Towards Managed Role Explosion

Aaron Elliott
Royal Military College of Canada
Kingston, Ontario
Aaron.Elliott@rmc.ca

Scott Knight
Royal Military College of Canada
Kingston, Ontario
Scott.Knight@rmc.ca

ABSTRACT

Role-based access control (RBAC) is a popular framework for securing information systems in medium to large organizations with hundreds or thousands of employees. However, very few descriptions of existing RBAC systems can be found in the literature. In this paper, we challenge the belief, notion or sense that the number of subjects far exceeds the roles found in enterprise systems. First, we analyze the RBAC system found at ACME University, comparing it to a recently introduced fragment of RBAC called bi-sorted role-based access control (RBÄC). Then we investigate how ACME performs access management, using our new hierarchical graphing model to better visualize the subject-permission mappings. Next, we present our results and introduce a new role-centric methodology for dynamically constraining access to information. Finally, we describe how organizational scalability is enhanced at ACME University by decoupling subject and permission management at the expense of *managed* role explosion.

CCS Concepts

•**Security and privacy** → **Formal security models;** *Security requirements;* Software security engineering;

Keywords

Role-based access control; Organizational structure; Scalability; Complexity; Least privilege

NSPW '15, September 08 - 11, 2015, Twente, Netherlands

DOI: http://dx.doi.org/10.1145/2841113.2841121

1. INTRODUCTION

Our motivation in this work is to challenge the long held belief, notion or sense that the number of subjects far exceeds the roles found in enterprise systems. We explain why *role explosion* occurs in medium to large organizations employing Role-Based Access Control (RBAC) and we make the following contributions:

- We introduce a role-centric approach for dynamically constraining access to information
- We introduce a graphing model to better visualize constrained subject-permission mappings
- We introduce our concept for *managed role explosion* in medium to large organizations

In the following subsection, we review some fundamental access control topics and place this work in context.

1.1 Access Control

Access to information systems is controlled by layers of security. The authorization layer specifies who has access to what. The who is a set of subjects (or users) with access to the system. The what is a set of permissions that have been assigned to the subject. This results in the access relation SP $\subseteq$ S $\times$ P, specifying who is authorized to do what in the information system. In this work we assume a subject has successfully authenticated to an information system, for instance by correctly entering their user name and password. After the subject has gained entry to the information system, they are only permitted to read or write information as authorized. We are concerned with the administration of RBAC, where relations between Subjects and Permissions are explicitly defined and maintained.

RBAC is a popular framework for implementing the authorization layer or who is authorized to do what [17]. Unlike the concept of groups, which only specify group membership, roles identify a set of subjects and a related set of permissions. The administration of RBAC is multi-faceted. Creating user-role, role-role and permission-role relationships are distinct actions that bring subjects and permissions together. For this reason, we depict SP (Figure 1) as a dashed line indicating that it is an implicit relation. An explicit representation includes at least one role, or a role hierarchy, between subjects and permissions.

Consider the case of a role *bank teller*, requiring twenty access control permissions. For this classic RBAC example, a security officer, responsible for access management, creates the role *bank teller* (1), assigns twenty (20) permissions

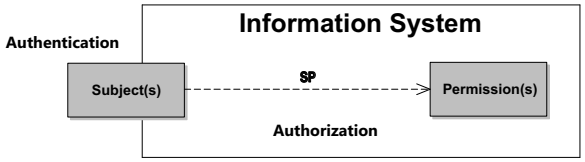

Figure 1: Roles map Subjects to Permissions at the Authorization layer

to the role *bank teller* and then enrolls a subject into the role *bank teller* (1) for a total of twenty-two administrative actions. The security officer invests two extra actions compared to simply assigning twenty permissions directly to the subject, however, the return on investment is nineteen fewer administrative actions for each additional subject enrolled into the *bank teller* role. With RBAC the administrative savings are easily demonstrable.

For large-scale RBAC systems, thousands of access control relationships must be maintained and the administrative effort to maintain user-role, role-role and permission-role relationships is a formidable task that is often highly centralized in a small team of security administrators [14]. The description of Dresdner Bank is one of the few examples of real-world RBAC implementations found in the literature [18]. This case study identifies 40,000 subjects and 1300 roles so one might conclude that the number of users far exceeds roles in enterprise systems. This conclusion might be validated with a sufficient sampling of real-world implementations but one must consider the fact that RBAC is an open technology. As a result, ten large banking organizations like Dresdner may have similar access control requirements and very different RBAC systems. Furthermore, it is important to note that this study was completed many years ago. Considering the technological change that has taken place during the past decade and a half we wonder whether this ratio of 3-4% roles to subjects remains today, noting the challenges described for access review in recent work [6].

Several models have been proposed for maintaining access control implementations. Proofs for the utility of these models are typically restricted to contrived examples that fail to reflect the complexity of medium to large organizations [18]. In the following subsection we review related work, observing scalability concerns for each model.

1.2 Related Work

Recent access control work seems focused on the debate between attribute based access control (ABAC) and RBAC. According to proponents of ABAC it is newer, simpler to implement, and accommodates real-time environmental states as access control parameters [1]. The major challenge of ABAC is the just-in-time evaluation of its rules making it extremely difficult, if not impossible, to determine the permissions available to a particular user [3]. On these grounds alone we consider ABAC insufficient as a security model. With ABAC how does one report on the *protection state* of their computer system when breaches inevitably occur [11][4]. We strongly support a hybrid RBAC-ABAC approach similar to what is described in the role-centric attribute-based access control model [7]. However, unlike this model we are not concerned with dynamic permission-role assignments. Instead, our intent is to dynamically constrain the information returned through static permission-role assignments. In section 2.3, we introduce a role-centric hybrid approach where access to information is dynamically constrained using role attributes.

Returning to our primary motivation of dispelling the long held belief that subjects far exceed roles in enterprise systems, we begin our literature review with the oral estimate given at the RBAC2000 Workshop where the number of roles found in an RBAC system is estimated at 3-4% of the subject population [18]. We are curious if oral estimates have been provided at similar workshops over the past fifteen years and we wonder whether this percentage holds today.

A literature review for the administration of RBAC typically begins with the administrative role-based access control (ARBAC) family of models including ARBAC97, ARBAC99 and ARBAC02 [15][16][14]. ARBAC97 describes the decentralized administration of subject-role enrollment, role-role grants and permission-role assignment with reference to the RBAC96 model [17]. In their introduction to ARBAC97 the authors presume that *in large enterprise-wide systems, the number of roles can be in the hundreds or thousands, and subjects in the tens of hundreds or thousands* suggesting that the ratio of roles to subjects is 10%. This belief, notion or sense that the number of subjects far exceeds the roles found in enterprise systems is repeated in each extension to the ARBAC97 model as is the example of a Director overseeing two projects with a Project Lead, Production Engineer, Quality Engineer and a (Junior) Engineer. However, if one considers the simple example of one employee in each role on each project we observe that this example defines eleven roles for seven subjects, a ratio of 160%. The reader is expected to intuit that this example depicts scenarios where a few roles are shared by many subjects.

We are not convinced that all Junior Engineers, for instance, will have the exact same access control requirement even in the context of one project. We understand this is implicit in the example provided. However, Sandhu et al. suggest *this structure can, of course, be extended to dozens and even hundreds of projects within the engineering department.* When we conceptually scale this toy model up to hundreds of projects in a society where the demand for skilled workers is not being met [8], we wonder whether Junior Engineers are allocated in quantity to single or multiple projects. If it is the latter then we suggest the 10% ratio of roles to subjects anecdotally described in this paper does not align with the example imagined in this same work.

SARBAC, the scoped administration of role-based access control, is intended to be used with RBAC96 as a complete role-based model for administration [2]. Unlike ARBAC, SARBAC does not assume the existence of a disjoint set of administrative roles. SARBAC develops a model for role hierarchy administration with the belief that it will be easier to then incorporate subject-role and permission-role administration. We observe that Crampton et al. reuse the example from ARBAC97 where the ratio of roles to subjects is 160%.

A-ERBAC, administrative enterprise role-based access control, describes the model employed in a commercial enterprise security management software solution [9]. Kern et al. suggest that Enterprise Roles are increasingly used by medium to large organizations as the basis for security man-

agement across different systems. A-ERBAC uses the concept of *scopes* to control the authority of administrators on a Target System. An administrator may be assigned the ability to view, insert, change or delete various RBAC elements such as subjects and roles provided they are assigned one or more administrative scopes within the hierarchy of objects (e.g. Organizational Units or Cost Centres). Furthermore, Kern et al. argue that the scopes of A-ERBAC provide a more comprehensive solution than the *pools* of ARBAC02 as each scope is optionally associated with attributes that enrich the administrative convenience. This work also provides a case study from a European bank where subjects are created and deleted using connections to the Human Resources database.

It is not explicitly stated in this work that Dresdner bank is the institution observed [18]. If indeed this case is based on another banking institution with 70,0000 subjects (vice 40,000 at Dresdner) it would have been interesting to know the number of roles implemented in this system. Instead, we observe that the example with functional and business roles depicted in this work identifies a scenario where eight roles are defined for five subjects. We understand that this disproportionate use of subjects and roles is not the focus of the A-ERBAC model and acknowledge that the authors assume the reader will intuit that as this example scales more subjects will be disproportionately enrolled in the defined business roles. Nevertheless, we find it interesting that the ratio of roles to subjects is 160% in the example depicted.

With all due respect to the Dresdner bank case study, consider the classic RBAC example of the *bank teller* role where hundreds of subjects share a generic, simplistic role with the exact same permissions. We argue that this scenario is still well-ingrained in the collective psyche of the research community and we have found it difficult to challenge this long held belief and gain traction. Anecdotally, the rise of on-line banking, automated teller machines and the proliferation of credit card transactions has impacted the number of employees, specifically bank tellers employed at banks. It would be interesting to review the ratio of subjects to roles at Dresdner bank today. We suspect that if one was to monitor this ratio on a yearly basis there would be a clear trend.

In this paper, we challenge the belief, notion or sense that the number of subjects far exceeds the roles found in enterprise systems. In section 2, we offer a new case study where the number of roles exceed subjects. This may seem counterintuitive to those visualizing a classic RBAC example where tens or hundreds of subjects are enrolled in the *bank teller* role which has been assigned several permissions.

It is not clear whether the classic subject-role-permission model remains dominant in medium to large organizations with highly skilled workforces. Over a decade and a half ago, the A-ERBAC model described a second layer of roles between subjects and permissions without expressly highlighting the fact that this implies subjects will always hold a minimum of two roles (i.e. subject-role-role-permission). We understand that these roles are meant to be shared but we believe there has been a fundamental shift in the way RBAC systems are implemented and maintained over the past decade. In our analysis we are unable to determine if the research community introduced the notion of at least two roles for each subject [13] or whether this was an organic by-product of real-world implementations [18] or both.

In the following section, we review the work of Kuiper et al. in detail as their model formalizes the conceptual boundaries described in A-ERBAC, suggesting that subjects must **always** have at least two roles. We strongly support this formalized approach and consider it the basis for our role-centric constraints introduced in section 2.3.

1.3 Support for RBÄC

In this section, we analyze a fragment of RBAC called bi-sorted role-based access control (RBÄC) which may be applied to existing RBAC implementations, the perceived added value lies within the conceptual boundaries it introduces, decoupling subject and permission management, thus introducing a higher administrative level for access management [10]. For practitioners, this decoupling implies that modeling (1) subjects and (2) permissions is broken into independent activities. With these two aspects maintained by suitable teams, security officers may configure access control rules at an appropriate level of abstraction.

In addition, RBÄC inherently facilitates many-to-many administrative mutations and ultimately leads to more organizational scalability. The speculation being that such an approach might prove beneficial in the following senses:

- subject management can be delegated as appropriate in organizations, reducing administrative overhead.
- application architects can focus on creating independent roles based on the functional requirements.
- security officers can perform access management at an appropriate level of abstraction.

RBÄC describes two distinct objects of indirection: the (1) proper role and (2) demarcation which are used to distinguish conceptual boundaries for the administration of RBAC. Unlike previous extensions to the classic RBAC model [17], RBÄC revisits first principles with the hypothesis that organizational scalability is facilitated when permissions are managed independently from subjects [10].

This is not a new idea as we have already said in the previous section. Kuijper and Ermolaev cite Oh and Park as the first researchers proposing permissions be grouped independently into *task-based roles* [13]. Next, the work of Kern et al. [9], on enterprise role-based access control (ERBAC) is provided as an example, clearly demonstrating the practicality of maintaining two distinct role hierarchies. Finally, the work of Nyanchama and Osborn [12] is referenced as further evidence that a dichotomy exists between subject and permission management. The important difference to consider with RBÄC is the assertion subjects and permissions are never linked by a single role. Instead, there is **always** *at least two roles* between a subject and a permission.

In contrast to previous extensions, RBÄC is presented as a *fragment* of RBAC that *proposes a conceptual split right down the middle through the core of the access control model.* The result is a conceptual shift from the triangular classic RBAC model to the square model of RBÄC where access relations are defined using an additional layer of abstraction.

In Figure 2a), we see that RBAC introduced the role as a layer of indirection between subjects and permissions. SP is the implicit result of assigning permissions to a role (PR) and then enrolling subjects into this role (SR). The result

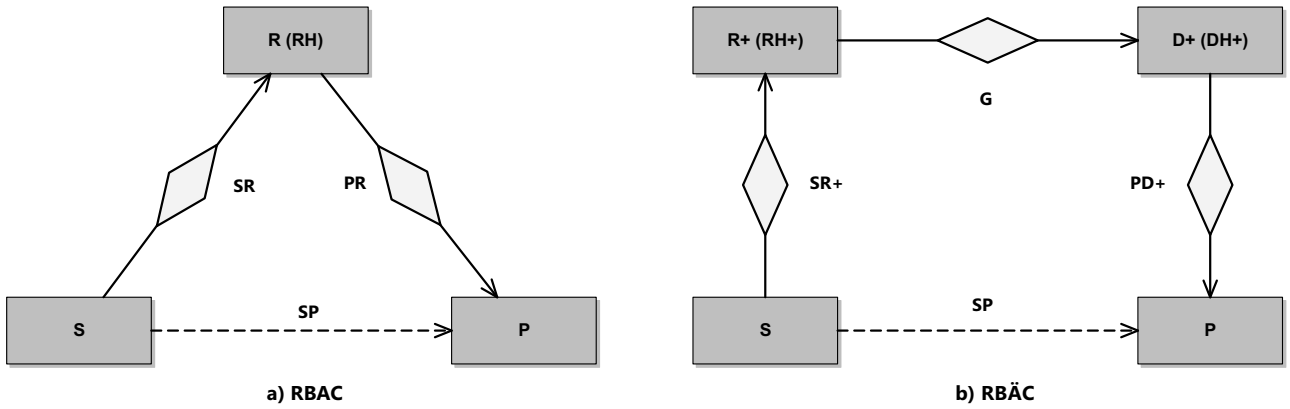

Figure 2: RBÄC proposes a new conceptual boundary between subjects and permissions. Adapted from [10].

is still SP as it was in Figure 1, however the number of access control relations is reduced, thus addressing the subject-permission explosion problem [10].

To achieve the square of RBÄC in Figure 2b), another layer of indirection is introduced with roles being categorized as either a proper role (R+) or a demarcation (D+) . Permissions are assigned to demarcations (PD+) and subjects are enrolled into proper roles (SR+). Permissions are never assigned to proper roles directly. Instead all subjects obtain permissions indirectly through the grant relation (G) where proper roles and demarcations are *linked up*. Figure 2 slightly adapts the work of Kuijper and Ermolaev. Consistent with Figure 1 we indicate that SP is an implicit relationship with a dashed line. In addition, we include directional arrows, depicting the explicit role hierarchy that exists between subjects and permissions.

Definition 1. RBÄC retains the principal semantic domains underlying RBAC (i.e. S and P) and defines the following syntax:

- Let S be a set of subjects
- Let P be a set of permissions
- Let R+ be a set of proper roles
- Let D+ be a set of demarcations [1]
- Let SR+ $\subseteq$ S $\times$ R+ be a subject-proper-role assignment relation
- Let PD+ $\subseteq$ P $\times$ D+ be a permission-demarcation assignment relation
- Let RH+ $\subseteq$ R+ $\times$ R+ be a proper-role-hierarchy relation, RH+ is required to be acyclic
- Let DH+ $\subseteq$ D+ $\times$ D+ be a demarcation-hierarchy relation, DH+ is required to be acyclic
- Let G $\subseteq$ R+ $\times$ D+ be a grant relation

[1] R+ and D+ are disjoint sets

Definition 2. The semantics of RBÄC identify the access relations SP $\subseteq$ S $\times$ P such that (s,p) $\in$ SP iff there exists roles r, r$'$ $\in$ R$^+$ and demarcations d, d$'$ $\in$ D$^+$ and the following conditions hold:

1. (s,r) $\in$ SR$^+$, i.e.: subject s is a member of proper role r.
2. r $\geq_r^+$ r$'$, i.e.: r = r$'$ or r is a senior role of r$'$ [2]
3. (r$'$,d$'$) $\in$ G, i.e.: proper role r$'$ is granted access to demarcation d$'$
4. d$'$ $\geq_r^+$ d, i.e.: d = d$'$ or d is a sub-demarcation of d$'$. [3]
5. (p,d) $\in$ PD$^+$, i.e. permission p is part of demarcation d

For small organizations where the number of roles remain relatively small, classic RBAC is often an adequate solution. However, for medium to large organizations where there is an ongoing requirement to support employee turnover, policy changes and reorganization, RBÄC seems to be a logical progression. Despite the advantages of RBAC, the administrative degrees of freedom become limited when practitioners utilize a triangular RBAC model where there are four basic mutations [10]:

1. Enroll a subject s $\in$ S to role r $\in$ R, i.e.: add (s,r) to SR
2. Disenroll a subject s $\in$ S from role r $\in$ R, i.e.: remove (s,r) from SR
3. Assign a permission p $\in$ P to role r $\in$ R, i.e.: add (p,r) to PR
4. Unssign a permission p $\in$ P from role r $\in$ R, i.e.: remove (p,r) from PR

The effect of an atomic RBAC mutation on SP is always one-to-many or many-to-one and never many-to-many. This is referred to as the administrative micro-stepping problem

[2] $\geq_r^+$ defines the transitive reflexive closure of RH$^+$
[3] $\geq_d^+$ defines the transitive reflexive closure of DH$^+$

[10]. With RBÄC the intent is to *break away* from the classic triangular example of RBAC, enforcing another degree of freedom and facilitating many-to-many mutations for SP. For practitioners this additional layer of abstraction permits administrative degrees of freedom not enjoyed under the classic triangular RBAC model. In the following sections, we see how ACME University takes advantage of this administrative flexibility to enhance the scalability of its RBAC system.

As previously, stated this is not a new idea and one could certainly design RBAC systems in this fashion prior to RBÄC. However, with this fragment of RBAC the suggestion is that one would **never** assign permissions to a role granted directly to users. In the strictest sense a subject would never receive a permission in a subject-role-permission mapping. This guarantees that every subject has at least two roles and often many more.

In extreme cases at ACME University, subjects have more than one hundred roles. Many of these roles are demarcations or discrete units of functionality delivered organically over time as functional or task-based roles. Readers who are familiar with this area of research may be experiencing a strong sense of disbelief at this point. Others might immediately decide this is a poorly designed RBAC system. Ten or more years ago we might have been equally critical but today we know that ACME University is not the only organization describing subjects with more than one hundred roles [6].

RBÄC is explained in the context of a physical access control system. In section 2, we analyze the RBAC implementation used by ACME University to secure its student information system where access is granted on a *need to know* basis. In particular, we describe an interesting scenario whereby Department Heads at ACME University obtain access to their respective course lists. We observe that role attributes are used to constrain access to information.

2. ACME UNIVERSITY

In this section we analyze a real-world RBAC system, performing an empirical study of the ACME University student information system (SIS), an Oracle® database. In the following subsections we present our observations at the macro level before performing a microanalysis on a scenario of special interest, introducing our new graphing model in order to better visualize the details of subject-role, role-role and permission-role mappings. Next, using our scenario of interest, we introduce a role-centric approach for dynamically constraining access to information with role attributes. Finally, we introduce our notion of *managed role explosion*.

2.1 Empirical Study

ACME University is located in North America. While the name of the institution is contrived for anonymity, the information that follows is based upon an existing RBAC system, supporting more than 700 employees. We performed queries against the data dictionary to determine the complete set of roles created in the SIS[1]. From this set we have excluded subjects and roles created at the installation of the database software[2].

[1]Relational databases typically include meta data repositories identifying objects created in the system such as roles

[2]Oracle® databases include several roles such as *DBA* that were considered out of scope for this study

Table 1: ACME Roles as at April 2015

Label	Count
Subject	351
Role	558

Table 2: ACME Grants as at April 2015

Grant	RBÄC	Count
Subject-Role	SR^+	386
Role-Role	RH^+	294
Role-Role	G	683
Role-Role	DH^+	215
Permission-Role	PD^+	2281

As listed in Table 1 we discovered that ACME University has a role to subject ratio of 160%. Coincidentally, this is the ratio we describe in the related work section 1.2 for both ARBAC97, SARBAC and A-ERBAC. This is clearly a huge deviation from the 3-4% ratio estimated at the RBAC2000 Workshop or the 10% ratio presumed by Sandhu et al. in ARBAC97.

When further investigating the available metadata, we learned that the ACME University role information was documented within the SIS, greatly facilitating this research. Each and every role was labeled an Appointment, a Positional, a Group or a Functional role in a table called *Role Documentation*. Understanding that RBÄC proper roles (R+) are granted to subjects and do not obtain permissions directly, we determined that Appointment, Positional and Group roles met this description. Like RBÄC demarcations (D+), Functional roles were assigned permissions directly and granted to proper roles.

Subjects were enrolled into Appointment roles with no direct permissions as described in RBÄC (SR^+). For example, when a faculty member is appointed as the Department Head for Electrical and Computer Engineering (ECE) for a three year term, the Human Resources (HR) group enters this information into the SIS. This data entry is used to automate the enrollment of a subject into the corresponding *Department Head - ECE* role for a three year term.

Subjects were also enrolled into Positional roles on an indeterminate or term basis (SR^+). Employees hired indeterminately were identified by a job title or position number and enrolled in the corresponding Positional role. Unlike Appointment roles, the enrollment of subjects into Positional roles was not automated. We understood this automation would be introduced at a future date.

With consideration for the examples presented in ARBAC, SARBAC and A-ERBAC this seems reasonable. ARBAC and SARBAC describe a model that is based on the positions found in the Engineering Department of a fictitious organization. A-ERBAC uses the example of business roles, similar to positions, in their account of the design found in

a European bank.

Using the *Role Documentation* and the data dictionary allowed us to determine the subject-role, role-role and permission-role mappings. Our query results are listed in Table 2. As further evidence that the RBAC system under study is indeed a practical example of RBÄC we queried role-role grants. We observed that Appointment and Positional roles were often enrolled into Group roles forming proper role hierarchies (RH^+). As one example, the role *Department Head - ECE* was enrolled into the *Department Head* Group role. Similarly, we observed that Functional roles were often assigned to one another in demarcation hierarchies (DH^+). As one example, the role *Approve Grades* was assigned to the role *Final Grades*. Finally, we observed that Functional roles were granted to Appointment, Positional and Group roles consistent with the Grant (G) relation of RBÄC.

2.2 Scenario of Interest

In this section we describe how ACME University performs access management, describing the relationship between the proper role *Department Head - ECE* and the demarcation *Approve Grades* within the Student Information System (SIS). In Figure 3 we visualize our scenario using the graphing model introduced by Kuijper and Ermolaev [10]. On the left side of the graph are the proper roles Department Head - ECE and Department Head. On the right side of the graph are the demarcations Final Grades and Approve Grades.

Example 1. Department Head - ECE responsible for Approving Grades.

- s_1 = Dr. George Scott
- p_1 = SELECT information FROM course
- R^+ = {Department Head - ECE, Department Head}
- RH^+ = {(Department Head - ECE, Department Head)}
- D^+ = {Final Grades, Approve Grades}
- DH^+ = {(Final Grades, Approve Grades)}
- SR^+ = {(s_1, Department Head - ECE)}
- PD^+ = {(p_1, Approve Grades)}
- G = {(Department Head, Final Grades)}

At the end of each academic term, the Department Head for ECE is responsible for approving final grades. Grades are entered by instructors who teach an academic course to students, set up an evaluation scheme and grade the student. Before the final grade is released to a student it must be approved by the Department Head. In Example 1, the subject-permission (SP) mapping required for the Department Head - ECE to approve final grades is *spelled out* in the language defined by RBÄC. In this example, the subject is Dr. George Scott (s_1) and the permission is SELECT information from course (p_1). In other words, Dr. George Scott may retrieve and view a list of courses in the SIS.

In Figure 4 we introduce our new hierarchical diagramming notation to better visualize the directional hierarchy of subject-role, role-role and permission-role grants. The elements of our diagram can be reconfigured to look like the *desirable square (or rectangle)* of RBÄC however, we

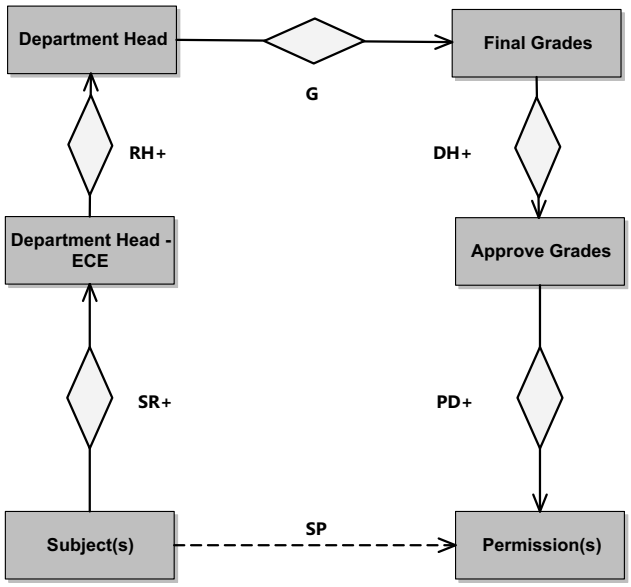

Figure 3: An RBÄC example found at ACME

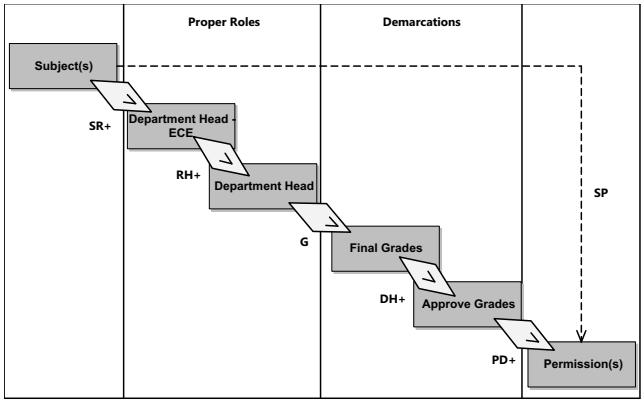

Figure 4: Hierarchical Graph for the example found at ACME

feel that the notion of hierarchy is an important aspect not well represented in Figure 3. Instead, we use *swim lanes* to depict the conceptual boundaries between subjects, proper roles, demarcations and permissions. To better comprehend real-world RBAC systems it is important to visualize the directionality of the enroll, grant and assign relationships and *see* the depth at which one actually obtains permissions to do something. In our scenario of interest there is a cascade of diamond shaped relationships between entities (i.e. rectangles) ultimately linking subjects and permissions together. We use arrows within the diamonds to indicate directionality. When one analyzes Figure 3 and Figure 4, it is important to understand the implicit relationship SP is non-trivial. There are four roles explicitly defined between Dr. George Scott and the permission SELECT information from course. As we will see, this new hierarchical diagramming technique will also facilitate the introduction of our notion of role-centric constraints and managed role explosion in the following sections.

At this juncture, the reader may be curious why ACME University has chosen to implement an RBAC system where four levels of roles, two proper roles and two demarcations, exist between Dr. George Scott and the permission required to approve final grades. We use our new diagramming notation to present dynamic constraints in the following section, providing our understanding of the rationale for this design.

2.3 Role-Centric constraints

In this section we present a new methodology for dynamically constraining permissions under RBÄC, providing additional validity for this new fragment of RBAC. We elaborate upon the scenario of interest from the previous subsection where the Department Head for ECE is responsible for approving final grades at the end of each academic term. Next, we present our results and introduce a new role-centric methodology for dynamically constraining access to information. Unlike ARBAC and SARBAC which describe constraints on subject-role enrollments and permission-role assignments our constraint model introduces constraints on permissions already assigned to one or more roles. Unlike RABAC [7], whose intent is to dynamically constrain the set of permissions available to users, our intent is to dynamically constrain the information returned by static role-permission grants.

At ACME University, each Department Head is responsible for approving final grades within their respective department. We observed that each Department Head was directly enrolled in a proper role indicative of their appointment, meaning one unique role for each and every Department Head. This was counterintuitive. Classic RBAC suggests that all subjects share a *grouping* role called *Department Head* in a triangular subject-role-permission design. However, we discovered that instead of subjects sharing a grouping role, there were eleven unique *Dept Head* roles enrolled in a *grouping* Department Head role. Using Figure 5 as a point of reference, consider the result. One could draw eleven diagrams exactly alike, simply substituting the *Department Head - ECE* role with each of the other ten roles found in the SIS.

Although we did not initially see the advantages of such an RBAC system, questioning the excessive number of roles, we began to appreciate the simplicity of the subject-role relations found *on the surface* as we tunneled deeper into the SIS design.

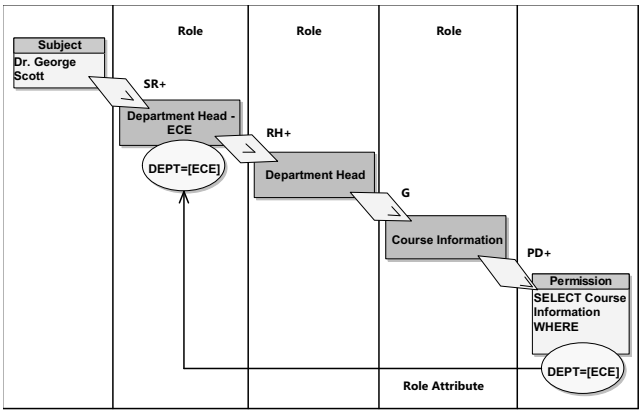

Figure 5: Contstraining RBÄC. The List of Courses that may be selected by the Department Head - ECE

Each and every Department Head shared similar if not identical permissions and this was implemented by granting demarcations (i.e. functionality) to the grouping Department Head role. However, each individual appointed to these roles is responsible for different information sets. At ACME University, proper roles are used to directly influence the records returned in permissions, for example, SELECT information from course. We learned that the proper role Department Head - ECE, was assigned the role attribute, or (name, value) pair (Department, ECE). Then when applicable queries were performed the WHERE clause used this role attribute to determine what records should be returned. This was derived directly from the roles held by the subject. When Dr. George Scott, the Department Head for ECE, accessed the SIS his role attributes were used to determine whether or not courses may be viewed. Figure 5 depicts the *match* condition used to restrict the List of Courses for the Department Head - ECE when approving final grades.

This was a recurring theme. We learned that the Department Head for ECE had SELECT access on similarly restricted lists of students, programs and staff within their respective Department. In each case, the WHERE clause for lists of information used a proper role attribute to restrict access to information. Figure 6 depicts the *match* condition used to restrict the List of Courses for both the Department Head - ECE and Department Head for Mechanical Engineering (MEC) when approving final grades.

We raised our concern with the practice of assigning the Department (name,value) pair to the applicable role vice simply using the Department assigned to the person in the Human Resources (HR) system [5]. In response to our concern, we were informed that the department of a subject or individual did not always reflect an appointment. We were informed that Deans, for instance, were faculty members within one academic department while simultaneously responsible for an entire faculty under their appointment. For instance, if the Dean of Engineering taught in the ECE department the HR system would be too restrictive since the Dean is permitted to view information for an entire Faculty or collection of Departments.

There were many examples of dynamic permission constraints found within the SIS. We were inundated with ex-

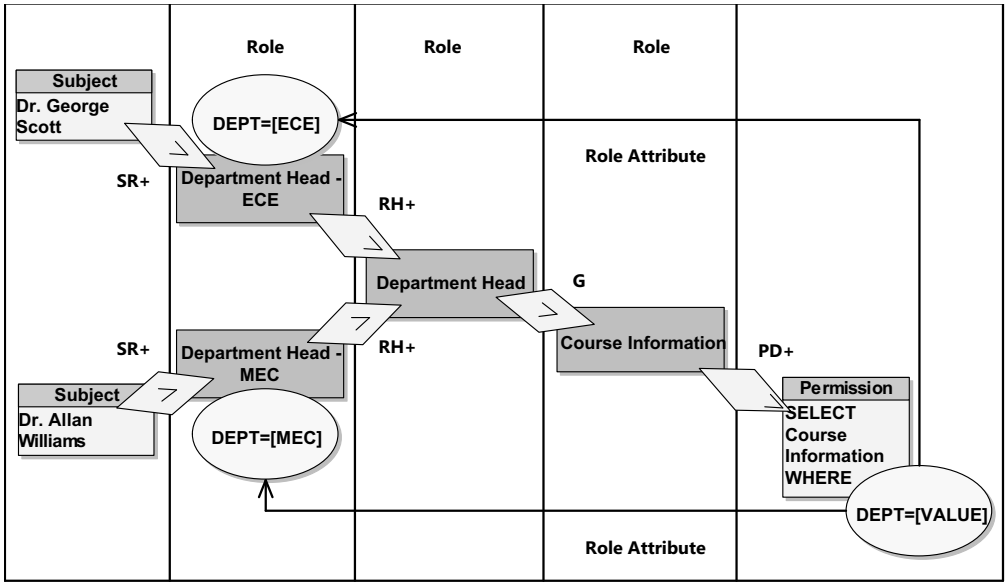

Figure 6: Determining the List of Courses that may be selected by two Department Heads.

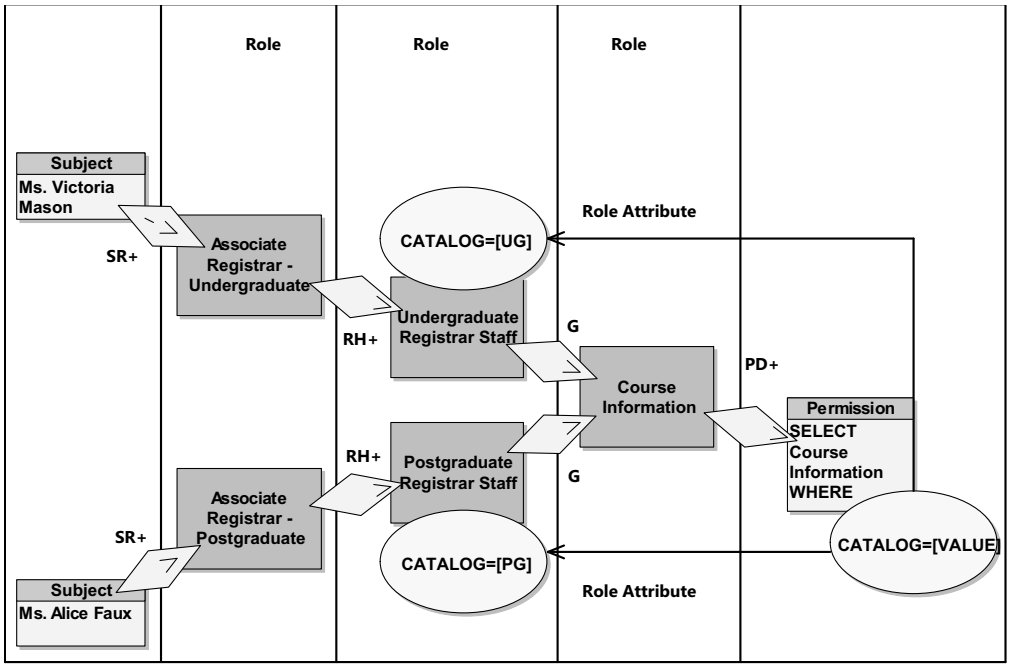

Figure 7: Determining the List of Courses that may be selected by two Staff Members.

amples of the practicality of this approach, especially when considering the degrees of freedom afforded to application architects. We were informed that the relative cost of creating and maintaining database objects for applications was expensive. On the other hand, the cost of creating and maintaining light-weight roles that could be easily delegated with no direct permissions was inexpensive. For this reason, the SIS designers had chosen to *hide* this design decision within the role information, affording their applications more flexibility and scalability. Organically, from the year 2000 onward, they had discovered that who has access to what was difficult to maintain in an environment with constant employee turnover. By aggregating permissions into Functional roles (i.e. demarcations) and assigning these roles to Appointment, Positional or Group roles (i.e. proper roles) in a role hierarchy they could avoid losing important security relationships when employees left the organization. If a subject had acquired several permissions directly or indirectly via grouping roles this was previously lost when subjects were dropped from the RBAC system. To avoid this loss, permissions are aggregated into Functional Roles and assigned to Appointment, Positional or Group roles.

Figure 7 depicts the *match* condition used to restrict the list of courses for both the Undergraduate (UG) and Postgraduate (PG) Associate Registrars. In this instance, we see that the role attribute, or (name, value) pair (Catalog, [Value]), is used to restrict access to information. When the Associate Registrar Undergraduate accesses course information the name value pair (Catalog, [UG]) is used in the WHERE clause, and when the Associate Registrar Postgraduate accesses course information the name value pair (Catalog, [PG]) is used in the WHERE clause.

Definition 3. Introduce constraints for RBÄC as a name value pair attribute of proper roles (R^+):

- Let A+ be a set of attributes of the form (name,value) pair
- Let $C \subseteq R^+ \times A^+$ be a proper role constraint relation

Example 1 continued. Department Head - ECE responsible for Approving Grades in their respective department. Department Head - ECE may only SELECT course information WHERE the DEPT=[ECE] (Figure 6)

- C = {(Department Head - ECE, (Department,ECE))}

Example 2. Department Head - MEC responsible for Approving Grades in their respective department. Department Head - MEC may only SELECT course information WHERE the DEPT=[MEC] (Figure 6)

- C = {(Department Head - MEC, (Department,MEC))}

Example 3. The Associate Registrar Undergraduate (UG) is responsible for maintaining UG Course Information. The Associate Registrar Undergraduate may only SELECT course information WHERE the CATALOG=[UG] (Figure 7)

- C = {(Associate Registrar Undergraduate, (Catalog,UG))}

Example 4. The Associate Registrar Postgraduate (PG) is responsible for maintaining PG Course Information. The Associate Registrar Postgraduate may only SELECT course information WHERE the CATALOG=[PG] (Figure 7)

- C = {(Associate Registrar Postgraduate, (Catalog,PG))}

RBÄC provides a stabilizing layer for RBAC. It facilitates the automation of enrolling employees into proper roles (SR^+) and it prevents the loss of important security relationships when employees leave the organization. In this work, we introduce dynamic constraints for RBÄC as a means of enforcing conditionals when subjects share similar roles but different contexts. This is an important design concern for medium to large organizations operating under a *need to know* security policy.

2.4 Managed Role Explosion

In the previous subsection we describe how organizational scalability is enhanced at ACME University by decoupling subject and permission management at the expense of *managed* role explosion. In Figure 8 one can visualize how ACME University uses the flexibility and scalability of Structured Query Language (SQL) to reuse the same permission for various user groups, providing consistent, context aware information to a variety of employees at the University[1].

We observe that ACME University is a real-world instance of RBÄC where subjects and permissions are never linked by a single role. Instead, there is **always** *at least two roles* between a subject and a permission. We understand that this happened organically over the past fifteen years and was directly influenced by the ARBAC [15], SARBAC [2], A-ERBAC [9] and T-RBAC [13] models.

Security practitioners at ACME University had discovered that who has access to what was difficult to maintain in an environment with constant employee turnover. By aggregating permissions into functional or task-based roles (i.e. demarcations) and assigning these roles to Appointment, Positional or Group roles (i.e. proper roles) in a role hierarchy they could avoid losing important security relationships when employees left the organization. This is exactly what RBÄC proposes with the following results:

- subject management is delegated as appropriate in organizations, reducing administrative overhead
- application architects are able to focus on creating independent roles based on the functional requirements
- security officers are able to perform access management at an appropriate level of abstraction

Finally, the fact that ACME University documents role information within their SIS is important to highlight. Not only did this practice greatly facilitate this research, it is directly responsible for our new concept of *managed role explosion*, the hypothesis being that if the business model of an organization directly informs its access control implementation the result is a security model that is more easily understood, more receptive to change and simpler to maintain. The practitioners at ACME University would argue that in their experience this has proven to be an invaluable design decision as they continue to integrate and develop their information systems over the span of months and years.

[1]Relational Databases like Oracle® use a standard language to both store and retrieve information

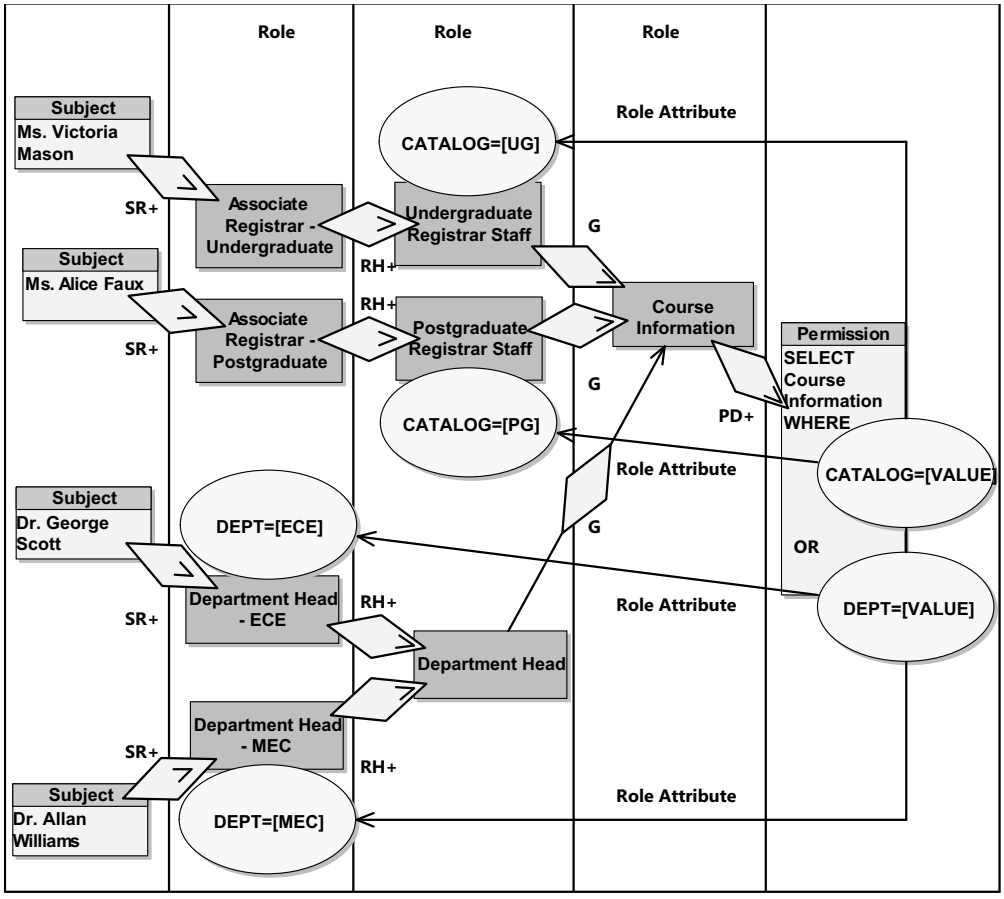

Figure 8: Determining the List of Courses that may be selected by a variety of subjects.

3. CONCLUSION

In this paper, we suggest that there is a presumption concerning the ratio of roles to subjects in the research community. First, we perform a literature review for administrative role-based access control models, highlighting the deviation between the presumed 3-10% ratio of roles to subjects and the examples described in these same models (160%). Next, we describe a recently introduced fragment of RBAC called bi-sorted role-based access control (RBÄC) and compare it to the RBAC system found at ACME University where we discover the ratio of roles to subjects is 160%. We learn that practitioners at ACME University were directly influenced by the ARBAC, SARBAC, A-ERBAC and T-RBAC models, organically resulting in a security design similar to what is described in bi-sorted role based access control (RBÄC).

In this paper we make the following contributions:

- We challenge the long held belief, notion or sense that the number of subjects far exceeds the roles found in enterprise systems
- We introduce a new role-centric approach for dynamically constraining access to information
- We introduce a new graphing model to better visualize constrained subject-permission mappings
- We introduce our new concept for *managed role explosion* in medium to large organizations

In future work, we intend to continue our analysis of the RBAC system found at ACME University. We are interested in RBAC systems that are directly influenced by the business model of an organization similar to what is described in RBÄC. Our hypothesis is that medium to large organizations are either *managing* their role explosion in a flexible, scalable RBAC system like ACME University or they are finding it difficult to work with an inflexible RBAC system where practitioners are overly concerned with maintaining a low ratio of roles to subjects. Alternatively, some organizations may be heavily invested in an attribute-based access control (ABAC) system where there may be unacceptable security trade-offs.

The objective of this work is to challenge a long held belief concerning the ratio of roles to subjects in support of our broader research objectives. We intend to deliver a model and methodology for implementing access control in medium to large organizations with hundreds or thousands of employees. This model and methodology are founded upon the notion of managed role explosion, assuming that we must organically integrate our model into both new and existing organizations. For the latter case, we understand that the rationale for doing this must be supported by analytical measures, providing evidence for using an *improved* access control model. Finding metrics to achieve this objective might prove challenging because organizations likely have no idea what their current access control systems are costing them. For this reason, the savings must be easily demonstrable as with the classic RBAC bank teller example. We expect these measures will relate to the degree of automation for proper roles (e.g. appointments and positions) and the degree of delegation for demarcations (e.g. functional roles).

This research has the potential to impact the perception of access control. Rather than being viewed as a necessary burden, access control has the potential to be a well understood, enabling technology, directly informed by the business model, a scaffolding for maintaining information systems, where the individual parts are simple but the flexibility and utility achieved through the sum of the parts provide the necessary framework for scalable access control systems.

Acknowledgement

We would like to thank Sean Peisert for his generosity, invaluable insight and thoughtful review of this work. We also appreciate the helpful guidance of the anonymous reviewers whose constructive suggestions have been addressed, strengthening the pre-proceedings version of this paper. Finally, we would like to thank the participants of NSPW 2015 for contributing to our research, in particular Bob Blakley and Mary Ellen Zurko, whose comments and guidance will directly influence future work.

4. REFERENCES

[1] E. J. Coyne and T. R. Weil. ABAC and RBAC: Scalable, flexible, and auditable access management. *IT Professional*, 15(3):14–16, 2013.

[2] J. Crampton and G. Loizou. Administrative scope. *ACM Transactions on Information and System Security*, 6(2):201–231, may 2003.

[3] EmpowerID. Best Practices in Enterprise Authorization : The RBAC/ABAC Hybrid Approach. Technical report, EmpowerID, 2013.

[4] G. S. Graham and P. J. Denning. Protection. In *Proceedings of the November 16-18, 1971, fall joint computer conference on - AFIPS '71 (Fall)*, page 417, New York, New York, USA, 1971. ACM Press.

[5] V. C. Hu, D. Ferraiolo, R. Kuhn, A. Schnitzer, K. Sandlin, R. Miller, and K. Scarfone. Guide to Attribute Based Access Control (ABAC) Definition and Considerations. Technical report, National Institute of Standards and Technology, Gaithersburg, MD, jan 2014.

[6] P. Jaferian and K. Beznosov. Poster : Helping users review and make sense of access policies in organizations. In *SOUPS '14: Proceedings of the Tenth Symposium On Usable Privacy and Security*, pages 301–320, 2014.

[7] X. Jin, R. Sandhu, and R. Krishnan. RABAC: Role-centric attribute-based access control. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7531 LNCS:84–96, 2012.

[8] L. Katz and R. Margo. Technical Change and the Relative Demand for Skilled Labor: The United States in Historical Perspective. Technical Report January, National Bureau of Economic Research, Cambridge, MA, feb 2013.

[9] A. Kern, A. Schaad, and J. Moffett. An administration concept for the enterprise role-based access control model. *Proceedings of the eighth ACM symposium on Access control models and technologies - SACMAT '03*, page 3, 2003.

[10] W. Kuijper and V. Ermolaev. Sorting out role based access control. *Proceedings of the 19th ACM symposium on Access control models and technologies - SACMAT '14*, pages 63–74, 2014.

[11] B. W. Lampson. Protection. *ACM SIGOPS Operating Systems Review*, 8(1):18–24, jan 1974.

[12] M. Nyanchama and S. Osborn. The role graph model and conflict of interest. *ACM Transactions on Information and System Security*, 2(1):3–33, feb 1999.

[13] S. Oh and S. Park. Task-role based access control (T-RBAC): An improved access control model for enterprise environment. *Database and Expert Systems Applications*, pages 264–273, 2000.

[14] S. Oh and R. Sandhu. A model for role administration using organization structure. *Proceedings of the seventh ACM symposium on Access control models and technologies - SACMAT '02*, page 155, 2002.

[15] R. Sandhu, V. Bhamidipati, and Q. Munawer. The ARBAC97 model for role-based administration of roles. *ACM Transactions on Information and System Security*, 2(1):105–135, feb 1999.

[16] R. Sandhu and Q. Munawer. The ARBAC99 model for administration of roles. In *Proceedings 15th Annual Computer Security Applications Conference (ACSAC'99)*, pages 229–238. IEEE Comput. Soc, 1999.

[17] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. Role-based access control models. *Computer*, 29(2):38–47, 1996.

[18] A. Schaad, J. Moffett, and J. Jacob. The role-based access control system of a European bank. In *Proceedings of the sixth ACM symposium on Access control models and technologies - SACMAT '01*, pages 3–9, New York, New York, USA, 2001. ACM Press.