RBAC-based Delegation Authorization with Trust Computing and Collaborative Security Strategy

Wei Sun

(Corresponding author: Wei Sun)

School of Computer and Information Technology, Xinyang Normal University
No. 237, Nanhu Road, Xinyang 464000, P. R. China
Email: sunny810715@xynu.edu.cn

(Received Feb. 4, 2023; Revised and Accepted June 4, 2023; First Online June 25, 2023)

Abstract

Role-based access control (RBAC) is very popular as it has various security-control strategies and provides a flexible authorization mechanism. Delegation authorization based on RBAC is an effective method that decentralizes the authorization management from the delegating subject to the delegated object. However, delegations are arbitrary using the existing approaches, particularly in large-scale distributed and collaborative systems. The delegation process becomes unreliable once the delegated object with lower trustworthiness is selected, and the security of the system status cannot be guaranteed. Aiming at these issues, this study proposes a novel delegation method based on the trust relationship and the collaborative security strategy and then presents its delegation-authorization process. First, to reflect the reliability of the delegation process, the trust degrees of different candidate objects are comprehensively calculated using the trust seniority, trust experience, and trust recommendation, and the delegated objects with higher values are selected. Second, to ensure system security, the collaborative division of duties is introduced, which can be implicitly enforced by the mutually exclusive-role constraints to determine further the most appropriate delegated object in the secure collaborative environment. The experimental results show that, compared to the existing studies, the proposed method effectively improves the reliability of the delegation process and satisfies the various security requirements of organizations.

Keywords: Collaborative Security Strategy; Mutually Exclusive Constraint; Role-based Access Control; Trust Relationship

1 Introduction

The role-based access control (RBAC), owing to the characteristic of the convenience for authorizations as well as its various security policies, had emerged as a very popular mechanism and had been widely adopted by the organizations of all scales over the past few decades [6, 8, 21]. As an important manifestation of the RBAC system, the delegation-authorization technique based on RBAC chooses an appropriate delegated object as the substitute for the delegating subject or user, in order to decentralize the centralized authorization management to some ordinary object. The delegation authorization has been proved to be flexible and useful in many practical applications [3, 4, 16]. With the high-speed advance and wide application of the emerging network-information technologies, including the Internet of Things (IoT), edge computing and blockchain, the security and reliability of the information system are regarded as two increasing prominent problems, which have been attracting much attention in both academia and industry in recent years.

In the computer-supported distributed and collaborative working systems, multiple users are required to cooperate and communicate with each other, in order to jointly make the access decision to information resources, while achieving the purpose of the mutual restrictions [1, 22, 24]. As an effective access-control strategy, the trust-collaboration mechanism meets the confidentiality and privacy requirements of information resources, and it reduces the possibility of abusing authorities due to the randomness of the delegation. Therefore, when some user is temporarily absent or on leave, how to develop a reliable and secure delegation scheme for choosing a trustworthy substitute to participate in the collaboration is very challenging.

In light of the above-mentioned problems, this study proposes a novel method, called RBAC-based delegation

authorization with the trust computing and collaborative security strategy (RDA_TC&CSS), and then presents its process in detail. The main contributions of this work are as follows:

1) To improve the reliability of the delegation process, we employ the trust relationship, including the trust seniority, trust experience and trust recommendation to comprehensively compute the trust degrees of different candidate objects, and then choose the objects with higher values. We demonstrate the effectiveness of the RDA_TC&CSS using a specific simulated system.

2) To ensure the system security, while avoiding the abuse of authorities due to the randomness of delegations, we utilize the collaborative security strategy to further determine the most appropriate delegated object in the specific collaborative scenario, and then indirectly implement the strategy by constructing the mutually-exclusive-role constraints. We demonstrate the efficiency of the RDA_TC&CSS using real-world datasets.

The rest of the article is structured as follows. We discuss some related work in Section 2. Section 3 introduces the necessary preliminaries used for our work. Section 4 proposes a novel delegation method and presents its delegation-authorization process. We present the experimental analysis in Section 5 and conclude the article and discuss future work in Section 6.

2 Related Work

Aiming at the problems of the heavy management burden and lack of the flexibility for the centralized authorization, Crampton *et al.* [7] proposed the RBAC-based delegation to flexibly transfer the authorization management to ordinary users, which alleviates the burden of the system management and has some reference for studying the delegation in the collaborative environment. Alsulaiman *et al.* [5] proposed a threshold-based collaborative access control model, called TBCACM, which associates the access privileges of the resources with the threshold, transfers permissions with the contribution coefficients to the delegated users to participate in collaboration and makes common decision for accessing resources. Yu *et al.* [23] proposed a locale-based access control model (LCACM) in the collaborative environment and presented a hierarchical authorization mechanism within the collaborative group, which could meet the security and flexibility requirements of the delegation. Khan *et al.* [9] proposed a security delegation model by restricting the delegation of sensitive permission as well as restriction from unauthorized access to resources, which could reduce the administrative burden and enable the automated delegation. Actually, the delegating subject should select competent and trustworthy delegated objects from many candidate objects with different degrees of trust and execution. However, there exists the problem of arbitrariness in most of the existing approaches. The delegation process becomes unreliable once the delegated object with lower trustworthiness is selected.

To realize the fine-grained, accurate and reliable delegation authorization, Liu *et al.* [13] combined conventional RBAC model and trust management and proposed a trust-based access control model (TBACM), which comprehensively calculated the trust degrees of different users and evaluated their behavior ability. According to the dynamic changes of user behavior, Zhu [25] divided the user trust into static trust level and dynamic trust level and proposed a user trust-based dynamic multi-level access control model (UTDMACM), which could realize the hierarchical access control and fine-grained dynamic authorization. According to the dynamic variability of the network environment and user status, Liu and Chang [12] introduced the concept of trust measure and context constraint and proposed a novel access control model based on the multi-dimension measurement and context, called MMCBACM, which could realize the dynamic and real-time control to the delegation authorization. To mitigate the malicious actions caused by the authenticated users, Abdul *et al.* [2] designed an access control mechanism by computing the trust degree based on the user's uncertain behavior, which could accurately detect and mitigate malicious users from the mobile cloud computing environment. These models or mechanisms can better reflect the reliability of the delegation process. However, they cannot meet the security requirements for a given collaborative working environment, and there may be potential security risks.

A key characteristic of RBAC is that it allows the specification and enforcement of various constraint policies [10, 15], such as the cardinality constraint and separation-of-duty constraint (SOD). As a significant constraint strategy discussed in this study, the SOD states that at least k users are required to complete a special task that requires n permissions, while any $(k-1)$ users are not allowed to together have all these permissions needed to complete the task [17]. It has been regarded as a critical principle of the information-system security and is widely used in large-scale distributed and collaborative systems. Sarana *et al.* [18] proposed a novel role-optimization method represented as RMP_SoD when the SOD constraints were present and developed three alternative approaches either during or after the role mining. In order to satisfy SOD constraints while ensuring authorization security, Sun *et al.* [20] proposed a role-mining method, called role-mining optimization with separation-of-duty constraints and security detection for authorizations (RMO_SODSDA). Subsequently, Sun *et al.* [19] proposed another novel policy-engineering method, called policy-engineering optimization with visual representation and separation of duty constraints (PEO_VR&SOD), which utilized the method of SAT-based model counting to reduce the constraints and constructed mutually exclusive constraints, in order to implicitly enforce the given

SOD constraints. In the above-mentioned approaches for constructing RBAC systems with the SOD constraints, however, an inadequately addressed key challenge is that occurrence of the delegation cases is not taken into consideration.

3 Preliminaries

Before proposing our methodology, some preliminaries are presented, including the trust relationship and collaborative security strategy.

3.1 Trust Relationship

The trust relationship between the delegating subject and the delegated object, referring to the literature [2], lies on three factors: the trust seniority, trust experience and trust recommendation. Specifically, the trust seniority indicates the inherent properties of the delegated object, including the basic seniority of the object itself and the affiliated seniority of the associated roles; the trust experience indicates the effect of the candidate object having completed the task on behalf of the delegator in the past; The trust recommendation indicates the trust evaluation of the third party for the candidate object. For convenience, both the subjects and objects represent persons in the following.

1) Task attribute ta_i: Multiple users $u_1, u_2, \cdots$ must have a series of qualification conditions $a_{i_1}, a_{i_2}, \cdots, a_{i_n}$ required by task t_i, before they are cooperating to perform t_i. The combinational set of $a_{i_1}, a_{i_2}, \cdots, a_{i_n}$ is regarded as the task attribute of t_i, denoted as $ta_i = \{a_{i_1}, a_{i_2}, \cdots, a_{i_n}\}$.

2) User attribute ua_j: Collaborative user u_j owns a series of qualification conditions $a_{j_1}, a_{j_2}, \cdots, a_{j_m}$, which are regarded as the user attribute, denoted as $ua_j = \{a_{j_1}, a_{j_2}, \cdots, a_{j_m}\}$.

3) Basic seniority $A_{j\to i}$: It is used to represent the basic seniority of user u_j with respect to task t_i. Assuming the weight values $wa_{i_1}, wa_{i_2}, \cdots, wa_{i_n}$ respectively represent the importance of $a_{i_1}, a_{i_2}, \cdots, a_{i_n}$ towards task t_i, where $0 \le wa_{i_r} \le 1$, and $\sum_{r=1}^{n} wa_{i_r} = 1$. Then $A_{j\to i} = \sum_{k=1}^{p} wa_{i_k}$, where $p \le |ua_j \cap ta_i—$, wa_{i_k} is the weight associated to a_{i_k}, and $a_{i_k} \in ua_j \cap ta_i$.

4) Affiliated seniority $RA_{j\to i}$: User u_d needs to delegate his authorities to another user u_j through role r_d, when u_d collaboratively participates in performing task t_i. If $dist(r_d, r_j)$ indicates the distance between r_d and r_j on the same role hierarchy RH, where $0 \le dist(r_d, r_j) \le 1$, then $dist(r_d, r_j)$ is regarded as the affiliated seniority of u_j with respect to t_i, denoted as $RA_{j\to i} = dist(r_d, r_j)$. Obviously, the greater the value of $RA_{j\to i}$, the closer connection between u_j and u_d is. If r_d and r_j are not on the same role hierarchy, then u_j and u_d have no connection, that is $RA_{j\to i} = 0$.

5) Trust seniority $P_{j\to i}$: The basic seniority $A_{j\to i}$ and the affiliated one $RA_{j\to i}$ together constitute the trust seniority, denoted as $P_{j\to i} = w_a \times A_{j\to i} + w_{ra} \times RA_{j\to i}$, where w_a, and w_{ra} respectively represent the contributions of $A_{j\to i}$ and $RA_{j\to i}$ for $P_{j\to i}$, $0 \le w_a, w_{ra} \le 1$, and $w_a + w_{ra} = 1$.

6) Trust experience $E_{j\to i}$: As a substitute for the delegating user u_d, the delegated user u_j has participated in the process of performing task t_i a few times in the past. The total effect of u_j completing the task is regarded as the trust experience, denoted as $E_{j\to i} = \sum_(k = 1)^n wt_{j_k} \times e_{j_k}$, where $wt_{j_k} = k/n$ $(1 \le k \le n)$ is used to represent the empirical coefficient of u_j in the k-th execution of t_i, and the more recent the execution time, the greater value of wt_{j_k} is; e_{j_k} is used to represent the effect of the k-th completing the task, $0 \le e_{j_k} \le 1$, which can be stated as follows:

$$e_{j_k} = \begin{cases} = 1 & \text{successful execution} \\ = 0 & \text{false execution} \\ \longrightarrow 1 & \text{performing well} \\ \longrightarrow 0 & \text{performing poor} \end{cases} \quad (1)$$

7) Trust recommendation $R_{j\to i}$: When u_d needs to delegate his authorities to u_j through the joint recommendation of m users $ur_1, ur_2, \cdots, ur_m$, the trust recommendation of u_j with respect to the task is denoted as: $R_{j\to i} = \frac{\sum_{k=1}^{m} t_k \times r_{k_j}}{\sum_{k=1}^{m} t_k}$, where $t_k (1 \le k \le m)$ is used to represent the trust value of the system to ur_k, rk_j represents the recommendation value of ur_k to u_j, and $0 \le t_k, r_{k_j} \le 1$.

3.2 Collaborative Security Strategy

The collaborative security strategy, including the collaborative division of duties and the mutually-exclusive-role constraints, can ensure the security and satisfiability of the system status. Similar to the well-known SOD principle, the collaborative division of duties is a common security strategy in the multi-user cooperative scenarios, which can be effectively enforced by the mechanism of the mutually-exclusive-role constraint [20]. For the sake of brevity, we assume that any role associates with only one permission.

1) System status γ: It is formalized as a triple $< UA, PA, RH >$, denoted as γ, where UA, PA and RH are the basic components of the RBAC model. $Roles_\gamma(u)$, and $Perms_\gamma(u)$ represent the roles and the permissions assigned to u under γ, respectively,

which are formalized as follows:

$$\begin{aligned} Roles_\gamma(u) &= \{r \in R | \exists r_1 \in R, \\ &\quad (u, r_1) \in UA \wedge (r_1, r) \in RH\}; \\ Perms_\gamma(u) &= \{p \in P | \exists r_2 \in R, \\ &\quad (p, r_2) \in PA \wedge (Roles_\gamma(u), r_2) \in RH\}. \end{aligned}$$

2) Collaborative division of duties $k-n$ CDOD: It states that at least k users are required to cooperate with each other, in order to together execute a specific task involving n roles and have all these roles, which can be expressed as $e = cdod < \{r_1, r_2, \cdots, r_n\}, k >$, where $1 < k \leq n$.

3) Static mutually-exclusive-role constraint t-m SMER: It states that any user cannot have t or more roles out of the given m roles, which can be expressed as $c = smer < \{r_1, r_2, \cdots, r_m\}, t >$, where $1 < t \leq m$. When $t = m$, it is also denoted as the $t - t$ SMER, which is more restricted than the $t - m$ SMER.

4) Security of the system status $sec(\gamma)$: Given a $k - n$ CDOD $e = cdod < \{r_1, r_2, \cdots, r_n\}, k >$ under the system status γ, if any $(k-1)$ users cannot have all these n roles under γ, then γ is secure, denoted as $sec_e(\gamma) = 1$; otherwise, γ is not secure, denoted as $sec_e(\gamma) = 0$. Let the set of variants of $k-n$ CDOD be $E = \{e_1, e_2, \cdots\}$, if γ is secure with respect to each e_i, then γ is secure with respect to E; otherwise, γ is not secure with respect to E. Whether or not the system status is secure can be formalized as follows:

$$\begin{aligned} \forall e \in E, \exists \{u_1, u_2, \cdots, u_{k-1}\} \subset U : \\ \{r_1, r_2, \cdots, r_n\} \nsubseteq \bigcup_{i=1}^{k-1} Roles_\gamma(u_i) \\ \Rightarrow sec_e(\gamma) = 1; \\ \exists e \in E, \exists \{u_1, u_2, \cdots, u_{k-1}\} \subset U : \\ \bigcup_{i=1}^{k-1} Roles_\gamma(u_i) \supseteq \{r_1, r_2, \cdots, r_n\} \\ \Rightarrow sec_e(\gamma) = 0. \end{aligned}$$

5) Satisfiability of the system status $sat(\gamma)$: Given a $t - m$ SMER $c = smer < \{r_1, r_2, \cdots, r_m\}, t >$ under the system status γ, if no user is allowed to have t or more roles out of all these m roles under γ, then γ is satisfied, denoted as $satc(\gamma) = 1$; otherwise, γ is not satisfied, denoted as $satc(\gamma) = 0$. Let the set of variants of $t - m$ SMER be $C = \{c_1, c_2, \cdots\}$, if γ is satisfied with respect to each ci, then γ is satisfied with respect to C; otherwise, γ is not satisfied with respect to C. Whether or not the system status is satisfied can be formalized as follows:

$$\begin{aligned} \forall c \in C, \exists u \in U : \\ |Roles_\gamma(u) \cap \{r_1, r_2, \cdots, r_m\}| < t \\ \Rightarrow sat_c(\gamma) = 1; \\ \exists c \in C, \exists u \in U : \\ |Roles_\gamma(u) \cap \{r_1, r_2, \cdots, r_m\}| \geq t \\ \Rightarrow sat_c(\gamma) = 0. \end{aligned}$$

4 Methodology

In this section, based on the trust relationship and collaborative security strategy, the framework of the proposed RDA_TC&CSS is presented as shown in Figure 1. It involves two main aspects: The delegation-authorization flows, and the connected components for the strategy deployment. First, definitions for the calculation of trust degree and the implicit enforcement of security strategy are presented as follows.

Definition 1. *Trust degree $T_{j\to i}$. The trust seniority $P_{j\to i}$, experience $E_{j\to i}$ and recommendation $R_{j\to i}$ together constitute the trust degree of u_j with respect to t_i, which can be calculated and expressed as $T_{j\to i} = w_p \times P_{j\to i} + w_e \times E_{j\to i} + w_r \times R_{j\to i}$, where w_p, we and w_r respectively represent the contribution coefficients of $P_{j\to i}$, $E_{j\to i}$ and $R_{j\to i}$ for $T_{j\to i}$, $0 \leq w_p, w_e, w_r \leq 1$, and $w_p + w_e + w_r = 1$.*

For the specific trust threshold $H_{j\to i}$ in the system, if $T_{j\to i} \geq H_{j\to i}$, then the candidate u_j is trustworthy; otherwise, u_j is not trustworthy.

According to Items (4) and (5) in Subsection 3.2, given the $k - n$ CDOD set E and $t - m$ SMER set C under the system status γ, the process of verifying whether or not γ is secure is in P; similarly, the process of verifying whether or not γ is satisfied is also in P. These two statements have been discussed in other related research, and then the enforcement condition of the secure strategy is presented, in order to determine the relationship between the $k - n$ CDOD and $t - m$ SMER.

Definition 2. *Implicit enforcement of the. $k-n$ CDOD. For the given collaborative strategy e and a constraint set C under γ, e can be implicitly enforced by C, if and only if $\forall c \in C : sat_c(\gamma) \Rightarrow sec_e(\gamma)$.*

4.1 Basic Components for the Strategy Deployment

The deployment for the RDA_TC&CSS mainly consists of the following four connected components:

1) Policy information point (PIP) is responsible for collecting and classifying the information records such as the user attributes, role attributes, historical interactions and user evaluations stored in database DB2, and then it effectively computes the trust relationship.

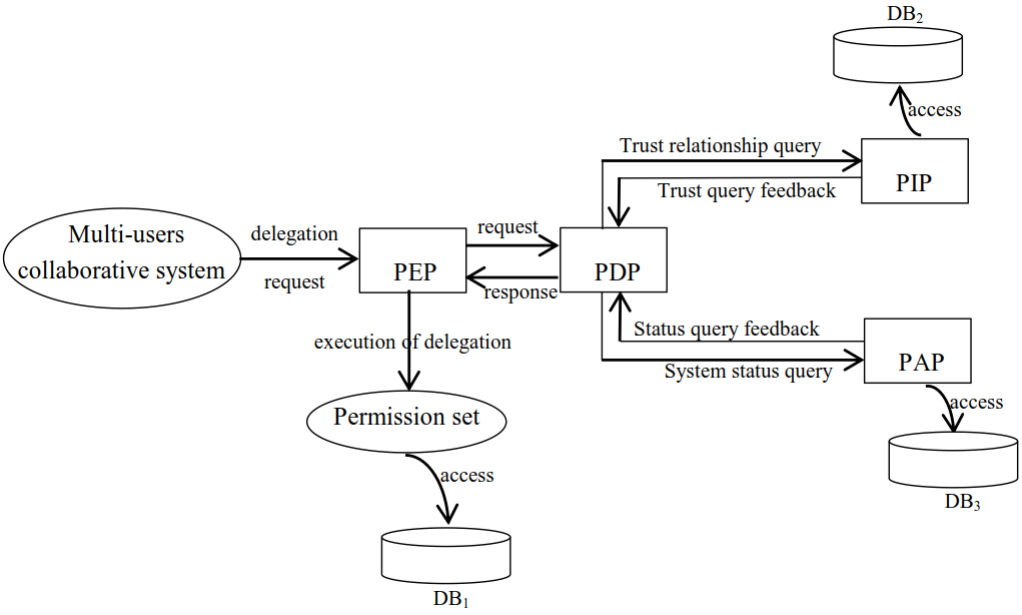

Figure 1: Framework of the RDA_TC&CSS

2) Policy administration point (PAP) is responsible for collecting and classifying the information records such as the collaborative division of duties and the mutually exclusive constraints stored in database DB3, and then it effectively analyzes the security and satisfiability requirements of the system status.

3) Policy decision point (PDP) is responsible for sending the query requests of the trust relationship and system status to PIP and PAP, respectively. It also makes the decision judgment and feeds it back to PEP according to the results of the trust computing and status analysis.

4) Policy enforcement point (PEP) is responsible for receiving the delegation request instructions from users. It also performs the delegation operation according to the decision feedback by the PDP and transfers the corresponding authorities of the delegating users stored in database DB1 to other users.

4.2 Process of the Delegation Authorization

To reflect the reliability of the delegation process and ensure the system security, the detailed procedure of the delegation authorization is presented as follows:

Step 1. In the working environment with collaborative execution of task t_i, user ud sends a delegation request instruction to PEP.

Step 2. PEP sends the request message to PDP and waits for PDP to make a decision response.

Step 3. PDP sends the query request of trust relationship to PIP, and PIP comprehensively calculates the trust degrees of candidate users $\{u_1, u_2, \cdots\}$ according to the records of user attributes, role attributes, historical interactions and user evaluations.

Step 4. Comparing the trust degree $T_{j\to i}$ of the candidate object u_j with the threshold $T_{j\to i}$ set in advance. If $T_{j\to i} \geq H_{j\to i}$, then u_j is inserted into the trustworthy object set and the procedure turns to Step 7; otherwise, u_j is moved from $\{u_1, u_2, \cdots\}$ and the procedure turns to Step 3.

Step 5. PDP sends the query request of system status to PAP, and PAP analyzes and presents the security and satisfiability requirements of the system status, according to records of the collaborative division of duties $\{e_1, e_2, \cdots\}$ and mutually exclusive constraints $\{c_1, c_2, \cdots\}$.

Step 6. Determining whether any e_l in $\{e_1, e_2, \cdots\}$ can be implicitly enforced by some subset of $\{c_1, c_2, \cdots\}$. If c_k can enforce e_l and it belongs to set C by constraint-construction method, then c_k is regarded as the minimal constraint; otherwise, c_k is moved from $\{c_1, c_2, \cdots\}$ and the procedure turns to Step 5.

Step 7. The result of the trust query in Step 4 and that of the status query in Step 6 are fed back to PDP, respectively.

Step 8. The trustworthy candidate objects are comprehensively investigated by PDP, in order to choose the most appropriate as the delegated user, such as u_j. Further, if u_j meets all the constraint requirements, then the decision result "allow delegation" is fed back to PEP; otherwise, u_j is moved from the candidates and other objects are reviewed.

Step 9. PEP executes the delegation-authorization operation on u_j.

4.3 Construction of the SMER Constraints

To implicitly enforce $k-n$ CDOD, we present an approach for constructing $t-m$ SMER constraints from the $k-n$ CDOD as shown in Algorithm 1. It is observed from Lines 2–5 of the algorithm that, the following statement is determined first.

Algorithm 1 Construction of $t-m$ SMER constraints

1: **Input:** $k-n$ CDOD constraint $cdod =< \{r_1, r_2, \cdots, r_n\}, k >$, where $1 < k \leq n$
2: **Output:** set C of $t-m$ SMER constraints
3: Initialize $C = \phi$;
4: **if** $k = 2$ **then**
5: $C = \{< \{r_1, r_2, \cdots, r_n\}, n >\}$;
6: **else if** $k = n$ **then**
7: $C = \{< \{r_1, r_2, \cdots, r_n\}, 2 >\}$;
8: **else**
9: **for** $(t = 2; \lfloor \frac{n-1}{k-1} \rfloor + 1; t++)$ **do**
10: $m = (k-1) \times (t-1) + 1$;
11: **for** any subset $\{r_1, r_2, \cdots, r'_m\}$ in $\{r_1, r_2, \cdots, r_n\}$ **do**
12: $C = C \cup \{< \{r_1, r_2, \cdots, r'_m\}, t >\}$;
13: **end for**
14: **end for**
15: **end if**

Statement 1. *For the given collaborative strategy $e = cdod < \{r_1, r_2, \cdots, r_n\}, k >$ under status γ, it can be precisely enforced by the constraint formalized as $c = smer < \{r_1, r_2, \cdots, r_n\}, t >$, if and only if $t = 2$ when $k = n$ (or $t = n$ when $k = 2$).*

Theorem 1. *For the given collaborative strategy $e = cdod < \{r_1, r_2, \cdots, r_n\}, k >$ under status γ , the SMER constraint set constructed by Algorithm 1 is minimal.*

Proof.
According to Statement 1, $\{< \{r_1, r_2, \cdots, r_n\}, 2 >\}$ and $\{< \{r_1, r_2, \cdots, r_n\}, n >\}$ is the minimal set required. Next, w_e need to verify whether if holds true when $2 < k < n$. The verification process considers the following two sides.

On one hand, Any $(k-1)$ users have $(k-1) \times (t-1)$ roles from $\{r_1, r_2, \cdots, r_m\}$ at most, since any user at most is allowed to have $(t-1)$ roles. Without loss of generality, let t takes the value $(\lfloor \frac{n-1}{k-1} \rfloor + 1)$, the number of roles covered by any $(k-1)$ users is: $(k-1) \times (\lfloor \frac{n-1}{k-1} \rfloor + 1 - 1) < n$, that is $sat_c(\gamma) \Rightarrow safe_e(\gamma) = 1$. When $t < (\lfloor \frac{n-1}{k-1} \rfloor + 1)$, it also holds true. Thus, for each $c = smer < \{r_1, r_2, \cdots, r_m\}, t >$ in the constructed C, c can enforce e.

On the other hand, the contradiction method is used. Assuming that c is not the minimal constraint to enforce e, and there exists another enforceable $t' - m'$ SMER c' that is less strict than c, then m' should not be greater than m, and t' should be greater than t. There are two cases:

1) When $m' = m$ and $t' > t$, the assumption is true. Then, it is concluded that $t' > \lfloor \frac{n-1}{k-1} \rfloor + 1$. Without loss of generality, let t takes the value $(\lfloor \frac{n-1}{k-1} \rfloor + 2)$. For c', there exists $(k-1)$ users and the number of roles covered by these users is: $(k-1) \times (\frac{n-1}{k-1} + 2 - 1) = n - 1 + k - 1 > n$, then $sec_e(\gamma) = 0$, which breaches e. Thus, the assumption is false.

2) When $m' < m$ and $t' = t$, the assumption is true. If $sat_{c'}(\gamma) = 1$, then $sat_c(\gamma) = 1$, which indicates that c' that is not weaker than c. Thus, the assumption is false and the theorem is proved.

□

5 Experimental Analysis

To evaluate the performance of the RDA_TC&CSS, we next conduct experiments using the simulated system and real-world datasets, in order to demonstrate the reliability and security of the proposal. All the experiments are compiled and run under the Java environment.

5.1 Performance Evaluations for Reliability of the RDA_TC&CSS

5.1.1 Problem Statement

In the following simulated system, Figure 2 presents the role-hierarchy relationship (RH) of the RBAC system when a specified organization is purchasing a batch of products. Table 1 and Table 2 represent the relationship of the original user–role assignments (UA), and that of the original role–permission assignments (PA) in the multi-department collaborative working environment, respectively. If the assistant manager c is on a business trip and he needs to temporarily delegate his authorities to another collaborator, such as the assistant manager e, inspector g, or storing man h, in order to execute the corresponding tasks as the substitute of c, then the analysis for the delegation authorization is as follows.

5.1.2 Trust Computing

If the correspondence between the task attribute $t_{S\_DM_2}$ = {manager level, warehouse storage} and weight is: $wa_{\text{manager level}} = 0.6$, $wa_{\text{warehouse storage}} = 0.4$, then the basic seniorities of e, g, and h for $t_{S\_DM_2}$ are $A_{e \to t_{S\_DM_2}} = 0.6$, $A_{g \to t_{S\_DM_2}} = 0.4$, and $A_{h \to t_{S\_DM_2}} = 0.4$, respectively. Moreover, it is seen from Figure 2 that, S_DM_2, QP, and WP are on the same branch, while S_DM_2 and A_DM belong to different branches. If $dist(S\_DM_2, QP) = dist(S\_DM_2, WP) =$

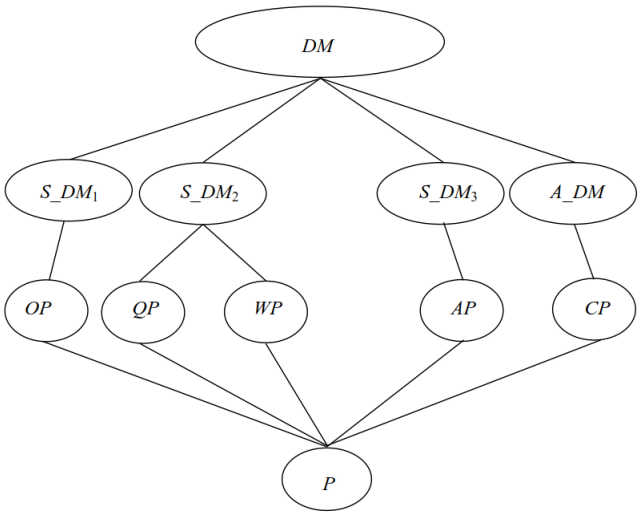

Figure 2: RH

Table 2: PA

Role	Permission	Annotation for permission
DM	p_1	It is responsible for general management
S_DM_1	p_2	It is responsible for ordering products
S_DM_2	p_3	It is responsible for keeping the warehouse
S_DM_3	p_4	It is responsible for financial accounting
A_DM	p_5	It is responsible for cash accounting
OP	p_6	Order products
QP	p_7	Inspect the product quality
WP	p_8	Store products
AP	p_9	keep accounts
CP	p_{10}	Revenue and expenditure cash
P	p_{11}	Collect and organize documents

Table 1: UA

User	Role	Annotation for role
a	DM	Department manager
b	S_DM_1	Assistant manager
c	S_DM_2	Assistant manager
d	S_DM_3	Assistant manager
e	A_DM	Assistant manager
f	OP	Ordering person
g	QP	Inspector
h	WP	Store keeper
i	AP	Accountant
j	CP	Cashier
k	P	Common staff

0.7, $dist(S\_DM_2, A\_DM) = 0$, then the affiliated seniorities of e, g, and h for $t_{S\_DM_2}$ are $RA_{e\to t_{S\_DM_2}} = 0$, $A_{g\to t_{S\_DM_2}} = 0.7$, and $A_{h\to t_{S\_DM_2}} = 0.7$, respectively.

If the basic seniority and affiliated seniority have the same contribution weight, then the trust seniorities of e, g, and h for $t_{S\_DM_2}$ are respectively calculated as:

$$\begin{aligned} P_{e\to t_{S\_DM_2}} &= w_a \times A_{e\to t_{S\_DM_2}} + w_{ra} \times RA_{e\to t_{S\_DM_2}} \\ &= 0.5 \times 0.6 + 0.5 \times 0 \\ &= 0.30 \\ P_{g\to t_{S\_DM_2}} &= w_a \times A_{g\to t_{S\_DM_2}} + w_{ra} \times RA_{g\to t_{S\_DM_2}} \\ &= 0.5 \times 0.4 + 0.5 \times 0.7 \\ &= 0.55 \\ P_{h\to t_{S\_DM_2}} &= w_a \times A_{h\to t_{S\_DM_2}} + w_{ra} \times RA_{h\to t_{S\_DM_2}} \\ &= 0.5 \times 0.4 + 0.5 \times 0.7 \\ &= 0.55 \end{aligned}$$

Taking the "year" as the measurement unit, the performances of e, g, and h in place of c in the last 5 years are investigated, respectively. Table 3 presents the completion effects with respect to the measurement unit and empirical coefficient, where "\" represents unknown. Providing that e, g, and h have the same empirical coefficient in the same unit of measurement. According to the table, the trust experiences of e, g, and h for $t_{S\_DM_2}$ are

respectively calculated as follows:

$$\begin{aligned} E_{e\to t_{S\_DM_2}} &= \sum_{k=1}^{5} wt_k \times e_{ek} \\ &= 0+0+0+0+1.0\times 0.8 \\ &= 0.80 \\ E_{g\to t_{S\_DM_2}} &= \sum_{k=1}^{5} wt_k \times e_{gk} \\ &= 0+0.4\times 0.6+0+0.8\times 0.4+0 \\ &= 0.56 \\ E_{h\to t_{S\_DM_2}} &= \sum_{k=1}^{5} wt_k \times e_{hk} \\ &= 0.2\times 0.7+0+0.6\times 0.5+0+0 \\ &= 0.44 \end{aligned}$$

In the existing collaboration environment, consider that the manager a, assistant managers b and d are selected as the referees of e, g, and h, respectively. Owing to the rank of a being above b, and d, the trust value of the system about the referee should be: $t_a > t_b$, and $t_a > t_d$. Table 4 presents the trust recommendations with respect to different referees. According to the table, the trust experiences of e, g, and h for $t_{S\_DM_2}$ are respectively calculated as follows:

$$\begin{aligned} R_{e\to t_{S\_DM_2}} &= \frac{t_a\times r_{ae}+t_b\times r_{be}+t_d\times r_{de}}{t_a+t_b+t_d} \\ &= \frac{0.6\times 0.9+0.2\times 0.5+0.2\times 0.1}{0.6+0.2+0.2} \\ &= 0.66 \\ R_{g\to t_{S\_DM_2}} &= \frac{t_a\times r_{ag}+t_b\times r_{bg}+t_d\times r_{dg}}{t_a+t_b+t_d} \\ &= \frac{0.6\times 0.2+0.2\times 0.4+0.2\times 0.7}{0.6+0.2+0.2} \\ &= 0.34 \\ R_{h\to t_{S\_DM_2}} &= \frac{t_a\times r_{ah}+t_b\times r_{bh}+t_d\times r_{dh}}{t_a+t_b+t_d} \\ &= \frac{0.6\times 0.3+0.2\times 0.5+0.2\times 0.6}{0.6+0.2+0.2} \\ &= 0.40 \end{aligned}$$

According to different contributions of the trust seniority, trust experience and trust recommendation to the trust degree, while taking the value $w_p = 0.2$, $w_e = 0.7$, $w_r = 0.1$ and $w_p + w_e + w_r = 1$, the trust degrees of e, g, and h for $t_{S\_DM_2}$ are comprehensively calculated as follows:

$$\begin{aligned} T_{e\to t_{S\_DM_2}} &= w_p\times P_{e\to t_{S\_DM_2}} + w_e\times E_{e\to t_{S\_DM_2}} \\ &\quad + w_r\times R_{e\to t_{S\_DM_2}} \\ &= 0.2\times 0.3+0.7\times 0.8+0.1\times 0.66 \\ &= 0.686 \\ T_{g\to t_{S\_DM_2}} &= w_p\times P_{g\to t_{S\_DM_2}} + w_e\times E_{g\to t_{S\_DM_2}} \\ &\quad + w_r\times R_{g\to t_{S\_DM_2}} \\ &= 0.2\times 0.55+0.7\times 0.56+0.1\times 0.34 \\ &= 0.536 \\ T_{h\to t_{S\_DM_2}} &= w_p\times P_{h\to t_{S\_DM_2}} + w_e\times E_{h\to t_{S\_DM_2}} \\ &\quad + w_r\times R_{h\to t_{S\_DM_2}} \\ &= 0.2\times 0.55+0.7\times 0.44+0.1\times 0.4 \\ &= 0.458 \end{aligned}$$

For the given threshold $H_{t_{S\_DM_2}} = 0.5$, it can be concluded that, $T_{e\to t_{S\_DM_2}} > T_{g\to t_{S\_DM_2}} > H_{t_{S\_DM_2}} > T_{h\to t_{S\_DM_2}}$, which indicates that e and g are more trustworthy than h.

5.2 Performance Evaluations for Security of the RDA_TC&CSS

5.2.1 Performance Evaluations Using the Simulated System

According to the result of the above analysis, e and g are considered as the trustworthy delegated objects. If γ_1 and γ_2 respectively represent the system status after implementing delegation on candidates e, and g, then for different collaborative strategy $cdod < \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}, k >$, where $1 < k \leq 5$, Table 5 shows the minimal constraint set constructed by Algorithm 1, as well as descriptions for the satisfiability and security of γ_1 and γ_2. From the table, the following observations are presented.

1) C_1 can implicitly enforce e_1. Notice that $|\{Roles_{\gamma_1}(e) = \{A\_DM, S\_DM_2\}\} \cap \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}| = 2 < 5$ and $|\{Roles_{\gamma_2}(g) = \{QP, S\_DM_2\}\}b \cap \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}| = 1 < 5$, which indicates that γ_1 and γ_2 are satisfied to C_1 and are also secure to e_1. Thus, the more trusted assistant manager e should be selected for the delegation authorization in such case.

2) C_2, C_3, and C_4 can implicitly enforce e_2, e_3, and e_4, respectively, while notice that $|\{Roles_{\gamma_1}(e) = \{A\_DM, S\_DM_2\}\} \cap \{S\_DM_2, S\_DM_3, A\_DM\}| = 2$, which indicates that e violates $smer < \{S\_DM_2, S\_DM_3, A\_DM\}, 2 >$ in C_2, $smer < \{DM, S\_DM_2, S\_DM_3, A\_DM\}, 2 >$ in C_3, and $smer < \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}, 2 >$ in C_4. γ_1 is unsatisfied to C_2, C_3 and C_4 and is also unsecure to e_2, e_3 and e_4. However, $Roles_{\gamma_2}(g) = \{QP, S\_DM_2\}$, which

Table 3: Completion effects

Measurement unit (k)	Empirical coefficient (w_{tk})	Completion effect (e_{ek})	Completion effect (e_{gk})	Completion effect (e_{hk})
1	0.2	-	-	0.7
2	0.4	-	0.6	-
3	0.6	-	-	0.5
4	0.8	-	0.4	-
5	1.0	0.8	-	-

Table 4: Trust recommendations

Referee (u_{rk})	Trust value of referee (t_k)	Recommendation value (r_{ke})	Recommendation value (r_{kg})	Recommendation value (r_{kh})
a	0.6	0.9	0.2	0.3
b	0.2	0.5	0.4	0.5
d	0.2	0.1	0.7	0.6

indicates that g meets any constraint requirement from C_2, C_3 and C_4, and then γ_2 is also secure to e_2, e_3 and e_4. Thus, only the inspector g is selected as the delegated object in such cases, in order to ensure the security of the system status.

5.2.2 Performance Evaluations Using the Real-world Datasets

To further evaluate the efficiency of the algorithm for the construction of SMER constraints, we consider the real-world datasets used in the work [14]. However, the Domino, Firewall1, Firewall2, and Healthcare datasets could not reflect the performance of the proposal, since some users in these datasets violate the security strategies, from which valid SMERs cannot be generated. Thus, only five datasets are taken into consideration for the experiments, including Americas-large, Americas-small, Apj, Customer, and Emea. Further, the regular mining tool RMiner [11] is used for mining the initial roles with no constraints, as well as UA.

Different types of the $k-n$ CDOD strategy are synthetically generated using a simulator. In terms of the length of constraint enforcement, we study four different cases: 2-2 CDOD, 2-3 CDOD, 3-5 CDOD, and 5-10 CDOD, where n permissions are randomly chosen from the set of all permissions. Meanwhile, the sizes of $k-n$ CDOD are fixed as 30, and 50, respectively. We consider the initial mining results and the $k-n$ CDOD constraints as inputs and repeatedly conduct the experiments 20 times. The average time for constructing SMERs from different CDOD strategies, as well as the results of the compared methods RMP_SoD and RMO_SODSDA, are shown in Tables 6 $\sim$ 13, where Ek-n CDOD indicates the CDOD strategy, and $C_{t-mSMER}$ indicates the SMER constraint set.

When $|E_{k-nCDOD}| = 30$, it is intuitively observed from Tables 6 $\sim$ 9 that, the time needed for construction of the t-t SMERs grows rapidly as the length of the CDOD strategy increases. Take the Americas-large dataset as an example, and the length of CDOD changes from 2 to 10. The time for constructing t-t SMERs is 149, 511, 2078, and 6267s, respectively. Similarly, the time needed for construction of the $t-m$ SMERs is 289, 567, 1315, and 7771s, respectively, which also grows rapidly as the length of the CDOD strategy increases. However, most of the time taken for the latter is longer than that of the former. This is because the t-t SMER is more restricted than the $t-m$ SMER, and the number of constructing $t-m$ SMERs should be greater than that of constructing t-t SMERs. Thus, the time cost for the construction is longer with the increasing number of constraints. Similar to the above analysis for $|E_{k-nCDOD}| = 30$, the detailed analysis of the time taken for $|E_{k-nCDOD}| = 50$, as shown in Tables 10 $\sim$ 13, is omitted owing to the limited space.

A further observation from Tables 6 $\sim$ 13 is that, the time taken for construction of SMERs using different methods are comparable. Specifically, taking the Americas-small dataset as an example, when $|E_{2\text{-}3CDOD}| = 30$, the time taken for construction of $C_{t-mSMER}$ using the three methods is 28, 29, and 28s, respectively; when $|E_{3\text{-}5CDOD}| = 30$, the time for $C_{t-mSMER}$ using these methods is 91, 92, and 94, respectively. Thus, RDA_TC&CSS performs as well as the RMP_SoD and RMO_SODSDA methods, in order to ensure the system security, while improving the reliability of the delegation process.

Table 5: Enforcement of the Security Strategy

$k-n$ CDOD	$t-m$ SMER	$sat(\gamma_1)$	$sec(\gamma_1)$	$sat(\gamma_2)$	$sec(\gamma_2)$
$e_1 = cdod < \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}, 2 >$	$C_1 = \{smer < \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}, 5 >\}$	$sat_{C_1}(\gamma_1) = 1$	$sec_{e_1}(\gamma_1) = 1$	$sat_{C_1}(\gamma_2) = 1$	$sec_{e_1}(\gamma_2) = 1$
$e_2 = cdod < \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}, 3 >$	$C_2 = \{smer < \{DM, S\_DM_1, S\_DM_2\}, 2 >, smer < \{DM, S\_DM_1, S\_DM_3\}, 2 >, smer < \{DM, S\_DM_1, A\_DM\}, 2 >, smer < \{DM, S\_DM_2, S\_DM_3\}, 2 >, smer < \{DM, S\_DM_2, A\_DM\}, 2 >, smer < \{DM, S\_DM_3, A\_DM\}, 2 >, smer < \{S\_DM_1, S\_DM_2, S\_DM_3\}, 2 >, smer < \{S\_DM_1, S\_DM_2, A\_DM\}, 2 >, smer < \{S\_DM_1, S\_DM_3, A\_DM\}, 2 >, smer < \{S\_DM_2, S\_DM_3, A\_DM\}, 2 >, smer < \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}, 3 >\}$	$sat_{C_2}(\gamma_1) = 0$	$se_{e_2}(\gamma_1) = 0$	$sat_{C_2}(\gamma_2) = 1$	$se_{e_2}(\gamma_2) = 1$
$e_3 = cdod < \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}, 4 >$	$C_3 = \{smer < \{DM, S\_DM_1, S\_DM_2, S\_DM_3\}, 2 >, smer < \{DM, S\_DM_1, S\_DM_2, A\_DM\}, 2 >, smer < \{DM, S\_DM_1, S\_DM_3, A\_DM\}, 2 >, smer < \{DM, S\_DM_2, S\_DM_3, A\_DM\}, 2 >, smer < \{S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}, 2 >\}$	$sat_{C_3}(\gamma_1) = 0$	$sec_{e_3}(\gamma_1) = 0$	$sat_{C_3}(\gamma_2) = 1$	$sec_{e_3}(\gamma_2) = 1$
$e_4 = cdod < \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}, 5 >$	$C_4 = \{smer < \{DM, S\_DM_1, S\_DM_2, S\_DM_3, A\_DM\}, 2 >\}$	$sat_{C_4}(\gamma_1) = 0$	$sec_{e_4}(\gamma_1) = 0$	$sat_{C_4}(\gamma_2) = 1$	$sec_{e_4}(\gamma_2) = 1$

Table 6: Performance comparison when $|E_{2\text{-}2CDOD}| = 30$

Dataset	**RMP_SoD(s)**		**RMO_SODSDA(s)**		**RDA_TC&CSS(s)**	
	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$
Americas-large	139.7	279.5	144.7	281.5	149.5	289.5
Americas-small	10.1	17.2	10.1	18.1	13.1	20.2
Apj	0.4	0.8	0.4	0.8	0.4	0.9
Customer	0.3	0.4	0.3	0.4	0.3	0.4
Emea	0.01	0.04	0.02	0.04	0.02	0.06

Table 7: Performance comparison when $|E_{2\text{-}3CDOD}| = 30$

Dataset	**RMP_SoD(s)**		**RMO_SODSDA(s)**		**RDA_TC&CSS(s)**	
	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$
Americas-large	505.5	552.9	509.5	552.9	511.7	567.3
Americas-small	30.5	28.3	31.8	29.3	32.2	28.7
Apj	1.4	1.2	1.4	1.2	1.4	1.5
Customer	0.3	0.3	0.3	0.3	0.3	0.3
Emea	0.07	0.06	0.07	0.06	0.07	0.06

Table 8: Performance comparison when $|E_{3\text{–}5CDOD}| = 30$

Dataset	RMP_SoD(s)		RMO_SODSDA(s)		RDA_TC&CSS(s)	
	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$
Americas-large	2022.7	1314.7	2054.3	1314.7	2078.4	1315.9
Americas-small	117.5	91.9	133.6	92.9	144.1	94.8
Apj	6.7	4.5	7.1	4.6	8.7	4.5
Customer	1.2	3.2	5.0	3.5	5.6	3.2
Emea	0.3	0.2	0.3	0.2	0.3	0.2

Table 9: Performance comparison when $|E_{5\text{–}10CDOD}| = 30$

Dataset	RMP_SoD(s)		RMO_SODSDA(s)		RDA_TC&CSS(s)	
	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$
Americas-large	6164.6	7771.9	6267.3	7778.4	6267.3	7771.9
Americas-small	260.7	460.5	371.4	465.7	371.4	464.5
Apj	19.7	28.9	23.3	29.9	23.3	29.3
Customer	0.9	34.8	28.1	36.8	28.1	35.7
Emea	0.7	0.8	0.7	0.8	0.7	0.8

Table 10: Performance comparison when $|E_{2\text{–}2CDOD}| = 50$

Dataset	RMP_SoD(s)		RMO_SODSDA(s)		RDA_TC&CSS(s)	
	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$
Americas-large	366.2	759.1	379.5	759.1	386.1	759.1
Americas-small	29.2	60.4	30.2	60.4	33.2	62.7
Apj	0.9	2.4	1.2	2.4	1.4	2.6
Customer	0.01	1.1	0.5	1.1	0.5	1.1
Emea	0.05	0.12	0.06	0.12	0.06	0.12

Table 11: Performance comparison when $|E_{2\text{–}3CDOD}| = 50$

Dataset	RMP_SoD(s)		RMO_SODSDA(s)		RDA_TC&CSS(s)	
	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$
Americas-large	1203.9	1078.1	1212.8	1078.1	1211.9	1070.7
Americas-small	89.9	85.6	96.3	85.6	97.4	85.6
Apj	3.6	2.7	3.1	2.7	3.0	2.3
Customer	1.3	0.6	0.7	0.6	0.6	0.5
Emea	0.02	0.1	0.1	0.1	0.1	0.1

Table 12: Performance comparison when $|E_{3\text{–}5CDOD}| = 50$

Dataset	RMP_SoD(s)		RMO_SODSDA(s)		RDA_TC&CSS(s)	
	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$
Americas-large	5762.9	3751.6	5861.9	3753.1	5872.2	3751.6
Americas-small	280.9	216.2	337.9	223.4	341.9	216.2
Apj	17.9	12.6	19.7	14.5	22.6	12.6
Customer	2.7	7.7	12.1	8.8	15.5	7.7
Emea	0.8	0.5	0.8	0.7	0.8	0.6

Table 13: Performance comparison when $|E_{5\text{–}10CDOD}| = 50$

Dataset	RMP_SoD(s)		RMO_SODSDA(s)		RDA_TC&CSS(s)	
	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$	$C_{t-tSMER}$	$C_{t-mSMER}$
Americas-large	14667.3	18612.1	15008.8	18876.6	15001.9	18612.1
Americas-small	5.9	1116.7	900.5	1155.1	897.1	1116.7
Apj	47.0	73.6	59.4	87.7	66.0	73.6
Customer	2.3	87.4	70.5	86.1	74.3	87.4
Emea	1.7	0.8	0.7	0.7	0.8	0.8

5.3 Discussion

From the above analysis for the reliability and security of the delegation-authorization process, we find the main benefits of the RDA_TC&CSS as follows.

1) Most of the existing delegation approaches have the problem of arbitrariness. The delegation process is unreliable and untrustworthy, and there exists the danger of abuse of privileges once the delegated object h with lower trustworthiness is selected. To improve the reliability of the delegation process, the proposed method comprehensively computes the trust degrees of different candidate objects based on quantitative analysis for the trust seniority, trust experience and trust recommendation. It eliminates the object h with low trust degree from the candidate set, and then chooses e, g as the trustworthy objects. Thus, the delegated object chosen via the RDA_TC&CSS becomes much more reliable.

2) On the basis of ensuring the reliability of the delegation process, the collaborative division strategies e_2, e_3, and e_4 in the simulated system are violated using the existing delegation approaches, and the security of the system status will be compromised after delegation. To ensure the system security, the proposed method utilizes the method of constructing the minimal set of mutually-exclusive-role constraints, which indirectly implements the collaborative security strategy, in order to further determine the most appropriate delegated object g in the specific collaborative scenario, while satisfying various constraint requirements of systems.

3) The proposed algorithm in the article intuitively reflects the satisfaction requirements of the system status. The time complexity for construction of the $t-m$ SMERs depends on the double loops. The execution number of the outer loop is $(\lfloor \frac{n-1}{k-1} \rfloor - 2)$; in the inner loop, for the particular m, it is necessary to combine any m roles from n roles in the collaboration. Thus, the total time complexity of the algorithm is $O(2^n)$. The efficiency of the algorithm decreases obviously as the value of n increases. In general, if m is small, then the efficiency of the algorithm is acceptable.

Compared to the existing research approaches, features of the proposal are presented as shown in Table 14, where a tick V indicates that the feature is available.

Nevertheless, the RDA_TC&CSS still has the limitation: As shown in Table 5, for the given collaborative division strategies e_1, e_2, e_3 and e_4 as well as the minimal constraint sets C_1, C_2, C_3 and C_4, it is seen that e_1, e_4 can be precisely enforced by C_1, and C_4, respectively. However, e_2, e_3 cannot be enforced by C_2 or C_3. Therefore, the minimal constraint set constructed by the algorithm may not be able to precisely implement the CDOD strategy.

6 Conclusions

A novel delegation-authorization method based on RBAC, called RDA_TC&CSS, was proposed in this study. First, we utilized the trust seniority, trust experience and trust recommendation to comprehensively compute the trust degrees of different candidate objects, and then chose the objects with higher values. Next, we adopted the collaborative security strategy to further determine the most appropriate delegated object in the specific collaborative scenario. Further, we presented the algorithm to indirectly implement the collaborative strategy by constructing the minimal set of constraints and verified the correctness of the algorithm. The experiments using a specific simulated system and the real-world datasets demonstrated that, the proposed method could improve the reliability of the delegation process, while ensuring the system security. Our future work will focus on studying how to implement the RDA_TC&CSS in practical scenarios such as the IoT, blockchain, and wireless sensor networks.

References

[1] D. Abdelfattah, H. A. Hassan, and F. A. Omara, "Enhancing highly-collaborative access control system using a new role-mapping algorithm," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 3, p. 2765, 2022.

[2] A. M. Abdul, A. A. K. Mohammad, P. Venkat Reddy, P. Nuthakki, R. Kancharla, R. Joshi, and N. Kan-

Table 14: Comparison of features

Feature	Sun *et al.* [21]	Pal *et al.* [16]	Ali *et al.* [4]	Khan *et al.* [9]	Abdul *et al.* [2]	Proposed method
Detailed implementation of delegation process	-	-	-	-	-	V
Fine-grained delegation	-	-	V	-	-	V
Reliability analysis	-	-	-	V	V	V
Security analysis	V	V	V	V	V	V

naiya Raja, "Enhancing security of mobile cloud computing by trust-and role-based access control," *Scientific Programming*, vol. 2022, pp. 1–10, 2022.

[3] M. U. Aftab, Z. Qin, N. W. Hundera, O. Ariyo, N. T. Son, and T. V Dinh, "Permission-based separation of duty in dynamic role-based access control model," *Sycmmetry*, vol. 11, no. 5, p. 669, 2019.

[4] G. Ali, N. Ahmad, Y. Cao, M. Asif, H. Cruickshank, and Q. E. Ali, "Blockchain based permission delegation and access control in Internet of Things (BACI)," *Computers & Security*, vol. 86, pp. 318–334, 2019.

[5] F. A. Alsulaiman, A. Miege, and A. E. Saddik, "Threshold-based collaborative access control(T-CAC)," in *Proceedings of International Symposium on Collaborative Technologies and Systems*, pp. 46–56, 2007.

[6] S. Ameer, J. Benson, and R. Sandhu, "Hybrid approaches (ABAC and RBAC) toward secure access control in smart home IoT," *IEEE Transactions on Dependable and Secure Computing*, 2022. doi: 10.1109/TDSC.2022.3216297.

[7] J. Crampton and H. Khambhammettu, "Delegation in role-based access control," *International Journal of Information Security*, vol. 7, no. 2, pp. 123–136, 2008.

[8] M. Ghafoorian, D. Abbasinezhad-Mood, and H. Shakeri, "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, 2019.

[9] K. H. Khan, I. U. Din, A. Almogren, H. A. Khattak, M. Ibrahim, and S. Nazir, "Secure delegation using enhanced capability model," *Security and Communication Networks*, vol. 2022, pp. 1–9, 2022.

[10] B. Lang, J. Wang, and Y. Liu, "Achieving flexible and self-contained data protection in cloud computing," *IEEE Access*, vol. 2017, no. 5, pp. 1510–1523, 2017.

[11] R. Li, H. Li, W. Wei, X. Ma, and X. Gu, "RMiner: a tool set for role mining," in *Proceedings of the 18th ACM Symposium on Access Control Models and Technologies*, pp. 193–196, 2013.

[12] F. Liu and C. Chang, "Access control model based on multidimensional measurement and context," *Computer Engineering*, vol. 37, no. 24, pp. 129–131, 135, 2011.

[13] W. Liu, H. Duan, H. Zhang, P. Ren, and J. Wu, "TRBAC: trust-based access control model," *Journal of Computer Research and Development*, vol. 48, no. 8, pp. 1414–1420, 2011.

[14] B. Mitra, S. Sural, J. Vaidya, and V. Atluri, "A Survey of Role Mining," *ACM Computing Surveys*, vol. 48, no. 4, pp. 1–37, 2016.

[15] F. Nazerian, H. Motameni, H. Nematzadeh, "Emergency role-based access control (E-RBAC) and analysis of model specifications with alloy," *Journal of Information Security and Applications*, vol. 45, pp. 131–142, 2019.

[16] S. Pal, T. Rabehaja, M. Hitchens, V. Varadharajan, and A. Hill, "On the design of a flexible delegation model for the Internet of Things using blockchain," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3521–3530, 2020.

[17] A. Roy, S. Sural, A. K. Majumdar, J. Vaidya, and V. Atluri, "Enabling workforce optimization in constrained attribute-based access control systems," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 4, pp. 1901–1913, 2019.

[18] P. Sarana, A. Roy, S. Sural, J. Vaidya, and V. Atluri, "Role mining in the presence of separation of duty constraints," in *Proceedings of the 11th International Conference (ICISS'15)*, pp. 98-117, 2015.

[19] W. Sun, H. Su, and H. Xie, "Policy-engineering optimization with visual representation and separation-of-duty constraints in attribute-based access control," *Future Internet*, vol. 12, no. 10, p. 164, 2020.

[20] W. Sun, S. Wei, H. Guo, and H. Liu, "Role-mining optimization with separation-of-duty constraints and security detections for authorizations," *Future Internet*, vol. 11, no. 9, p. 201, 2019.

[21] W. Sun, X. Yuan, and H. Su, "Role-engineering optimization with user-oriented cardinality constraints in role-based access control," *International Journal of Network Security*, vol. 23, no. 5, pp. 845–855, 2021.

[22] C. Uikey and D. S. Bhilare, "RBACA: Role-based access control architecture for multi-domain cloud environment," *International Journal of Business Information Systems*, vol. 28, no. 1, pp. 1–17, 2018.
[23] G. Yu, R. Li, Z. Lu, W. Song, and Z. Tang, "Locale-based access control model in collaborative environment," *Computer Science*, vol. 36, no. 1, pp. 81–85, 2009.
[24] J. Zhang, T. Li, Z. Ying, and J. Ma, "Trust-based secure multi-cloud collaboration framework in cloud-fog-assisted IoT," *IEEE Transactions on Cloud Computing*, 2022.
[25] Y. Zhu, "Dynamic multi-level access control model based on user trust," *Computer Engineering*, vol. 37, no. 23, pp. 129–131, 2011.

Biography

Wei Sun received his B.S. and M.S. degrees from the School of Information Engineering, Zhengzhou University, China, in 2003 and 2008, respectively. He is currently working in the School of Computer and Information Technology, Xinyang Normal University. His current research interests include access control and system security.