International Journal of Database Theory and Application
Vol.9, No.10 (2016), pp.163-176
http://dx.doi.org/10.14257/ijdta.2016.9.10.14

Research of Automatically Generate Mapping Mechanism Based on the Semantic Role

Feng wang[1], Lei Cui[2], Yizhen Wang[3] and Xinjiang Wei[1]

[1]School of Mathematics and Statistics Science, Ludong University, Yan Tai 264025, China
Key Laboratory of Language Resource Development and Application of Shandong Province, Yan Tai
[2]Information Engineering Department, Yan Tai Vocational College, Yan Tai, 264025, China
[3]Tilburg University, Tilburg, the Netherlands
wangfenglw@126.com

Abstract

Role based access control (RBAC) has been widely adopted in industrial and government. However RBAC is only suitable for closed enterprise environment. With modern Internet based application, collaboration and sharing among multiple organizations become essential and RBAC is no longer sufficient. Role mapping has been the solutions to deal with multiple domains, where the roles in the hierarchy of one organization are mapped to the roles in the hierarchy of another organization. But role mapping can be a tedious task for the security officers if it is done fully manually. Yet, performing role mapping automatically incur security risks. In this paper, we introduce a semi-automated role mapping process, where promising role mappings are generated automatically and recommended to the security officer(s). The security officers then approve or modify the recommended role mappings. We present a method for automatically generate role mappings based on the similarities of the roles in two role hierarchies. We use an example to illustrate our approach and show its feasibility.

Keywords*: role-based access control, role mapping, concept extraction, role similarity*

1. Introduction

With the rapid proliferation of Internet technologies, sharing and collaboration have become the common paradigm in the new, globalized cyber world. For example, cloud provides large-scale sharing of hardware, software, and services and facilitates collaboration among them. Federated data warehousing offers information sharing. Applications, such as globalized supply chains, emergency response, distributed design, *etc.*, require a wide range of sharing and collaboration. These sharing and collaborations raise security concerns. Proper access control and security defenses should be in place to assure that certain resources are only viewed, used, or modified by the entities who are intended to be allowed to view, use, or modify those resources.

There have been many advanced research results in access control technologies over the last two decades.

In the early era, basic access control schemes, such as access control matrix and capability lists, have been used [1]. From late 90s, role-based access control (RBAC) [2] becomes the major paradigm, especially for large enterprises and organizations. Role-based hierarchy semantically reflects the structure of authorities and responsibilities of the personnel in an organization and, hence, the access rights can be defined accordingly. Also, compared to other access control models, RBAC can greatly cut down

ISSN: 2005-4270 IJDTA

the cost for access control policy specifications.

Most of the traditional access control models, including RBAC, are proposed which are based on a closed system in which the users, roles, activities, and the accessed resources are well defined. Therefore, these traditional access control models cannot be directly applied to multi-domain systems, where cross domain accesses cannot always be properly defined in advance.

Attribute-based access control (ABAC) [3] is another access control model that has been extensively investigated in recent years. ABAC are suitable for the open systems where requesters (users and processes) are rarely pre-known to the access module. Attribute-based access control can be regarded as a natural extension of many conventional access control models (*e.g*. multi-level security model, role-based model, *etc*.), and is highly expressive. However, the cost of policy specification and decision making in ABAC greatly depends on the set of attributes selected for the involved domains (*e.g*. its size). Moreover, there is no well formed standard for attribute-based models yet, and, thus, are hard to be put to use in practice.

Cross domain role mapping is another potential solution to achieve access control in multi-domain systems. It extends the RBAC model and maps the roles of foreign collaborative domains into the local roles of each domain [2,4-8]. There are generally two approaches in role mapping. First, a trusted mediator can be used to integrate the role hierarchies of two correlative domains [5]. This approach has the scalability problem and requires a fully trusted mediator to perform the integration. Sometimes, the access rights defined in the global role hierarchy have conflicts with the access control requirements of the individual domains. That is, a subordinate user in the hierarchy may acquire the permission of a superior user through a chain of inter-domain mappings of roles or security classes. Hence, many of the works in the literature focus on conflict resolution and the optimization of this process. (in terms of fairness: due to conflict resolution inter-domain mappings of some domains are removed but the mappings of others remain). In [8], mediator-free solutions are proposed to secure cross-domain interoperation. They do not perform the integration of the hierarchies. They record all the inter-domain mappings that are activated to enable the access, and deny the access when there is a cycle.

In existing role mapping approaches [5-10], the associations of roles from different domains are done solely by security officers. Such manual process can be tedious and lack of agility. In some applications, the role mapping may be required on demand. For example, in an emergency response scenario, new parties may come to the aid due to special needs in special circumstances, and they will need immediate information and resources sharing with all assisting teams. In order to assure proper access control, role mappings among the new and existing responding organizations need to be done dynamically in real-time. Thus, a certain degree of automation is needed in order to achieve the timeliness goal in role mapping. However, security is a serious issue and automation is definitely not a full solution. Thus a rigorous process should be defined such that automated analysis can be done to come up with "recommendations" of role associations and mappings to cut the cost and time for the role mapping process. Security officers of the involved domains should verify and validate the recommendations to assure proper access control policy definition and enforcement.

The automated role mapping generation process is based on similarity of roles. It extracts key concepts for each role from the role name, role descriptions, role responsibilities, and the descriptions of the permissions assigned to the role. Based on the key concepts, the plain similarity between two roles is computed. Conceptually, the position of the role in the role hierarchy, *i.e*., the parents and children of the role, also provides plenty of information about what the role is. Thus, we also consider the role hierarchy in the similarity metric for the roles. Based on an aggregated similarity measures between roles, role mappings are generated and recommended. Such

recommendation of role mappings can be altered individually or system-wide to improve its flexibility. Also, the recommendation can help security officers and cut down the time and efforts for role mappings.

The rest of this paper is organized as follows. Section II provides a running example to illustrate the role mapping process. Section III discusses the semi-automated role mapping process and the corresponding system architecture. Section IV focuses on the techniques for automated analysis and recommendation of role mappings. Section V states the conclusion of the paper.

2. A Running Example

Healthcare systems contain numerous private data and are frequently considered in security research. In this paper, we take some health care systems as the running examples.

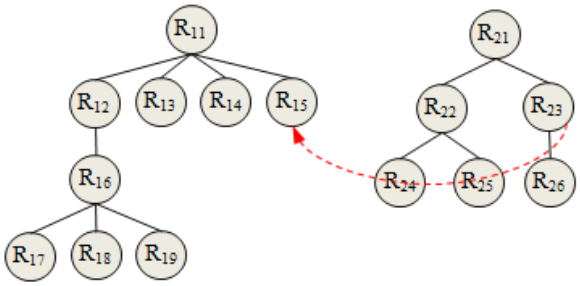


Figure 1. The Role Hierarchies of Two Health Care Systems

Figure 1 shows the role hierarchies of two health care units. In Figure 1 (a), the role hierarchy of a hospital (only showing a partial set of roles) and the names and descriptions of the roles are given in Table 1. Figure 1 (b) shows the role hierarchy of a long-term care nursing home (LCNH). The roles, names and descriptions of the roles in the LCNH are given in Table 2.

Table 1. The Roles and their Descriptions in the Hospital Case

Role	Role Name	Part of role description
R_{11}	Hospital President	The president of a hospital. In charge of doctors, nurses, pharmacists, administration of the hospital
R_{12}	Surgical Dept Head	The head of the surgical department. In charge of surgeons who perform surgeries, nurses, surgical facilities
R_{13}	OB/Gyn Dept Head	The head of the obstetricians and gynecologists. In charge of the doctors, nurses, and facilities
R_{14}	Pharmacist Dept Head	Head of the pharmacists In charge of pharmacists, prescriptions, drugs
R_{15}	Nursing Director	Head of the nurses In charge of nurses, nurse hospitalized patients, assist doctors
R_{16}	Surgeon	Doctor who performs surgical operations
R_{17}	Medical Intern	Medical_intern is an advanced student or graduate in medicine gaining supervised practical experience (`houseman' is a British term)
R_{18}	Surgical Nurse	Nurse, specialized to take care of patients after surgery, knowledgeable in cleaning wound, removing surgical

		threads, IV and other injection, simple medical checks, such as blood pressure, body temperature.
R_{19}	Medical Technicians	Operate special medical devices, perform medical tests, and interpret the results. Examples include radiology tech, cardiology tech

Table 2. The Roles and their Descriptions in the Long-Term Care Case

Role	Role Name	Part of role description
R_{21}	LCNH Director	The director of the long-term care nursing home. In charge of nurses and facilities in the nursing home, administrations, correspondence with medical director for health and care advices.
R_{22}	Medical Director	The doctor provides long-term care advices, is involved at all levels of care and supervision for the individual patients in the LCNH.
R_{23}	Nursing Director	Head of the nurses. Manage nurses for all long-term care patients.
R_{24}	Visiting Doctor	The doctor that provides regular visits to the LCNH, assessing health conditions of patients.
R_{25}	Physical Therapist	Develop physical therapy programs for individual patients, such as exercises, equipment assisted exercises, massage, evaluate patient progress in physical conditions.
R_{26}	Nurse	Take care of patients, perform needed care, injections, simple medical checks, such as blood pressure, body temperature.

Our goal is to perform semi-automated role mappings for the two role hierarchies. More specifically, we plan to recommend role mappings based on similarities between pairs of roles. For example, role R_{23}, which is a nursing director role in a long-term care nursing home can very likely be mapped to R_{15}, the nursing director in the hospital role hierarchy.

In RBAC, permissions are to be assigned to roles to grant their accesses to various resources. There may be a large number of permissions in a real system. In Table 3, we list a few example permissions for the hospital, including the permission IDs and the functional descriptions of the permissions.

Table 3. The Permissions List

Permission ID	Function description
P_1	Record the nursing care information of patients
P_2	View medicine information of patients
P_3	Need to prepare and dispense drugs
P_4	View diagnoses of patient
P_5	Responsibility for curing the patient in the ear
P_6	Check the operation information of the patient

The example permissions given in Table 3 are further assigned to the roles in hospital role hierarchy. The president of a hospital R_{11} owes permission $\{P_1\}$ and the role R_{14}, R_{15} and R_{18} owes permissions respectively $\{P_2\}$, $\{P_3, P_4, P_5, \}$,$\{P_6\}$.

3. Semi-Automated Role Mapping Process

3.1.The Role Mapping Manager

We consider having a role mapping manager (RMM) in each domain to manage role mapping related tasks. RMM includes a role mapping recommender (RMR), a role mapping approver (RMA), a role mapping management unit (RMMU), a role mapping approval policy (RMAP), and a role mapping activation manager (RMAM).The architecture of RMM is shown in Figure 2.

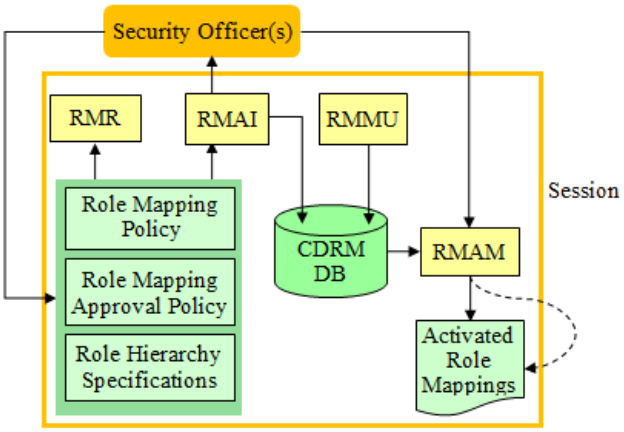


Figure 2. Architecture of the Role Mapping Manager

RMR. In a collaborative task, the roles of domain A that need to access some resources in domain B need to be mapped to domain B. To ease the task, the role mapping recommender (RMR) automatically generates potential cross-domain mappings for the involved roles and prepares them for approval. Since role mappings involve multiple domains, the RMRs of the involved domains need to work together to generate the role mappings. In Section IV, we discuss the techniques for generating role mapping recommendations.

RMAI. To assure security, each role mapping generated by RMR, say (r_x^A, r_y^B), which maps a role in a domain A to a role in domain B, needs to be manually approved by the SOs of domain B. Some mappings may only require the approval by one SO and more critical mappings may require the approval by additional SOs. The approval policy specifies such approval requirements. Once a mapping (r_x^A, r_y^B) is approved, it is sent to the RMMU and stored in the cross-domain role mapping database (CDRMDB) with an approval signature.

RMMU. RMMU manages the cross-domain role mappings in its database. The role mappings approved manually are stored in the regular database while the role mappings approved automatically are stored in a tagged database. Other entities in the system can retrieve the role mappings through RMMU. RMMU also provides tools and interfaces for the security officers to define role mappings manually and to modify and/or remove cross-domain role mappings in the database.

RMAM. The cross-domain role mappings stored in the regular database of the RMMU are deactivated by default. When a specific collaborative task (*e.g.*, the execution of a workflow, *etc.*) occurs, RMAM activates a set of cross-domain role mappings and resolve the potential conflicts among them to facilitate the execution and proper data accesses of the task.

3.2. The Role Mapping Process

With the RMM architecture, the semi-automated role mapping process for generating cross-domain role mappings from domain A to domain B can be described as follows:

1) The RMR in domain Aidentifies the roles in domain A that may need to access some data and other resources in domain B in potential collaborative tasks. It then sends a role mapping request with all the identified roles to the RMR in domain$B$.
2) Upon receiving the role mapping request, for each role r_i^A in the request, the RMR in domain B searches its role hierarchy for a similar role to it, say r_j^B. In this step, RMR may perform semantic similarity analysis between r_i^A and r_j^B and trust analysis of domain A and the involved entities in domain A. The RMR also validates the potential mapping (r_i^A, r_j^B) against its local role mapping policies to decide whether the role mapping is acceptable.
3) The RMR in domain B, after generated the list of recommended cross-domain role mappings, passes them to the RMAP for approval. RMAP notify the security officers regarding the approval request. The SO (or SOs) of domain *B* goes through the role mappings using RMAP's GUI tools to view these mappings and makes the approval decisions.

a. The SO may approve the role mapping (r_i^A, r_j^B).In this case, (r_i^A, r_j^B) is sent to RMMU and stored in the CDRMDB.
b. In case a role mapping (r_i^A, r_j^B)may create security concerns, the SO may first try to identify a different role in domain B, $r_{j'}^B$, such that the mapping of $(r_i^A, r_{j'}^B)$ can be approved. In case the SO in domain B has to reject (r_i^A, r_j^B) and cannot find an appropriate mapping for r_i^A, the SO may send r_i^A back to the RMR in domain A. The RMR will alert the SO in domain A about the failed maipping. The SOs in domains A and B may manually negotiate based on the collaborative task needs to determine a suitable mapping for r_i^A.

4) The list of approved mappings is returned to domain A.

3.3. On-the-Fly Role Mapping

Sometimes, some collaborative tasks may take place under an emergency situation and the cross-domain role mappings for the involved domains may not be available or some specific role mappings may be missing and it is necessary to dynamically create them. In this case, an automated approval may be necessary to allow on-time completion of urgent tasks. RMAP defines the conditions and constraints such automated approval are allowed. Also, such mapping was for coping with urgent need to access some specific information, thus, the least privilege will be granted to facilitate the successful completion of the task. After approval in emergency, RMAP sends the approved role mappings to RMMU and the RMMU stores them in a tagged database. Since defining and using role mappings without manual approval can be dangerous, the SOs need to perform further analysis to ensure no damage done by these mappings and then formally approve them manually.

4. Automated Role Mapping Analysis and Recommendation

We use semantic technologies to find similar roles in different domains and use the information to recommend potential role mappings between two domains.

In order to facilitate the semantic similarity analysis among roles, we first define the role model in a relatively formal way, *i.e.*, we define an OWL model to specify the roles

and the role hierarchy and the corresponding information. Thus, a uniform way for specifying roles and role hierarchy can be used as the basis for further semantic analysis. The OWL-based role model is discussed in Section II.A. From the OWL model, we extract information to build the concept set for each role based on its role description, role property and role responsibility. The method for concept extraction is given in Section II.B. In Section II.C, we define similarities between roles based on the concepts extracted for the corresponding roles and the role hierarchy. We define a similarity metric and the corresponding algorithm role similarity computation.

4.1. OWL-Based Role Specification Model

OWL is based on description logics [11] and can be used in defining classes, properties and conceptual relationships of objects *etc*. Description logics are a family of logics that are decidable fragments of first-order predicate logic. These logics focus on describing classes and roles, and have a set-theoretic semantics. OWL is a specific language based on description logics [11].We extend OWL model to support the specification of RBAC and the related information. The extended model is given in Figure 3.

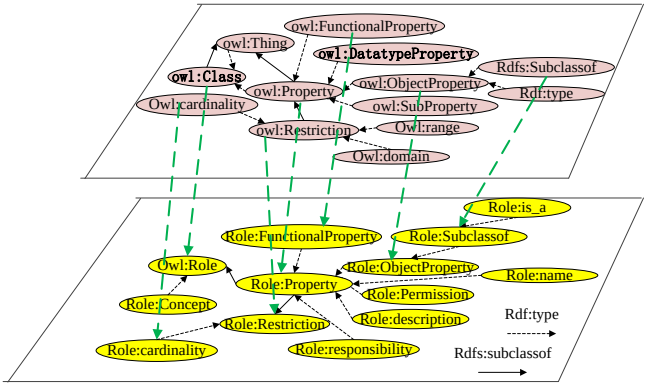


Figure 3. The Extended OWL

To clarify the instantiation and extension, at the top of Figure 3, the original OWL ontology is given. The node <owl:Role> is a class that defines the role itself. The role hierarchy is defined by the "SubclassOf" node which is an instantiation from the OWL ontology. If a role y is a child node of role x in the role hierarchy, then y "is_a" "SubclassOf" specified by "ObjectProperty" in the "Property" specification of role class x. Under "Property" specification, there are also the extended classes "Name", "Description", and "Responsibility", which specify the role name, the description about the role, and the responsibilities of the role, respectively. Another "Property" node "Permission" specifies all the permissions granted to the role that are in addition to its children's permissions. The "concept" class provides the concepts that can be used to define/describe the role, which is extracted from the "Properties" of the role. The "cardinality" is a "Restriction" specified for role instantiation. For example, it can be used to specify the constraint on the number of users that can be mapped to the role or the number of sessions that can be activated for the users with this role.

We will use the information in this OWL-based role ontology as input to perform concept extraction, similarity derivation, *etc.* To simplify the discussion in the subsequent sections, we define some notations to represent the data provided in the role ontology. For a role r, let $r.n$,$r.desc$, and $r.resp$ denote the name, the description document, and responsibility specification document of the role, respectively. Also, let $r.C$ denote the concept set of r, and $r.C = \{r.c_i\}$, where $r.c_i$, for all i, is an individual concept in the set (could be a keyword or a phrase). Finally, let $r.P$ denote the set of permissions

assigned to r that is in addition to the permissions of r's children's, and $r.P = \{r.p_i\}$, where $r.p_i$ is an individual permission.

4.2. Concept Extraction

We build the concept set of each role from the role description, role responsibilities, and property in the OWL based role model [12]. Various concept extraction methods have been introduced in the literature [13]. Most of these methods are statistical based and require a significant amount of input data in order to extract special features such as paragraph features, thematic word features, uppercase word features, *etc.*

We develop a simple matrix-based analysis method for concept extraction from various descriptions related to each role in the OWL-based role model. The concept extraction process is discussed in the following.

First, we extract nouns from role name, role description, role responsibility, and permissions ($r.n$, $r.desc$, $r.resp$ and $r.P$) and record their appearance frequencies using the Stanford Log-linear Part-Of-Speech Tagger [ref].Let $T = \{t_1, t_2, \dots, t_n\}$ denote the set of n nouns obtained and $F = [f_{ij}], 1 \le i \le m, 1 \le i \le n,$ denote the $m \times n$frequency matrix, where f_{ij} is the frequency of the word t_j appearing in the i-th document. Here first, second, and third documents are $r.n$, $r.desc$, and $r.resp$ and the remaining documents are the descriptions for $r.p_k$, for all k. We also define $f_j = \sum_i f_{ij}$.

To ensure proper extraction of relevant concepts, we eliminate nouns that do not reach a frequency requirement. Here we require that a noun is considered as the key concept for the role only if it appears in at least f_{min}documents. In other words, we can compute

$$f'_{ij} = \begin{cases} 1, \text{ if } f_{ij} \ge 1 \\ 0, \text{ if } f_{ij} = 0 \end{cases}$$

and $f'_j = \sum_i f'_{ij}$. If $f'_j > f_{min}$then t_j is copied form T into the keyword list T'. We also rebuild F' from F based on the new keyword set T' by removing all the columns correspond to nouns t_j with $f'_j \le f_{min}$. Now we compute the weight of keyword $t_j \in T'$.

$$Weight_{t_j} = \frac{1}{m} \times f'_j \times f_j$$

Finally, we define the concept set $r.C$. We sort $t_j \in T'$ by the weight of t_j. If $t_j \in T'$ is among the top of the sorted list T', then t_j is included in $r.C$. Here we consider four keywords in each concept set and, hence, t_j is included in $r.C$ if it has the top four weight in T'.

We applied the approach and derived the concept sets for the roles in the hospital role hierarchy and the long-term care nursing home role hierarchy, and parts of the results are given in Table 4 and Table 5.

Table 4. The Concept List in the Hospital Case

Role	Concept
R_{11}	Hospital President, hospital administrator
R_{12}	Surgical Dept Head, surgeons, doctor, director
R_{13}	Nursing Director, nurse
…	…..

Table 5. The Concept List in the Long-Term Care Case

Role	Concept
R_{21}	LCNH Director, administrator
R_{22}	Medical Director, doctor
R_{23}	Nursing Director, patient care, nurse
…	…

4.3. Similarity Between Roles

We use two independent metrics to define the similarities between roles. The first metric defines the similarity between the concept sets of the roles. Concept is a group of semantically equivalent or very similar words. The concept of the roles provides the general level of agreement in the use of words for describing the roles and detects equivalent words that are likely to have been used to refer to similar roles in different domains. Similarities in the concept of two roles can be used as a basic similarity assessment approach to detect equivalent or synonymous roles, which may be an inconclusive form of similarity assessment because roles in two domains may be very different in terms of their role names, descriptions and responsibilities. Note that the role hierarchies also carry important information which can be used to help with similarity assessment. Thus, our second similarity metric attempts to incorporate semantics into the similarity measure by using distinguishing features as another indicator of how similar roles are.

(1) *Measuring similarity of two roles based on their concept sets.* We calculate similarity of concept sets of two roles based on the lexicon that is WordNet. The organizational structure of the words in the WordNet is expressed by the use of the vocabulary matrix model. As shown in Table 6.

Table 6. The Vocabulary Matrix Model of Wordnet

Word meaning	Morphology				
	F1	F2	F3	…	Fn
M1	E(1,1)	E(1,2)			
M2		E(2,2)			
M3			E(3,3)		
. . .	...			...	. . .
Mm					E(m,n)

In Table 6, each column represents the morphology and each row represents the meaning of a word. For example, morphology of the element E(1,1) in the table is F1 and its word meaning is M1. If there are two elements in the same column, this word (with the specific morphology) has two meanings and it is a polysemy. If there are two elements in the same row, this word has two morphologies and the word is a synonym.

According to the above structure of WordNet, we can compare the individual terms of the concept sets of two roles with the synonym in the WordNet as well as string matching. If two concept terms of two roles are synonyms (in the same column in the WordNet vocabulary matrix), we consider them to be the same. Accordingly, we define the similarity of the terms of the concept sets of two roles as follows.

$$simC\left(r_i^A.c_{i_1}, r_j^B.c_{j_1}\right) = \begin{cases} 1, & \text{if } r_i^A.c_{i_1} \text{ and } r_j^B.c_{j_1} \text{ are the same} \\ 0, & \text{otherwise} \end{cases}$$

Here, $r_i^A.c_{i_1}$ is the i_1-th term in r_i^A's concept set and $r_j^B.c_{j_1}$ is the j_1-th term in r_j^B's concept set. Based on the similarity definition of individual pairs of terms, we define the

similarity of two roles r_i^A and r_j^B.

$$SimR(r_i^A, r_j^B) = \frac{\sum_{c_1 \in r_i^A} \sum_{c_2 \in r_j^B} simC(c_1, c_2)}{|r_i^A.C| \times |r_j^B.C|}$$

Recall that $r_i^A.C$ and $r_j^B.C$ are the concept sets of the roles r_i^A and r_j^B, respectively. $|r_i^A.C|$ and $|r_j^B.C|$ are the total number of words in the concept sets of roles r_i^A and r_j^B, respectively.

(2)Measuring similarity of two roles based on therole hierarchy.Conceptually, the role hierarchies should also contribute to the definition of the similarity of two roles from two different domains. We consider not only the basic similarity of two roles alone, but also consider the similarities of its parents and children (or similarities of the ancestors and descendants of the roles). We use a weighted sum to express the cumulative similarities of the roles themselves and among their ancestors and descendants.

Here we consider the role-hierarchy-based similarity for a pair of roles by also factor in the impact of two layers up and two layers down the role hierarchy. Let $r.Parents$ and $r.Chidren$ denote the set of parents and the set of children of role r. Also, let $r.GrandP$ and $r.GrandC$ denote the set of grandparents and the set of grandchildren of role r, *i.e.*,

$r.GrandP = \{x, \forall x, y, x \in y.Parents \text{ and } y \in r.Parents\}$
$r.GrandC = \{x, \forall x, y, x \in y.Children \text{ and } y \in r.Children\}$

Now we define the similarity of two roles based on the role hierarchy as follows.

$$Sim(r_i^A, r_j^B) = w'^2 \times SGP + w' \times SP + w \times SimR(r_i^A, r_j^B) + w' \times SC + w'^2 \times SGC$$

$$SP = \begin{cases} 1, \text{ if } r_i^A.\text{Parent} = \emptyset \text{ and } r_j^B.\text{Parent} = \emptyset \\ 0, \text{ if } r_i^A.\text{Parent} = \emptyset \text{ and } r_j^B.\text{Parent} \neq \emptyset \\ 0, \text{ if } r_i^A.\text{Parent} \neq \emptyset \text{ and } r_j^B.\text{Parent} = \emptyset \\ \dfrac{\sum_{u \in r_i^A.\text{Parent}, v \in r_j^B.\text{Parent}} SimR(u,v)}{|r_i^A.\text{Parent}| \times |r_j^B.\text{Parent}|}, \text{ otherwise} \end{cases}$$

$$SGP = \begin{cases} 1, \text{ if } r_i^A.\text{GrandP} = \emptyset \text{ and } r_j^B.\text{GrandP} = \emptyset \\ 0, \text{ if } r_i^A.\text{GrandP} = \emptyset \text{ and } r_j^B.\text{GrandP} \neq \emptyset \\ 0, \text{ if } r_i^A.\text{GrandP} \neq \emptyset \text{ and } r_j^B.\text{GrandP} = \emptyset \\ \dfrac{\sum_{u \in r_i^A.\text{GrandP}, v \in r_j^B.\text{GrandP}} SimR(u,v)}{|r_i^A.\text{GrandP}| \times |r_j^B.\text{GrandP}|}, \text{ otherwise} \end{cases}$$

$$SC = \begin{cases} 1, \text{ if } r_i^A.\text{Children} = \emptyset \text{ and } r_j^B.\text{Children} = \emptyset \\ 0, \text{ if } r_i^A.\text{Children} = \emptyset \text{ and } r_j^B.\text{Children} \neq \emptyset \\ 0, \text{ if } r_i^A.\text{Children} \neq \emptyset \text{ and } r_j^B.\text{Children} = \emptyset \\ \dfrac{\sum_{u \in r_i^A.\text{Children}, v \in r_j^B.\text{Children}} SimR(u,v)}{|r_i^A.\text{Children}| \times |r_j^B.\text{Children}|}, \text{ otherwise} \end{cases}$$

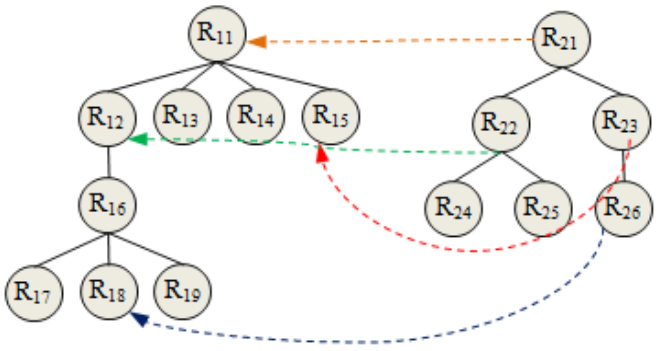
$$SGC = \begin{cases} 1, \text{ if } r_i^A.\text{GrandC} = \emptyset \text{ and } r_j^B.\text{GrandC} = \emptyset \\ 0, \text{ if } r_i^A.\text{GrandC} = \emptyset \text{ and } r_j^B.\text{GrandC} \neq \emptyset \\ 0, \text{ if } r_i^A.\text{GrandC} \neq \emptyset \text{ and } r_j^B.\text{GrandC} = \emptyset \\ \dfrac{\sum_{u \in r_i^A\text{GrandC}, v \in r_j^B.\text{GrandC}} SimR(u,v)}{|r_i^A.\text{GrandC}| \times |r_j^B.\text{GrandC}|}, \text{ otherwise} \end{cases}$$

The derivations for SGP, SC, SCP are similar to that of SP. Here,w is the weight for the basic similarity of the role itself, and $2w'^2 + 2w' + w = 1$, . For the roles that are further away from the current roles r_i^A and r_j^B (*i.e.*, their grandparents and grandchildren), the weight w'^2 is used in the weighted sum of similarity to show a reduced impact.

For example, we give two role hierarchies in the Section II. There is a role R_{23}in a long-term care nursing home, which is a nursing director role can very likely be mapped

to R_{15}, the nursing director in the hospital role hierarchy. Analysis in detail is as follows.

We can obtain the concept sets of role $R_{23}$and $R_{15}$by the method of extraction concept from role name, role description, role responsibility, and permissions. The concept set of $R_{23}$is (Nursing Director, patient care, nurse) and the concept set of R_{15} is (Nursing Director, nurse). We need to input two concept sets into WordNet, and then obtain sense number and if more than one sense number[*†], we judgment whether they are equal or not. The file data.noun of WoreNet gives the same sense number is 10212056. We can find 10212056(nurse, nursing director) in the our file offset. Then we get similarity of concept sets is $SimR(R_{23},R_{15}) = 0.80$. In the same way, we can get the similarity of grandparents, parents, children and grandchildren of two roles R_{15},R_{23} are respectively $SGP = 1$, $SP = 0.50$, $SC = 0$, $SGC = 1$. Therefore, the similarity of two roles $Sim(R_{25},R_{15}) = 2 \times 0.1 + 0.2 \times 0.50 + 0.4 \times 0.80 = 0.62$. So, the role of R_{23} is mapped to the role R_{15}. According to the above analysis, we can give the entire role mapping of two role hierarchies. As shown in Figure 4.Two roles R_{21},R_{11}'s similarity of concept sets is $SimR(R_{21},R_{11}) = 0.75$, and the similarity of two roles $Sim(R_{21},R_{11}) = 0.76.$ Therefore, the role of R_{21} is mapped to the role R_{11}. Two roles R_{22},R_{12}'s similarity of concept sets is $SimR(R_{22},R_{12}) = 0.50$, and the similarity of two roles $Sim(R_{22},R_{12}) = 0.60$.So, the role of R_{22} is mapped to the role R_{12}.Two roles R_{26},R_{19}'s similarity of concept sets is $SimR(R_{26},R_{18}) = 0.67$, and the similarity of two roles $Sim(R_{26},R_{18}) = 0.68.$ Therefore, the role of R_{26} is mapped to the role R_{18}.



(a) (b)

Figure 4. The Result of Two Role Hierarchies Mapping

4.4. Role Mapping Recommendation

The process for establishing role mappings from domain A to domain B can start from mapping the root role(s) of domain A. Let rr^A denote the root role in domain A's role hierarchy. We find the role r^B in domain B such that $Sim(rr^A,r^B) = Max_j\, Sim(rr^A,r_j^B)$ and put (rr^A,r_j^B) in the recommendation list. Once rr^A is mapped to r_j^B, the children of rr^A can only be mapped to the children of r_j^B. Thus, the search can proceed in the sub-hierarchies and the recommended role

[*] http://wordnetcode.princeton.edu/5papers.pdf

[†] http://wordnet.princeton.edu/wordnet/download/current-version/#win

mappings can be added progressively to the recommendation list till all the roles to be mapped in domain A have been processed. The final recommendation list is then presented to the security officer(s) for approval.

Similarity between roles may not be the only factor to be considered for role mapping. Security officers can specify various mapping policies to guild the automated mapping recommendation process. For example, consider mapping roles from domain A to domain B. The trust level of each role from domain A can be matched against the criticality of the roles in the local domain B following the mapping policy. Also, specific permissions required for the collaborative tasks can be analyzed and specified in advance to guild the mapping with least privileges. Since the methods for evaluating the additional metrics that may be used in the role mapping recommendation process is not in the scope of this paper, we do not consider it further.

4.5. Modify Role Hierarchy for Role Mapping

Sometimes, the security officer in domain B may try to find a suitable role mapping for a foreign role r_i^A with least privilege, but there is no suitable role in local role hierarchy to fulfill the mapping. Similarly, during an emergency role mapping, it is desirable to find a local role r_j^B for a foreign role r_i^A, such that the least privilege required by r_i^A to accomplish the emergency task can be granted. It is also possible that there is no exactly appropriate local role to fulfill the mapping goal. In these cases, it is desirable to create a dummy role in the local role hierarchy to best satisfy the least privilege requirement.

Assume that after the automated role mapping recommendation process, the recommended role mapping for a foreign role r_i^A to domain B is (r_i^A, r_j^B). Also, assume that the permission set required for r_i^A to access the resources in domain B is P_{AB}. The following procedure "Refine_mapping" finds the most suitable child node of r_j^B in the role hierarchy to fulfill the mapping request.

Refine_mapping (r_i^A, r_j^B):

If P_{AB} is a subset of $r_j^B.P$ then

For all $r \in r_j^B.Child$

If $P_{AB} = r.P$ then return r;

If P_{AB} is a subset of $r.P$ then

Refine_mapping (r_i^A, r);

If P_{AB} is not a subset of $r.P$ then

Add r_j^B into Candiate_list;

Endfor;

Else return "Fail to find an appropriate mapping"

Note that the procedure Refine_mapping returns the suitable mapping role when one is found in the descendants of r_j^B with the matching privilege. If starting from the beginning, r_j^B does not have enough privilege for r_i^A, then refinement is useless and the role mapping effort immediately fails. If it is neither of the above two cases, then a Candidate_list is returned. The Candidate_list should be initialized to empty set before

starting the Refine_mapping procedure. Once a Candidate_list is returned, one of the roles in the list, say r, can be selected to start the Add_role procedure. In the Add_role procedure, one dummy role dr is created with the exact privilege set P_{AB} and dr is the child node of r.

5. Conclusions

In this paper, we introduce an automated role mapping recommendation process to help generate recommended role mappings between two domains to ease the role mapping task that is to be perform by the security officers manually. The approach in this paper focuses on using the semantic similarities between roles to determine the role mappings. We plan to investigate additional metrics that are frequently considered by security officers in determining role mappings and develop techniques to evaluate these metrics to guild the automated role mapping recommendation process.

Acknowledgment

The work is supported by National Science Foundation of China (61374108).

References

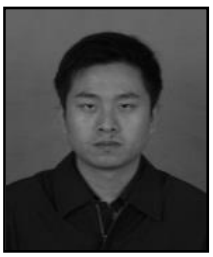
[1] B. W. Lampson, "Protection", ACM SIGOPS Operating Systems Review, vol. 8, no. 1, **(1974)**.
[2] R. S. Sandhu, E. J. Coyne, H. L. Feinstein and C. E. Youman, "Role-based access control models", IEEE Computer, vol. 29, no. 2, **(1996)**, pp. 38-47.
[3] E. Yuan and J. Tong, "Attribute based access control (ABAC) for web services", IEEE ICWS, **(2005)**, pp. 561-569.
[4] W. She, I. L. Yen, B. Thuraisingham and E. Bertino, "Role-Based Integrated Access Control and Data Provenancefor SOA Based Net-Centric Systems", SOSE, IEEE International Conference on Web Services, **(2011)**, pp. 225-234.
[5] B. Shafiq, J. B. D. Joshi, E. Bertino and A. Ghafoor, "Secure interoperation in a multidomain environment employing RBAC policies", IEEE TKDE, vol. 17, no. 11, **(2005)**, pp. 1557-1577.
[6] P. A. Bonatti, M. L. Sapino and V. S. Subrahmanian, "Merging heterogeneous security orderings", European Symposium on Research in Computer Security, **(1996)**, pp. 183-197.
[7] M. Shehab, E. Bertino and A. Ghafoor, "Secure collaboration in mediator-free environments", ACM CCS, **(2005)**, pp. 58-67.
[8] S. Dawson, S. Qian and P. Samarati, "Providing security and interoperation of heterogeneous systems", Distributed and Parallel Databases, vol. 8, **(2000)**, pp. 119-145.
[9] A. Kapadia, J. A. Muhtadi and R. Campbell, "IRBAC2000 : Secure interoperability using dynamic role translation", University of Illinois,Urbana, IL, U.S.A, **(2000)**.
[10] W. She, I. L. Yen, B. Thuraisingham and E. Bertino, "Role-Based Integrated Access Control and Data Provenancefor SOA Based Net-Centric Systems", SOSE, IEEE International Conference on Web Services, **(2011)**, pp. 225-234.
[11] S. C. Welty and D. McGuinness, "OWL Web Ontology Language Guide", W3C Recommendation, http://www.w3.org/TR/2004/REC-owl-guide-20040210/, **(2004)**.
[12] T. Nomoto and Y. Matsumoto, "A new approach to unsupervised text summarization", ACM SIGIR Conference on Research and Development in Information Retrieval, **(2001)**, pp. 26-34.
[13] K. Julian, J. O. Pedersen and F. A. Chen, "Trainable Document Summari", ACM SIGIR Conf. on Research and Development in Information Retrieval, Seattle, WA, **(1995)**, pp. 68-73.

Authors

Feng Wang, is currently the Associate Professor of School of Mathematics and Statistics Science, Ludong University. He has contributed over 9 papers and 3 of them indexed by EI, such as Semantic computing, Ontology learning, *etc.* His current research interests include Semantic computing, Intelligent algorithm

Lei Cui, is currently the Lecturer of Information Engineering Department, YanTai Vocational College. She has published several articles in the professional journals,such as Electronic communication, Network Security.Her current research interest include Trust Manager,Computational algorithm

Yi Zheng Wang, studies at Tilburg University. She has published 3 articles in the professional journal of indexed by EI.Her research interests include Intelligent algorithm, information economy.



Xinjiang Wei, is currently the Professor of School of Mathematics and Statistics Science, Ludong University. He has contributed over 17 articles to professional journals, such as Security and Communication Netwroks, Journal on Communications, *etc.* His current research interests include Intelligent algorithm.