}$ with the restriction that $\mathcal{A}$ cannot submit $\hat{S}$ to $\mathcal{O}_{sk}$. When $\mathcal{A}$ issues $(\hat{S}, \mathbb{A})$ to $\mathcal{O}_{rk}$, $\mathcal{B}$ selects $\delta' \in_R \mathbb{Z}_p^*$, and sets $D = g^{-b} g^{\delta'}$, $K' = g^\alpha g^{bx}, L' = g^{\delta'}, K'_{\rho(j)} = g^{\delta' at'_{\rho(j)}}$, $K'_i = g^{at_i}$, $i \in \hat{S}\backslash\{\rho(j)\}$, then encrypts D under $\mathbb{A}'$ and outputs $RK'_{\hat{S}\to\mathbb{A}'} = (\hat{S}, \mathbb{A}', K', L', \{K'_i\}_{i\in\hat{S}}, \mathbb{C}_{\mathbb{A}'})$. Finally, $\mathcal{A}$ outputs a private key $\widehat{SK}_{\hat{S}}$ and $RK_{\hat{S}\to\mathbb{A}''}$.

Suppose that $\mathcal{A}$ succeeds in generating the valid private key $\widehat{SK}$ and $RK_{\hat{S}\to\mathbb{A}''}$ with overwhelming probability. Let $\widehat{SK} = (\hat{S}, K, L, \{K_i\}_{i\in\hat{S}})$ and $RK_{\hat{S}\to\mathbb{A}''} = (\hat{S}, \mathbb{A}'', K'', L'', \{K''_i\}_{i\in\hat{S}})$. For a valid ciphertext $CT_{\mathbb{A}}$, it holds that $g^\alpha L^x \prod_{\rho(i)\in\hat{S}} (L^{-t_{\rho(i)}} \cdot K_{\rho(i)})^{s_i\omega_i} = K$ with $s_i \in_R \mathbb{Z}_p^*$, which implies that $L^{-t_{\rho(i)}} \cdot K_{\rho(i)} = 1$. Specially, $K_{\rho(j)} = L^{at'_{\rho(j)}}$. For $(\hat{S}, \mathbb{A}'')$ has been issued to $\mathcal{O}_{rk}$, $\mathcal{B}$ can find the re-encryption key $(\hat{S}, \mathbb{A}'', K'', L'', \{K''_i\}_{i\in\hat{S}})$ such that $(\hat{S}, K'/K'', L'/L'', \{K'_i/K''_i\}_{i\in\hat{S}})$ is a valid private key. Then $K'_{\rho(j)}/K''_{\rho(j)} = (L'/L'')^{at'_{\rho(j)}}$ and $K'_{\rho(j)} = L'^{at'_{\rho(j)}}$ implies that $K''_{\rho(j)} = L''^{at'_{\rho(j)}}$. Because $(L''/L)^{at'_{\rho(j)}} = g^{a\delta t'_{\rho(j)}} = (K'_{\rho(j)}/g^{abt'_{\rho(j)}})$, it holds that $g^{ab} = (K'_{\rho(j)} K_{\rho(j)} / K''_{\rho(j)})^{(t'_{\rho(j)})^{-1}}$. That is, $\mathcal{B}$ is able to output g^{ab} with overwhelming probability, which contradicts to the CDH assumption.

We provide a brief proof for the selective weak master key security of AB-VPRE-3 subsequently:

$\mathcal{B}$ is given $\hat{\mathbb{A}} = (\hat{\mathbf{A}}_{m\times n}, \hat{\rho})$ and chooses S where $f(S, \hat{\mathbb{A}}) = 1$. Then $\exists\omega_i$, s.t. $\sum_{\hat{\rho}(i)\in S} \omega_i A_i = \vec{1}$. $\mathcal{B}$ sends $PP = (g, g^a, e(g,g)^\alpha, \{g^{bt'_i}\}_{i\in U}, w_1, w_2, H, \mathsf{KDF}, \ell)$ to $\mathcal{A}$ with $\{t'_i\}_{i\in U} \in_R \mathbb{Z}_p^*$. When $\mathcal{A}$ issues $(\hat{\mathbb{A}}, S')$ to $\mathcal{O}_{rk}$, $\mathcal{B}$ sets $D = g^b$, and $K' = g^\alpha g^{ar_0}$, $\{L'_i = g^{aA_i\cdot\vec{v} - bt'_{\rho(i)} r'_i}, R'_i = (g^a)^{a_{i1}/t'_{\rho(i)}} g^{r'_i}\}_{i\in[m]}$, where $r_0, \{r'_i\}_{i\in[m]}, \{v_{i+1}\}_{i\in[n-1]} \in_R \mathbb{Z}_p^*$. Finally, $\mathcal{A}$ outputs $\widehat{SK} = (\hat{\mathbb{A}}, K, L_i, R_i, i \in [m])$ and $RK_{S\to\hat{\mathbb{A}}''} = (\hat{\mathbb{A}}'', K'', L''_i, R''_i, i \in [m])$ with overwhelming probability. By definition of the scheme, $g^\alpha \prod_{\hat{\rho}(i)\in S} (L_i R_i^{bt'_{\rho(i)}})^{\omega_i} = K$, $g^\alpha \prod_{\hat{\rho}(i)\in S} (L'_i R'^{bt'_{\rho(i)}}_i)^{\omega_i} = K' \cdot g^{ab}$ and $(\hat{\mathbb{A}}, K'/K'', L'_i/L''_i, R'_i/R''_i, i \in [m])$ is a valid private key, $g^\alpha \prod_{\hat{\rho}(i)\in S} ((L'_i/L''_i)(R'_i/R''_i)^{bt'_{\rho(i)}})^{\omega_i} = K'/K''$ implies that $\prod_{\hat{\rho}(i)\in S} (R''_i/R'_i)^{b\omega_i t'_{\rho(i)}} = (g^\alpha K'')/K' \prod_{\hat{\rho}(i)\in S} (L'_i/L''_i)^{\omega_i}$. Because $\prod_{\hat{\rho}(i)\in S} (R'_i/R_i)^{b\omega_i t'_{\rho(i)}} = \prod_{\hat{\rho}(i)\in S} (R'_i/R''_i)^{b\omega_i t'_{\rho(i)}}$, $\mathcal{B}$ can compute $g^{ab} = (K/K') \prod_{\hat{\rho}(i)\in S} (L'_i/L_i)^{\omega_i} (R'_i/R_i)^{b\omega_i t'_{\rho(i)}}$. That is, $\mathcal{B}$ is able to output g^{ab} with overwhelming probability, which contradicts to the CDH assumption. □

5.5. Comparisons

We compare our work with previous AB-PRE schemes in terms of computation costs, functionality, and security. Considering the resource-constrained users, we mainly discuss the computation costs of encryption, re-encryption key generation, and decryption for original and re-encrypted ciphertexts, respectively.

To the best of our knowledge, our schemes are the first to achieve re-encryption verifiability for AB-PRE, which guarantees the correctness of the re-encrypted ciphertexts and simultaneously detects the malicious behaviors of the proxy. As shown in Table II, our verification only requires two modular exponential operations for encryption and decryption, respectively. Our re-encryption leads to one more pairing operation during decryption, while others [6,20] result in more than doubled computation costs on both modular exponential and pairing operations for decryption. Our computation costs of AB-VPRE-2 and AB-VPRE-3 for the users are less than [6] and [20], respectively. The computation for the re-encryption of our schemes can be outsourced to the public cloud, and the

Table II. Comparisons.

Scheme	KP/CP	Access policy	Computation costs				Verify re-enc	Data privacy
			Enc	RKGen	Dec	REVer.Dec		
Ref. [2]	CP	AND	$(n+3)$E	$(3n+4)$E	$(n+1)$P	$(n+2)$P	×	CPA
Ref. [6]	CP	LSSS	$(3m+6)$E	$(3m+2n'+14)$E	$(3m'+1)$E+ $(3m'+10)$P	$(8m'+4)$E+ $(7m'+24)$P	×	CCA
Ref. [20]	KP	LSSS	$(n'+7)$E	$((\sigma+2)m+n'+7)$E	$(2m'+4)$E +4P	$(4m'+13)$E +8P	×	CCA
AB-VPRE-1	CP	AND	$(n+5)$E	$(3n+4)$E	2E + $(n+1)$P	2E + $(n+2)$P	✓	CPA
AB-VPRE-2	CP	LSSS	$(3m+5)$E	$(3m+n'+6)$E	$(m'+2)$E+ $(2m'+1)$P	$(m'+2)$E+ $(2m'+2)$P	✓	CPA
AB-VPRE-3a	KP	LSSS	$(n'+5)$E	$((\sigma+1)m+n'+5)$E	$(2m'+2)$E +2P	$(2m'+2)$E +3P	✓	CPA

‡ E and P denote a modular exponential and a paring operation, respectively. n, n', m, m', and σ indicate the number of the attribute universe U, the size of the attribute set S, the number of rows of the secret-sharing matrix **A**, the rows used during decryption for LSSS and the number of distinct attributes that appear in **A**, respectively.

returned ciphertexts will be verified. The schemes of [6,20] achieve CCA security but still cannot guarantee that the qualified users receive the correct data without any check for the correctness of the re-encrypted ciphertexts.

6. CONCLUSIONS

For AB-PRE where the public cloud is modeled as the proxy, the re-encryption may not be performed honestly. We propose a technique to verify the re-encryption such that the correctness of the re-encrypted ciphertexts can be guaranteed. We present a generic construction of AB-PRE with verifiable re-encryption and provide three instantiations in both KP and CP settings. Compared with the AB-PRE schemes proposed before, our schemes require less computation for the resource-constrained users and the verification can be achieved efficiently.

ACKNOWLEDGEMENTS

This work was supported by the Foundation of Science and Technology on Information Assurance Laboratory (no. KJ-14-002), Strategic Priority Research Program of the Chinese Academy of Sciences (no. XDA06010703), The One Hundred Talents Project of CAS, National Natural Science Foundation of China (no. 61272478, 61472416, 61379142).

REFERENCES

1. Sahai A, Waters B. Fuzzy identity-based encryption, *Proceedings of Advances in Cryptology-EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, Aarhus, Denmark, 2005; 457–473.
2. Liang X, Cao Z, Lin H, Shao J. Attribute based proxy re-encryption with delegating capabilities. *Proceedings of the 2009 ACM Symposium on Information, Computer and Communications Security, ASIACCS*, Sydney, Australia, 2009; 276–286.
3. Goyal V, Pandey O, Sahai A, Waters B. Attribute-based encryption for fine-grained access control of encrypted data. *Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS*, Alexandria, VA, USA, 2006; 89–98.
4. Liang K, Fang L, Susilo W, Wong D. A ciphertext-policy attribute-based proxy re-encryption with chosen-ciphertext security. *International Conference on Intelligent Networking and Collaborative Systems, 2013*, IEEE, Xi'an City, Shaanxi Province, China, 2013; 552–559.
5. Liang K, Au MH, Susilo W, Wong DS, Yang G, Yu Y. An adaptively CCA-secure ciphertext-policy attribute-based proxy re-encryption for cloud data sharing. *Proceedings of Information Security Practice and Experience*, Springer, Fuzhou, China, 2014; 448–461.
6. Liang K, Au MH, Liu JK, Susilo W, Wong DS, Yang G, Yu Y, Yang A. A secure and efficient ciphertext-policy attribute-based proxy re-encryption for cloud data sharing. *Future Generation Computer Systems* 2015; **52**: 95–108.
7. Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryption. *Proceedings of IEEE Symposium on Security and Privacy*, IEEE, Oakland, California, USA, 2007; 321–334.
8. Cheung L, Newport C. Provably secure ciphertext policy ABE. *Proceedings of the 2007 ACM Conference on Computer and Communications Security, (CCS)*, Alexandria, Virginia, USA, 2007; 456–465.
9. Ostrovsky R, Sahai A, Waters B. Attribute-based encryption with non-monotonic access structures, *Proceedings of the 2007 ACM Conference on Computer*

and Communications Security, (CCS), Alexandria, Virginia, USA, 2007; 195–203.
10. Green M, Hohenberger S, Waters B. Outsourcing the decryption of ABE ciphertexts. *Proceedings of 20th USENIX Security Symposium*, San Francisco, CA, USA, 2011.
11. Lai J, Deng RH, Guan C, Weng J. Attribute-based encryption with verifiable outsourced decryption. *IEEE Transactions on Information Forensics and Security* 2013; **8**(8): 1343–1354.
12. Blaze M, Bleumer G, Strauss M. Divertible protocols and atomic proxy cryptography. *Proceeding of Advances in Cryptology - EUROCRYPT '98, International Conference on the Theory and Application of Cryptographic Techniques*, Springer, Espoo, Finland, 1998; 127–144.
13. Ateniese G, Fu K, Green M, Hohenberger S. Improved proxy re-encryption schemes with applications to secure distributed storage. *ACM TISSEC* 2006; **9** (1): 1–30.
14. Hanaoka G, Kawai Y, Kunihiro N, Matsuda T, Weng J, Zhang R, Zhao Y. Generic construction of chosen ciphertext secure proxy re-encryption, *Proceedings of Topics in Cryptology - CT-RSA 2012 - The Cryptographers' Track at the RSA Conference*, Springer, San Francisco, CA, USA, 2012; 349–364.
15. Ohata S, Kawai Y, Matsuda T, Hanaoka G, Matsuura K. Re-encryption verifiability: how to detect malicious activities of a proxy in proxy re-encryption. *Proceedings of Topics in Cryptology - CT-RSA 2015, The Cryptographer's Track at the RSA Conference*, Springer, San Francisco, CA, USA, 2015; 410–428.
16. Luo S, Hu J, Chen Z. Ciphertext policy attribute-based proxy re-encryption. *Proceedings of Information and Communications Security: 12th International Conference, ICICS*, Springer, Barcelona, Spain, 2010; 401–415.
17. Yu S, Wang C, Ren K, Lou W. Achieving secure, scalable, and fine-grained data access control in cloud computing. *Proceedings of the 29th Conference on Computer Communications*, IEEE, San Diego, CA, USA, 2010; 1–9.
18. Yu S, Wang C, Ren K, Lou W. Attribute based data sharing with attribute revocation. *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, ASIACCS*, Beijing, China, 2010; 261–270.
19. Seo HJ, Kim H. Attribute-based proxy re-encryption with a constant number of pairing operations. *Journal of Information and Communication Convergence Engineering* 2012; **10**(1): 53–60.
20. Liang K, Susilo W. Searchable attribute-based mechanism with efficient data sharing for secure cloud storage. *IEEE Transactions on Information Forensics and Security* 2015; **10**(9): 1981–1992.
21. Attrapadung N. Dual system encryption via doubly selective security: framework, fully secure functional encryption for regular languages, and more. *Proceedings of Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, Copenhagen, Denmark, 2014; 557–577.
22. Wee H. Dual system encryption via predicate encodings. *Proceedings of Theory of Cryptography - 11th Theory of Cryptography Conference, TCC 2014*, Springer, San Diego, CA, USA, 2014; 616–637.
23. Chen J, Gay R, Wee H. Improved dual system abe in prime-order groups via predicate encodings, *Proceedings of Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, Sofia, Bulgaria, 2015; 595–624.
24. Pedersen TP. Non-interactive and information-theoretic secure verifiable secret sharing. *Proceedings Advances in Cryptology - CRYPTO '91, 11th Annual International Cryptology Conference*, Springer, Santa Barbara, California, USA, 1992; 129–140.
25. Waters B. Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization. *Proceedings of Public Key Cryptography - PKC 2011 - 14th International Conference on Practice and Theory in Public Key Cryptography*, Springer, Taormina, Italy, 2011; 53–70.
26. Hohenberger S, Waters B. Attribute-based encryption with fast decryption. *Proceedings of Public-Key Cryptography - PKC 2013 - 16th International Conference on Practice and Theory in Public-Key Cryptography*, Springer, Nara, Japan, 2013; 162–179.