IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 25, 2020, accepted January 12, 2021, date of publication January 18, 2021, date of current version January 27, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3052463

A Secure and Policy-Controlled Signature Scheme With Strong Expressiveness and Privacy-Preserving Policy

XURUI ZHENG[1], FUYONG ZHENG[2], XIANMING LIU[2], DEJUN WANG[1], JUN WANG[1], AND BO MENG[1]

[1]School of Computer Science, South-Central University for Nationalities, Wuhan 430074, China
[2]Information and Communication Branch, Jiangxi Electric Power Company Ltd., Nanchang 330077, China

Corresponding author: Bo Meng (mengscuec@gmail.com)

This work was supported in part by the Fundamental Research Funds for the Central Universities, South-Central University for Nationalities under Grant CZT20013, Grant CZT20015, and Grant QSZ17007; in part by the National Natural Science Foundation of China under Grant 62062019; and in part by the Natural Science Foundation of Hubei Province under Grant 2019CFB815 and Grant 2018ADC150.

ABSTRACT The policy-controlled signature (PCS) scheme uses the access policy to control signature verification permission. However public access policy that may contain private information will leak user privacy. At the same time, the expressiveness of access structures in the PCS schemes is weak. Therefore, we propose a policy-controlled signature scheme with strong expressiveness and privacy-preserving policy (PCS-PP), in which linear secret sharing schemes is to design access structure which has strong expression, the three primes composite order bilinear groups is used to hide the attribute value into the attribute name that may expose the privacy data by data distortion concept. The proposed PCS-PP scheme not only has correctness and privacy-preserving policy, but also supports fine-grained signature verification. In addition, the unforgeability is proved in the random oracle model. Compared to the related schemes, the proposed PCS-PP scheme has superiority in features, computation cost and storage.

INDEX TERMS Policy-controlled signature, privacy-preserving, linear secret sharing scheme, composite order bilinear groups.

I. INTRODUCTION

The policy-controlled signature (PCS) scheme is a key part of digital signatures and supports access control of signature verifier. The verifier can verify the signature with the condition that the attributes of the verifier satisfy the access policy. Consider the following scenario, the electronic medical records of the patients can only be used by the authorized doctor. A PCS scheme can implement the access control for electronic medical records of the patients. While access policy in the PCS maybe has the private information of the authorized doctor, for example, home address. Owning that the access policy is public, the privacy information of the authorized doctor may be leaked.

According to our analysis, the PCS schemes [1]–[3] do not take account of the privacy-preserving access policy, nor do they also provide strong expressiveness of access structures. The privacy preservation of access policy depends on the access structure and the privacy-preserving method. Privacy- preserving method is closely related to the specific access structure. The access structure determines the expressiveness of the access policy. Access structures mainly include monotonous AND gate access structures [4]–[6], access structures that support threshold gate [7]–[9], access structures based on bounded access tree [10]–[12], and linear secret sharing schemes (LSSS) matrix [13], [14]. Among them, the access policy expressiveness of the first three access structures is relatively weak, the last one has strong expressiveness. Privacy-preserving methods mainly include data distortion [15], data encryption [16], and data anonymity [17]. Functional encryption [18], [19] is one of data encryption for privacy preservation in attribute-based encryption (ABE). Dual-system encryption and hidden vector encryption [20] are used to provide privacy preservation in ABE schemes with full hidden policy, which lose the expressiveness of access policy. However, data distortion

The associate editor coordinating the review of this manuscript and approving it for publication was Sathish Kumar.

and dual-system encryption can take advantage of partially hidden policy with privacy preservation while remaining expressiveness which is more appropriate for LSSS.

Therefore, this paper employs LSSS to design a strong expressiveness of access structure and borrows the thought of data distortion to propose a policy-controlled signature scheme with strong expressiveness and privacy-preserving policy (PCS-PP) scheme. The main works are presented as follows:

1) Use an LSSS matrix to develop the access structure that has strong expressiveness; apply three primes composite order bilinear groups [21] based on data distortion to expose the public attribute name and hide the attribute value, and then present a PCS-PP scheme.
2) Formalize and define the security model and general requirements for the proposed PCS-PP scheme; analyze the proposed PCS-PP scheme and the results show that it supports correctness, unforgeability, privacy-preserving policy, and fine-grained signature verification. To the best of our knowledge, the proposed PCS-PP scheme is the first construction that achieves these properties.
3) Compared to the related signature schemes, the proposed PCS-PP scheme has superiority in features, computation cost, and storage.

The rest of the paper is organized as follows. Section II discusses the related works of the privacy-preserving scheme and the access structures based on the policy signature. Section III makes a simple review of preliminaries; Section IV defines the PCS-PP security model. Section V presents the PCS-PP scheme and its correctness proof; Section VI applies the security model to analyze its security; Section VII analyses expressiveness of access structures in the PCS-PP scheme; Section VIII presents analysis and comparison of the performance and features. Finally, Section IX presents the conclusion and future works.

II. RELATED WORKS

The proposed PCS-PP mainly involves the privacy-preserving scheme and the access structures based on the policy signature. Therefore, in this section, we review related works including policy-based signature (PBS), PCS, and access policy hidden structures in ciphertext-policy ABE (CP-ABE).

Mihir *et al*. [22] proposed a PBS scheme, which is similar to the authentication of functional encryption and attribute-based signature (ABS) [23]. In PBS, limiting the signer's signature permission and not restricting the verifier's permission is more secure in privacy-preserving than ABS. PCS was proposed by Pairat *et al*. [1], [2] in 2017. A signer can sign a message and attach it with some policies. Only a verifier who satisfies the policies attached can verify the authenticity of the message. The difference between the PCS and the designated-verifier signature (DVS) [24] is that the PCS can be applied to allow multiple verifiers that conform to the access policy to verify the signature, while the DVS allows only one verifier to verify the signature. But the disclosure of access policy may have privacy risks. Liu *et al*. [3] proposed the first quantum-resistant scheme based on the PCS. But it does not involve the privacy of information, and a strong expressive access-policy cannot be implemented using lattice.

ABE was developed from identity-based encryption (IBE) proposed by Shamir *et al*. [25]. Goyal *et al*. [26] proposed key-policy ABE (KP-ABE) and apply the policy to decrypt the ciphertext, which introduced an access structure that supports bounded access tree. Beth *et al*. [27] proposed CP-ABE, which combines access policy with ciphertext and can only be decrypted when the user satisfies the policy. Access structure is the key technique of access policy hidden in attribute encryption. Ostrovsky *et al*. [28] proposed ABE scheme with an access structure that supports AND, OR, and NOT gates. However, the policy expression of these schemes is relatively weak. Water *et al*. [29] proposed CP-ABE that applied the LSSS to express access policy which has strong expression.

However, there is a drawback preventing ABE from being applied to data sharing like cloud applications. In ABE, the access policy is public type, which is likely to contain private information, so they need to be hidden. For the policy hidden method in the monotonous AND gate access structure, Lai *et al*. [30] used dual-system encryption technology to achieve secure access policy hidden. Yadav *et al*. [31] calculated the corresponding user private key and ciphertext by giving each attribute in the access structure three states, and then achieve a completely hidden access policy. Yang *et al*. [32] used three different group elements to represent the attributes of three possible values to achieve a completely hidden access policy on a Tree-Based CP-ABE. Song *et al*. [33] proposed a hidden policy CP-ABE based on the tree structure. Under the decisional bilinear Diffe-Hellman (DBDH) assumption of the standard model, they proved that it resists the plaintext attack. Huang *et al*. [34] proposed Privacy-Preserving Constant CP-ABE (PP-CP-ABE) which is more flexible than Broadcast Encryption (BE). But it only supports a conjunctive access policy. Xiong *et al*. [35] proposed a CP attribute Broadcast encryption scheme (CP-AB-BE) that introduced the concept of data distortion to protect private information in an access policy. It includes partially hidden policy, direct revocation, and verifiable outsourced decryption. However, it is challenging to provide a more expressive access policy. Khan *et al*. [36] proposed the hidden CP-ABE based on the LSSS access structure, which does not expose access policy, but uses hidden vector encryption for subset conditional query, which is inefficient for decryption. Chen *et al*. [37] used the three primes composite order bilinear group to construct the CP-ABE scheme, which not only realized policy hidden based on the LSSS matrix, but also improved efficiency.

III. PRELIMINARIES

A. COMPOSITE ORDER BILINEAR GROUPS

Composite order bilinear groups were first introduced by Boneh *et al.* [38]. The order of bilinear groups we used is the product of three distinct primes.

Let G and G_T are cyclic multiplicative groups with the same order $N = pqr$, where p, q and r are different prime Numbers. G_p, G_q and G_r subgroups of group G with orders p, q and r. Then g_p, g_q and g_r are generators of G_p, G_q and G_r respectively. Let $e : G \times G \to G_T$ presents three primes composite order bilinear groups mapping:

1) Bilinearity: For all $g_1, g_2 \in G$ and $a, b \in Z_p$. $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$.
2) Nondegenerate: There exist $g \in G$ such that the order of $e(g, g)$ in G_T is N.
3) Computability: There exist an efficient algorithm to compute $e(g_1, g_2)$ for all $g_1, g_2 \in G$.
4) Orthogonality: $e(g_p, g_q) = 1$ for any $g_p \in G_p$ and any $g_q \in G_q$.

B. POLICY-CONTROLLED SIGNATURE

A signer can sign a message and attach it with some policies in PCS. Only a verifier who meets the attached policies can verify the validity of the signature. This type of signature schemes has many applications, especially dealing with sensitive data, where the signer does not want to allow anyone who is unauthorized to verify the authenticity of the messages.

PCS is composed of six phases: system parameters generation, trust Authority (TA) key generation, signer key generation, verifier credential generation, PCS signature generation and PCS signature verification.

C. ACCESS STRUCTURE

In the proposed PCS-PP scheme, attributes are replaced with the participant p_i and the access structure A [39] designed by us contains the set of authorized attributes.

Let $\{p_1, p_2, \cdots, p_n\}$ represent a set of parties of n participants, a collection $A \subseteq 2^{\{P_1, P_2, \cdots, P_n\}}$ is monotone if $\forall B, C$: if $B \in A$ and $B \subseteq C$ then $C \in A$. An access structure is a collection A of non-empty subsets $\{p_1, p_2, \cdots p_n\}$, i.e. $A \subseteq 2^{\{P_1, P_2, \cdots, P_n\}}/\{\emptyset\}$. The sets in A are called the authorized sets, and the sets not in A are called the unauthorized sets.

D. LINEAR SECRET SHARING SCHEME

LSSS matrix [40] can express any monotonic access structure. A LSSS $\prod$ over a set of parties is called linear over Z_P if the following two conditions are satisfied:

1) Shares combination from all parties form a vector over Z_P.
2) A Share-generating matrix M for $\prod$ has i rows and n columns. For all $i = \{1, 2, \cdots, l\}$, the i_{th} row of M, we define a mapping $p(i)$ as party labeling row i. Let a column vector $v = \{s, v_2, \cdots, v_n\}$ be a sharing vector where $s \in Z_P$ is the secret to be shared, and $v_2, \cdots, v_n$ are chosen at random from Z_P.Then M_v is the vector of i share of S according to $\prod$. Each share $\lambda_i = (M^*\vec{v})_i$ belongs to entity $p(i)$.

Each LSSS obtained by the above definition has the attribute of linear reconstruction. We assume that $\prod$ is a LSSS corresponding to access structure A. Define $I = \{i : p(i) \in s\} \subset \{2, \cdots, l\}$ for any authorization set $S \in A$. For linear reconstruction, we have constants of the form $\{w_i \in Z_P\}_{i \in I}$. Such that, if $\{\lambda_i\}$ are valid shares of secret s according to $\prod$, then s can be reconstructed by $\sum_{i \in I} w_i \lambda_i = s$. For any unauthorized set, there is the vector $w \in Z_P$ such that $w^*(1, 0, \cdots, 0)^T = -1, w^* M_i = 0, i \in I$.

E. COMPLEXITY ASSUMPTION

Computational Diffe-Hellman (CDH) assumption and subgroup decision problem for 3 primes (SDP) assumption are used to analyze the security of the proposed PCS-PP scheme.

1) CDH ASSUMPTION

The definition of CDH problem as follows: Choose a cyclic multiplication group G of prime order p according to the security parameter. For $a, b \in Z$, given 3-tuple (g, g^a, g^b) as input, outputs $DH_{gp}(g^a, g^b) = g^{ab}$. An algorithm A has an advantage ε in solving the CDH problem in G, if $\Pr[A(g, g^a, g^b) = g^{ab}] \geq \varepsilon$ where the probability is related to the random selection of the generator $g \in G$; $a, b \in Z_P$ and the random number determined by A.

Definition 1 (CDH Assumption): We say that the (t, ε) -CDH assumption holds if no PPT algorithm with time complexity t(·) has an advantage at least in solving the CDH problem.

2) SDP ASSUMPTION

The SDP assumption contains the Assumption 1, Assumption 2, and Assumption 3. Pr is the probability function.

Assumption 1: Given a double line group generator Φ, and define $G_\Phi = (Q = pqr, G, G_T, e) \leftarrow \Phi$, $g_p \leftarrow G_P$, $g_q \leftarrow G_q$, $D = (G, g_p, g_q)$, $T_1 \leftarrow G_{pr}$, $T_2 \leftarrow G_p$.We define the probability that algorithm Ψ breaks assumption 1:

$$Adv1_{\Phi,\psi(\lambda)} = |\Pr[\Psi(D, T_1) = 1] - \Pr[\Psi(D, T_2) = 1]|$$

Assumption 2: Given a double line group generator Φ, and defines $G_\Phi = (Q = pqr, G, G_T, e) \leftarrow \Phi, (g_p, Z_1) \leftarrow G_P$, $Z_2 \leftarrow G_r$, $g_q \leftarrow G_q$, $D = (G, g_p, g_q, Z_1, Z_2)$, $T_1 \leftarrow G_{pr}$, $T_2 \leftarrow G_p$.We define the probability that algorithm Ψ breaks assumption 2:

$$Adv2_{\Phi,\psi(\lambda)} = |\Pr[\Psi(D, T_1) = 1] - \Pr[\Psi(D, T_2) = 1]|$$

Assumption 3: Given a double line group generator Φ, and defines $G_\Phi = (Q = pqr, G, G_T, e) \leftarrow \Phi, (\alpha, s) \leftarrow Z_N, g_p \leftarrow G_P, X \leftarrow G_q, (X_2, Y, Z_2) \leftarrow G_r, D = (G, g_p, g_P^\alpha X_2, g_r, g_p^s Y, Z_2), T_1 = e(g_p, g_p)^{\alpha s}, T_2 \leftarrow G_p$. We define the probability that algorithm Ψ breaks assumption 3:

$$Adv3_{\Phi,\psi(\lambda)} = |\Pr[\Psi(D, T_1) = 1] - \Pr[\Psi(D, T_2) = 1]|$$

Definition 2 (SDP Assumption): If the probability that algorithm Ψ breaks assumption 1, 2, 3 is no-negligible, then we can come to the conclusion that G_Φ satisfies the assumption.

IV. SECURITY MODEL

In this section, the security models composed of unforgeability property and privacy property are presented for the proposed PCS-PP scheme.

A. UNFORGEABILITY PROPERTY

The unforgeability property of the proposed PCS-PP scheme means that an attacker $\boldsymbol{A}$ accessing the verifier's credential cannot generates a forgery policy-controlled signature δ^* on a new message M^*. So, the model was named a security against existential unforgeability under adaptive chosen message and credentials exposure attack (EUF-CMCEA) [2]. EUF-CMCEA model provides an assurance that, with accessing to the signing oracle (SO), the credential generation oracle (CRO), the hash oracle (HO) and PK_{TA}, no one should be able to forge a PCS-PP signature on a new message M^* even if it arbitrarily chooses policy POL, message M and entire credential Cre_V as input. Let $\boldsymbol{F}$ be a simulator. To describe the ability of breaking the unforgeability for adversaries, SO and CRO oracles are illustrated as follows.

1) OS: This oracle can be called at most q_S times, we express this repetition through foreach $i_s < q_s$ do SO, which means that the provided q_S here is the copies of SO, each one with a different index value [1, q_S]. In addition, the variables defined in the repeated oracle are arrays, and each one called to the oracle has a cell so that we can remember the values used in all calls to the oracle. In this case, m is an array indexed by i_S. Similarly, the copy of the oracle SO itself i_S indexed by i_S, so that the caller can specify which copy of the operating system he wants to call by calling SO [i_S].
The variables in repeated oracles are arrays, with one cell for each call, to remember the values used in each oracle call. These arrays are indexed with the call number i_S.
$\boldsymbol{A}$ can make q_S queries for a signature δ on its choice of a message m. And then, SO responses with δ to A. SO updates a query to the SO [i_S]. So, we formalize a queried signature in Fig 1.

$$\text{foreach } i_s \le q_s \text{ do } SO[i_s](M[i_s]:bitstring) :=$$
$$\text{return}(sign(sk, M[i_s], pk_s, pk_{TA}, POL, param))$$

FIGURE 1. Signature query.

2) CRO: s is an attribute of the verifier; this oracle can be called at most q_g times. At most q_g times, $\boldsymbol{A}$ can make at most q_g queries for credential Cre_v corresponding to the arbitrarily chosen attributes. CRO responses with Cre_v to $\boldsymbol{A}$. CRO updates a query to the CRO [i_S]. And then, we formalize the queried credential in Fig 2.

$$\text{foreach } i_s \le q_g \text{ do } CRO(s:bitstring) :=$$
$$return(CreGen(param, pk_{TA}, s))$$

FIGURE 2. Credential query.

$$\begin{aligned}
&\text{Expt}^{A_{EUF-CMCEA}}(k):\\
&\quad \text{param} \leftarrow Setup(1^k)\\
&\quad st \leftarrow A_1^{SO,CRO,HO}(\text{param})\\
&\quad \delta^* \leftarrow A_2(M^*, st, PK_{TA}, POL)\\
&\quad \text{return } \delta^*\\
&A \text{ win the above game if}\\
&\quad verify(M^*, PK_{TA}, Cre, \delta^*) \text{ then}\\
&\quad \text{find } u \le q_s \text{ suchthat } (\text{defined}(M[u]) \wedge M^* = M[u])\\
&\quad \text{then end else event } forge)
\end{aligned}$$

FIGURE 3. The existential unforgeability game.

At last, we denote by $\boldsymbol{A}_{EUF\text{-}CMCEA}$ the adaptive chosen message and credentials exposure adversary and let $\boldsymbol{F}$ be a simulator. Let st be the state of information that $\boldsymbol{A}$ obtains during in the learning state. we define the game between $\boldsymbol{F}$ and $\boldsymbol{A}$ is defined to describe the existential unforgeability of PCS-PP scheme in the Fig 3.

Definition 3 (Unforgeability): The PCS-PP scheme is $(Q_H, Q_S, Q_C, \varepsilon)$-secure existential unforgeability under a chosen message and credential exposure attack if there are no PPT adversary $\boldsymbol{A}_{EUF\text{-}CMCEA}$ with a non-negligible probability $Succ_{EUF\text{-}CMEA}(k) = \varepsilon$ in k, where $\boldsymbol{A}_{EUF\text{-}CMCEA}$ runs in time at most t, make at most Q_H queries to the hash random oracle, and at most Q_S, and Q_C queries to the SO and CRO respectively.

B. PRIVACY PROPERTY

The privacy-preserving policy is implemented by the confidentiality of attribute code, which is constructed for each attribute value using a three primes composite order bilinear groups. We will give the detail about attribute code in section V and section VI. So, if the attribute code with confidentiality, the attacker cannot get the verifier of private information. The privacy in proposed PCS-PP scheme is defined using Selective Plaintext Attack Game (IND-CPA), which is a simulation between a challenger $\boldsymbol{C}$ and an adversary $\boldsymbol{A}$. In the game, the $\boldsymbol{C}$ simulates an execution environment of algorithms to answer the adversary's query request. The steps to define game process are described as follows:

1) **Setup:** Frist, $\boldsymbol{C}$ runs setup, which outputs SK_{TA} and PK_{TA}. And then sends PK_{TA} to $\boldsymbol{A}$.
2) **Phase 1:** At the end of the first phase, $\boldsymbol{A}$ decides to challenge and inputs POL^* and M^*. Attacker never issued a request for a PCS-PP signature ***SO*** queries.

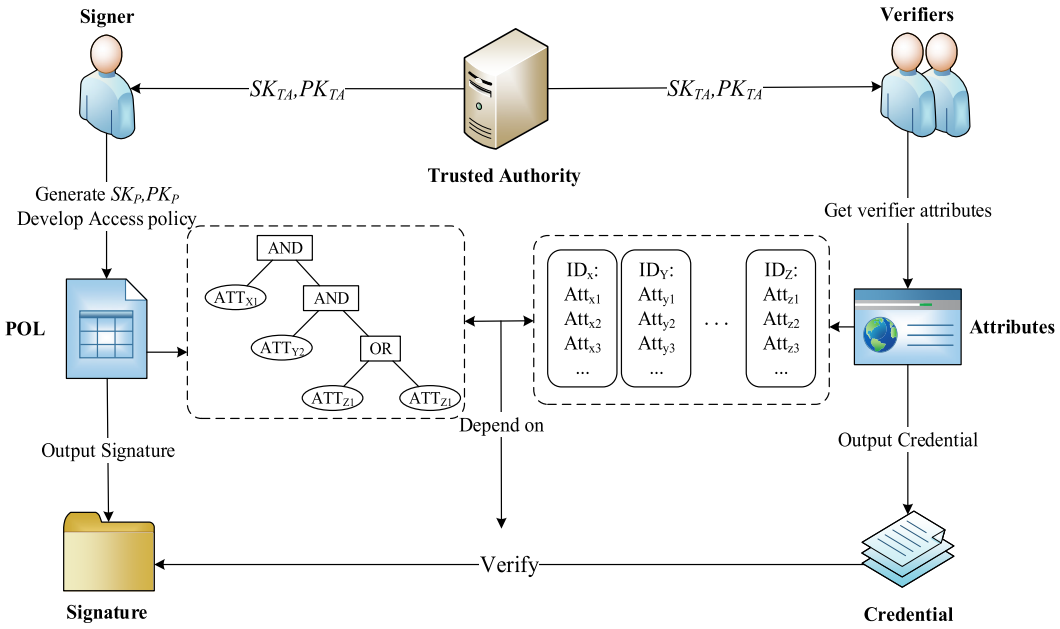


FIGURE 4. Framework of PCS-PP scheme.

C runs the Key and credential generation algorithm to get *SKp* and returns the ***A***.

3) **Challenge:** ***A*** submits two plaintext messages of the same length S_0 and S_1, and two access structures T_0 and T_1 to ***C***. And then after tossing a fair coin $b \in \{0, 1\}$, ***C*** encrypts the message S_b using the access structure T_b, and sends the ciphertext *CT* to ***A***.
4) **Phase 2:** Repeat phase 1.
5) **Guessing:** On the challenge M^*, POL^*, ***A*** outputs a guess b'. The distinguisher wins the game if $b = b'$. Let $|\Pr[b' = b] - \frac{1}{2}|$ be the success probability function of that attacker *A* wins the above game.

Definition 4 (Privacy): The proposed PCS-PP scheme has privacy under chosen message attacker ***A*** if there are no PPT distinguishers such that the success probability is non-negligible.

V. THE PROPOSED PCS-PP SCHEME

In this section, the proposed PCS-PP scheme in detail is presented. The proposed PCS-PP scheme uses a LSSS matrix to develop access structure which has strong expressiveness and applies three primes composite order bilinear groups based on data distortion to expose the public attribute name and hide the attribute value to implement the privacy-preserving policy.

In order to have a general view of the proposed PCS-PP scheme, we first review the proposed PCS-PP scheme and give the scheme framework in Fig. 4. There are three entities in the proposed PCS-PP scheme: signer, verifier, and TA.

TA is a trusted authority in the system. It is responsible for generating and distributing system parameters. Meanwhile, TA generates authenticated public keys and verifier certificates for each authorized verifier. The signer generates the signing private key and public key by using public parameters, and the signer can generate a signature on the message by attaching a defined access policy POL. The signature generated by the signer can only be verified by the verifier who owns the attributes in certificate issued by TA that satisfies the access policy in the signature.

A. THE OVERVIEW

The proposed PCS-PP scheme consists of the four phases: Setup, Key and Credential generation, Signature generation and Signature Verification. The first three are randomized.

1) **Setup:** System Parameters Generation (Setup): Setup is a PPT algorithm that, on input security parameters λ, and outputs public *params*.
2) **Key and credential generation:**
   a) TA Key Generator (TAKeyGen): TAKeyGen is a PPT algorithm, on input public *param*s. And then outputs the third party key PK_{TA} and private key SK_{TA}.
   b) Signer Key Generator(SKeyGen): SKeyGen is a PPT algorithm that, on input public *param*s and the third party public key. And then outputs singer's public and private key SK_P.
   c) Verifier Credential Generator(CreGen):CreGen is a PPT algorithm that, on input public *params*, third party public key and user's attribute. And then outputs user credential *Cre*.
3) **Signature generation:** On input singer public key PK_P, private key SK_P, third party public key PK_{TA}, attribute value access policy POL by the singer, message M and public param, then finally outputs signature δ.

4) **Signature verification:** On input singer public key PK_P, third party public key PK_{TA}, attribute value access policy POL' by the singer, the signature δ and user credential Cre_v. If the signature is valid, accepts the output, otherwise rejects the output.

The proposed PCS-PP scheme is formally defined as

$$\prod_{PCS-PP} = \left[\begin{array}{l} (params) \leftarrow Setup(1^\lambda) \\ (SK_{TA}, PK_{TA}) \leftarrow TAKeyGen(params) \\ (SK_p, PK_p) \leftarrow KeyGen(params, PK_{TA}) \\ (Cre_V) \leftarrow CreGen(params, PK_{TA}, S_V) \\ (\delta) \leftarrow Sign(params, PK_{TA}, PK_P, SK_P, POL, M) \\ (1/0) \leftarrow Verify(params, PK_{TA}, PK_P, POL', \delta, Cre_v) \end{array}\right]$$

B. SETUP

1) **Public parameters generation:** Based on input security parameter λ, a trusted authority randomly chooses a prime $p \approx poly(1^\lambda)$, outputs tuples $\Phi = (N = pqr, G, G_T, \hat{e})$, where p, q and r are three different prime numbers. G and G_T are cyclic multiplicative groups with the same order N, $\hat{e} : G \times G \rightarrow G_T$ is a three primes composite order bilinear groups mapping. g_p, g_q and g_r are generators of G_p, G_q and G_r respectively. And then it randomly chooses $h_1, h_2, \ldots, h_n \in G_p$, $(a, t_{1,1}, \cdots, t_{1,m_1}, \cdots, t_{n,1}, \cdots, t_{n,m_n}) \in Z_N$. For public attributes, a specific value is calculated for the value of each attribute name. The set of attribute names is $A_1 = \{h_1 g_q^{t_{1.1}}, \cdots, h_1 g_q^{t_{1.m_1}}\}, \cdots, A_n = \{h_n g_q^{t_{n,1}}, \cdots, h_n g_q^{t_{n,m_n}}\}$. And then generate $param = \{\hat{e}, \{A_i\}, g_p^a, g_p, \Phi\}$.
2) **Selection of hash functions:** Define three hash functions, the first is the file hash function $H_1 : m \rightarrow G_P$, which maps file m to cyclic G_P. The second is the identity hash function $H_2 : \{0, 1\}^* \rightarrow Z_p^*$, which maps a $\{0, 1\}$ string of any length to a hash operation of elements on a finite field Z_p^*. P is assumed to be an attribute value. The last is the collision-resistant hash function $H_3 : \{0, 1\}^* \rightarrow Z_p^*$.

C. KEY AND CREDENTIAL GENERATION

1) **TKeyGen**: On input system parameters $param$, a trusted authority (TA) randomly chooses $\alpha, r \in Z_N/0$, Let $U = g_p^\alpha, W = g_p^r$. Therefore, TKeyGen returns $SK_{TA} = (\alpha, r)$ as private key of the trusted authority and $PK_{TA} = (U, W)$ as the public key of the trusted authority.
2) **CreGen**: On input user attribute set S, Public parameters $param$ and PK_{TA}. TA randomly chooses $t \in Z_P/0$, Let $K = g_p^\alpha g_p^{at}, L = g_p^t, SK_x = h_x^t, x \in S$. And then, CreGen returns $Cre_V = \{K, L, SK_X\}$ to the verifier as a credential. It outputs $Cre_V = \{K, L, SK_X\}$.
3) **SKeyGen**: On input PK_{TA} and Public parameters $params$, TA randomly chooses $k \in Z_p$, Let $X = g_p^k, X' = W^k$. SKeyGen returns $SK_P = k$, $PK_P = (X, X')$ as a private key and public key of the signer. It outputs $SK_P = k$, $PK_P = (X, X')$.

D. SIGNATURE GENERATION

Given MSK, PK_P, SK_P, The policy POL, Message M and $param$, signature is generated according the following.

1) **LSSS Generation**: The matrix M is a share-generating matrix for POL, which has n rows and l columns. The function p corresponds each row in the matrix M to the attribute name in the access policy POL. The LSSS is denoted by (M, p) to expose the public attribute name and hide the attribute value.
2) **Attribute code Generation**: Given a column vector $v = (s, y_2, \cdots, y_l)$, where $s \in Z_p^n$, s is the secret to be shared and $y_2, \cdots, y_l \in Z_p^n$ are randomly chosen. For $i = 1, 2, \cdots, n$, calculates $\lambda_i = v \times M_i.r_1, r_2, \cdots, r_n \in Z_N, \{Y_i, Y_i' \in G_{P_2}\}_{i \in \{1,2,\cdots l\}}$, generates an attribute code for the public attribute based on the attribute value in the POL. Let $C = g_p^{a\lambda_i} A_{i(j)}^{-r_1} Y_i'$ if the attribute value appears in the POL. Otherwise randomly chooses $\beta_i \in Z_N/0$ and $\lambda_i \neq \beta_i$, then computes $C = g_p^{a\beta_i} A_{i(j)}^{-r_1} Y_i'$. C_i is the attribute code set corresponding to each attribute value of the i_{th} attribute name, finally computes:

$$C_i = \left\{\begin{array}{ll} g_p^{a\lambda_i} A_{i(j)}^{-\lambda_i} Y_i', & (L_i \in L) \\ g_p^{a\beta_i} A_{i(j)}^{-r_i} Y_i', & other \end{array}\right\}$$

where $(\{C_1\}, D_1 = g_p^{r_1} Y_1,) \cdots (\{C_n\}, D_n = g_p^{r_n} Y_n)$ and $C' = g_p^s$.
3) **Sign**: Chooses random $v, f \leftarrow Z_P$, computes partial signature $\delta_1 = g_p^v, \delta_2 = X^v, \delta_3 = X'^v$, and then computes:

$$\begin{aligned} \Omega &= \delta_1 || \delta_2 || \delta_3 || f || PK_P || PK_{TA} || m \\ R &= f \oplus H_3(H_3(\Omega) || e(g_p^s, U)) \\ M &= \delta_1 || \delta_2 || \delta_3 || f || PK_P || PK_{TA} || R || \{\{C_m\}, D_m\} \end{aligned}$$

Finally, a signer will generate signature δ for a signed message M, where

$$\begin{aligned} \delta &= \{H_3(\Omega), \delta_1, \delta_2, \delta_3, \delta_4, \{\{C_{\mathrm{m}}\}, D_m\}, m, R, POL'\} \\ \delta_4 &= H_1(M)^x \end{aligned}$$

E. SIGNATURE VERIFICATION

On input signature δ, Cre_V, PK_{TA}, the verifier verifies the signature based on X in Cre_V, W in PK_{TA}, and param g_p. First verifier checks whether the equation:

$$e(\delta_2, g_p) = e(\delta_1, X), \quad e(\delta_3, g_p) = e(\delta_2, W)$$

Hold or not. If not, the verifier outputs "rejection." Otherwise, Verifier's attribute S is an authorized set, and $I \in \{1, \cdots, m\}$, $l = \{i : p(i) \in S\}$. There exists constants $\{\omega_i \in Z_N\}_{i \in I}$ satisfying $\sum_{i \in l} \omega_i \lambda_i = s$ in time polynomial in the size of the share-generating matrix M. so if $\{\lambda_i\}$ are valid shares of any secret s according to (M, p), then

calculates:

$$\theta = e(C', K)/(\prod_{i\in l}(e(C_i, L)e(D_i, SK_{p(i)}))^{\omega_i})$$

After that, computes:

$$f' = R \oplus H_3(H_3(\Omega), \theta)$$
$$M' = \delta_1||\delta_2||\delta_3||f'||PK_P||PK_{TA}||R||\{\{C_m\}, D_m\}||m$$

Then, the verifier checks

$$H_3(M) \stackrel{?}{=} H_3(M'), \quad e(\delta_4, g_p) \stackrel{?}{=} e(H_1(M'), X)$$

hold or not. If not, then it outputs "rejection." Otherwise, it outputs "acceptance."

F. CORRECTNESS

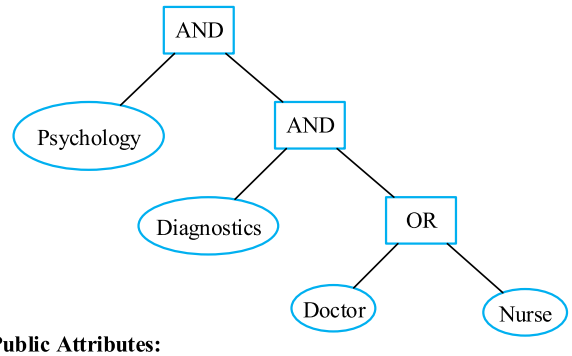
In this section, we analyze the correctness of the proposed PCS-PP scheme. We need to check the case that a policy-controlled signature can be verified by the verifier whose attribute set S_a satisfies the access policy set $l_a = \{i : p(i) \in S\}$ on signature. Verifier can find attribute code for its own attribute value from the attached public attribute hash value. And then we use the attribute code to reconstruct $\{w_i\}_{i\in l}$ according to the attributes of the LSSS matrix $\sum_{i\in L}\lambda_i\omega_i = s$, which can recover s. The detailed derivation process is as follows:

$$\begin{aligned}
\theta &= \frac{e(C', K)}{(\prod_{i\in l}(e(C_i, L)e(D_i, SK_{p(i)}))^{\omega_i})} \\
&= \frac{e(g_p^s, g_p^{\alpha} g_p^{at})}{(\prod_{i\in l}(e(g_p^{a\lambda_i}A_{i(j)}^{-r_1}Y_i', g_p^t)e(g_p^{r_i}Y_i, h_{p(i)}^t))^{\omega_i})} \\
&= \frac{e(g_p, g_p)^{s\alpha}e(g_p, g_p)^{ast}}{(\prod_{i\in l}(e(g_p^{a\lambda_i}(h_{p(i)}g_q^{t_{p(i)}})^{-r_i}Y_i', g_p^t)e(g_p^{r_i}Y_i, h_{p(i)}^t))^{\omega_i})} \\
&= \frac{e(g_p, g_p)^{s\alpha}e(g_p, g_p)^{ast}}{(\prod_{i\in l}(e(g_p^{a\lambda_i}h_{p(i)}^{-r_i}, g_p^t)e(g_p^{r_i}Y_i, h_{p(i)}^t))^{\omega_i})} \\
&= \frac{e(g_p, g_p)^{s\alpha}e(g_p, g_p)^{ast}}{(\prod_{i\in l}(e(g_p^{a\lambda_i}, g_p^t)e(h_{p(i)}^{-r_i}, g_p^t)e(g_p^{r_i}, h_{p(i)}^t))^{\omega_i})} \\
&= \frac{e(g_p, g_p)^{s\alpha}e(g_p, g_p)^{ast}}{(\prod_{i\in l}(e(g_p^{a\lambda_i}, g_p^t))} \\
&= \frac{e(g_p, g_p)^{s\alpha}e(g_p, g_p)^{ast}}{e(g_p, g_p)^{at\sum_{i\in L}\lambda_i\omega_i}} \\
&= e(g_p, g_p)^{\alpha s}
\end{aligned}$$

With $e(g_p, g_p)^{\alpha s}$, verifier can verify signature correctly.

VI. EXPRESSIVENESS ANALYSIS

The expressiveness of access structure stipulates that an eligible access policy should support various combinations of attributes in it. PCS schemes adopt a Boolean formula as an access structure which has a weak expressiveness. For example, Psychology ∧Diagnostics ∧ (Doctor ∨ Nurse) represents that the one who can verify the signature correctly must have attributes psychology, diagnostics, doctor or psychology, diagnostics, nurse. Access structure can also be expressed in a more comprehensible way, like a general access tree, as shown in Fig 5.



Public Attributes:
- Job: Doctor, Nurse
- Department: Dermatology, Psychology
- Affiliation: Emergency, Diagnostics

FIGURE 5. General access tree structure.

In the proposed PCS-PP scheme, we adopt LSSS as an access structure to express access policy. The LSSS matrix is generated by inputting an access tree representing a monotone Boolean formula. The output is an LSSS matrix and the number of rows of the matrix is equal to the number of leaf nodes on the input access tree. So, any monotonic access structure can be converted into an LSSS representation by standard techniques. Hence an access structure based on LSSS has stronger expressiveness.

And in the signature generation, we generate attribute code for the public attribute value, and hide the attribute value that may expose the privacy data into the attribute name by data distortion concept. So, we can transform the access tree to an LSSS matrix as shown in Fig 6.

$$A = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{array}{l} p(1)\text{=Department:*} \\ p(2)\text{=Affiliation:*} \\ p(3)\text{=Job:*} \\ p(4)\text{=Job:*} \end{array}$$

Public Attributes:
- Job: Doctor: $C_{1,1}$, Nurse: $C_{1,2}$
- Department: Dermatology: $C_{2,1}$, Psychology: $C_{2,2}$
- Affiliation: Emergency: $C_{3,1}$, Diagnostics: $C_{3,2}$

FIGURE 6. LSSS with Privacy-preserving policy.

p maps each row of the matrix to attribute name job, department, and affiliation respectively. Given attribute code $\{C_{i,j}\}$ to public attribute value. Users cannot know the detail about attribute values that satisfy access policy except attribute names in an access policy, public attribute values corresponding to attribute codes and their attribute values in the credential. So long as the confidentiality of the attribute codes is

implemented, the privacy of the signature can be guaranteed. And we will give complete proof of privacy in section VII.

Compared with other access structures, the proposed PCS-PP scheme not only has stronger expressiveness, but also supports privacy-preserving policy.

VII. SECURITY ANALYSIS

In this section, we analyze the privacy and unforgeability of the proposed PCS-PP scheme. Dual system encryption is used to prove the privacy and a random oracle model is used to prove the unforgeability.

A. UNFORGEABILITY

Theorem 1: As defined in Definition 3, the proposed PCS-PP scheme is existentially unforgeable under an adaptive chosen message and exposure attack if the CDH assumption holds in the random oracle model.

Proof: Suppose that there exists a forger attacker $\boldsymbol{A}$, which runs the existentially unforgeable game. Start with the construction of queries and run the existentially unforgeability:

On input (G, p, g, g^a, g^b) as an instance of the CDH problem, which satisfies $a, b \in Z_p^*, g \in G$.

1) **Setup**: Simulator $\boldsymbol{F}$ runs setup algorithm, and sends public param to forger A.
   a) Simulator $\boldsymbol{F}$ randomly chooses $h_1, h_2, \ldots, h_n \in G_p$, $a, t_{1,1}, \cdots, t_{1,m_1}, \cdots, t_{n,1}, \cdots, t_{n,m_n} \in Z_N$. Calculates a specific value $A_1 = \{h_1 g_q^{t_{1.1}}, \cdots, h_1 g_q^{t_{1.m_1}}\}, \cdots, A_n = \{h_n g_q^{t_{n,1}}, \cdots, h_n g_q^{t_{n.m_n}}\}$ for the attribute value of each attribute name, resulting in a collection of attribute names.
   b) Randomly chooses $\alpha, r \in Z_N/0$, let $X = g_p^{ka}, X' = W^{ka}$ be trusted authority's public key.
   c) Randomly chooses $k \in Z_p$, let $X = g_p^{ka}, X' = W^{ka}$ be signer's public key.
2) **Queries:** $\boldsymbol{A}$ is given an access to the above queries, and carries out $q_{H_1}, q_{H_2}, q_{H_3}, q_s, q_c$ times hash random oracle *HO* query, singing oracle SO query, certification oracle CO query.
   a) **HO** query: H_1: $\boldsymbol{F}$ maintains $List_{H1}$ for H_1 oracle, assume that forger $\boldsymbol{A}$ can make up to q_{H_1} times for H_1 queries. If $\boldsymbol{F}$ receives message M_i queries from $\boldsymbol{A}$, and then $\boldsymbol{F}$ checks if it exists in the list. Otherwise, picks an integer random $\eta \epsilon [1, q_{H1}]$ and does the following: If $i \neq \eta$, C randomly chooses $\bar{l} \leftarrow Z_P$, let $H_1(M_i) = g_p^l =$ and returns it to $\boldsymbol{A}$. If $i = \eta$, $\boldsymbol{F}$ chooses $H_1(M_i) = g_{p^b}$ and returns it to $\boldsymbol{A}$.
   H_2: $\boldsymbol{F}$ maintains $List_{H2}$ for H_2 oracle, assume that forger $\boldsymbol{A}$ can make up to q_{H_2} times for H_2 queries. If $\boldsymbol{F}$ receives identity attribute value $\mathrm{ID_i}$ queries from $\boldsymbol{A}$, and then $\boldsymbol{F}$ checks if it exists in the list, and then returns result. Otherwise, picks an integer random $l_1 \leftarrow Z_P$ and returns $H_2(\mathrm{IDi}) = l_1$ as result to $\boldsymbol{A}$.
   H_3: $\boldsymbol{F}$ maintains List $_{\mathrm{H3}}$ for H_3 oracle, assume that $\boldsymbol{A}$ can make up to q_{H_3} H_3 queries. $\boldsymbol{F}$ randomly chooses $l_2 \leftarrow Z_P$ for $H_3(M)$, and then let $H_3(M) = l_2$.
   b) **CRO** query: $\boldsymbol{F}$ runs the credentials generation algorithm, generates the credentials Cre for attribute set S, and returns the credentials Cre to $\boldsymbol{A}$.
   c) **SO** query: On input POL and message M, $\boldsymbol{A}$ computes a policy-controlled signature with privacy-preservation as follows:
   Generated $(\{C_1\}, D_1 = g_p^{r_1} Y_1), \cdots (\{C_n\}, D_n = g_p^{r_n} Y_n)$ by using the established access policy, and then randomly chooses $v, f \leftarrow Z_P$ to get partly signature $\delta_1 = g_p^v, \delta_2 = X^v, \delta_3 = X'^v$. Let

$$\Omega = \delta_1 || \delta_2 || \delta_3 || f || PK_P || PK_{TA} || M$$
$$R = f \oplus H_3(H_3(\Omega) || e(g_p^s, U))$$

   And then checks

$$H_1(\delta_1 || \delta_2 || \delta_3 || f || PK_P || PK_{TA} || R || \{\{C_m\}, D_m\}) \stackrel{?}{=} g^b$$

   If not, let $\delta_4 = X^{\bar{l}}$, otherwise outputs failure. Finally returns $\delta = \{H_3(\Omega), \delta_1, \delta_2, \delta_3, \delta_4, \{\{C_m\}, D_m\} M, R, POL'\}$.
3) **Forging**: After $\boldsymbol{A}$ access the above query, $\boldsymbol{A}$ generates a forged signature δ^* on the new message M' and on the access policy POL^*. If δ^* is valid and the signature list of the oracle does not look up, then $\boldsymbol{A}$ wins the game.

Assuming that $\boldsymbol{A}$ must do a hash query before performing certificate query and signing query or using defined message M' and POL^* before generating a signature.

Let $Succ_{EU-CMCEA} = \varepsilon$ be the probability that $\boldsymbol{A}$ wins the game. We denote by e the base of the natural logarithm and let $q_{H_1} \geq q_s$ be a polynomial upper bound on the number of queries that $\boldsymbol{A}$ makes to the HO queries. Therefore, we can analyze the success probability that $\boldsymbol{A}$ output a signature δ^* on message M', where $\delta_4^* = H_1(M)^{ks} = (g^b)^{ks}$, and wins the above game as follows:

$\mathbf{E_1}$: $\boldsymbol{F}$ does not abort during the issuing of queries to the SO. The probability of this event is $(1 - \frac{1}{q_{H_1}})^{q_s}$, Because the signature query must end when the last hash query is left, the maximum number of signature queries is $q_{H_1} - 1$. So probability of this event is greater than $(1 - \frac{1}{q_{H_1}})^{q_{H_1}-1}$.

$\mathbf{E_2}$: After the signature is generated, the $\boldsymbol{F}$ does not stop producing the signature. Therefore, $\boldsymbol{A}$ must leave a hash query at the end to generate the signature.

However, if $H_1(\delta_1 || \delta_2 || \delta_3 || f || PK_P || PK_{TA} || R || \{\{C_m\}, D_m\}) = H_1(M') \neq g^b$ for δ_4^*, then $\boldsymbol{F}$ aborts the simulation. Hence, the probability of this event is greater than $(1 - \frac{1}{q_{H_1}})^{q_{H_1}-1}$.

So the probability that $\boldsymbol{A}$ wins the above game and outputs a signature on a message M', where $\delta_4^* = H_1(M')^x = (g^b)^x$ is $\varepsilon((1 - \frac{1}{q_{H_1}})^{q_{H_1}-1})^2$. From the above outputs by $\boldsymbol{A}$, $\boldsymbol{F}$ obtains $\delta_4^* = H_1(M')^x = (g^b)^x$ when $x = a$, and then $\boldsymbol{F}$ returns

$\delta_4 = g^{ab}$ as an output for the CDH problem with non-negligible probability as mentioned above.

B. PRIVACY

In the proposed PCS-PP scheme, LSSS is to develop access structure to express access policy. The three primes composite order bilinear groups are to implement privacy-preserving policy based on data distortion.

In the signature generation, the signer designs access policy. And attribute value of access policy is privacy by hiding the attribute value and exposing attribute name. For example, math teacher and 45 years old substitute teacher and age in a way that privacy treat attribute values. The signature code is constructed for each attribute value using a three primes composite order bilinear groups.

Privacy-preserving policy is implemented by confidentiality of attribute code. So, if the attribute code with confidentiality, the attacker cannot get the verifier of private information. And then the proposed PCS-PP scheme has privacy.

Theorem 2: As defined in Definition 4, the proposed PCS-PP scheme is privacy with the **Assumption** 1, 2 and 3 holding.

The dual-system encryption proposed by Water *et al*. [23] is applied to prove the confidentiality of attribute codes. In the dual-system system, if the private key and ciphertext are generated by the system key or by the encryption algorithm, it is called general ciphertext and key. Then the definition of semi-functional key and semi-functional ciphertext are introduced. This means when the user meets the access structure, the regular private key can decrypt the regular ciphertext and the semi-functional ciphertext, and the semi-functional key can decrypt the regular ciphertext but cannot decrypt the semi-functional ciphertext. Then we define a series of games in which the challenge ciphertext is changed to semi-functional ciphertext and the private key is gradually changed to the semi-functional private key. At last, prove the confidentiality by indistinguishable series games when both challenge ciphertext and private key are semi-functional.

Proof: Assume that the opponent makes n private key queries in a game, and the game is defined as follows:

$Game_{real}$: The game is real, and all ciphertext and private keys are regular and normal.

$Game_0$: In the game, the private key can be normally questioned, challenge ciphertext is a semi-function.

$Game_{k.1}$: In game k, the first k keys are semi-functional while the remaining keys are normal.

$Game_{final}$: The challenge ciphertext is a semi-functional ciphertext generated by encrypting a random message. All private keys are semi-functional private keys.

Lemma 1: Supposes there exists an algorithm such that $Adv_{\Phi,\Psi(\lambda)}^{Game_{real}} - Adv_{\Phi,\Psi(\lambda)}^{Game_0} = \varepsilon$. Then we can build an algorithm ψ with advantage $> \frac{\varepsilon}{2}$ in breaking Assumption 1 in section.

Proof: Given $D = (G, g_p, g_q)$ in **Assumption** 1.

1) **Setup**: The challenger $\boldsymbol{C}$ runs the Setup algorithm to generate public parameters in Section VII.
2) **Phase** 1: The adversary $\boldsymbol{A}$ can make Create, Delegate, and Reveal certificate queries about attributes.
3) **Challenge**: The $\boldsymbol{A}$ gives the $\boldsymbol{C}$ two secret S_0 and S_1 and a challenge access structure T_0 and T_1. $\boldsymbol{C}$ sets $b \in \{0, 1\}$ randomly, and get linear secret sharing of S_b by using access structure T_b. And then $\boldsymbol{C}$ uses these secret to produce a ciphertext $C = e(g_p^{Sb}, Z)^{\alpha}$. B chooses $\beta_i \in Z_N/0$ and $\lambda_i \neq \beta_i$ randomly when $l_i \in POL$, set $C_i' = Z^{a\lambda_i} A_{i(j)}^{-r_i} Y_i'$. If is not, let $C_i' = Z^{a\beta_i} A_{i(j)}^{-r_i} Y_i'$, $D_i' = Z^{r_i} Y_i$, $i = 1, 2, \cdots$. Then $\boldsymbol{C}$ sends $\boldsymbol{A}$ $((C_i', D_i')_{i=1,2,\cdots,l})$.
4) **Phase** 2: Repeat Phase 1.
5) **Guess**: The adversary $\boldsymbol{A}$ must output a guess b' for b.

If $Z = T_1 \in G_{pr}$, then this is a semi-functional ciphertext. If $Z = T_2 \in G_p$, this is a normal ciphertext. Hence, $\boldsymbol{C}$ can use the output of $\boldsymbol{A}$ to distinguish between G_{pr} and G_p with nonnegligible probabilities.

Lemma 2: Suppose there exists an algorithm such that $Adv_{\psi}^{Game_{1,K-1}} - Adv_{\psi}^{Game_{1,K}} = \varepsilon$. Then we can build an algorithm ψ with advantage $> \frac{\varepsilon}{2}$ in breaking Assumption 2.

Proof: Given $D = (G, g_p, g_q, Z_1, Z_2)$ in **Assumption** 2.

1) **Setup**: Repeat Lemma 1.
2) **Phase**1: The attacker $\boldsymbol{A}$ makes the i_{th} query through the certificate generation algorithm. If $i \leq k$, then this is a semi-functional certificate $K = g_p^{\alpha} T, L = T$. Otherwise this is a normal certificate.
3) **Challenge**: The $\boldsymbol{A}$ gives the challenger $\boldsymbol{C}$ two secret S_0 and S_1 and a challenge access structure T_0 and T_1. $\boldsymbol{C}$ sets $b \in \{0, 1\}$ randomly, and get linear secret sharing of S_b by using access structure T_b. And then $\boldsymbol{C}$ uses this secret to produce a ciphertext $C = e(g_p^{Sb}, Z_1 Z_2)^{\alpha}$. $\boldsymbol{C}$ chooses $\beta_i \in Z_N/0$ and $\lambda_i \neq \beta_i$ randomly when $l_i \in POL$, set $C_i' = (Z_1 Z_2)^{a\lambda_i} A_{i(j)}^{-r_i} Y_i'$. If not, let $C_i' = (Z_1 Z_2)^{a\beta_i} A_{i(j)}^{-r_i} Y_i'$, $D_i' = (Z_1 Z_2)^{r_i} Y_i (i = 1, 2, \cdots)$. Then $\boldsymbol{C}$ sends $\boldsymbol{A}$ ciphertext $((C_i', D_i')_{i=1,2,\cdots,l})$.
4) **Phase**2: Repeat Phase 1.
5) **Guess**: The $\boldsymbol{A}$ must output a guess b' for b.

If $Z = T_1 \in G_{pr}$, then this is a semi-functional ciphertext. If $Z = T_2 \in G_p$, this is a normal ciphertext. Hence, $\boldsymbol{C}$ can use the output of $\boldsymbol{A}$ to distinguish between G_{pr} and G_p with nonnegligible probabilities.

Lemma 3: Suppose there exists an algorithm such that $Adv_{\psi}^{Game_{q,1}} - Adv_{\psi}^{Game_{final}} = \varepsilon$. Then we can build an algorithm ψ with advantage $> \frac{\varepsilon}{2}$ in breaking Assumption 3.

Proof: Given $D = (G, g_p, g_P^{\alpha} X_2, g_p^s Y, Z_2, X)$ in **Assumption** 3.

1) **Setup**: Repeat Lemma 1.
2) **Phase**1: The attacker $\boldsymbol{A}$ makes the query through the certificate generation algorithm. Randomly choose $w \in Z_N$, and generate certificate $K = g_p^{\alpha} X_2 (g^t Z_2)^a$, $L = (g_p^t Z_2)^w$.
3) **Challenge**: $\boldsymbol{A}$ gives the challenger $\boldsymbol{C}$ two secret S_0 and S_1 and a challenge access structure T_0 and T_1. $\boldsymbol{C}$ sets $b \in \{0, 1\}$ randomly, and get linear secret sharing of S_b

TABLE 1. The comparison of the related privacy-preserving schemes.

Category/ Scheme	Group of order	Method of calculation	Security feature	Expressiveness	Scope of hidden	Primitives
[34]	Prime order	Outsourced	Adaptively- CPA	AND-Gates	PartlyHidden	Encryption
[35]	Prime order	Outsourced	Adaptively-CPA	AND-Gates	PartlyHidden	Encryption
[36]	Prime order	Outsourced	Adaptively-CCA	LSSS	Full Hidden	Encryption
[37]	Composite order	Local	Adaptively-CCA	LSSS	Full Hidden	Encryption
PCS-PP	Composite order	Local	EU-CMA	LSSS	PartlyHidden	Signature

by using access structure T_b. And then $\boldsymbol{C}$ uses this secret to produce a ciphertext. $\boldsymbol{C}$ choose $\beta_i \in Z_N/0$ and $\lambda_i \neq \beta_i$ randomly when $l_i \in POL$, set $C_i' = (g_p^s Y)^{a\lambda_i} A_{i(j)}^{-r_i} Y_i'$. If not, let $C_i' = (g_p^s Y)^{a\beta_i} A_{i(j)}^{-r_i} Y_i'$, $D_i' = (g_p^s Y)^{r_i} Y_i$, i $= 1, 2, \cdots$. Then $\boldsymbol{C}$ sends A ciphertext $((C_i', D_i')_{i=1,2,\cdots,l})$.

4) **Phase**2: Repeat Phase 1.
5) **Guess**: $\boldsymbol{A}$ must output a guess b' for b.

If $Z = T_1 = e(g_p, g_p)^{\alpha s}$, then this is a semi-functional ciphertext. If $Z = T_2 \in G_p$, this is a normal ciphertext. Hence, $\boldsymbol{C}$ can use the output of $\boldsymbol{A}$ to distinguish between $T_1 = e(g_p, g_p)^{\alpha s}$ and G_p with nonnegligible probabilities.

If *Assumption* 1, 2 and 3 hold, then attribute code of prime composite order construction in the proposed PCS-PP scheme is secure, so then the proposed PCS-PP is privacy.

VIII. COMPARISONS

In this section, we analyze and compare the proposed PCS-PP scheme with the related works [1], [2], [34]–[37] from features and performance.

A. FEATURES

The proposed PSC-PP scheme uses an LSSS matrix to develop an access structure which has strong expressiveness and applies three primes composite order bilinear groups based on data distortion to expose the public attribute name and hide the attribute value to provide privacy preservation. However, there is no other policy-controlled scheme that supports policy privacy preservation. Only the recent works [34]–[37] are related to our work, but they differ. So, we mainly compare it from the group of order, the method of calculation, the security feature, the expressiveness, the scope of hidden and primitives.

Schemes [34] and [35] extend the privacy preservation of CP-ABE to CP-AB-BE, which also provide kinds of methods like partially hidden policy, direct revocation, and verifiable outsourced decryption. However, they only support the AND Gate access policy which has weak expressiveness. A CP-ABE with a full hidden policy [36] is based on LSSS. But, since [36] will be a similar matching process before the calculation with full hidden policy, which also increases the computation. So, schemes [34]–[36] give all the computation to outsourced decryption. Once the outsourced institution is attacked, it will also leak user information. The scheme [37] also provides full hidden CP-ABE on LSSS. But its computations are too heavy without outsourced decryption. Due to the nature of the policy-controlled signature, the proposed PCS-PP scheme supports fine-grained access to verifiers. Table 1 illustrates that the proposed PCS-PP scheme is the only policy-controlled signature scheme that provides strong expression, unforgeability, and privacy-preserving policy.

B. PERFORMANCE

We analyze the performance of the proposed PCS-PP scheme from four aspects: key certificate storage, signature operation, verification operation, and total operation. To the best of our knowledge, this is the first PCS construction that supports privacy-preserving policy, so we compare PCS [1] and universal policy-controlled signature UPCS [2] with our proposed PCS-PP scheme. The result is shown in Table 2, where E be the computational cost of exponentiation in G_P or G_T, M is the computational cost of multiplication in G_P, P is the computational cost of a bilinear pairing function, N is all of the attributes in the access policy, and I is the attribute that satisfies the policy. Let $|G_P|$ be the actual size of the element in group G_P, which is about 170 bits, and $|p|$ be the actual element size of the element in Z_P, which is about 160 bits.

And then we provide an experimental evaluation of our proposed PCS-PP scheme and other schemes [1], [2] on the hardware platform, which consists of AMD Ryzen 5 3600 @3.60GHz processor and 16GB memory, and software platform, which includes Ubuntu operating system for 64 bits and Golang from the Stanford Pairing-Based Crypto library [41].

Fig.7 and Fig.8 present the comparison of the practical computation cost on the execution of verification phase and sign phase between the proposed PCS-PP scheme and the relevant schemes [1], [2] in the case with the same number of attributes. Fig.7 shows that the computation cost of the proposed PCS-PP scheme is almost the same as schemes [1], [2] in the sign phase and the efficiency of the sign phase is proportional to the number of attributes.

However, Fig.8 demonstrates that the computation cost of the proposed PCS-PP scheme is the lowest among the schemes [1], [2] in the verification phase. The reason is that the proposed PCS-PP scheme hides the original policy and exposes the attribute name policy, the verifier does not need

TABLE 2. Comparison of policy-controlled signature schemes with storage.

Category/Scheme	[1]	[2]	PCS-PP
PK_{TA}	$2\|G_P\|$ (256 Bytes)	$2\|G_P\|$ (256 Bytes)	$2\|G_P\|$ (256 Bytes)
SK_{TA}	$2\|p\|$ (40 Bytes)	$2\|p\|$ (40 Bytes)	$2\|p\|$ (40 Bytes)
PK_p	$\|G_P\|$ (128 Bytes)	$2\|G_P\|$ (256 Bytes)	$2\|G_P\|$ (256 Bytes)
SK_p	$\|p\|$ (20 Bytes)	$\|p\|$ (20 Bytes)	$\|p\|$ (20 Bytes)
Cre_v	$3\|G_P\|$(384 Bytes)	$3\|G_P\|$ (384 Bytes)	$2\|G_P\|$ (256 Bytes)
Signature Operation	$(9+n)$ $E+$ $(1+n)$ $M+3P$	$(4+n)E+$ $(n-1)$ $M+P$	$(5+n)$ $E+nM+P$
Verification Operation	$3E+2M+$ $(9+2n)$ P	$(6+2n)$ P	$(7+2i)$ P
Total Operation	(12) $E+$ $(3+n)$ M $+$ $(12+2n)$ P	$(4+n)$ $E+$ $(n-1)$ $M+(7+2n)P$	$(5+n)$ $E+nM$ $+(8+2i)P$

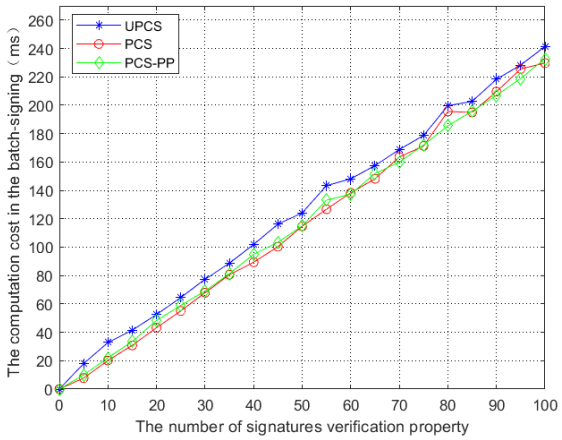


FIGURE 7. Comparison of time cost in signature generation.

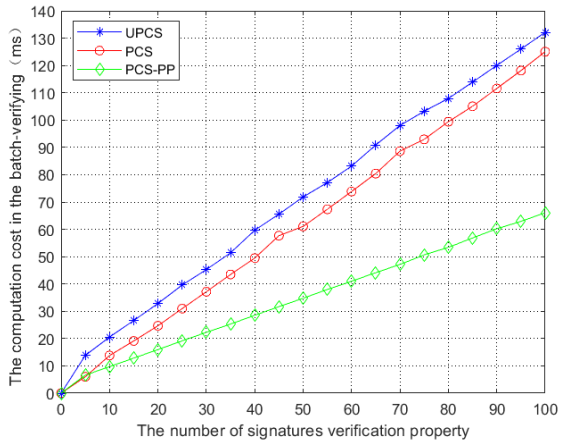


FIGURE 8. Comparison of time cost in signature verification.

to calculate all the attributes in the verification stage, but just calculates the signature component of the exposed attribute name and the attributes owned by his/her certificate, so as to reduce the pairing calculation problem.

According to the performance analysis, the proposed PCS-PP scheme has superiority in aspects of features, computation cost, storage compared with the existing signature schemes.

IX. CONCLUSION

In order to improve the expressiveness of access structures and provide the privacy preservation of policy-controlled signature, we use an LSSS matrix to develop access structure which has strong expressiveness and apply three primes composite order bilinear groups based on data distortion to expose the public attribute name and hide the attribute value to provide the privacy preservation in the proposed PSC-PP scheme which supports correctness, unforgeability and privacy-preserving policy and fine-grained signature verification. Compared to the related signature schemes, the proposed PSC-PP scheme has certain superiority in features, computation cost and storage.

Nevertheless, there are some remaining issues that can be handled in future work. The computation of signature generation and size of the signature is linear to the size of its policy and disturbance attributes. Therefore, the next step is to research how to design construction of the policy-controlled signature in which the cost of signature is constant.

REFERENCES

[1] P. Thorncharoensri, W. Susilo, and Y. Mu, "Policy-controlled signatures and their applications," *Comput. Standards Interface*, vol. 50, pp. 26–41, Feb. 2017, doi: 10.1016/j.csi.2016.08.005.

[2] P. Thorncharoensri, W. Susilo, and Y. Mu, "Policy controlled system with anonymity," *Theor. Comput. Sci.*, vol. 745, pp. 87–113, Oct. 2018, doi: 10.1016/j.tcs.2018.05.038.

[3] Z. Liu, J. Hsu, R. Tso, and T. Wu, "Policy-controlled signature from NTRU lattice," in in *Proc. 9th IFIP Int. Conf. New Technol., Mobility Secur. (NTMS)*, Paris, France, 2018, pp. 1–5, doi: 10.1109/NTMS.2018.8328712.

[4] L. Cheung and C. Newport, "Pr ovably secure ciphertext policy ABE," in *Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS)*, New York, NY, USA, Oct. 2007, pp. 456–465, doi: 10.1145/1315245.1315302.

[5] V. Odelu, A. K. Das, Y. S. Rao, S. Kumari, M. K. Khan, and K.-K.-R. Choo, "Pairing-based CP-ABE with constant-size ciphertexts and secret keys for cloud environment," *Comput. Standards Interface*, vol. 54, pp. 3–9, Nov. 2017, doi: 10.1016/j.csi.2016.05.002.

[6] V. Odelu, A. K. Das, M. Khurram Khan, K.-K.-R. Choo, and M. Jo, "Expressive CP-ABE scheme for mobile devices in IoT satisfying constant-size keys and ciphertexts," *IEEE Access*, vol. 5, pp. 3273–3283, 2017, doi: 10.1109/ACCESS.2017.2669940.

[7] Y. Jiang, W. Susilo, Y. Mu, and F. Guo, "Flexible ciphertext-policy attribute-based encryption supporting AND-gate and threshold with short ciphertexts," *Int. J. Inf. Secur.*, vol. 17, no. 4, pp. 463–475, Aug. 2018, doi: 10.1007/s10207-017-0376-y.

[8] Y. Cheng, H. Zhou, J. Ma, and Z. Wang, "Efficient CP-ABE with non-monotonic access structures," *Cloud Comput. Secur.*, vol. 10603, pp. 315–325, Nov. 2017, doi, 10.1007/978-3-319-68542-7_26.

[9] Y. Zhang and D. Zheng, "Anonymous attribute-based encryption with large universe and threshold access structures," in *Proc. 7 IEEE Int. Conf. Comput. Sci. Eng. (CSE)*, Guangzhou, China, Jul. 2017, pp. 870–874, doi: 10.1109/CSE-EUC.2017.175.

[10] J. Li, W. Yao, J. Han, Y. Zhang, and J. Shen, "User collusion avoidance CP-ABE with efficient attribute revocation for cloud storage," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1767–1777, Jun. 2018, doi: 10.1109/JSYST.2017.2667679.

[11] S. Belguith, N. Kaaniche, and G. Russello, "PU-ABE: Lightweight attribute-based encryption supporting access policy update for cloud assisted IoT," in *Proc. IEEE 11th Int. Conf. Cloud Comput. (CLOUD)*, San Francisco, CA, USA, Jul. 2018, pp. 924–927, doi: 10.1109/CLOUD.2018.00137.

[12] H. Tang, Y. Cui, C. Guan, J. Wu, J. Weng, and K. Ren, "Enabling ciphertext deduplication for secure cloud storage and access control," in *Proc. 11th ACM Asia Conf. Comput. Commun. Secur.*, New York, NY, USA, May 2016, p. 59, doi: 10.1145/2897845.2897846.

[13] Y. Zhang, D. He, and K. K. R. Choo, "BaDS: Blockchain-based architecture for data sharing with ABS and CP-ABE in IoT," *Wireless Commun. Mobile Comput.*, 2018, Nov. 2018, Art. no. 2783658.

[14] Q. M. Malluhi, A. Shikfa, and V. C. Trinh, "A ciphertext-policy attribute-based encryption scheme with optimized ciphertext size and fast decryption," in *Proc. ACM Asia Conf. Comput. Commun. Secur.*, New York, NY, USA, Apr. 2017, pp. 230–240, doi: 10.1145/3052973.3052987.

[15] S.-G. Zhou, F. Li, Y.-F. Tao, and X.-K. Xiao, "Privacy preservation in database applications: A survey," *Chin. J. Comput.*, vol. 32, no. 5, pp. 847–861, Aug. 2009.

[16] R. C. Wing, L. Jiuyong, F. Ada, and K. Wang, "(A,K)-anonymity: An enhanced k-anonymity model for privacy preserving data publishing," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, 2006, pp. 754–759, doi: 10.1145/1150402.1150499.

[17] A. Machanavajjhala, D. Kifer, J. Abowd, J. Gehrke, and L. Vilhuber, "Privacy: Theory meets practice on the map," in *Proc. IEEE 24th Int. Conf. Data Eng.*, Cancun, Mexico, Apr. 2008, pp. 277–286, doi: 10.1109/ICDE.2008.4497436.

[18] T. Junichi, M. Abe, and T. Okamoto, "Efficient functional encryption for inner-product values with full-hiding security," in *Proc. Int. Conf. Inf. Secur.*, vol. 9866. Cham, Switzerland: Springer, Aug. 2016, pp. 408–425, doi: 10.1007/978-3-319-45871-7_24.

[19] D. Boneh, A. Sahai, and B. Waters, "Functional encryption: Definitions and challenges," in *Proc. 8th Theory Cryptogr. Conf. (TCC)*, Berlin, Germany, vol. 6597, 2011, pp. 253–273, doi: 10.1007/978-3-642-19571-6_16.

[20] M. Murad and Y. Rodhaan, "Computationally efficient fine-grain cube cp-abe scheme with partially hidden access structure," *Commun. Comput. Inf. Sci.*, vol. 1210, pp. 135–156, Aug. 2020, doi: 10.1007/978-981-15-7530-3_10.

[21] A. Nuttapong, "Dual system encryption framework in prime-order groups via computational pair encodings," in *Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, Berlin, Germany, Nov. 2016, pp. 591–623, doi: 10.1007/978-3-662-53890-6_20.

[22] B. Mihir and G. Fuchsbauer, "Policy-based signatures," in *Proc. Int. Workshop Public Key Cryptogr.*, Berlin, Germany, vol. 8383, 2014, pp. 520–537, doi: 10.1007/978-3-642-54631-0_30.

[23] H. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-based signatures: Achieving attribute-privacy and collusion-resistance," *IACR Cryptol. Eprint Arch.*, vol. 15, p. 328, Apr. 2008.

[24] Y. Chen, Y. Zhao, H. Xiong, and F. Yue, "A certificateless strong designated verifier signature scheme with non-delegatability," *Int. J. Netw. Secur.*, vol. 19, no. 4, pp. 573–582, 2017.

[25] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology*, vol. 196. Berlin, Germany: Springer, 1984.

[26] V. Goyal, A. Jain, O. Pandey, and A. Sahai, "Bounded ciphertext policy attribute based encryption," in *Proc. 35th Int. Colloq. Autom., Lang. Program.*, vol. 5126, Jul. 2008, pp. 579–591.

[27] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy (SP)*, Berkeley, CA, USA, May 2007, pp. 321–334, doi: 10.1109/SP.2007.11.

[28] T. Nishide, K. Yoneyama, and K. Ohta, "Attribute-based encryption with partially hidden encryptor-specified access structures," in *Proc. 6th Int. Conf. Appl. Cryptogr. Netw. Secur.*, Berlin, Germany, vol. 5037, Jun. 2008, pp. 111–129.

[29] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Proc. Int. Workshop Public Key Cryptogr.*, vol. 6571. Berlin, Germany: Springer, 2011, pp. 53–70, doi: 10.1007/978-3-642-19379-8_4.

[30] J. Lai, R. H. Deng, and Y. Li, "Expressive CP-ABE with partially hidden access structures," in *Proc. 7th ACM Symp. Inf., Comput. Commun. Secur. (ASIACCS)*, New York, NY, USA, 2012, p. 18, doi: 10.1145/2414456.2414465.

[31] U. C. Yadav and S. T. Ali, "Ciphertext policy-hiding attribute-based encryption," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Kochi, Kerala, Aug. 2015, pp. 2067–2071, doi: 10.1109/ICACCI.2015.7275921.

[32] R. Xu, Y. Wang, and B. Lang, "A tree-based CP-ABE scheme with hidden policy supporting secure data sharing in cloud computing," in *Proc. Int. Conf. Adv. Cloud Big Data*, Nanjing, China, Dec. 2013, pp. 51–57, doi: 10.1109/CBD.2013.9.

[33] S. Yan, H. Zhen, L. F. Mei, and L. Lei, "Attribute-based encryption with hidden policies in the access tree," *J. Commun.*, vol. 4, pp. 119–126, Dec. 2015.

[34] Z. Zhou, D. Huang, and Z. Wang, "Efficient privacy-preserving ciphertext-policy attribute based-encryption and broadcast encryption," *IEEE Trans. Comput.*, vol. 64, no. 1, pp. 126–138, Jan. 2015, doi: 10.1109/TC.2013.200.

[35] H. Xiong, Y. Zhao, L. Peng, H. Zhang, and K.-H. Yeh, "Partially policy-hidden attribute-based broadcast encryption with secure delegation in edge computing," *Future Gener. Comput. Syst.*, vol. 97, pp. 453–461, Aug. 2019, doi: 10.1016/j.future.2019.03.008.

[36] F. Khan, H. Li, L. Zhang, and J. Shen, "An expressive hidden access policy CP-ABE," in *Proc. IEEE 2nd Int. Conf. Data Sci. Cyberspace (DSC)*, Shenzhen, China, Jun. 2017, p. 178, doi: 10.1109/DSC.2017.29.

[37] C. D. Wei and T. Bo, "An attribute-based encryption scheme with hidden policy based on LSSS," *Comput. Technol. Develop.*, vol. 28, pp. 119–124, Dec. 2018.

[38] D. Boneh, E. J. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts," in *Theory Cryptography*. Cham, Switzerland: Springer, 2005, pp. 325–341.

[39] H. Yin, J. Zhang, Y. Xiong, L. Ou, F. Li, S. Liao, and K. Li, "CP-ABSE: A ciphertext-policy attribute-based searchable encryption scheme," *IEEE Access*, vol. 7, pp. 5682–5694, 2019, doi: 10.1109/ACCESS.2018.2889754.

[40] D. Zheng, A. Wu, Y. Zhang, and Q. Zhao, "Efficient and privacy-preserving medical data sharing in Internet of Things with limited computing power," *IEEE Access*, vol. 6, pp. 28019–28027, 2018, doi: 10.1109/ACCESS.2018.2840504.

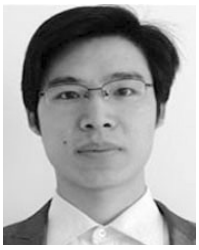
[41] B. Lynn. (2013). *The Stanford Pairing Based Crypto Library*. [Online]. Available: http://crypto.stanford.edu/pbc

XURUI ZHENG was born in China, in 1996. He is currently a Graduate Student in computer application technology from the School of Computer, South-Central University for Nationalities, China. His current research interests include policy-controlled signature, privacy preservation, and identity authentication on blockchain.

FUYONG ZHENG was born in China, in 1973. He received the B.S. degree in communication engineering from North China Electric Power University. He is currently the Deputy General Manager of the Information and Communication Branch, State Grid Jiangxi Electric Power Company. His current research interests include electric power informatization, data middle platform, and cyberspace security.

XIANMING LIU was born in China, in 1977. He received the Ph.D. degree in computer software and theory from Sun Yat-sen University, China. He is currently the Director of the Data Management of Information and Communication Branch, State Grid Jiangxi Electric Power Company. He has received three provincial and ministerial science and technology progress awards, seven invention patents, and more than ten EI search articles. His current research interests include electric power informatization, data middle platform, and cyberspace security.

DEJUN WANG was born in 1974. He received the Ph.D. degree in information security from Wuhan University, China. He is currently an Associate Professor with the School of Computer, South-Central University for Nationalities, China. He has authored/coauthored over 20 papers in international/national journals and conferences. His current research interests include security protocols and formal methods.



JUN WANG received the Ph.D. degree in computer science from Wuhan University, China, in 2017. He is currently a Lecturer with the College of Computer Science, South-Central University for Nationalities, China. His research interests include information security and privacy in wireless mobile networks.

BO MENG was born in China, in 1974. He received the M.S. degree in computer science and technology and the Ph.D. degree in traffic information engineering and control from the Wuhan University of Technology, Wuhan, China, in 2000 and 2003, respectively. From 2004 to 2006, he worked at Wuhan University, as a Postdoctoral Researcher in Information Security. From 2014 to 2015, he worked at the University of South Carolina, as a Visiting Scholar. He is currently a Full Professor with the School of Computer Science, South-Central University for Nationalities, China. He has authored/coauthored over 50 papers in international/national journals and conferences. In addition, he has also published two books *Automatic Generation, Verification, And Implementation of Security Protocols* and *Secure Remote Voting Protocol* (Science Press) in China. His current research interests include block chain, security protocols, and formal method.