Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH Open Access

A designated verifier multi-signature scheme in multi-clouds

Chaoyue Tan[1], Yuling Chen[1,2*], Yongtang Wu[2], Xiaochuan He[1] and Tao Li[1]

Abstract

Multi-cloud computing provides services by used different clouds simultaneously multi-signature can be used as the interactive technology between multi-cloud and users. However, the limited resources of some terminal devices make multi-signature, which based on bilinear map, is not suitable for multi-cloud computing environment. In addition, the signers are disclosure in multi-signature so there is the risk of attack. To solve this issues, this paper proposes a certificateless designated verifier multi-signature scheme based on multivariable public key cryptography (MPKC). Firstly, the formalized definition and security model of the proposed scheme are given. Secondly, it is proved that the proposed scheme is against adaptive chosen-message attacks. Finally, the analysis shows that the proposed scheme is more efficiency and suitable for multi-cloud. Moreover, the proposed scheme can hidding signature source to achieve privacy protection.

Keywords: Muti-cloud, MPKC, Designated verifier, Efficiency, Hiding information

Introudction

With the development of fifth-generation (5G) technology [1], the data processing capacity requirements of the Internet of Things (IoT) have been increasing [2, 3], so, many users entrust data processing to cloud computing for more efficiency [4]. Cloud computing is a paradigm of distributed computing, which can execute computing by dynamically scalable virtualized resources on the basis of the existence of the internet [5]. However, there are many security issues in cloud computing. Such as malicious attacks [6], privacy of data [7], real-time data processing [8] and issues caused by multi-cloud computing [9].

A. Malicious attacks

Cloud computing does not use virtual private network (VPN) [10], which means that servers in cloud computing can access the Internet, it makes cloud computing vulnerable to malicious attacks [11]. Blockchain is a decentralized technology, by deploying cloud computing on the alliance chain, users participating in cloud computing need to be authenticated [12], can effectively prevent cloud computing from being attacked by malicious users.

B. Privacy of data

Cloud data is usually stored in plain text, which seriously affects the privacy of data. Besides, some attacks caused by combined with blockchain also bring privacy issues [13]. Federated learning (FL), as an emerging artificial intelligence basic technology, can ensure efficient machine learning with protecting the privacy of terminal data and personal data [14]. So the combination of FL and cloud computing could solve privacy of data.

C. Real-time data processing

With the popularity of smart wearable devices [15], a lot of data needs to be processed and feedback in real time [16]. Due to the delay in the cloud computing data transmission process, it cannot be used for real-time data processing [17], which is the third issue faced by cloud computing [18]. For example, cars need to process the data generated by the surrounding environment in real-time to form the instructions for the driving process of

*Correspondence: ylchen3@gzu.edu.cn

[2] Blockchain Laboratory of Agricultural Vegetables Weifang Key Laboratory of Blockchain on Agricultural Vegetables, Weifang University of Science and Technology, Weifang, China
Full list of author information is available at the end of the article

the car [19]. Edge computing, puts more emphasis on the edge, has more real-time and faster data processing capabilities. Data processing is also faster due to reduced intermediate transfers [20]. The combination of cloud computing and edge computing solves the issue of data transmission delay in cloud computing and makes data processing more accurate.

D. Multi-cloud computing

Multi-cloud computing is a techology which uses two or more cloud servicer providers(CSPs) to satisfy the needed of all users [21], but trust and security have prevented businesses from fully accepting cloud platforms [22]. Multi-signature, a solution of trust and security in issues with multi-user participation, can be used in multi-cloud. This paper mainly improves the multi-signature technology used in cloud computing, and focusing on its security and efficiency.

Related work

The original multi-signature scheme uses a certificate-based public key cryptosystem [23], that is, when using a public key, it is necessary to verify the validity of public key certificate(CA) before using it to verify signatures or encrypt data [24]. The certificateless multi-signature scheme reduces the computation overhead and storage cost, so is more widely used than the certificate-based multi-signature scheme [25, 26]. In 2018, Yanai et al. [27] proposed a three-round interactive multi-signature party constructed by using global hashing, and reduced the security of the scheme to the The Computational Diffie-Hellman(CDH) problem in bilinear groups. In the same year, Maxwell et al. [28] proposed a new Schnorr multi-signature scheme. The signature process of this scheme only requires two rounds of interaction, and the security is reduced to the discrete logarithmic difficulty. In 2019, Drijvers et al. [29] analyzed the multi-signature scheme of the two-round interaction, pointed out that the existing scheme has subtle defects in the security proof, and proposed a variant of the BCJ scheme [30] mBCJ scheme, the security of the proposed scheme is reduced to the assumption of discrete logarithmic difficulty in random oracle model. But the development of quantum computing poses a serious threat to the public key cryptosystem [31, 32], and also has an impact on the multi-signature scheme constructed which based on the public key cryptosystem. Therefore, how to construct a new public key cryptosystem to defend against quantum computer attacks has become a research hotspot in cryptography. At present, the effective methods for quantum computer attacks mainly include code-based encryption, lattice-based encryption, multivariate quadratic equation-based encryption. encryption and hash-based encryption [33]. In 2020, Kansal et al. [34] proposed the first lattice-based multi-signature scheme, which supports signature compression and public key aggregation. In 2021, they improved the scheme of [34], while ensuring the security of the scheme, the new scheme reduces its communication and storage overhead [35]. In 2021, Yu et al. [36] proposed the first multi-signature scheme based on MPKC. The security of MPKC relies on solving quadratic polynomial equations over finite fields.

For these issues, this paper proposes a certificateless designated verifier multi-signature scheme based on MPKC(MPKC-DVMS), and gives the formalized definition and security model of the scheme. Compared with the scheme [36], this scheme reduces the number of signature participants and improves the computational efficiency. In addition, this paper proves that the scheme is existential unforgeability against adaptive chosen-message attacks in random oracle model. Finally, it is found that the proposed scheme can hide the signature source and protect user privacy. The contributions of this paper are as follow.

1 We give the application of MPKC-DVMS in multi-clouds and build the formalized definition and security model of the proposed scheme, and prove that the scheme is existential unforgeability against adaptive chosen-message attacks in random oracle model.
2 This paper proposes a multi-signature scheme based on MPKC. This proposed scheme does not need to compute bilinear pairing, reduces the calculation steps and improves the calculation efficiency, which makes the signature more efficiency in multi-cloud.
3 We add a designated verifier, who can generate signatures that are indistinguishable from n signature participants, to hidding signature source to achieve privacy protection

The rest of the paper is organized as follows. Section 2 gives the related work of this paper. Section 3 gives some preliminaries. Section 4 introduces the details of the proposed scheme. Section 5 provides some experimental results and evaluation analysis of our secheme. Finally, Section 6 concludes the paper and gives the future work.

Preliminary

This section mainly introduces some mathematical knowledge and theorems used in the proposed scheme.

Finite field

Let k be a set of non-empty elements, if two operations are defined in k: addition and multiplication, and the following conditions are satisfied [37]:

1 k constitutes an Abel group with respect to addition, and its addition identity element is denoted as 0.
2 All pairs of multiplications in k form an Abel group (except zero elements), and the multiplication identity element is denoted as 1.
3 Addition and multiplication have the following distributive laws:

$$ab + c = ab + ac \tag{1}$$

$$b + ca = ba + ca \tag{2}$$

Then k is a field. If the field k contains only a finite number of elements, then the field k is called a finite field, also known as Galois (Galois field). where q is the number of elements in the field. The number of elements in the field is called the order of the finite field. A finite field of order q, usually expressed as $GF(q)$ or Fq.

Multivariate polynomial equations in finite fields

Let x_1, x_2, x_3, ..., x_n be n variables on a finite field k, then a polynomial of n variables on the field k is represented by f_i, the degree is d, and m polynomials form a polynomial group, denoted by F, then [38]:

$$F := (f_1, f_2, f_3, \ldots, f_m) \tag{3}$$

where $f_i := \sum a_j \prod x_j$, $1 \le i \le n$, $1 \le j \le m$. Let y_1, y_2, y_3, ..., y_n be the elements on the finite field k, then the multivariate polynomial equation system has the form:

$$\begin{cases} y_1 = f_1(x_1, \ldots, x_n) \\ y_2 = f_2(x_1, \ldots, x_n) \\ \cdots \\ y_m = f_m(x_1, \ldots, x_n) \end{cases} \tag{4}$$

Multivarivate quadratic problem (MQ Problem)

The MQ problem refers to solving a system of quadratic polynomial equations in the following field $k = GF(q)$ [39]:

$$\begin{cases} f_1(x_1, \ldots, x_n) = 0 \\ f_2(x_1, \ldots, x_n) = 0 \\ \cdots \\ f_m(x_1, \ldots, x_n) = 0 \end{cases} \tag{5}$$

where f_i is the polynomial equation on the domain k, which defines f_i in the system of quadratic multivariate polynomial equations of the same formula. The MQ problem has been shown to be NP-hard, even for the smallest field $k = GF(2)$. Therefore, the MQ problem has become an important tool for constructing public key cryptosystems over finite fields.

Isomorphism of polynomials problem (IP Problem)

Let P, Q be a multivariate system of two random n-element g equations on a finite field k, and P and Q are isomorphic, then there are $P = T \circ Q \circ V$, where T and V are respectively denoted as two reversible affine transformations on k^n, the (T, V) problem of finding isomorphism from $P \sim Q$ is called an IP problem, this is a polynomial isomorphism problem [40].

Affine transformation

Affine transformation, also known as affine mapping, means that in geometry, a vector space undergoes a linear transformation followed by a translation to transform into another vector space.

Definition 1

The order of every finite field must be a power of a prime number.

Definition 2

Let k be a finite field, and k^n be an n-dimensional isomorphic vector space over the finite field k, which is z linear polynomials over k such that:

$$L_1(X), L_2(X), \ldots, L_z(X) = L_1(L_2(\ldots L_z(X))) \tag{6}$$

A designated verifier multi-signature scheme in multi-clouds

In this section, we first give the application of MPKC-DVMS in multi-clouds, and then bulid the formalized Definition and security model of MPKC-DVMS, Finally, we describe the MPKC-DVMS in detail.

Application of MPKC-DVMS in multi-clouds

The application of MPKC-DVMS in multi-clouds is showing in Fig. 1, the entire procedure is comprised of the following phases.

As the Fig. 1 shows, user submits tasks to multiple CSPs for computing. Then CSPs sign the result and sends it to other clouds after computing, if the number of signers of the result is 2/3 of the total of CSPs or over, the result is considered to be correct and can be returned to the user, and the CSPs receive the corresponding remuneration, else it cannot be sent to the user.

In order to prevent the CSPs from maliciously delaying the calculation, the CSPs should promise a time commitment before submitting the task, the CSPs cannot get the

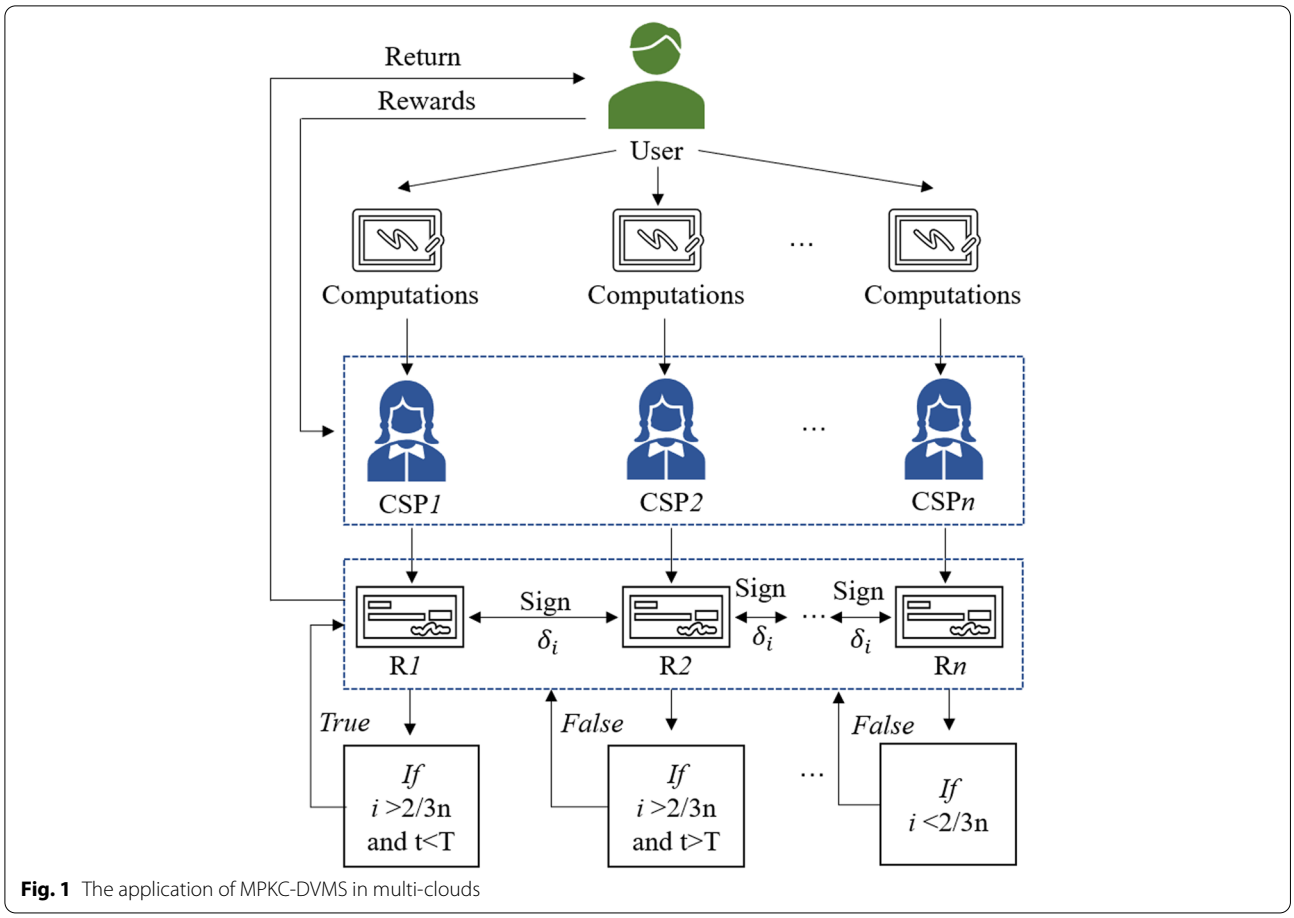

Fig. 1 The application of MPKC-DVMS in multi-clouds

paid of commitment if the timeout and cannot participate in the subsequent task.

Formalized definition and security model

Formalized definition

The participating entities of the MPKC-DVMS scheme include a secret key generation center (KGC), v signers ID_i where i=0...$v-1$, and a designated verifier ID_v. This scheme consists of the following five algorithms.

1 Setup: KGC (Key Generate Center) inputs the security parameter K, and outputs the system master key S and system params.
2 Partial-key-extract: KCG inputs params and S, outputs part of the public key Pksub and part of the private key S_{sub} of the system, and sends Pk_{sub} and S_{sub} to the signing participant through a secure channel.
3 User-key-generate: The user inputs identity ID_i, reversible affine transformations L_{1i}, L_{2i}, some public and private keys Pk_{sub}, S_{sub}, and outputs the user's public and private keys Pk_i, Sk_i.
4 Sign: ID_i input params, message m to be signed, identity set of v signers $ID_{set} = ID_0, ID_1, ..., ID_{v-1}$, signer's private key Sk_i and the identity and identity of the specified verifier The public key (ID_v, Pk_v) outputs a multi-signature on m.
5 Simulation: Specify the verifier input, params, the public key Pk_i of all signing participants, specify the verifier's identity and private key (ID_v, Pk_v), determine whether the signature is valid, and if it is valid, generate an indistinguishable signature.

Security model

In this scheme, there are two adversaries A_1 and A_2 with different attack capabilities. The first type of adversary, A_1, simulates an external adversary, a malicious user. A_1 holds the signer's secret value (reversible affine transformation) and can arbitrarily replace the user's public key, but A_1 does not know the system master key and some private keys. The second type of

adversary, A_2, simulates a malicious but passive KGC. Mastering the system master key can obtain part of the user's private key, but does not know the user's secret value and cannot replace the user's public key. The scheme in this paper simulates the security model of this scheme through the Game between the challenger C and the adversary A (A_1 or A_2).

Setup: C runs the setup algorithm, generates the system master key S and public parameters params, secretly saves S and sends the params to the adversary A. If the adversary is of type A_2, send the params and S to A_2.

Query: A can adaptively query the following oracles:

1 Hash query: When the adversary A asks any Hash function, C outputs the corresponding Hash value to the adversary A.
2 Public-key query: When A inputs ID_i, if ID_i has been created, C returns the corresponding public key to A. Otherwise, C runs the Partial-key-extract algorithm and the User-key-generate algorithm to generate part of Pk_{sub}/S_{sub} and Pk_i/Sk_i. At this time, the ID_i is said to be created by the user. Finally, C returns Pk_i of ID_i to A.
3 Public-key-replace query: A enters ID_i for public key replacement query, and C replaces public key(this oracle is only for A_1-type adversaries).
4 Secret-Value-Extract: When A asks the secret value of ID_i (invertible affine transformation), C returns the corresponding secret value (L_{1i}, L_{2i}) to A. If ID_i's public key has been replaced, C outputs *NULL*.
5 Partial-key-extract query: When A asks for the partial private key of the system, C inputs ID_i and returns the partial private key Ssub to A (this oracle is only for A_1-type adversaries).
6 Sign query: A input message, signer/specified verifier identity (ID_i, ID_v), public key (Pk_i, Pk_v), signer identity set ID_{set}=$ID_0, ID_1, \ldots, ID_{v-1}$, and partial signatures, C runs the Sign algorithm to generate multiple signatures and returns them to A.
7 Forgery: Adversary A has forged a multi-signature on message m^* about $ID_{set}{}^*$=$ID_0{}^*, ID_1{}^*, \ldots, ID_{v-1}{}^*$ with a valid designated verifier, and satisfy:

   (a) the A_1 adversary did not ask the Partial-key-extract oracle.
   (b) the A_2 adversary did not ask the Secret-Value-Extract oracle.

If there is no probabilistic polynomial-time adversary A that can win with a non-negligible advantage in the above Game, then the proposed scheme proposed in this paper is existential unforgeability against adaptive chosen-message attacks.

Details of MPKC-DVMS

Setup

KCG randomly selects the security parameter K to generate a finite field $k = GF(q)$, where q is the order of the finite field, p and l are large prime numbers. Choose two positive integers m and n, where m is the number of multivariable equations and n is the number of variables. Selected as a cryptographic hash function. Randomly select a central map F which is an easy-to-invert nonlinear mapping from k^n to k^m, and randomly select two reversible affine transformations I and J, where I is the reversible affine transformation from k^n to k^n, and J is the reversible affine transformation from k^m to k^m . Compute $\overline{F} = I \circ F \circ J$ as the system public key. The system generates params=$\{k, q, p, l, m, n, H, \overline{F}\}$ and secretly stores the system master key $S = \{I, F, J\}$.

Partial key-extract

KGC randomly selects two reversible affine transformations, L_1 and L_2, calculation $Pk_{sub} = L_2 \circ \overline{F} \circ L_1$ as part of the public key of the system and part of the private key of the system $S_{sub} = \{I \circ L_2, F, J \circ L_1\}$. KGC sends part of the public and private keys to the signing participants through a secure channel.

User key-generate

For the convenience of description, the identity of the signing participant in this scheme is expressed as $ID_i \in 01^*$. Among them, the specified verifier identity is represented as ID_v. The signing participant randomly selects two reversible affine transformations L_{1i}, L_{2i}, of which $L_{1i} \in k^n \to k^n$, $L_{2i} \in k^m \to k^m$. Calculate $Pk_i = L_{2i} \circ Pk_{sub} \circ L_{1i}$ as the public key and $Sk_i = \{L_{2i} \circ L_2 \circ I, F, L_{1i} \circ L_1 \circ J\}$ as the private key.

Sign

User needs to perform the following steps to sign the message.

Step 1

1 Calculate $h_0 = H(M||ID_0||ID_v)$.
2 Calculate $\sigma_0 = L_{20}^{-1} \circ L_2^{-1} \circ I^{-1} \circ (F^{-1} \circ (L_{10}^{-1} \circ L_1^{-1} \circ J^{-1}(h_0)))$.
3 To output the partial signature σ_i, ID_0 sends σ_i, message hash, identity set $ID_{set} = \{ID_0\}$ and the specified verifier identity ID_v to the closest ID_1.

Step 2

After received a partial signature, ID_i verifies whether $h_i' = L_{2i} \circ Pk_{sub} \circ L_{1i}(\sigma_i)$=$h_i$. If not, ID_i rejected. If succeeds, the following steps are performed:

1 Calculate $h_i = H(M||ID_i||ID_v)$.
2 Calculate $\sigma_i = L_{2i}^{-1} \circ L_2^{-1} \circ I^{-1} \circ (F^{-1} \circ (L_{1i}^{-1} \circ L_1^{-1} \circ J^{-1}(h_i)))$.

3 To output the partial signature σ_i, ID_i sends the σ_i, message hash, identity set $ID_{set} = \{ID_0, ID_1, \ldots, ID_i, \ldots, ID_{v-1}\}$ and the specified verifier identity ID_v to ID_v.

Step 3

ID_v verifies whether $h_i' = L_{2i-1} \circ Pk_{sub} \circ L_{1i-1}(\sigma_i)$ is equal to h_i after receiving the partial signature. If the verification fails, the signature is rejected. If the verification succeeds, the following steps are performed.

1 Calculate $h_v = H(M||ID_v||ID_v)$.
2 Calculate $\sigma_v = L_{2v}^{-1} \circ L_2^{-1} \circ I^{-1} \circ (F^{-1} \circ (L_{1v}^{-1} \circ L_1^{-1} \circ J^{-1}(h_v)))$.
3 calculate $\sigma = \prod_{i=0}^{v} \sigma_i$.

Simulation

After the signature is completed, ID_v performs the following steps.

1 Verify $\prod_{i=0}^{v} h_i = \prod_{i=0}^{v} L_{2i} \circ L_2 \circ I \circ F \circ L_{1i} \circ L_1 \circ J(\sigma_i)$.
2 If the verification is successful, ID_v generates a signature σ that is indistinguishable from $\sigma' = L_{2i}^{-1} \circ L_2^{-1} \circ I^{-1} \circ F^{-1} \circ L_{1i}^{-1} \circ L_1^{-1} \circ J^{-1}(h_i)$ and publishes it.

Analysis

This section verifies the usability of the proposed scheme through correctness analysis, security analysis and efficiency analysis.

Correctness analysis

Theorem 1 *This scheme is correct.*

Proof

$$
\begin{aligned}
h'' &= (\prod_{i=0}^{v} \sigma_i) \\
&= \prod_{i=0}^{v} Pk_i \left(\prod_{i}^{v} \sigma_i \right) \\
&= \prod_{i=0}^{v} Pk_i \left(\prod_{i}^{v} Sk_i(h_v) \right) \\
&= \left[\prod_{i=0}^{v} L_{2i} \circ L_2 \circ I \circ F \circ L_{1i} \circ L_1 \circ J \right] w = h'
\end{aligned}
$$

where, $w = \left(\prod_{i}^{v} L_{2i}^{-1} \circ L_2^{-1} \circ I^{-1} \circ (F^{-1} \circ (L_{1i}^{-1} \circ L_1^{-1} \circ J^{-1}(h_v))) \right)$.

It is obvious that $h'' = h'$, so it can be deduced that the proposed scheme is correct.

Security analysis

existential unforgeability

In this paper, the security of the proposed scheme is reduced to the security of the message signature by any signer ID_i The specific proof is as follow.

Theorem 2 *Assuming that the MQ and IP problems are difficult, the proposed scheme proposed in this paper is existentially unforgeable to A_1 class adversaries. In random oracle model, it is assumed that A_1 breaks the proposed scheme in this paper with an advantage ε within the probability polynomial time t, and the maximum times A_1 queries Hash, Public-key, Partial-private-key, and Sign is q_h, q_c, q_p and q_s, there is an algorithm C that A_1 can solve the MQ problem with the advantage of $\varepsilon' \geq \varepsilon(\frac{v}{q_c q_h})(1 - \frac{2}{q_c})^{q_p}(1 - \frac{1}{q_c q_h})^{q_s}$ within time $t' < t + 6q_c t_c + v(q_h + q_s)t_h + 10q_s(t_s + t_{inv})$, where t_c represents the time to calculate a mapping synthesis on the finite field, t_{inv} represents the time to obtain an inverse on the finite field, and t_s represents the time to calculate the first-order polynomial on the finite field.*

Proof

It is assumed that A_1 can attack the signature scheme of this paper with a non-negligible probability, thereby solving the MQ and IP problems. This scheme involves v signing users $ID_0, ID_2, \ldots, ID_{v-1}$, and the designated verifier ID_v. This scheme assumes that all users except user $ID_I \in ID_0, ID_2, \ldots, ID_{v-1}$ are bribed by A_1. Let C be the challenger, given any instance of the MQ problem over a finite field $k = GF(q)$, $Y' = (y_1', \ldots, y_m') \in k = GF(q)$, the ultimate purpose of C is to solve the polynomial equation system, that is, to find $\bar{F}(x_1, \ldots x_n) = Y'$.

Setup: Challenger C builds a system and returns the system parameters $params = (k, q, p, l, m, n, H, \bar{F})$ to adversary A_1. The C maintenance lists H^{list} and K^{list} represent Hash query, Public-key query, Partial-Private-key query and Sign query respectively. The list is initially empty. An adversary can adaptively interrogate the following oracles.

- Hash query: The challenger maintains the list $H^{\text{list}}(M, ID_i, ID_v, h_i)$. When receiving the $H(M||ID_i||ID_v)$ query from A_1, it first searches the H^{list}. If it exists, return it directly to A_1, if not, randomly select $h_i \in k^n$ and return it to A_1, and add the record to the H^{list}.
- Public-key query: C maintains a list $K^{\text{list}}(ID_i, L_{1i}, L_{2i}, Pk_i)$, when C receives public-key query, finds K^{list} exists in the list. If it exists, it returns directly to A_1, if not, C executes as follows.

– If $ID_i \neq ID_I$, ID_v, C randomly selects four reversible affine transformations L_1, L_2, L_{1i}, $L_{2i}k$, and calculates the partial public key $Pk_{\text{sub}} = L_2 \circ \overline{F} \circ L_1$, partial private key $S_{\text{sub}} = I \circ L_2, F, J \circ L_1$, user's public key $Pk_i = L_{2i} \circ Pk_{sub} \circ L_1$ is returned to A_1, and $(ID_i, L_1, L_2, L_{1i}, L_{2i}, Pk_{sub}, Sk_{sub}, Pk_i)$ are stored in the K^{list}.
– If $ID_i = ID_I$, C randomly selects four invertible affine transformations L_1, L_2, L_{1i}, $L_{2i}k$, and calculates $Pk_{\text{sub}} = \perp, S_{\text{sub}} = \perp, Pk_i = L_{2i} \circ Pk_{sub} \circ L_{1i}$, return Pk_i to A_1, and $(ID_i, L_1, L_2, L_{1i}, L_{2i}, Pk_{sub}, Sk_{sub}, Pk_i)$ are stored in the K^{list}.
– If $ID_i = ID_I$, C randomly selects four invertible affine transformations L_1, L_2, L_{1v}, $L_{2v}k$ and calculates $Pk_{\text{sub}} = \perp$, $S_{\text{sub}} = \perp$, $Pk_v = L_{2v} \circ Pk_{sub} \circ L_{1v}$, Pk_i to A_1, and $(ID_i, L_1, L_2, L_{1v}, L_{2v}, Pk_{sub}, Sk_{sub}, Pk_i)$ are stored in the K^{list}.

- Public-key-replace query: When C receives the public-key-replac query, C replaces Pk_i in the original list with Pk_i', makes $L_{1i} = L_{2i} = \perp$ return to A_1, and Store $\left(ID_i, Pk_i'\right)$ into the K^{list}.
- Secret-value query: When C receives the secret-value query, checking whether the item (ID_I, L_{1i}, L_{2i}) exists in the K^{list}, if $L_{1i} = L_{2i} = \perp$, the public key has been replaced, and C ends the query. Otherwise, return (L_{1i}, L_{1i}) to A_1.
- Partial-private-key query: When C receives the partial-private-key query, checking the K^{list}, if $ID_i \neq ID_I$, ID_v, return Sk_{sub}, otherwise, stop the simulation, output *failure*.
- Sign query: A_1 inputs the message $M \in 0, 1^*$, the signer/specified verifier identity (ID_i, ID_j), the public key (Pk_i, Pk_j), the signer identity set ID_{set}, and C performs the following operations.

  – If $ID_i \neq ID_I$, ID_v, C brings up the list $K^{\text{list}}(ID_i, L_1, L_2, L_{1i}, L_{2i}, Pk_{sub}, Sk_{sub}, Pk_i)$ and the list $H^{\text{list}}(M, ID_i, ID_v, h)$, calculate $h = H\left(M||\text{ID}_i||ID_v\right)$, $\sigma_i = L_{2i}^{-1} \circ L_2^{-1} \circ I^{-1} \circ (F^{-1} \circ (L_{1i}^{-1} \circ L_1^{-1} \circ J^{-1}(h)))$ and returns the signature σ_i to A_1.
  – If $ID_i = ID_I$ and ID_jâ‰ ID_v, C brings up the list $K^{\text{list}}(ID_i, L_1, L_2, L_{1i}, L_{2i}, Pk_{sub}, Sk_{sub}, Pk_i)$ and the list $H^{\text{list}}(M, ID_i, ID_v, h)$, calculate $h = H\left(M||\text{ID}_i||ID_v\right)$, $\sigma_i = L_{2i}^{-1} \circ L_2^{-1} \circ I^{-1} \circ (F^{-1} \circ (L_{1i}^{-1} \circ L_1^{-1} \circ J^{-1}(h)))$ and returns the signature σ_i to A_1.
  – If $ID_i = ID_I$, $ID_j = ID_v$, C stops the game and outputs *failure*.

- Forgery: A_1 outputs the forged signature σ^* for M^*, if $ID_I \notin \left\{ID_1^*, ID_2^*, \ldots, ID_{v-1}^*\right\}$,and $ID_j^* \neq ID_v$ then C terminates the game. Otherwise, C bring the list $K^{\text{list}}(ID_i^*, L_1^*, L_2^*, L_{1i}^*, L_{2i}^*, Pk_{sub}^*, Sk_{sub}^*, Pk_i^*)$, $H^{\text{list}}(M^*, ID_i^*, ID_v, h^*)$, because σ is valid, so there is $\sigma^* = L_{20}^{-1} \circ L_2^{-1} \circ I^{-1} \circ (F^{-1} \circ (L_{10}^{-1} \circ L_1^{-1} \circ J^{-1}(h)))(\prod_{i=2}^{v} L_{2i}^{-1^*} \circ L_2^{-1^*} \circ I^{-1^*} \circ (F^{-1^*} \circ (L_{1i}^{-1^*} \circ L_1^{-1^*} \circ J^{-1^*}(h^*))))$, so that $h^* = \prod_{i=1}^{v} L_{2i}^* \circ Pk_{sub}^* \circ L_{1i}^*$, that is, C can solve the MQ problem, but the MQ problem is difficult at present, so the proposed scheme is existentially unforgeable for A_1-type adversaries.

Then calculating the probability of C success. The four events represented by E_1, E_2, E_3 and E_4 are as follows.

E_1: C answer partial-private-key query successfuly.
E_2: C answer sign query successfuly.
E_3: A_1 successfully forged a multi-signature on the message, in which the identities of v signers are $D_1^*, D_2^*, , D_{v-1}^*$, specifying the identity of the verifier is D_v^*.
E_4: There is $ID_v^* = ID_v$ and $ID_I \in \left\{ID_1^*, ID_2^*, \ldots, ID_{v-1}^*\right\}$ when E_3 occurs.

Assuming that the probabilities of events E_1, E_2, E_3, and E_4 are $P(E_1)$, $P(E_2)$, $P(E_3)$ and $P(E_4)$, where

$$P(E_1) \geq (1 - \frac{2}{q_c})^{q_p}$$
$$P(E_2|E_1) \geq (1 - \frac{1}{q_c q_h})^{q_s}$$
$$P(E_3 \mid E_2 \wedge E_1) \geq \varepsilon$$
$$P(E_4|E_3 \wedge E_2 \wedge E_1) \geq \frac{v}{q_c q_h}$$

So the probability of C success is:
$P\left(E_1 \wedge E_2 \wedge E_3 \wedge E_4\right) = P(E_1)P(E_2|E_1)P(E_3|E_2 \wedge E_1)P(E_4|E_3 \wedge E_2 \wedge E_1) \geq \varepsilon(\frac{v}{q_c q_h})(1 - \frac{2}{q_c})^{q_p}(1 - \frac{1}{q_c q_h})^{q_s}$

Theorem 3 *Assuming that the MQ and IP problems are difficult, the proposed scheme proposed in this paper is existentially unforgeable to A_2 class adversaries. In random oracle model, it is assumed that A_2 breaks the proposed scheme in this paper with an advantage ε within the probability polynomial time t, and the maximum times of A_2 queries Hash, Public-key, Partial-private-key, and Sign is q_h, q_c, q_p and q_s, there is an algorithm C that A_2 can solve the MQ problem with the advantage of $\varepsilon' \geq \varepsilon(\frac{v}{q_c q_h})(1 - \frac{2}{q_c})^{q_{se}}(1 - \frac{1}{q_c q_h})$ within time $t' < t + 6q_c t_c + v(q_h + q_s)t_h + 10q_s(t_s + t_{inv})$, where t_c represents the time to calculate a mapping synthesis on the finite field, t_{inv} represents the time to obtain an inverse on the finite field, and t_s represents the time to calculate the first-order polynomial on the finite field.*

Proof

Let C be the challenger, given any instance of the IP problem on the finite field $k = GF(q)\bar{F}(x_1, \ldots x_n) = Y'$, *the ultimate goal of C is to find The equation system Q that is isomorphic to the polynomial equation system is to find* $P = T \circ Q \circ V$, *where T and V are the reversible affine transformations on the finite field.*

Setup: Challenger *C* builds a system and returns the system parameters $params = \left(k, q, p, l, m, n, H, \bar{F}\right)$ to adversary A_2. The C maintenance lists H^{list} and K^{list} represent Hash query, Public-key query, Partial-Private-key query and Sign query respectively. The list is initially empty. An adversary can adaptively interrogate the following oracles.

- Hash query: The challenger maintains the list $H^{\text{list}}(M, ID_i, ID_v, h_i)$. When receiving the $H(M||ID_i||ID_v)$ query from A_2, it first searches the H^{list}. If it exists, return it directly to A_2, if not, randomly select $h_i \in k^n$ back to A_2 and add the record to the H^{list}.
- Public-key query: *C* maintains a list $K^{\text{list}}(ID_i, L_{1i}, L_{2i}, Pk_i)$, when *C* receives public-key query, finds K^{list} exists in the list. If it exists, it returns directly to A_2, if not, *C* randomly selects two reversible affine transformations $L_1, L2 \in k$, and calculates $Pk_{\text{sub}} = L_2 \circ \overline{F} \circ L_1$, $S_{\text{sub}} = \{I\ circL_2, F, J \circ L_1\}$, and then calculate the public key of ID_i as follows.
  - If $ID_i \neq ID_I$, ID_v, *C* randomly selects two reversible affine transformations L_{1i}, $L_{2i}k$, and calculate the user's public key $Pk_i = L_{2i} \circ Pk_{sub} \circ L_1 i$ and return it to A_2, then stored $(ID_i, L_1, L_2, L_{1i}, L_{2i}, Pk_{sub}, Pk_i)$ in the K^{list}.
  - If $ID_i = ID_I$, *C* sets $Pk_i = T \circ Pk_{sub} \circ V$, $L_{1i}, L_{2i} = \perp$ and return Pk_i to A_2, then stored $(ID_i, L_1, L_2, L_{1i}, L_{2i}, Pk_{sub}, Pk_i)$ in the K^{list}.
  - If $ID_i = ID_I$, *C* randomly selects two reversible affine transformations L_{1v}, $L_{2v}k$, and calculate $Pk_v = L_{2v} \circ Pk_{sub} \circ L_1 v$ and return it to A_2, then stored $(ID_i, L_1, L_2, L_{1v}, L_{2v}, Pk_{sub}, Pk_v)$ in the K^{list}.
- Secret-value query: If $ID_i \neq ID_I$, ID_v, *C* browses K^{list} and return (L_{1i}, L_{2i}) to A_2 when receives the secret-value query. Otherwise, return *failure*.
- Sign query: A_2 inputs the message $M \in 0, 1^*$, the signer/specified verifier identity (ID_i, ID_j), the public key (Pk_i, Pk_j), the signer identity set ID_{set}, *C* browses K^{list} and H^{list}, computing $h_i = H(M||ID_i||ID_v)$ $\sigma_i, = L_{2i}^{-1} \circ L_2^{-1} \circ I^{-1} \circ (F^{-1} \circ (L_{1i}^{-1} \circ L_1^{-1} \circ J^{-1}(h_i)))$,then return σ_i to A_2.
- Forgery: A_2 outputs the forged signature σ^* for M^*, if $ID_I \notin \left\{ID_1^*, ID_2^*, \ldots, ID_{v-1}^*\right\}$,and $ID_j^* \neq ID_v$ then *C* terminates the game. Otherwise, *C* bring K^{list}, H^{list} to query log $(ID_1^*, L_1^*, L_2^*, L_{11}^*, L_{21}^*, Pk_{sub}^*, Sk_{sub}^*, Pk_1^*)$ and (M^*, ID_1^*, ID_v, h^*), If the record does not exist in the list, *C* terminates the game and outputs failure. Otherwise, A2 is forged successfully. It means *C* can select a record containing the correct (T, V) from K^{list} with the probability of $\frac{1}{q_c q_h}$ to solve the IP problem, but IP problem is still difficult, so the proposed scheme presented in this paper is existentially unforgeable to A_2-type adversaries .

Then calculating the probability of C success. The four events represented by E_1, E_2, E_3 are as follows.

E_1: C answer Secret-value query successfuly.
E_2: A_2 successfully forged a multi-signature on the message, in which the identities of v signers are $D_1^*, D_2^*, , D_{v-1}^*$, specifying the identity of the verifier is D_v^*.
E_3: There is $ID_v^* = ID_v$ and $ID_I \in \left\{ID_1^*, ID_2^*, \ldots, ID_{v-1}^*\right\}$ when E_2 occurs.

Assuming that the probabilities of events E_1, E_2 and E_3 are $P(E_1)$, $P(E_2)$ and $P(E_3)$, where

$$P(E_1) \geq (1 - \frac{2}{q_c})^{q_{se}}$$

$$P(E_2|E_1) \geq \varepsilon(1 - \frac{1}{q_c q_h})$$

$$P(E_3|E_2 \wedge E_1) \geq \frac{v}{q_c q_h}$$

So the probability of *C* success is:
$P\left(E_1 \wedge E_2 \wedge E_3\right) = P(E_1)P(E_2|E_1)P(E_3|E_2 \wedge E_1) \geq \varepsilon(\frac{v}{q_c q_h})$ $(1 - \frac{2}{q_c})^{q_{se}}(1 - \frac{1}{q_c q_h})$

Hidding signature source

Theorem 4 *Assuming that there is an adversary* $T(T_1$ *or* $T_2)$ *that cannot distinguish the signature* σ *is generated by* ID_i *or* ID_v, *this scheme can hide the source of signature.*

Proof

1 Assuming that the adversary T_1 has the ID_i of all signers and the public and private keys of v signers. For a message signature σ, if T_1 wants to infer the true signer, it needs to compute $\sigma_i = L_{2i}^{-1} \circ L_2^{-1} \circ I^{-1} \circ (F^{-1} \circ (L_{1i}^{-1} \circ L_1^{-1} \circ J^{-1}(h_i)))$

and $h_i = H(M||ID_i||ID_v)$, to obtain the identity of a signer, it means, T_1 needs to solve the MQ problem and the Hash problem in probabilistic polynomial time, but the above two problems are still difficult, so the T_1-type adversary cannot infer the signer.

2 Assuming that the adversary T_2 has the private key Sk_i of all signers and the public key Pk_v of the designated verifier in addition to the attack capability of T_1. For a message signature σ, T_2 can use the ID_i of all signers to calculate $h_i = \prod_{i=0}^{v} H(M||ID_i||ID_v)$, then use all signers' ID_i, Sk_i calculates the message signature$\sigma \prod_{i=0}^{v} L_{2i}^{-1} \circ L_2^{-1} \circ I^{-1} \circ (F^{-1} \circ (L_{1i}^{-1} \circ L_1^{-1} \circ J^{-1}(h_i)))$, and use the specified verifier's public key to calculate $\sigma_v = L_{2i}^{-1} \circ L_2^{-1} \circ I^{-1} \circ F^{-1} \circ L_{1i}^{-1} \circ L_1^{-1} \circ J^{-1}(h_v)$, where $h_v = \prod_{i=0}^{v} H(M||ID_i||ID_v)$, since $\sigma = \sigma_v$, so the T_2-type adversary cannot infer the signer.

To sum up, this scheme can hide the signature source.

Efficiency analysis

Let f denote the calculation of the last multiplication of the finite field k, *map* to denote the last polynomial calculation of the finite field, P_r and S_m to denote a scalar multiplication calculation and a bilinear pair calculation on the group. Assuming that the participating users of the scheme are v, the comparison between proposed scheme and other schemes is shown in Table 1.

As shown in Table 1, bilinear pairing is used in Du et al. scheme and Du scheme for signature operation, so the signature and verification cost are higher than our scheme. In addition, Yu et al. scheme requires centralized verification, but our scheme do not require centralized verification, so the cost of Yu et al. scheme is also higher than MPKC-DVMS.

Furthermore, this paper compares the signature time between the proposed scheme and Yu et al. scheme. As can be seen from Fig. 2, since this paper only needs

Table 1 Performance comparison table

	Signature complexity	Signature cost	Verification complexity	Verification cost	Hide signature source
Du et al. scheme [41]	$O(n)$	$v(Pr + 3Sm)$	$O(n)$	$1Pr + 3vSm$	No
Du scheme [42]	$O(n^2)$	$3Pr$	$O(n^2)$	$vPr + 3Sm$	Yes
Yu et al. scheme [36]	$O(n)$	$v(f + map)$	$O(n)$	vf	No
Our scheme	$O(n)$	$vmap$	$O(n)$	f	Yes

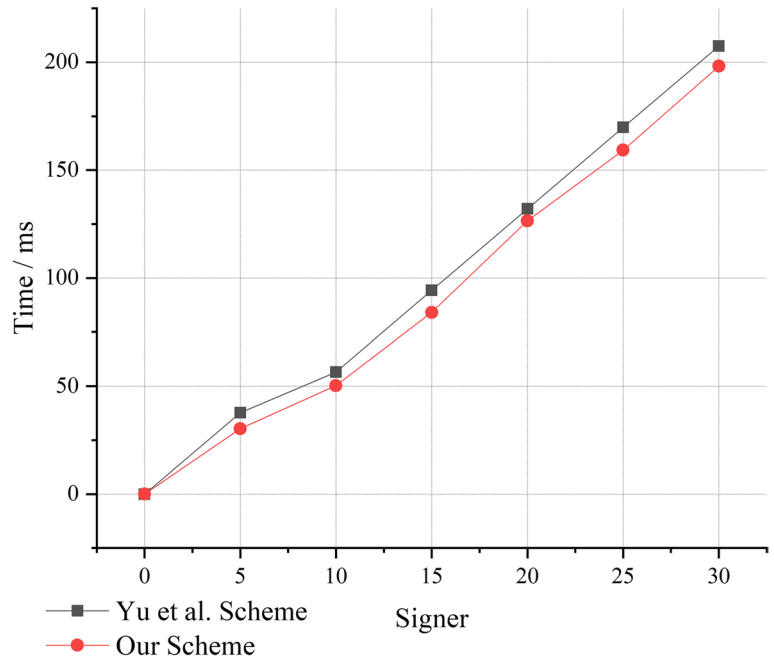

Fig. 2 Signature time comparison

to perform one Hash operation and does not need to perform centralized signature verification, so the signature time of this paper is lower than that of Yu et al. scheme.

Conclusion

Multi-cloud uses multiple CSPs to provide services for users. Due to the increase of participants, multi-cloud is faced with increased signature cost and security issues. In response to the above issues, this paper proposes a MPKC-based multi-signature scheme for designated verifiers , and proves the security in the random oracle model. Morever, the proposed scheme does not need to calculate bilinear pairings, the computational complexity is lower than that traditional multi-signature scheme, so it is more suitable for multi-cloud. In addition, the proposed scheme can hide the signature source, which can protection privacy for users, and improve the security of multi-cloud. The future work is to design a more efficient signature scheme.

Acknowledgements
We are thankful to State Key Laboratory of Public Big Data of Guizhou University for providing an environment for editing manuscripts and experiments.

Authors' Contributions
Chaoyue tan contributed to the conception of the study and bulit the formalized Definition and security model of MPKC-DVMS. Yuling Chen contributed significantly to analysis and manuscript preparation. Yongtang Wu perform the analysis with constructive discussions. Xiaochuan He performed the experiment and Tao Li refined the formalized definition and security model. All authors reviewed the manuscript.

Funding
This work is financially supported by the National Natural Science Foundation of China under Grant No. 61962009. In part by Top Technology Talent Project from Guizhou Education Department (Qian jiao ji [2022]073). And in part by Foundation of Guangxi Key Laboratory of Cryptography and Information Security(GCIS202118).

Availability of data and materials
The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Ethics approval and consent to participate
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]State Key Laboratory of Public Big Data, College of Computer Science and Technology, Guizhou University, Guiyang, China. [2]Blockchain Laboratory of Agricultural Vegetables Weifang Key Laboratory of Blockchain on Agricultural Vegetables, Weifang University of Science and Technology, Weifang, China.

Received: 18 July 2022 Accepted: 18 September 2022
Published online: 01 October 2022

References

1. Zhang Y, Zhang H, Cosmas J, Jawad N, Ali K, Meunier B et al (2020) Internet of radio and light: 5G building network radio and edge architecture. Intell Converged Netw 1(1):37–57. https://doi.org/10.23919/ICN.2020.0002
2. Chen Y, Sun J, Yang Y, Li T, Niu X, Zhou H (2022) PSSPR: a source location privacy protection scheme based on sector phantom routing in WSNs. Int J Intell Syst 37(2):1204–1221
3. Yuan F, Chen S, Liang K, Xu L (2021) Research on the coordination mechanism of traditional Chinese medicine medical record data standardization and characteristic protection under big data environment. Shandong People's Publishing House, Shandong
4. Chen Y, Liu Z, Zhang Y, Wu Y, Chen X, Zhao L (2021) Deep reinforcement learning-based dynamic resource management for mobile edge computing in industrial internet of things. IEEE Trans Ind Inform 17(7):4925–4934
5. Zhang W, Chen X, Jiang J (2020) A multi-objective optimization method of initial virtual machine fault-tolerant placement for star topological data centers of cloud systems. Tsinghua Sci Technol 26(1):95–111
6. Li T, Wang Z, Chen Y, Li C, Jia Y, Yang Y (2021) Is semi-selfish mining available without being detected? Int J Intell Syst
7. Xu J, Li D, Gu W et al (2022) UAV-assisted Task Offloading for IoT in Smart Buildings and Environment via Deep Reinforcement Learning. Build Environ. https://doi.org/10.1016/j.buildenv.2022.109218
8. Wang Y, Li T, Liu M, Li C, Wang H (2022) STSIIML: Study on token shuffling under incomplete information based on machine learning. Int J Intell Syst 1–23. https://doi.org/10.1002/int.23033
9. Chen Y, Gu W, Li K (2022) Dynamic task offloading for Internet of Things in mobile edge computing via deep reinforcement learning. Int J Commun Syst. https://doi.org/10.1002/dac.5154
10. Tan X, Zhang J, Zhang Y, Qin Z, Ding Y, Wang X (2020) A PUF-based and cloud-assisted lightweight authentication for multi-hop body area network. Tsinghua Sci Technol 26(1):36–47
11. Qi L, Lin W, Zhang X, Dou W, Xu X, Chen J (2022) A Correlation Graph based Approach for Personalized and Compatible Web APIs Recommendation in Mobile APP Development. IEEE Trans Knowl Data Eng. https://doi.org/10.1109/TKDE.2022.3168611
12. Li T, Chen Y, Wang Y, Wang Y, Zhao M, Zhu H, et al (2020) Rational Protocols and Attacks in Blockchain System. Secur Commun Netw 2020
13. Li T, Wang Z, Yang G, Cui Y, Chen Y, Yu X (2021) Semi-Selfish Mining based on Hidden Markov Decision Process. Int J Intell Syst 36(7):3596–3612
14. Huang J, Tong Z, Feng Z (2022) Geographical POI recommendation for Internet of Things: A federated learning approach using matrix factorization. Int J Commun Syst. https://doi.org/10.1002/dac.5161
15. Chen Y, Xing H, Ma Z, et al (2022) Cost-Efficient Edge Caching for NOMA-enabled IoT Services. China Commun
16. Li K, Zhao J, Hu J et al (2022) Dynamic Energy Efficient Task Offloading and Resource Allocation for NOMA-enabled IoT in Smart Buildings and Environment. Build Environ. https://doi.org/10.1016/j.buildenv.2022.109513
17. Chen Y, Zhao F, Lu Y, Chen X (2021) Dynamic task offloading for mobile edge computing with hybrid energy supply. Tsinghua Sci Technol. https://doi.org/10.26599/TST.2021.9010050
18. Huang J, Lv B, Wu Y, Chen Y, Shen X (2022) Dynamic admission control and resource allocation for mobile edge computing enabled small cell network. IEEE Trans Veh Technol 71(2):1964–1973
19. Dong J, Wu W, Gao Y, Wang X, Si P (2020) Deep reinforcement learning based worker selection for distributed machine learning enhanced edge intelligence in internet of vehicles. Intell Converged Netw 1(3):234–242
20. Chen Y, Zhao F, Chen X, Wu Y (2022) Efficient multi-vehicle task offloading for mobile edge computing in 6G networks. IEEE Trans Veh Technol 71(5):4584–4595. https://doi.org/10.1109/TVT.2021.3133586
21. Sandhu AK (2021) Big data with cloud computing: Discussions and challenges. Big Data Min Analytics 5(1):32–40
22. Qi L, Hu C, Zhang X, Khosravi MR, Sharma S, Pang S et al (2021) Privacy-aware data fusion and prediction with spatial-temporal context for smart city industrial environment. IEEE Trans Ind Inform 17(6):4159–4167
23. Diffie W, Hellman M (1976) New directions in cryptography. IEEE Trans Inf Theory 22(6):644–654

24. Lu Y, Li J (2019) Constructing pairing-free certificateless public key encryption with keyword search. Front Inf Technol Electron Eng 20(8):1049–1060
25. Itakura K, Nakamura K (1983) A public-key cryptosystem suitable for digital multisignatures. NEC Res Dev 71(71):474–480
26. Hafizul Islam S, Sabzinejad Farash M, Biswas G, Khurram Khan M, Obaidat MS (2017) A pairing-free certificateless digital multisignature scheme using elliptic curve cryptography. Int J Comput Math 94(1):39–55
27. Yanai N (2018) Meeting tight security for multisignatures in the plain public key model. IEICE Trans Fundam Electron Commun Comput Sci 101(9):1484–1493
28. Maxwell G, Poelstra A, Seurin Y, Wuille P (2019) Simple schnorr multi-signatures with applications to bitcoin. Des Codes Crypt 87(9):2139–2164
29. Drijvers M, Edalatnejad K, Ford B, Kiltz E, Loss J, Neven G, et al (2019) On the security of two-round multi-signatures. In: 2019 IEEE Symposium on Security and Privacy (SP) in San Francisco, CA, USA. IEEE, pp 1084–1101
30. Bagherzandi A, Cheon J, Jarecki S (2008) Multisignatures secure under the discrete logarithm assumption and a generalized forking lemma. In: Proceedings of the 15th ACM conference on Computer and communications security in Alexandria Virginia USA. ACM, pp 449–458
31. Jordan S, Liu Y (2018) Quantum Cryptanalysis: Shor, Grover, and Beyond. IEEE Secur Priv 16(5):14–21
32. Chen Y, Dong S, Li T, Wang Y, Zhou H (2021) Dynamic multi-key FHE in asymmetric key setting from LWE. IEEE Trans Infor Forensic Secur 16:5239–5249
33. Li C, Chen X, Chen Y, Hou Y, Li J (2019) A New Lattice-Based Signature Scheme in Post-Quantum Blockchain Network. IEEE Access 7:2026–2033
34. Kansal M, Dutta R (2020) Round optimal secure multisignature schemes from lattice with public key aggregation and signature compression. In: International Conference on Cryptology in Africa, Cairo, Egypt. Springer, pp 281–300
35. Kansal M, Singh AK, Dutta R (2021) Efficient multi-signature scheme using lattice. Comput J 65(9): 2421-2429
36. Yu H, Fu S, Liu Y, Zhang S (2020) Certificateless Broadcast Multisignature Scheme Based on MPKC. IEEE Access 8:12146–12153
37. Chudnovsky DV, Chudnovsky GV (1988) Algebraic complexities and algebraic curves over finite fields. J Complex 4(4):285–316
38. Ding J, Gower JE, Schmidt DS (2006) Zhuang-Zi: A New Algorithm for Solving Multivariate Polynomial Equations over a Finite Field. Cryptology ePrint Archive, paper 2006/038. https://eprint.iacr.org/2006/038. Accessed 3 Dec 2006
39. Wolf C, Preneel B (2005) Taxonomy of Public Key Schemes based on the problem of Multivariate Quadratic equations. IACR Cryptol ePrint Arch 2005:77
40. Patarin J, Goubin L, Courtois N (1998) Improved algorithms for isomorphisms of polynomials. In: International Conference on the Theory and Applications of Cryptographic Techniques in Espoo, Finland. Springer, pp 184–200
41. Du h, Wen q, (2016) Certificateless strong designated verifier multi-signature. J Commun 37(6):20–28
42. Du h, (2016) A Safe and Efficient Ordered Multi-Signature Mechanism for Vehicle Networks. Appl Res Comput 33(10):3105–3108

Publisher's Note