

Intent-Driven Attribute-Based Outsourcing Encryption Scheme

Ke Li¹, Guowei Wu¹ and Jun Shen¹

¹ China Telecom Corporation Limited Research Institute, GuangZhou 510000, China
lik8@chinatelecom.cn

Abstract. With the development of the Internet, the communication between users is increasingly concerned about the protection of private information. Attribute-based encryption is one of the important means to protect private data. It uses attributes as certificates for decryption, preventing private data from being leaked or tampered. However, the traditional attribute-based encryption has problems, such as inflexible encryption process and heavy computing burden for users. We propose an intent-driven attribute-based outsourcing encryption scheme, which integrates user intent parameters into the encryption algorithm to improve the flexibility and reliability of the encryption process. Edge nodes have powerful computing and storage capabilities. We outsource some encryption and decryption operations from users to edge nodes, which helps reduce the computational cost for users or terminals. The hierarchical relationship of attributes can help users quickly match attributes. We construct attributes as attribute trees and determine the decryption permission of users based on the hierarchical relationship between user attributes. Finally, the scheme analysis is provided, including the security proof, performance cost and functional comparison.

Keywords: Intent-Driven, Attribute-Based Encryption, Hierarchical Attributes, Outsourced.

1 Introduction

1.1 A Subsection Sample

As an important medium for data transmission and storage, how to ensure the security of private data in the network is a concern for consumers. Attribute-Based Encryption (ABE) is widely used to protect user data security due to its high efficiency, flexibility and scalability. Among them, Hierarchical Attribute-Based Encryption (HABE) utilizes the path matching relationship of attributes in the attribute tree to encrypt and decrypt files. HABE is more convenient and efficient compared with other encryption schemes. With the continuous development of Internet technology, the needs of users have gradually upgraded from the data confidentiality and reliability to the automation and intelligence of the encryption process. This evolution not only requires the data to be tamper-proof and leak-proof, but also requires defining the data encryption process and selecting the encryption features. Intent driven technology is a new type of network

intelligence technology. We integrate intent-driven method into the existing attribute encryption process and improve them through operations such as intent input, intent translation and intent matching. This helps users filter effective information, improve network system utilization efficiency, and increase the automation and intelligence capabilities of private data management.

The Chair of the ONF NBI Working Group published a draft standard called "Intent: Don't Tell Me What to Do! Tell Me What You Want" [1], which proposes features and concepts for intent-based networks. Subsequently, intent-driven technology and intent-driven network (IDN) were gradually discussed. L.Pang[2] reviewed the research progress of intent-driven network, including basic architecture, key technologies, and technology applications. And L.Pang analyzed the achievements of intent-driven technology from a macro level, pointing out the direction for potential research on IDN. J. Huang[3] and Y. Ouyang[4] combined artificial intelligence with IDN to provide methods of intelligent intent and intelligent management. J. Zhang[5] proposed a Quadruple-based Intent Conflict Resolution (QICR) engine to solve intent conflict in IDN. The QICR has implemented an intent conflict resolution scheme that converts potential conflicting intents to conflict free intents. G. LYU[7] applied intent to the edge computing environment and built a new active ideographic network i-ECAN based on network edge computing capability, which solved the end-to-end interconnection problem of human network cooperation. The above intent-driven researches lacks flexibility and scalability, mainly focusing on intent network, and user intents have not been extend to other scenarios.

Q. Huang[8] proposed a secure and fine-grained data access control scheme with computation outsourcing in fog computing, which reduces the computational cost of data owner encryption, end user decryption, and re-encryption to the number of attributes in the policies. In the HABE scheme from Q. Huang[9], the majority of computing costs were delegated to Cloud Service Providers (CSP) and attribute authorities (AA) were hierarchically managed. H. Peng[10] proposed the ABE scheme in edge computing, outsourcing decryption calculation to edge nodes and using multi-authority (MA) to meet the performance requirements of users' cross-domain access. Q. Leng[11] and K. Huang[12] proposed ABE schemes for outsourcing encryption and outsourcing decryption in cloud environment. In the scheme[12], massive decryption operations are outsourced to near edge servers to reduce the computational cost of decryption.

Based on the above research, the existing ABE schemes follow traditional encryption and decryption operations, but suffer from issues, such as insufficient flexibility in security levels, insufficiently fine-grained access control policies, and inability to customize parameters. Such scheme cannot meet the scalable encryption requirements of users in the digital age. We propose an intent-driven attribute-based outsourcing encryption scheme. We combine intent parameters with the encryption process to represent attributes such as security level, security time and cracking difficulty, in order to enhance user autonomy, system security and management convenience. By utilizing edge nodes for computing outsourcing, complex encryption and decryption calculations are processed, improving computational efficiency and saving resources for users and the central cloud.

2 Preliminaries

2.1 Intent-Driven Network

In order to promote the intelligence and automation of future networks, IDN and Intent-driven technology were proposed as a novel network management framework [13]. According to the research of the standards organization [14][15], the definitions of IDN are different, but the essence is the same, that is, according to the business intents issued by the user, the network drives itself to realize policy configuration and network management. This process does not require manual participation, and can intelligently realize the transformation of the network state and improve the availability and flexibility of the network.

2.2 Bilinear Groups Pairing

Boneh et al. introduced bilinear groups pairing [16]. Let G_0 and G_1 be two multiplicative cyclic groups of prime order p . Let g be a generator of G_0 and e be a bilinear groups pairing. $e: G_0 \times G_0 \rightarrow G_1$, which has the following properties:

Bilinear: All $\mu, v \in G_0$ and all $a, b \in Z_p$ satisfy the equation $e(\mu^a, v^b) = e(\mu, v)^{ab}$.

Non-degeneracy: The pairing does not map all the elements in $G_0 \times G_0$ to the unit of G_1 , that is, there exists $g \in G_0$ such that $e(g, g) \neq 1$.

Computable: Randomly select two elements μ, v , there is an effective algorithm to calculate $e(\mu, v)$.

Note that the pairing $e(g, g)$ is symmetric because $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$.

2.3 Linear secret sharing scheme

The secret sharing scheme is known as the threshold scheme by A. Beimel [17]. A secret is divided into n copies and distributed to n managers (e.g. users or attribute authority). In the (k, n) threshold, the secret shares only meet more than k to restore the original secret. This process can be described by Lagrange interpolation.

Lagrange interpolation:

$$\Delta_{i,S}(x) = \prod_{\vartheta \in S, \vartheta \neq i} \frac{x - \vartheta}{i - \vartheta} \quad (1)$$

Choose any k shares, and restore the secret:

$$F(x) = \sum_{i=1}^k (y_i \times \Delta_{i,S}(x)) \quad (2)$$

Among them, the elements in the set S are composed of Z_p , and $i \in Z_p$, y_i is the secret share and $y_i = F(x_i)$.

2.4 Security Problem and Assumption

Determination Bilinear Diffie-Hellman (DBDH) problem: Let G and G_T be two multiplicative cyclic groups of prime order p . Let g be a generator of G and e be a bilinear

groups pairing, $e: G \times G \rightarrow G_T$. Randomly choose $a, b, c \in Z_p$ and $T \in G_T$. Let $\vec{y} = (g, g^a, g^b, g^c)$, if

$$|Pr[A(\vec{y}, e(g, g)^{abc}) = 0] - Pr[A(\vec{y}, T)] = 0| \geq \varepsilon \quad (3)$$

Then determine whether $T = e(g, g)^{abc}$.

Determination Bilinear Diffie-Hellman (DBDH) assumption: There is no polynomial time algorithm A that can solve the DBDH problem with the non-negligible advantage of ε .

3 Scheme Model

3.1 Model Design

Based on the edge computing scenario, we design an intent-driven attribute-based encryption scheme. Edge nodes are used to transfer computing from the user side to the edge side, and transfer storage from the central cloud to the edge side. According to the users' intents, select the intent parameters that meet users' requirements for file encryption, ciphertext transmission, and ciphertext decryption.

Data Owner (DO) transmits the intent parameters to the edge nodes, and the edge nodes are responsible for generating and storing the ciphertext. In the scheme, intents include security time and security level. Security time refers to the validity period of the ciphertext. If security time exceeds the specified value, the ciphertext will become invalid and the user will not be able to read the information unless DO updates the intents. Security level is determined based on the complexity of private key. We believe that the more complex the key, the more difficult to crack, and correspondingly, the higher the security level. In addition, based on the common intents of DO and Data User (DU), the secret segmentation threshold is determined through negotiation. The threshold is related to the difficulty of the attackers cracking ciphertext. Edge nodes push the ciphertext to DU according to his intents. With the powerful computing power of edge nodes, DU decrypts together with edge nodes. Edge nodes undertake a large number of complex decryption processes, while DU only performs a small amount of decryption operations. This process ensures that edge nodes cannot obtain complete plaintext, and only qualified users can obtain plaintext .

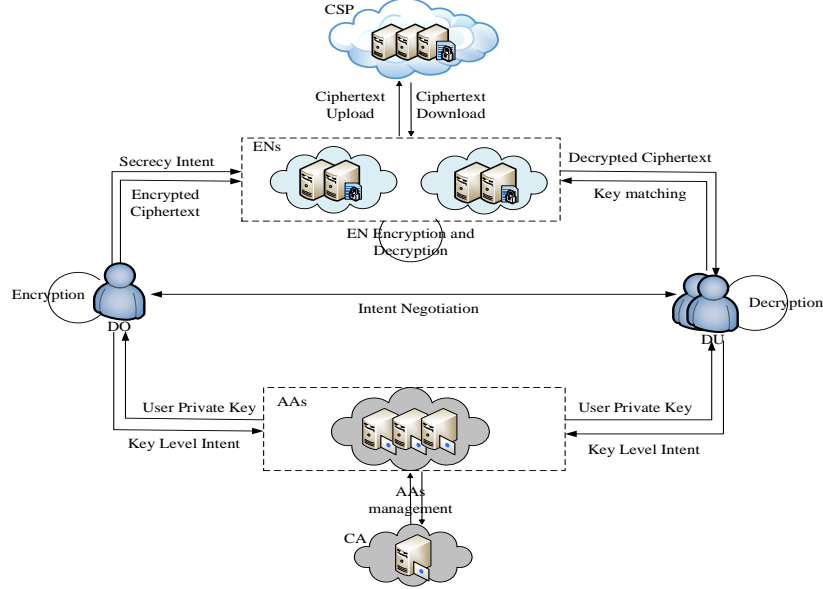


Fig. 1. Scheme Model.

3.2 Scheme Structure

Setup. Let G and G_T be two cyclic groups of prime order p . Let g be a generator of G and $e: G \times G \rightarrow G_T$. Define a hash function: $H_1: \{0,1\}^* \rightarrow Z_p$. There are N attributes, which are divided into n attribute trees. The root nodes of attribute trees are $U_0 = \{\omega_{10}, \omega_{20}, \dots, \omega_{n0}\}$. l_i is the depth of the i -th attribute tree, and the most depth of attribute trees is $l = \max\{l_1, l_2, \dots, l_n\}$. Randomly select attribute parameters from G , denoted as $V = \{v_1, v_2, \dots, v_n\}$ and $U = \{u_1, u_2, \dots, u_l\}$. And there are M attribute authorities in the system, which manage different attribute sets. Randomly select element y as the input parameter, and each authority randomly selects parameters α_i and β_i to satisfy $y = \sum_{i=1}^M \alpha_i \cdot \beta_i$. So we obtain the public key GPK , the master private key MSK , the public key PK_i and the private key SK_i of the i -th authority.

$$GPK = \{e(g, g)^y, g, g^y, V, U\} \quad (4)$$

$$MSK = \{y\} \quad (5)$$

$$PK_i = \{g^{\alpha_i}, g^{\beta_i}\} \quad (6)$$

$$SK_i = \{\alpha_i, \beta_i\} \quad (7)$$

Intent Negotiation. Users in the system jointly negotiate the secret segmentation threshold d , and construct the $d - 1$ degree polynomial $q(x)$ for each attribute authority, which satisfies the equation $q(0) = \beta_i$. The larger d , the more complex the polynomial $q(x)$ and the more difficult to attack.

Key Generation. The user private key is closely related to user attributes. The user attributes are represented as $t_{i\varphi} |_{1 \leq i \leq n, 1 \leq \varphi \leq h_2}$, where i represents the number of the attribute tree and φ represents the depth of the attribute. To trace the user identity, the hash function $H_1(ID)$ is used to represent identity information.

According to the user's intents, select the intent parameter γ to determine the secret level of the key. γ indicates the complexity of the key. So the larger the value of γ , the more complex the key. The system can customize the value of γ based on user requirements. However, γ cannot be 0. If it is 0, the key does not exist.

$$SK_u = \left\{ \begin{array}{l} SK_1 = g^\gamma, \\ SK_2 = g^{\beta_i \gamma}, \\ SK_3 = g^{\alpha_i \gamma}, \\ SK_4 = g^{\beta_i H_1(ID)}, \\ SK_5 = g^{\gamma + \beta_i H_1(ID)}, \\ SK_j = \left(v_j \prod_{\varphi=1}^{h_2} u_{\varphi}^{H_1(t'_{j\varphi})} \right)^\gamma \cdot g^{q(H_1(t'_{j\varphi}))}, \\ SK_{j,h+1} = u_{h+1}^\gamma, \dots, SK_{j,l_j} = u_{l_j}^\gamma \end{array} \right\} \quad (8)$$

The scheme utilizes edge nodes for complex calculations and requires the use of keys for encryption and decryption. However, the key is saved by the users themselves, so sharing some key components is feasible. In this way, even if edge nodes are not trusted or attacked by intermediaries, they cannot obtain the complete private key. On the one hand, it ensures the security of private key, and on the other hand, it fully utilizes the computing power of edge nodes. Therefore, the edge nodes will get the key components $\{SK_1, SK_2, SK_3, SK_j\}$.

Secrecy Intent. According to the encryption intents, the DO selects intent parameters s and t , where s indicates the ciphertext security time and t indicates the ciphertext security level. The validity time of the ciphertext can be customized according to the marking mode specified by the system.

Encryption. DO selects the ciphertext attribute set, and there is a hierarchical relationship between attributes. The ciphertext attributes can be represented as $t_{j\delta} |_{1 \leq j \leq n, 1 \leq \delta \leq h_1}$, where j represents the number of the attribute tree, δ represents the depth of the attribute, and h_1 represents the depth of j -th attribute tree. Input PK_i and intent parameter s . Then, the encrypted ciphertext of the edge nodes is CT_1 .

$$CT_1 = \left\{ \begin{array}{l} C_1 = \left(v_j \prod_{\delta=1}^{h_1} u_{\delta}^{H_1(t_j \delta)} \right)^s, \\ C_2 = g^{\alpha_i s}, \\ C_3 = g^s, \\ C_4 = g^{\beta_i s} \end{array} \right\} \quad (9)$$

Based on the edge nodes encryption, DO needs to embed the plaintext in intermediate ciphertext CT_1 to construct the final ciphertext. DO encrypts the intermediate ciphertext using the intent parameter t to obtain the final ciphertext.

$$CT = \left\{ \begin{array}{l} CT_1, \\ C_0 = me(g, g)^{y(t+s)}, \\ C_7 = g^t \end{array} \right\} \quad (10)$$

Decryption. The edge nodes use the key components for initial decryption to obtain the intermediate ciphertext CT_3 . This process transfers complex operations from users to edge nodes, which helps reduce the computational load on the client side, allowing the solution to still be implemented on weak terminals.

$$CT_2 = \prod_{j=1, t \in U} \left[\frac{e(C_2, SK_j) e(C_3, SK_2)}{e(C_1, SK_3) e(C_4, SK_1)} \right]^{\Delta_{H(t), s(0)}} \quad (11)$$

$$CT_3 = \prod_{i=1}^M CT_2 \quad (12)$$

DU uses full private key to further decrypt and obtain plaintext.

$$m = \frac{C_0}{e(C_7, \overline{SK}_4) \cdot CT_3} \quad (13)$$

4 Scheme Analysis

4.1 Security Proof

Theorem 1 If the DBDH assumption is true, the attacker A cannot win the security game in the probability polynomial time. The advantage of security in this paper is the possibility of solve the DBDH problem.

Proof Suppose that an attacker A can attack the scheme with a non-negligible advantage ε , we think A simulator B can solve the DBDH problem with a non-negligible advantage $\varepsilon/2$.

In the process of proof, we construct a simulator B . There is a challenger C in the security game, and let attacker A attack the game. Let set two groups G_1, G_2 and bilinear mapping e , the generator of G_1 is g . All attributes are divided into the attribute trees.

The depth of the i -th attribute tree is l_i , and the maximum depth of all attribute trees is l and expressed as $l = \max\{l_i\}_{1 \leq i \leq n}$. Define a hash function $H_1: \{0,1\}^* \rightarrow Z_p$. Select a bit $\xi \in \{0, 1\}$, set the tuple (A, B, C, δ) , then randomly select elements $a, b, c, z \in Z_p$.

There is $(A, B, C, \delta) = (g^a, g^b, g^c, e(g, g)^{abc})$, only if $\xi = 0$.

And $(A, B, C, \delta) = (g^a, g^b, g^c, e(g, g)^z)$, only if $\xi = 1$.

The challenger C sends (A, B, C, δ) to simulator B for the security game.

Init. The attacker A arbitrarily selects an access structure τ^* to be challenged and sends it to a challenger C . The attribute t^* is in the attribute tree with root node and depth. The challenger C is initialized the system model and the simulator B generates parameters g and p .

Setup. Randomly select two sets of parameters $V = \{v_i\}_{1 \leq i \leq n}$ and $U = \{u_i\}_{1 \leq i \leq l}$. C gets the system public key, the master key, the public and private keys of authorities. Then C retains the master key and sends the system public key to A .

$$GPK = \{e(g, g)^y, g, g^y, V, U\} \quad (14)$$

$$MSK = \{y\} \quad (15)$$

Phase 1. A arbitrarily constructs an attribute set R which does not satisfy the access structure τ^* . For $\forall r \in R$, the attribute r is in the d -th tree, the depth is p , let's denote its path as $L_r = (r_{d0}, r_{d1}, \dots, r_{d,p-1}, r)$. Assume that the attribute r in the set R is managed by the i -th authority, the attribute is recorded as t_{ir} . A requests the private key of the attribute set R to C , and C enter the attribute set R in the simulator B to get the private key. Then C returns the private key to the A .

$$SK_u = \left\{ \begin{array}{l} SK_1 = g^y, \\ SK_2 = g^{\beta_i y}, \\ SK_3 = g^{\alpha_i y}, \\ SK_4 = g^{\beta_i H_1(ID)}, \\ SK_5 = g^{y + \beta_i H_1(ID)}, \\ SK_j = \left(v_j \prod_{\varphi=1}^{h_2} u_{\varphi}^{H_1(t_{ir})} \right)^y \cdot g^{q(H_1(t_{ir}))}, \\ SK_{j,h+1} = u_{h+1}^y, \dots, SK_{j,l_j} = u_{l_j}^y \end{array} \right\} \quad (16)$$

Challenge. A completes the key query in phase 1 and selects two equal-length plaintexts M_0 and M_1 , then sends to the challenger C . C randomly throws coins to get a bit, and B also selects the same bit $\mu \in \{0, 1\}$. Based on the value of μ , B encrypts message M_μ which satisfies the access structure τ^* . B generates a ciphertext CT^* and sends to A .

Phase 2. A makes the second key query for challenge ciphertext M_μ . The interaction between A and C is the same as the phase 1.

Guess. A answers which message is encrypted and outputs $\mu_0 = 0$ or $\mu_0 = 1$. The advantage of winning the game is $Pr[\mu = \mu_0] - 1/2$.

Analysis. If the simulator B outputs $\xi = 0$ and $\mu = \mu_0$, it means that the attacker A gets the encrypted ciphertext, that is, $Z = e(g, g)^{abc}$. It is assumed that the advantage of attacking is ε , there is $Pr[\mu = \mu_0 | \xi = 0] = 1/2 + \varepsilon$. If B outputs $\xi = 1$, it means that A cannot get the ciphertext. Because of $Z = e(g, g)^z$ and z is a random number. So A cannot recover plaintext, there is $Pr[\mu = \mu_0 | \xi = 1] = 1/2$. Based on the above discussion, A wins the game for the advantage P .

$$\begin{aligned}
 P &= Pr[\mu = \mu_0] - \frac{1}{2} \\
 &= Pr(\mu = \mu_0 | \xi = 0) \cdot Pr(\xi = 0) + Pr(\mu \neq \mu_0 | \xi = 1) \\
 &\quad \cdot Pr(\xi = 1) - \frac{1}{2} \\
 &= \left(\varepsilon + \frac{1}{2}\right) \times \frac{1}{2} + \frac{1}{2} \times \frac{1}{2} - \frac{1}{2} = \frac{1}{2}\varepsilon
 \end{aligned} \tag{17}$$

so the simulator B cannot solve the DBDH problem with a non-negligible advantage $\varepsilon/2$, we think the attacker A cannot attack the scheme with a non-negligible advantage ε .

4.2 Performance Analysis

The performance analysis of the scheme mainly includes storage cost and calculation cost. The storage cost can describe the space usage of the scheme. The smaller space occupation, the richer scenarios. The calculation cost describes the computational usage of the scheme on elements such as public key, private key, and ciphertext. The less computation, the higher the system efficiency. The following two tables compare the performance parameters of references [18] and [19] with our scheme.

Table 1. Storage Cost.

	GPK		MSK		CT	SK_u
[18]	$(n \cdot l + 13) G + G_T $		$2 G $		$(U_1 + 9) G $	$(U_2 + 7) G $
[19]	$(3n + 2) G $		$4 G $		$(2 U_1 + 1) G $	$(U_2 + 2) G $
	GPK	PK_i	MSK	SK_i		
Ours	$(n + l + 3) G $	$2 G $	$ G $	$2 G $	$(U_1 + 6) G $	$(U_2 + 2) G $

Table 2. Calculation Cost.

	Encryption		KeyGeneration	Decryption	
				Cloud	User
[18]	$14\tau_m + \tau_e + 3\tau_r$		$22\tau_m + 5\tau_r$	$(U_2 + 11)\tau_e + 8\tau_m$	$3\tau_e + \tau_m$
[19]	Fog $2n \cdot \tau_m$	User $(2n + 2)\tau_m$	$(d + 2)\tau_m$	Fog $ U_2 \tau_m + 4\tau_e$	User τ_m
Ours	Edge $(h_1 + 4)\tau_m$	User $\tau_m + \tau_e$	$(h_2 + 8)\tau_m$	Edge $ U_2 \tau_m + 6\tau_e$	User τ_e

4.3 Function Analysis

Table 3 compares the functions of several similar schemes and displays the capabilities and advantages of different schemes.

From the functional perspective, our scheme meets the timeliness requirements, outsourced computing, and hierarchical attribute matching. Compared with other schemes, our scheme has significant functional advantages .

Table 3. Function Comparison.

	Time access function	Outsourced computing	Hierarchical attribute	intent-driven
[18]	√	×	×	×
[19]	×	√	×	×
Ours	√	√	√	√

5 Conclusion

In the scheme, the user intents are combined with traditional encryption processes. In the stages of secret segmentation threshold, key generation, user encryption, and edge nodes encryption, encryption features are independently defined by the users, which

increases their decision-making ability and improves the intelligence of the scheme. In addition, the scheme utilizes edge nodes to outsource complex calculations, and how to encrypt is decided by the user intent parameters. This solution avoids man-in-the-middle attacks and data leakage on the cloud. Finally, we conduct security verification and performance analysis on the scheme.

However, the scalability of this scheme can be further improved. The encryption intent chosen by the user, including security time and security level. If other intents are added in the future, it is necessary to include the intent parameters again and update the existing encryption and decryption algorithms. Therefore, the next step will be to design a highly scalable intent-driven attribute encryption scheme to be applied to more scenarios.

References

1. "Intent: Don't Tell Me What to Do! (Tell Me What You Want)." [Online]. Available: <https://www.sdxcentral.com/articles/contributed/networkintentsummitperspectivedavidlenrow/2015/02/>.
2. Pang Lei, Yang Chungang, et al. A Survey on Intent-Driven Networks[J]. IEEE Access. DOI:10.1109(2020).
3. Huang J, Yang C, Kou S, et al. A Brief Survey and Implementation on AI for Intent-Driven Network[C]. 2022 27th APCC. IEEE, 2022: 413-418(2022).
4. OUYANG, YING YANG, et al. A Brief Survey and Implementation on Refinement for Intent-Driven Networking[J]. 2021,35(6):75-83. DOI:10.1109(2021).
5. Zhang J, Guo J, Yang C, et al. A conflict resolution scheme in intent-driven network[C]. 2021 IEEE/CIC ICC. IEEE, 2021: 23-28.
6. Zhang L, Dong R, Li F, et al. Intent-Driven Internet of Things: Architectures, Technology, and Challenges[C]. 2023 6th WCCCT. IEEE, 2023: 112-117.
7. LYU Gaofeng, SUN Zhigang, LI Tao. Active computing method in network edge for user intention[J]. 2018,40(06):68-74(2018).
8. Q. Huang, Y. Yang and L. Wang. Secure Data Access Control With Ciphertext Update and Computation Outsourcing in Fog Computing for Internet of Things. in IEEE Access, vol. 5, pp. 12941-12950(2017).
9. Huang Q, Yang Y, Shen M. Secure and efficient data collaboration with hierarchical attribute-based encryption in cloud computing[J]. Future Generation Computer Systems, 2017, 72: 239-249(2017).
10. PENG H., LING J., QIN S., et al. Attribute-based encryption scheme for edge computing[J]. Computer Engineering, 2021, 47(1):37-43(2021).
11. Q.S. Leng, W.P. Luo. Attribute-based Encryption with Outsourced Encryption[J]. Communications Technology, 2021, 54(9): 2242-2246(2021).
12. K. Huang. Multi-Authority Attribute-Based Encryption for Resource-Constrained Users in Edge Computing. ITCA49981.2019.00078(2019).
13. Cohen R, Barabash K, Rochwerger B, et al. An intent-based approach for network virtualization[C]. Integrated Network Management (IM 2013), IEEE, 2013
14. Bi J, Cheng Y, Xie C, et al. Policy-Based Management Framework for the Simplified Use of Policy Abstractions (SUPA)[J]. Policy, 2018
15. Behringer M, Pritikin M, Bjarnason S, et al. Autonomic networking: Definitions and design goals (RFC 7575) [J]. Internet Research Task Force, 2015, 10.

16. Dan Boneh and M. Franklin. Identity-based encryption from the weil pairing. Society for Industrial and Applied Mathematics(2001).
17. A. Beigel. Secure schemes for secret sharing and key distribution. PhD thesis, Israel Institute of Technology, Technion(1996).
18. M. XU, M. FANG. Cloud Outsourcing Support Aging Access Attributes of Anonymous Encryption Scheme [J]. Journal of Chinese Computer Systems, 2018, 39(02):225-229(2018).
19. WANG Zheng, SUN Xiao. A compact attribute-based encryption scheme supporting computing outsourcing in fog computing [J]. Computer Engineering and Science, 2022, 44(03): 427-435(2022).
20. Y. Li, Z. Dong, K. Sha, C. Jiang, J. Wan and Y. Wang. TMO: Time Domain Outsourcing Attribute-Based Encryption Scheme for Data Acquisition in Edge Computing, in IEEE Access, vol. 7, pp. 40240-40257(2019).
21. Q. TONG, H. HE, et al. Hierarchical Lightweight Access Control Scheme in Cloud Environment[J]. Computer Engineering and Applications, 2022, 58(21): 109-118(2022).
22. NIU Shufen, GE Peng, et al. A Privacy Protection Scheme Based on Attribute Encryption in Mobile Social Networks[J]. Journal of Electronics & Information Technology, 2023, 45(03): 847-855(2023).