EAI Endorsed Transactions
on Scalable Information Systems
Research Article EAI.EU

An efficient access control scheme for smart campus

Yiru Niu[1], Hong Jiang[2], Bo Tian[3], Hong Xiang[3], Yiming Liu[1], Xiaofeng Xia[3] and Yue Zhao[1,*]

[1]Science and Technology on Communication Security Laboratory, Chengdu, China
[2]No.30 Research Institute of China Electronics Technology Group Corporation, Chengdu, China
[3]School of big data and software engineering, Chongqing University, Chongqing, China

Abstract

With the great concern of our country and the continuous development of the epidemic, the development of smart campus is getting faster and faster, the safety of teachers and students becomes more and more important. To ensure the safety of users, the first step is to control at the doors. Usually, the access control method is used in computer system to protect the documents and data, few people use it at doors, but it's a very effective way to improve safety. So we design a two-factor authentication protocol to verity the user's identity, and improve the attribute-based access control (ABAC) model to fit the smart campus. We analyze the protocol theoretically and verify its security. Compare with others, our scheme can be more efficient and safer.

Keywords: identity authentication, access control, radio frequency identification (RFID), policy conflict detection.

Received on 01 February 2022, accepted on 17 March 2022, published on 21 March 2022

doi: 10.4108/eai.21-3-2022.173712

1. Introduction

With the advent of the smart era, the concept of "smart" has been gradually accepted by people and penetrated into all fields of social life. Smart campus is an important attempt and achievement for people to comply with the trend of Education informationization era and explore the in-depth application of information technology in the field of education. The construction of smart campus has also been strongly supported by the state. In 2017, the 13th five year plan for the development of National Education issued by the national development and Reform Commission clearly proposed to "support schools at all levels to build smart campuses and explore new models of education in the future". In 2018, the State Administration of market supervision and administration published *the overall framework of smart campus* [1], which has carried out the overall framework on how to build a smart campus, and the construction of smart campus has a clearer policy basis.

An important aspect of smart campus construction is smart campus management. In recent years, with the emergence of the COVID-19, the control of personnel flow on campus has become more and more strict. How to ensure the safety of personnel in and out is an urgent problem to be solved.

The first step to achieve security control is to correctly identify the user's identity. Common methods of identity authentication include password based authentication, smart card based authentication, biometric based authentication and so on. In order to improve security, the above methods can be combined to realize multi-factor authentication. Challa et al.[2] presented a provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks, but the security problems that may be encountered in the process of card reading are not considered. Li et al.[3] designed a lightweight RFID system information security authentication protocol, which can effectively resist the retransmission, tracking, deception, cloning and synchro-

*Corresponding author. Email: yuezhao@foxmail.com

nization, but when the background database authenticate the tag, it needs to traverse all real ID of card reader and tag and compute its hash value. With the increase of the number of tags and readers, it will consume a lot. In this paper, we propose a multi factor authentication scheme based on face information and RFID information to realize safe and reliable identity authentication.

After the authentication is passed, it is necessary to decide whether the user has the authority to access. The previous permission control methods, such as discretionary access control (DAC), mandatory access control (MAC) and role-based access control (RBAC), have many security problems, such as low security, poor flexibility, coarse management granularity and so on. On the contrary, ABAC can achieve high flexibility and more fine-grained control, and support dynamic authorization and multi-party joint authorization. But at the same time, the problem is that a large number of policies are prone to conflict. Although eXtensible Access Control Markup Language (XACML) has provided a variety of basic algorithms for combining policies and solving policy conflicts, these algorithms are only used to resolve conflicts when conflicts occur, and can not detect possible conflicts in the policy set in advance. Liu et al.[4] thinks that most of the previous conflict detection methods could not detect implicit conflicts. In order to solve this problem, a method to transform implicit conflict into explicit conflict is proposed. However, it does not take into account the multiple complex value cases that attribute fields may have. Wang et al.[5] proposed a conflict detection method based on expression tree, which can detect attribute fields with different values, such as unique value, continuous value, discrete value, fixed interval value, and values related to other fields. At the same time, it can detect implicit conflicts with accurate results. However, it only considers conflicting policies which have opposite decision results, but does not detect redundant situations which have the same decision results. In addition, it needs to traverse all policy trees during detection, which is complicated. Vijayalakshmi et al.[6] analyzed five species policy conflicts in ABAC policy sets and proposed corresponding solutions. However, their method to resolve policy redundancy is delete the policy with small scope, which may lead to further policy conflicts. At present, most of the access control schemes are designed for computer resources, which can not be well applied to the scenario of smart campus. And most current studies fail to take into account the potential conflicts that may occur when multiple administrators jointly formulate policies[7]. Therefore, this paper studies a policy conflict detection method which is more suitable for the security management of smart campus under the condition of multi-administrator joint authorization based on the expression tree method proposed by Wang.

Our main contribution can be summarized in the following two points.

1. We design a two-factor authentication protocol, which contains RFID and facial information. We consider the security of authentication process and the RFID tag information reading process and use Elliptic Curve Cryptography (ECC) to improve the efficiency. So our protocol can be safer and more efficient than others.

2. The previous access control schemes are applied to computer systems, in order to fit the access control system of smart campus, we propose a new policy index structure, which can make the policy conflict detection process and policy search process more efficient.

2. System Model

The system model of smart campus access control system is shown in Figure 1.

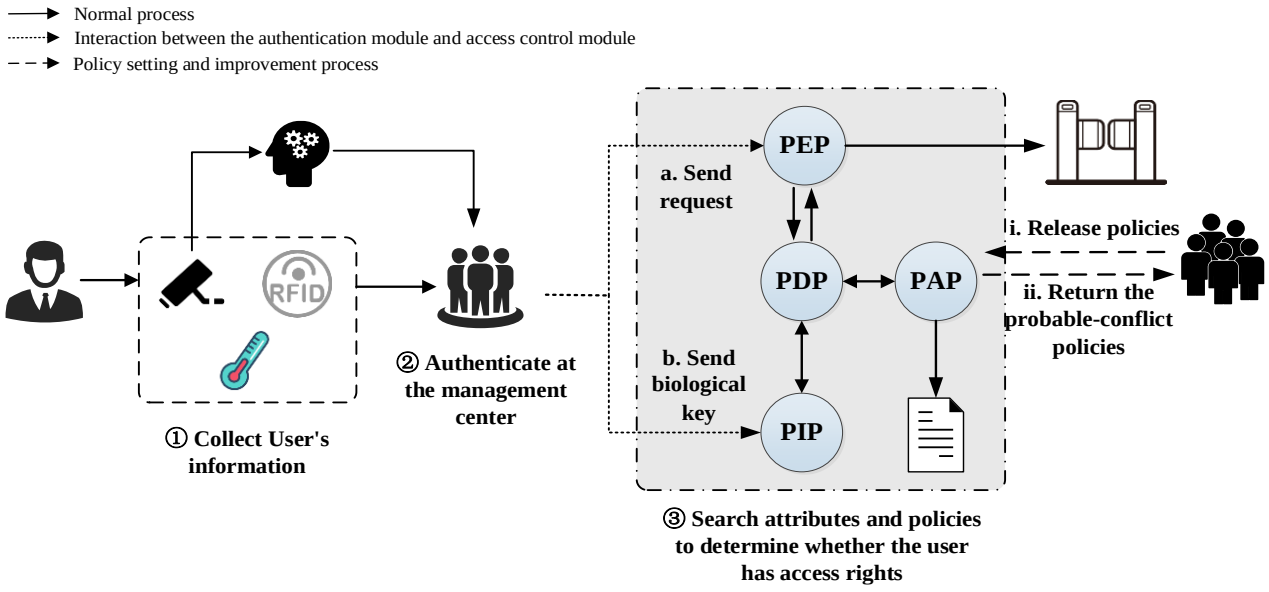

Figure 1. System model

When a user sends an access request, the user information will be collected first, including face information for identification, RFID tag information, and temperature information required for access control under the epidemic situation. The face information collected by the camera is analyzed by the third-party artificial intelligence (AI) and sent to the management center, which integrates the face information and the RFID tag information obtained by the card reader for identity authentication. If the authentication is successful, the management center will send relevant request information to the policy decision module. The policy decision module uses the ABAC model to check the attributes of the subject, object, and environment according to the specific policy and determine whether the subject has access to the object.

The dotted line in the right of the figure indicates that when making policies, multiple administrators can jointly publish policies, detect conflicts on policies, and return possible conflicts to administrators for modification.

3. Identity Authentication

When an access request comes, identity authentication is first required to confirm the user's legal identity. The participants involved in the authentication process are management center (*MC*), third-party AI, user (*U*), RFID tag (*RFID*) and reader (*R*) (management center as RFID background database). The cryptography methods mainly include ECC algorithm, hash function and fuzzy extractor. Authentication involves initialization phase and authentication phase. The notations are listed in Table 1:

Table 1. Notations of protocol

Notation	Description
U_i, R_j, MC	i^{th} user, j^{th} reader, management center
$RFID_i$	U_i's RFID tag
ID_i , ID_{R_j}	U_i's identity, R_j's identity
P , SK , PK	An ECC point in elliptic curve $E_p(a,b)$, MC's private key, MC's public key
SK_{ID_i} , PK_{ID_i}	U_i's private key, U_i's public key
SK_{R_j} , PK_{R_j}	R_j's private key, R_j's public key
$Gen(\cdot)$, $Rep(\cdot)$	Probabilistic generation and deterministic reproduction procedures in fuzzy extractor
$h(\cdot)$	one-way hash function
Bio_i , σ_i , τ_i , T_{u_i}	U_i's personal biometrics template, biometric secret key, public reproduction parameter and valid period

3.1. Initialization phase

MC performs the following steps to initialize parameters and register for users:

Step Ini.1 *MC* chooses an elliptic curve $E_p(a,b)$ on a finite field Z_p , where p is a large prime, the base point P of order n , the private key SK , and the public key $PK = SK.P$.

Step Ini.2 A one-way cryptographic hash function $h(\cdot)$ is selected by *MC*.

Step Ini.3 Fuzzy extractor is used for biometric authentication, which including randomized generation function $Gen(\cdot)$ and deterministic reproduction function $Rep(\cdot)$. So the public parameters are $\{E_p(a,b),\ p,\ P,\ PK,\ h(\cdot),\ Gen(\cdot),\ Rep(\cdot)\}$.

Step Ini.4 User's RFID tag $RFID_i$ will be setup by *MC*. *MC* generate ID_i for user U_i , stored in $RFID_i$ and *MC*. Run $Gen(Bio_i) = (\sigma_i, \tau_i)$ with Bio_i to obtain the biometric key σ_i and the recovery parameter τ_i . (The biometric key will be used at 4.1). Compute $h(\sigma_i)$ to store in $RFID_i$, and τ_i can be used for secret partition. The secret partition between the two participants can be simply performed by XOR method, that is, *MC* selects t_{i1} of the same length as τ_i randomly, compute $t_{i2} = \tau_i \oplus t_{i1}$, t_{i1} and t_{i2} are stored in *MC* and $RFID_i$ respectively. Finally, *MC* set the valid period T_{u_i} for $RFID_i$, compute $t = t_{i2} \oplus T_{u_i}$, saved in *MC*. So in *MC* : $\{ID_i,\ t_{i1},\ t\}$, $RFID_i$: $\{ID_i,\ h(\sigma_i),\ t_{i2}\}$.

3.2. Authentication phase

Authentication needs to integrate two parts of information to carry out, which are face information and RFID tag information.

Step Auth.1 The camera obtains the face image and hands it to the third-party AI for face recognition. The third-party AI will send the user's identity information ID_i and biometric information $Bio_i^{'}$ to the management center for multi-factor authentication together with the information obtained in the RFID tag.

Step Auth.2 When the reader senses the RFID tag, it will send the authentication request to the tag.

Step Auth.3 After receiving the authentication request, $RFID_i$ will sign the timestamp of the current moment and the stored information. Then, $RFID_i$ encrypts user's identity in-

formation and the signed information sig_{RFID} with the public key of *MC* and sends it to reader R_j.

Step Auth.4 After receiving the message from $RFID_i$, *R* will sign it, encrypt the signed information and the identity information of *R* using the public key of *MC*, and send it to *MC*.

Step Auth.5 After receiving the information, *MC* first decrypts it to get the real identity of *R* and the message signed by *R*, verifies the signature to obtain the encrypted information sent by the $RFID_i$. After decryption, the identity of user and the information signed by $RFID_i$ will be obtained. Verify the signature of $RFID_i$ to get the timestamp and the information used to authenticate the user. If the timestamp is reasonable, the message will be considered valid, or else the authentication will fail.

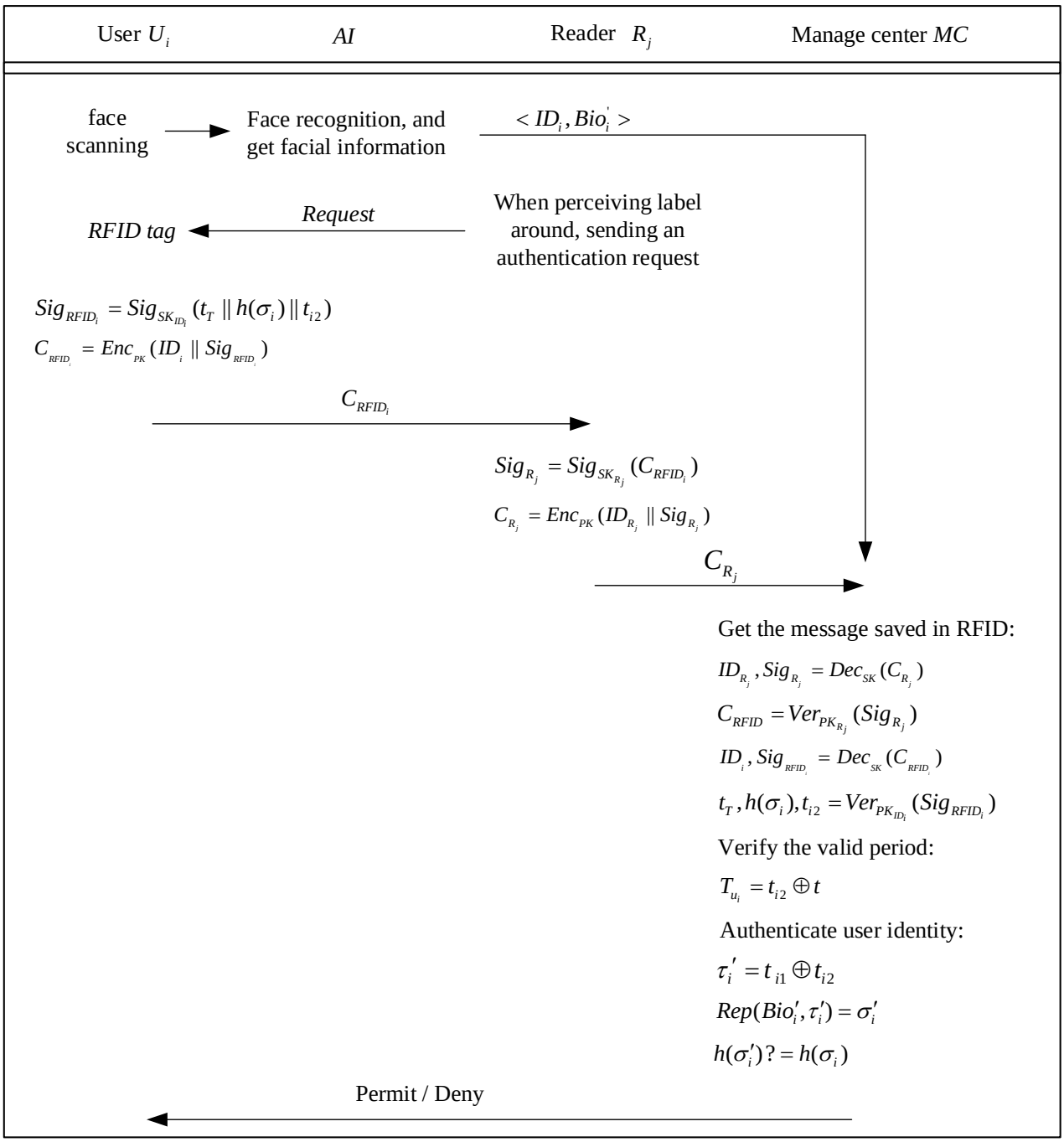

Figure 2. Authentication phase

Step Auth.6 Both the user's information stored in *MC* and the message received from $RFID_i$ are used to recover the valid period T_{u_i} of tag and the recovery parameters τ_i which is needed to run the $Rep(\cdot)$ function. The result of this function is the biometric key σ_i', if $h(\sigma_i') = h(\sigma_i)$, the user will pass the authentication.

4. Access Control for Smart Campus

The attribute-based access control model is relatively mature. In this paper, we mainly propose two improvements for the smart campus scenario, which are the storage of subject attributes and the management and conflict detection of policies.

4.1. The storage of subject attributes

In order to improve the security of the subject attribute and prevent it from being tampered at will, the attribute certificate (AC) method is used in storage. The application, issuance, maintenance and revocation of the attribute certificate is run by the attribute authority (in this scheme, we still use the management center), which manages the entire life cycle of attribute certificates. The subject attributes are collected by the management center, stored in the attribute certificate, which is symmetrically encrypted with the user's biometric key generated during the authentication process. The encrypted certificate is stored in the Policy Information Point (PIP) of the ABAC model and can be retrieved by user ID. At the same time, when PIP wants to decrypt the attribute certificate, in order to securely transmit the biometric key from the MC to the PIP, the message will be encrypted by the ECC algorithm.

The format of the attribute certificate is defined by the X.509v4, as shown in Figure 3. The version number, owner, issuer, signature, serial number, validity period, attribute and signature algorithm are the basic information that constitutes the attribute certificate. Using the signature algorithm specified in the certificate, the first nine items in the certificate are taken as input to obtain the signature value, which is appended to the attribute certificate.

4.2. Policy Management and Conflict Detection

XACML adopts a nested structure when managing policies. The outermost layer is the policy set, which mainly includes target, policies and policy combination algorithms; policies include target, rules and rule combination algo-rithms; rules include target, utility and conditions (conditions are not considered in this paper). In the above elements, target represents the object to which the rules apply, including four basic elements: subject (S), object (O), operation (op) and environment (E). The utility represents the decision (d) of the rule, includes two values: permit and deny. Two common conflicts in policy sets are redundancy and discrepancy, the definitions are given as follows and the examples can be seen at Conflict Detection.

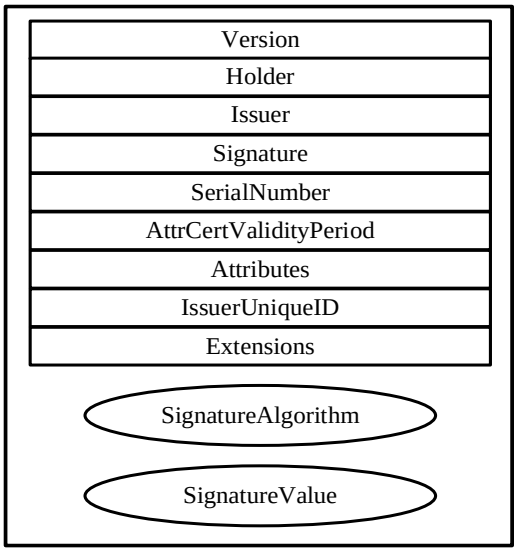

Figure 3. The format of the attribute certificate.

Definition 4.1 The rules-redundancy will occur when:
(1) {S, O, E} of R1 is a subset of R2 or {S, O, E} of R2 is a subset of R1:
{S(R1) ⊆ S(R2), O(R1) ⊆ S(R2), S(R1) ⊆ S(R2)} or
{S(R2) ⊆ S(R1), O(R2) ⊆ S(R1), S(R2) ⊆ S(R1)}
(2) The operations of R1 and R2 are the same:
op(R1) = op(R2)
(3) The decisions of two rules R1 and R2 are the same:
d(R1) = d(R2)

Definition 4.2 The rules-discrepancy will occur when:
(1) {S, O, E} of R1 is a subset of R2 or {S, O, E} of R2 is a subset of R1:
{S(R1) ⊆ S(R2), O(R1) ⊆ S(R2), S(R1) ⊆ S(R2)} or
{S(R2) ⊆ S(R1), O(R2) ⊆ S(R1), S(R2) ⊆ S(R1)}
(2) The operations of R1 and R2 are the same:
op(R1) = op(R2)
(3) The decisions of two rules R1 and R2 are different:
d(R1) ≠ d(R2)

Policy Management

In order to make the policy management of XACML language more suitable for the access control system of smart campus, the index structure is established according to the object attribute when storing the policy. The object attribute has a hierarchical structure (except the campus gate), and there is a corresponding policy set at each level, which is shown in Figure 4. In policy conflict detection and resolution, only the policies in the policy set with the same object attribute need to be detected. When a request for policy retrieval occurs, the search can also be performed according to the object attribute, which improves the efficiency.

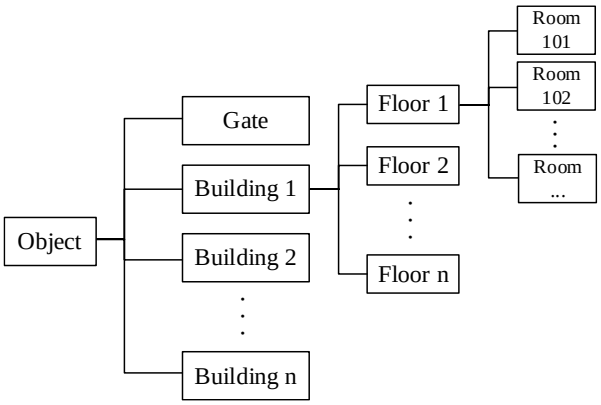

Figure 4. The index structure of object attrib utes.

Conflict Detection

In the conflict detection, the expression tree method mentioned in Wang's paper is used. Construct expression trees for each policy corresponding to the same object attribute, each attribute expression is a leaf node, two leaf nodes and an AND or OR relation constitute a minimum binary tree, and an expression tree contains several minimum binary trees. When comparing expression trees, a recursive method is used: first compare the leaf nodes, then compare the leaf nodes with the minimum binary tree, then compare between the minimum binary trees, and finally complete the comparison of the entire expression tree. When comparing leaf nodes, attribute fields with different values can be detected, such as unique value, discrete value, continuous value, fixed value range, and values related to other attribute fields (hierarchical relationship). The final return result is whether there is an intersection between the two expression trees, and if there is an intersection, the intersection type of the intersection attribute expression is returned. For more details, please refer to [5].

In the detection process of the above paper, the author only considers the rules-discrepancy, like R1 and R2 shown in Table 2, but the case of rules-redundancy is not studied, like R1 and R3. But different policies may be formulated by different administrators, when there is rules-redundancy as shown in Table 2, the administrator who formulated R1 thinks that only teachers can enter Room 302 from 7:00 to 23:00, but students are not mentioned, default Deny; and the administrator who formulated R3 thinks that both students and teachers can enter Room 302 from 7:00 to 23:00. In this case, there is a conflict as to whether students can enter, but because of the decision of two rules are both Permit, they will not be detected using the method mentioned in the above paper. Therefore, the redundant rules should also be detected and returned to the administrator for modification[8]. On the basis of Wang's method, we can modify the policy set to be detected into all policy sets corresponding to the same object, and add the conflict type to the return results. In this way, both rules-redundancy and rules-discrepancy can be detected.

Table 2. Sample rules

Rule	Subject	Object	Environment	Opration	Decision
R1	Role = teacher	Room = 302	Time = 7:00-23:00	Access	Permit
R2	Role = teacher	Room = 302	Time = 7:00-23:00	Access	Deny
R3	Role = teacher, student	Room = 302	Time = 7:00-23:00	Access	Permit

In the hierarchical structure of XACML language, policy combination algorithm and rule combination algorithm are effective methods for conflict resolution. XACML provides combination algorithms such as Deny-overrides, Permit-overrides, First-applicable and Only-one-applicable. In specific applications, in order to achieve more secure management and control, the Deny-overrides algorithm is selected as the main combination algorithm. At the same time, considering that some subjects may be granted the privilege to access an object, a set of policies for privileged users are added in addition to all policies, the combination with the original policy set uses the First-applicable algorithm. The specific use cases of the combination algorithm are as Figure 5.

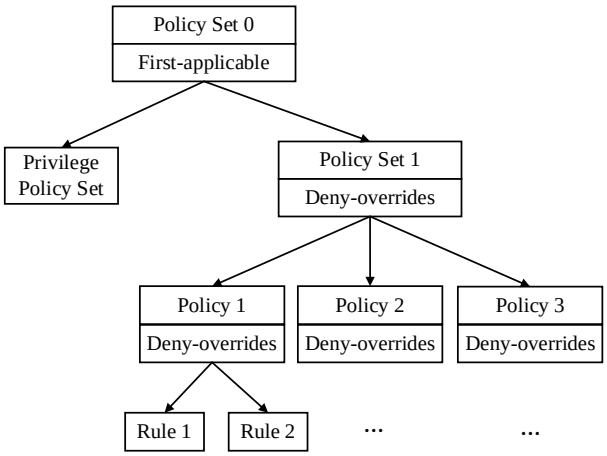

Figure 5. Specific use cases of the combination algorithm.

5. Security analysis

In order to verify the security of the authentication protocol proposed in this paper, several common attack types are analyzed and explained below.

Card theft attack: The information stored in the card is $< ID_i, h(\sigma_i), t_{i2} >$. If the card is stolen, the attacker can only get the valid information of the user's real ID. The biological key cannot be recovered through the hash value

of the biological key, and the complete recovery parameters cannot be obtained only through the value of t_{i2}. The biological key cannot be recovered through the hash value of the biological key, and the complete recovery parameters cannot be obtained only through the value of t_{i2}.

Insider attack: The information stored in the management center is $< ID_i, t_{i1}, t >$.If there is an internal privileged person who wants to modify the valid period of user, it cannot be achieved only by t, and RFID tags are required to complete it together; if they want to obtain recovery parameters, RFID tags are also required. Therefore, it is considered that the insider attack can be deducted.

Man-in-the-Middle Attack: The messages sent between RFID tags, readers, and the management center are ciphertext except for the random number, and can be decrypted only by using the manager's private key. Assume that the messages are eavesdropped by a middleman, and there is also no way to get the information passed. At the same time, because the random number and timestamp are updated during each session, replay attacks cannot be used to pretend to be legitimate users.

User anonymity: During the whole process of acquiring the RFID tag content from the background database, no message directly contains the user's ID information, so it can be considered that the user is anonymous to the card reader.

Tracing attack: The information sent by the tag contains only one ciphertext message, no identity-related content, and changes with each session. Therefore, the tag is considered to be resistant to tracing attack.

We also compare the functionality and security features of our protocol with other related schemes[9-12] in Table 3. In this table, we can see our protocol can resist more attacks.

Table 3. Security features comparison

Feature	[9]	[10]	[11]	[12]	Proposed
Card theft attack	√	×	×	√	√
Insider attack	×	×	√	×	√
Man-in-the-Middle Attack	√	√	√	×	√
User anonymity	×	×	√	√	√
Tracing attack	×	×	√	√	√

Because there are few people do research on how to use the access control model in door lock, we don't compare the conflict detection part. But we will do some experiment to show the efficiency of our scheme in the future.

6. Conclusion

Nowadays, the epidemic is still not completely over and security and privacy issues are getting more and more attention, how to do a better job in campus control is an important issue concerning the safety of all teachers, students and other staffs. This paper proposes a control scheme for the access control of smart campus, which organically combines multi-factor authentication and attribute-based access control, not only ensuring security, but also more suitable for the scenario of smart campus. In the future research, the log records generated in the process of access control can be stored in Merkle trees to achieve traceability of personnel flow.

References

[1] State Administration for Market Regulation standardization Committee of China. "Overall framework of smart campus". Beijing: Standards press of china, 2018: 1-2.

[2] Sravani Challa, Ashok Kumar Das, Vanga Odelu, Neeraj Kumar et al., "An efficient ECC-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks". Computers & Electrical Engineering. 2018; 69: 534-554.

[3] Rongrong Li, Jiantao Kou, Gang Dong, Liangrui Tang. "A security authentication scheme in RFID system for smart park". Telecom Science. 2016; 2:164-169.

[4] Gang Liu, Wenxian Pei, Yumin Tian, Chen Liu et al., "A novel conflict detection method for ABAC security policies". Journal of Industrial Information Integration. 2021; 22.

[5] X. Wang et al., "Expression Tree-based Policy Conflict Detection Algorithm". 2021 International Conference on Networking and Network Applications (NaNA), 2021, p. 361-366.

[6] K. Vijayalakshmi et al., "Identifying Considerable Anomalies and Conflicts in ABAC Security Policies". 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), 2021, p. 1273-1280.

[7] Y. Zhao et al., "A Security Access Control Scheme of Smart Community under Epidemic Situation". 2021 6th International Conference on Communication, Image and Signal Processing (CCISP), 2021. p. 85-89.

[8] Zetao Jiang, Zhen Xie, Qi Wang, Wenhui Zhang. "ABAC Static Policy Conflict and Redundancy Detection Algorithm Based on Mask Key". Computer Science. 2018; 2:197-202.

[9] M. Wazid, A. K. Das, V. Odelu, N. Kumar, M. Conti, M. Jo. "Design of secure user authenticated key management protocol for generic IoT networks". IEEE Internet of Things Journal. 2018; 5: 269-282.

[10] Y. Choi, D. Lee, S. Kim, J. Jung, J. Nam, and D. Won, "Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography". Sensors. 2014; 14: 10081-10106.

[11] Chen C.M., Huang Y., Wang K.H., Kumari S., Wu M.E. "A secure authenticated and key exchange scheme for fog computing". Enterprise Information Systems. 2020; 4: 1-16.

[12] Akram M.A., Ghaffar Z., Mahmood K., Kumari S., Agarwal K., Chen C.M. "An anonymous authenticated key-agreement scheme for multi-server infrastructure". Human-Centric Computing and Information Sciences. 2020; 10: 1-18.