Towards Role Mappings in Hybrid Cloud Environments: A Systematic Literature Review

Maximilian Niedermeier and Holger Wittges

Technical University of Munich, Boltzmannstr. 3, 85748 Garching, Germany,
{max.niedermeier, holger.wittges}@tum.de

Abstract. In current IT landscapes, there is a trend towards deploying multiple services each incorporating its own identity management system. When implementing role-based access control (RBAC), each system might utilize different roles adjusted to its own domain. We consider inter-domain role mapping as a solution to this problem. In contrast to most of the existing work, we focus on synchronizing multiple access control systems used by a single organization. Therefore, we first introduce a framework as well as requirements for the successful implementation of role mappings from one central, organizational domain to various target domains. Next, we conduct a systematic literature review and show the current state-of-the-art in inter-domain role mapping. Finally, we compare the contents of the analyzed literature with our requirements to find open issues for effectively managing RBAC in hybrid cloud environments.

Keywords: Inter-Domain · Role Mapping · Interoperability · RBAC · Access Control · Hybrid Cloud · SaaS · Systematic Literature Review.

1 Introduction

Role-based access control (RBAC) is one of the most popular models for access management. For example, RBAC is the preferred choice for managing the security of electronic health record systems [1]. Today, organizations tend to not just use a single software, but a hybrid landscape consisting of multiple services. For example, current research in the field of ERP system design deals with the question of how to successfully integrate on-premise infrastructure and software-as-a-service (SaaS) products [2]. Such implementations correspond to the definition of hybrid clouds which can, for example, consist of a private cloud hosted on-premise and several public or community cloud-based extensions [3]. As interoperability is a requirement for identity management systems [4], hybrid cloud deployments require companies to synchronize the roles of various independent RBAC units.

If, on the other hand, an organization views their access control systems independently of each other, there is a risk that it will lose track of its user authorizations. The IRBAC 2000 model [5] aims to bring interoperability to RBAC by adding inter-domain role mappings. Such mappings connect roles of different domains and thereby specify which permissions a user of a certain role has in each other domain. There are many research papers expanding the IRBAC 2000 model by deep-diving into various aspects of inter-domain role mapping. However, conducting role mappings is not trivial: Security officers always need to keep in mind that unwise mappings could result in the violation of security guidelines. To the best of our knowledge, an overview of existing work is missing. This hampers researchers to quickly find open issues or answers to cross-literature questions. Also, an overview would allow security officers to conduct role mappings according to state-of-the-art guidelines.

In this work, we are interested in the question whether role mapping is suited for hybrid cloud architectures consisting of several distributed domains that all belong to the same organization. Therefore, we conduct a literature review on the topic of role mapping and aim to answer the following two research questions:

RQ1: What is the current state-of-the-art in inter-domain role mapping research?

RQ2: What are open challenges in inter-domain role mapping research, especially in regards to hybrid clouds consisting of several stand-alone domains?

Whereas our literature review directly answers *RQ1*, we propose a more complex method for answering *RQ2*: First, we propose a hybrid cloud role mapping framework which is based on the IRBAC 2000 model. In contrast to the original model, our framework maps roles between domains that all belong to the same organization. Also, our mappings are unidirectional and connect central, organizational roles with domain-specific roles. Next, we introduce several requirements which an organization needs to consider when mapping roles in such a setting. Finally, we compare these requirements to the results of our literature review and thereby find open challenges which current research does not address.

The following paper contents are structured as follows: Section 2 presents background knowledge necessary to fully understand our research contribution. In Section 3, we propose our hybrid cloud role mapping framework and the corresponding implementation requirements. Section 4 explains the methodology of our scientific literature review. Afterwards, Section 5 shows the current state-of-the-art in inter-domain role mapping. In Section 6, we answer our research questions as previously explained, mention the limitations of our contribution and show a future research agenda. Finally, we conclude our findings and summarize our research in Section 7.

2 Background

In this section, we consolidate fundamental concepts required for understanding the following contents of this research paper. Section 2.1 summarizes key features of RBAC, an access control model that utilizes roles for granting permissions to users. Afterwards, Section 2.2 presents the IRBAC 2000 model which introduces role-to-role mappings between different domains. Finally, Section 2.3 explains hybrid hierarchies and the inter-domain role mapping (IDRM) problem.

2.1 Role-Based Access Control (RBAC)

Ferraiolo et al. [6] propose a NIST standard for *Role-Based Access Control* (RBAC). In our summary, we do not address every single element of their RBAC model, but rather explain the concepts which are generally used in inter-domain role mapping research; the core RBAC concepts are depicted in Figure 1.

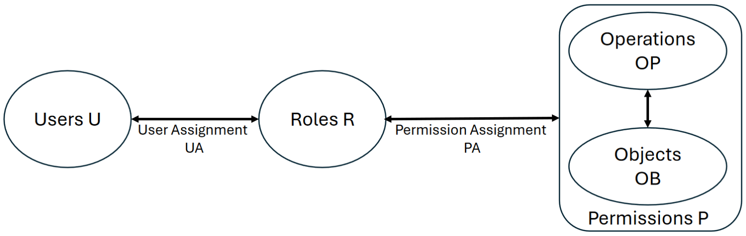

Fig. 1. The interaction between users, roles and permissions in RBAC.

- Users U are human persons, for example employees or guests, who need access to at least one object.
- Roles R are the key concept of RBAC. In a simplified example, a company could use their job positions as roles.
- Objects OB are use-case dependent, access-restricted resources. For example, an object could be a text file, an executable program or a database table.
- Operations OP are executed by users on objects. For example, reading or writing to a text file (object) are operations.
- Permissions $P = 2^{(OB \times OP)}$ are a set of all possible authorization subset combinations including the empty set and the full set. The Cartesian product $OB \times OP$ includes all combinations of object and operation elements.
- The user assignment $UA \subseteq U \times R$ is a many-to-many relation between users and roles. This means, that a user can be assigned to multiple roles and a role could be granted to different users.
- The permission assignment $PA \subseteq P \times R$ is a many-to-many relation between roles and permissions. This means, that a role could be assigned to multiple permission elements and a permission could be granted to different roles.

The authors define general role hierarchies H as partial orders on the role set R, thus $H \subseteq R \times R$. If $(x, y) \in H$ (also written as $x > y$), then the role x is an ancestor of y and inherits all permissions granted to the successor role y. Figure 2 shows a small example in which $R = \{$Administrator, Consultant, Developer$\}$ and $H = \{($Administrator, Consultant$)$, $($Administrator, Developer$)\}$. This hierarchy follows, that users of the ancestor role *Administrator* have all permissions assigned to the successor roles *Consultant* and *Developer*.

Core RBAC also includes sessions S, which allow RBAC systems to distinguish between assigned and activated roles. More precisely, a session is a mapping

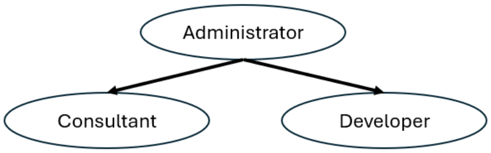

Fig. 2. A simple example of a general role hierarchy.

between a user and some of their assigned roles, namely the active roles. Each session therefore describes a time frame in which only some roles are active. During a session, users can only use permissions of currently active roles. Next, we propose two security principles which have to be maintained in RBAC.

Principle of Least Privilege Sandhu et al. [7] describe the *Principle of Least Privilege* as an administrative security approach which grants users only those rights which they actually need to perform their job, and no more. In regards to RBAC, assigning this minimized set of permissions works by defining a proper user and permission assignment.

Separation of Duty Simon and Zurko [8] define *Separation of Duty* (SoD) as a security approach tackling fraud by requiring multiple people for the completion of certain tasks. The authors differ between two variants: *Static Separation of Duty* prevents a user from being a member of conflicting roles. *Dynamic Separation of Duty* allows users being assigned conflicting roles if certain conditions are met. For example, users may not activate conflicting roles in the same session.

2.2 IRBAC 2000 Model

Kapadia et al. [5] propose the *IRBAC 2000* model and extend RBAC by adding a role mapping framework for collaborative environments. To be more precise, the authors consider two different administrative domains D_0 and D_1, each having their own role set R_0, R_1 and hierarchy H_0, H_1. The organizations now decide to work together and give users of domain D_1 access to domain D_0. Therefore, let us assume any roles $x \in R_0$ and $y \in R_1$. In the following, we represent these role set assignments as x_{R_0} or respectively y_{R_1}. An association $y_{R_1} \mapsto x_{R_0}$ implies, that there is a role translation and users of role y_{R_1} in domain D_1 are now considered as of role x_{R_0} in domain D_0. Figure 3 shows an example.

In this setting, the company of domain D_1 sends human resources to a project in domain D_0. The administrator of domain D_0 conducts a role mapping (shown as dashed arrows) from H_1 to H_0. If a translation is marked with *NT*, it is non-transitive. Otherwise, role mappings are transitive. If our exemplary role mapping $y_{R_1} \mapsto x_{R_0}$ is transitive, it follows that $\forall z \in R_1$, if $z_{R_1} > y_{R_1}$ then $z_{R_1} > x_{R_0}$. This means, that all ancestors z_{R_1} of role y_{R_1} inherit its role translations and are implicitly mapped to the role x_{R_0} of domain D_0. For non-transitive role mappings, this property does not hold.

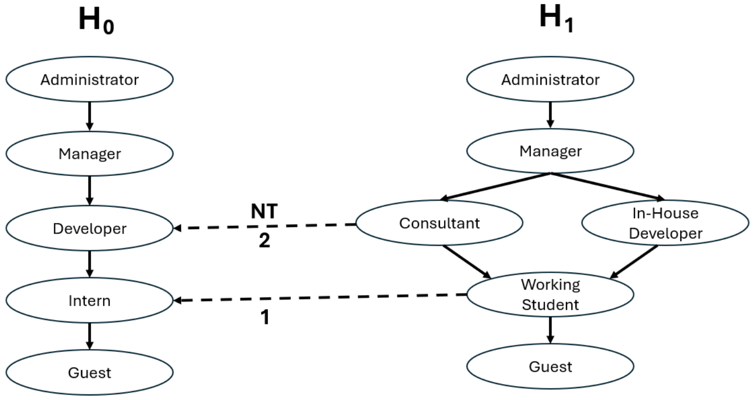

Fig. 3. A simple role mapping example between two hierarchies H_0 and H_1.

Figure 3 includes a transitive role mapping $WorkingStudent_{R_1} \mapsto Intern_{R_0}$. Thus, all users of domain D_1 which are not only a $Guest_{R_1}$, have at least the same rights as an $Intern_{R_0}$ in domain D_0. There is also a non-transitive role translation $Consultant_{R_1} \mapsto_{NT} Developer_{R_0}$. Thus, all $Consultant_{R_1}$ users of domain D_1 have the same access rights as a $Developer_{R_0}$ in domain D_0. However, users of role $Manager_{R_1}$ or $Administrator_{R_1}$ in domain D_1 are still considered an $Intern_{R_0}$ in domain D_0.

2.3 Role Mapping in Hybrid Hierarchies

The *Generalized Temporal Role-Based Access Control* (GTRBAC) model extends RBAC by putting emphasis on temporal constraints for role activations [9]. In [10], Joshi et al. distinguish between three types of role hierarchies which are suitable for working with the GTRBAC model. For simplicity, we only introduce the unrestricted version of each hierarchy. In an *I-hierarchy*, the *permission-inheritance* known from the previously introduced general role hierarchy holds. Therefore, users who activate an ancestor role can also use the permissions actually assigned to successor nodes. An *A-hierarchy* relies on *activation-inheritance*, which means that users who can currently activate a certain ancestor role can also activate the corresponding successor roles. Finally, in an *IA-hierarchy*, both of the previously explained concepts apply.

Du and Joshi [11] define a *hybrid hierarchy* as a role hierarchy whose relation set can contain any relation that belongs to one of the three hierarchy types just introduced. They show that in a hybrid hierarchy, the complexity of finding the minimum role set that fulfills the permissions requested by a user is NP-complete. This issue is referred to as the *Inter-Domain Role Mapping* (IDRM) *problem*. The IDRM problem aims to fulfill the principle of least privilege for role mappings based on collaborative permission requests in hybrid hierarchies.

3 Hybrid Cloud Role Mapping Framework

In this section, we propose a centralized role mapping framework for hybrid clouds that connects one central domain to multiple target domains. Each target domain provides services for a cross-domain application and includes an own RBAC unit. The central domain does not contain a service, but is crucial for synchronizing the target domain role sets. Figure 4 shows an exemplary model.

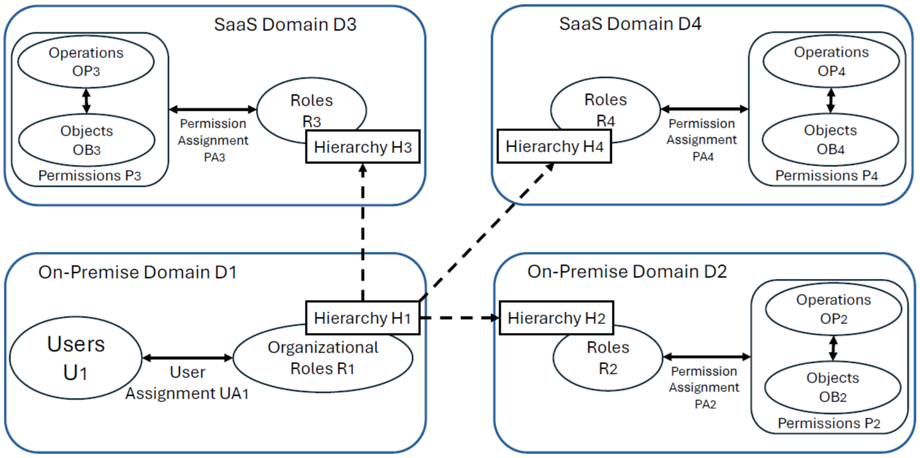

Fig. 4. Our centralized role mapping framework for hybrid cloud infrastructures in a simplified setting which consists of one central domain and three target domains.

As all domains belong to the same organization, there is only a single set of users U_1 stored in a central user database. These users require different permissions P_2, P_3 and P_4 in the target domains D_2, D_3 and D_4. In theory, it would be possible to consider all target domains independently of each other and only work with three user assignments UA_2, UA_3 and UA_4. However, this method could result in a security hazard as the overall access control structure would be difficult to trace, especially when there are more than three domains.

In our example, domain D_1 includes the central user database and thus we refer to it as the central domain. Therein, the user assignment UA_1 directly maps the users U_1 to organizational roles R_1. Since it is important to have full control over the user database, D_1 is an on-premise domain. Just as in the IRBAC 2000 model, role mappings (shown as dashed arrows) between the role hierarchies determine which roles in R_2, R_3 or R_4 are assigned to the users U_1. In contrast to the original model, our framework only allows unidirectional mappings from H_1 to the other hierarchies H_2, H_3 and H_4. Thereby, we intend to simplify the user management and also bring interoperability to the distributed access control systems. After all, the mappings unify the roles of all domains based on which organizational role(s) the users are assigned to in the central domain.

Depending on the organization design, different role sets and role hierarchies may be more or less similar, which makes the creation of role mappings more or less difficult. Cloud services could have a similar role structure to on-premise services if the company only uses them as replicas. For example, cloud backups can be used for disaster recovery [12]. However, different domains often complement each other and therefore deploy different services, each with its own role structure. Based on our framework, we present six requirements for successfully conducting role mappings in a hybrid cloud consisting of heterogeneous domains.

- **REQ1:** Liu and Huang [13] mention that role mapping is limited to a coarse-grained user-role assignment as it only allows exact mappings between roles of different domains. The authors state that in a more realistic scenario, organizations could prefer to only map some users of a certain role to another role. Similarly, we state that organizations might also prefer to only map parts of the permissions assigned to a role. Both issues correlate to the principle of least privilege and thus follow that *non-requested permissions granted by role mappings should be minimized.*
- **REQ2:** Role mappings between different domains may generate conflicts in the access control structure. In [5], Kapadia et al. already mention some security issues as well as possible resolutions. However, the authors do not mention how to guarantee separation of duty. We require that there are *no role mappings that allow a user to activate conflicting roles at the same time.*
- **REQ3:** Due to the increased complexity and workload faced when integrating various, rapidly changing cloud-based identity management systems, generating role mappings should be *automated as much as possible.*
- **REQ4:** Li et al. [14] state that changing the role of a user might lead to unauthorized access and could even have cascading effects. In our case, modifying a user assignment may have major consequences for their assigned permissions in all other domains as the mappings for the new role directly apply. Thus, it is important that security officers can *effectively monitor* role mappings as well as their impact.
- **REQ5:** Usually, each target domain is managed by an own designated administrator who is specialized in the respective domain topic. Thus, establishing role mappings from the central domain to each other domain involves both the target domain administrators and the security officers who create the mappings. As a result, mapping roles is a highly collaborative task which needs to be *easily understandable* for each actor taking part.
- **REQ6:** Our role mapping framework synchronizes role-based access control across different domains. However, there is a need for assuring that *only trusted domains participate in this process.* In the worst case, a foreign domain that belongs neither to the own organization nor to a known collaborating company would be able to map their roles to the local infrastructure.

4 Research Methodology

Our goal is not only to find the current state-of-the-art in inter-domain role mapping, but also to examine the emerging challenges in hybrid cloud environments.

Therefore, we compare the requirements introduced in Section 3 against results from a concept-centric literature review according to Webster and Watson [15].

In April 2024, we began to search for suitable articles by systematically querying the *Scopus* database. Thereby, we only included work having the exact term *role mapping* written in the paper abstract. Also, we limited our search by only including journal articles and conference papers. In this process, we could generate 124 hits. Similarly, we queried the *ACM Digital Library*, the *AIS eLibrary* and the *IEEE Xplore* database. When analyzing the results of searching the *ACM Digital Library* and *IEEE Xplore*, we found that the hits were a subset of our *Scopus* search results. Querying the *AIS eLibrary* resulted in two new hits.

Besides *Scopus*, the *Web of Science Core Collection* is another world-leading source for academic work [16]. Thus, in August 2024, we also queried this database and found 65 hits. When comparing the results to our *Scopus* hits, we noticed that the search in the *Web of Science Core Collection* resulted in four new hits.

In total, we could generate 130 unique hits. We first read the abstract of each paper and if we afterwards still had doubts whether the article is relevant for our research, we continued to look into the paper contents. As for inclusion criteria, we classified a hit as relevant if it contributes to **expanding** the research field of inter-domain role mapping presented in our background Section 2. Also, the article had to be **available** to us in any online library. After analyzing all hits, we found 44 relevant sources. Finally, we also performed a backward and forward search, which resulted in four more relevant publications. This means, that we include a total of 48 papers in our review. Table 1 provides an overview of the scientific literature search we just proposed in this section.

Table 1. An overview of our scientific literature search to the topic of inter-domain role mapping between April and August 2024.

Database	**Limited to**	**Search String**	**Hits**	**Relevant**
ACM Digital Library	RESEARCH-ARTICLE, ARTICLE	Abstract:("role mapping")	2	2
AIS eLibrary	Journal, Conference, Series	abstract:"role mapping"	2	0
IEEE Xplore	Journals, Conferences	("Abstract":"role mapping")	32	19
Scopus	Article, Conference Paper	ABS("role mapping")	124	43
Web of Science Core Collection	Article, Proceeding Paper	"role mapping" (Abstract)	65	26
Sum (Unique)			**130**	**44**
Backward & Forward Search				4
In Total				**48**

5 Findings

When scanning the documents in more detail, we assign each article at least one concept and thereby group publications topic-wise. In the following subsections, we further explain each concept and focus on mentioning the respective contributions of each article. Table 2 shows our concept matrix with columns ordered according to the concept sizes. Overall, we can find six different topics.

Table 2. Concept matrix resulting from our systematic literature review.

		Concept					
Authors	**Article**	Conflict Resolution	Principle of Least Privilege	Trust Management	Implementation Project	Cloud Computing	Automation
Abdelfattah et al.	[17]		•			•	
Abdelfattah et al.	[18]		•			•	
Chen and Crampton	[19]		•				
Chen and Crampton	[20]	•	•				
Chen et al.	[21]	•					
Chen et al.	[22]			•			
Deng et al.	[23]	•					
Deng et al.	[24]	•		•			
Diao et al.	[25]						•
Du et al.	[26]	•					
Fan et al.	[27]			•			
Fan et al.	[28]	•			•		
Geethakumari et al.	[29]			•	•		
Ghosh et al.	[30]		•			•	
Guo et al.	[31]			•			
Hu et al.	[32]	•	•				
Hu et al.	[33]	•					
Hu et al.	[34]	•		•			
Huang et al.	[35]		•			•	
Huang et al.	[36]	•					
Kamath et al.	[37]	•			•		•
Kun et al.	[38]			•			
Li et al.	[39]	•		•			
Li et al.	[40]	•		•	•	•	
Li et al.	[41]						•
Lv et al.	[42]	•		•			
Pan et al.	[43]	•	•		•		
Shafiq et al.	[44]	•	•				
Shehab et al.	[45]	•		•			
Solanki et al.	[46]					•	
Sun et al.	[47]			•	•		
Tang et al.	[48]	•	•				
Tang et al.	[49]	•					
Unal and Caglayan	[50]				•		
Wang et al.	[51]			•			
Wang et al.	[52]	•					
Wang et al.	[53]	•					
Wang et al.	[54]	•	•				
Xia	[55]	•	•				
Xiang et al.	[56]	•	•				
Yang et al.	[57]	•			•		•
Yu et al.	[58]	•					
Zhang and Joshi	[59]	•	•				
Zhang and Joshi	[60]		•				
Zhang and Li	[61]			•	•	•	
Zhang et al.	[62]		•				
Zhang et al.	[63]						•
Zuo et al.	[64]	•			•		

5.1 Conflict Resolution

The largest subset of role mapping research is concerned with the algorithmic resolution of associated conflicts. There are particularly many articles resolving separation of duty conflicts. We find research about both assuring static SoD [23, 24, 26, 32–34, 49, 52, 54, 55, 64] and dynamic SoD [43–45, 48, 59]. In terms of static SoD, some authors directly mention to work with constraints for *Static Mutually Exclusive Roles* (SMER) [23, 24, 26, 32–34, 49, 64]. In [33] and [34], the authors also discuss the possibility to adjust the RBAC policies of the target domain in order to enable interoperation. Determining whether a static SoD problem can be solved is NP-complete [20].

When it comes to dynamic SoD, Zhang and Joshi [59] propose and solve the *User Authorization Query* (UAQ) problem which describes the issue of finding sufficient roles which can be activated during one session. Solving the UAQ problem is NP-hard [20]. Research also suggests to conduct role mappings based on *activation-inheritance* when connecting conflicting roles in a hybrid hierarchy [43]. This also applies to the case where two users are not allowed to activate the same role at the same time [48]. Shehab et al. [45] examine multi-domain access paths based on role mappings in order to resolve constraints for *Dynamic Mutually Exclusive Roles*. It may happen, that the joint use of two role mappings leads to a SoD conflict. Shafiq et al. [44] show how to formulate integer programs that efficiently decide which mapping should be removed.

Besides examining separation of duty conflicts, focus also lies on resolving cyclic inheritance [21, 36, 39, 40, 42–45, 52–54, 56, 58]. This issue describes a situation in which cyclic role mappings across different domains map a successor role to a higher-ranking ancestor role from the same domain. As a result, users which are actually assigned to the successor role can now also use the rights assigned to the ancestor role. Lastly, some contributions [28, 37, 44, 57] resolve semantic conflicts when creating a global access control policy based on role mappings. For example, different local policies may differ in naming conventions.

5.2 Principle of Least Privilege

Our results show that examining the principle of least privilege is another large subset of role mapping research. For example, we find solution approaches based on rules [54], also allowing direct permission assignments [32], creating new roles [17, 18] or splitting existing roles [43, 44, 48, 62]. Latter technique maps foreign roles to new subsets of local roles which are created based on the requested permissions. In [48], the authors also utilize request-splitting and thereby create subsets of permission requests. This approach is helpful if not all requested permissions can be acquired in the target domain.

Huang et al. [35] use a greedy approach for mining a minimal role set in a role mapping scenario. We find various articles proposing greedy algorithm(s) for solving the IDRM problem [19, 30, 55, 56, 59]. In [19], the authors state that their solution approach is based on an *availability* point of view which aims to find a minimal role set being assigned the requested permissions but only a minimized set of additional permissions. In contrast, the *safety* perspective is about finding a minimal role set granting the maximal set of requested permissions, but no other permissions. Similarly, Zhang and Joshi [59] also differ between an *availability* and a *least privilege*-based approach. Latter option is similar to the *safety* perspective introduced in [19]. In terms of computational complexity, the IDRM-availability problem is NP-hard, but the IDRM-safety problem is in P [20].

Ghosh et al. [30] solve the IDRM-availability problem and use various evaluations metrics for showing that their approach outperforms [11] and [19]. In [55] and [56], the authors improve the greedy algorithm introduced in [11]. In [60], Zhang and Joshi introduce the role-based domain discovery problem which is about finding domains that contain all resources correlating to a set of requested permissions. For fulfilling the principle of least privilege, the authors use one of the greedy algorithms presented in [59].

5.3 Trust Management

Before mapping roles from foreign domains to the local infrastructure, companies have to define trust relationships with the requesting organizations. After all, no untrusted users shall access the own systems. In relation to this topic, we find that most articles suggest [51] or use [22, 27, 31, 38, 40] a *Public Key Infrastructure* (PKI) for their authentication framework. In [22], the authors show research about role mappings in different circles of trust. Each cycle deploys its own PKI, which already connects different identity providers. When defining a trust relationship between identity providers of different trust cycles, these exit points require certificates of both PKIs associated with the two trust domains. Other articles do not directly mention PKIs, but also utilize private and public keys for secure communication between domains [39, 45, 47].

In [39] and [40], there is a central server connecting collaborating domains. When conducting role mappings, participating domains are never linked directly but the mappings always pass through a virtual role hierarchy. In [29] and [42], a central server is used for defining a global ranking system of domain-specific roles. In contrast, we also find research focusing on a distributed approach: Zhang and Li [61] only deploy a client-side and a target-side authentication module.

Some articles look at the topic of trust from a perspective other than authentication. Deng et al. [24] consider the migration of SMER constraints between collaborating domains. The authors state that the domain migrating a constraint needs to trust the other domains to understand and not manipulate the transferred constraint. In the architecture provided by Hu et al. [34], each domain implements a monitor module for evaluating the risk of an incoming request.

5.4 Implementation Project

Some research articles present practical projects that implement inter-domain role mappings. For example, Sun et al. [47] use a blockchain to make role mapping rules readable for the public. Some authors [28, 29, 40, 43, 57, 61] mention to use XACML for defining access control policies. SAML is used to define role memberships [40] or general authentication mechanisms [57, 61]. Both technologies are markup languages based on XML. Zuo et al. [64] use XML to define role mappings between domains. The authors structure their XML document as follows: Each domain contains its roles as sub-elements and, in turn, each role contains the roles to which it is mapped as sub-elements. Kamath et al. [37] use X-RBAC [65], which is a XML-based language for defining RBAC policies in multi-domain settings. Unal and Caglayan [50] introduce an own XML-based language for inter-domain access control including role mappings.

Besides the technologies used, we consider two projects to be especially useful for security officers who need to define role mappings: Fan et al. [28] develop a tool that automatically detects conflicts which are based on role mappings. The tool also summarizes the conflicts in corresponding analysis reports. Pan et al. [43] present a tool for visualizing multi-domain RBAC policies and illustrate in-between role mappings by connecting roles from different domains.

5.5 Cloud Computing

Due to our research question, we are especially interested in how role mapping is applied in cloud environments. We find that most research assumes different organizations which collaborate by sharing a cloud service [17, 18, 30, 46]. Some articles [17, 18, 46] directly mention the multi-tenancy of cloud products and focus on role mappings between the tenants. In [46], Solanki et al. introduce a super tenant which holds a mediaton role for the final mapping specifications. Ghosh et al. [30] assume that the provider domain conducts all role mappings based on the requests of the remote domains. In [40], the authors mention that there is a trend towards building a central, virtual server for connecting private and public cloud resources.

Even if not directly mentioning cloud computing, we also assign articles that focus on web services to this topic [35, 61]. In our understanding, web services are an equivalent to SaaS products. In [35], Huang et al. consider a composite web service which is similar to our hybrid cloud model consisting of multiple domains. Just like our considerations in Section 3, the authors mention that there are two options when granting employees access to a new domain: Either the organization creates additional users for all employees in each new domain (this correlates to extending the user assignments) or a single sign-on mechanism based on role mappings is implemented.

5.6 Automation

Zhang et al. [63] propose an algorithm for calculating the semantic similarity between two roles in a single role hierarchy. Therefore, the authors consider both the distance between the roles and their similarity in terms of assigned permissions. Similar to this approach, Diao et al. [25] and Li et al. [41] introduce inter-domain role mapping recommendations based on semantic similarities. This means that the latter two publications focus on the similarity of roles in different hierarchies. As for recommendation criteria, both articles use the following factors:

- **Similarity of concept sets:** The concept set of a role consists of various properties such as for example its name, its permissions and its description. When comparing concept sets between two roles, not only the actual terms but also WordNet-based synonyms are taken into consideration.
- **Similarity based on role hierarchy:** The position of a role within a hierarchy and its relationships to the other roles are also important factors to consider. To give a simple example, two roles from different domains may be very similar if both only have successors but no ancestors.

Other research aims to automate role mappings based on attributes [37, 57]. In [37], roles are considered more similar if the respectively assigned users share similar attributes. The authors also consider the synonyms of the attributes. Yang et al. [57] first translate attributes to numerical values and thus also prevent the issue that different domains may have assigned unequal terms to their roles.

6 Discussion

In this section, we interpret our results from Section 5. First, Subsection 6.1 shows a state-of-the-art in inter-domain role mapping research as we map the requirements from Section 3 to the concepts shown in Table 2. If we cannot map each requirement to an existing concept, we discover open issues that have not yet been addressed in scientific publications. In Subsection 6.2, we briefly mention the limitations of our research contribution. Finally, in Subsection 6.3, we conclude our discussion by showing our concrete future research directions.

6.1 Main Findings

Our literature review shows that existing work is very theoretical in its approach. Most research (see Subsections 5.1 and 5.2) aims to find algorithmic solutions for separation of duty conflicts or for the fulfillment of the principle of least privilege. We map both concepts to our requirements **REQ1** and **REQ2**. However, we have further remarks for applying the principle of least privilege: First, the original IDRM problem aims to find sufficient roles for a single user request. As to our understanding, however, it is possible to apply shown solutions for role-to-role mappings if entire user groups (sharing a common role) require certain permissions in another domain. Secondly, in contrast to approaches such as role splitting, solutions to the IDRM problem never create new roles and therefore prevent a role explosion. However, each variant of the IDRM problem also has disadvantages: Solving the IDRM-availability problem can lead to a solution which grants users additional, non-requested permissions and thus does not fully comply with the principle of least privilege. When solving the IDRM-safety problem, users may not be provided with every permissions needed. The bottom line is that security officers have to decide on a particular course of action, but would maybe prefer a trade-off between fulfilling the principle of least privilege and not generating a role explosion while still granting all permissions as requested.

Next, Subsection 5.3 shows how to set up a role mapping environment that only allows trusted domains to join a collaboration. We map this concept to our requirement **REQ6** which corresponds to this topic. Existing sources focus on PKIs but in [34], the authors also mention to evaluate the risks of foreign requests. We believe that such approaches are important because role mappings could contain errors. False mappings would allow users to exploit roles which they actually should not be assigned to. In order to prevent such issues, we recommend to investigate fraud detection mechanisms in inter-domain role mapping.

REQ3 states that role mappings should be automated as far as possible. We map this requirement to the corresponding concept that describes current automation approaches. As Subsection 5.6 shows, these are based on semantic similarity or attribute mappings. Both methods are not based on permission requests and are therefore difficult to reconcile with the principle of least privilege. However, we suggest to investigate a specific use case: Organizations may rent various cloud infrastructure platforms and thus use several, predefined roles provided by different cloud vendors. We wonder if roles predefined by different cloud vendors are similar and, if so, whether those can be mapped semantically.

REQ4 emphasizes the importance of efficiently monitoring role mappings. In found literature, there is only one approach which visualizes inter-domain role mappings [43]. We see this project as a first step in the right direction, but could not find an article which focuses on monitoring large role mapping architectures, whose impact may be difficult for humans to track. Thus, we do not consider this requirement to be met. Subsection 5.4 summarizes used technology stacks behind practical projects which implement role mappings. Researchers in the fields of security or software architecture can use this knowledge and build monitoring solutions which reduce the risk of losing track of complex mapping structures.

REQ5 states that conducting role mappings is a collaborative task and should therefore be easy to understand for everyone involved. However, we cannot find guidelines that explain how to proceed in a realistic setting. As said, most of the found sources are theory-based. Thus, current research cannot fulfill this requirement. Subsection 5.5 summarizes existing publications dealing with role mappings in cloud environments. However, current research does not focus on practical cloud computing examples in enterprise structures. Rather, cloud computing serves as a more modern example for role mapping algorithms due to its multi-tenancy capabilities. We suggest to extend this research domain by showing how to implement role mappings in a corporate cloud setting.

6.2 Limitations

First of all, it is important to mention that this article only considers RBAC. As already mentioned in the introduction, the reason for this is that RBAC is a popular choice for access control. Also, many systems support its implementation.

However, there are also other access control models, such as *Attribute-Based Access Control* (ABAC) [66]. In contrast to RBAC, ABAC does not use roles but attributes and policies to decide on authorization requests. In complex environments with multiple domains, the use of ABAC would therefore prevent security officers from having to manage a large number of roles.

Secondly, our literature review only includes role mapping research. When creating inter-domain role mappings based on permission requests, role mapping research assumes that both collaborating domains already contain role sets. Even if the role sets in target domains can be manipulated by splitting roles or adding new roles, the question remains as to how an organization should restructure their unified access control systems or build a solution from the ground up.

6.3 Future Research Agenda

Given the users U, the permissions P and the permissions each user requires, the *basic Role Mining Problem* (RMP) aims to find roles R, user assignments UA and permission assignments PA while minimizing the number of roles and matching the requested permissions with the permissions obtained by assigning the proposed roles [67]. For example, mining roles can be based on clustering [68], an **unsupervised machine learning** technique. In [67], the authors not only present the basic RMP, but also several of its variants, such as the *Minimal Noise Role Mining Problem* (MinNoise RMP), which fixes the number of roles while minimizing the difference between required permissions and actually assigned permissions. Referring to our findings in Subsection 6.1, role mining could thus enable security officers to find trade-offs between only assigning required permissions and generating a reasonable number of roles. Also, now referring to the limitations outlined in Subsection 6.2, role mining outputs the entire role set and its assignments to users and permissions. Role mining therefore suits the (re-)organization of role structures from the bottom up.

We are interested in examining how the RMP can be applied to a multi-domain environment. In particular, we aim to find role mining solutions for the framework we presented in Section 3. As a next step, we will investigate literature in the domain of role mining. Thereby, we aim to find out whether current research already contains solution approaches for hybrid cloud frameworks which include one central domain mapping roles to several target domains. Referring to our exemplary model introduced in Section 3, we suggest that a multi-domain role mining algorithm for hybrid clouds should follow the procedure below:

1. For all target domains D_2, D_3 and D_4, find out which permissions in P_2, P_3 and P_4 each user in U_1 requires.
2. For all target domains D_2, D_3 and D_4, solve some variant of the RMP.
3. For the central domain D_1, apply an algorithm which mines the organizational roles R_1 and takes into account that:
   - UA_2, UA_3 and UA_4 can be replaced by adding role mappings between the hierarchy H_1 and the target domain hierarchies H_2, H_3 and H_4.
   - The number of organizational roles R_1 is not too large.

- The deviation between the permissions required and the permissions received by using R_1, UA_1 and role mappings (instead of using UA_2, UA_3 and UA_4) is not too large.

After reviewing existing literature on the topic of role mining, we will further develop the proposed algorithm and test whether it is suited for realistic settings.

7 Conclusion

In this paper, we show two major research contributions: First, we define a hybrid cloud framework for inter-domain role mappings. In contrast to most of the existing literature, we assume role mappings between different domains that all belong to the same organization. We highlight several requirements which an organization should consider when implementing role mappings in such a distributed environment. Secondly, we conduct a systematic literature review and present the current state-of-the-art in the field of inter-domain role mapping. Our results show that existing research is mainly concerned with algorithmic solutions and less with practical examples in the field of cloud computing.

Finally, we combine our two contributions by mapping the requirements for our framework to the different concepts we found in existing literature. In doing so, we reveal various open challenges which future researchers can address to further improve the feasibility of a hybrid cloud role mapping framework. For example, current role mapping research lacks a direct comparison between fulfilling the principle of least privilege and generating a reasonable number of roles. We consider role mining as a suitable solution for filling this research gap. Thus, our next step is to further develop a multi-domain role mining algorithm.

Acknowledgements

This work is part of a research project supported by SAP SE to investigate *Very Large Business Applications* (VLBA). Also, we would like to thank the anonymous reviewers for their constructive feedback.

References

1. Fernández-Alemán, J. L., Señor, I. C., Lozoya, P. Á. O., Toval, A.: Security and privacy in electronic health records: A systematic literature review. Journal of Biomedical Informatics **46**(3), pp. 541–562 (2013).
2. Niranga, M., Wickramarachchi, R.: A Model for On-Premises ERP System and Cloud ERP Integration. In: Proceedings of the International Conference on Industrial Engineering and Operations Management, pp. 1381–1392. IEOM Society International, Dubai (2020).
3. Mell, P.M., Grance, T.: The NIST Definition of Cloud Computing. National Institute of Standards & Technology, Gaithersburg (2011).

4. Pöhn, D., Hommel, W.: An Overview of Limitations and Approaches in Identity Management. In: Proceedings of the 15th International Conference on Availability, Reliability and Security, pp. 1–10. ACM, Virtual Event Ireland (2020).
5. Kapadia, A., Al-Muhtadi, J., Campbell, R. H., Mickunas, D.: IRBAC 2000: Secure Interoperability Using Dynamic Role Translation. In: 1st International Conference on Internet Computing, pp. 231–238. Las Vegas (2000).
6. Ferraiolo, D. F., Sandhu, R., Gavrila, S., Kuhn, D. R., Chandramouli, R.: Proposed NIST Standard for Role-Based Access Control. ACM Transactions on Information and System Security **4**(3), pp. 224–274 (2001).
7. Sandhu, R., Ferraiolo, D., Kuhn, R.: The NIST Model for Role-Based Access Control: Towards A Unified Standard. In: Proceedings of the Fifth ACM Workshop on Role-Based Access Control, pp. 47–63. ACM, Berlin (2000).
8. Simon, R. T., Zurko, M. E.: Separation of Duty in Role-Based Environments. In: Proceedings 10th Computer Security Foundations Workshop, pp. 183–194. IEEE, Rockport (1997).
9. Joshi, J. B., Bertino, E., Latif, U., Ghafoor, A.: A Generalized Temporal Role-Based Access Control Model. IEEE Transactions on Knowledge and Data Engineering **17**(1), pp. 4–23 (2005).
10. Joshi, J. B., Bertino, E., Ghafoor, A.: Temporal Hierarchies and Inheritance Semantics for GTRBAC. In: Proceedings of the Seventh ACM Symposium on Access Control Models and Technologies, pp. 74–83. ACM, Monterey (2002).
11. Du, S., Joshi, J. B.: Supporting Authorization Query and Inter-domain Role Mapping in Presence of Hybrid Role Hierarchy. In: Proceedings of the Eleventh ACM Symposium on Access Control Models and Technologies, pp. 228–236. ACM, Lake Tahoe (2006).
12. Abualkishik, A. Z., Alwan, A. A., Gulzar, Y.: Disaster Recovery in Cloud Computing Systems: An Overview. International Journal of Advanced Computer Science and Applications **11**(9), pp. 702–710 (2020).
13. Liu, S., Huang, H.: Role-Based Access Control for Distributed Cooperation Environment. In: 2009 International Conference on Computational Intelligence and Security, pp. 455–459. IEEE, Beijing (2009).
14. Li, W., Wan, H., Ren, X., Li, S.: A Refined RBAC Model for Cloud Computing. In: 2012 IEEE/ACIS 11th International Conference on Computer and Information Science, pp. 43–48. IEEE, Shanghai (2012).
15. Webster, J., Watson, R. T.: Analyzing the Past to Prepare for the Future: Writing a Literature Review. MIS Quarterly **26**(2), pp. 13–23 (2002).
16. Zhu, J., Liu, W.: A tale of two databases: The use of Web of Science and Scopus in academic papers. Scientometrics **123**(1), pp. 321–335 (2020).
17. Abdelfattah, D., Hassan, H. A., Omara, F. A.: A novel role-mapping algorithm for enhancing highly collaborative access control system. Distributed and Parallel Databases **40**(2), pp. 521–558 (2022).
18. Abdelfattah, D., Hassan, H. A., Omara, F. A.: Enhancing highly-collaborative access control system using a new role-mapping algorithm. International Journal of Electrical and Computer Engineering **12**(3), pp. 2765–2782 (2022).
19. Chen, L., Crampton, J.: Inter-domain Role Mapping and Least Privilege. In: Proceedings of the 12th ACM Symposium on Access Control Models and Technologies, pp. 157–162. ACM, Sophia Antipolis (2007).
20. Chen, L., Crampton, J.: Set Covering Problems in Role-Based Access Control. In: 14th European Symposium on Research in Computer Security, pp. 689–704. Springer Berlin Heidelberg, Saint-Malo (2009).

21. Chen, X., Wu, D., Lin, J., Zhu, M.: A Security Violation Detection Method for RBAC Based Interoperation. In: 2006 International Conference on Computational Intelligence and Security, pp. 1491-1496. IEEE, Guangzhou (2006).
22. Chen, J., Wu, G., Ji, Z.: Secure interoperation of identity managements among different circles of trust. Computer Standards & Interfaces **33**(6), pp. 533–540 (2011).
23. Deng, L., He, Y., Xu, Z.: Enforcing Separation of Duty in Ad Hoc Collaboration. In: 2008 The 9th International Conference for Young Computer Scientists, pp. 1545–1552. IEEE, Hunan (2008).
24. Deng, L., Xu, Z., He, Y.: Trust-Based Constraint-Secure Interoperation for Dynamic Mediator-Free Collaboration. Journal of Computers **4**(9), pp. 862–872 (2009).
25. Diao, L., Wang, H., Alsarra, S., Yen, I. L., Bastani, F.: A Smart Role Mapping Recommendation System. In: 2019 IEEE 43rd Annual Computer Software and Applications Conference, pp. 135–140. IEEE, Milwaukee (2019).
26. Du, J., Chen, C., Zhu, J., Li, X.: Research on Association Securities in Cross-domain Interoperation Model in Pervasive Computing. In: 2008 Third International Conference on Pervasive Computing and Applications, pp. 953–958. IEEE, Alexandria (2008).
27. Fan, H., Xian, Z., Guanglin, X.: Distributed role-based access control for coaliagion application. Geo-spatial Information Science **8**(2), pp. 138–143 (2005).
28. Fan, B., Liang, X., Luo, Y., Bo, Y., Xia, C.: Conflict Detection Model of Access Control Policy in Collaborative Environment. In: 2011 International Conference on Computational and Information Sciences, pp. 377–381. IEEE, Chengdu (2011).
29. Geethakumari, G., Negi, A., Sastry, V. N.: A Cross-Domain Role Mapping and Authorization Framework for RBAC in Grid Systems. International Journal of Computers and Applications **6**(1), pp. 1–12 (2009).
30. Ghosh, N., Chatterjee, D., Ghosh, S. K.: An Efficient Heuristic-Based Role Mapping Framework for Secure and Fair Collaboration in SaaS Cloud. In: 2014 International Conference on Cloud and Autonomic Computing, pp. 227–236. IEEE, London (2014).
31. Guo, X., Chen, C., Du, J., Li, X.: Design of a Cross-Domain Privilege Management Prototype System. In: 2008 9th International Conference on Computer-Aided Industrial Design and Conceptual Design, pp. 1091–1095. IEEE, Kunming (2008).
32. Hu, J., Li, R., Lu, Z.: Establishing RBAC-Based Secure Interoperability in Decentralized Multi-Domain Environments. In: Information Security and Cryptology - ICISC 2007: 10th International Conference, pp. 49–63. Springer Berlin Heidelberg, Seoul (2007).
33. Hu, J., Li, R., Lu, Z.: On Role Mappings for RBAC-Based Secure Interoperation. In: 2009 Third International Conference on Network and System Security, pp. 270–277. IEEE, Gold Coast (2009).
34. Hu, J., Li, R., Lu, Z., Lu, J., Ma, X.: RAR: A role-and-risk based flexible framework for secure collaboration. Future Generation Computer Systems **27**(5), pp. 574–586 (2011).
35. Huang, C., Sun, J. L., Wang, X. Y., Si, Y. J.: Minimal role mining method for Web service composition. Journal of Zhejiang University-SCIENCE C (Computers & Electronics) **11**(5), pp. 328–339 (2010).
36. Huang, C., Sun, J., Wang, X., Wu, D.: Inconsistency Resolution Method for RBAC Based Interoperation. IEICE TRANSACTIONS on Information and Systems **93**(5), pp. 1070–1079 (2010).
37. Kamath, A., Liscano, R., El-Saddik, A.: User-Credential Based Role Mapping in Multi-Domain Environment. In: Proceedings of the 2006 International Conference

on Privacy, Security and Trust: Bridge the Gap Between PST Technologies and Business Services, Article No.: 62. ACM, Markham (2006).

38. Kun, H., Jing, Y., Xiaoming, D., Lu, W.: Distributed Access Control Model over Multi-trust Domain. In: 2012 International Conference on Computer Science and Electronics Engineering, pp. 595–598. IEEE, Hangzhou (2012).
39. Li, J., Huai, J., Hu, C.: PEACE-VO: A Secure Policy-Enabled Collaboration Framework for Virtual Organizations. In: 2007 26th IEEE International Symposium on Reliable Distributed Systems, pp. 199-208. IEEE, Beijing (2007).
40. Li, J., Huai, J., Hu, C., Zhu, Y.: A secure collaboration service for dynamic virtual organizations. Information Sciences **180**(17), pp. 3086–3107 (2010).
41. Li, F., Wang, H., Diao, L., Yen, I. L., Bastani, F.: Toward Semi-Automated Role Mapping for IoT Systems in Smart Cities. In: 2019 IEEE International Smart Cities Conference (ISC2), pp. 205–211. IEEE, Casablanca (2019).
42. Lv, B., Zhang, D., Mao, R., Yang, H.: A Multi-Level Cross-Domain Access Control Model Based On Role Mapping. In: 2016 4th International Conference on Mechanical Materials and Manufacturing Engineering, pp. 230–235. Atlantis Press, Wuhan (2016).
43. Pan, L., Liu, N., Zi, X.: Visualization Framework for Inter-Domain Access Control Policy Integration. China Communications **10**(3), pp. 67–75 (2013).
44. Shafiq, B., Joshi, J. B., Bertino, E., Ghafoor, A.: Secure Interoperation in a Multidomain Environment Employing RBAC Policies. IEEE Transactions on Knowledge and Data Engineering **17**(11), pp. 1557–1577 (2005).
45. Shehab, M., Bertino, E., Ghafoor, A.: SERAT : SEcure Role mApping Technique for Decentralized Secure Interoperability. In: Proceedings of the Tenth ACM Symposium on Access control Models and Technologies, pp. 159–167. ACM, Stockholm (2005).
46. Solanki, N., Zhu, W., Yen, I. L., Bastani, F., Rezvani, E.: Multi-tenant Access and Information Flow Control for SaaS. In: 2016 IEEE International Conference on Web Services (ICWS), pp. 99–106. IEEE, San Francisco (2016).
47. Sun, S., Chen, S., Du, R.: Trusted and Efficient Cross-Domain Access Control System Based on Blockchain. Scientific Programming **2020**(1), Article ID 8832568 (2020).
48. Tang, Z., Li, R., Lu, Z.: A Request-Driven Role Mapping for Secure Interoperation in Multi-Domain Environment. In: 2007 IFIP International Conference on Network and Parallel Computing Workshops (NPC 2007), pp. 83–90. IEEE, Dalian (2007).
49. Tang, G. Y., Wang, H. F., Cui, L. J.: Research of role-mapping associate conflict detection methods. Applied Mechanics and Materials **380**, pp. 2699-2702 (2013).
50. Unal, D., Caglayan, M. U.: XFPM-RBAC: XFPM-RBAC: XML Based Specification Language for Security Policies in Multi-Domain Mobile Networks. Security and Communication Networks **6**(12), pp. 1420–1444 (2013).
51. Wang, J., Zhang, H., Zhang, B.: Research on Safe Privilege Management Model in Trusted-Domains. In: 2008 International Symposium on Knowledge Acquisition and Modeling, pp. 350–355. IEEE, Wuhan (2008).
52. Wang, X., Gu, T., Guo, Y., Zheng, Y., Zong, J.: An Efficient Algorithm of Role Mapping across Security Domains in Data-Sharing Environments. In: 2008 The Ninth International Conference on Web-Age Information Management, pp. 606–611. IEEE, Zhangjiajie (2008).
53. Wang, X., Sun, J., Yang, X., Huang, C., Wu, D.: Security Violation Detection for RBAC Based Interoperation in Distributed Environment. IEICE Transactions on Information and Systems **91**(5), pp. 1447–1456 (2008).

54. Wang, X., Gu, T., Guo, Y., Zheng, Y., Zong, J., Gong, B.: An Algorithm for Role Mapping Across Multi-domains Employing RBAC. Chinese Journal of Electronics **18**(1), pp. 37–41 (2009).
55. Xia, X.: An Equivalent Access Based Approach for Building Collaboration Model between Distinct Access Control Models. In: 14th International Conference on Communications and Multimedia Security (CMS), pp. 185–194. Springer Berlin Heidelberg, Magdeburg (2013).
56. Xiang, H., Xia, X., Hu, H., Wang, S., Sang, J., Ye, C.: Approaches to Access Control Policy Comparison and the Inter-Domain Role Mapping Problem. Information Technology and Control **45**(3), pp. 278–288 (2016).
57. Yang, Z., Yang, L., Luo, X., Ma, L., Kou, B. S., Zhang, K.: Model of Domain based RBAC and Supporting Technologies. Journal of Computers **8**(5), pp. 1220–1229 (2013).
58. Yu, G., Li, Z., Li, R., Mudar, S.: Centralized Role-Based Access Control for Federated Multi-Domain Environments. Wuhan University Journal of Natural Sciences **11**(6), pp. 1688–1692 (2006).
59. Zhang, Y., Joshi, J. B.: UAQ: A Framework for User Authorization Query Processing in RBAC extended with Hybrid Hierarchy and Constraints. In: Proceedings of the 13th ACM Symposium on Access Control Models and Technologies, pp. 83–92. ACM, Estes Park (2008).
60. Zhang, Y., Joshi, J. B.: Role-based Domain Discovery in Decentralized Secure Interoperations. In: 2010 International Symposium on Collaborative Technologies and Systems, pp. 84–93. IEEE, Chicago (2010).
61. Zhang, W., Li, Y.: Federation Access Control Model Based on Web-Service. In: 2010 International Conference on E-Business and E-Government, pp. 38–41. IEEE, Guangzhou (2010).
62. Zhang, S., Kong, X., Wang, B.: Study on Role-Spliting and Its Ontology-Based Evaluation Methods during Role Mapping of Inter-domain. In: 2008 International Conference on Computer Science and Software Engineering, pp. 642–645. IEEE, Wuhan (2008).
63. Zhang, S., Chen, J., Wang, B.: The research of semantic similarity algorithm consideration of multi-factor ontology-based in access control. In: 2010 International Conference on Computer Application and System Modeling (ICCASM 2010), pp. V3-538–V3-542. IEEE, Taiyuan (2010).
64. Zuo, C., Li, R., Han, H., Lu, Z.: Security Assurance for Dynamic Role Mapping in a Multi-Domain Environment. In: 2007 International Conference on Computational Intelligence and Security (CIS 2007), pp. 735–739. IEEE, Harbin (2007).
65. Joshi, J. B., Bhatti, R., Bertino, E., Ghafoor, A.: Access-Control Language for Multidomain Environments. IEEE Internet Computing **8**(6), pp. 40–50 (2004).
66. Hu, V. C., Ferraiolo, D., Kuhn, R., Friedman, A. R., Lang, A. J., Cogdell, M. M., Schnitzer, A., Sandlin, K., Miller, R., Scarfone, K.: Guide to Attribute Based Access Control (ABAC) Definition and Considerations (Draft). NIST Special Publication **800**(162), pp. 1–54 (2013).
67. Vaidya, J., Atluri, V., Guo, Q.: The Role Mining Problem: Finding a Minimal Descriptive Set of Roles. In: Proceedings of the 12th ACM Symposium on Access control Models and Technologies, pp. 175–184. ACM, Sophia Antipolis (2007).
68. Vaidya, J., Atluri, V., Warner, J.: RoleMiner: Mining Roles using Subset Enumeration. In: ACM Conference on Computer and Communications Security, pp. 144–153. ACM, Alexandria (2006).