Designs, Codes and Cryptography
https://doi.org/10.1007/s10623-024-01373-2

Anonymous attribute-based broadcast encryption with hidden multiple access structures

Tran Viet Xuan Phuong[1,2]

Received: 2 November 2022 / Revised: 20 November 2023 / Accepted: 24 January 2024

Abstract
Due to the high demands of data communication, the broadcasting system streams the data daily. This service not only sends out the message to the correct participant but also respects the security of the identity user. In addition, when delivered, all the information must be protected for the party who employs the broadcasting service. Currently, Attribute-Based Broadcast Encryption (ABBE) is useful to apply for the broadcasting service. (ABBE) is a combination of Attribute-Based Encryption (ABE) and Broadcast Encryption (BE), which allows a broadcaster (or encrypter) to broadcast an encrypted message, including a predefined user set and specified access policy to install the authorization mechanism. It is desirable to hide all the information when producing in the ciphertext, which has not been considered in the previous works of ABBE. Motivated by the above issue, we devise a solution to achieve anonymity for the ABBE scheme, which not only hides the access structures but also anonymizes the user's identity. In this work, we propose two schemes as Anonymous Key Policy (AKP)-ABBE and Anonymous Ciphertext Policy (ACP)-ABBE with supporting multiple access structures by using OR/AND gates. Specifically, we present the generic constructions of AKP/ACP-ABBE on the building block of the Inner Product Encryption (IPE), which enables the hidden user's identity and complex OR/AND-Gate access structure. We show that our proposed schemes are secured under the standard models.

Keywords Anonymous · Broadcast · Attribute based · Viète · OR/AND gates · Wildcard · Polynomial · Root

Mathematics Subject Classification 35A01 · 65L10 · 65L12 · 65L20 · 65L70

Communicated by K. Matsuura.

✉ Tran Viet Xuan Phuong
ptran@ualr.edu

[1] School of Cybersecurity, Old Dominion University, Norfolk, VA 23529, USA

[2] Department of Computer Science, University of Arkansas at Little Rock, Little Rock, AR 72204, USA

Published online: 06 April 2024

1 Introduction

The resources of the broadcasting channel are protected by allowing only the authorized person to be accessible to these resources [27]. By installing the access control into this channel, the individual has the attributes that will be attested to participate in the system. In addition, the access control is set up not only by the identity's user but also by the predicate of attributes (age, career, address, etc.). Currently, there are many broadcasting systems that integrate fine-grained access control into the authorization of user-accessible such as mobile pay-TV [35], 5G direct access via satellite [12], and Internet of Thing [24]. The incorporation of access control to the broadcasting systems not only controls the filtered users when using the service but also prevents from an unauthorized attempt to the system.

Among all the existing cryptographic tools, Attribute-Based Broadcast Encryption (ABBE) [22, 36] is well-suitable to construct an efficient mechanism. It creates the complicated access control that enables the broadcasting system. ABBE is a combination of Attribute-Based Encryption (ABE) [4, 6, 11, 16, 21, 28, 32] and Broadcast Encryption (BE) [5, 7, 14]. In the first proposal, the ABBE [22] allows the broadcaster to select groups of users defined by their attributes. This scheme is restricted the access policy for the group of users who satisfy the access policy can decrypt the ciphertext of broadcast encryption scheme. Technically, a user joint to the ABBE system is issued by a secret key SK associated with a user identity ID and a set of user's attributes L. Then the broadcaster who launched the ABBE system creates a ciphertext CT, which is associated with a list S of the user's identity and an access policy W. In addition, the access policy W is expressed by the predicate of the specified attributes. In the end, a user whose SK can decrypt the ciphertext CT if and only if the user ID belongs to the set S of valid user's identity, and the user's attributes L satisfies access policy W.

Motivation The Pay TV system wants to uphold customer service by offering exclusive prices and benefits. The system selects promising customers to participate in this campaign. However, all the information, including the price, benefits, and customers, cannot be unveiled publicly. Only the authorized person can intercommunicate with the system to obtain this information. For example, the broadcaster encrypts the data associated with the group of k customers $\{\mathsf{ID}_1, \mathsf{ID}_2, \ldots, \mathsf{ID}_k\}$, and the access policy as "(Town A AND Age $>$ 22 AND No home-phone line) OR (Town C AND Registered home-phone line)". Therefore, the broadcaster needs to protect all the information when public on the channel. Indeed, suppose the access policy is hidden when producing in the ciphertext. In that case, the competitor/the adversary can not extract the customer's information and learn from Pay TV's strategy to attract customers. Eventually, the customers who have satisfied the access structure can subscribe to their favorite channels. The existing ABBE schemes [10, 20, 30, 33, 36] have not considered the issue of hidden access policy when generating the ciphertext to deliver in the broadcasting channel.

Contribution In order to anonymize both the information of the group of ID users and the access structures, this work proposes two Anonymous Key Policy Attribute Based Broadcast Encryption (AKP-ABBE) and Anonymous Ciphertext Policy Attribute Based Broadcast Encryption (ACP-ABBE) schemes. Our proposed schemes can hide the information of the group of ID users and the access structures when delivering to the broadcasting system. The access structure is expressed by the predicate of positive and negative attributes, which are concatenated by the Boolean gates (AND, OR). Formally, both the descriptions of AKP-ABBE and ACP-ABBE are similar to KP-ABBE [2, 30] and CP-ABBE [2, 30]. To strengthen the anonymity, we devise the solution to adapt two schemes KP-ABBE, CP-ABBE with

OR/AND Gates with positive, negative attributes by exploiting the "*attribute-hiding*" Inner Product Encryption (IPE) [1, 3, 9, 19, 23, 26, 29] to achieve the A-KP-ABBE and A-CP-ABBE. We then enable the generic constructions for AKP-ABBE, ACP-ABBE.

In AKP-ABBE, to generate the ciphertext, we input a set of indices S and an attribute list L containing positive, and negative attributes. We create the polynomial $\mathcal{P}_{\mathsf{S}}$ from all the n elements of set S. In order to generate the coefficient of $\mathcal{P}_{\mathsf{S}}$, we apply the Viète theorem [31] to compute all the coefficients $(a_n, a_{n-1}, \ldots, a_1, a_0)$ of polynomial by using the all the elements of set S. Additionally, we aggregate all the attributes in the list L into one value b, then generate $(b^m, b^{m-1}, \ldots, 1)$, where m is the total attributes in L. Subsequently, we produce the ciphertext by calling the IPE's encryption with the input of $\vec{v} = (a_n, a_{n-1}, \ldots, a_1, a_0, b^m, b^{m-1}, \ldots, 1)$ and message M. In order to generate the secret key, we input a user ID and the complex access structure $\mathsf{W} = (\underbrace{(\mathsf{AND}_{i\in\{1,\ldots,m\}} A_i)}_{\mathsf{W}_1} \mathsf{OR} \underbrace{(\mathsf{AND}_{i\in\{1,\ldots,m\}} A_i)}_{\mathsf{W}_2} \mathsf{OR} \ldots \mathsf{OR} \underbrace{(\mathbf{AND}_{i\in\{1,\ldots,m\}} A_i)}_{\mathsf{W}_m})$. We encodes ID to integer value x_{ID}, then, generate as $(x_{\mathsf{ID}}^n, x_{\mathsf{ID}}^{n-1}, x_{\mathsf{ID}}^{n-2}, \ldots, 1)$. Similarly, we create the polynomial $\mathcal{P}_{\mathsf{W}}$ from the set of $(\mathsf{W}_1, \mathsf{W}_2, \ldots, \mathsf{W}_m)$ by Viète theorem, and obtain $(b_m, b_{m-1}, \ldots, b_1, b_0)$. We then produce the secret key by calling the IPE's key generation with the input of $\vec{x} = (x_{\mathsf{ID}}^n, x_{\mathsf{ID}}^{n-1}, x_{\mathsf{ID}}^{n-2}, \ldots, 1, b_m, b_{m-1}, \ldots, b_1, b_0)$. As a result, if the inner product of $(\vec{v}, \vec{x})$ equals zero, the IPE's decryption will return the message M. This means that the ID belongs to set S, and the attribute list L satisfies the access structure W. Mathematically, x_{ID} and aggregated b of L are the roots of polynomial $\mathcal{P}_{\mathsf{S}}$ and $\mathcal{P}_{\mathsf{W}}$, respectively.

On the other hand, ACP-ABBE is a inversion form of AKP-ABBE. A set of indices S and a complex access structure W into a vector $\vec{v}$, which is used for encryption. The user identity ID and user's attributes L containing positive and negative symbols is transformed into another vector $\vec{x}$, which is used in key generation. The decryption is successful if the ID belongs to set S, and the attribute list L satisfies the access structure W.

Our proposed schemes utilize the IPE manner to achieve the hidden access structures. Hence, we apply the security proof of IPE scheme in [19, 26] to prove that our AKP-ABBE and ACP-ABBE are secure in the standard model. We then compare with ABBE schemes to show our efficiency regarding hidden access structures and anonymity. Moreover, the generic constructions for AKP-ABBE, ACP-ABBE can be applied to many cryptography preliminaries to achieve the anonymous for ABBE schemes.

Related work Several ABBE schemes [2, 18, 22, 30] have been proposed in the literature. In [22], Lubicz and Sirvent proposed a CP-ABBE scheme which allows access policies to be expressed in disjunctive normal form, with the OR function provided by ciphertext concatenation. Attrapadung and Imai [2] proposed two KP-ABBE and two CP-ABBE schemes, which are constructed by algebraically combining some existing BE schemes (namely, the Boneh–Gentry–Waters BE scheme [7] and the Sahai–Waters BE scheme [29]) with some existing ABE schemes (namely, the KP-ABE scheme by Goyal et al. [16] and the CP-ABE scheme by Waters [32]). Junod and Karlov [18] also proposed a CP-ABBE scheme that supports boolean access policies with AND, OR and NOT gates. Junod and Karlov's scheme achieved direct revocation by simply treating each user's identity as a unique attribute in the attribute universe. In [30] scheme has proposed CP-ABBE and KP-ABBE scheme, which is constant ciphertext size with AND Gates positive, negative attributes and wildcard. In addition, [10] presented an efficient constant-size private key ciphertext-policy ABBE scheme for disjunctive normal form supporting fast decryption, and [34] proposed an efficient ciphertext-policy attribute-based encryption scheme for partially hidden policy, direct revocation, and verifiable outsourced decryption. However, most of current ABBE schemes do not concern

about the anonymous access structures, which are essential when outsourcing the data in the broadcasting system.

Attribute-based encryption (ABE) [4, 6, 11, 16, 21, 28, 32], which was introduced by Sahai and Waters [28] and extensively studied in recent years [6, 16, 21, 32], provides a fine-grained access control of encrypted data. In a Ciphertext Policy Attribute Based Encryption (CP-ABE) system, the secret user key is associated with a set of attributes, and the ciphertext is associated with an access policy. The ciphertext can be decrypted by a secret key if and only if the attributes associated with the secret key satisfy the access policy. A Key Policy ABE (KP-ABE) the system can be defined in a similar way by swapping the positions of the attributes and the access policy. In BE setting, a center is allowed to broadcast a secret to any subset of privileged users out of a universe of size n so that conjunctions of k users not in the privileged set cannot learn the secret. Apart from this, several broadcast encryption schemes were adopted with many interesting problems as [7, 8, 13, 15, 17] with solutions for collusion resistance, trace, and revoke for BE.

1.1 Paper organization

We present the preliminaries and definitions in Sect. 2, which is followed by our generic constructions in Sect. 3 and our analyzing the security proof in Sect. 4. We discuss the extensions in Sect. 5, then give the comparisons in Sect. 6. The paper is concluded in Sect. 7.

2 Preliminaries

2.1 Bilinear map and its related assumptions

Let $\mathbb{G}$ and $\mathbb{G}_{\mathbb{T}}$ be two multiplicative cyclic groups of same prime order p. Let e: $\mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_{\mathbb{T}}$ be a bilinear map with the following properties:

1. Bilinearity: $e(u^a, v^b) = e(u^b, v^a) = e(u, v)^{ab}$. for any $u, v \in \mathbb{G}$ and $a, b \in \mathbb{Z}_p$.
2. Non-degeneracy: $e(g, g) \neq 1$.

Definition 1 The Decisional Bilinear Diffie–Hellman (DBDH) problem in $\mathbb{G}$ is defined as follows: given a tuple $(g, g^a, g^b, g^c, T) \in \mathbb{G}^4 \times \mathbb{G}_T$, decide whether $T = e(g, g)^{abc}$ or $T = e(g, g)^r$ where a, b, c, r are randomly selected from $\mathbb{Z}_p$. An algorithm A has advantage ϵ in solving the DBDH problem in $\mathbb{G}$ if

$$\begin{aligned} \mathrm{Adv}_A^{\mathrm{DBDH}}(k) = &\Pr[A(1^k, g, g^a, g^b, g^c, Z) = 1 \| Z = e(g, g)^{abc}] \\ &- \Pr[A(1^k, g, g^a, g^b, g^c, Z) = 1 \| Z = g^r] \leq \epsilon. \end{aligned}$$

We say that the DBDH assumptions holds in $\mathbb{G}$ if ϵ is negligible for any PPT algorithm A.

Definition 2 The Decisional Linear (DLIN) problem in $\mathbb{G}$ defined as follows: given a tuple $(g, g^a, g^b, g^{ac}, g^d, Z) \in \mathbb{G}^5 \times \mathbb{G}_T$, decide whether $T = g^{b(c+d)}$ or Z in random in $\mathbb{G}$. An algorithm A has advantage ϵ in solving the DLIN problem in $\mathbb{G}$ if

$$\begin{aligned} \mathrm{Adv}_A^{\mathrm{DLIN}}(k) = &\Pr[A(1^k, g, g^a, g^b, g^{ac}, g^d, Z) = 1 \| Z = g^{b(c+d)}] \\ &- \Pr[A(1^k, g, g^a, g^b, g^{ac}, g^d, Z) = 1 \| Z = g^r] \leq \epsilon \end{aligned}$$

where $a, b, c, d, r \in_R \mathbb{Z}_p$. We say that the DLIN assumptions holds in $\mathbb{G}$ if ϵ is negligible for any PPT algorithm A.

2.2 Anonymous key-policy attribute based broadcast encryption definition

Let U denote the set of all user indices and N as the set of all user attributes. An Anonymous Key-Policy Attribute Based Broadcast Encryption (AKP-ABBE) scheme consists of four algorithms:

- $\mathsf{Setup}(1^\lambda)$ The setup algorithm takes the security parameter 1^λ as input and outputs the public parameters PK, and a master key MSK.
- $\mathsf{Encrypt}(\mathsf{M}, \mathsf{S}, \mathsf{L}, \mathsf{PK})$ The encryption algorithm takes as input the public parameters PK, a message M, a set of user index $\mathsf{S} \subseteq \mathsf{U}$, a set of attributes $\mathsf{L} \subseteq \mathsf{N}$, and outputs a ciphertext as CT.
- $\mathsf{KeyGen}(\mathsf{ID}, \mathsf{W}, \mathsf{MSK}, \mathsf{PK})$ The key generation algorithm takes as input the master key MSK, public parameters PK, a user index $\mathsf{ID} \in \mathsf{U}$, an access structure W, and outputs a user secret key SK.
- $\mathsf{Decrypt}(\mathsf{CT}, \mathsf{SK})$ The decryption algorithm takes as input a ciphertext CT, and a private key SK, then it outputs a message M or an error symbol '$\perp$'.

Security definition for AKP-ABBE We define the Selective IND-CPA security for AKP-ABBE via the following game.

- **Init** The adversary commits to the challenge user indices $(\mathsf{S}_0^*, \mathsf{S}_1^*)$ and target attribute sets $(\mathsf{L}_0^*, \mathsf{L}_1^*)$.
- **Setup** The challenger runs the Setup algorithm and gives PK to the adversary.
- **Phase 1** The adversary queries for private keys with pairs of user index and access structure $(\mathsf{ID}, \mathsf{W})$ following the cases:
  - $(\mathsf{L}_0^* \not\models \mathsf{W}$ and $(\mathsf{L}_1^* \not\models \mathsf{W})$ or $(\mathsf{ID} \notin \mathsf{S}_0^*$ and $\mathsf{ID} \notin \mathsf{S}_1^*)$.
  - $(\mathsf{L}_0^* \models \mathsf{W}$ and $(\mathsf{L}_1^* \models \mathsf{W})$ and $(\mathsf{ID} \in \mathsf{S}_0^*$ and $\mathsf{ID} \in \mathsf{S}_1^*)$.

  Then the challenger gives the adversary the corresponding secret key SK. Otherwise, it outputs $\perp$.
- **Challenge** The adversary submits the two messages $\mathsf{M}_0, \mathsf{M}_1$ to the challenger with respect to the challenge user indices $(\mathsf{S}_0^*, \mathsf{S}_1^*)$ and target attribute sets $(\mathsf{L}_0^*, \mathsf{L}_1^*)$. The challenger flips a random coin β and passes the ciphertext $\mathsf{CT}^* = \mathsf{Encrypt}(\mathsf{PK}, \mathsf{M}_\beta, \mathsf{L}_\beta^*, \mathsf{S}_\beta^*)$ to the adversary.
- **Phase 2** Phase 1 is repeated.
- **Guess** The adversary outputs a guess β' of β.

Definition 1 We say an AKP-ABBE scheme is selective IND-CPA secure if for any probabilistic polynomial time adversary

$$\mathbf{Adv}_{kp}^{\text{s-ind-cpa}}(\lambda) = |\Pr[\beta' = \beta] - 1/2|$$

is a negligible function of λ.

2.3 Anonymous ciphertext-policy attribute-based broadcast encryption definition

An Anonymous Ciphertext-Policy Attribute-Based Broadcast Encryption (ACP-ABBE) scheme consists of four algorithms:

- $\mathsf{Setup}(1^\lambda)$: The setup algorithm takes the security parameter 1^λ as input and outputs the public parameters PK, and a master key MSK.

- Encrypt(M, §, W, PK): The encryption algorithm takes as input the public parameters PK, a message M, a set of user index S ⊆ U, and an access structure W, then outputs a ciphertext as CT.
- KeyGen(ID, L, MSK, PK): The key generation algorithm takes as input the master key MSK, public parameters PK, a user index ID ∈ U, and a set of attributes L ⊆ N, and outputs a user secret key SK.
- Decrypt(CT, SK): The decryption algorithm takes as input a ciphertext CT, and a private key SK, then outputs a message M or an error symbol '⊥'.

Security definition for ACP-ABBE We define the Selective IND-CPA security for ACP-ABBE via the following game.

- **Init** The adversary commits to the challenge user indices $(\mathsf{S}_0^*, \mathsf{S}_1^*)$ and target access structures $(\mathsf{W}_0^*, \mathsf{W}_1^*)$.
- **Setup** The challenger runs the Setup algorithm and gives PK to the adversary.
- **Phase 1** The adversary queries for private keys with pairs of user index and a user attribute list (ID, L) following the cases:

– $(\mathsf{L} \not\models \mathsf{W}_0^*$ and $(\mathsf{L} \not\models \mathsf{W}_1^*)$ or $(\mathsf{ID} \notin \mathsf{S}_0^*$ and $\mathsf{ID} \notin \mathsf{S}_1^*)$.
– $(\mathsf{L} \models \mathsf{W}_0^*$ and $(\mathsf{L} \models \mathsf{W}_1^*)$ and $(\mathsf{ID} \in \mathsf{S}_0^*$ and $\mathsf{ID} \in \mathsf{S}_1^*)$.

- **Challenge** The adversary submits messages $\mathsf{M}_0, \mathsf{M}_1$ to the challenger with respect to the challenge user indices $(\mathsf{S}_0^*, \mathsf{S}_1^*)$ and target access structures $(\mathsf{W}_0^*, \mathsf{W}_1^*)$. The challenger flips a random coin β and passes the ciphertext $\mathsf{CT}^* = \mathsf{Encrypt}(\mathsf{PK}, \mathsf{M}_\beta, \mathsf{W}_\beta^*, \mathsf{S}_\beta^*)$ to the adversary.
- **Phase 2** Phase 1 is repeated.
- **Guess** The adversary outputs a guess β' of β.

Definition 2 We say a ACP-ABBE scheme is selective IND-CPA secure if for any probabilistic polynomial time adversary

$$\mathbf{Adv}_{cp}^{\text{s-ind-cpa}}(\lambda) = |\Pr[\beta' = \beta] - 1/2|$$

is a negligible function of λ.

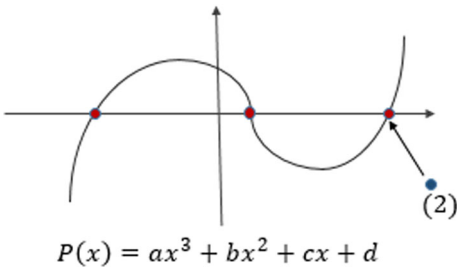
2.4 Inner product encryption

Let $\Sigma \in \mathbb{Z}$ be the set of attributes involving vectors $\vec{v}$ of dimension n, and $\mathcal{F}$ be the class of predicates involving inner-products over vectors $\mathcal{F} = \{f_{\vec{v}}, \vec{v} \in \Sigma\}$ such that $f_{\vec{v}}(\vec{x}) = 1$ iff $< \vec{v}, \vec{x} >= 0$. An inner-product encryption (IPE) scheme for the class of predicate $\mathcal{F}$ over the set of attributes consists of four algorithms as follows:

- IPE.Setup$(1^\lambda, n)$ on input a security parameter 1^λ and the vector length $n = poly(\lambda)$, the algorithm outputs a public key PK and a master secret key MSK.
- IPE.Encrypt(M, PK, $\vec{v} = (v_1, v_2, \ldots, v_n)$): on input a message M, the public key PK, and a vector $\vec{v} \in \Sigma^n$, it outputs a ciphertext CT.
- IPE.KeyGen(MSK, $\vec{x} = (x_1, x_2, \ldots, x_n)$): on input the master secret key MSK, a vector $\vec{x} \in \Sigma$, the algorithm outputs a secret key SK.
- IPE.Decrypt(CT, SK): on input a secret key SK (w.r.t. a vector $\vec{x}$) and a ciphertext CT (w.r.t. a vector $\vec{v}$), if $f_{\vec{v}}(\vec{x}) = 0$, the algorithm outputs a message M; otherwise, it outputs $\perp$.

Security model IPE scheme Following [19], we define the security, i.e., attribute-hiding property, of the IPE scheme. The security is defined by the following game interacted between an attacker $\mathcal{A}$ and a challenger $\mathcal{C}$. We assume that $(\Sigma, \mathcal{F})$ are given to both $\mathcal{A}$ and $\mathcal{C}$ in advance.

Fig. 1 Checking one root of $\mathcal{P}$



- **Init** $\mathcal{A}$ outputs two vectors $\vec{v}, \vec{x} \in \Sigma$
- **Setup** $\mathcal{C}$ runs Setup to obtain the public key PK and master secret key MSK. $\mathcal{A}$ is given PK.
- **Query Phase 1** $\mathcal{A}$ adaptively issues private key queries for any vectors $\vec{v_1}, \ldots, \vec{v_n} \in \Sigma$, subject to the restriction that, $\forall i, < \vec{v_i}, \vec{x} >= 0$ if and only if $< \vec{v_i}, \vec{x} >= 0$. $\mathcal{C}$ responds with $\mathsf{SK}_{\vec{v_i}} \leftarrow \mathsf{KeyGen}(\mathsf{SK}, \vec{v_i})$.
- **Challenge** $\mathcal{A}$ outputs two messages $\mathsf{M}_0, \mathsf{M}_1$ with equal length. If $\mathsf{M}_0 \neq \mathsf{M}_1$, then it is required that $< \vec{v}, \vec{x} > \neq 0 \neq < \vec{x}, \vec{x} >$ for any $\vec{x}$ appeared in Query Phase 1. $\mathcal{C}$ flips a random coin $b \in \{0, 1\}$. If $b = 0$, $\mathcal{C}$ returns $\mathsf{CT} \leftarrow \mathsf{Encryption}(\mathsf{PK}, \vec{v}, \mathsf{M}_0)$ to $\mathcal{A}$; otherwise, if $b = 1$, $\mathcal{C}$ returns $\mathsf{CT} \leftarrow \mathsf{Encrypt}(\mathsf{PK}, \vec{x}, \mathsf{M}_1)$ to $\mathcal{A}$.
- **Query Phase 2** Phase 1 is repeatedly.
- **Guess** $\mathcal{A}$ outputs a guess bit b' and succeeds if $b' = b$.

The advantage of $\mathcal{A}$ in this game is defined as $Adv_{\mathcal{A}}(\lambda) = \Pr[b' = b] - \frac{1}{2}$.

Definition 3 We say that an IPE scheme is attribute-hiding if for all polynomial time adversaries $\mathcal{A}$, we have that $Adv(\mathcal{A})$ is negligible.

In fact, the challenge ciphertext is given to $\mathcal{A}$ as: if $b = 0$ then $\mathsf{CT} \leftarrow \mathsf{Encrypt}(\mathsf{PK}, \vec{v}, \mathsf{M}_0)$ and if $b = 1$ then $\mathsf{CT} \leftarrow \mathsf{Encrypt}(\mathsf{PK}, \vec{x}, \mathsf{M}_1)$. As well as similar $Adv(\mathcal{A})$ to the one above, we say that a IPE scheme is attribute-hiding if for all polynomial time adversaries $\mathcal{A}$, we have that $Adv(\mathcal{A})$ is negligible.

2.5 Polynomial and roots

Consider that a polynomial $\mathcal{P}$ has degree n is defined as:

$$\mathcal{P} = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0 \tag{1}$$

We then extract the coefficients of P to create a vector $\vec{v}$ as follows:

$$\vec{v} = (a_n, a_{n-1}, \ldots, a_1, a_0).$$

In addition, we create the a vector $\vec{x}$ by choosing a integer value x randomly as follows:

$$\overrightarrow{x} = \left(\underbrace{x \cdot x \cdots x}_{\text{n}}, \underbrace{x \cdots x}_{\text{n-1}}, \ldots, x, 1 \right)$$

If $(\vec{v} \cdot \overrightarrow{x}) = 0$, then we conclude that x is a root of polynomial $\mathcal{P}$ (Fig. 1).

Table 1 List of attributes and AND positive/negative attributes policies

Attributes Description	Att_1 CS	Att_2 EE	Att_3 Professor	Att_4 Faculty	Att_5 Student	Att_6 Tutor
Alice	+	−	−	−	+	+
Bob	−	+	−	+	−	−
Carol	+	+	−	+	−	−
W_1	+	−	−	−	+	+

2.6 Consequence of Viète formula

We apply consequence of the Viète's formula to reconstruct all the coefficients of $\mathcal{P}$ in (1) as follows:

$$\begin{cases} x_1 + x_2 + \ldots + x_n & = \left(-\frac{a_{n-1}}{a_n}\right) \\ (x_1x_2 + x_1x_3 + \ldots + x_1x_n) & \\ +(x_2x_3 + x_3x_4 + \ldots + x_2x_n) + \ldots + x_{n-1}x_n & = \left(\frac{a_{n-2}}{a_n}\right) \\ \ldots & \\ x_1x_2 \ldots x_n & = (-1)^n \frac{a_0}{a_n} \end{cases}$$

Generally, we write: $\sum_{1 \le i_1 < i_2 < \cdots < i_k \le n} x_{i_1} x_{i_2} \cdots x_{i_k} = (-1)^k \frac{a_{n-k}}{a_n}$ for $k = 1, 2, ..., n$. Apart from Sect. 2.4, we can rewrite the $\vec{v}$ as

$$\vec{v} = \left(1, -\frac{a_{n-1}}{a_n}, \ldots, (-1)^n \frac{a_1}{a_n}, (-1)^n \frac{a_0}{a_n}\right).$$

Then we have the $\vec{x} = (x^n, x^{n-1}, \ldots, x, 1)$. If $< \vec{v} \cdot \overrightarrow{x} >= 0$, then we conclude that x is a root of polynomial $\mathcal{P}$.

3 Generic constructions

3.1 AND/OR gates access structure

3.1.1 AND gates positive/negative attributes

Let $\mathsf{U} = \{\mathsf{Att}_1, \mathsf{Att}_2, ..., \mathsf{Att}_n\}$ be the universe of the attributes in the system. Each Att_i is represented by a unique value A_i. When a user joins the system, the user is tagged with an attribute list defined as $\mathsf{S} = \{\mathsf{S}_1, \mathsf{S}_2, ..., \mathsf{S}_n\}$ where each symbol S_i has two possible values: '+' and '−'. Let $\mathsf{W} = \{\mathsf{S}'_1, \mathsf{S}'_2, ..., \mathsf{S}'_n\}$ denote an AND-gate access policy where each symbol S'_i has two possible values: '+', '−'. We use the notation $\S \models \mathsf{W}$ to denote that the attribute list S of a user satisfies W.

We illustrate the AND gates with positive/negate attribute by the following example. Suppose that $\mathsf{U} = \{\mathsf{Att}_1 = \text{CS}, \mathsf{Att}_2 = \text{EE}, \mathsf{Att}_3 = \text{Professor}, \mathsf{Att}_4 = \text{Faculty}, \mathsf{Att}_5 = \text{Student}, \mathsf{Att}_6 = \text{Tutor}\}$. Alice is a student and tutor in the CS department; Bob is a faculty in the EE department; Carol is a faculty holding a joint position in the EE and CS departments. All attribute lists are expressed in Table 1. In addition, the access structure W_1 is designed to allow all the CS students and tutors in only CS departments to access to the system.

Table 2 List of attributes and AND then OR policies

Attributes Description	Att_1 CS	Att_2 EE	Att_3 Professor	Att_4 F.Officer	Att_5 Student	Att_6 Tutor
W_{11}	+	−	+	−	−	−
OR						
W_{12}	+	−	−	−	+	+

Observably, only Alice is the student/tutor of CS departments, which is attested to access to the system since the Alice's attributes satisfy the access structure W_1.

3.1.2 Multiple OR/AND gates

In this work, we consider the complex access structures, which are expressed the predicate of attributes by both of the OR and AND gates.

Suppose that we have an access structures W_1 as follows:

$$\mathsf{W}_1 = ((\mathsf{AND}_{i\in\{1,\ldots,m\}} A_i)\mathsf{OR}(\mathsf{AND}_{i\in\{1,\ldots,m\}} A_i)\mathsf{OR}\ldots\mathsf{OR}(\mathsf{AND}_{i\in\{1,\ldots,m\}} A_i)$$

as the Disjunctive Normal Form (DNF). Utilizing the set of attributes $\mathsf{U} = \{\mathsf{Att}_1, \mathsf{Att}_2, \ldots, \mathsf{Att}_n\}$ in AND gate access structure, W_1 is expressed as:

$$\mathsf{W}_1 = (\underbrace{(\mathsf{Att}_1\ \mathsf{AND}\ \mathsf{Att}_3))}_{\mathsf{W11}}\ \mathsf{OR}\ \underbrace{(\mathsf{Att}_1\ \mathsf{AND}\ \mathsf{Att}_6\ \mathsf{AND}\ \mathsf{Att}_5))}_{\mathsf{W12}},$$

Regarding the Table 2, we decouple W_1 into the two access structures W_{11} and W_{12}. Then if a user has the set of attributes satisfy W_{11} or W_{12}, the he is valid to decrypt the message.

Next, we consider the Conjunctive Normal Form (CNF) access structures W_2 as follows:

$$W_2 = ((\mathsf{OR}_{i\in\{1,\ldots,m\}} A_i)\mathsf{AND}(\mathsf{OR}_{i\in\{1,\ldots,m\}} A_i)\mathsf{AND}\ldots\mathsf{AND}_{m-1}(\mathsf{OR}_{i\in\{1,\ldots,m\}} A_i)).$$

In practice, W_2 is expressed by the attributes in set U as:

$$\mathsf{W}_2 = ((\mathsf{Att}_1\ \mathsf{OR}\ \mathsf{Att}_2))\ \mathsf{AND}\ (\mathsf{Att}_3\ \mathsf{OR}\ \mathsf{Att}_4)).$$

We then transform the W_2 in the other observation:

$$\mathsf{W}_2 = \underbrace{((\mathsf{Att}_1\ \mathsf{AND}\ \mathsf{Att}_3)}_{\mathsf{W}_21}\ \mathsf{OR}\ \underbrace{(\mathsf{Att}_1\ \mathsf{AND}\ \mathsf{Att}_4)}_{\mathsf{W22}}\ \mathsf{OR}\ \underbrace{(\mathsf{Att}_2\ \mathsf{AND}\ \mathsf{Att}_3))}_{\mathsf{W23}}$$
$$\mathsf{OR}\ \underbrace{(\mathsf{Att}_2\ \mathsf{AND}\ \mathsf{Att}_4))}_{\mathsf{W24}}$$

Regarding the Table 3, we interpret W_2 into the the set of access structures $(\mathsf{W}_{21}, \mathsf{W}_{22}, \mathsf{W}_{23}, \mathsf{W}_{24})$. Then if a user has the set of attributes satisfy W_{21} or W_{22} or W_{23} or W_{24}, then he is valid to decrypt the message. As a result, we realize that when a user joins the system, the user is tagged with an attribute list defined as $\mathsf{S} = \{A_i\}_{i\in\{1,m\}}$. We conclude the two statements as follows:

- $\mathsf{S} \models \mathsf{W}_1$, if the set attributes in S satisfies **one** of AND literals in W_1.
- $\mathsf{S} \models \mathsf{W}_2$, if the set attributes in S satisfies **all** of OR literals in W_2.

Table 3 List of attributes and OR then AND policies

Attributes Description	Att_1 CS	Att_2 EE	Att_3 Professor	Att_4 Faculty	Att_5 Student	Att_6 Tutor
W_{21}	+	−	+	−	−	−
OR						
W_{22}	+	−	−	+	−	−
OR						
W_{23}	−	+	+	−	−	−
OR						
W_{24}	−	+	−	+	−	−

3.2 Original IPE construction

In this section, we represent the original of IPE scheme [25], which is a building block to construct our proposed work later.

Setup$(1^k, n)$: The setup algorithm first randomly generates $(g, \mathbb{G}, \mathbb{G}_T, p, e)$ and n is the maximum length of vector. It then chooses randomly $\gamma_1, \gamma_2, \theta_1, \theta_2, \{u_{1,i}\}_{i=1}^n, t_1, \{t_{1,i}\}_{i=1}^n, \{t_{2,i}\}_{i=1}^n, \{w_{1,i}\}_{i=1}^n, \{z_{1,i}\}_{i=1}^n, \{z_{2,i}\}_{i=1}^n$ in $\mathbb{Z}_p$ and g_2 in $\mathbb{G}$. Then it selects a random $\Delta \in \mathbb{Z}_p$ and obtains $\{u_{2,i}\}_{i=1}^n, \{w_{2,i}\}_{i=1}^n, w_2, u_2$ under the condition: $\Delta = \gamma_1 u_{2,i} - \gamma_2 u_{1,i} \Delta = \theta_1 w_{2,i} - \theta_2 w_{1,i}$.
For i from 1 to n, it creates:

$$U_{1,i} = g^{u_{1,i}}, U_{2,i} = g^{u_{2,i}}, W_{1,i} = g^{w_{1,i}}, W_{2,i} = g^{w_{2,i}}, T_{1,i} = g^{t_{1,i}}, T_{2,i} = g^{t_{2,i}},$$
$$Z_{1,i} = g^{z_{1,i}}, V_1 = g^{\gamma_1}, V_2 = g^{\gamma_2}, X_1 = g^{\theta_1}, V_2 = g^{\theta_2}.$$

Next it sets $g_1 = g^{\Delta}$, $Y = e(g, g_2)$, and the public key PK and master key MSK as

$$\mathsf{PK} = (g, \mathbb{G}, \mathbb{G}_T, p, e, g_1, Y, \{U_{1,i}, U_{2,i}, T_{1,i}, T_{2.i}, W_{1.i}, W_{2,i}, Z_{1,i}, Z_{2,i}\}_{i=1}^n, \{V_i, X_i\}_{i=1}^2)$$
$$\mathsf{MSK} = (g_2, \{u_{1,i}, u_{2,i}, t_{1,i}, t_{2,i}, w_{1,i}, w_{2,i}, z_{1,i}, z_{2,i}\}_{i=1}^n, \{v_i, x_i\}_{i=1}^2).$$

Encrypt$(\mathsf{PK}, \vec{v}, \mathsf{M})$: The encryption algorithm chooses random $s_1, s_2, \alpha, \beta \in \mathbb{Z}_p$ and creates the ciphertext as follows:

$$C_m = M \cdot Y^{s_2}, C_A = g^{s_2}, C_B = g_1^{s_1},$$
$$\{C_{1,i}, C_{2,i}\} = \{U_{1,i}^{s_1} T_{1,i}^{s_2} V_1^{v_i \alpha}, U_{2,i}^{s_1} T_{2,i}^{s_2} V_2^{v_i \alpha}\},$$
$$\{C_{3,i}, C_{4,i}\} = \{W_{1,i}^{s_1} Z_{1,i}^{s_2} X_1^{v_i \beta}, W_{2,i}^{s_1} Z_{2,i}^{s_2} X_2^{v_i \beta}\},$$

where $\vec{v} = (v_1, \ldots, v_n)$, then ciphertext CT is set as:

$$\mathsf{CT} = (C_m, C_A, C_B, \{C_{1,i}, C_{2,i}, C_{3,i}, C_{4,i}\}_{i=1}^n.$$

KeyGen$(\mathsf{PK}, \vec{x}, \mathsf{MSK})$: The key generation algorithm chooses randomly $r_{i,1}, r_{i,2}$ for $i = 1$ to n, and $f_1, f_2, r_1, r_2 \in \mathbb{Z}_p$, and then creates the secret key as follows:

$$\{K_{1,i}, K_{2,i}\} = \{g^{-\gamma_2 r_{1,i}} g^{f_1 x_i u_{2,i}}, g^{\gamma_1 r_{1,i}} g^{-f_1 x_i u_{1,i}}\},$$
$$\{K_{3,i}, K_{4,i}\} = \{g^{-\theta_2 r_{2,i}} g^{f_2 x_i w_{2,i}}, g^{\theta_1 r_{2,i}} g^{-f_2 x_i w_{1,i}}\},$$
$$K_A = g_2 \cdot \prod_{i=1}^n K_{1,i}^{-t_{1,i}} K_{2,i}^{-t_{2,i}} K_{3,i}^{-z_{1,i}} K_{4,i}^{-z_{2,i}},$$
$$K_B = \prod_{i=1}^n g^{-(r_{1,i} + r_{2,i})}.$$

where $\vec{x} = (x_1, \ldots, x_n)$, the secret key is set as:

$$\mathsf{SK} = (K_A, K_B, \{K_{1,i}, K_{2,i}, K_{3,i}, K_{4,i}\}_{i=1}^{n}.$$

Decrypt(SK, CT): The decryption algorithm returns

$$\frac{C_m}{e(C_A,K_A)\cdot e(C_B,K_B)\prod_{j=1}^{4}\prod_{i=1}^{n} e(C_{j,i},K_{j,i})} = \frac{M}{e(g,g)^{(\sum_{i=0}^{n} v_i x_i)(f_1\alpha\Delta + f_2\beta\Delta)}}.$$

Therefore, the message M will be returned iff $(\vec{v}, \vec{x}) = 0$ meaning the attributes list in user key SK satisfies the access policy in the ciphertext CT.
Following the description of the above Multiple OR/AND gate access structures and the original IPE construction, we present two Anonymous Key Policy Attribute Based Broadcast Encryption and Anonymous Ciphertext Policy Attribute Based Broadcast Encryption schemes with OR/AND Gate with positive, negative attributes in access structure.

3.3 Generic construction of AKP-ABBE from IPE

In our AKP-ABBE scheme, we only consider two values, positive, negative, of attributes. In order to construct, we desire an (n + m)- dimensional IPE scheme, where n is the number of set indices, and m is the maximum number of access structures. In this scheme, we present the construction of DNF access structure since the CNF form can converse to the DNF.

Let U denote the set of all user indices, and N as the set of all user attributes and given an IPE scheme with four algorithms: (IPE.Setup, IPE.KeyGen, IPE.Enc, IPE.Dec), we construct an AKP-ABBE scheme with the corresponding four algorithms Setup, KeyGen, Encrypt, Decrypt, which we elaborate as follows:

Setup(1^k): The algorithm chooses a suitable encoding τ_1 sending each of the n indicies $\mathsf{ID} \in \mathbb{N}$ onto an element $\tau_1(\mathsf{ID}) = x_1 \in (\mathbb{Z}/\mathrm{p}\mathbb{Z})^*$, and choose $t_1, \ldots, t_{2_n}$ randomly in $\mathbb{Z}_p$. It runs IPE.Setup($1^k, n + m$) with m as the number of attributes to construct to access structure, and outputs public parameters PK and a master key MSK.
Encrypt(PK, M, S, L): The algorithm inputs a user index set $\mathsf{S} = \{\mathsf{ID}_a.\mathsf{ID}_b, \mathsf{ID}_c, \ldots \mathsf{ID}_s\} \subseteq \mathsf{U}$, and message M, attribute list L. The algorithm transforms (S, L) into $\vec{v}$ as:
The user index set is input as $\mathsf{S} = (\mathsf{ID}_a, \mathsf{ID}_b, \mathsf{ID}_c, \ldots, \mathsf{ID}_s) \subseteq \mathsf{U}$. We denote Δ as the total number elements in set S, then the algorithm applies the Viète's formula to compute:

$$\begin{cases} \tau_1(\mathsf{ID}_a) + \tau_1(\mathsf{ID}_b) + \tau_1(\mathsf{ID}_c) + \ldots + \tau_1(\mathsf{ID}_s) & = a_\Delta \\ (\tau_1(\mathsf{ID}_a)\tau_1(\mathsf{ID}_b) + \tau_1(\mathsf{ID}_a)\tau_1(\mathsf{ID}_c) + \ldots + \tau_1(\mathsf{ID}_a)\tau_1(\mathsf{ID}_s) & \\ \ldots + \tau_1(\mathsf{ID}_{\Delta-1})\tau_1(\mathsf{ID}_s) & = a_{\Delta-1} \\ \ldots & \\ \tau_1(\mathsf{ID}_a)\tau_1(\mathsf{ID}_b)\tau_1(\mathsf{ID}_c)\ldots\tau_1(\mathsf{ID}_s) & = a_0 \end{cases} \tag{2}$$

The algorithm converts an attribute user list L by generating:

$$\text{If } \begin{cases} \mathsf{Att}_i \text{ is } + & : r_i' = t_i \\ \mathsf{Att}_i \text{ is } - & : r_i' = t_{2_i} \end{cases} \tag{3}$$

Then set $b = \sum_{\mathsf{Att}_i \in \mathsf{L}} r'_i$, and it computes based on b:

$$\begin{cases} b_m & = b^m \\ b_{m-1} & = b^{m-1} \\ b_{m-2} & = b^{m-2} \\ \dots & \\ b_0 & = 1 \end{cases}$$

The $\vec{v}$ is produced as

$$\vec{v} = (1_0, a_\Delta, a_{\Delta-1}, \dots, a_0, b_m, \dots, 1_m).$$

Then it runs $\mathsf{CT} \leftarrow \mathsf{IPE.Enc}(\mathsf{PK}, \vec{v}, \mathsf{M})$, and output the ciphertext CT.

KeyGen($\mathsf{MSK}, \mathsf{ID}, \mathsf{W} = (\mathsf{W}_1 \ \mathsf{OR} \ \dots \ \mathsf{OR} \ \mathsf{W}_m)$): Suppose that a user joins the system with the a given user identity ID and the access structure $\mathsf{W} = (\mathsf{W}_1 \ \mathsf{OR} \ \dots \ \mathsf{OR} \ \mathsf{W}_m)$, the algorithm inputs (ID, W), and transforms them into a vector $\vec{z}$ by generating:
It encodes ID by $\tau_1(ID) = x_{\mathsf{ID}} \in (\mathbb{Z}/p\mathbb{Z})^*$. Then, we compute x_{ID} as the one of the roots of polynomial degree n:

$$\begin{cases} a'_n & = x_{\mathsf{ID}}^n \\ a'_{n-1} & = x_{\mathsf{ID}}^{n-1} \\ a'_{n-2} & = x_{\mathsf{ID}}^{n-2} \\ \dots & \\ a'_0 & = 1 \end{cases} \tag{4}$$

Next, the access structure W is interpreted as:

$$\mathsf{W} = (\underbrace{(\mathsf{AND}_{i\in\{1,\dots,m\}} A_i)}_{W_1} \ \mathsf{OR} \ \underbrace{\mathsf{AND}_{i\in\{1,\dots,m\}} A_i)}_{W_2} \ \mathsf{OR} \dots \mathsf{OR} \ \underbrace{(\mathsf{AND}_{i\in\{1,\dots,m\}} A_i)}_{\mathsf{W}_m}).$$

Then the algorithms computes as follows:

$$\text{Each } \mathsf{W}_i, \text{ If } \begin{cases} \mathsf{Att}_j \text{ is } + & : r_j = t_i, \\ \mathsf{Att}_j \text{ is } - & : r_j = t_{2_i} \end{cases};$$

Then set $\mathsf{W}_i = \sum_{\mathsf{Att}_j \in \mathsf{W}_i} r_j$.
Next apply the Viète's formula as (2) to computes the whole access structure W:

$$\begin{cases} \mathsf{W}_1 + \mathsf{W}_2 + \dots + \mathsf{W}_m & = b'_{m-1} \\ \mathsf{W}_1\mathsf{W}_2 + \mathsf{W}_1\mathsf{W}_3 + \dots + \mathsf{W}_{m-1}\mathsf{W}_m & = b'_{m-2} \\ \dots & \\ \mathsf{W}_1\mathsf{W}_2 \dots \mathsf{W}_m & = b'_0 \end{cases} \tag{5}$$

The $\vec{z}$ is produced as

$$\vec{z} = (a'_n, a'_{n-1}, \dots, 1_n, 1_m, b'_{m-1}, \dots, b'_0).$$

Then it runs $\mathsf{SK} \leftarrow \mathsf{IPE.KeyGen}(\mathsf{PK}, \vec{z}, \mathsf{MSK})$, and output the secret key SK.

Decrypt(SK, CT) the algorithms runs IPE.Dec(CT, SK) and outputs the message M iff $< \vec{v}, \vec{z} >== 0$.

Correctness for the vector $vecv = (1_0, a_\Delta, a_{\Delta-1}, \ldots, a_0, b_m, \ldots, 1_m))$ corresponding to the set user indices S and attribute list L in the ciphertext CT and the vector $\vec{z} = (a'_n, a'_{n-1}, \ldots, 1_n, 1_m, b'_{m-1}, \ldots, b'_0)$ corresponding to the secret key component SK in the AKP-ABBE, we have:

$$\begin{aligned}
\sum_{i=0}^{n+m} v_i.x_i &= \sum_{i=0}^{n} v_i.x_i + \sum_{i=n+1}^{n+m} v_i.x_i \\
&= \sum_{i=0}^{n} a_i \cdot x_{\mathsf{ID}}^i + \sum_{i=n+1}^{n+m} b_{i-n} \cdot b'_{i-n} \\
&= (1_0 \cdot x_{\mathsf{ID}}^n + a_\Delta \cdot x_{\mathsf{ID}}^{n-1} + \ldots + a_0 \cdot 1_n) \\
&\quad + \left(\left(\sum_{\mathsf{Att}_i \in \mathsf{L}} r'_i \right)^m \cdot 1_m + \left(\sum_{\mathsf{Att}_i \in \mathsf{L}} r'_i \right)^{m-1} \cdot b'_{m-1} + \ldots + b'_0 \right) \\
&= \left(x_{\mathsf{ID}}^n + [\tau_1(\mathsf{ID}_a) + \ldots + \tau_1(\mathsf{ID}_s)] \cdot x_{\mathsf{ID}}^{n-1} + \ldots + [\tau_1(\mathsf{ID}_a) \ldots \tau_1(\mathsf{ID}_s)] \right) \\
&\quad + \Bigg(\left(\sum_{\mathsf{Att}_i \in \mathsf{L}} r'_i \right)^m + \left(\sum_{\mathsf{Att}_i \in \mathsf{L}} r'_i \right)^{m-1} \cdot \left[\sum_{i=1}^{m} \sum_{\mathsf{Att}_j, j=1 \in \mathsf{W}_i}^{m} r_j \right] \\
&\quad + \ldots + \left[\prod_{i=1}^{m} \sum_{\mathsf{Att}_j, j=1 \in \mathsf{W}_i}^{m} r_j \right] \Bigg).
\end{aligned}$$

If $\sum_{i=0}^{n+m} v_i.x_i = 0$, the algorithm return M. This means that the ID user is belongs to the set of indices S, and the attribute list L satisfies the user's access structures $\mathbb{W}$. Otherwise, the algorithms return $\perp$.

Theorem 1 *Our AKP-ABBE scheme is secure under the standard assumption if the underlying IPE is secure under the standard assumption.*

3.4 Generic construction of ACP-ABBE from IPE

The ACP-ABBE scheme is a dual form of AKP-ABBE.

3.4.1 Main scheme

Given an IPE scheme with four algorithms: (IPE.Setup, IPE.KeyGen, IPE.Enc, IPE.Dec), we construct an ACP-AABBE scheme with the corresponding four algorithms: Setup, KeyGen, Encrypt, Decrypt) as follows:

Setup(1^k): The algorithm chooses a suitable encoding τ_1 sending each of the n indicies $\mathsf{ID} \in \mathbb{N}$ onto an element $\tau_1(\mathsf{ID}) = x_1 \in (\mathbb{Z}/\mathrm{p}\mathbb{Z})^*$, and choose $t_1, \ldots, t_{2_n}$ randomly in $\mathbb{Z}_p$. It runs IPE.Setup($1^k, n+m$) with m as the number of attributes to construct to access structure, and outputs public parameters PK and a master key MSK.

Encrypt$(\mathsf{PK}, \mathsf{M}, \mathsf{S}, \mathsf{W} = (\mathsf{W}_1\ \mathsf{OR} \ldots \mathsf{OR}\ \mathsf{W}_m))$: The algorithm inputs a user index set $\mathsf{S} = \{\mathsf{ID}_a.\mathsf{ID}_b, \mathsf{ID}_c, \ldots \mathsf{ID}_s\} \subseteq \mathsf{U}$, and message M, the access structure $\mathsf{W} = (\mathsf{W}_1\ \mathsf{OR} \ldots \mathsf{OR}\ \mathsf{W}_m)$.The algorithm transforms (S, W) into $\vec{v}$ as:

The user index set is input as $\mathsf{S} = (\mathsf{ID}_a, \mathsf{ID}_b, \mathsf{ID}_c, \ldots, \mathsf{ID}_s) \subseteq \mathsf{U}$. We denote Δ as the total number elements in set S, then the algorithm applies the Viète's formula to compute:

$$\begin{cases} \tau_1(\mathsf{ID}_a) + \tau_1(\mathsf{ID}_b) + \tau_1(\mathsf{ID}_c) + \ldots + \tau_1(\mathsf{ID}_s) & = a_\Delta \\ (\tau_1(\mathsf{ID}_a)\tau_1(\mathsf{ID}_b) + \tau_1(\mathsf{ID}_a)\tau_1(\mathsf{ID}_c) + \ldots + \tau_1(\mathsf{ID}_a)\tau_1(\mathsf{ID}_s) & \\ \ldots + \tau_1(\mathsf{ID}_{\Delta-1})\tau_1(\mathsf{ID}_s) & = a_{\Delta-1} \\ \ldots & \\ \tau_1(\mathsf{ID}_a)\tau_1(\mathsf{ID}_b)\tau_1(\mathsf{ID}_c)\ldots\tau_1(\mathsf{ID}_s) & = a_0 \end{cases} \tag{6}$$

Next, the access structure W is interpreted as:

$$\mathsf{W} = (\underbrace{(\mathsf{AND}_{i\in\{1,\ldots,m\}}\mathsf{Att}_i)}_{W_1}\ \mathsf{OR}\ \underbrace{(\mathsf{AND}_{i\in\{1,\ldots,m\}}\mathsf{Att}_i)}_{W_2}\ \mathsf{OR}\ldots\mathsf{OR}\ \underbrace{(\mathsf{AND}_{i\in\{1,\ldots,m\}}\mathsf{Att}_i)}_{W_m}).$$

Then the algorithms computes as follows:

$$\text{Each}\mathsf{W}_i, \text{If}\ \begin{cases} \mathsf{Att}_j \text{ is } + & : r_j = t_i, \\ \mathsf{Att}_j \text{ is } - & : r_j = t_{2_i} \end{cases};$$

Then set $\mathsf{W}_i = \sum_{\mathsf{Att}_j \in \mathsf{W}_i} r_j$.
Next apply the Viète's formula as (2) to computes the whole access structure W:

$$\begin{cases} \mathsf{W}_1 + \mathsf{W}_2 + \ldots + \mathsf{W}_m & = b'_{m-1} \\ \mathsf{W}_1\mathsf{W}_2 + \mathsf{W}_1\mathsf{W}_3 + \ldots + \mathsf{W}_{m-1}\mathsf{W}_m & = b'_{m-2} \\ \ldots & \\ \mathsf{W}_1\mathsf{W}_2\ldots\mathsf{W}_m & = b'_0 \end{cases} \tag{7}$$

Then it produces a vector:

$$\vec{v} = \left(1_0, a_\Delta, a_{\Delta-1}, \ldots, a_0, 1_m, b'_{m-1}, \ldots, b'_0\right)$$

Then it runs $\mathsf{IPE.Enc}(\mathsf{PK}, \vec{v}, \mathsf{M})$, and output the ciphertext CT.
KeyGen(MSK, ID, L):Suppose that a user joins the system with the a given user identity ID and his attribute list L, the algorithm inputs $(\mathsf{ID}, \mathsf{L})$, and transforms them into a vector $\vec{z}$ by generating:
It encodes ID by $\tau_1(ID) = x_{\mathsf{ID}} \in (\mathbb{Z}/p\mathbb{Z})^*$. Then, we compute x_{ID} as the one of the roots of polynomial degree n:

$$\begin{cases} a'_n & = x_{\mathsf{ID}}^n \\ a'_{n-1} & = x_{\mathsf{ID}}^{n-1} \\ a'_{n-2} & = x_{\mathsf{ID}}^{n-2} \\ \ldots & \\ a'_0 & = 1 \end{cases} \tag{8}$$

The algorithm converts an attribute user list L by generating:

$$\text{If}\ \begin{cases} \mathsf{Att}_i \text{ is } + & : r'_i = t_i \\ \mathsf{Att}_i \text{ is } - & : r'_i = t_{2_i} \end{cases} \tag{9}$$

Then set $b = \sum_{\mathsf{Att}_i \in \mathsf{L}} r_i'$, and it computes based on b:

$$\begin{cases} b_m & = b^m \\ b_{m-1} & = b^{m-1} \\ b_{m-2} & = b^{m-2} \\ \cdots & \\ b_0 & = 1 \end{cases}$$

We then produce a vector:

$$\vec{z} = \left(a_n', a_{n-1}', \ldots, 1_n, b_m, \ldots, 1_m\right)$$

Then it runs IPE.KeyGen(PK, $\vec{z}$, MSK), and output the secret key SK.

Decrypt(CT, SK): the algorithm inputs the ciphertext CT and the user's secret key SK, then it runs IPE.Dec(CT, SK) and outputs the message M iff $< \vec{v}, \vec{z} >== 0$. Otherwise, the algorithms the symbol $\perp$.

Correctness: for the vector $\vec{v} = (1_0, a_\Delta, a_{\Delta-1}, \ldots, a_0, 1_m, b_{m-1}', \ldots, b_0')$ corresponding to the set user indices S and access structure W embedded in the ciphertext CT and the vector $\vec{z} = (a_n', a_{n-1}', \ldots, 1_n, b_m, \ldots, 1_m)$ corresponding to the secret key component SK in the ACP-AABBE., we have:

$$\begin{aligned} \sum_{i=0}^{n+m} v_i.x_i &= \sum_{i=0}^{n} v_i.x_i + \sum_{i=n+1}^{n+m} v_i.x_i \\ &= \sum_{i=0}^{n} a_i \cdot x_{\mathsf{ID}}^i + \sum_{i=n+1}^{n+m} b_{i-n}' \cdot b_{i-n} \\ &= \left(1_0 \cdot x_{\mathsf{ID}}^n + a_\Delta \cdot x_{\mathsf{ID}}^{n-1} + \ldots + a_0 \cdot 1_n\right) \\ &\quad + \left(\left(\sum_{\mathsf{Att}_i \in \mathsf{L}} r_i'\right)^m \cdot 1_m + \left(\sum_{\mathsf{Att}_i \in \mathsf{L}} r_i'\right)^{m-1} \cdot b_{m-1}' + \ldots + b_0'\right) \end{aligned}$$

If $\sum_{i=0}^{n+m} v_i.x_i = 0$, the algorithm return M. This means that the ID user is belongs to the set of indices S, and the user attribute list L satisfies the access structures $\mathbb{W}$. Otherwise, the algorithms return $\perp$.

***Constructions of secret keys** We assume $\sum_{att_i \in L} \gamma_1 \neq \sum_{att_i \in L'} \gamma_1$ in both of AKP-ABBE and ACP-ABBE.

If there exist L and L′(L ≠ L′) such that $\sum_{\mathsf{Att}_i \in \mathsf{L}} \gamma_1 = \sum_{\mathsf{Att}_i \in \mathsf{L}'} \gamma_1$, a user with attribute list L can decrypt a ciphertext associated with W, where L′ ⊭ W and L ⊨ W.

Hence, the assumption holds with overwhelming probability:

$$\frac{p(p-1)(p-(N-1))}{p^n} > \frac{(p-N+1)^N}{p^N} = \left(1 - \frac{N-1}{p}\right)^N > 1 - \frac{N(N-1)}{p} > 1 - \frac{N^2}{p},$$

where p is the prime number which chosen in the first step, $N = \prod_{i=1}^{2n} \gamma_i$. If each secret key γ_i is chosen at random from $\mathbb{Z}_p$, then our assumption is natural. Then, the advantage of $\mathcal{A}$ in this game is defined as $\mathbf{Adv} \cdot (1 - \frac{N^2}{p})$.

Theorem 2 *Our ACP-ABBE scheme is secure under the standard assumption if the underlying IPE is secure under the standard assumption.*

4 Security analysis

Our AKP-ABBE and ACP-ABBE utilize the IPE manner to achieve the hidden access structures. Indeed the access structure and the user index set are transformed into the vector. In this part, we choose AKP-ABBE to elaborate the security analysis. Hence, in order to prove that our AKP-ABBE scheme is access structure hiding, we apply the indistinguishability, in which the adversary cannot distinguish two vectors $\vec{v}$ and $\vec{x}$. These two vectors correspond to $(\mathsf{S}_0^*, \mathsf{L}_0^*)$ and $(\mathsf{S}_1^*, \mathsf{L}_1^*)$, respectively, which have been used to generate the two ciphertexts M_0 and M_1.

Based on these above games, we apply the security proof of [19] to our Theorems 1 and 2 directly. To prove the AKP-ABBE be secured in the indistinguishable chosen plaintext attack, we consider two cases $M_0 = \mathsf{M}_1$ and $\mathsf{M}_0 \neq \mathsf{M}_1$:

- $\mathsf{M}_0 = \mathsf{M}_1$, we only consider the following game sequence from **Game**$_1$ to **Game**$_5$. In this case, we prove the property of attribute hiding.
- $\mathsf{M}_0 \neq \mathsf{M}_1$, we consider the whole proof from **Game**$_0$ to **Game**$_6$.

We then present a description of each game, where the challenge ciphertexts $\mathsf{CT}_1, \ldots, \mathsf{CT}_6$ are generated by the IPE's encryption scheme:

– **Game**$_0$: The challenge ciphertext CT_0 is generated under $(\vec{v}, \vec{v})$ and M_0.
– **Game**$_1$: The challenge ciphertext CT_1 is generated under $(\vec{v}, \vec{v})$ and a random message R.
– **Game**$_2$: The challenge ciphertext CT_2 is generated under $(\vec{v}, \vec{0})$ and a random message R.
– **Game**$_3$: The challenge ciphertext CT_3 is generated under $(\vec{v}, \vec{x})$ and a random message R.
– **Game**$_4$: The challenge ciphertext CT_4 is generated under $(\vec{0}, \vec{x})$ and a random message R.
– **Game**$_5$: The challenge ciphertext CT_5 is generated under $(\vec{x}, \vec{x})$ and a random message R.
– **Game**$_6$: The challenge ciphertext CT_6 is generated under $(\vec{x}, \vec{x})$ and message M_1.

PROOF Suppose that the adversary commits to the challenge user indices $\mathsf{S}_0^*, = (\mathsf{ID}_{0a}^*, \mathsf{ID}_{0b}^*, \mathsf{ID}_{0c}^*, \ldots \mathsf{ID}_{0\,s}^*)$ and $\mathsf{S}_1^* = (\mathsf{ID}_{1a}^*, \mathsf{ID}_{1b}^*, \mathsf{ID}_{1c}^*, \ldots, \mathsf{ID}_{1\,s}^*) \subseteq \mathsf{U}$, and the target attribute sets $\mathsf{L}_0^* = (\mathsf{Att}_{01}^*, \ldots, \mathsf{Att}_{0m}^*)$ and $\mathsf{L}_1^* = (\mathsf{Att}_{11}^*, \ldots, \mathsf{Att}_{1m}^*)$ at the beginning of the game.

The $\vec{v}$ is produced of $\mathsf{S}_0^*, = (\mathsf{ID}_{0a}^*, \mathsf{ID}_{0b}^*, \mathsf{ID}_{0c}^*, \ldots \mathsf{ID}_{0\,s}^*)$, and $\mathsf{L}_0^* = (\mathsf{Att}_{01}^*, \ldots, \mathsf{Att}_{0\,m}^*)$ by using (2), (3) from the original construction.

The $\vec{x}$ is produced of $\mathsf{S}_1^*, = (\mathsf{ID}_{1a}^*, \mathsf{ID}_{1b}^*, \mathsf{ID}_{1c}^*, \ldots \mathsf{ID}_{1\,s}^*)$, and $\mathsf{L}_1^* = (\mathsf{Att}_{11}^*, \ldots, \mathsf{Att}_{1\,m}^*)$ by using (2), (3) from the original construction.

We also note that in the query phase the adversary is issued the SK corresponding to the access structure W and the user identity ID. It is also considered that the SK is related to $\vec{y}$, where he $\vec{y}$ is produced of the access structure W and the user identity ID by using (4), (5) from the original construction.

We use the above sequence of hybrid games to prove that the adversary cannot win the original security game with the non-negligible security. We begin with game **Game**$_0$.

Indistinguishability between Game$_0$ **and Game**$_1$ If the adversary obtain the secret key SK corresponding to the access structure W and the user identity ID satisfying such that $(\mathsf{L}_0^* \models \mathsf{W}$

and ($\mathsf{ID} \in \mathsf{S}_0^*$) (meanwhile $< \vec{v}, \vec{y} >= 0$), then the challenge ciphertext is generated correctly. We consider that the challenge ciphertext is distributed in $\mathbf{Game}_0$.

On the other hand, if the adversary obtains the secret key SK with corresponding to the access structure W and the user identity ID where ($\mathsf{L}_0^* \not\models \mathsf{W}$ and ($\mathsf{ID} \notin \mathsf{S}_0^*$) (meanwhile $< \vec{v}, \vec{y} >\neq 0$), then the challenge ciphertext component C_m of IPE scheme is a random element in $\mathbb{G}_T$ regardless of the random choice, while the rest of the challenge ciphertext are generated in an original way. Then we consider that the challenge ciphertext is distributed in $\mathbf{Game}_1$.

Indistinguishability between $\mathbf{Game}_1$ and $\mathbf{Game}_2$

If the adversary obtains the secret key SK with corresponding to the access structure W and the user identity ID where ($\mathsf{L}_0^* \not\models \mathsf{W}$ and ($\mathsf{ID} \notin \mathsf{S}_0^*$), or ($\mathsf{L}_0^* \models \mathsf{W}$ and ($\mathsf{ID} \in \mathsf{S}_0^*$) or ($\mathsf{L}_0^* \not\models \mathsf{W}$ and ($\mathsf{ID} \in \mathsf{S}_0^*$) (meanwhile $< \vec{v}, \vec{y} >\neq 0$), or ($\mathsf{L}_0^* \models \mathsf{W}$ and ($\mathsf{ID} \in \mathsf{S}_0^*$) (meanwhile $< \vec{v}, \vec{y} >= 0$), then the challenge ciphertext is generated correctly. We consider that the challenge ciphertext is distributed in $\mathbf{Game}_1$.

On the other hand, if the adversary obtains the secret key SK with corresponding to the access structure W and the user identity ID by relaxed generation, then the two challenge ciphertext components $C_{3,i}$ and $C_{4,i}$ are the random elements in $\mathbb{G}$ regardless of the random choice, while the rest of the challenge ciphertext is generated in an original way. Then we consider that the challenge ciphertext is distributed in $\mathbf{Game}_2$.

Indistinguishability between $\mathbf{Game}_2$ and $\mathbf{Game}_3$ If the adversary obtain the secret key SK corresponding to the access structure W and the user identity ID satisfying such that ($\mathsf{L}_0^* \models \mathsf{W}$ and ($\mathsf{L}_1^* \models \mathsf{W}$) and ($\mathsf{ID} \in \mathsf{S}_0^*$ and $\mathsf{ID} \in \mathsf{S}_1^*$) (meanwhile $< \vec{v}, \vec{y} >=< \vec{x}, \vec{y} >= 0$), then the challenge ciphertext is generated correctly. We consider that the challenge ciphertext is distributed in $\mathbf{Game}_2$.

On the other hand, if the adversary did not obtain the secret key SK with corresponding to the access structure W and the user identity ID satisfying the constrain of ($\mathsf{L}_0^* \models \mathsf{W}$ and ($\mathsf{L}_1^* \models \mathsf{W}$) and ($\mathsf{ID} \in \mathsf{S}_0^*$ and $\mathsf{ID} \in \mathsf{S}_1^*$) (meanwhile $< \vec{v}, \vec{y} >=< \vec{x}, \vec{y} >\neq 0$), then the two challenge ciphertext components $C_{3,i}$ and $C_{4,i}$ are the random elements in $\mathbb{G}$ regardless of the random choice, while the rest of the challenge ciphertext are generated in a original way. Then we consider that the challenge ciphertext is distributed in $\mathbf{Game}_3$.

Due to the symmetric observation, the rest of the proof is similar to the above proofs:

- the indistinguishability between $\mathbf{Game}_3$ and $\mathbf{Game}_4$ can be proved in the same way as for $\mathbf{Game}_2$ and $\mathbf{Game}_3$;
- the indistinguishability between $\mathbf{Game}_4$ and $\mathbf{Game}_5$ can be proved in the same way as for $\mathbf{Game}_1$ and $\mathbf{Game}_2$;
- the indistinguishability of $\mathbf{Game}_5$ and $\mathbf{Game}_6$ can be proved in the same way as for $\mathbf{Game}_0$ and $\mathbf{Game}_1$.

The ACP-ABBE is proved secure under standard assumption by the similar arguments of AKP-ABBE, where $\vec{v}$ is produced of $\mathsf{S}_0^*, = (\mathsf{ID}_{0a}^*, \mathsf{ID}_{0b}^*, \mathsf{ID}_{0c}^*, \ldots \mathsf{ID}_{0s}^*)$, and $\mathsf{W}_0^* = (\mathsf{W}_{01}^*, \ldots, \mathsf{W}_{0m}^*)$ by using (4), (5) from the original construction. In addition, the $\vec{x}$ is produced of $\mathsf{S}_1^*, = (\mathsf{ID}_{1a}^*, \mathsf{ID}_{1b}^*, \mathsf{ID}_{1c}^*, \ldots \mathsf{ID}_{1s}^*)$, and $\mathsf{W}_1^* = (\mathsf{W}_{11}^*, \ldots, \mathsf{W}_{1m}^*)$ by using (6), (7) from the original construction. It is also considered that the SK is related to $\vec{y}$, where he $\vec{y}$ is produced of the access structure W and the user identity ID by using (8), (9) from the original construction.

Our proposal utilizes the work of [26] as a building block to construct the AKP-ABBE and ACP-ABBE schemes. Inherently, the strategy of our security proof is also argued as [19, 26], in which we directly apply the DBDH and DLIN assumption as [26] to prove our AKP-ABBE and ACP-ABBE be secured in the standard assumption. Therefore, by underlying the

Table 4 Comparisons in AKP-ABBE

KP-ABBE	Ciphertext	Decryption	Access structure	Attributes	Complexity assumption
[2]	$O(k_{max}+r)$	$O(k_{max}+r)$	LSSS	+ Attributes	q-BDHE
[30]	$O(m+2)$	$O(3)$	AND gates	± w/ wildcard	q-BDHE
AKP-ABBE	$O(n+m)$	$O(n+m)$	OR/AND gates	± Attributes w/ wildcard	DBDH, D-linear

secured IPE of [26] under the standard assumption, we conclude that our AKP-ABBE and ACP-ABBE schemes are secure under the standard assumption.

5 Extensions

We extend how our proposed scheme can also achieve the Anonymous ABBE, which access structure supports AND Gates with positive, negative, wildcard [means "don't care" (i.e., both positive and negative attributes are accepted)]:

Firstly, we choose the suitable encoding τ_2 sending each of the m attributes $\mathsf{Att} \in \mathsf{U}$ onto an element $\tau_2(\mathsf{Att}) = x_2 \in (\mathbb{Z}/\mathrm{p}\mathbb{Z})^*$

$$\text{If } \begin{cases} \mathsf{Att}_i \text{ is } + & : b_{2i-1} = \tau_2(\mathsf{Att}_i), b_{2i} = -1 \\ \mathsf{Att}_i \text{ is } - & : b_{2i-1} = -\tau_2(\mathsf{Att}_i), b_{2i} = -1 \\ \mathsf{Att}_i \text{ is } * & : b_{2i-1} = 0, b_{2i} = 0 \end{cases}; \tag{10}$$

then we generate the $\vec{v}$ as:

$$\vec{v} = (b_1, , \ldots, b_m)$$

For an attribute user list L, it computes:

$$\text{If } \begin{cases} att_i \text{ is } + & : b'_{2i-1} = 1; b'_{2i} = \tau_2(att_i) \\ att_i \text{ is } - & : b'_{2i-1} = 1; b'_{2i} = -\tau_2(att_i) \end{cases}; \tag{11}$$

, then we generate the $\vec{z}$ as:

$$\vec{v} = (b'_1, , \ldots, b'_m)$$

If $< \vec{v}, \vec{z} >= 0$, we conclude that $\mathsf{L} \models \mathsf{W}$.

6 Comparisons

In this section, we give a comparison among ABBE schemes in Tables 5 and 4. The schemes are compared in terms of the order of the underlying group, ciphertext size, decryption cost, access structure, and complexity assumption. In the table, N—number of clauses in a policy, M—maximum number of attributes in the given clause, k-number of attributes for a given user, r-number of revoked users, k_{max}—maximum number of attributes in access structure, n-total of the user's identity, m—number of universe attributes.

As can be seen in Tables 4 and 5, our encryption and decryption are linear depending on the size of the user's indices and the size of the access structure. In fact, both two proposed

Table 5 Comparisons in ACP-ABBE

CP-ABBE	Ciphertext	Decryption	Access structure	Attributes	Complexity assumption
[22]	$O(N \cdot R)$	$O(N.k)$	AND gates	± Attributes	Generic group model
[2]	$O(k_{max} + r)$	$O(k_{max} + r)$	LSSS	+ Attributes	q-BDHE
[18]	$O(N \cdot M)$	$O(M + m)$	OR/AND gates	+ Attributes	GDHE
[30]	$O(1)$	$O(3)$	AND gates	± w/ wildcard	q-BDHE
[10]	$O(N)$	$O(2)$	AND gates	+ Attributes	GDDHE
[34]	$O(N)$	$O(3)$	LSSS	+ Attributes	n q-MEBDH
ACP-ABBE	$O(n + m)$	$O(n + m)$	OR/AND gates	± Attributes w/ wildcard	DBDH, D-linear

schemes implement the IPE's encryption to produce the ciphertext, and invoke the IPE's decryption to recover the message. In addition, the cost of IPE scheme relies on the input of vectors. Our access structures are designed with flexibility by employing both AND/OR gates with negative/positive attributes and wildcards. This idea is well-suitable in practice, where the architecture of access control always requires multiple authorizations. In terms of security proof, both KP-ABBE and CP-ABBE can be proved in the standard assumptions, such as DBDH and DLinear assumptions. *Specifically, we highlight that our proposed ABBEs can achieve anonymity due to the inherence of attribute hiding from the IPE scheme*. Therefore, our ABBE schemes achieve anonymity with multiple access structures.

7 Conclusion

This paper proposes two new constructions of Anonymous Attribute-Based Broadcast Encryption as AKP-ABBE and ACP-ABBE for complex access structure by considering the OR/AND Gates with positive, and negative attributes. We present our proposed schemes in generic constructions, achieving anonymity. We also proved the security of our schemes be secured in the standard model. One open problem is to construct our AKP/ACP-ABBE schemes that have constant ciphertext and secret key, and we leave it as our future work.

Acknowledgements This work is supported by the Commonwealth Cyber Initiative (CCI).

References

1. Abdalla M., Bourse F., Caro A., Pointcheval D.: Public-Key Cryptography—PKC 2015, pp. 733–751. Springer, Berlin, Heidelberg (2015).
2. Attrapadung N., Imai H.: Conjunctive broadcast and attribute-based encryption. In: Pairing-Based Cryptography, pp. 248–265 (2009).
3. Attrapadung N., Libert B.: Functional encryption for inner product: achieving constant-size ciphertexts with adaptive security or support for negation. In: Public Key Cryptography—PKC 2010, pp. 384–402. Springer, Berlin, Heidelberg (2010).
4. Attrapadung N., Libert B., Panafieu E.: Expressive key-policy attribute-based encryption with constant-size ciphertexts. In: Public Key Cryptography—PKC 2011, Volume 6571 of Lecture Notes in Computer Science, pp. 90–108 (2011).
5. Berkovits S.: How to broadcast a secret. In: EUROCRYPT, pp. 535–541 (1991).
6. Bethencourt J., Sahai A., Waters B.: Ciphertext-policy attribute-based encryption. In: Security and Privacy, 2007. SP '07. IEEE Symposium on, pp. 321–334 (2007).
7. Boneh D., Gentry C., Waters B.: Collusion resistant broadcast encryption with short ciphertexts and private keys. In: CRYPTO, pp. 258–275 (2005).
8. Boneh D., Waters B.: A fully collusion resistant broadcast, trace, and revoke system. In: ACM CCS, pp. 211–220 (2006).
9. Boneh D., Waters B.: Conjunctive, subset, and range queries on encrypted data. In: Proceedings of the 4th Conference on Theory of Cryptography, TCC'07, pp. 535–554. Springer-Verlag (2007).
10. Canard S., Phan D.-H., Trinh V.C.: Attribute-based broadcast encryption scheme for lightweight devices. IET Inf. Secur. **12**(1), 52–59 (2018).
11. Cheung L., Newport C.: Provably secure ciphertext policy ABE. In: Proceedings of the 14th ACM Conference on Computer and Communications Security, CCS '07, pp. 456–465. New York, NY, USA (2007).
12. Cioni S., Lin X., Chamaillard B., El Jaafari M., Charbit G., Raschkowski L.: Physical layer enhancements in 5G-NR for direct access via satellite systems. Int. J. Satell. Commun. Netw. **41**(3), 262–275 (2022).
13. Delerablée C., Paillier P., Pointcheval D.: Fully collusion secure dynamic broadcast encryption with constant-size ciphertexts or decryption keys. In: Pairing-Based Cryptography, pp. 39–59 (2007).
14. Fiat A., Naor M.: Broadcast encryption. In: CRYPTO, pp. 480–491 (1993).
15. Goodrich M.T., Sun J.Z., Tamassia R.: Efficient tree-based revocation in groups of low-state devices. In: CRYPTO, pp. 511–527 (2004).
16. Goyal V., Pandey O., Sahai A., Waters B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS '06, pp. 89–98. ACM (2006).
17. Halevy D., Shamir A.: The LSD broadcast encryption scheme. In: CRYPTO, pp. 47–60 (2002).
18. Junod P., Karlov A.: An efficient public-key attribute-based broadcast encryption scheme allowing arbitrary access policies. In: ACM Workshop on Digital Rights Management, pp. 13–24 (2010).
19. Katz J., Sahai A., Waters B.: Predicate encryption supporting disjunctions, polynomial equations, and inner products. In: Proceedings of the Theory and Applications of Cryptographic Techniques 27th Annual International Conference on Advances in Cryptology, EUROCRYPT'08, pp. 146–162 (2008).
20. Lai, J., Deng, R.H., Li, Y.: Fully secure cipertext-policy hiding CP-ABE. In: Proceedings of the 7th International Conference on Information Security Practice and Experience, ISPEC'11, pp. 24–39. Springer-Verlag.
21. Lewko, A.B., Waters, B.: New proof methods for attribute-based encryption: achieving full security through selective techniques. In: CRYPTO, pp. 180–198 (2012)
22. Lubicz, D., Sirvent, T.: Attribute-based broadcast encryption scheme made efficient. In: AFRICACRYPT, pp. 325–342 (2008)
23. Okamoto, T., Takashima, K.: Adaptively attribute-hiding (hierarchical) inner product encryption. In: Advances in Cryptology—EUROCRYPT 2012, Volume 7237 of Lecture Notes in Computer Science, pp. 591–608 (2012)
24. Ouaddah A., Mousannif H., Abou Elkalam A., Ouahman A.A.: Access control in the internet of things: big challenges and new opportunities. Comput. Netw. **112**, 237–262 (2017).
25. Park J.H.: Efficient hidden vector encryption for conjunctive queries on encrypted data. IEEE Trans. Knowl. Data Eng. **23**, 1483–1497 (2011).
26. Park J.H.: Inner-product encryption under standard assumptions. Des. Codes Cryptogr. **58**(3), 235–257 (2011).
27. Ruzakova, O.A.: Digital platforms and media-regulatory framework. In: The Platform Economy, pp. 203–214. Springer (2022)

28. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: Proceedings of the 24th Annual International Conference on Theory and Applications of Cryptographic Techniques, EUROCRYPT'05, pp. 457–473. Springer-Verlag (2005)
29. Shi, E., Waters, B.: Delegating capabilities in predicate encryption systems. In: Proceedings of the 35th International Colloquium on Automata, Languages and Programming, Part II, ICALP '08, pp. 560–578 (2008)
30. Phuong Tran, V.X., Yang, G., Susilo, W., Chen, X.: Attribute based broadcast encryption with short ciphertext and decryption key. In: Computer Security—ESORICS 2015, Volume 9327 of Lecture Notes in Computer Science, pp. 252–269. Springer International Publishing (2015)
31. Viète F.: Opera mathematica. Bonaventura et Abr, Elzevir (1970).
32. Waters, B.: Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization. In: Public Key Cryptography, pp. 53–70 (2011)
33. Wesolowski, B., Junod, P.: Ciphertext-policy attribute-based broadcast encryption with small keys. In: ICISC 2015, pp. 53–68. Springer (2015)
34. Xiong H., Zhao Y., Peng L., Zhang H., Yeh K.-H.: Partially policy-hidden attribute-based broadcast encryption with secure delegation in edge computing. Future Gener. Comput. Syst. **97**, 453–461 (2019).
35. Zhang X., Zhong H., Cui J., Gu C., Bolodurina I., Liu L.: AC-SDVN: an access control protocol for video multicast in software defined vehicular networks. IEEE Trans. Mob. Comput. (2022). https://doi.org/10.1109/TMC.2022.3180809.
36. Zhou Z., Huang D., Wang Z.: Efficient privacy-preserving ciphertext-policy attribute based-encryption and broadcast encryption. IEEE Trans. Comput. **64**(1), 126–138 (2013).