User-Credential Based Role Mapping in Multi-domain Environment

Ajith Kamath  
akamath@site.uottawa.ca  
School of Information Technology and Engineering  
University of Ottawa, Canada

Ramiro Liscano  
rliscano@ieee.org  
Fac. of Eng. and Applied Science  
Un. of Ontario Institute of Technology, Canada

Abdulmotaleb El Saddik  
elsaddik@site.uottawa.ca  
School of Information Technology and Engineering  
University of Ottawa, Canada

Abstract

Collaboration between multiple organizations creates new opportunities for business. With such collaborations becoming a reality, it is necessary to have access control policy integration approach to form a global policy consistent with the partner organizations. Research on policy integration has led to the proposal of several frameworks to uniformly express policies and to integrate such policies. But most of these frameworks are complex, and compromise on the privacy of the constituent domains by sharing all the components of access control policy including Access Control Lists. In this paper, we describe a unique policy integration technique to merge Role Based Access Control (RBAC) policies of multiple domains in heterogeneous environment. The proposed mechanism uses user-credential associated with roles as the main criteria in mapping inter-domain roles. We also show the results of initial prototype of system. Based on the current positive experience, we are investigating on conflict resolution technique that guarantees consistency of the composed policy across all participating domains.

Keywords: Policy Integration, Role Based Access Control (RBAC), Community Authorization Service (CAS).

1. Introduction

The operating environment of organizations has changed drastically during the last decade. Increased connectivity has facilitated in distributed multiple organization interoperation. In particular, Web Services and Grid based systems are emerging systems in multi-domain environments. However, with this increase in information sharing, there is a growing concern on security in such a diverse environment. Different organizations forming independent domains not only aim for sharing the resources but also for secure access to the resources shared. It is self evident that such collaborations require a security framework which should entail formation, administration, and enforcement of a global security policy, which is consistent with the policies of all the participating domains.

Many research and industrial groups have contributed to security in grid systems, where all the institutes coordinate resource sharing and problem solving in dynamic virtual organizations [5]. One of major work on security framework in grid systems is Community Authorization Service (CAS) [6]. CAS [6] is developed to allow community to flexibly and expressibly authorize access to resources in a distributed grid system. In particular, in a CAS system, major emphasis is given on the enforcement of global security policy. There are many proposals [11, 12] which extends CAS especially in maintenance or enforcement of policy. However, most of these proposed approaches pay no attention to the formation of the global security policy.

Policy composition that is consistent with the participating autonomous domain policies is a challenging task. It is essential to have a uniform representation of the policies in such an environment. Recent years have witnessed considerable work on RBAC models. RBAC models can be effectively used for uniform representation of the security policies of autonomous domains. As a result, few approaches have been proposed to integrate RBAC policies to form a single global RBAC policy [3, 8]. The proposed models are more suitable for environments which have homogeneous policies and cannot be used to capture the heterogeneity requirements of emerging application.

We propose a unique policy integration technique for heterogeneous environment that ensures secure interoperation between multiple organizations. The proposed mechanism represents a fundamental shift in the traditional paradigm of comparing and merging RBAC policies based upon role privileges. XML based role
Related work

In this section, we begin by briefly reviewing CAS model. In the later part of the section, we will examine some of the policy integration mechanisms in RBAC model.

The Community Authorization Service (CAS) [6, 7] is a software, integrated with Globus Toolkit [13], to allow the creation of virtual organization (VO). CAS is built on Grid Security Infrastructure (GSI) [10], which is a set of libraries and tools that focuses on authentication, message protection, single sign-on algorithms, and delegation mechanisms. VO allows administrators to manage access control in a large distributed grid network. A central CAS server will be responsible for managing the policies governing access to resources in the community. To enable this infrastructure, initially resource providers in the community delegate subset of their policy space to the VO. The VO maintains a global policy: once users are authorized, certificates are issued containing the credentials. Users request access to the resources by presenting the certificates. The resources enforce VO policies carried as credentials along with user requests. However, the existing system or the proposed models in grid security system do not address the problem of global policy creation technique at the VO.

Policy Integration is the mechanism by which a global security policy is formed which is consistent with participant domains. To ease the process of integration, access control implementations at the participating domains needs to be standardized. Thus, for policy integration, we need a formal access control system that allows us to capture the set of controls in each domain. In this respect, RBAC [1] has emerged as a powerful model, as it supports uniform representation of security policies. However, relatively less work has been done in RBAC policy integration and most of the work assumes the basic notion of matching permission sets for mapping the roles. Rest of this section, summarizes related work in the field of RBAC policy integration models.

Shafiq et al. [3] defined a set of relations between inter-domain roles for the composition of a consistent RBAC policy. The roles are mapped using the permission set associated with the corresponding roles. Such composition of access control policy depends on homogeneity of permission sets attached to mapping roles and therefore the integration technique cannot be optimal in a heterogeneous environment. Another approach for secure interoperation between multiple organizations is SERAT: Secure Role Mapping Technique [8]. This paper proposes a novel approach of access paths for secure interoperation. An access path is defined as a sequence in which user acquires roles in multiple domains for a particular session [8]. But, initial interoperability in SERAT is enabled by creating cross links which interconnect domains. Cross links are either explicitly permitted or restricted links, decided by the collaborating domain administrators. Cross link formation may not be an optimal solution as it involves a lot of factors such as policy complexity, privacy considerations, and administrator expertise.

Policy integration of multiple domains is also influenced by privacy policies of the constituent organizations. Many of the proposed integration models [3, 8, and 9] require that the collaborating organizations share their access control list/permission sets before integrating the policies. This could have major implications on the privacy policy of the organizations because it is essential that an organization adheres to its privacy policy during collaboration. We also envision that the integrated policy be flexible enough so that the central authorization server be able to assign the user explicitly to the roles as in the CAS but such assignments should be in accordance with the participating domains. Such integration will support change in user list at the center without affecting the collaborating domains, as opposed to each role having set of predefined member list.

3. User-credential based role mapping for policy integration

We assume that all the constituent domains adopt RBAC model [1] to ease the integration process. RBAC is a preferred model because it reduces policy complexity and is suitable for any organizational security policy. Traditionally in RBAC, users are assigned to one or more roles, where role usually denotes a job responsibility in the organization.
Each role in RBAC is associated with a permission set, i.e., access to information. Thus, RBAC maps itself well to the organizational structure. In the rest of the Section, we describe the RBAC policy representation in multi-domain environment and our proposed policy integration technique.

3.1. X-RBAC overview

X-RBAC is an XML-based RBAC language for modeling RBAC elements mainly ‘user’, ‘role’ and ‘permission’ and the relationship associated between these elements [2]. XML schemas are defined to capture each component of the RBAC. Thus, information about users, roles and permissions are available from the corresponding XML sheets - XML User Sheet (XUS), XML Role Sheet (XRS) and XML Permission Sheet (XPS). The assignments between these elements are captured using XML User to Role Assignment Sheet (XURAS) and XML Permission to Role Assignment Sheet (XPRAS). In our proposed integration model, we use XRS and XURAS as the input to the system. XRS specifies the attributes of the roles where as XURAS specifies the user credentials required to be assigned to a role. If no explicit user-credentials are specified for a role, then any user can take up the role which could be a guest role. For readers who need more detail on X-RBAC can refer to Ref [2].

3.2. Policy integration technique

The proposed algorithm to integrate the constituent local RBAC policies is presented in this Section. We first discuss the sharing of local security policies. We envision a generic information sharing model as illustrated in Figure 1. The domain which intends to share its resources must export its local X-RBAC policy components to the trusted third party. As a result, the global policy becomes populated due to the collaboration between multiple domains. The trusted third party could be another autonomous domain that provides a service of integration and maintenance of global policy for different collaborations.

In the context of RBAC, the integration of local policies would mean mapping between the roles in different domains. Our proposed policy integration mechanism governs mapping roles based upon the user-credentials associated with them. User-credential is a set of attribute-value (AV) pairs that need to be satisfied by the user to be assigned to the corresponding role. For instance, the set of AV pairs associated with a Doctor role might include (qualification = Doctor) AND ((level = 4) OR (age > 35)). Logical expressions such as AND, OR can be used between AV pairs to define a complete credential set.

The main challenges in user-credential mapping are:
- **Attribute naming conflicts**: They generally arise in heterogeneous environments as participating domains construct local policies independent of each other.
- **Conflicts in user-credential expressions**: These conflicts arise when level and sequence of AV pairs forming the user-credential differs from one another.

Naming conflicts can be tackled using the schema integration technique [4]. If the conflicts cannot be resolved, then the system requests input from the administrator. Here, we can argue that the level of heterogeneity in user-credential is minimum as compared to any other component of the RBAC policy, including permission sets associated with the roles. This is because the service collaborations occur for a particular
community or a group of users. As roles usually represent organizational responsibilities, integrating the policies at the role level is difficult. However, roles are associated with permission sets and user-credentials. Permission set is a collection of object and permission pairs. Resolving conflicts between permission sets during policy integration is also difficult as objects/permissions in each organization can be totally different. For example, a graduate student (user) can act as intern (role) in an enterprise which has limited privileges (permission set). However, the same graduate student may have a student role having more privileges in the university domain. When both the university and the enterprise domains need to collaborate for a partnership research project, it is easy to capture the equivalency between the student role and the intern role based upon their user-credentials i.e., the attributes, the user is required to have to take up the role.

Conflicts in expressions can be resolved using regular expressions. Regular expression is used to describe complex user-credentials in a compact way. Each user-credential is assigned a unique literal and is grouped together to form a regular expression representing a user-credential set associated with a RBAC role. For instance, a Doctor role having a user-credential set (qualification = Doctor) AND ((level = 4) OR (age > 35)) is represented as `(a(b|c))`, where literal ‘a’ represents (qualification = Doctor), ‘b’ represents (level = 4) and ‘c’ represents (age > 35). The alternation operator, the vertical bar (“|”), represents logical function ‘OR’. Once represented, the user-credential set can be easily searched for patterns to match the roles.

Table 1 depicts the formal steps of the proposed algorithm. The algorithm takes XRS and XURAS of local policies as input and generates XRS and XURAS for the integrated policy. It should be noted here that other components of RBAC such as XPS, XPRAS are not generated as they are not needed for the functioning of the trusted third party to assign and authorize the users to the roles. Once integrated roles are created at the trusted third party and the participating domains delegate their local roles to the integrated RBAC policy roles.

The algorithm integrates two policies in each iteration. As shown in the Table 1, for mapping the inter-domain roles, each condition/operator in user-credential set is mapped. Such mappings are done by generating a synonym set for attribute, value and operator. Semantic coherence between such synonym set is determined and evaluated. If a decision cannot be taken, then an administrator input is requested. Once the operators are mapped, regular expressions are generated to express the whole set of user-credentials. These regular expressions are evaluated against each other to map the inter-domain roles.

Based on this concept, we can have four different types of relations between any two cross domain roles R_A and R_B.

1. **Equivalent:** Role R_A and R_B are said to be equivalent if the user-credential set for the user list of R_A is equivalent to user-credential set for the user list of R_B. Both these roles shall be merged into one single role in the integrated policy.

<table>
<thead>
<tr>
<th>Algorithm: policy-integrate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input: Policy_1 to Policy_N components (XRS &amp; XURAS)</td>
</tr>
<tr>
<td>Output: Policy_INTEGRATE component (XRS &amp; XURAS)</td>
</tr>
<tr>
<td>1: for i=1 to N do</td>
</tr>
<tr>
<td>2: parse XRS_i and XURAS_i</td>
</tr>
<tr>
<td>3: call policy-map(Policy_INTEGRATE, Policy_i)</td>
</tr>
<tr>
<td>4: end if</td>
</tr>
<tr>
<td>5: generate XRS &amp; XURAS from Policy_INTEGRATE</td>
</tr>
<tr>
<td>6: return XRS &amp; XURAS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm: policy-map</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input: Policy_1 and Policy_2</td>
</tr>
<tr>
<td>Output: Policy_x</td>
</tr>
<tr>
<td>1: for each unmapped role1 € Policy_1 and each unmapped role2 € Policy_2 do</td>
</tr>
<tr>
<td>2: for each unmapped user1 € role1 and unmapped user2 € role2 do</td>
</tr>
<tr>
<td>3: Map all op1 € user-credential1 and all op2 € user-credential2 using synonym comparison approach</td>
</tr>
<tr>
<td>4: if operator synonyms are coherent but no complete match do</td>
</tr>
<tr>
<td>5: request user input</td>
</tr>
<tr>
<td>6: end if</td>
</tr>
</tbody>
</table>

/*To generate regular expression for user credential*/
7: switch op1 and op2 do
8: case equivalent:
9: assign same literal to both operators
10: case op1 contains op2:
11: assign op2 any literal – 'x' and op1 ‘(x|y)’
12: case op2 contains op1:
13: assign op1 any literal - ‘x’ and op2 ‘(x|y)’
14: case op1 intersects op2:
15: assign op1 ‘(x|y)’ literals and op2 ‘(y|z)’
16: case not-related:
17: assign different literals to operators
18: end switch
19: generate regular expressions re1 and re2 from user-credential1 and user-credential2 using assigned operator literals
20: Map user1 and user2 based on regular expression matching
21: end for
22: Map role1 and role2 based on each users mapping between them
23: end for
2. Contain: Role R_A is said to contain role R_B, if the user-credential set for the user list of R_B is included in the user-credential set for the user list of R_A. In this case, R_B will have a junior role to R_A in the integrated policy.

3. Intersect: Role R_A and R_B said to intersect if some of the user-credential set for the user list of R_A is equivalent to some of the user-credential set for user list of R_B. In the integrated policy, new role R_C is created which is junior to both the roles R_A and R_B and is associated with subset of matching user-credentials of both the senior roles.

4. Non-Related: Role R_A and R_B are said to be non-related if the user-credential set for any user of R_A is not matching to user-credential set for any user of R_B. No mapping is defined between R_A and R_B in the integrated policy.

It is worthy to mention here that such merging of RBAC policies might create conflicts in the global policy. Such conflicts may arise due to role hierarchy and separation of duties between roles. Moreover, while mapping the user-credentials in our proposed mechanism, the system occasionally requests user input. The user input may also be a factor for the inconsistencies in the global security policies. This topic is out of the scope of this paper and hence not discussed any further.

4. Results

In this Section, the implementation of the policy integration mechanism presented in the preceding Section is explained using an example. Consider two autonomous domains: Astronomical Society (AsS) and Scientific Computational Society (SCS). RBAC policies can be intuitively described using graphs. As an example, figure 2 depicts such graph-based formalism for the RBAC policies for the above stated two autonomous domains. These need to be specified by the security administrator from each domain. In this graph based model, roles are represented as nodes and the edges represent the inheritance relationship between roles. Inheritance relationship allows the users of a senior role to inherit all permissions of junior roles. The capital letters in Figure 2 represents the role labels. The user-credential set associated with the roles is given in the Table 2.

The services which the AsS provides to its users include different levels of access to its astronomical data while the SCS provides access to its computational resources. When a group of scientists needs to work with both these services, they should invite these services for collaboration. XRS and XURAS policy components from both the domains are fed to the policy integration prototype. The prototype generates a single policy which includes XRS and XURAS of the integrated policy. The generated X-RBAC policies components (XRS and XURAS) are applied into an X-RBAC framework. The integrated RBAC policy thus generated and applied is shown in the figures 3 and 4. Figure 3 displays the XURAS generated for the role ‘Astronomer’ while Figure 4 denotes the XRS generated for ‘Astronomer’ role. As we notice from Table 2, the user-credentials for Researcher role is (Designation equal Professor) and that of Astronomer role is ((Position equal Faculty AND Dept equal Astrophysics) OR (Certified by AsS)). Using synonym comparison method we can conclude that user-credential (Designation equal Professor) and (Position equal Faculty) are same as a professor is a position in the faculty. However, a Researcher role does not contain all the credentials associated with an Astronomer role. So the user-credentials of a Researcher role are contained within the user-credentials of an Astronomer role which makes, in the integrated policy, an Astronomer role a parent role to a Researcher role. Since the parent can take on the role of the child due to their qualifications, which would mean user who can take up the role of ‘Astronomer’ in AsS domain, now can take up the role of ‘Researcher’ in SCS domain. Finally, figure 5 depicts the whole integrated policy as a RBAC graph.

![Figure 2. RBAC Models of two autonomous domains before policy integration](image-url)
The equivalence mapping between Data Analyst and Engineer role in the above stated example would result in a consistent RBAC policy. To detect and remove such role mappings, a conflict resolution algorithm is needed.

Table 2. User-credentials of roles in AsS and SCS domains

<table>
<thead>
<tr>
<th>DOMAIN</th>
<th>ROLE</th>
<th>LABEL</th>
<th>USER CREDENTIAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>AsS</td>
<td>Administrator</td>
<td>AD</td>
<td>User: Dr. Martin (Professor at University of Ottawa)</td>
</tr>
<tr>
<td>AsS</td>
<td>Astronomer</td>
<td>AS</td>
<td>Position equal Faculty AND Dept equal Astrophysics OR (Certified by AsS)</td>
</tr>
<tr>
<td>AsS</td>
<td>Member</td>
<td>MB</td>
<td>Member of Recognized Astronomical Societies</td>
</tr>
<tr>
<td>AsS</td>
<td>Telescope Operator</td>
<td>TO</td>
<td>(Professional Engineer)</td>
</tr>
<tr>
<td>AsS</td>
<td>Data Analyst</td>
<td>DA</td>
<td>Educational Degree equal Engineer</td>
</tr>
<tr>
<td>AsS</td>
<td>Student</td>
<td>ST</td>
<td>(Student at Recognized University AND level &gt; Undergraduate)</td>
</tr>
<tr>
<td>SCS</td>
<td>Administrator</td>
<td>AD</td>
<td>User: Dr. Martin (Professor at University of Ottawa)</td>
</tr>
<tr>
<td>SCS</td>
<td>Researcher</td>
<td>RS</td>
<td>Designation equal Professor</td>
</tr>
<tr>
<td>SCS</td>
<td>Project Leader</td>
<td>PL</td>
<td>Account at SCS</td>
</tr>
<tr>
<td>SCS</td>
<td>Student</td>
<td>ST</td>
<td>(Student at Recognized University AND level &gt; Undergraduate)</td>
</tr>
<tr>
<td>SCS</td>
<td>Operator</td>
<td>OP</td>
<td>(Professional Engineer AND Branch = Computer Engineering)</td>
</tr>
<tr>
<td>SCS</td>
<td>Engineer</td>
<td>EG</td>
<td>Qualification equal Engineer</td>
</tr>
</tbody>
</table>

5. Conclusions and future work

In this paper, a new approach for the integration of RBAC policies from heterogeneous domains is proposed. This mechanism for policy integration is simple and clean. User-credential matching for inter-domain role mapping provides a greater flexibility to the trusted third party so that it can dynamically assign users to roles. This is one of the improvements over previous models in which a static user list was associated with the roles. Another benefit of this model is its applicability to heterogeneous domains as no assumptions on the objects or the actions on the objects are made. One more advantage of this mechanism is it protects the privacy policy of the constituent domains. Early experiences of using the proposed technique in multi-domain systems have been positive. We intend to report the results of our on-going investigation of conflict resolution algorithm in future works. We also plan to integrate the proposed technique into CAS which leverages SAML standard to enforce the global policy into community resources.
6. Acknowledgments

Our sincere thanks for generous financial support from the Ontario Centers of Excellence OCE - CITO partnership program and Instantel Inc.

7. References


