Journal of Systems Architecture 160 (2025) 103331

Contents lists available at ScienceDirect

Journal of Systems Architecture

journal homepage: www.elsevier.com/locate/sysarc

A CP-ABE-based access control scheme with cryptographic reverse firewall for IoV

Xiaodong Yang [a], Xilai Luo [a,*], Zefan Liao [a], Wenjia Wang [a], Xiaoni Du [b], Shudong Li [c]

[a] College of Computer Science and Engineering, Northwest Normal University, China
[b] College of Mathematics and Statistics, Northwest Normal University, China
[c] Cyberspace Institute of Advanced Technology, Guangzhou University, China

ARTICLE INFO

Keywords:
Attribute-based encryption
Multi-authority
Internet of Vehicles
Cryptographic reverse firewall
Outsource decryption

ABSTRACT

The convergence of AI and internet technologies has sparked significant interest in the Internet of Vehicles (IoV) and intelligent transportation systems (ITS). However, the vast data generated within these systems poses challenges for onboard terminals and secure data sharing. To address these issues, we propose a novel solution combining ciphertext policy attribute-based encryption (CP-ABE) and a cryptographic reverse firewall (CRF) mechanism for IoV. This approach offers several advantages, including offline encryption and outsourced decryption to improve efficiency. The CRF mechanism adds an extra layer of security by re-randomizing vehicle data, protecting sensitive information. While single-attribute authority schemes simplify access control, they are not ideal for IoV environments. Therefore, we introduce a multi-authority scheme to enhance security. Performance analysis demonstrates our scheme's ability to optimize encryption and decryption while safeguarding vehicle data confidentiality. In summary, our solution improves data management, access control, and security in the IoV, contributing to its safe and efficient development.

1. Introduction

Advances in 5G technology, coupled with the growing volume of vehicular traffic, have intensified concerns regarding traffic safety, travel efficiency, and environmental impact. In response, Intelligent Transport Systems (ITS) and the IoV have emerged as critical components of modern transportation infrastructure. The functionality of the IoV relies on three key elements: the internal vehicle network, the vehicle-to-vehicle communication network, and the in-vehicle mobile internet. These elements integrate technologies such as sensors, RFID (Radio Frequency Identification), and automated control systems, operating under established communication protocols to enable seamless, dynamic data exchange between vehicles and the broader network.

While drivers benefit from applications like navigation and traffic information sharing, the limited computing power of onboard terminals is insufficient for computationally intensive tasks such as autonomous driving and AI-based obstacle avoidance [1]. A potential solution is offloading data processing to cloud servers, but the large volume of vehicle-generated data introduces high latency in communication between the onboard terminal and the cloud, compromising real-time decision-making [2–4]. This latency, coupled with the risks associated with data leakage and theft in semi-trusted cloud environments, raises significant concerns about data security [5]. Therefore, cloud-based solutions alone are insufficient to meet the demands of the IoV. To mitigate these issues, edge computing [6], fog computing [7], and Roadside Units (RSUs) [8] have been proposed. RSUs, with their higher computational capabilities, can process data more efficiently and upload it to cloud servers in real time, addressing the challenges of latency and limited onboard processing power.

However, data security remains a critical issue. One potential solution is encrypting data before transmission, which introduces challenges in ciphertext sharing. Traditional symmetric encryption, requiring a one-to-one correspondence between keys and users, proves inefficient for securing large volumes of data in IoV environments. Conventional asymmetric encryption algorithms also struggle with ciphertext sharing and are ill-suited for the frequent updates characteristic of IoV applications. A more appropriate approach is Attribute-Based Encryption (ABE), which enables fine-grained access control, supports encryption for multiple recipients, and facilitates the creation of complex access policies [9–11]. ABE allows data owners to control who can access their data, but the decryption process is computationally intensive, requiring numerous pairing and exponential operations. This places a significant burden on resource-constrained onboard terminals,

* Corresponding author.
E-mail addresses: yangxd200888@163.com (X. Yang), 2023222208@nwnu.edu.cn (X. Luo), lzf0097@163.com (Z. Liao), neuer1130@163.com (W. Wang), duxiaonwnu@163.com (X. Du), lishudong@gzhu.edu.cn (S. Li).

https://doi.org/10.1016/j.sysarc.2025.103331
Received 11 August 2024; Received in revised form 4 December 2024; Accepted 2 January 2025
Available online 17 January 2025

hindering timely data retrieval and impeding efficient communication. As the number of attributes increases, the decryption complexity grows, leading to slower decryption times and higher resource consumption.

To address these challenges, several outsourced ABE schemes have been proposed [12–15], which offload expensive operations to cloud servers, alleviating the computational load on onboard terminals. However, even secure theoretical implementations of ABE are vulnerable to practical attacks. Sophisticated adversaries may exploit backdoors [16], manipulate pseudo-random number generators [17,18], or intercept hardware interactions to gain unauthorized access to sensitive data. To counter these threats, the concept of a Cryptographic Reverse Firewall (CRF) was introduced [19]. The CRF, positioned between the user and the server, intercepts and alters messages to ensure data security, even if the user is compromised.

Moreover, traditional ABE schemes rely on a single attribute authority, which poses a risk of key leakage if the authority colludes with an adversary. To mitigate this, we propose a multi-authority ABE scheme, integrated with a CRF, to enhance security and prevent collusion attacks. The key contributions of this paper are as follows:

1. We propose a CP-ABE-based scheme that enables more granular access control policies, enhancing the system's flexibility. This proves particularly beneficial in IoV scenarios such as IoV communication, where data access can be dynamically adjusted in accordance with the context.
2. The scheme integrates multiple attribute authorities to prevent collusion attacks and guarantee secure key management. Each authority is responsible for managing vehicle attribute keys, enhancing the security and efficiency of key generation, which is ideal for environments like smart cities or autonomous vehicle fleets.
3. We enhance the CRF module by incorporating key parameter re-randomization within the multi-authority ABE framework, strengthening security in IoV communications, even if certain parts of the system are compromised.
4. The scheme optimizes decryption efficiency through the use of online-offline encryption techniques and offloading decryption operations. Decryption time does not increase linearly with the number of attributes, making it suitable for real-time applications like hazard detection and traffic optimization.
5. The scheme also supports message integrity verification, which can be easily carried out by onboard terminals using simple hash functions, ensuring the authenticity of IoV messages and preventing malicious tampering in safety-critical communications.

The paper is organized as follows: Section 2 reviews existing attribute-based encryption schemes and the application of CRFs. Section 3 provides an overview of the system and security models. Section 4 discusses the base scenario and the extended CRF module. Section 5 presents security proofs for the base scheme and the CRF-enhanced scheme. Section 6 reports on experiments and results. Finally, Section 7 concludes the paper.

2. Related work

Sahai [10] introduced fuzzy identity-based encryption, which paved the way for Attribute-Based Encryption (ABE). ABE later branched into two forms: Key-Policy ABE (KP-ABE) [9] and Ciphertext-Policy ABE (CP-ABE) [11]. Initially, both schemes used access trees to define policies. However, the first CP-ABE scheme only provided security under the random oracle model. Waters [20] introduced an LSSS-based CP-ABE scheme that encodes policies using matrices. This foundational model has influenced many subsequent ABE schemes, which have expanded into diverse domains, particularly cloud computing. For example, Yu et al. [21] proposed a KP-ABE scheme enabling data delegation to semi-trusted cloud servers while ensuring confidentiality. Yang et al. [22] introduced a CP-ABE scheme for dynamic big data updates, and Feng et al. [23] developed a CP-ABE scheme for industrial IoT. Other schemes [24,25] have improved security and efficiency, broadening ABE's application to the Internet of Medical Things (IoMT).

CP-ABE enables fine-grained access control, making it highly applicable in sectors such as smart healthcare and intelligent transportation. However, single-attribute authority ABE schemes are vulnerable to collusion attacks. To address this, it is desirable to delegate each attribute to different attribute authorities. Chase [26] was the first to introduce the concept of multiple attribute authorities within the ABE framework, where various authorities oversee different attributes. Lewko and Waters [27] later introduced the initial decentralized ABE framework with multiple authorities. Following this, Chaudhary et al. [28] proposed a multi-authority CP-ABE scheme tailored for the Internet of Vehicles (IoV) context.

Considering the constrained computing capabilities of user terminals, Green et al. [12] introduced an ABE scheme that delegates decryption computations to the cloud. Lai et al. [13] improved upon this by achieving verifiability of outsourced decryption. Zhong et al. [29] further enhanced the efficiency of outsourced decryption ABE schemes and applied them to smart healthcare scenarios.

Mironov and Stephens-Davidowitz [19] were the first to introduce the concept of a reverse firewall. They proposed a generic architecture to prevent user tampering, which could lead to data leakage. However, the previous approach was found unsuitable for ABE schemes, prompting Ma et al. [30] to introduce a cryptographic reverse firewall utilizing the CP-ABE scheme. Additionally, Hong et al. [31] proposed a KP-ABE scheme with multiple authorities. Due to the limitations of KP-ABE in achieving fine-grained access control, Zhao et al. [32] proposed a CP-ABE scheme incorporating a CRF and leveraged outsourced decryption to alleviate computational burdens. However, these approaches suffer from drawbacks, such as reliance on a single attribute authority or excessive computational overhead. Moreover, there is a risk of system compromise, which could lead to data leakage, especially in the context of IoV, characterized by constrained computational resources and stringent data privacy requirements. At the same time, the development of IoV places higher demands on the security and flexibility of access control. Therefore, the proposed scheme combines CP-ABE, CRF, and multi-authority models to meet the requirements for security, flexibility, and low computational overhead.

3. System model and definitions

3.1. Preliminaries

1. *Bilinear Maps:* Involve two multiplicative cyclic groups of prime order p, denoted as G and G_T, with g representing a generator of G. A bilinear map $e : G \times G \rightarrow G_T$ must satisfies the following three features:
   (a) Non-degeneracy: $e(g, g) \neq 1$.
   (b) Computability: Efficient computation of $e(M, N)$ for any elements $M, N \in G$ is achievable through a polynomial-time algorithm.
   (c) Bilinearity: Efficient computation of $a, b \in Z_p$ for any elements $M, N \in G$ we can acquire $e(M^a, N^b) = e(M, N)^{ab}$.

2. *Access Structure:* Consider a set $P = \{P_1, P_2, \ldots, P_n\}$ representing n users. A collection Q is deemed monotone if, for any subsets $\forall K, L$: if $K \in Q$ and $K \subseteq L$, then $L \in Q$. Let Q bbe a nonempty subset of P that is monotonic, i.e. $Q \subseteq 2^{\{P_1, P_2, \ldots, P_n\}} \backslash \{\varnothing\}$, then call Q a monotone access structure. In the context of access control, sets included in Q are identified as authorized, while those that are not included are referred to as unauthorized sets.

3. *Linear Secret Sharing Scheme (LSSS):* Let $\tilde{A} = \{\tilde{A}_1, \tilde{A}_2, \ldots, \tilde{A}_N\}$ be defined as the set that includes all possible attribute names. Corresponding to each attribute name $\tilde{A}_i \in \tilde{A}$ within A, there is an associated set of attribute values, denoted as $\widetilde{A}_i = \{A_{i,1}, A_{i,2}, \ldots, A_{i,b_i}\}$, where b_i is the order of $\tilde{A}_i$. The policy for access is denoted as $T = (M, \rho, V)$ Within the context of a linear secret sharing scheme, M denotes a matrix structured with l row size and n column size. ρ denotes a function that associates each row of M with an attribute name in $\tilde{A}_i$. $V = \{v_{\rho(i)}\}_{i\in[1,l]}$ represents the set of attribute values associated with $T = (M, \rho)$. A LSSS encompasses the following pair of algorithms:

   (a) Distribute: Regarding the confidential value $s \in Z_p$, arbitrarily choose a vector $f = (s, f_2, \ldots, f_n)$, where $f_2, \ldots, f_n \in Z_p$. Calculate $\lambda_i = M_i \cdot f$, where M_i is the i_{th} row of matrix M. λ_i is a share of s that corresponds to $\rho(i)$.

   (b) Reconstruct: Let $S \in \tilde{A}$ is permissible for any recognized group and $I = \{i : \rho(i) \in S\} \subseteq \{1, 2, \ldots, l\}$, then, there is a collection of constants $\{\omega_i \in Z_p\}$ satisfy $\sum_{i\in I} \omega_i M_i = (1, 0, \ldots, 0)$. The secret s could be reconstructed by us via calculating $\sum_{i\in I} \omega_i M_i = s$.

   Assume S= $\{I_u, S\}$ represents the collection of attributes for users. $I_u \subseteq \tilde{A}$ represents a collection of user attribute names. $S = \{s_i\}_{i\in I_u}$ denotes a set that includes all the attribute values of the user. For $\forall i \in I$, where $I = \{i : \rho(i) \in S\} \subseteq \{1, 2, \ldots, l\}$, if i satisfies (M, ρ) and $s_{\rho(i)} = v_{\rho(i)}$, thereafter, we identify S as matching T.
4. *q-BDHE problem:* Suppose G and G_T represent two cyclic groups with multiplication as their operation, and the order of each is the prime p, and g be a generator of G. G_T has a bilinear map $e : G \times G \to G_T$. Choose $t, f \in Z_p$ at random, and calculate $J = (g, g^t, g^f, g^{f^2}, \ldots, g^{f^q}, g^{f^{q+2}}, \ldots, g^{f^{2q}})$. In the context of the q-BDHE problem, it is posited that no algorithm operating within polynomial time can differentiate between $e(g, g)^{f^{q+1}t} \in G_T$ and $K \in G_T$ with a significant advantage.
5. *Cryptographic Scheme:* The cryptographic scheme $\mathcal{P}$ defines the interaction between parties $(P_1, P_2, \ldots, P_l)$ with states. The process of scheme establishment is denoted by $setup(1^\lambda)$, where λ refers to the security parameters. Each party enters the public parameters P_g and related messages, and then runs the system initialization algorithm to obtain the corresponding state $(v_{P_i})_{i=1}^{l}$ for each party. According to the order in which the scheme proceeds, the parties process messages from other parties in the scheme. Also, each party must have the corresponding algorithms $next_{P_i}(v_{P_i})$ and $receive_{P_i}(v_{P_i})$. $next_{P_i}(v_{P_i})$ is used to output the updated message, $receive_{P_i}(v_{P_i})$ is used to output the states of the parties after the message update. After the scheme is completed, each party has algorithm $output_{P_i}(v_{P_i})$ return the results of the scheme. We assume that the scheme $\mathcal{P}$ meets functionality requirement $\mathcal{F}$ and security requirements $\mathcal{S}$.
6. *Cryptographic Reverse Firewall:* $\mathcal{W}$, the stateful algorithm, is synonymous with the Cryptographic Reverse Firewall. When provided with a current state and an input message, the algorithm processes them and subsequently outputs an updated state and message. For ease of presentation, the state of $\mathcal{W}$ is not explicitly written out in the definition. Given that P is a party and $\mathcal{W}$ is a firewall, the expression $\mathcal{W}\circ P$ is introduced to indicate the party that emerges from their composition.

$$\begin{aligned}\mathcal{W}\circ P &= receive_{\mathcal{W}\circ P}(v, \mathcal{W}) \\ &= receive_P(v, \mathcal{W}(m)) \\ &= next_{\mathcal{W}\circ P} = \mathcal{W}(next_P(v)) \\ &= output_{\mathcal{W}\circ P}(v) = output_P(v)\end{aligned} \tag{1}$$

When the composite party participates in the scheme, the initial state of the firewall $\mathcal{W}$ is set as the public parameter P_g. If

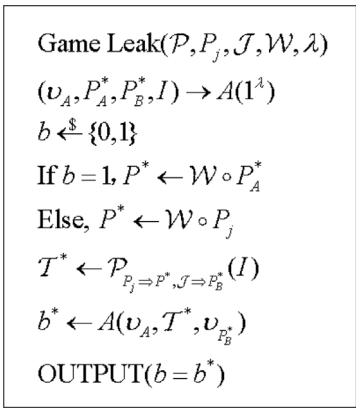

Fig. 1. Leak game.

$\mathcal{W}$ and a party P form a composed party, then we call $\mathcal{W}$ a cryptographic reverse firewall for P. Next we give definitions of three properties of CRFs:

(a) Function Maintaining: In the context of any given reverse firewall identified by $\mathcal{W}$ and any given party identified by P, let $\mathcal{W}^1 \circ P = \mathcal{W} \circ P$. For $k \geq 2$, let $\mathcal{W}^k \circ P = \mathcal{W} \circ (\mathcal{W}^{k-1} \circ P)$. For a framework $\mathcal{P}$ that adheres to the functionality requirement $\mathcal{F}$, we define the reverse firewall $\mathcal{W}$ maintains functionality if the composed party $\mathcal{W} \circ P$ guarantees the functionality of the party P under the scheme $\mathcal{P}$ in polynomial time.

(b) Weakly Security-preserving: $\mathcal{P}$ operates under the premise that it will fulfill the functionality need $\mathcal{F}$ and the security need $\mathcal{S}$. When faced with any polynomial-time adversary B, we say that the scheme $\mathcal{S}$ satisfies weakly security-preserving if $\mathcal{W} \circ P$ satisfies the security requirement $\mathcal{S}$.

(c) Weakly Exfiltration-resistant: The game Leak($\mathcal{P}, P_j, \mathcal{J}, \lambda$), as depicted in the Fig. 1, is the work of designers Mironov and Stephens-Davidowitz [19]. The game is a security game between a reverse firewall $\mathcal{W}$ of party P and a scheme $\mathcal{P}$ containing a tampering party $\mathcal{J}$. The adversary may control a party by hacking into the party's algorithm $receive, next, output$.
The purpose of the game is to let the adversary discern whether the party's actions are honest or tampered with. Thus, a reverse firewall with leak resistance can make it impossible for an adversary to tell if party P has been tampered with, or if the party is known to have been tampered with but does not know if the operation is honest, hence protecting the important privacy of the party.
If adversary B within the Leak($\mathcal{P}, P_j, \mathcal{J}, \lambda$) game cannot succeed in polynomial time with a noticeable advantage and while maintaining the party's functionality $\mathcal{F}$, then we label the reverse firewall $\mathcal{W}$ as weakly capable of resisting exfiltration.

3.2. System model

Fig. 2 depicts the four components that constitute our scheme: Attribute authorities (AA), Cloud server (CS), Data user (DU), Data owner (DO). In addition, the system contains three reverse firewalls. To implement data re-randomization within the RSU, three firewalls are strategically positioned: $\mathcal{W}_{AA}$, the reverse wall for AA; $\mathcal{W}_{DO}$, acting as the reverse firewall for DO; and $\mathcal{W}_{DU}$, fulfilling the same role for DU.

CS is mainly deployed to store cipher text and conversion key.

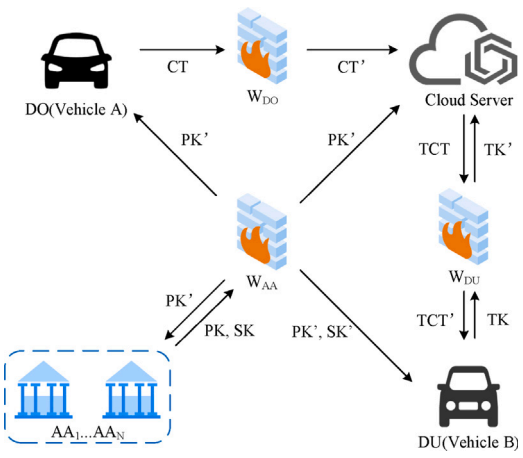

Fig. 2. System model.

AA is charged with the responsibility of establishing the public parameters and generating the master secret keys.

DU includes setting the access policy that guides the encryption process and producing a verification credential. After these steps are accomplished, the DU uploads both the encrypted data and the verification credential to the cloud server.

DO initiates the process by generating a conversion key, which is then uploaded to the cloud server. Following this, the DO retrieves the ciphertext and the verification credential from the cloud server to carry out the concluding stages of decryption and integrity verification.

$\mathcal{W}_{AA}$ includes the re-randomization of public parameters and the secret keys that belong to users.

$\mathcal{W}_{DO}$ is responsible to rerandomize cipher texts.

$\mathcal{W}_{DU}$ is responsible to rerandomize conversion keys and conversion ciphertexts.

3.3. Security model

The DO and the DU in our system are considered completely trustworthy. However, the reverse firewalls and cloud server are deemed "honest and curious", meaning they will comply with the algorithm's steps but will also endeavor to discover any private information within the data. Furthermore, there is a risk of the Attribute Authority colluding with an adversary. In response to this challenge, we have put in place a selective CPA security game, and the sequence of events within this game is as follows:

1. *Init Phase:* The rival B declares a set of malicious attribute authorities $R = (\hat{A}_i)_{i\in I}$ and access policies $(M_i{}^*, \rho_i{}^*)_{i\in I^*}$ to be challenged, where $I \subseteq \{1,2,\ldots,N\}, I^* \subseteq \{1,2,\ldots,N\}$. Then B sends algorithms $Global setup^*$, $AASetup^*$, $KeyGen^*$, $Key.ran^*$, $enc.offline^*$, $enc.online^*$ to challenger F.
2. *Setup Phase:* F executes algorithms $Global setup^*$ and $AASetup^*$ to obtain the public parameter $Params$, attribute authorities public key PK and private key pairs $(PK_i, A_{SK_i})_{i\in I}$. Subsequently, the reverse firewall puts the $W_{AA}.SetUp$ algorithm into action to generate and announce the new public key PK', and in doing so, also retains the corresponding random number f. B can receive $PK_i{}'$ from all non-malicious attribute authorities and $(PK_i, A_{SK_i})_{i\in I}$ from all malicious attribute authorities.
3. *Query Phase 1:* Adaptive requests for secret keys regarding attribute sets $S_1, S_2, \ldots, S_q$ can be made by B. Each time B performs a key query, when submitting a set of attributes, it is imperative that they do not comply with the access structure rules outlined by $(M_i{}^*, \rho_i{}^*)_{i\in I^*}$, nor come from a malicious attribute authority $R = (\hat{A}_i)_{i\in I}$. For every query S_i, F executes algorithm $KeyGen$ and obtains corresponding secret key SK_i. Then F executes algorithm $\mathcal{W}_{AA}.KG$ and gets the re-randomized private key $SK_i{}'$. Subsequently, F executes $KeyGen.ran$ to get conversion key TK_i. Then F executes $\mathcal{W}_{DU}.TKUpdate$ to obtain re-randomized conversion key $TK_i{}'$. Eventually, F sends $(SK_i{}', TK_i{}')$ to B.
4. *Challenge Phase:* Two equal-length plaintexts, m_0, m_1, are delivered by B as part of the protocol. F randomly chooses $b \in \{0,1\}$ and executes $Enc.Offline^*$, $Enc.Online^*$ to obtain challenge ciphertext CT_b. Then F calls $\mathcal{W}_{DO}.Enc.Offline$, $\mathcal{W}_{DO}.Enc.Online$ to get updated cipher text $CT_b{}'$. F sends $CT_b{}'$ to B.
5. *Query Phase 2:* Same as Query Phase 1.
6. *Guess Phase:* B outputs the guess $b' \in \{0,1\}$ for b.

Definition 1. The criterion for the basic scheme's selective CPA-secure is met when the probability of adversary B's success in the game during polynomial time is negligible.

4. System construction

4.1. Basic scheme

The scheme contains N attribute authorities, each attribute authority managing one class of attributes $\widetilde{A}_i = \{A_{i,1}, A_{i,2}, \ldots, A_{i,b_i}\}$, $A_{i,1} \in Z_p$, $i = 1,2,\ldots,N$, $j = 1,2,\ldots,b_i$.

1. *Global Setup:* Attribute authority AA_1 sets commonly known parameters $Params = \{g, u, v, w, h, G, G_T, H_0()\}$ and publishes them, H_0 is the designated collision-resistant hash function for generating robust verification credentials within the system. $H_0() : \{0,1\}^* \to \{0,1\}^{\mathcal{L}_{H_0}}$.
2. *AASetup:*
   (a) For each Attribute Authority, the process involves randomly choosing $\alpha_i \in Z_p$, determining $Y_i = e(g,g)^{\alpha_i}$, and then distributing Y_i to other attribute authorities. As the process concludes, each attribute authority carries out the calculation for $Y = \prod_{i=1}^{N} Y_i = e(g,g)^{\sum_{i=1}^{N}\alpha_i} = e(g,g)^{\alpha}$, where $\alpha = \sum_{i=1}^{N}\alpha_i$.
   (b) Each attribute authority $\hat{A}_i$ operates as follows:
   - Randomly select $N-1$ elements $s_{ik} \in Z_p (k \in \{1,2,\ldots,N\}\backslash\{i\})$, calculate $g^{s_{ik}}$ and send it to other attribute authorities.
   - After receiving $N-1$ components $g^{s_{ki}}$ from other ascribe powers $\hat{A}_k (k \in \{1,2,\ldots,N\}\backslash\{i\})$, the master key MK_i is calculated by the following formula:

$$MK_i = \prod_{k\in\{1,2,\ldots,N\}\backslash\{i\}} (g^{s_{ik}}/g^{s_{ki}}) = g^{\left(\sum_{k\in\{1,2,\ldots,N\}\backslash\{i\}} s_{ik} - \sum_{k\in\{1,2,\ldots,N\}\backslash\{i\}} s_{ki}\right)}, \quad (2)$$

   where $\prod_{i=1}^{N} MK_i = 1$.
   - For each attribute $A_{i,j} \in \widetilde{A}_i$, calculate $u^{A_{i,j}} h$.

   Attribution authority publishes public key $PK = (g, u, h, w, v, e(g,g)^{\alpha}, G, G_T)$ and keeps its own private key $A_{SK_i} = \{\alpha_i, (u^{A_j} h)_{A_j\in\hat{A}_i}, MK_i\}$.
3. *KeyGen:* Each attribute authority $\hat{A}_i$ execute algorithm as follows:
   (a) Select $\theta_i \in Z_p$ at random, thereafter derive the elements of the secret key, denoted as $MK_i \cdot g^{\theta_i}, MK_i \cdot v^{-\theta_i}, MK_i \cdot g^{\alpha_i} \cdot w^{\theta_i}$ and subsequently convey these elements to the pertinent attribute authorities.

(b) Upon obtaining the components from various attribute authorities, proceed to compute the secret key utilizing the following steps:

$$K_0 = \prod_{i=1}^{N} MK_i \cdot g^{\alpha_i} \cdot w^{\theta_i} = g^{\sum_{i=1}^{N} \alpha_i} w^r \tag{3}$$

$$K_1 = \prod_{i=1}^{N} MK_i \cdot g^{\theta_i} = g^{\sum_{i=1}^{N} \theta_i} = g^r \tag{4}$$

$$K_v = \prod_{i=1}^{N} MK_i \cdot v^{-\theta_i} = v^{-r} \tag{5}$$

(c) For each attribute $\sigma \in [S_{ID} \cap \hat{A}_i]$, randomly choose $r_\sigma \in Z_p$, where $\sigma \leq N$ and S_{ID} denotes the set of users. Calculate $K_{i,2} = g^{r_i}$, $K_{i,3} = (u^{A_i} h)^{r_i} \cdot K_v = (u^{A_i} h)^{r_i} v^{-r}$.
Then user gets the secret key $SK = \{K_0, K_1, \{K_{i,2}, K_{i,3}\}_{i\in[1,\sigma]}, S_{ID}\}$.

4. *KeyGen.ran:* Upon inputting SK, the data user independently selects a random element from the finite field $\tau \in Z_p$, and proceeds to calculate $K_0' = {K_0}^{1/\tau} = g^{\alpha/\tau} w^{r/\tau}$, $K_1' = {K_1}^{1/\tau} = g^{r/\tau}$. For $i = 1, 2, \ldots, \sigma$, the data user calculates $K_{i,2}' = {K_{i,2}}^{1/\tau} = g^{r_i/\tau}$, $K_{i,3}' = {K_{i,3}}^{1/\tau} = (u^{A_i} h)^{r_i/\tau} v^{-r/\tau}$. The transformation key, designated as $TK = (S_{ID}, K_0', K_1', \{K_{i,2}', K_{i,3}'\}_{i\in[1,\sigma]})$ and the recovery key, denoted as $RK = \tau$, serve distinct functions within the cryptographic framework.
5. *Enc.Offline:* Enter the PK, and let N' denote the upper limit on the count of rows within the secret sharing matrix. The data owner randomly chooses $s \in Z_p$, calculates $\hat{C} = e(g,g)^{\alpha s}, \hat{C}_0 = g^s$. For $j = 1, 2, \ldots, N'$, the data owner randomly chooses $d_j \in Z_p$ and calculates $\hat{C}_{j,1} = v^{d_j}, \hat{C}_{j,2} = h^{-d_j}, \hat{C}_{j,3} = g^{d_j}$. The intermediate ciphertext $MT = (s, \hat{C}, \hat{C}_0, \{d_j, \hat{C}_{j,1}, \hat{C}_{j,2}, \hat{C}_{j,3}\}_{j\in[1,N']})$.
6. *Enc.Online:* Input MT, plaintext m, access structure (M, ρ), where M is a matrix of l rows and n columns ($l \leq N'$). The data owner randomly chooses vector $\vec{y} = (s, y_2, \ldots, y_n) \in Z_p^{n\times 1}$. The secret share is $\vec{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_l)^T = M\vec{y}$. Then the data owner calculates $Token = H_0(m)$, $C = m \cdot \hat{C} = m \cdot e(g,g)^{\alpha s}$, $C_0 = \hat{C}_0 = g^s$. For $j = 1, 2, \ldots, l$, data owner computes $C_{j,1} = \hat{C}_{j,1} \cdot w^{\lambda_j} = w^{\lambda_j} v^{d_j}, C_{j,2} = \hat{C}_{j,2} \cdot u^{-\rho(j) d_j} = (u^{-\rho(j)} h)^{-d_j}, C_{j,3} = \hat{C}_{j,3} = g^{d_j}$. The ciphertext $CT = ((M,\rho), C, C_0, \{C_{j,1}, C_{j,2}, C_{j,3}\}_{j\in[1,l]})$ and the verification credential is $Token$.
7. *Dec.Out:* If the user's attributes set, identified by S_{ID}, does not conform to the access structure, the cloud server will return a null value $\perp$ and terminate the algorithm. Otherwise, cloud server collects $I = \{i, \rho(i) \in S_{ID}\}$ and calculates $\{\omega_i \in Z_p\}_{i\in I}$, where $\sum_{i\in I} \omega_i \cdot M_i = (1, 0, \ldots, 0)$ and M_i is the ith row of matrix M. Then the cloud server calculates

$$A = \frac{e(C_0, K_0')}{\prod_{i\in I} (e(C_{i,1}, K_1') \cdot e(C_{i,2}, K_{j,2}') \cdot e(C_{i,3}, K_{j,3}'))^{\omega_i}} = e(g,g)^{\alpha s/\tau}, \tag{6}$$

in the given context, j represents the position or identifier for the attribute value $\rho(i)$ in S_{ID} ().
8. *Dec.User:* The data user uses the conversion key RK to decrypt as follows:

$$\frac{C}{A^\tau} = \frac{e(g,g)^{\alpha s} m}{(e(g,g)^{\alpha s/\tau})^\tau} = m, \tag{7}$$

then data user uses the verification credential $Token$ to complete the ciphertext verification, if $H_0(m) = Token$ holds, the ciphertext is correct. Otherwise, the ciphertext may have been tampered with.

4.2. CRF scheme

1. *Initialization:* The attribute authorities runs $GlobalSetup$ and $AASetup$, each attribute authority sends α_i to $\mathcal{W}_{AA}$, then $\mathcal{W}_{AA}$ executes algorithms as follows:
$\mathcal{W}_{AA}.SetUp$: Upon receiving the parameters from AA, the CRF $\mathcal{W}_{AA}$ calculates $\alpha = \sum_{i=1}^{N} \alpha_i$, then randomly chooses $a, b, c, d, e, f \in Z_p$ and calculates $g' = g^a$, $u' = u^b$, $h' = h^c$, $w' = w^d$, $v' = v^e$, $\alpha' = \alpha + f$, $e(g', g')^{\alpha'} = e(g,g)^{a^2(\alpha+f)}$. $\mathcal{W}_{AA}$ stores f and publishes the updated $PK' = (g', u', h', w', v', e(g', g')^{\alpha'}, G, G_T)$. After receiving PK', AA executes $KeyGen$ to generate secret key $SK = \{K_0, K_1, \{K_{i,2}, K_{i,3}\}_{i\in[1,\sigma]}, S_{ID}\}$ and sends SK to CRF $\mathcal{W}_{AA}$. $\mathcal{W}_{AA}$ runs the following algorithm for re-randomization.
$\mathcal{W}_{AA}.KG$: Provide PK', f and N as input, where N represents the total number of attributes. $\mathcal{W}_{AA}$ randomly selects $r', r_1', r_2', \ldots, r_N' \in Z_p$, calculates $\widetilde{K_0'} = g'^f w'^{r'}$, $\widetilde{K_1'} = g'^{r'}$. For $i = 1, 2, \ldots, N$, W_{AA} computes $\widetilde{K_{i,2}'} = g'^{r_i'}$, $K_v' = v'^{-r'}$, $\widetilde{K_{i,3}'} = (u'^{A_i} h')^{r_i'} \cdot K_v' = (u'^{A_i} h')^{r_i'} v'^{-r'}$. The intermediate key $ZSK = (\widetilde{K_0'}, \widetilde{K_1'}, \{r_i', \widetilde{K_{i,2}'}, \widetilde{K_{i,3}'}\}_{i\in[1,N]})$.
Eventually, $\mathcal{W}_{AA}$ computes $K_0' = K_0 \cdot \widetilde{K_0'} = g'^{\alpha+f} w'^{r+r'} = g'^{\alpha'} w'^{r+r'}$, $K_1' = K_1 \cdot \widetilde{K_1'} = g'^{r+r'}$. For $i = 1, 2, \ldots, \sigma$, where $\sigma \leq N$, $\mathcal{W}_{AA}$ calculates $K_{i,2}' = K_{i,2} \cdot \widetilde{K_{i,2}'} = g'^{r_i + r_i'}$, $K_{i,3}' = K_{i,3} \cdot \widetilde{K_{i,3}'} = (u'^{A_i} h')^{r_i + r_i'} v'^{-r-r'}$. $\mathcal{W}_{AA}$ sends the updated $SK' = (K_0', K_1', \{K_{i,2}', K_{i,3}'\}_{i\in[1,\sigma]}, S_{ID})$ to data user.
2. *Data Upload:* The data owner invokes the $Enc.Offline$ and $Enc.Online$ to obtain ciphertext $CT = ((M,\rho), C, C_0, \{C_{j,1}, C_{j,2}, C_{j,3}\}_{j\in[1,l]})$ and verification credential $Token$, then sends CT and $Token$ to CRF $\mathcal{W}_{DO}$, $\mathcal{W}_{DO}$ executes algorithm as follows:
$\mathcal{W}_{DO}.Enc.Offline$: Input PK' and N', the notation N' is used to represent the highest possible number of rows that are allowed in the access structure. $\mathcal{W}_{DO}$ randomly chooses $s' \in Z_p$ as secret value and calculates $\hat{C}' = e(g', g')^{\alpha' s'}, \hat{C}_0' = g'^{s'}$. For $j = 1, 2, \ldots, N'$, $\mathcal{W}_{DO}$ randomly chooses $d_j' \in Z_p$ and calculates $\hat{C}_{j,1}' = v'^{d_j'}$, $\hat{C}_{j,2}' = h'^{-d_j'}$, $\hat{C}_{j,3}' = g'^{d_j'}$. Enter the transitional encryption, denoted as $MT' = (s', \hat{C}', \hat{C}_0', \{\hat{C}_{j,1}', \hat{C}_{j,2}', \hat{C}_{j,3}'\}_{j\in[1,N']})$.
$\mathcal{W}_{DO}.Enc.Online$: Input PK', MT' and CT. The CRF $\mathcal{W}_{DO}$ randomly selects vector $\vec{y'} = (s', y_2', ..., y_n')^T \in Z_p^{n\times 1}$, then secret shared vectors $\vec{\lambda'} = (\lambda_1', \ldots, \lambda_n')^T = M\vec{y'}$. Then $\mathcal{W}_{DO}$ computes $C' = C \cdot \hat{C}' = m \cdot e(g', g')^{\alpha'(s+s')}$, $C_0' = C_0 \cdot \hat{C}_0' = g'^{s+s'}$. For $j = 1, 2, \ldots, l$, where $l \leq N'$, $\mathcal{W}_{DO}$ calculates

$$C_{j,1}' = C_{j,1} \cdot \hat{C}_{j,1}' \cdot w'^{\lambda_j'} = w'^{\lambda_j + \lambda_j'} v'^{d_j + d_j'}, \tag{8}$$

$$C_{j,2}' = C_{j,2} \cdot \hat{C}_{j,2}' \cdot u'^{-\rho(j) d_j'} = (u'^{\rho(j)} h')^{-(d_j + d_j')}, \tag{9}$$

$$C_{j,3}' = C_{j,3} \cdot \hat{C}_{j,3}' = g'^{d_j + d_j'}. \tag{10}$$

The $\mathcal{W}_{DO}$ transmits the ciphertext $CT' = (C', C_0', \{C_{j,1}', C_{j,2}', C_{j,3}'\}_{j\in[1,l]}, (M,\rho))$, which has been re-randomized, along with the $Token$, to the cloud server.
3. *Data Download:* The data user runs $KenGen.ran(SK')$ and sends $TK = (S_{ID}, K_0'', K_1'', \{K_{i,2}'', K_{i,3}''\}_{i\in[1,\sigma]})$ to CRF $\mathcal{W}_{DU}$. Then $\mathcal{W}_{DU}$ executes algorithm as follows:
$\mathcal{W}_{DU}.TKUpdate$: $\mathcal{W}_{DU}$ randomly chooses $\varphi \in Z_p$ and calculates

$$K_0''' = {K_0''}^{1/\varphi} = g'^{\alpha'/\tau\varphi} w'^{(r+r')/\tau\varphi}, \tag{11}$$

$$K_1''' = {K_1''}^{1/\varphi} = g'^{(r+r')/\tau\varphi}, \tag{12}$$

$$K_{i,2}''' = {K_{i,2}''}^{1/\varphi} = g'^{(r_i + r_i')/\tau\varphi}, \tag{13}$$

$$K_{i,3}''' = {K_{i,3}''}^{1/\varphi} = (u'^{A_i} h')^{(r_i + r_i')/\tau\varphi} v'^{-(r+r')/\tau\varphi}. \tag{14}$$

$\mathcal{W}_{DU}$ stores $\varphi \in Z_p$ and sends re-randomize conversion key $TK' = (S_{ID}, K_0''', K_1''', \{K_{i,2}''', K_{i,3}'''\}_{i\in[1,\sigma]})$ to the cloud server. When receiving a decryption request from a data user, the cloud server performs $Dec.Out(TK', CT')$ to acquire a partially decrypted ciphertext TCT. The cloud server sends $TCT = (C', A = e(g', g')^{\alpha'(s+s')/\tau\varphi})$ and $Token$ to $\mathcal{W}_{DU}$, $\mathcal{W}_{DU}$ runs algorithms as follows.

$\mathcal{W}_{DU}.Dec$: The CRF $\mathcal{W}_{DU}$ computes $A' = A^{\varphi} = e(g', g')^{\alpha'(s+s')/\tau}$ and sends $TCT' = (C', A')$ and $Token$ to the data user.

After receiving re-randomize partially decrypted ciphertext, data user runs $Dec.User$ to recover plaintext m. Then the data user uses the verification credential $Token$ to finish the ciphertext verification, if $H_0(m) = Token$ holds, the ciphertext is correct.

5. Security analysis

5.1. Security proof

Theorem 1. *Given that the q-BDHE assumption holds true, the proposed scheme is deemed secure against selective CPA.*

Proof. If a polynomial-time adversary B can effectively compromise the proposed scheme with a significant advantage, then we can develop a challenger F to solve the q-BDHE problem with a significant advantage. The process is as follows:

Init Phase: The adversary B submits access policies $(M_i{}^*, \rho_i{}^*)_{i\in I^*}$ and a set of malicious attribute authorities $R = (\hat{A}_i)_{i\in I}$, where $M_i{}^*$ is a $l * n$ matrix. Furthermore, the attributes within the access structure must originate from trusted attribute authorities and cannot be maliciously manipulated.

Setup Phase: The challenger F executes algorithms *AASetup* and *GlobalSetup* to generate public parameter $Params = \{g, u, v, w, h, G, G_T, H_0()\}$ and private keys $(PK_i, A_{SKi})_{i\in I}$. The reverse firewall $\mathcal{W}_{AA}$ executes the algorithm $\mathcal{W}_{AA}.SetUp$ to re-random public key, then $\mathcal{W}_{AA}$ publishes updated public key PK'.

Query Phase 1: During this phase, B can dynamically request secret keys for attribute sets $S_1, S_2, \ldots, S_q$. For every query S_i, F executes algorithm *KeyGen* to obtain corresponding secret key SK_i. Then F executes algorithm $\mathcal{W}_{AA}.KG$ to get re-randomized secret key SK_i'. Subsequently, F executes *KeyGen.ran* to get conversion key TK_i. Then F runs $\mathcal{W}_{DU}.TKUpdate$ to get re-randomized conversion key TK_i'. C returns (SK_i', TK_i') to B.

Challenge Phase: B provides two messages, m_0 and m_1, of equal length. F randomly selects $b \in \{0, 1\}$ and runs *Enc.Offline** and *Enc.Online** to get challenge ciphertext $CT_b = ((M, \rho), C, C_0, \{C_{j,1}, C_{j,2}, C_{j,3}\}_{j\in[1,l]})$.

Then F executes $\mathcal{W}_{DO}.Enc.Offline$ and $\mathcal{W}_{DO}.Enc.Online$ Obtain a ciphertext CT_b'. F that has been re-randomized sends CT_b' to B.

Query Phase 2: The challenger F proceeds as in *Query Phase 1*.

Guess Phase: B outputs a bit $b' \in \{0, 1\}$. If $b' = b$, then F outputs 0 (meaning that B obtains the normally generated ciphertext). If $b' \neq b$, then F outputs 1(meaning that B obtains the randomly selected element). Hence, the adversary B has advantage of ϵ security game directly correlates to the ability of function F to resolve the q-BDHE problem with the same level of probability.

5.2. Security analysis

The features of the proposed scheme include:

1. *Function Maintaining*
   If the collection of attributes associated with the secret key constitutes an authorized set, then the equation $\sum_{i\in I} \omega_i \cdot (\lambda_i + \lambda_i{}') = s + s'$ holds. Thus,

$$
\begin{aligned}
A' &= \frac{e(C_0', K_0''')}{\prod_{i\in I}\left(e(C_{i,1}', K_1''') \cdot e(C_{i,2}', K_{j,2}''') \cdot e(C_{i,3}', K_{j,3}''')\right)^{\omega_i}} \\
&= \frac{e(g', g')^{\alpha'(s+s')/\tau\varphi}}{\prod_{i\in I} e(g', w')^{(r+r')(\lambda_i+\lambda_i{}')\omega_i/\tau\varphi}} \cdot \frac{e(g', w')^{(r+r')(s+s')/\tau\varphi}}{\prod_{i\in I} e(g', v')^{(r+r')(d_i+d_i{}')\omega_i/\tau\varphi}} \\
&\cdot \frac{1}{\prod_{i\in I} e(g', u')^{-\rho(i)(d_i+d_i{}')(r_i+r_i{}')\omega_i/\tau\varphi}} \\
&\cdot \frac{1}{\prod_{i\in I} e(g', h')^{-(d_i+d_i{}')(r_i+r_i{}')\omega_i/\tau\varphi}} \\
&\cdot \frac{1}{\prod_{i\in I} e(g', u')^{A_i(d_i+d_i{}')(r_i+r_i{}')\omega_i/\tau\varphi}} \\
&\cdot \frac{1}{\prod_{i\in I} e(g', h')^{(d_i+d_i{}')(r_i+r_i{}')\omega_i/\tau\varphi}} \cdot \frac{1}{\prod_{i\in I} e(g', v')^{-(r+r')(d_i+d_i{}')\omega_i/\tau\varphi}}
\end{aligned} \tag{15}
$$

$$
= \frac{e(g', g')^{\alpha'(s+s')/\tau\varphi} e(g', w')^{(r+r')(s+s')/\tau\varphi}}{e(g', w')^{(r+r')\sum_{i\in I}(\lambda_i+\lambda_i{}')\omega_i/\tau\varphi}} = e(g', g')^{\alpha'(s+s')/\tau\varphi}. \tag{16}
$$

$$
\frac{C'}{A'^{\tau}} = \frac{C'}{A^{\varphi\tau}} = \frac{m \cdot e(g', g')^{\alpha'(s+s')/\tau}}{e(g', g')^{\alpha'(s+s')/\tau}} = m \tag{17}
$$

It is evident from the aforementioned equations that the message 'm' remains decryptable under normal circumstances even after the implementation of a cryptographic reverse firewall. Consequently, the functionality of the cryptographic reverse firewalls is preserved.

2. *Weakly Security-preserving and Weakly Exfiltration-resistant*
   We assume the following security game process.
   Game 0: Same as chapter 3 security games.
   Game 1: In the init phase, attribute authorities' PK, A_{SKi} are generated by algorithms *GlobalSetup* and *AASetup* of basic scheme, not *GlobalSetup*, AASetup** and $\mathcal{W}_{AA}.SetUp$. The subsequent algorithms are carried over unchanged from Game 0.
   Game 2: During both phase 1 and phase 2, the secret key SK is derived from the *KeyGen* algorithm of the foundational scheme, rather than being produced by *KeyGen** or the $\mathcal{W}_{AA}.KG$. The TK is produced using the *KeyGen.ran* function of the underlying scheme, and not through *KeyGen.ran** or the $\mathcal{W}_{DU}.TKUpdate$. The subsequent algorithms mirror those utilized in Game 1.
   Game 3: During the challenge phase, the ciphertext labeled as CT_b is constructed through the process of encryption denoted by *Enc.offline, Enc.online*, not *Enc.offline*, Enc.online**, $\mathcal{W}_{DO}.Enc.offline$ and $\mathcal{W}_{DO}.Enc.online$. Actually, Game 3 is the security game of basic scheme.
   We then proceed to demonstrate the indistinguishability between Game 0 and Game 1, followed by Game 1 and Game 2, and finally between Game 2 and Game 3, each in isolation. Between Game 0 and Game 1, it is observed that no matter the modifications introduced by the tampered *GlobalSetup** and, AASetup* algorithms, after the application of re-randomization via the W_{AA} reverse firewall, the public parameter PK' always corresponds to the structure of the PK that is generated by the standard algorithm. This uniformity is due to the malleability of the key in question. Consequently, there is no distinguishable difference between Game 0 and Game 1.
   Given that the secret key SK and the conversion key TK, which are produced for the user by the attribute authority, also possess malleability, it follows that Game 1 and Game 2 are indistinguishable. When it comes to Game 2 and Game 3, the CT will undergo rerandomization by the reverse firewall, resulting in a new ciphertext CT', a process that is a consequence of the ciphertext's malleable nature. Thus, regardless of how the *Enc.offline** and Enc.online* algorithms operate, the ultimate configuration of the ciphertext aligns with that of the basic scheme's ciphertext structure. Consequently, there is no distinguishable difference between Game 2 and Game 3. In summary,

Table 1
Function comparison.

Scheme	With CRFs	Outsource	Offline encryption	Multi-authority	Ciphertext verification	Access structure
Guo et al. [25]	×	✓	✓	×	×	Tree
Chaudhary et al. [28]	×	✓	×	✓	×	LSSS
Hong et al. [31]	✓	×	×	✓	×	LSSS
Zhong et al. [29]	×	✓	×	×	×	Tree
Zhao et al. [32]	✓	✓	✓	×	×	Tree
Jin et al. [33]	✓	×	×	×	×	LSSS
Elhabob et al. [34]	✓	×	×	×	✓	Tree
Ours	✓	✓	✓	✓	✓	TREE

we deduce that Game 0 and Game 3 are equivalent in terms of their indistinguishability. Given that the foundational scheme is secure, it follows that the proposed scheme is also secure.

3. *Message Verification*
The data user(vehicle/RSU) use parameters $Token, m$ and hash function $H_0()$ to check whether equation $H_0(m) = Token$ holds true. With the help of the verification procedure described, the data user can identify any tampering that may have occurred with the message. Additionally, it provides assurance regarding the completeness and dependability of the received message. If the message changes, the equation will not holds. Therefore, the proposed scheme supports the message verification.
4. *Collusion Resistance*

Theorem 2. *Should the difficulty of the discrete logarithm problem remain uncompromised, the proposed scheme can defend against collusion attacks initiated by up to $N-1$ attribute authorities.*

According to the encryption process, each attribute authority randomly chooses $s_{ik} \in Z_p$ and attribute authority extends the value $g^{s_{ik}}$ to all the other attribute authorities involved. Given the difficulty inherent in the discrete logarithm problem, it would be problematic for an adversary B to deduce s_{ik} from $g^{s_{ik}}$ alone. Hence, even with the combined efforts of $N-2$ attribute authorities working in tandem with the adversary, guessing a valid MK_i remains an unattainable task for the adversary. Consequently, the adversary cannot devise a valid secret key SK. This renders the proposed scheme resistant to collusion attacks carried out by $N-1$ attribute authorities.

5.3. Informal security analysis

1. *Side channel attack defenses*
The proposed scheme utilizes CRF technology, which significantly reduces the computational overhead while enhancing security. By leveraging CRF, it reduces the risk of messages being attacked and complicates potential threats. In addition, multi-authorization technology maximizes the security of the entire system, effectively preventing single-point leakage, while balancing power consumption and execution time. These two methods not only improve the efficiency, but also provide strong protection against side channel attacks.
In short, the scheme effectively combines efficiency and enhanced security, making it suitable for secure communication in vehicular networks that are susceptible to side channels.
2. *Man-in-the-Middle attack defense0*
The proposed scheme uses CP-ABE technology. This technique uses a ciphertext policy, which embeds the access policy into the ciphertext. This improves the security and flexibility of access control and reduces the risk of man-in-the-middle attack (MITI) due to identity forgery.
In addition, we enhance the CRF module by integrating key parameter re-randomization within the multi-authority ABE framework. In addition, the proposed scheme also supports message integrity verification, easily executable by onboard terminals using simple hash functions.
By combining the above technologies, this method not only protects the communication channel, but also improves the security of information.

6. Performance evaluation

6.1. Experimental setup

The following outlines the hardware and software contexts utilized for conducting the experiment:

- The experimental apparatus consists of a desktop computer equipped with a 3.2 GHz AMD Ryzen 5 5600x CPU, 16 GB of RAM, and runs the Windows 11 Professional (x64) OS.
- The experimental schemes are realized using Java 8 and the JPBC 2.0.0 library [32]. The prime-order bilinear pairings are constructed upon a 160-bit elliptic curve group, which is founded on the equation $y^2 = x^3 + x$.

6.2. Theoretical analysis

Table 1 provides a side-by-side comparison to examine the functionality of our proposed scheme in relation to other schemes. Scheme [25] supports outsourced decryption and online encryption, but the rest of the functionality is not realized. Scheme [28] introduced multiple authorities to protect against collusion attacks. Scheme [29] only provides outsource decryption, thus the efficiency of encryption phase is not good enough. Scheme [31–34], add CRF modules between entities based on the above schemes. However, these schemes either do not have outsourced decryption or do not have multiple attribute authorities, which has some disadvantages. Our scheme provides both of these features, taking into account both efficiency and security. Through comparison, we can find that the proposed scheme adds cryptographic reverse firewalls between entities. By employing these firewalls, the system is fortified with a layer of defense that maintains its functional integrity against potential subversion attacks and any attempts to tamper with its algorithms.

The introduction of multi-attribute authorities ensures that the system is resistant to collusion attacks. The proposed scheme also provides outsourcing decryption as well as offline encryption, which requires low computation for the users to obtain the ciphertext. Additionally, verification credentials empower users to check and ensure the ciphertext's integrity.

The following notations are applied within Tables 2 and 3 are as follows: E signifies an exponential operation, and P denotes a bilinear pairing operation. In the given context, M signifies the number of rows in a matrix as well as the number of leaf nodes in an access tree. The symbol l is used to denote the total number of attributes possessed by users, while k signifies the minimum number of attributes from the access structure required to fulfill the decryption criteria.

As shown in Table 2, our scheme is in the middle of the $KeyGen$ phase. However, our scheme achieves the lowest computational overhead in the $Enc.Online$ phase. In the $Dec.Out$ phase, our scheme does not achieve significant advantages. But in $Dec.User$ phase, our scheme requires only a single exponential operation, reaches a constant level of computational overhead.

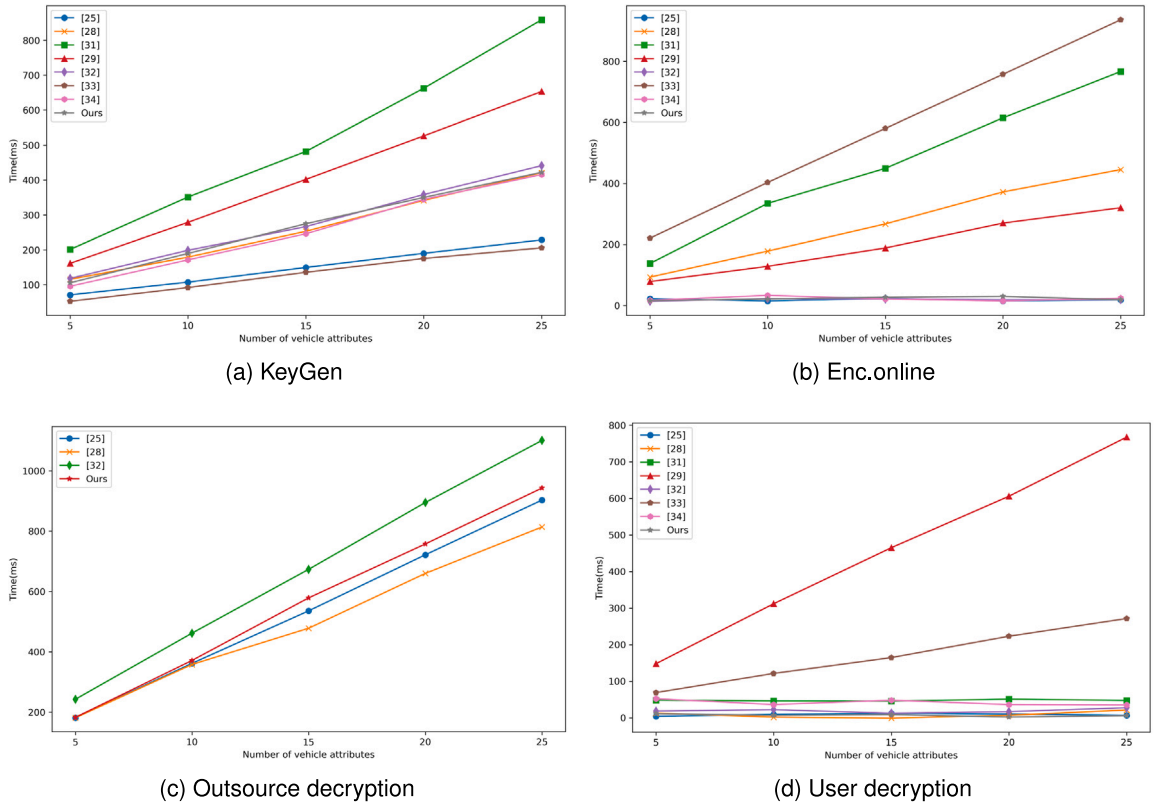

Fig. 3. Time consumption of basic scheme.

Table 2
Computation comparison.

Scheme	KeyGen	Encryption		Outsource decryption	User decryption
		Offline	Online		
Guo et al. [25]	$(l+4)E$	$(3M+1)E$	$3E$	$2lE+2lP$	E
Chaudhary et al. [28]	$(2l+2)E$	×	$(3M+1)E$	$(4l+2)E$	E
Zhong et al. [29]	$(3l+6)E$	×	$(2M+2)E$	×	$2lE+(l+1)P$
Hong et al. [31]	$(4l+2)E+P$	×	$(5M+2)E$	×	$E+(3k+1)P$
Zhao et al. [32]	$(2l+4)E$	$3ME+P$	$3E$	$(3l+1)E+(2l+1)P$	$2E$
Jin et al. [33]	$lE+P$	×	$6ME+3P$	×	$lE+2P$
Elhabob et al. [34]	$(2l+2)E$	×	$4E$	×	$3E$
Ours	$(2l+3)E$	$(2M+2)E$	$3E$	$lE+3lP$	E

Table 3
Time consumption of CRFs.

Scheme	$\mathcal{W}_{AA}.SetUp$	$\mathcal{W}_{AA}.KG$	$\mathcal{W}_{DO}.Enc.Online$
Hong et al. [31]	$2lE+2lP$	$(5l+2)E$	$2lE+P$
Zhao et al. [32]	$2E$	$(2l+3)E$	$4E$
Jin et al. [33]	$(l+2)E$	$(2l+2)E$	P
Elhabob et al. [34]	$2E$	$(2l+3)E$	$4E$
Ours	$5E$	$(2l+3)E$	$2E$

In terms of CRFs' time consumption, our scheme achieves time consumption of constant level in $\mathcal{W}_{AA}.SetUp$ phase as illustrated in 3, the time overhead does not fluctuate based on the count of attributes within the system. Moreover, our scheme achieves the highest efficiency in terms of the $\mathcal{W}_{DO}.Enc.Online$ phase, and requires only two exponential operations.

6.3. Practical analysis

In light of the hardware and software environment described within the *xperimental Setup* section, Fig. 3 presents a performance comparison of the multiple phases of our scheme.

Fig. 3(a) demonstrates that our scheme has a low computational overhead., is observed to be low. As shown in Fig. 3(b), when comparing the computational overhead of the $Enc.Online$ phase, our scheme, which benefits from the preprocessing performed in the $Enc.Offline$ phase, has the lowest computational overhead of all the schemes evaluated. In terms of Fig. 3(c), the efficiency of our scheme is in the middle of the $Dec.Out$ phase. While in the $Dec.User$ phase, our scheme maintains the lowest computational overhead, It is also significant to observe that the overhead does not fluctuate with varying counts of attributes in the system.

As depicted in Fig. 4, there is a performance comparison for the re-randomization of secret keys by CRF $\mathcal{W}_{AA}$. Our scheme's computational overhead is similar to that of scheme [32], which is at the lower level. Moreover, as shown in Fig. 5, the computational overhead of our scheme in the $\mathcal{W}_{DO}.Enc.Online$ phase is the most efficient and does not escalate linearly with an increase in vehicle attributes, which is a distinct advantage over other scheme [31]. And compared with [33, 34], the proposed scheme still has an advantage in the computational overhead of $\mathcal{W}_{AA}.SetUp$ phase.

In summary, our scheme reduces resource consumption on the user side and improves the efficiency of data flow in vehicles with limited computing power.

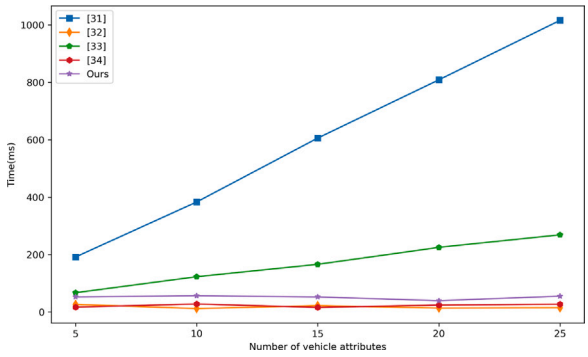

Fig. 4. Time consumption of $\mathcal{W}_{AA}.SetUp$.

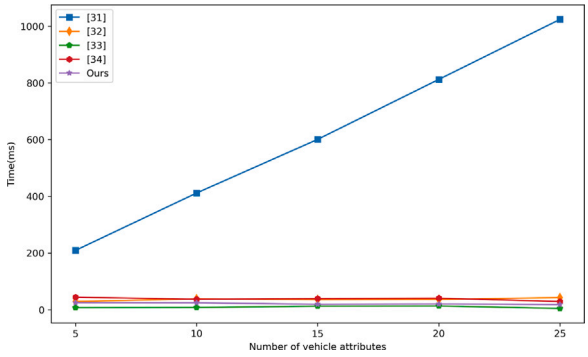

Fig. 5. Time consumption of $\mathcal{W}_{DO}.Enc.Online$.

7. Conclusion

In the IoV environment, securing the encryption and sharing of the vast amounts of data generated by vehicles, while preventing data leakage due to device tampering, presents significant challenges. To address these challenges, we propose an advanced attribute-based encryption scheme, enhanced with a cryptographic reverse firewall, specifically designed for the IoV ecosystem. This scheme is supported by multiple attribute authorities, which not only defend against collusion attacks but also enable offline encryption and outsourced decryption. These integrated features greatly improve the computational efficiency of vehicular onboard units. Additionally, we deploy RSUs with CRFs between the entities, ensuring that data remains secure even in the event of device tampering. The proposed attribute-based encryption scheme, combined with the reverse firewall mechanism, shows great promise in securing data transmission and storage within the IoV, while protecting against unauthorized access and data leakage.

CRediT authorship contribution statement

Xiaodong Yang: Writing – review & editing, Writing – original draft. **Xilai Luo:** Writing – review & editing, Writing – original draft. **Zefan Liao:** Writing – review & editing, Writing – original draft. **Wenjia Wang:** Writing – review & editing, Writing – original draft. **Xiaoni Du:** Writing – review & editing, Writing – original draft. **Shudong Li:** Writing – review & editing, Writing – original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by Key project of Gansu Science and Technology Plan (23YFGA0081), Gansu Province College Industry Ssupport Plan (2023CYZC-09), National Natural Science Foundation of China (No. 62362059).

Data availability

The authors do not have permission to share data.

References

[1] Siyi Liao, Jun Wu, Jianhua Li, Ali Kashif Bashir, Shahid Mumtaz, Alireza Jolfaei, Nida Kvedaraite, Cognitive popularity based AI service sharing for software-defined information-centric networks, IEEE Trans. Netw. Sci. Eng. 7 (4) (2020) 2126–2136.

[2] Rich Miller, Rolling zettabytes: Quantifying the data impact of connected cars, Data Cent. Front. (2020).

[3] Kayhan Zrar Ghafoor, Linghe Kong, Sherali Zeadally, Ali Safaa Sadiq, Gregory Epiphaniou, Mohammad Hammoudeh, Ali Kashif Bashir, Shahid Mumtaz, Millimeter-wave communication for internet of vehicles: status, challenges, and perspectives, IEEE Internet Things J. 7 (9) (2020) 8525–8546.

[4] Soheila Ghane, Alireza Jolfaei, Lars Kulik, Kotagiri Ramamohanarao, Deepak Puthal, Preserving privacy in the internet of connected vehicles, IEEE Trans. Intell. Transp. Syst. 22 (8) (2020) 5018–5027.

[5] Liang Zhao, Hongmei Chai, Yuan Han, Keping Yu, Shahid Mumtaz, A collaborative V2X data correction method for road safety, IEEE Trans. Reliab. 71 (2) (2022) 951–962.

[6] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, Lanyu Xu, Edge computing: Vision and challenges, IEEE Internet Things J. 3 (5) (2016) 637–646.

[7] Zhenyu Zhou, Haijun Liao, Bo Gu, Shahid Mumtaz, Jonathan Rodriguez, Resource sharing and task offloading in IoT fog computing: A contract-learning approach, IEEE Trans. Emerg. Top. Comput. Intell. 4 (3) (2019) 227–240.

[8] Xingwang Li, Zhen Xie, Zheng Chu, Varun G Menon, Shahid Mumtaz, Jianhua Zhang, Exploiting benefits of IRS in wireless powered NOMA networks, IEEE Trans. Green Commun. Netw. 6 (1) (2022) 175–186.

[9] Vipul Goyal, Omkant Pandey, Amit Sahai, Brent Waters, Attribute-based encryption for fine-grained access control of encrypted data, in: Proceedings of the 13th ACM Conference on Computer and Communications Security, 2006, pp. 89–98.

[10] Amit Sahai, Brent Waters, Fuzzy identity-based encryption, in: Advances in Cryptology–EUROCRYPT 2005: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005. Proceedings 24, Springer, 2005, pp. 457–473.

[11] John Bethencourt, Amit Sahai, Brent Waters, Ciphertext-policy attribute-based encryption, in: 2007 IEEE Symposium on Security and Privacy, SP'07, IEEE, 2007, pp. 321–334.

[12] Matthew Green, Susan Hohenberger, Brent Waters, Outsourcing the decryption of {abe} ciphertexts, in: 20th USENIX Security Symposium, USENIX Security 11, 2011.

[13] Junzuo Lai, Robert H. Deng, Chaowen Guan, Jian Weng, Attribute-based encryption with verifiable outsourced decryption, IEEE Trans. Inf. Forensics Secur. 8 (8) (2013) 1343–1354.

[14] Suqing Lin, Rui Zhang, Hui Ma, Mingsheng Wang, Revisiting attribute-based encryption with verifiable outsourced decryption, IEEE Trans. Inf. Forensics Secur. 10 (10) (2015) 2119–2130.

[15] Cong Zuo, Jun Shao, Guiyi Wei, Mande Xie, Min Ji, CCA-secure ABE with outsourced decryption for fog computing, Future Gener. Comput. Syst. 78 (2018) 730–738.

[16] James Ball, Julian Borger, Glenn Greenwald, et al., Revealed: how US and UK spy agencies defeat internet privacy and security, Know Your Neighb. (2013).

[17] Stephen Checkoway, Ruben Niederhagen, Adam Everspaugh, Matthew Green, Tanja Lange, Thomas Ristenpart, Daniel J Bernstein, Jake Maskiewicz, Hovav Shacham, Matthew Fredrikson, On the practical exploitability of dual {ec} in {tls} implementations, in: 23rd USENIX Security Symposium, USENIX Security 14, 2014, pp. 319–335.

[18] Yevgeniy Dodis, Chaya Ganesh, Alexander Golovnev, Ari Juels, Thomas Ristenpart, A formal treatment of backdoored pseudorandom generators, in: Advances in Cryptology–EUROCRYPT 2015: 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I 34, Springer, 2015, pp. 101–126.

[19] Ilya Mironov, Noah Stephens-Davidowitz, Cryptographic reverse firewalls, in: Advances in Cryptology-EUROCRYPT 2015: 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part II 34, Springer, 2015, pp. 657–686.

[20] Brent Waters, Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization, in: International Workshop on Public Key Cryptography, Springer, 2011, pp. 53–70.
[21] Shucheng Yu, Cong Wang, Kui Ren, Wenjing Lou, Achieving secure, scalable, and fine-grained data access control in cloud computing, in: 2010 Proceedings IEEE INFOCOM, IEEE, 2010, pp. 1–9.
[22] Kan Yang, Xiaohua Jia, Kui Ren, Ruitao Xie, Liusheng Huang, Enabling efficient access control with dynamic policy updating for big data in the cloud, in: IEEE INFOCOM 2014-IEEE Conference on Computer Communications, IEEE, 2014, pp. 2013–2021.
[23] Jun Feng, Hu Xiong, Jinhao Chen, Yang Xiang, Kuo-Hui Yeh, Scalable and revocable attribute-based data sharing with short revocation list for IIoT, IEEE Internet Things J. 10 (6) (2022) 4815–4829.
[24] Qian Mei, Hu Xiong, Yeh-Cheng Chen, Chien-Ming Chen, Blockchain-enabled privacy-preserving authentication mechanism for transportation cps with cloud-edge computing, IEEE Trans. Eng. Manage. (2022).
[25] Rui Guo, Geng Yang, Huixian Shi, Yinghui Zhang, Dong Zheng, O 3-R-CP-ABE: An efficient and revocable attribute-based encryption scheme in the cloud-assisted IoMT system, IEEE Internet Things J. 8 (11) (2021) 8949–8963.
[26] Melissa Chase, Multi-authority attribute based encryption, in: Theory of Cryptography: 4th Theory of Cryptography Conference, TCC 2007, Amsterdam, the Netherlands, February 21-24, 2007. Proceedings 4, Springer, 2007, pp. 515–534.
[27] Allison Lewko, Brent Waters, Decentralizing attribute-based encryption, in: Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 2011, pp. 568–588.
[28] Chandan Kumar Chaudhary, Richa Sarma, Ferdous Ahmed Barbhuiya, RMA-CPABE: A multi-authority CPABE scheme with reduced ciphertext size for IoT devices, Future Gener. Comput. Syst. 138 (2023) 226–242.
[29] Hong Zhong, Yiyuan Zhou, Qingyang Zhang, Yan Xu, Jie Cui, An efficient and outsourcing-supported attribute-based access control scheme for edge-enabled smart healthcare, Future Gener. Comput. Syst. 115 (2021) 486–496.
[30] Hui Ma, Rui Zhang, Guomin Yang, Zishuai Song, Shuzhou Sun, Yuting Xiao, Concessive online/offline attribute based encryption with cryptographic reverse firewalls—Secure and efficient fine-grained access control on corrupted machines, in: Computer Security: 23rd European Symposium on Research in Computer Security, ESORICS 2018, Barcelona, Spain, September 3-7, 2018, Proceedings, Part II 23, Springer, 2018, pp. 507–526.
[31] Bo Hong, Jie Chen, Kai Zhang, Haifeng Qian, Multi-authority non-monotonic KP-ABE with cryptographic reverse firewall, IEEE Access 7 (2019) 159002–159012.
[32] Yang Zhao, Yuwei Pang, Xingyu Ke, Bintao Wang, Guobin Zhu, Mingsheng Cao, A metaverse-oriented CP-ABE scheme with cryptographic reverse firewall, Future Gener. Comput. Syst. 147 (2023) 195–206.
[33] Jin C., Chen Z., Qin W., et al., Blockchain-based proxy re-encryption scheme with cryptographic reverse firewall for IoV, Int. J. Netw. Manage. (2024) e2305.
[34] Elhabob R., Eltayieb N., Xiong H., et al., Equality test public key encryption with cryptographic reverse firewalls for cloud-based E-commerce, IEEE Trans. Consum. Electron. (2024).

Xiaodong Yang (Member, IEEE) received the M.S. degree in cryptography from Tongji University, Shanghai, China, in 2005, and the Ph.D. degree in cryptography from Northwest Normal University, Lanzhou, China, in 2010.

In his role as a Postdoctoral Researcher at China's State Key Laboratory of Cryptology in Beijing during 2016, he played a significant part in advancing the field. Today, he holds the position of Professor at the College of Computer Science and Engineering, Northwest Normal University. The core of his research is anchored in public-key cryptography, information security protocols, and the application of wireless sensor networks.

Xilai Luo is presently a master's degree candidate at the College of Computer Science and Engineering, Northwest Normal University, located in China. His academic pursuits are focused on the areas of artificial intelligence, information security, and cryptography.

Zefan Liao is actively working towards his master's degree in the College of Computer Science and Engineering at Northwest Normal University, China. His areas of research interest include the fields of edge computing, information security, and cryptography.

Wenjia Wang is pursuing her master's degree within the College of Computer Science and Engineering at Northwest Normal University, China. Her research interests are centered on the topics of data security and network security.

Xiaoni Du received the Ph.D. degree in cryptography from Xidian University, Xi'an, China, in 2008.

She worked as a Visiting Scholar with the University of Kentucky, Lexington, KY, USA, and Hong Kong University of Science and Technology, Hong Kong, in 2011 and 2014, respectively. She is currently a Professor with the College of Mathematics and Statistics, Northwest Normal University, Lanzhou, China. Her main research interests include information security, cryptography, and coding.

Shudong Li received the M.S. degree in applied mathematics from Tongji University, Shanghai, China, in 2005, and the Ph.D. degree in Posts and Telecommunications from Beijing University, Beijing, China, in 2012.

From 2013 to 2018, he held the position of a postdoctoral researcher at the National University of Defense Technology in Changsha, China. He now serves as a Professor at the Cyberspace Institute of Advanced Technology at Guangzhou University. His primary research interests are in the realms of Big Data and its security, malware identification, and cloud computing.