Hindawi
Security and Communication Networks
Volume 2021, Article ID 4646087, 18 pages
https://doi.org/10.1155/2021/4646087

Research Article

Service-Based Hybrid Access Control Technology with Priority Level for the Internet of Vehicles under the Cloud Architecture

Pengshou Xie, Haoxuan Yang, Liangxuan Wang, Shuai Wang, Tao Feng, and Yan Yan

School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China

Correspondence should be addressed to Haoxuan Yang; yanghx@lut.edu.cn

Received 22 July 2021; Revised 20 August 2021; Accepted 12 November 2021; Published 3 December 2021

Academic Editor: Marimuthu Karuppiah

The communication process of devices in IoV under cloud architecture needs to be protected by access control models. However, existing access control models have difficulty establishing the appropriate granularity of permissions in the face of large amounts of data in IoV. Moreover, the access control model may need to temporarily change user privileges to accommodate the dynamic nature of IoV scenarios, a requirement that is difficult to implement for traditional access control models. The unstable connection status of devices in IoV also creates problems for access control. The service (composed of role and attribute) based access control model (in IoV) S-RABAC (V), under the Cloud computing architecture, introduces a formal theoretical model. The model uses attribute grouping and prioritization mechanisms to form a hierarchical structure. The permission combination pattern in the hierarchical structure can avoid duplicate permissions and reduce the number of permissions while ensuring fine-grained permissions. Different layers in the model have different priorities, and when a user's permission requires temporary changes, it can be adjusted to the corresponding layers according to the user's priority. In addition, users are allowed to keep their assigned privileges for a period to avoid frequent access control because of unstable connections. We have implemented the proposed access control model in Alibaba Cloud Computing and given six example demonstrations. The experiment shows that this is an access control model that can protect IoV security more effectively. Various unique mechanisms in the model enable S-RABAC(V) to improve the overall access control efficiency. The model adds some extra features compared to ABAC and RBAC and can generate more access control decisions using the priority mechanism.

1. Introduction

In recent years, with the rapid growth of the smart car industry, there has been a great interest in the Internet of Vehicles (IoV) technology. The generation of IoV can reduce road congestion, improve traffic management, ensure road safety, and thus enhance the experience of road users [1]. IoV does not receive problems directly from inadequate storage and energy consumption. These problems often manifest themselves indirectly through a number of phenomena. However, large scaling environments require processing of extensive amount of information and it is absolutely a challenging issue [2, 3]. With such challenges, cloud platforms become a preferred technology when interacting with IoV resources. Various private cloud companies, such as Ali, Huawei, and Amazon, have developed various IoV cloud computing platforms in collaboration with the vehicle industry. The large number of heterogeneous cloud architectures (cloud computing architectures) generated by IoV has generated various cybersecurity requirements. When the IoV combined with in a cloud architecture, the IoV places new demands on access control technology as a security technology.

Access control technology, as a security technology, protects user privacy and authentication. Access control technology, like most network security technologies, faces different requirements in different cloud architecture systems. Since there are many heterogeneous cloud access devices in the cloud architecture of IoT, there is a need to

consider many heterogeneous services provided by different vendors [4, 5]. The IoV cloud architecture has the same basic requirements as the IoT cloud architecture. However, IoV has some designs that differ from IoT, such as the virtual device layer for unstable connections [6]. The various cloud architecture designs that bring convenience to IoV and IoT also create new security issues. How not to disclose user privacy and ensure the security of accessed data during information exchange is a problem that needs to be addressed by cloud architectures. It is crucial to apply security technologies to the special environment in which the cloud architecture is located.

Access control technology has different access control models, such as Discretionary Access Control (DAC), Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), and Medium Access Control (MAC) protocols, such as the practical application of access control scenarios. Some of these methods are not effectively utilized in the access control model but are widely used in other related areas of access control technology, such as prioritization mechanisms that are described in the next paragraph. Access control models often ensure compliant access to data in a large environment. Access control models for IoV and IoT in cloud architectures need to meet the needs of both IoV and IoT, as well as the needs of the cloud architecture itself. RBAC itself has become a widely used access control model because of its simple structure and arrangement. However, RBAC always suffers from the problem that the granularity of permissions is difficult to grasp, and adjusting the rationality of RBAC roles and the scale of permissions is a common way to solve this problem [7, 8]. Also, since RBAC security is constrained by roles, the security of each role will be directly related to the security of data. Adding quantitative computation of trust values to the roles of RBAC can improve the overall security of RBAC to some extent [9, 10]. ABAC has better flexibility than the RBAC access control model, allowing fine-grained access control that can meet the requirements of multiple attributes of users and devices in IoT [11]. However, at the same time, the high flexibility and fine granularity make the model much more complex, making it particularly difficult to manage.

Prioritization mechanisms are widely used in access control systems. The priority mechanism can prioritize different events in Industrial Wireless Sensor Networks (IWSNs) to ensure communication efficiency and low latency communication for high priority nodes, thus improving the overall network structure's communication efficiency [12]. The same priority mechanism can provide special services for emergency services in IWSNs and reduce the channel access delay of emergency service nodes [13]. And, for the Random access process (RAP), the priority mechanism can be used for effective prioritization, thus satisfying the diversity of access devices in the IoT [14]. The priority ranking of applications in IoT is used to provide different quality of service for different applications and to satisfy the low latency requirements of high priority services [15]. And for IoV networks, different service efficiencies are provided for different levels of IoV networks through multi-layer hierarchical techniques in the protocol [16]. It is easy to find that the priority classification, priority ranking, and other functions provided by the priority mechanism can effectively improve the efficiency of information access control techniques in IoT and IoV and provide the possibility of special services for emergency events. However, in the access control models of the same access control technology, whether RBAC or ABAC or other common access control models, the prioritization mechanism is not well utilized. Even with the improved various access control models, the utilization of the prioritization mechanism is still not very high.

To address the above issues, we propose an access control model under the cloud architecture to protect the access process of heterogeneous resources within the IoV cloud computing architecture. The access control model integrates ABAC and RBAC, keeping their respective features, and adding service modules and role sub-modules to improve the efficiency of the overall access control model. In the access control model, we carry out inheritance and hierarchical design for the model permission unit, which makes the whole access control model systematic and reduces the difficulty of managing the model. In addition, we introduce the priority mechanism common in MAC protocols, where different layers have different priorities. We also assign priorities to permission units in each access control model, making the set of permissions diverse. More precisely, our contributions are as follows:

(1) A proposed IoV-oriented access control model in cloud computing architecture.

(2) A hybrid of RBAC and ABAC access control model, with services as selective sets of attributes. The role module in the original RBAC is kept, so that it assists the service module for permission assignment and implements the access permission reservation function.

(3) Introduces the priority mechanism into the access control model, realizing hierarchical priority ranking. And, the priority is assigned to each permission unit, which makes the number of decisions increase. In addition, the priority makes it possible to adjust access rights in emergency situations.

(4) Our model is implemented in Alibaba Cloud Computing platform and five basic cases and one typical nondata access case are demonstrated.

The rest of this article is arranged as follows: Section 2 introduces the work related to the current access control model. Section 3 introduces the current problems faced by access control on the Internet of Vehicles. Section 4 proposes a service-based hybrid access control model under the cloud architecture. Section 5 first describes the implementation of the model and shows the access process. In Section 6, the model is researched and evaluated from the perspective of the Internet of Things and security, and compared with other related access control models. Finally, Section 7 summarizes the work.

2. Related Work

Authentication is a mechanism to protect the IoV against attacks due to the malicious entities in the network. Hence, it is considered to be the first line of defense against any kind of attacks in IoV [2]. Vijayakumar et al. proposed an efficient privacy preserving anonymous authentication scheme using signatures and certificates. The authentication result in this scheme exists in disputed and ordinary identities, and it can revoke the authentication for vehicles with disputed identities [17]. Song split IoV into multiple fog services and proposed a fog-based authentication mechanism for high-speed vehicles [18]. An efficient authentication scheme to bypass TA is proposed by Xu et al. [19]. The above literature, besides achieving an efficient privacy identity authentication process, also differentiates the final authentication results. However, the above authentication mechanisms focus on the study of the secure authentication process, causing the authentication results not to be diverse.

Fine-grained authentication results can help security system users to better manage permissions. A more fine-grained authentication mechanism can be implemented with the help of the access control model. Currently, the main research on access control models is focused on the study of attribute-based access control models. The RBAC access control model is an access control model that clusters attributes by roles. Djilali et al. improved the RBAC model by proposing an enhanced dynamic team access control model that incorporates the concept of team in roles [20]. But, RBAC always has the problem of granularity of permissions, which is an important reason for the instability of RBAC security. Lu et al. conducted a study on the role permission assignment process under preconditions and verified that it is a reasonable role permission assignment [21]. Kamboj et al. use blockchain-based smart contracts to manage user role permissions in organizations [22]. The literature [23] implements permission dependencies between different roles in RBAC using the dependencies between user permissions. Although the static role permissions are set at a fine-grained level can improve the granularity of RBAC, since RBAC permissions are "packaged" by static roles, there is always the disadvantage of immutable permissions due to static roles. That is, the overall permission system cannot affect role permissions.

Using attributes to generate permissions is an approach to fine-grained permission architecture, where the entire model's permission units are changed adaptively based on the abstracted attributes. This access control model is called ABAC. Abirami et al. added trust attribute values to ABAC, which are got from different trust evaluation algorithms or models, and used them to improve the security performance of the model [24, 25]. Zhang et al. use signatures to ensure that the verification process does not disclose attributes [26]. However, the many attribute values added to ABAC make it more difficult to assign and manage permissions, which has been a drawback of the ABAC model. Literature [27] used binary sequences to reduce the number of matching attribute-value pairs in rules and improve the efficiency of policy retrieval. In fact, the data in IoV are characterized by aggregation. Data in IoV often exist as a group. Abbasian et al. pointed out that the classification processing of data in IoV and IoT domains is necessary and prevalent [28]. Álvarez-Bazo et al. divided sensors in vehicular networks into two major categories: on-time data and cross-sectional data [29]. Since data can be grouped in the IoV, it can be grouped and protected. For this reason, Maanak et al. proposed an ABAC access control model of inheritance characteristics applied to IoV based on cloud architecture, in which grouping is performed according to the sensor collection and forms a hierarchical group with up-down logical relationships [30]. The literature [31] similarly uses attribute aggregation to produce a hierarchical structure in ABAC. Servos and Osborn used the hierarchical group attribute architecture (HGAA) structure to layer the ABAC access control model and implemented this scheme [32]. Although these improved ABACs have a hierarchical structure, they make no further use of the results of the hierarchy. In addition, even the attribute structure with certain hierarchical structure becomes difficult to manage as the number of attributes keeps increasing.

In order to reconcile the respective problems of RBAC and ABAC, some researchers have tried to mix the two models. Aftab et al. proposed a hybrid R-ABAC access control, which uses a combination of attribute values to generate roles, thus solving the problem of permission assignment present in ABAC and RBAC [33–36]. The flexibility of the ABAC model was also introduced in the RBAC model by adding the concept of negative authorization [37]. Hu et al. proposed a group-based access control (oGBAC) framework using the features of OSN social member grouping, which proposed the concept of information flow in defining and controlling the resources and information for access control [38]. An important issue with the hybrid access control model is how to reconcile the disadvantages of both while avoiding fresh problems, for example, the problem of role explosion caused by the increase of attributes. This requires researchers to control the relationship between attributes and aggregation units with strict constraints when reconciling the two.

Priority mechanisms have the effect of enhancing model dynamics and improving model efficiency. Several researchers have experimented with applying prioritization mechanisms to a variety of different access control models. Thakare et al. added a prioritization mechanism to RBAC to reduce the operational burden on cloud servers because the same authentication and authorization mechanism were used for "user classification" [39]. Vijayalakshmi and Jayalakshmi proposed an approach to avoid ABAC policy conflicts using priority levels, which are used as additional parameters to construct security policies in access control [40]. Cheng et al. used the priority mechanism to deal with the role authority conflict problem. When there is a conflict between the permissions corresponding to different roles owned by a user, the role with the highest priority permissions gets the corresponding permissions [41]. Although the prioritization mechanism brings enhancements to various parts of the access control model, the above literature only uses the prioritization mechanism to deal with various

Table 1: Summary of related work.

Refs	Pros	Cons
[2, 17–19]	Various efficient authentication mechanisms	Single, undiversified authentication result
[20–23]	Enhance the dynamic nature of RBAC, security, as far as possible to weaken the role of permissions encapsulation generated by the role of granularity issues	There is always the problem of static role permissions. The overall permission system cannot affect role permissions
[24–32]	The ABAC attribute mechanism is used to improve the granularity and the orderliness of permissions. To enhance ABAC security by adding unique attribute values	ABAC makes the overall model more difficult to manage as the number of attributes increases. Second, the large number of permission units generated by many attributes is difficult to track
[33–38]	Use the characteristics of RBAC privilege encapsulation to manage many attributes in ABAC, thus reconciling the disadvantages of each	Many unordered attribute aggregation units can make the mixed access control model create the problem of role explosion. But, too little attribute aggregation units can create the problem of insufficient granularity of permissions in RBAC
[39–41]	Enhance the dynamics of various access control mechanisms using priority mechanisms.	Lack of effective use of prioritization mechanisms in the access control model

special problems. An effective method to use the prioritization mechanism to solve the distinct problems that appear in the access control model is lacking. Table 1 shows the summary of related work.

3. Problems Faced by Access Control Technology in the Internet of Vehicles

3.1. Data Protection Difficulties due to the Self-Organizing Networks of the Internet of Vehicle. As a self-organizing network, the Internet of Vehicles has distinct individual specificity when establishing the authority of resources, making it difficult to establish reasonable resource permissions. When establishing the permission set of these resources, these problems should be considered.

(1) For the same type of resources, different individuals may have different priority levels

(2) The permission set of the current resource generates multiple subsets

(3) With massive data permissions, the access control model should provide an easy-to-use resource search method to ensure the user's ability to search for resources in the access control model

(4) Permissions cannot be managed and searched because of massive resource permissions

(5) When faced with many permissions, a complex set of attributes reduces the execution speed of access control

3.2. Some Problems with the Traditional Access Control Model. The shortcomings of the traditional access control model make it difficult to apply to the Internet of Vehicles.

(1) The identity design of RBAC is not flexible enough for mass data units. Directly designing the identity will inevitably lead to the problem of too many identities and unreasonable permissions. Second, the assigned permissions of individuals in the Internet of Vehicles will change, and this variable permission level is difficult for RBAC.

(2) Although ABAC is more flexible in authorization design, the large number of authorization sets it generates is difficult for the management of the access control system in the later period.

(3) Although the hybrid access control after the combination of RBAC and ABAC makes up for some shortcomings of the two, it also inherits some other shortcomings.

(4) The access speed of the access control model decreases as the complexity of the access control model increases. Therefore, it is necessary to weigh the pros and cons between these two models, which can not only meet the needs of fine-grained permission setting and management of the Internet of Vehicles but also have a certain access speed.

3.3. Permission Adjust Requirements in the Emergencies of the Internet of Vehicles. At certain times, users with low data authority levels in the Internet of Vehicles need to access certain data that can only be accessed with high levels. Here, the authority level can be lowered, or they can create sBut lowering the authorization level will lower the security level of the complete system, and creating an additional authority unit will waste resources. Therefore, it is more appropriate to establish a privilege mechanism to ensure that the user's authority is elevated to a certain level when a certain special situation is met. We have listed two typical permissions adjust situations.

(1) When these emergencies occur, the access control terminal often needs to adjust the current user's data access permissions to implement certain access requests. For example, when the police are investigating criminal vehicles in a certain area, they can achieve a higher level of authority by increasing the authority level for performing official duties in the area, to get the location information of vehicles of the same brand in the area.

(2) When the bumps of the vehicle exceed normal conditions, the average bumps data of the road section can help judge whether the current vehicle is

malfunctioning. Here, the data that the vehicle need to access belong to the internal data of the vehicle company, so there is a need for dynamic permission adjusts.

4. S-RABAC(V) Frameworks

To address the above needs and problems, a service (composed of role and attribute) based access control model S-RABAC (V) for the IoV under the Cloud computing architecture is proposed, where the roles and attributes comprise service. The model is improved and mixed based on the RBAC and ABAC access control models, and we add a service module. The service module comprises two parts: attributes and roles. Attributes are the major part of the service. It will combine different priorities to form the service entity. The role module assists service entities in implementing finer-grained functionality, such as access permission reservation features.

We can divide the overall vehicle services into two major categories: vehicle native resource services and services provided by third parties. The former aims to protect the security of the basic resource access of each model of vehicle, and the latter meets the expansion needs of subsequent vehicle data services. The service module has a hierarchical structure and inheritance design, as shown in Figure 1.

4.1. Definition of S-RABAC(V). Figure 2 depicts the model diagram of the S-RABAC(V) hybrid access control model. The formal definition of the S-RABAC(V) model is shown below.

Basic components:

(i) The IoV subject (IS) is the subject of access behavior in the Internet of Vehicles; resource (R) is the entity that accesses the resource, which represents the various contents that are accessed, such as data, mobile nodes.

(ii) Attribute (ATT) is a description of the resource entity R, and $\mathrm{ATT} = \{\mathrm{att}_1, \ldots, \mathrm{att}_i, \ldots, \mathrm{att}_n\}$; priority level (PL) is a description of the authority value of the resource entity, and $\mathrm{PL} = \{\mathrm{pl}_1, \ldots, \mathrm{pl}_i, \ldots, \mathrm{pl}_n\}$. There is a mapping relationship ATT $\longrightarrow$ PL between ATT and PL, and the definition of the mapping relationship is described later.

(iii) The element att in ATT exists in two forms: ATT = {atomic, set}, that is, att exists in set form or in atomic form. The set contains at least two atomics. There are many possible element forms in PL, namely, PL = {high, mid, low} or $\mathrm{PL} = \{\mathrm{prioritylevel}_1, \ldots, \mathrm{prioritylevel}_i, \ldots, \mathrm{prioritylevel}_n\}$.

(iv) Service Entity (SE) is the entity of the service component in the model. SE contains two groups of content: SEATT and SEPL. SEATT is the ATT included in SE. System operation (SOP) generates SE components and represents system access policy. These two characterize the service entity and describe the authority of the service entity. There is a mapping relationship SEATT $\longrightarrow$ SEPL. Role Entity (RE) is the entity of the role component in the model. Role Operation (ROP) is generated RE internally and representative custom access policy. RE also including an entity user ID (UI) is used to record certain continuous access processes. The role of the UI is to ensure the ability to reconnect after the access behavior is disconnected.

Model composition:

(i) Each attribute att in ATT maps R to an attribute value, which can be formally defined as

$$\mathrm{att}: R \longrightarrow \begin{cases} \mathrm{Array}(\mathrm{att}) \cup \{\perp\} \mathrm{attType}(\mathrm{att}) = \mathrm{atomic}, \\ 2^{\mathrm{Array}(\mathrm{att})} \mathrm{attType}(\mathrm{att}) = \mathrm{set}, \\ \quad \mathrm{Array}(\mathrm{att}) \cup 2^{\mathrm{Array}(\mathrm{att})} \mathrm{attType}(\mathrm{att}) \\ = \{\mathrm{set} \cup \mathrm{atomic}\}. \end{cases} \tag{1}$$

(ii) There is a mapping relationship RA between R and ATT: $R \longrightarrow \mathrm{ATT}$, equivalently $\mathrm{RTA} \subseteq R \times \mathrm{ATT}$.

(iii) Each priority level pl in PL maps ATT to a priority level value, which can be formally defined as

$$\mathrm{pl}: \begin{cases} \mathrm{att}_i \longrightarrow \mathrm{pl}_i \mathrm{att} = \mathrm{Array}(\mathrm{att}) f_1, \\ \mathrm{att}_j \longrightarrow \mathrm{pl}_j \mathrm{att} = 2^{\mathrm{Array}(\mathrm{att})} f_2, \\ f_1 \cup f_2 \mathrm{att} = \mathrm{Array}(\mathrm{att}) \cup 2^{\mathrm{Array}(\mathrm{att})}. \end{cases} \tag{2}$$

(iv) There is a mapping relationship AP: ATT $\longrightarrow$ PL, equivalently $\mathrm{ATP} \subseteq \mathrm{ATT} \times \mathrm{PL}$.

(v) There is a mapping relationship AS: ATT $\longrightarrow$ SEATT, equivalently $\mathrm{ATSA} \subseteq \mathrm{ATT} \times \mathrm{SEATT}$.

(vi) There is a mapping relationship PS: PL $\longrightarrow$ SEPL, And, there is only one element sepl in SEPL. The element type is sepl $\in$ PL , equivalently $\mathrm{PTSP} \subseteq \mathrm{PLL} \times \mathrm{SEPL}$.

(vii) There is a mapping relationship SS in the SE entity: SEATT $\longrightarrow$ SEPL, equivalently, $\mathrm{SATSP} \subseteq \mathrm{SEATT} \times \mathrm{SEPL}$.

(viii) $\exists f$: IS $\longrightarrow$ UI, Then ITU is a restricted mapping of f: f/IS_1, f/IS_1: $\mathrm{IS}_1 \longrightarrow$ UI, and $\mathrm{IS}_1 \subseteq$ IS, equivalently $\mathrm{ITU} \subseteq \mathrm{UI}^{\mathrm{IS}}$.

Hierarchical service:

(i) $\mathrm{SH}_1 \subseteq \mathrm{SEATT} \times \{\mathrm{SEATT} \cup \mathrm{ATT}\}$, a partial order relation $\succcurlyeq_{\mathrm{SHATT}}$ on SEATT, equivalently parentSEATT: SEATT $\longrightarrow 2^{\mathrm{SEATT}}$, mapping service to a set of parent services in the hierarchy. The elements in 2^{SEATT} have a mapping relationship parentSEPL: $2^{\mathrm{SEATT}} \longrightarrow \mathrm{SEPL}_P$, $\mathrm{SEPL}_P \subseteq \mathrm{SEPL}$.

(ii) SH_2: $(\mathrm{sepl}_{p1} \wedge, \ldots, \wedge \mathrm{sepl}_{pn} \wedge pl_1, \ldots \wedge pl_i \ldots, \wedge pl_n) \longrightarrow \mathrm{sepl}$, $(\mathrm{sepl}_{p1} \ldots \mathrm{sepl}_{pn}) \in \mathrm{SEPL}_P$, $(pl_1 \ldots pl_i \ldots pl_n) \in \mathrm{PL}$, sepl $\in$ SEPL. The mapping relationship exists between the mapped SEPL and the mapped SEATT.

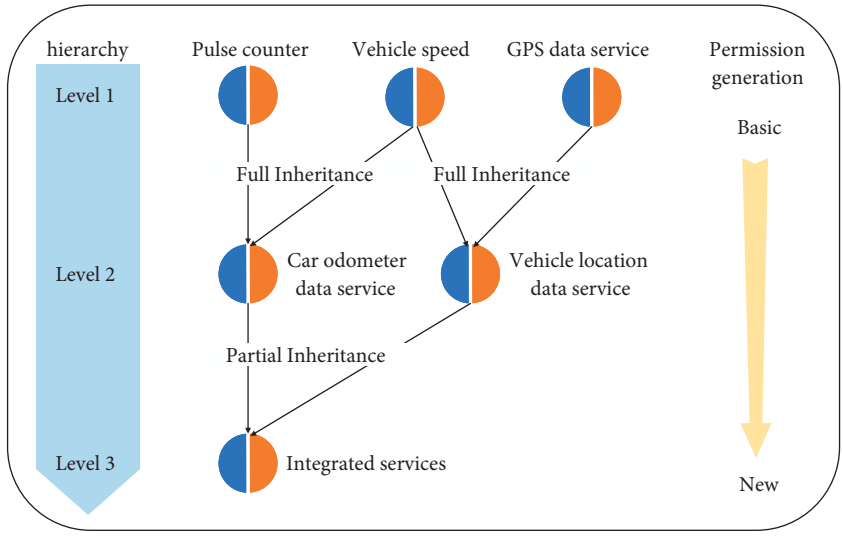


Figure 1: Hierarchies.

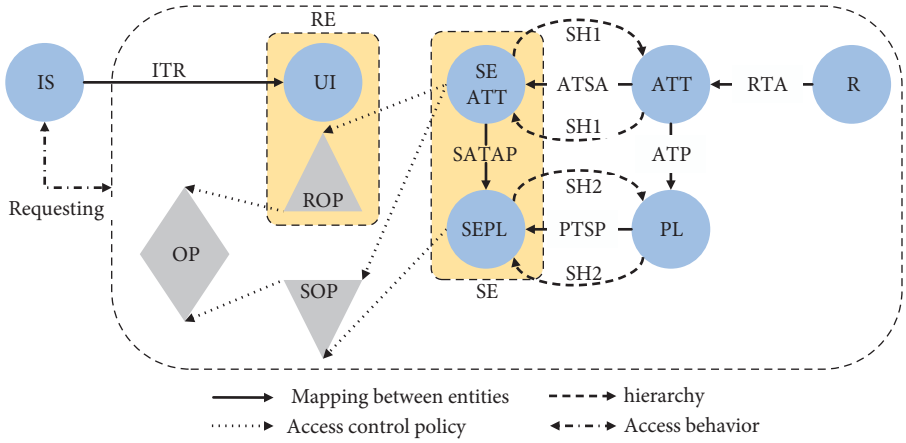


Figure 2: Architecture diagram of the S-RABAC(V) access control model.

Access control policy:

(i) For any atomic attribute $\{att_i\}$ in ATT, $\exists$ Attribute predicates ap: att, $\propto$, val, where att represents the attribute name, $\propto$ refers to the operator $(=, \neq, <, >, \leq, \geq, \gtrless, \lesssim$, etc), and val represents the value range corresponding to the attribute. When attType(att) = set, att may contain a triplet, which is att_i, $\propto$, att_j,which represents the relationship between atomic of att when attType(att) = set. ap can be written formally as

$$\begin{aligned} &\text{ap(atomic)}: \langle \text{att} \propto \text{val} \rangle, \\ &\text{ap(set)}: \begin{cases} \langle \text{att} \propto \text{val} \rangle, \\ \langle \langle \text{att}_i \propto \text{att}_j \rangle \propto \text{val} \rangle. \end{cases} \end{aligned} \tag{3}$$

(ii) APA = ap_1 (atomic)$\wedge \cdots ap_i$ (atomic)$\cdots \wedge ap_n$ (atomic) is the access constraint of att of atomic to resources, APS = ap_1 (set)$\wedge \cdots ap_i$ (set)$\cdots \wedge ap_n$ (set) is the resource access restriction of att of set.

(iii) SOP$\longleftarrow$(APA, APS, SEPL, ACT), ROP$\longleftarrow$(APA, APS, ACT), OP$\longleftarrow$(SOP, ROP). The values of SOP, ROP, and OP are, in two cases: *permit, deny*. The specific conditions of OP, as the final access control policy, can be formally defined as

If SOP = permit, ROP = deny: OP = deny case one,

If SOP = permit, ROP = permit: OP = permit case two,

If SOP = deny: OP = deny case three,

If SOP = permit, ISSEPL$\uparrow$SEPL: OP = permit case four. (4)

Among them, *case four* is the model privilege mechanism, ISSEPL represents the level of priority granted to the IS

vehicle network to access objects, SEPL is the level of authority for the current service, and ISSEPL↑SEPL means ISSEPL is higher than SEPL, corresponding to the upper layer service of SEPL. At this time, the system adjusts the relationship between the current user and the service, pointing it to the service entity above the current service, and granting access. For the specific implementation of the privilege mechanism of tracing upwards, see Algorithm 1 in 4.2.4. The privilege mechanism gives the current user access to services beyond the service capabilities (the inherited content between services does not always inherit the original content, but inherits restrictively and selectively), mainly for emergencies.

4.2. Service Definition and Authority Establishment

4.2.1. Service Definition. The service is the interaction bridge between the access control model and the user. The user uses the service name to apply for the corresponding data service authority. The internal structure of the service is shown in Figure 3.

The priority level between services is got from the inherited original service authority value and the authority value of the new attribute. The authority value of each service is calculated according to the following formula:

$$\text{newpl} = \begin{cases} \left\lceil \dfrac{\sum_{i=1}^{n} \text{spl}_i + \sum_{j=1}^{m} \text{pl}_j}{n+m} - \dfrac{1}{2} \right\rceil + 1, & \dfrac{\sum_{i=1}^{n} \text{spl}_i + \sum_{j=1}^{m} \text{pl}_j}{n+m} - \dfrac{1}{2} \neq N, \\ \dfrac{\sum_{i=1}^{n} \text{spl}_i + \sum_{j=1}^{m} \text{pl}_j}{n+m} + \dfrac{3}{2}, & \dfrac{\sum_{i=1}^{n} \text{spl}_i + \sum_{j=1}^{m} \text{pl}_j}{n+m} - \dfrac{1}{2} = N, \end{cases} \tag{5}$$

where spl_i represents the inherited parent service authority value, where the new service has inherited n parent services. pl_j represents the authority value corresponding to the newly added attribute and there are m new attributes. The sum of the authority values of the two is added and the average value is calculated. The average value is rounded up to get the final new service authority value newpl.

Note: The priority level represents the importance of the resource and is one indicator to determine the overall resource security level. For some composite services, multiple resources are included. Although their resource importance may not be high, the large number of attributes corresponding to many data items implies a high security level. The number of attributes and the priority level are used to measure the permission size of a permission unit comprehensively, implicitly designing a dual-indicator approach to permission measurement.

4.2.2. Resource List Structure. The resource list is mainly used to help manage each service module in S-RABAC(V), which is stored in each service, and each service has its own resource list, whose basic structure is shown in Table 2.

4.2.3. Service Creation and Deletion. In S-RABAC (V), services are created from new attributes and previous services, and the data are abstracted to get new attribute values. So how do these new attributes determine which existing services to combine with? There are two methods; the first one is vehicle local resource combination, i.e., using the vehicle's own inherent resource relationship for combination. This combination simply follows the vehicle architecture, and each of its resulting services is also the basis for subsequent service expansion. The other method is custom addition, as shown in Figure 4. This method of service creation is mainly based on the first service with a lot of expansion, and each custom service is built on top of the existing base service. When a custom service is created, the resource list in the service can be traversed from the bottom to the top until the required resource type is found. Custom framing services themselves are managed and documented as separate systems.

Bottom layer services are like leaf nodes in a tree, and they are often the ones most likely to generate the need to add or remove (the root service of the model is at a top layer). Bottom layer services can be added or removed without affecting the use of upper layer resource services, which can be adjusted at will. Based on the stability of the model, it may not delete the high-level root service arbitrarily. But if there is a demand for deletion, the specific method is to first find the sub-services of the service and delete the call statement of the service module in the sub-services.

4.2.4. Privilege Mechanism and Authority Setting. Priority identifies the importance of each resource, with more important resources having a higher priority. Priorities are ranked from high to low. Each service has its own level of authority, with decreasing priority from high. The importance level of a resource collection (service) depends not only on the number of attributes of the resource but also on the permission level of the current service. This design gives great flexibility and operability in the design of resource collection permissions.

There are two major cases of privilege mechanisms, triggered by privileged services. A privileged service is not a separate service organization model but a "normal service" that is given priority above its own level. The first case is between different priority levels. Since the service comprises a parent service and a new attribute, and the new attribute does not have the space for upward permission, this case is

```
(1) Input: Prioritylevel, Goalprioritylevel, User, Role
(2) Initialize the Resource storage.
(3) Initialize the User storage.
(4) Output: Prioritylevel, Role.
(5) User storage ⟸ User
(6) Priorityadjust(Userid, Prioritylevel, Goalprioritylevel)
(7) if Prioritylevel ! = Goalprioritylevel then
(8)     Prioritylevel = Prioritylevel-1
(9)     Priorityadjust(UserID, Prioritylevel, Goalprioritylevel)
(10) else
(11)        Role ⟸ User storage.
(12)        User ⟸ Resource storage.
(13)        Return Role, Prioritylevel
(14) end if
(15) End
```

Algorithm 1: Priority Adjust.

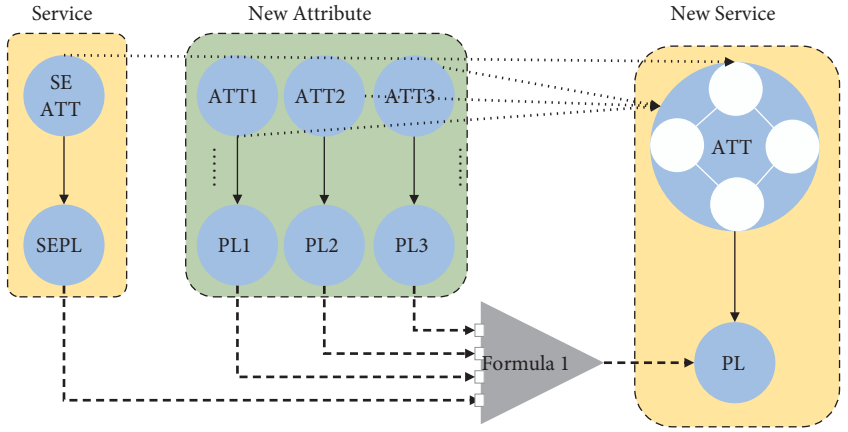


Figure 3: Service permission inheritance mechanism explained.

Table 2: Resource list structure.

Service name	New attribute list	Parent service1	Total inherited attributes	Inherited attributes set1 Inherited attributes set2 Inherited attributes set3	Resource function Resource function Resource function	Child service list
		Parent service2	Total inherited attributes	.....		

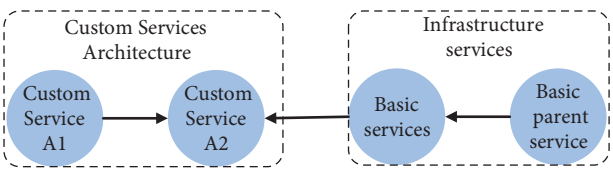


Figure 4: Generation and expansion of services.

just upward to the parent service of the corresponding level. In the process of adjusting permissions upward, with multiple service inheritance, the parent service is traced according to the service corresponding to the adjusted permission level. If there are multiple parent services with the same permission level, then the service with more inherited attributes is selected as the parent service to be traced upward.

The second case is for services between the same priority, in which case the parent service with the most inherited attributes from the resource list is traced directly. The privilege mechanism's privilege traceability can be customized besides the system settings. The algorithm description for the privilege mechanism is shown in Algorithm 1.

4.2.5. Role Setting in S-RABAC(V). The purpose of the role module in S-RABAC(V) is mainly to separate access control policies and disconnected access reservations for specific access processes. It protects core policies in the access control system by separating access control policies. And, when facing the situation of unstable connection of the vehicle network, the access control model is given the ability of "permission reservation" to enhance the efficiency of the access control model.

Once the service module completes the first step of systematic access policy validation, it transmits the validation results to the role entity, which decides whether to allocate resources to users based on the results, and rejects the allocation directly for illegitimate users. For legitimate users, the role module will continue to run custom access control policies internally, and still refuse to allocate resources when the user does not meet the custom access control policy. When a user satisfies a custom policy, the requested resource is invoked for the user and the user is assigned an access role. The role assigned to the user maintains the user information for a period after the user disconnects. If the user accesses the resource again within this period, the access verification process can be skipped and the access process resumed to directly access the resource.

4.3. Access Control Process. Figure 5 depicts the steps of executing the S-RABAC (V) model to complete an access request, as shown below:

(1) The client uploads a request to the IoT cloud platform, and the upload request contains the service name to be accessed and the verification content required by the service.

(2) The Internet of Things cloud platform transfers the content in the request to the access policy enforcement point in the cloud service according to the cloud transfer rules. At the policy enforcement point, the service verifies the upload attributes. If the verification is qualified, then proceed to the next step.

(3) After the attribute verification is completed, if the permission attribute in the request differs from the service permission value recorded in the service, the current service direction is changed according to the access policy, and the new service is specified as the user request service. If the permission in the attribute's request is the same as the service authority value recorded in the service, the current service direction is not changed.

(4) After completing the third step, submit the processing result to the role module. The system refuses to assign roles to illegal-access users, and the user is notified of the problem in the request. Legal users receive custom access to the role module in the last step. For control settings verification, users who pass the verification are assigned roles and access the corresponding data, and users who fail the verification are rejected here.

Figure 6 describes the service establishment process in the S-RABAC (V) model, as described below:

(1) The system administrator establishes a new service based on a certain service and new attributes, and provides relevant information about the new service to the access control policy center.

(2) The policy center first checks whether the new service name is in the sub-service list of the original service resource list and whether the same service name and service attribute set already exist. If it does not exist, it proceeds to the third step. Otherwise, the new service creation is rejected.

(3) The strategy center searches for the resources to be inherited by the current new service in the resource list, and uses the corresponding attributes of the resources as the inherited attributes in the new service.

(4) The new resource abstracts the attributes of the new dataset and assigns permission values. The new attributes and the original service permission values get the new service permission values according to the method in Formula (5), and the inherited attribute modules in the original service are combined with the new attributes into a new service.

(5) The new attribute set gets a new service, the system administrator is notified of the new service name and service verification contents, and the new service is established.

5. Results

In this paper, we rely on Alibaba Cloud Computing for experimental simulation. We use the IoT platform in Alibaba Cloud Computing as an IoV platform and use the Message Queuing Telemetry Transport (MQTT) protocol for information transfer to achieve the simulation effect. When a user sends an access request, the access request arrives at the IoT platform with service information and attribute information. The information delivered in the protocol is initially processed in the IoT platform and passed through the

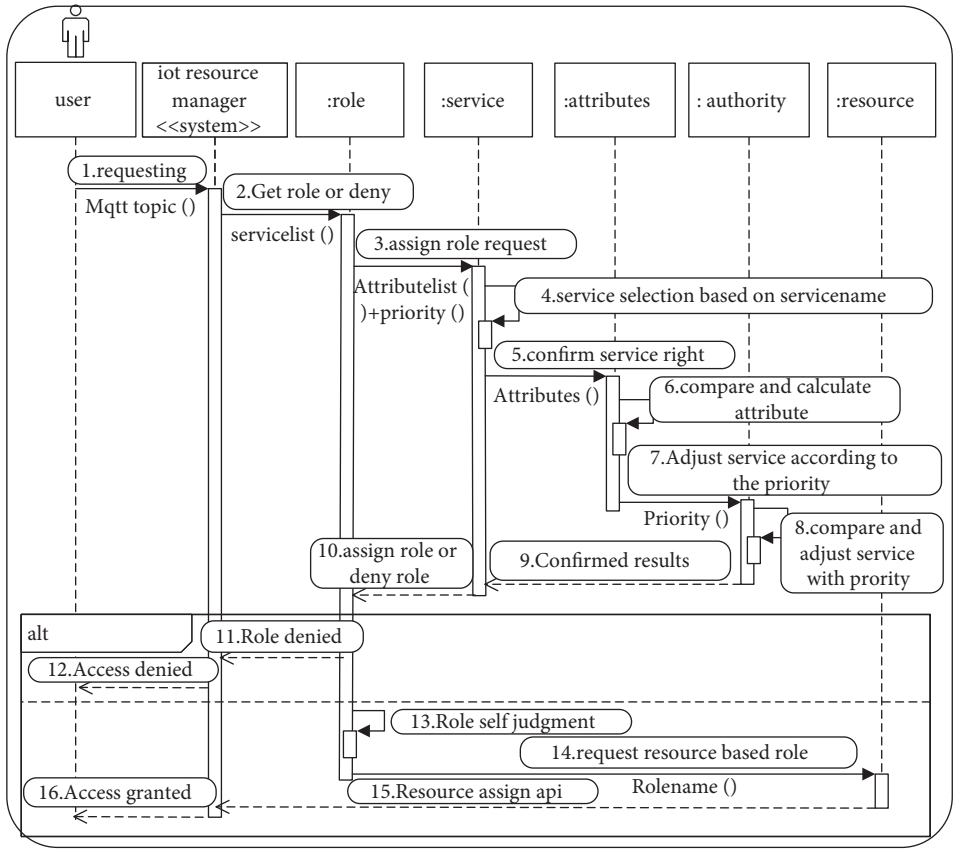


Figure 5: The access process of S-RABAC(V).

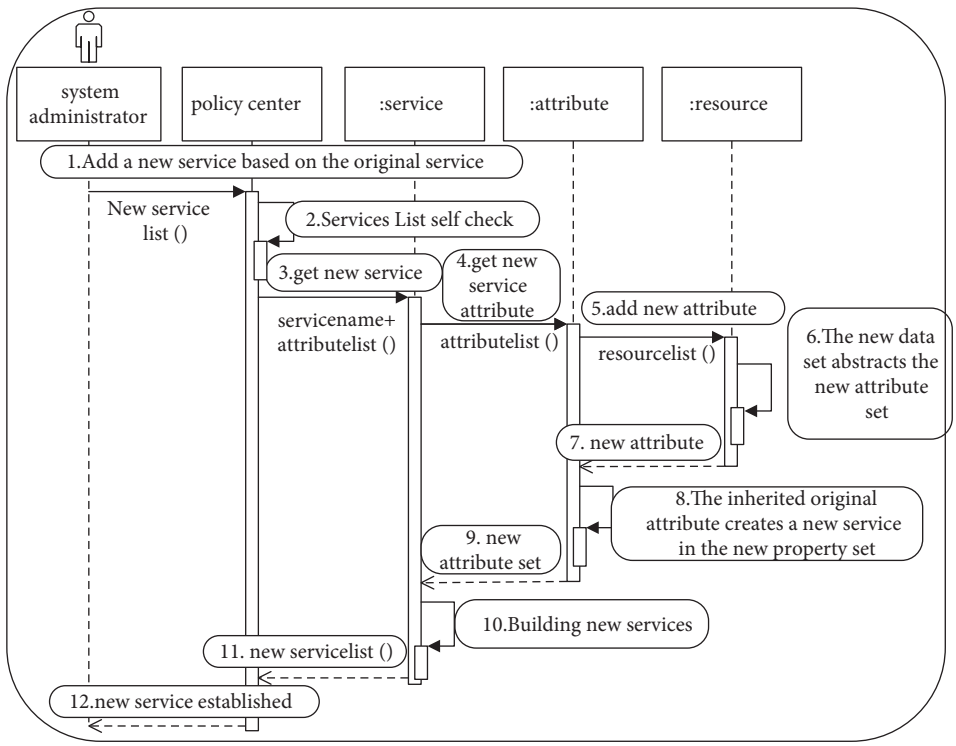


Figure 6: Setting up a service module.

database language to the final function calculation module [42]. As soon as the function calculation unit receives the delivered access request, it immediately executes the access control policy and then returns the policy execution results to the user using the IoT platform api. The entire experiment utilizes the Alibaba Cloud Computing IoT platform MQTTsdk to build the access control system client [43], which serves as the issuing end of access requests and system management requests. The function calculation module of Ali Cloud Computing is used as the place where the access control policy is actually executed.

The experiments use python to simulate vehicles to upload vehicle data or requests to both the Baidu Eagle Eye platform [44] and the IoT platform of Alibaba Cloud Computing. The properties and service settings of the experiment are shown in Figure 7. The resource lists of the three services in the figure are shown in Table 3, and the priority levels of the attributes and the correct attribute values are recorded in Table 4.

5.1. Implement Service-Based Access. This experiment simulates five access control scenarios. Situation 3 shows the case of continuous data service disconnection and reaccess. And, Situation 4 is the demonstration of privileged access mechanism. The location data uploaded by the vehicles in the experiments are from Microsoft [45, 46].

Situation 1. The user enters the correct service selection, but the entered attributes are incorrect. As shown in Figure 8, a user with an ID of 1996 accesses the "serviceGPS." The service name is correct, but the authentication attribute in the service is wrong. The wrong access attribute is judged in the function computing server as access failure, and the user is notified of the result.

The result shows that the service corresponds to attribute error, the access failed attribute is incorrect, and access to data failed.

Situation 2. This is the legal access process. As shown in Figure 9, the user with ID 004 accesses the "serviceGPS" and submits the correct access attributes. After being identified, the access control system assigns an identity to the user and transmits the data to the user.

The result shows that Access succeeded role004 1116.48067 39.98843, the attribute is correct, and the access is successful.

Situation 3. Instability in the connectivity of IoV devices is common for some continuously accessed resources, such as vehicle trajectories on a vehicle's intended travel route. If the current access port is closed when the connection between the vehicle and the data center side is lost, then it will require the vehicle node to reestablish a new path service when the connection is restored. For this case, S-RABAC (V) provides a disconnected service hold feature to provide disconnected recovery data access for certain persistent resource access process.

For the convenience of demonstration, the experiment uses python dictionary-type data instead of cloud stream database. When the user with id 001 logs out of the access system, the system sets aside 3 seconds for the user. If user 001 reestablishes the connection within 3 seconds, user 001 can continue the current access process (3 seconds belong to the experimental custom time setting, the actual time setting can be changed according to the characteristics of the service).

As shown in Figure 10, the 001 user accesses the "serviceGPS." Because the user already exists in the system and the disconnection time is not long, user 001 skips the verification phase during this visit and directly accesses the data. The result shows: allocated 116.44903 39.94121. The user was allocated and got data access.

Situation 4. This experiment imitated the environment sensor to change the priority of the uploaded service, and to change the priority of the "groupservice" from 2 to 1. Thus, in the access control system, the identity of its parent service should be assigned. According to the inheritance relationship, the authority should be adjusted to the "serviceGPS" and assigned the corresponding role. It showed the access process in Figure 11.

The results showed that the permissions were adjusted and assigned to the appropriate services, and the corresponding data were procured.

5.2. Access Control between Mobile Nodes. This experiment embodies a more fine-grained form of access control in S-RABAC (V), i.e., node-to-node access control. This access control is more than access to nondata resources, such as a car accessing another car, a vehicle accessing a road base. This form of access is more common in vehicular networking compared to data resource access.

The experiment assumes that when the vehicle "Bob's car" passes through region B and goes to region A, it wants to access services related to region A, such as the locations of all supermarkets in the region, the coordinates of "Bob's car" itself, and other vehicle nodes. However, "Bob's car" is restricted to access these contents only in region B, so when "Bob's car" crosses region B to region A, it does not have access to these nodes in region B, so when "Bob's car" wants to access the vehicle "Alice's car" in region A, the access request is rejected. The experiment first set up and partitioned the map for the whole access process, and the map partitioning is shown in Figure 12 (the experiment used the third service "findid" in 5.1).

In the experiment, "Bob's car" starts from the vehicle starting point in the figure and finally arrives at the vehicle request point to send access request, while "Alice's car" as the vehicle visited by "Bob's car" moves, as shown in Figure 13. Figure 13 shows the location information of the two cars on the actual map during the entire process. The location information comes from the trajectory query function of Baidu Eagle Eye service, and the coordinate position data come from literature 45 and 46.

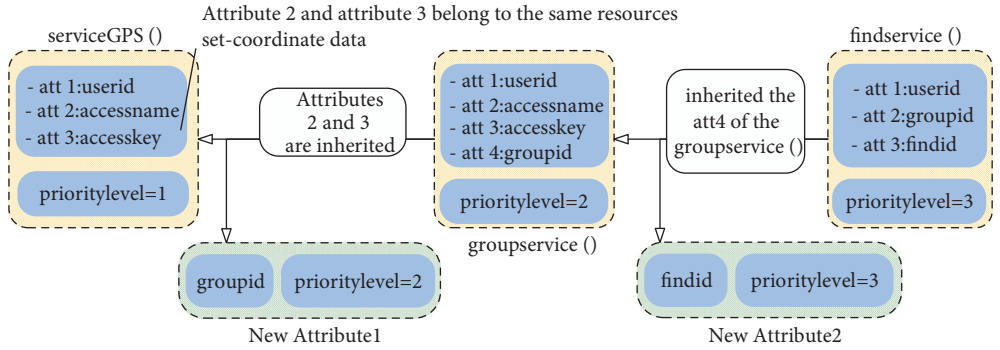


Figure 7: Experimental design of S-RABAC(V).

Table 3: Resource list.

Service name	New attribute	New attribute function	Parent service	Inherited attribute	Child service
serviceGPS	Accessname, accesskey	Longitude and latitude	Null	Null	Groupservice
Groupservice	Groupid	Vehicle area grouping	serviceGPS	Accessname, accesskey	Findservice
Findservice	Findid	Regional vehicle search	Groupservice	Groupid	Null

Table 4: Attribute comparison table.

Attribute name	Priority level	Correct value
Accessname, accesskey	Priority level:1	Cloud, 1
Groupid	Priority level:2	2
Findid	Priority level:3	3

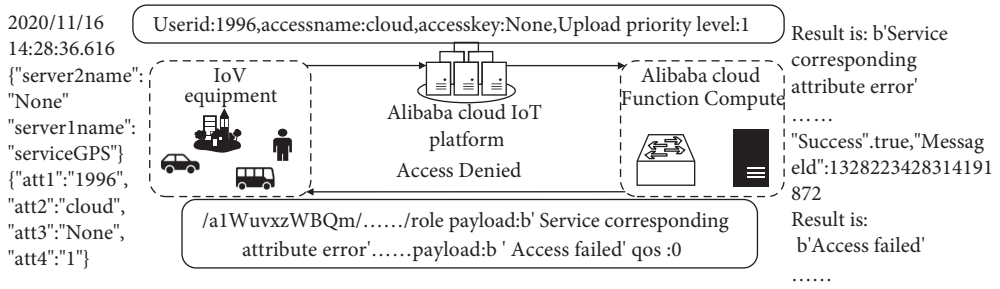


Figure 8: Wrong service attributes.

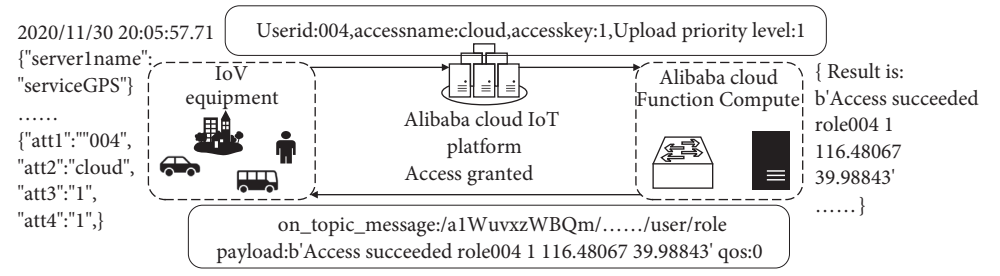


Figure 9: A legitimate access process.

In the experiment, to meet the needs of the simulation, "Alice's car" uploaded the coordinate data to the cloud two minutes after "Bob's car" uploaded the data to the cloud. "Bob's car's" access request will be received by "Alice's car" at 13 : 54 if no area restriction is applied, as shown in Figure 14. "Alice's car" client receives the message: "Bob calls at 116.3023739.95867."

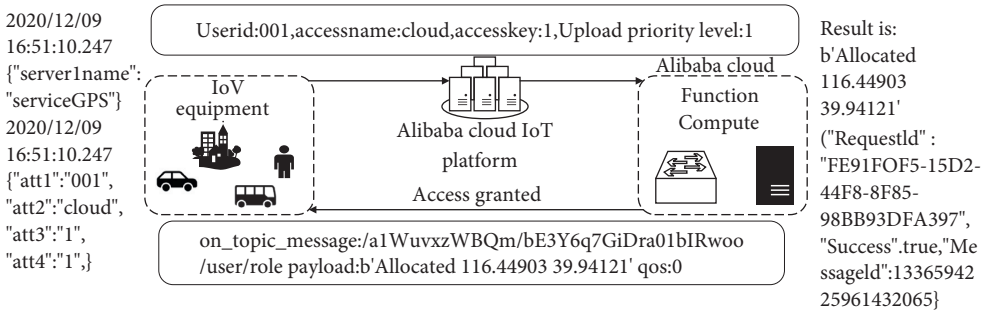


Figure 10: Permission reservation.

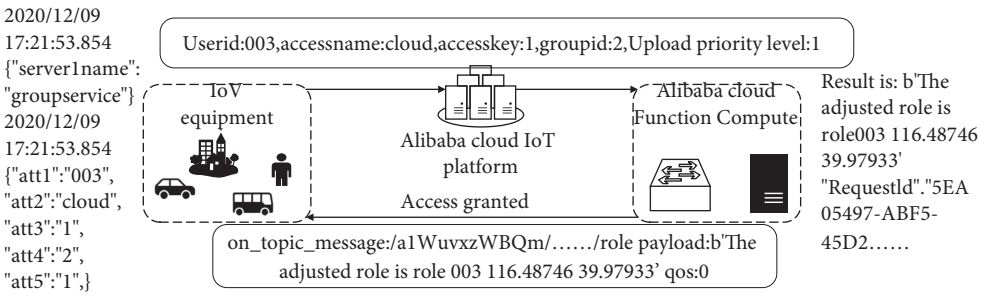


Figure 11: Privilege access mechanism.

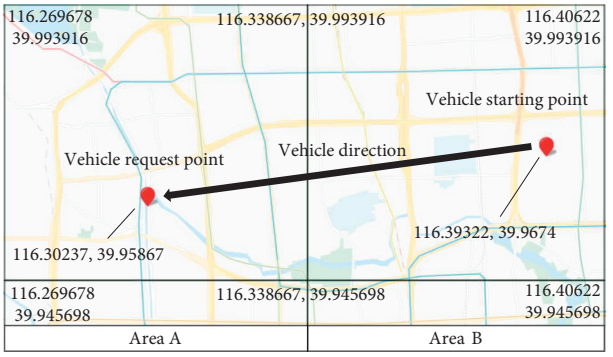


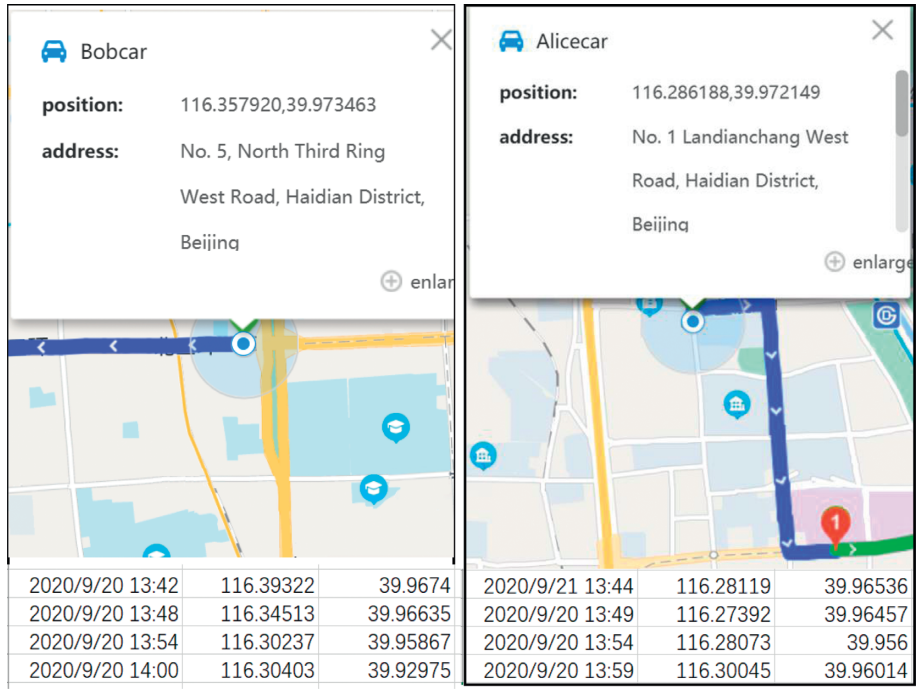
Figure 12: The area where the vehicle is located and the point where the access request is issued.

However, after the access control restriction, "Bob's car" crossed the boundary of area B into area A and sent an access request to "Alice's car" at 13 : 54. S-RABAC (V) denies the access request and informs it is out of bounds. As shown in Figure 15, the "Bob's car" client receives the message: "No permission."

6. Discussion

6.1. Security Analysis. We base the overall security analysis of the access control model on the Alibaba Cloud Computing implementation method adopted in this article. The following is the specific security analysis process.

The entire process of access control goes through three stages: client, third parties, and cloud. In this article, we used the Alibaba Cloud Computing IoT platform as the initial access request acceptance point, and the cloud handles the implementation of access control strategies and user data collection. Because of the powerful functions of the cloud, user data and its access control policies are secure. But, outside the cloud, the access control model has certain security risks. The first obvious security hazard is when access control is completed, as data forwarded by third parties to users are exposed. Such problems actually exist in the model [20, 30–32, 34, 39]. Second, the honesty of third parties cannot be guaranteed. When the third party is a corrupt authorized entity, it will cause errors in the access control process and leak related data in the settings. For this kind of problem, there is a more perfect solution, i.e., using attribute encryption as the access carrier of the overall model to spread, and changing the original simple matching process of access control model attributes into a ciphertext key decryption process. We can use the encryption algorithm proposed by



2020/9/20 13:42	116.39322	39.9674	2020/9/21 13:44	116.28119	39.96536
2020/9/20 13:48	116.34513	39.96635	2020/9/20 13:49	116.27392	39.96457
2020/9/20 13:54	116.30237	39.95867	2020/9/20 13:54	116.28073	39.956
2020/9/20 14:00	116.30403	39.92975	2020/9/20 13:59	116.30045	39.96014

Figure 13: Vehicle position coordinates from the Baidu Eagle Eye server. Since the deviation correction function of the Baidu Map Eagle Eye service is enabled, the coordinates displayed on the map differ slightly from the actual coordinates.

```
property_data post success:0, request_id:'2'
on_thing_prop_post request id:2, code:200, data:{} message:success
116.28073
39.956
1600667695.0
<Response [200]>
property_data post success:0, request_id:'3'
on_thing_prop_post request id:3, code:200, data:{} message:success
on_topic_message:/a1WuvxzWBQm/LfVxRwjpNtnFUcNCpNBe/user/role payload:b'Bob calls at 116.3023739.95867' qos:0
116.28221
39.95597
1600667695.0
<Response [200]>
property_data post success:0, request_id:'4'
```

on_topic_message:/a1WuvxzWBQm/LfVxRwjpNtnFUcNCpNBe/user/role
payload:b'Bob calls at 116.3023739.95867' qos:0

Figure 14: Alice's car client.

```
116.30403
39.92975
1600668419.0
<Response [200]>
property_data post success:0, request_id:'3'
on_thing_prop_post request id:3, code:200, data:{} message:success
on_topic_message:/a1WuvxzWBQm/LJQE9SbyWcVVC4R0GZAB/user/role payload:b'No permission' qos:0
116.30388
39.9153
1600668807.9999998
<Response [200]>
property_data post success:0, request_id:'4'
```

on_topic_message:/a1WuvxzWBQm/LJQE9SbyWcVVC4R0GZAB/user/role
payload:b'No permission' qos:0

Figure 15: Bob's car.

RW15 [47] to improve the access control process. The specific approach is as follows:

(1) Upload the attributes of the access request according to the organization of the service.

(2) The cloud receives the corresponding attribute set and generates the key corresponding to the attribute. Because of the encryption algorithm setting of the multi-authority, multiple clouds can exist in this step.

(3) If the attribute set meets the server-side attribute set requirements, the ciphertext sent by the cloud can be decrypted, otherwise the decryption cannot be completed.

The only difference between the above process and the original access control process is: The original access control process directly produces access control results. The result of access control after applying attribute encryption is passively generated, i.e., whether the key can release the ciphertext to generate the corresponding plain text.

6.2. Access Control Time. Also, based on implementing the Alibaba Cloud Computing, this article analyzes the efficiency of the access control model S-RABAC (V). Figure 16 shows the time spent in the process of five independent visits under the Alibaba Cloud service. In Table 5, we show the results of the access time statistics analysis. It can be seen from the figure and table that assigning roles to access users can shorten the time of access control and improve the efficiency of access control. Note that the role here is mainly used to assist the ABAC model, and its function differs from that of the role in RBAC.

The increase in the number of access policy decisions with the addition of priority rules is shown in Figure 17, without distinguishing between access and denial. We assumed three priority levels, namely, high, mid, low, and the attribute value generated by participating in the decision could be 3-tuple, 2-tuple, or 4-tuple. As shown in the figure, because of the existence of priority rules, there are more authority decisions under the same number of attributes. These decisions are generated based on the original attribute set decisions and are an expansion of the original decisions. A typical form for two vehicles with the same attributes but different priorities is that the higher-priority vehicle can access the relevant data of the area, but the lower-priority vehicle is denied access or can only access part of the data. Figure 18 shows the impact of the increase in the number of attributes on the time of the entire access control process of S-RABAC(V). The increase in the number of attributes mainly affects the execution speed of cloud access control policies. However, because of the powerful performance of the cloud server, the number of attributes does not have a significant impact on the overall access time.

6.3. Function Comparison. This article compares several other access control technologies and lists their characteristics, as shown in Table 6:

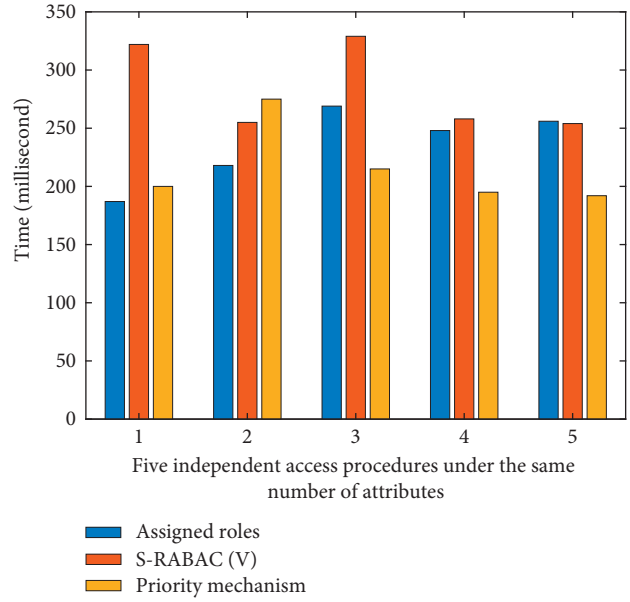


Figure 16: S-RABAC(V) performance.

Table 5: Access time statistics analysis.

	Assigned roles	S-RABAC(V)	Privilege mechanism
First access	187	322	200
Second access	218	255	275
Third access	269	329	215
Fourth access	248	258	195
Fifth access	256	254	192
SUM	1178	1418	1077
AVG	235.6	283.6	215.4
Time unit	millisecond	millisecond	millisecond

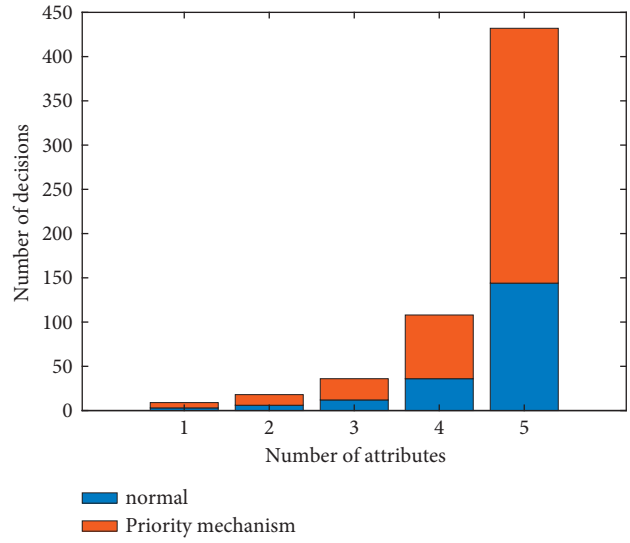


Figure 17: Comparison of the number of decisions.

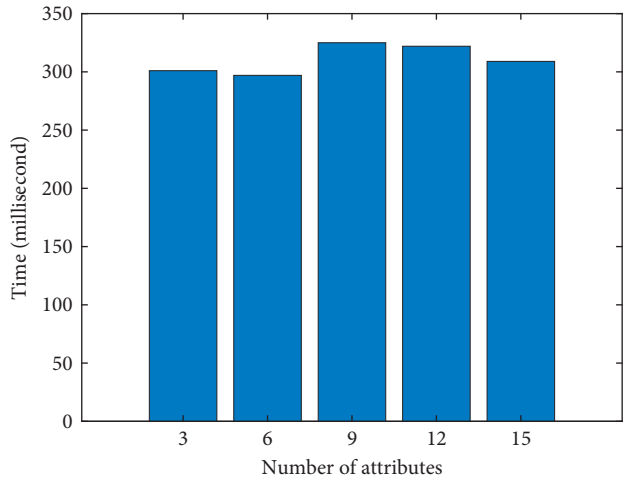


Figure 18: The number of attributes affects the access time.

Table 6: Comparison of access control models. (✓ indicates that it has the function, ✗ means not having).

Scheme	Cr_1	Cr_2-	Cr_3	Cr_4	Cr_5	Cr_6	Cr_7	Cr_8	Cr_9	Cr_{10}	Cr_{11}
Ref. [20]	Low	Medium	Low	✓	✓	✗	✗	✗	✗	✗	✓
Ref. [26]	High	Medium	High	✗	✓	✗	✗	✗	✗	✗	✓
Ref. [30]	High	Low	High	✗	✓	✓	✗	✗	✗	✓	✓
Ref. [31]	High	Medium	Medium	✗	✓	✓	✗	✗	✗	✗	✗
Ref. [32]	High	Medium	High	✗	✓	✓	✗	✗	✗	✓	✓
Ref. [33]	Medium	✗	Based on attribute combination	✗	✓	✗	✗	✗	✗	✗	✗
Ref. [39]	High	Medium	High	✗	✓	✓	✓	✗	✗	✗	✓
Our scheme	High	Medium	Medium	✗	✓	✓	✓	✓	✓	✓	✓

*Cr-Criterion; Cr_1- Scalability; Cr_2- Context awareness; Cr_3- Granular; Cr_4- Delegation support; Cr_5- Dynamic; Cr_6-hierarchies; Cr_7- Priority mechanism; Cr_8- Adjustable permissions; Cr_9- Permission reservation; Cr_{10}- Permission inheritance mechanism; Cr_{11}- Access control model practical running experiments.

As Table 6 shows, S-RABAC (V) has better scalability and basic context-awareness because of the attribute aggregation mechanism. In addition, the ordered aggregation of attributes puts the entire access control model at a moderate granularity of permissions, which is required by IoV. The cumbersome permission design can make it difficult for individuals in IoV to use because users cannot manipulate the access control model in a fine-grained way.

Although no delegation mechanism is designed in S-RABAC (V), the hierarchy can provide a variety of different permission setting needs. In fact, although the delegation mechanism can dynamically realize different authority requirements, it is more prone to authority disputes and authority proliferation than a static hierarchy. In addition, S-RABAC (V) generates unique adjustable permissions and permission reservation capability under the priority mechanism and permission inheritance mechanism. These features can enhance the dynamic change of the hierarchy in the vehicle network. Imagine that in IoV, where individual scenarios are frequently switched, users can complete permission switching without reestablishing the access control process, which not only provides convenience for IoV users but also improves the efficiency of the entire access control model. This is crucial in IoV, a scenario that requires high efficiency in information exchange.

We conclude with a preliminary implementation of S-RABAC (V) using Alibaba Cloud Computing and show the various access scenarios we propose. The cloud platform access control architecture ideas presented in the experiments can be widely used for various cloud access control techniques.

7. Conclusions

In this paper, we have proposed a new access control model for protecting the information exchange process of IoV cloud architectures. The proposed S-RABAC (V) is not only able to establish the appropriate granularity of privileges when facing the enormous resources to be protected in IoV under cloud shelves but also to meet various requirements of dynamic scenarios. Our proposed S-RABAC (V) adds some extra features to enhance the flexibility of the access control model based on RBAC and ABAC. Compared with the access control models in papers 20, 26, 30, 31, 32, 34, 41, S-RABAC (V) can provide "Adjustable permissions" and "Permission reservation" as two extra features. It also has "hierarchies" and "priority mechanism" functions. The new function module does not slow down the execution of access control. Instead, the "Permission reservation" feature speeds up specific access control processes, improving the overall

efficiency of access control. In addition, the number of decisions that can be generated by the access control model is increased based on the mechanism of prioritization and attribute combination, which allows users of the access control model to develop more fine-grained access control plans.

Future extension of this work is to combine with CP-ABE to improve the security of the overall access control model. Currently, there are many improved algorithms for CP-ABE as well, such as multi-authorization CP-ABE, hierarchical CP-ABE, and so on. In the next plan, we will try to fuse and improve the CP-ABE algorithm to fit our proposed S-RABAC(V). At the end, the proposed S-RABAC (V) is fused with our future planned improved CP-ABE encryption algorithm to achieve a secure and flexible access control process.

Data Availability

Previously reported GPS data were used to support this study and are available at https://www.microsoft.com/en-us/research/publication/t-drive-trajectory-data-sample/. These prior studies (and datasets) are cited at relevant places within the text as references [45, 46].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the National Natural Science Foundations of China under Grant Nos. 61862040, 61762059, and 61762060.

References

[1] E. S. Ali, M. K. Hasan, R. Hassan et al., "Machine learning technologies for secure vehicular communication in internet of vehicles: recent advances and applications," *Security and Communication Networks*, vol. 2021, Article ID 8868355, 2021.

[2] A.. Maria, P.. Vijayakumar, and L. J. Deborah, "Comprehensive survey on security services in vehicular ad-hoc networks," *IET Intelligent Transport Systems*, vol. 7, no. 6, pp. 379–388, 2016.

[3] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A survey on IoT security: application areas, security threats, and solution architectures," *IEEE Access*, vol. 7, Article ID 82743, 2019.

[4] F. Longo, D. Bruneo, S. Distefano, G. Merlino, and A. Puliafito, "Stack4Things: a sensing-and-actuation-as-a-service framework for IoT and cloud integration," *Annals of Telecommunications*, vol. 72, no. 1-2, pp. 53–70, 2017.

[5] M. Tao, J. Zuo, Z. Liu, A. Castiglione, and F. Palmieri, "Multi-layer cloud architectural model and ontology-based security service framework for IoT-based smart homes," *Future Generation Computer Systems*, vol. 78, pp. 1040–1051, 2018.

[6] M. Gupta and R. Sandhu, "Authorization framework for secure cloud assisted connected cars and vehicular internet of things," in *Proceedings of the 23nd ACM on Symposium On Access Control Models And Technologies*, pp. 193–204, Indianapolis, IN, USA, June 2018.

[7] K. R. Rao, A. Nayak, I. G. Ray, Y. Rahulamathavan, and M. Rajarajan, "Role recommender-RBAC: optimizing user-role assignments in RBAC," *Computer Communications*, vol. 166, pp. 140–153, 2021.

[8] N. Pan, Z. Zhu, L. He, and L. Sun, "An efficiency approach for RBAC reconfiguration with minimal roles and perturbation," *Concurrency and Computation: Practice and Experience*, vol. 30, 2018.

[9] M. Ghafoorian, D. Abbasinezhad, and S. Hassan, "A thorough trust and reputation based RBAC model for secure data storage in the cloud," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 4, pp. 778–788, 2018.

[10] H. C. Chen, "Collaboration IoT-based RBAC with trust evaluation algorithm model for massive IoT integrated application," *Mobile Networks and Applications*, vol. 24, no. 3, pp. 839–852, 2019.

[11] B.. Bruhadeshwar, H.. Kyle, and R.. Indrakshi, "Securing home IoT environments with attribute-based access control," in *Proceedings of the Third ACM Workshop On Attribute-Based Access Control*, pp. 43–53, New York, NY, USA, March 2018.

[12] N. T. T. Hang, N. C. Trinh, N. T. Ban, M. Raza, and H. X. Nguyen, "Delay and reliability analysis of p-persistent carrier sense multiple access for multi-event industrial wireless sensor networks," *IEEE Sensors Journal*, vol. 20, no. 20, Article ID 12414, 2020.

[13] L. Liu, Y. Cao, L. Ding, F. Yang, L. Qian, and C. Zhi, "A priority-enhanced slot allocation mac protocol for industrial wireless sensor networks," in *Proceedings of the 25th Asia-Pacific Conference On Communications (APCC)*, pp. 88–94, Posts & Telecommunicat Inst Technol, Ho Chi Minh City, VIETNAM, November 2019.

[14] J. Liu, M. Agiwal, M. Qu, and H. Jin, "Online control of preamble groups with priority in cellular IoT networks," in *Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications*, pp. 2136–2144, ELECTR NETWORK, Toronto, ON, Canada, July 2020.

[15] Y. Sim and D.-H. Cho, "Performance analysis of priority-based access class barring scheme for massive MTC random access," *IEEE Systems Journal*, vol. 14, no. 4, pp. 5245–5252, 2020.

[16] G. Abbas, Z. H. Abbas, and S. Haider, "Thar Baker, Saadi Boudjit, and Fazal Muhammad. PDMAC: a priority-based enhanced TDMA protocol for warning message dissemination in VANETs," *Sensors*, vol. 20, no. 1, 2020.

[17] P. Vijayakumar, V. Chang, L. Jegatha Deborah, B. Balusamy, and P. G. Shynu, "Computationally efficient privacy preserving anonymous mutual and batch authentication schemes for vehicular ad hoc networks," *Future Generation Computer Systems*, vol. 78, no. 3, pp. 943–955, 2018.

[18] L. Song, G. Sun, H. Yu, X. Du, and M. Guizani, "FBIA: a fog-based identity authentication scheme for privacy preservation in internet of vehicles," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5403–5415, 2020.

[19] Z. Xu, L. Xiong, J. Xu, W. Liang, and K. R. Choo, "A secure and computationally efficient authentication and key agreement scheme for Internet of Vehicles," *Computers & Electrical Engineering*, vol. 95, 2021.

[20] H. B. Djilali, D. Tandjaoui, and H. Khemissa, "Enhanced dynamic team access control for collaborative Internet of Things using context," *Trans. Emerg. Telecommun. Technol.*vol. 32, no. 5, 2020.

[21] J. Lu, Y. Xin, Z. Zhang, H. Peng, and J. Han, "Supporting user authorization queries in RBAC systems by role-permission reassignment," *Future Generation Computer Systems*, vol. 88, pp. 707–717, 2018.

[22] P. Kamboj, S. Khare, and S. Pal, "User Authentication Using Blockchain Based Smart Contract in Role-Based Access Control," *Peer-To-Peer Network and Application*, vol. 14, pp. 1–16, 2021.

[23] W. Bai, Z. Pan, S. Guo, and Z. Chen, "RMMDI: A Novel Framework for Role Mining Based on the Multi-Domain Information," *Security and Communication Networks*, vol. 2019, Article ID 8085303, 15 pages, 2019.

[24] G. Abirami and R. Venkataraman, "Performance analysis of ABAC and ABAC with trust (ABAC-T) in fine grained access control model," in *Proceedings of the 2019 11th International Conference On Advanced Computing (ICoAC)*, pp. 372–375, Chennai, India, May 2019.

[25] H. Ouechtati, N. B. Azzouna, and L. B. Said, "A fuzzy logic based trust-ABAC model for the Internet of Things," *Advanced Information Networking and Applications*, Springer, in *Proceedings of the International Conference On Advanced Information Networking and Applications*, pp. 1157–1168, March 2019.

[26] R. Zhang, G. Liu, S. Li, Y. Wei, and Q. Wang, "AB$_S$AC: attribute-based access control model supporting anonymous access for smart cities," *Security and Communication Networks*, vol. 2021, Article ID 5531369, 11 pages, 2021.

[27] R. Pan, G. Wang, and M. Wu, "An attribute-based access control policy retrieval method based on binary sequence," *Security and Communication Networks*, vol. 2021, Article ID 5582921, 12 pages, 2021.

[28] S. D. Abbasian, K. Farajzadeh, J. Rezazadeh et al., "A survey on data aggregation techniques in IoT sensor networks," *Wireless Networks*, vol. 26, no. 2, pp. 1243–1263, 2020.

[29] F. Álvarez-Bazo, S. C. Sánchez, D. Vallejo, C. G. Morcillo, A. Rivas, and I. Gallego, "A low-cost automatic vehicle identification sensor for traffic networks analysis," *Sensors*, vol. 20, 2020.

[30] G.. Maanak, B.. James, P.. Farhan, and S.. Ravi, "Dynamic groups and attribute-based access control for next-generation smart cars," in *Proceedings of the Ninth ACM Conference On Data And Application Security And Privacy*, pp. 61–72, Richardson, TX, USA, March 2019.

[31] L. Aliane and M. Adda, "HoBAC: toward a higher-order attribute-based access control model," *Procedia Computer Science*, vol. 155, pp. 303–310, 2019.

[32] D. Servos and S. L. Osborn, "HGAA: an architecture to support hierarchical group and attribute-based access control," in *Proceedings of the Third ACM Workshop on Attribute-Based Access Control*, pp. 1–12, Tempe, AZ, USA, March 2018.

[33] M. U. Aftab, Y. Munir, A. Oluwasanmi et al., "A hybrid access control model with dynamic COI for secure localization of satellite and IoT-based vehicles," *IEEE Access*, vol. 8, Article ID 24208, 2020.

[34] M. U. Aftab, Z. G. Qin, S. F. Quadri, Zakria, A. Javed, and X. Y. Nie, "Role-based ABAC model for implementing least privileges," in *Proceedings of the 2019 8th International Conference On Software And Computer Applications*, pp. 467–471, Penang, Malaysia, February 2019.

[35] M. U. Aftab, Z. G. Qin, K. Hussain et al., "Negative authorization by implementing negative attributes in attribute-based access control model for internet of medical things," in *Proceedings of the 2019 15th International Conference On Semantics*, pp. 167–174, Knowledge and Grids (SKG), Guangzhou, China, September 2019.

[36] M. U. Aftab, Z. G. Qin, N. W. Hundera et al., "Permission-based separation of duty in dynamic role-based access control model," *Symmetry*, vol. 11, 2019.

[37] M. U. Aftab, Z. G. Qin, J. Zakria, S. Ali, Pirah, and J. Khan, "The evaluation and comparative analysis of role based access control and attribute based access control model," in *Proceedings of the 2018 15th International Computer Conference On Wavelet Active Media Technology And Information Processing (ICCWAMTIP)*, pp. 35–39, Chengdu, China, September 2018.

[38] D. Hu, C. Hu, Y. Fan, and X. Wu, "oGBAC—a group based access control framework for information sharing in online social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, pp. 100–116, 2018.

[39] A. Thakare, E. Lee, A. Kumar, V. B. Nikam, and Y.-G. Kim, "PARBAC: priority-attribute-based RBAC model for azure IoT cloud," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2890–2900, 2020.

[40] K. Vijayalakshmi and V. Jayalakshmi, "A priority-based approach for detection of anomalies in ABAC policies using clustering technique," in *Proceedings of the 2020 Fourth International Conference On Computing Methodologies And Communication (ICCMC)*, pp. 897–903, Erode, India, March 2020.

[41] X. Cheng, F. Dai, M. Hu, and Q. Gui, "An improved privacy-preserving and security hybrid access control mechanism," in *Proceedings of the China Conference On Wireless Sensor Networks*, pp. 169–180, Springer, Chongqing, China, October 2019.

[42] "Function compute," 2020, https://www.alibabacloud.com/help/en/product/50980.html.

[43] "Use custom topics for communication, ," 2020.

[44] "Hawkeye track service," 2021, http://lbsyun.baidu.com/index.php?title=yingyan.

[45] J. Yuan, Y. Zheng, X. Xie, and G. Sun, "Driving with knowledge from the physical world," in *Proceedings of the 17th ACM SIGKDD International Conference On Knowledge Discovery And Data Mining*, pp. 316–324, San Diego, CA, USA, August 2011.

[46] J. Yuan, Y. Zheng, C. Zhang et al., "T-drive: Driving directions based on taxi trajectories," in *Proceedings of the 18th SIGSPATIAL International Conference On Advances In Geographic Information Systems*, pp. 99–108, San Jose, CA, USA, November 2010.

[47] Y. Rouselakis and B. Waters, "Efficient statically-secure large-universe multi-authority attribute-based encryption," in *Proceedings of the International Conference On Financial Cryptography And Data Security*, pp. 315–332, Juan, Puerto Rico, November 2015.