Article

Implementing an Efficient Secure Attribute-Based Encryption System for IoV Using Association Rules

Mu Han *, Mengli Zhu, Pengzhou Cheng, Zhikun Yin and Haixin Qu

School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, China; 2221908067@stmail.ujs.edu.cn (M.Z.); 2221908010@stmail.ujs.edu.cn (P.C.); 2221808044@stmail.ujs.edu.cn (Z.Y.); 3180604017@stmail.ujs.edu.cn (H.Q.)

* Correspondence: hanmu@ujs.edu.cn

Abstract: As the Internet of vehicles (IoV) is the perceptual information subject, the intelligent connected vehicle (ICV) is establishing an interconnected information transmission system through opening more external interfaces. However, security communication problems thereby are generated, attracting massive attention for researchers. Hence, the in-vehicle network system is responsible for controlling the state of the ICV and has a major impact on driving safety. In this paper, we designed an efficient secure ciphertext-policy attribute-based encryption (CP-ABE) system for protecting communication. The research focuses on mining the frequency features between vehicle nodes through the max-miner association rules algorithm, aiming to build frequent item sets. Furthermore, an improved asymmetric ABE scheme can implement secure communication in-vehicle nodes that belong to the same classification set. Through the hardware platform and in-vehicle network simulator (INVS) to evaluate our scheme, the results demonstrate that the work possesses enough security without reducing communication efficiency, meanwhile improve bus load performance.

Keywords: ICV; in-vehicle network; attribute-based encryption; association rules algorithm

Citation: Han, M.; Zhu, M.; Cheng, P.; Yin, Z.; Qu, H. Implementing an Efficient Secure Attribute-Based Encryption System for IoV Using Association Rules. *Symmetry* **2021**, *13*, 1177. <https://doi.org/10.3390/sym13071177>

Academic Editors:
Basil Papadopoulos and José Carlos R. Alcantud

Received: 26 May 2021
Accepted: 26 June 2021
Published: 29 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Figure 1. Model diagram of an IoV system in an intelligent transport environment.

As the core of IoV, intelligent connected vehicles (ICVs) are also attracting a lot of attention and developing rapidly. In order to facilitate the connection of vehicle to the Internet, ICV opens up a number of interfaces, for example, cellular connection, Wi-Fi, Bluetooth, USB and onboard diagnostics (OBD-II) [3]. However, this also increases the likelihood that the vehicle will be successfully attacked [4]. Through these interfaces, the attacker can intrusion the in-vehicle network and inject malicious messages, finally achieve illegal control of the vehicle. In recent years, various attacks incidents frequent occurrences, especially remote attacks [5,6]. For example, the attacker takes control of the vehicle by sending a forged message from the entertainment system to the CAN bus via a cellular network connected to the vehicle [7].

Initially, the in-vehicle communication protocol only considers some characteristics, such as reliability, comfort, and convenience but yet ignores a crucial point that the message is transmitted under a secure environment [8]. Hence, as the appicate in-vehicle communication bus, the controller area network (CAN) does not provide any security mechanisms. In addition, the broadcast transmission mechanism between electronic control units (ECUs) also increases a security risk for ICV. Additionally, the ECUs are the core electronic components of a smart networked vehicle and are considered to be the in-vehicle computer. ECUs communicate with sensors on the CAN network to control the driving state of the vehicle. It is also an untrustworthy exchange of information method. In a nutshell, the safe transmission of information is a prerequisite for safe driving, necessary to be considered.

In order to solve the security problem of in-vehicle networks, many researchers have provided solutions based on cryptography methods. These methods can be divided into two categories. One is to verify the integrity of the message and the origin of the data by means of a message authentication mechanism, for example by generating a MAC authentication message based on cryptography and transmitting an authentication message of equal length corresponding to this message on the CAN bus [9]. Another approach is to ensure the confidentiality of the message by means of an encryption mechanism, for example, by encrypting the data in groups using AES [10]. However, these approaches are based on information systems and ignore the fact that ICV is a physical information system, so they are inefficient or not applicable to in-vehicle networks. Therefore, it is considered necessary to combine the physical characteristics of smart vehicles with cryptographic methods to design secure and effective in-vehicle communication systems.

In this paper, we first analyze the communication list between ECUs of a vehicle brand, finding that ECU could receive all the data in the CAN, but it did not use all the data received. Hence, it is important for the message transmitting mechanism to make a pre-isolated communication scheme. Not only does it improve the efficiency of message transmission, but it also prevents other ECUs from being attacked by a break-in to affect themselves in terms of security. Furthermore, efficient encryption solutions are also necessary for data transmission to achieve integrity and confidentiality. In this paper, we perform a correlation analysis of communication frequencies based on real in-vehicle communication data and combines the analysis results with asymmetric ciphertext-policy attribute-based encryption (CP-ABE) algorithms [11,12] to design an in-vehicle access control strategy to achieve isolation of ECUs without communication needs.

The main contributions of this paper are as follows:

- (1) First, we collect the ECUs communication list of a company. Through the max-miner association rules algorithm, the frequency features are mined between vehicle nodes. The ECU whose communication frequency reaches the threshold value is grouped into one category and regarded as having the same frequency attribute. We find that this part filter unnecessary communications, laying the foundation for the implementation of the ABE pre-isolation system.
- (2) Second, we propose an improved attribute-based encryption algorithm to build an attribute isolation architecture of the in-vehicle network after getting the frequency sets. This architecture only allows ECU with the same frequency attribute to communicate, thus reducing the load on the bus while isolating ECU nodes that do not

need to communicate. Most importantly, it improves the security of automotive node communication.

- (3) Third, we analyze the communication architecture security. Through the performance evaluation based on hardware is performed, as well as the simulation based on vehicle network simulator (IVNS), the results of security analysis and evaluation show that the architecture meets the security and real-time requirements of the in-vehicle network.

The rest of this paper is organized as follows: In Section 2, we review more related works. In Section 3, ECU frequency attributes are classified. In Section 4, we present the details of isolated communication system. The analysis of the security for the communication system in Section 5. Simulations are presented in Section 6. Finally, the conclusions of this research and future work are presented in Section 7.

2. Related Work

In the past decade, in order to solve the problem of information security of ICV, many researchers have been working on designing a secure in-vehicle network architecture. As the most widely used in-vehicle bus communication protocol, CAN is currently the focus of research. The characteristics of non-encryption and non-authentication are the main reasons why CAN networks are vulnerable to malicious attacks. Therefore, enhancing the security of the in-vehicle network with encryption and authentication functions is one of the effective measures to ensure the confidentiality and reliability of CAN frames. In this section, research related to the proposed security communication for in-vehicle CAN is presented.

The first method of ensuring the security of in-vehicle networks through authentication mechanisms was proposed by Nilsson et al. [13], who provided integrity and authentication through a 64-bit message authentication codes (MACs) tag. Their method requires four messages to send a 64-bit tag by inserting a 16-bit tag in the CRC field. Therefore, their method increases the bus load and takes up CAC field resulting in errors that cannot be verified during transmission.

With the development of ICV, CAN is gradually unable to meet its demands in transmission rate and bandwidth. In order to solve this problem, Robert Bosch GmbH developed a new communication protocol—CAN with flexible data rate (CAN-FD) [14]. Soon afterward, Patsakis et al. [15] proposed a distributed secure communication architecture for modern vehicles under CAN-FD in which ECU performs secure multi-party calculations for authentication and asymmetric encrypted communication. However, ECU nodes need additional data interaction in the communication process, which increases the bus load and limits the applicability of this architecture in the vehicle real-time communication system. In [16], Woo et al. proposed a practical vehicle CAN-FD security architecture. In their architecture, each ECU performs the initial session key generation process with the GECU in a fixed order, and the ECUs perform authentication and encryption based on HMAC and AES. Han et al. [17] creatively put forward a kind of attribute isolation communication architecture. Their scheme is to divide the ECU's functional attributes according to the characteristics of ICV, and build the attribute isolation communication architecture based on the ABE algorithm. Agrawal et al. [18] implement encrypted communication for ECU groupings based on characteristics and enable communication between ECUs of different groups through GECU. However, the forwarding of data frames from different groups through the GECU can cause communication delays. Groza et al. [19] proposed a broadcast authentication for CAN-FD. They used split keys and mixed authentication tags to improve the security of the in-vehicle network.

While the above methods provide authentication or encryption of data for in-vehicle networks, they also suffer from response delays or increased bus loads. In this paper, we propose a frequency attribute isolation system based on CP-ABE for CAN-FD. ECUs encrypt data according to access structure without concern for the number of ECUs in the network, which is suitable in in-vehicle networks where the number of ECUs is increasing

dramatically. By associating the key with the frequency attribute of the ECUs, there is no need to store a large number of keys and key management is simplified. In addition, the proposed system reduces the response time and bus load rate compared to [18,19].

3. ECU Frequency Attribute Classification

In this section, we will analyze the communication frequency correlation based on real in-vehicle communication data and cluster the ECUs.

3.1. Data Pre-Processing

The CAN bus is a multi-master bus system where all ECUs connected to the network can send data when it is idle. When an ECU on the CAN sends data, it is broadcast to all other ECUs in the network in the form of a data frame. However, according to the CAN bus communication list we obtained from one company, the data frames on the bus are not required for every ECU, part of the communication list is shown in Table 1. For example, when the engine management system (EMS) sends a data frame with the ID FB, the gear shift module (GSM) and instrument cluster unit (ICU) do not use it. The communication list contains the destination and source addresses of the data frames, this information is confidential and is provided by the automotive companies we work with. Therefore, isolation of ECUs that have no communication needs does not interfere with the normal operation of the vehicle. In addition, when a node in the CAN is hijacked, it can prevent all nodes in the network from being controlled.

Table 1. Part of the communication list.

ID	EMS	TCU	GSM	ICU	ESP	
FB	T	R			R	
1A1	R	T				
1A3		R	T	R		
431	R			T		
211	R	R			T	
... ..						

Annotation: T = sender, R = user.

To ensure that the results of the correlation analysis are more accurate, we use a real dataset of in-vehicle communications for the analysis. We collected 870,000 CAN bus data for two hours of vehicle driving with a USBCAN. The communication dataset contains the DLC, Data, etc. of the data frames without destination and source addresses, but from the communication dataset we can obtain the frequency of each data frame.

Furthermore, we have produced a heat map (Figure 2) based on the pre-processed data set, from which we can get a more intuitive impression of the connection between the ECUs. Dark colors indicate a high frequency of communication between ECUs and light colors indicate a low frequency of communication, we can see that some ECUs communicate at a high frequency and some at almost zero.

Figure 2. Heat map of some ECUs.

3.2. Communications Frequency Correlation Analysis

Association rule analysis is the process of mining hidden information from a data set, and the maximum frequent itemset mining algorithm is an important algorithm in association rule analysis [20,21]. A frequent item set is a set of items that occur in a data set with a frequency no less than that used to specify a threshold. For example, a set of items, such as EMS and transmission control unit (TCU), that occur frequently together in dataset, is a frequent item-set. The sets of ECUs that appear frequently in the dataset are in demand for communication. However, since the number of frequent item sets is numerous and the maximum frequent itemset is relatively less and can contain all frequent item sets, maximum frequent itemset mining is more suitable for us to uncover the communication relationships between ECUs [22].

Complete set-enumeration tree is the primary data structure for the maximum frequent itemset mining algorithm. The process of data mining is transformed into the search process of a set enumeration tree by describing the set of items as in Figure 3, enumerating all possible combinations of items. Figure 3 shows the complete set-enumeration tree for $\{TCU, EMS, ESP, GSM, ICU\}$. Each node g of the tree is represented by two item sets. The first item set is called the prefix, denoted as $h(g)$, which is represented by the enumeration item of the current node of the tree; the second item set is called the suffix, denoted as $t(g)$, and it is composed of all the items of the child nodes of the current node after removing the items contained in the current node. For example, for node EMS, $h(EMS) = \{EMS\}$, $t(EMS) = \{ESP, GSM, ICU\}$. We denote the parent node of node g as g_p and the child node as g_c . The generation method of g_c is $h(g_c) = h(g) \cup i$, $i \in t(g)$; $t(g_c) = \{j | j \in t(g), j > i\}$.

Figure 3. Complete set-enumeration tree over five items.

Maximum frequent itemset mining for complete set-enumeration tree according to the max-miner algorithm [23]. Maximal frequent itemset mining is the continuous pruning of the enumeration tree. There are two principles of pruning: firstly, all item sets containing infrequent subsets are infrequent; secondly, if the superset of an item set is a frequent item set, this item set must not be the maximum frequent item set. We prune the enumeration tree according to these two principles. First, the 1-itemset are sorted by frequency of occurrence in the dataset and the candidate itemset C and the frequent itemset F . Second, the frequency of $h(g) \cup t(g)$ of element g in C is calculated and added to F if it is greater than the threshold value. Otherwise, add its child nodes $h(g_c)$ to C . Third, update the candidate itemset C and the frequent itemset F . Repeat second and third until $C = \{\}$. The max-miner algorithm is presented as Algorithm 1.

Algorithm 1 Max-Miner.

```

1: Input: Data T
2: Output: Maximal frequent item sets
3: Set of Candidate Group  $C \leftarrow \{\text{Frequent 1- item sets}\}$ 
4: Set of Itemset  $F \leftarrow \{\text{Frequent 1- item sets}\}$ 
5: While  $C$  is non-empty do
6:   Scan  $T$  to count the frequency of all candidate groups in  $C$ 
7:   For each  $g \in C$  such that  $h(g) \cup t(g)$  is frequent do
8:      $F \leftarrow F \cup \{h(g) \cup t(g)\}$ 
9:   For each  $g \in C$  such that  $h(g) \cup t(g)$  is infrequent do
10:     $C \leftarrow C \cup \{h(g_c)\}$ 
11:   Remove from  $F$  any itemset with a proper superset in  $F$ 
12:   Remove from  $C$  any group  $g$  such that  $h(g) \cup t(g)$  has a superset in  $F$ 
13: Return  $F$ 

```

3.3. Results of Clustering

Through constant parameter tuning, we found that the clustering effect is the best when the threshold is 0.1. A total of six maximum frequent item sets are mined, and these six groups are visualized by R language [24], as shown in Figure 4. The circle diagram in Figure 4 is a Venn diagram used to show the relationship between sets of frequency attributes, with the overlapping parts being the intersection between different sets and the non-overlapping parts being elements specific to the set. However, the Venn diagram is not very readable, and the upset diagram (i.e., bar chart) in Figure 4 can show the relationship between sets more clearly. According to the clustering results, match the corresponding frequency attribute set for ECUs. For example, the gl_1, gl_2 and gl_3 collections all have EMS, and the frequency attribute set of EMS is $S = \{gl_1, gl_2, gl_3\}$. The set of frequency attributes of automatic parking assist system (APA) is $S_{APA} = \{gl_4, gl_5\}$ and the set of frequency attributes of intelligent remote car anti-theft alarm (GSM) is $S_{GSM} = \{gl_3, gl_5\}$. According to the in-vehicle communication system designed in this paper, EMS and GSM have the same frequency attribute gl_3 , so they can communicate, while EMS and APA do not have the same frequency attribute, so they cannot communicate.

Figure 4. The results of clustering.

4. Communication Architecture

In this section, the in-vehicle communication system based on the above ECU frequency attribute clustering is elaborated. The communication system consists of a GECU and ECUs which are equipped in vehicle. There are four phases to consider for the proposed in-vehicle communication system, namely system initialization, registration, setting the matching strategy and isolated communication.

GECU: The GECU is considered trustworthy and has better computation power and storage capacity than the ECU. The GECU is used to verify the identity of the ECUs.

ECU: ECUs are also known as car computers and their purpose is to control the state of the car and implement its various functions. According to the frequency attribute clustering above, each ECU has a different set of frequency attributes.

4.1. System Initialization

The GECU follows the steps below to generate the public parameters and master key, before broadcasting the public parameters across the network.

1. The GECU inputs the security parameter 1^λ , generate an additive group G and a multiplicative G_1 with prime order p , g is a generator in G . Define a bilinear mapping $e : G \times G \rightarrow G_1$.
2. It can be seen from the above that there are a total of six group labels for ECUs, GECU randomly selects six number for group labels and marks them as $g^{l_1}, g^{l_2}, g^{l_3}, g^{l_4}, g^{l_5}, g^{l_6}$ in Z_p^* .
3. The GECU randomly picks $y \in Z_p, \theta, \alpha, \beta \in G$ and publishes the public parameters are: $PK = ((p, g, G, G_1, e), \theta, \alpha, \beta, e(g, g)^y)$. Meanwhile, the master key is $MK = y$.

4.2. Registration

During the registration phase, the GECU verifies the legitimacy of the ECU identity and sends the master key for the ECU with a legitimate identity. To improve the speed of registration, symmetric encryption is used at this stage. Specific registration process is shown in Algorithm 2.

4.2.1. ECU Sends Registration Request Information to GECU

1. ECU_I randomly chooses $a \in Z_q^*$ to generate the request information $(ag || ID_{ECU_I})$.
2. ECU_I signs the request information $(ag || ID_{ECU_I})$ to obtain the signature information Sig_I .
3. ECU_I sends $Msg_1(ag || ID_{ECU_I} || t || Sig_I || Certificate_I)$ to GECU.

4.2.2. GECU Verifies ECU Identity

1. The GECU verifies the validity of the timestamp by Formula (1). The maximum time difference allowed by the in-vehicle network is T , the current time is t' .

$$(\Delta t = t' - t) < T \quad (1)$$

2. If the timestamp is valid, the GECU confirms the integrity and validity of ECU_I by verifying the signature and certificate of the Msg_1 .
3. After verify the legal identity of ECU_I , the GECU randomly choose $b \in Z_q^*$ to generate response information $(bg || ID_{ECU_I})$, and a signature on response information.
4. The GECU generates a temporal session key $SK = abg$, and uses SK to encrypt master key y .
5. The GECU sends $Msg_2(bg || ID_{GECU} || t' || Sig_{GECU} || Certificate_{GECU} || E_{SK}(y))$ to ECU_I .

4.2.3. GECU Sends MK to ECU

1. ECU_I verifies the validity of the timestamp by Formula (2). The maximum time difference allowed by the in-vehicle network is T , the current time is t'' .

$$(\Delta t = t'' - t') < T \quad (2)$$

2. If the timestamp is valid, ECU_I confirms the integrity and validity of GECU by verifying the signature and certificate of the Msg_2 .
3. After verify the legal identity of GECU, ECU_I decrypts $E_{SK}(y)$ in Msg_2 by SK and obtains y .

Algorithm 2 ECU Registration Protocol.

```

1:  $ECU_I$ : Generate the registration request information
2:  $ECU_I \rightarrow GECU$ :  $Msg_1(ag||ID_{ECU_I}||t||Sig_I||Certificate_I)$ 
3: GECU:
4: if  $(\Delta t = t' - t) < T$ 
5:   if  $ECU_I$  is legal
6:      $GECU \rightarrow ECU_I$ :  $Msg_2(bg||ID_{GECU}||t'||Sig_{GECU}||Certificate_{GECU}||E_{SK}(y))$ 
7:   else
8:     Refuse the request information
9:   endif
10:  $ECU_I$ :
11: if  $(\Delta t = t'' - t') < T$ 
12:   if GECU is legal
13:     decrypt  $E_{SK}(y)$  in  $Msg_2$  by  $SK$  and obtains  $y$ 
14:   else
15:     Execute the registration protocol again
16:   endif

```

4.3. Setting the Matching Strategy

Based on the above clustering of frequency attributes define the set of frequency attributes as $U = \{gl_1, gl_2, gl_3, gl_4, gl_5, gl_6\}$. A tree structure T is used to represent access policies. When x is an internal node, it represents the relationship “or”, when x is a leaf node, it represents the frequency attribute. For example, the set of frequency attributes of EMS is $S = \{gl_1, gl_2, gl_3\}$. The EMS access structure tree T is shown in Figure 5.

Figure 5. Access structure of EMS.

In order to represent simply, we define the following two functions. Firstly, $parent(x)$ is used to denote the parent of node x . Secondly, the function $att(x)$ returns the frequency attribute associated with x when x is a leaf node.

Let the root node of the access tree T be r . The access tree T is denoted T_r and the subtree with root node x is denoted T_x . A set of frequency attributes S satisfying the access tree structure T_x is denoted $T_x(S) = 1$, otherwise, $T_x(S) = 0$. The value of T_x is computed recursively, and if x is non-leaf node, the value of T_x is computed for each child node x' of $T_x(S)$. When at least one child node returns 1, $T_x(S) = 1$. When x is a leaf node, when $att(x) \in S$, $T_x(S) = 1$. If $T_r(S) = 1$ then it means that the set S of frequency attributes satisfies the access T_r .

Based on the clustering results in the previous section, the set of frequency attributes of automatic parking assist system (APA) is $S_{APA} = \{gl_4, gl_5\}$ and the set of frequency attributes of intelligent remote car anti-theft alarm (GSM) is $S_{GSM} = \{gl_3, gl_5\}$. The access structure of the EMS is denoted T , then $T(S_{APA}) = 0$ and $T(S_{GSM}) = 1$. This means that the GSM satisfies the EMS access structure and that messages sent encrypted by the EMS can be successfully decrypted by the GSM, but not by the APA.

4.4. Isolated Communication

We propose an isolated communication system for in-vehicle networks based on asymmetric CP-ABE algorithms. In this system, the key is associated with a frequency

attribute. Only ECUs with the same frequency attribute as the sender ECU can decrypt the ciphertext, which guarantees the confidentiality of the architecture. We take ECU_I , ECU_J and ECU_K as an example to illustrate the way in which the ECUs communicate with each other (see Figure 6). The specific steps include the following three algorithms: encrypt, keygen and decrypt.

Figure 6. Communicate architecture.

1. *Encrypt* (S, M, PK): ECU_I generates the ciphertext CT according to the encrypt algorithm. The encrypt algorithm takes frequency attribute set S_1 of ECU_I , a message M and the public parameters PK as input. ECU_I picks up a random value $s \in Z_p$, where s is secret value. We denote the number of elements in set S as n . ECU_I randomly picks up n elements $k_i \in Z_p$ for $gl_i \in S_1$ and set $C = Me(g, g)^{ys}$, $C_1 = g^s$. For $gl_i \in S_1$, the algorithm computes $C_{gl} = \beta^s + \alpha^{k_i}$, $C'_{gl} = \theta^{-H(gl)k_i}$, $C''_{gl} = g^{k_i}$ and hashes the value of the gl_i in S_1 and construct an access structure tree T as shown in Figure 7. The ciphertext $CT = (T, C, C_1, \forall gl \in N : C_{gl}, C'_{gl}, C''_{gl})$.

Figure 7. Access structure T .

2. *Keygen* (S, MK, PK): ECU_J generates key SK according to the keygen algorithm. The keygen algorithm takes frequency attribute set S_3 of ECU_J , the master key MK and the public parameters PK as input. ECU_J denotes the number of elements in S_3 is m , and randomly picks up m elements $t_i \in Z_p$ for $gl_i \in S_3$. ECU_J randomly chooses $t \in Z_p$ and computes $D_1 = g^y + \beta^t$, $D_2 = g^t$. For $gl_i \in S_3$, the algorithm computes $D'_{gl} = g^{t_i}$ and $D''_{gl} = \theta^{H(gl)t_i} + \alpha^{-t}$. The secret key $SK = (D_1, D_2, \forall gl \in GL : D'_{gl}, D''_{gl})$. The steps for ECU_K are the same as for ECU_J .
3. *Decrypt* (CT, SK, S): ECU_J decrypts the ciphertext CT according to the decrypt algorithm. The decrypt algorithm takes the frequency attribute set S_3 of ECU_J , the ciphertext CT and secret key SK as input. ECU_J computes the hash value of the

elements in the set S_3 and denotes it as the set S'_3 . The algorithm chooses D'_{gl}, D''_{gl} for gl , which the hash value matches the T. ECU_J computes as follows:

$$B = e(C_1, D_1) / e(C_{gl}, D_2) e(C'_{gl}, D'_{gl}) e(C''_{gl}, D''_{gl}) \quad (3)$$

$$B = e(g^s, g^y + \beta^t) / e(\beta^s + \alpha^{k_i}, g^t) e(\theta^{H(gl)k_i}, g^{t_i}) e(g^{k_i}, \theta^{-H(gl)t_i} + \alpha^{-t}) \quad (4)$$

$$B = e(g^s, g^y + \beta^t) / e(\beta^s + \alpha^{k_i}, g^t) e(g^{k_i}, \alpha^{-t}) \quad (5)$$

$$B = e(g^s, g^y + \beta^t) / e(\beta^s, g^t) \quad (6)$$

$$B = e(g, g)^{sy} \quad (7)$$

ECU_J obtains the $M = CB$. The frequency attribute set S_2 of the ECU_K does not satisfy the access structure tree T, so the ECU_K decryption fails.

5. Security Analysis of the Proposed Scheme

In this section, we present a theoretical proof of the security of the proposed communication architecture for IoV.

Theorem 1. Assuming that the computational Diffie–Hellman (CDH) assumption is established, the MK in the proposed system cannot be obtained.

Proof of Theorem 1. If the adversary A can compute temporal session key $SK = abg$, it is possible for A to obtain MK. The advantage of a successful attack by A is Adv_A . We use A to construct algorithm A_{CDH} to solve the CDH problem.

A_{DL} randomly picks $y, \theta, \alpha, \beta \in Z_p$ and publishes the public parameters are: $PK = ((p, g, G, G_1, e), \theta, \alpha, \beta, e(g, g)^y)$ and saves the master key $MK = y$. A can make queries about A_{CDH} to q_{CDH} times.

Query: A makes queries about SK, the algorithm A_{DL} returns ag and bg to A .

Challenge: After A receives ag and bg , A uses (ag, bg) to call algorithm A_{DL} . That is given ag, bg , compute abg . The advantage of A challenge success in this process is $Adv_A = q_{CDH} \times Adv_{CDH}$. The advantage Adv_A of the algorithm in successfully solving the CDH problem in the polynomial time is negligible. Therefore, the adversary A does not have access to the temporal session keys SK and MK. $\square$

Theorem 2. Assuming that the decisional Diffie–Hellman (DDH) assumption is established, plaintext cannot be extracted from ciphertext.

Proof of Theorem 2. If there is a polynomial-time adversary A that can attack our scheme with advantage of ϵ , we can construct a simulator B to play the DDH game with advantage of ϵ . Given a DDH paradigm (g, g^a, g^b, Z) , B creates the following simulation.

Init: The adversary A commits to simulator B the challenge access tree T . The simulator B sets the public parameters according to the following steps. It randomly picks $y, m, n \in Z_p$ and calculates $\theta = g^m, \beta = g^n$ and sets the public parameter $PK = ((p, g, G, G_1, e), \theta, \alpha = g^a, \beta, e(g, g)^y)$ and the master key $MK = y$. The simulator sends the PK to the adversary A .

Phase 1: The adversary A can query the SK of the frequency attribute set S . If S cannot satisfy the challenge access tree, B will calculate SK and send it to A as follows. B randomly pick $t \in Z_p, \{t_i \in Z_p\}_{1 \leq i \leq m}$ and $D_1 = g^y + \beta^t, D_2 = g^t$. For $gl \in S$, the algorithm computes $D'_{gl} = g^{t_i}$ and $D''_{gl} = \theta^{-H(gl)t_i} + \alpha^{-t}$. The secret key $SK = (D_1, D_2, \forall gl \in GL : D'_{gl}, D''_{gl})$.

Challenge: The adversary A submits two messages M_0 and M_1 of equal length to B . The simulator randomly selects a message M_b to encrypt and sends the encrypted ciphertext CT to A . The ciphertext is output as:

$$CT = (P', C, C_1, \forall gl \in N : C_{gl}, C'_{gl}, C''_{gl})$$

$C = Me(g, g)^{ys} = me(g, g)^{yab} = me(g^a, g^b)^y$ which implies $s = ab, C_1 = Z$. For $gl \in N$, the algorithm computes $C_{gl} = g^{ns} + g^{ak_i}, C'_{gl} = \theta^{H(gl)k_i}, C''_{gl} = g^{k_i}$. For $gl \notin N$, the algorithm computes $C_{gl} = g^{ns} + g^{ak_i}, C''_{gl} = g^{k_i}$ and randomly selects $C'_{gl} \in G$.

Phase 2: Repeat Phase 1 and any S cannot satisfy the challenge access tree.

Guess: The adversary A computes M . When $C_1 = Z = g^{ab}$, the ciphertext is well-formed and the plaintext M is computed correctly. A can guess b' correctly. When Z is randomly chosen, m is random and A can only guess b' randomly, So if $b' = b$, B return $\eta = 1$. Otherwise, B returns $\eta = 0$. However, DDH difficulty problem is unsolvable in probabilistic polynomial time. Therefore, we can conclude that the proposed scheme satisfies IND-CPA secure. $\square$

6. Simulation and Evaluation

To evaluate the performance of the proposed scheme, we carried out hardware-based performance evaluation and software simulation based on in-vehicle network simulator (IVNS) [25]. The performance evaluation environment is illustrated in Figure 8, and the specifications of equipment used for evaluation are listed in Table 2.

Figure 8. Performance evaluation environment.

Table 2. Tools used for the simulation.

Tools	Remarks
Hardware	STM32H743, STM32H743IIT6
Compiler	MDK5
Software	IVNS
CPU	Intel core i5-8259U 1.6 GHz
RAM	8GB

6.1. Hardware-Based Performance Evaluation

To measure the time parameters of registration, keygen, encrypt, decrypt success and failure, we carried out hardware-based performance evaluation. In our scheme, all ECU must send a registration request to GECU before the vehicle starts, and the ECU passing the registration request can obtain important parameters for communication. After obtaining the important parameters, ECU generates the key for decryption according to its own set of frequency attributes. When ECU_I sends a data frame, it encrypts the message according to the set of frequency attributes, hashes the frequency attributes of the set and construct an access structure tree T to achieve the effect of protecting the privacy of the ECU_I , and embeds T in the ciphertext for broadcasting in the network. When the ECU_J receives the ciphertext, it hashes its own set of frequency attributes first, if it meets the access structure tree T , it is considered successful decryption, otherwise, the decryption failed. We apply this scheme to the microcontrollers of STM32H743 and STM32H743IIT6 and evaluate their performance. To minimize errors, these four algorithms were run 100,000 times respectively, and the average, maximum and minimum execution time were measured.

As shown in Figure 9, the average time of successful registration is 8.24 ms. The ignition time of a car is about 1 s generally, and the number of ECU nodes in the car will not exceed 100, so the scheme can meet the real-time requirements of current intranet. The average time of key generation, encryption, decryption success and failure are 2.78 ms, 4.12 ms, 6.9 ms and 1.98 ms, respectively.

Figure 9. Consumption time of the five algorithms.

6.2. Network Simulator-Based Evaluation

We use the software IVNS, STM32H743 and STM32H743IIT6 microcontrollers to build an evaluation environment similar to the real in-vehicle network environment. IVNS is an in-vehicle network simulator developed by Mundhenk et al., based on the discrete event simulation framework (SimPy) in Python language in 2016. It can evaluate the real-time performance of in-vehicle network well. We imported the average execution time of registration, key generation, encryption, successful decryption and failure decryption based on the hardware performance evaluation into the IVNS database, and performed the performance evaluation based on the network simulator. STM32H743 serves as the GECU and STM32H743IIT6 serves as the ECU.

6.2.1. Analysis of Calculation Time Consumption

For the measurement of time calculation consumption, we defined the following scenarios and performed a performance evaluation.

- (1) Sender-ECU encrypts data frames based on the set of frequency attributes.
- (2) Sender-ECU broadcasts the ciphertext and access structure tree T on the network.

- (3) Receiver-ECU receives ciphertext and access structure tree T .
- (4) If the frequency attributes set of receiver-ECU meets the requirements of the access structure tree T , decryption will be carried out and plaintext will be obtained. Otherwise, decryption fails.

We measured the execution time from the time the sender ECU encrypts the data frame to the time all the receiver ECUs decrypt the message. By fixing the bit rate of the arbitration segment at 0.5 Mbit/s, the bit rate of the data segment at 4 Mbit/s, and the data transmission cycle of the ECU at 50ms, the communication response time of the ECU at same CPU clock rate is measured. We measured the response time of Sec suggestion in [18] and LiBrA suggestion in [19] under the same circumstances and compared them to the proposed scheme. The communication response time under different ECU numbers is shown in Figure 10. Since the encrypted data frames are forwarded by the GECU, the Sec's response time is higher than ours. At the same time, since there are 2^n communication groups for n nodes, LiBrA's response time is higher than ours when the number of ECUs increases. For example, with 80 ECUs in the vehicle, the response time of our solution is 34% faster than Sec and 57% faster than LiBrA.

Figure 10. Communication response time.

6.2.2. Analysis of Bus Load Rate

Bus load rate refers to the actual legend of the total number of bits per unit time of the bus and the total number of bits per unit time of the legend of the bus, and is an important index to measure the performance of the in-vehicle network communication, we can read the communication of the bus load rate through the IVNS added monitor. By fixing the bit rate of the arbitration segment at 0.5 Mbit/s, the bit rate of the data segment at 4 Mbit/s, and the data transmission cycle of the ECU at 20 ms, the bus load rate of the architecture in this paper is evaluated. We measured the bus load rate of Sec suggestion in [18] and LiBrA suggestion in [19] under the same circumstances and compared them to the proposed scheme. The bus load rate under different ECU numbers is shown in Figure 11. Due to the fact that there are 2^n communication groups for n nodes and that the ECUs store multiple keys, the bus load rate of the LiBrA increases significantly when the number of ECUs increases. Since the encrypted data frames are forwarded by the GECU, the Sec's bus load rate is higher than ours. As the encryption and decryption of our solution is less affected by the number of ECUs, the bus load rate of our solution is 7.8% lower than Sec and 20.6% lower than LiBrA at a number of 80 ECUs in the vehicle.

Under normal circumstances, the bus load rate must be maintained below 30% in order to meet the real-time requirements of intelligent networked vehicles for in-vehicle communication. As shown in Figure 11, the bus load rate of our scheme is kept below 30%, which meets the real-time communication requirements of vehicles. The proposed scheme simplifies key management and reduces bus load by constructing access policies

for each ECU and isolating unauthorized ECUs, meeting the real-time needs of the in-vehicle network.

Figure 11. Bus load rate.

7. Conclusions

In this paper, a secure and efficient in-vehicle communication system is designed based on an asymmetric ABE algorithm. First, the ECU communication frequency relationship is analyzed according to the max-miner algorithm of the maximum frequent term set, and the frequency attributes of ECUs are classified based on the mining results. Secondly, access structures are designed according to the set of frequency attributes of the ECUs to build the in-vehicle network communication system. In our scheme, only ECUs that meet the ciphertext requirements can decrypt the data, ensuring data confidentiality while reducing the risk of attacks on the in-vehicle network. In addition, we demonstrate through theoretical proof that the scheme in this paper achieves IND-CPA security. Finally, our solution is evaluated by means of a hardware platform and IVNS software. The evaluation results show that the scheme meets the requirements of in-vehicle networks in terms of communication latency and busload.

In the future, as driverless technology becomes more popular, there will be higher requirements for data transmission efficiency in in-vehicle networks. Designing faster and more secure in-vehicle communication solutions is the focus of our research.

Author Contributions: Conceptualization, M.H. and M.Z.; methodology, M.H.; software, M.Z.; validation, M.Z.; formal analysis, P.C.; investigation, M.Z.; resources, M.Z.; data curation, M.H.; writing—original draft preparation, M.Z.; writing—review and editing, M.H. and P.C.; visualization, H.Q.; supervision, M.H. and Z.Y.; project administration, M.H.; funding acquisition, M.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the Key Research and Development Plan of Jiangsu province in 2017 (Industry Foresight and Generic Key Technology) (BE2017035) and the Project of Jiangsu University Senior Talents Fund (1281170019).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this paper:

$h(g)$	Prefix of node g
$t(g)$	Suffix of node g
g_p	Parent node of g
g_c	Child node of g
ECU_I	The I -th ECU
ID_{ECU_I}	Identity of ECU_I
Sig_I	Signature of ECU_I
$Certificate_I$	Certificate of ECU_i
SK	Temporal session key
S	Set of frequency attributes of ECU
gl_i	The i -th frequency attribute

References

1. Trubia, S.; Severino, A.; Curto, S.; Arena, F.; Pau, G. Smart Roads: An Overview of What Future Mobility Will Look Like. *Infrastructures* **2020**, *5*, 107. [\[CrossRef\]](#)
2. Kaiwartya, O.; Abdullah, A.H.; Cao, Y.; Altameem, A.; Prasad, M.; Lin, C.; Liu, X. Internet of Vehicles: Motivation, Layered Architecture, Network Model, Challenges, and Future Aspects. *IEEE Access* **2016**, *4*, 5356–5373. [\[CrossRef\]](#)
3. Woo, S.; Jo, H.J.; Lee, D.H. A Practical Wireless Attack on the Connected Car and Security Protocol for In-Vehicle CAN. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 993–1006. [\[CrossRef\]](#)
4. Han, M.; Cheng, P.; Ma, S. PPM-InVIDS: Privacy Protection Model for In-Vehicle Intrusion Detection System Based Complex-Valued Neural Network. *Veh. Commun.* **2021**, 100374. [\[CrossRef\]](#)
5. Koscher, K.; Czeskis, A.; Roesner, F.; Patel, S.; Kohn, T.; Checkoway, S.; McCoy, D.; Kantor, B.; Anderson, D.; Shacham, H.; et al. Experimental Security Analysis of a Modern Automobile. In Proceedings of the 2010 IEEE Symposium on Security and Privacy, Oakland, CA, USA, 16–19 May 2010; pp. 447–462.
6. Kim, J.H.; Seo, S.; Hai, N.; Cheon, B.M.; Lee, Y.S.; Jeon, J.W. Gateway Framework for In-Vehicle Networks Based on CAN, FlexRay, and Ethernet. *IEEE Trans. Veh. Technol.* **2015**, *64*, 4472–4486. [\[CrossRef\]](#)
7. Miller, C.; Valasek, C.J.B.H.U. Remote exploitation of an unaltered passenger vehicle. *Black Hat USA* **2015**, 2015, S91.
8. Humayed, A.; Lin, J.; Li, F.; Luo, B. Cyber-Physical Systems Security—A Survey. *IEEE Internet Things J.* **2017**, *4*, 1802–1831. [\[CrossRef\]](#)
9. Radu, A.-I.; Garcia, F.D. LeiA: A Lightweight Authentication Protocol for CAN. In Proceedings of the Computer Security—ESORICS 2016, Heraklion, Greece, 26–30 September 2016; pp. 283–300.
10. Dariz, L.; Selvatici, M.; Ruggeri, M.; Costantino, G.; Martinelli, F. Trade-off analysis of safety and security in CAN bus communication. In Proceedings of the 2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS), Naples, Italy, 26–28 June 2017; pp. 226–231.
11. Bethencourt, J.; Sahai, A.; Waters, B. Ciphertext-Policy Attribute-Based Encryption. In Proceedings of the 2007 IEEE Symposium on Security and Privacy (SP '07), Berkeley, CA, USA, 20–23 May 2007; pp. 321–334.
12. Li, J.; Zhang, Y.; Ning, J.; Huang, X.; Poh, G.S.; Wang, D. Attribute Based Encryption with Privacy Protection and Accountability for CloudIoT. *IEEE Trans. Cloud Comput.* **2020**, *1*. [\[CrossRef\]](#)
13. Nilsson, D.K.; Larson, U.E.; Jonsson, E. Efficient In-Vehicle Delayed Data Authentication Based on Compound Message Authentication Codes. In Proceedings of the 2008 IEEE 68th Vehicular Technology Conference, Calgary, AB, Canada, 21–24 September 2008; pp. 1–5.
14. Hartwich, F. CAN with flexible data-rate. In Proceedings of the Proc. iCC, Palatinate, Germany, 5 March 2012; pp. 1–9.
15. Patsakis, C.; Dellios, K.; Bouroche, M. Towards a distributed secure in-vehicle communication architecture for modern vehicles. *Comput. Secur.* **2014**, *40*, 60–74. [\[CrossRef\]](#)
16. Woo, S.; Jo, H.J.; Kim, I.S.; Lee, D.H. A Practical Security Architecture for In-Vehicle CAN-FD. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 2248–2261. [\[CrossRef\]](#)
17. Han, M.; Wan, A.; Zhang, F.; Ma, S. An Attribute-Isolated Secure Communication Architecture for Intelligent Connected Vehicles. *IEEE Trans. Intell. Veh.* **2020**, *5*, 545–555. [\[CrossRef\]](#)
18. Agrawal, M.; Huang, T.; Zhou, J.; Chang, D. CAN-FD-Sec: Improving Security of CAN-FD Protocol. In Proceedings of the Security and Safety Interplay of Intelligent Software Systems, Barcelona, Spain, 6–7 September 2019; pp. 77–93.
19. Groza, B.; Murvay, S.; Van Herreweghe, A.; Verbauwhede, I. LiBrA-CAN: Lightweight Broadcast Authentication for Controller Area Networks. *ACM Trans. Embed. Comput. Syst.* **2017**, *16*, 28. [\[CrossRef\]](#)
20. Dhanaseelan, R.; Jeya Sutha, M. Diagnosis of coronary artery disease using an efficient hash table based closed frequent itemsets mining. *Med. Biol. Eng. Comput.* **2018**, *56*, 749–759. [\[CrossRef\]](#) [\[PubMed\]](#)

21. Tran, T.; Vo, B.; Le, T.T.N.; Nguyen, N.T. Text Clustering Using Frequent Weighted Utility Itemsets. *Cybern. Syst.* **2017**, *48*, 193–209. [[CrossRef](#)]
22. Burdick, D.; Calimlim, M.; Flannick, J.; Gehrke, J.; Yiu, T. MAFLA: A maximal frequent itemset algorithm. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 1490–1504. [[CrossRef](#)]
23. Bayardo Jr, R.J. Efficiently mining long patterns from databases. In Proceedings of the ACM SIGMOD International Conference on Management of Data, Seattle, WA, USA, 1–4 June 1998; pp. 85–93.
24. Lex, A.; Gehlenborg, N.; Strobel, H.; Vuilleumot, R.; Pfister, H. UpSet: Visualization of Intersecting Sets. *IEEE Trans. Vis. Comput. Graph.* **2014**, *20*, 1983–1992. [[CrossRef](#)] [[PubMed](#)]
25. Mundhenk, P.; Mrowca, A.; Steinhorst, S.; Lukasiewicz, M.; Fahmy, S.A.; Chakraborty, S. Open source model and simulator for real-time performance analysis of automotive network security. *ACM Sig. Rev.* **2016**, *13*, 8–13. [[CrossRef](#)]