ROLE MODEL OPTIMIZATION FOR SECURE ROLE-BASED IDENTITY MANAGEMENT

Complete Research

Fuchs, Ludwig, Universität Regensburg, Regensburg, Germany, ludwig.fuchs@wiwi.uni-regensburg.de

Kunz, Michael, Universität Regensburg, Regensburg, Germany, michael.kunz@wiwi.uni-regensburg.de

Pernul, Günther, Universität Regensburg, Regensburg, Germany, guenther.pernul@wiwi.uni-regensburg.de

Abstract

In the recent past, the application of role-based access control for streamlining Identity and Access Management in organizations has gained significant importance in research and practice. After the initial setup of a role model, the central challenge is its operative management and strategic maintenance. In practice, organizations typically struggle with a high number of potentially outdated and erroneous role definitions leading to security vulnerabilities and compliance violations. Applying a process-oriented approach for assessing and optimizing role definitions is mandatory to keep a role model usable and up to date. Existing research on role system maintenance only provides a limited technical perspective without focusing on the required guidance and applicability in practice. This paper closes the existing gap by proposing ROPM, a structured Role Optimization Process Model for improving the quality of existing role definitions. Based on comprehensive tool support it automates role optimization activities and integrates both, a technical as well as a business-oriented perspective. It is based on the iterative application of role cleansing and role model extension activities in order to reduce erroneous role definitions and (re-)model roles according to organizational requirements. In order to underline applicability, this paper provides a naturalistic evaluation based on real-life data.

Keywords: Role Maintenance, Role Optimization, Role Mining, Identity and Access Management, RBAC.

1 Motivation

Effectively administrating employees' access to sensitive applications and data is one of the biggest security challenges for today's organizations. A typical large organization manages millions of user access privileges that are spread across thousands of IT resources. Employees are assigned to digital identities which allow them to access these resources. National and international regulations like the Sarbanes-Oxley Act (2002), Basel III (2011), the EU Directive 95/46 (1995) and its successor[1] together with internal policies force enterprises to audit and control actions within their systems, stressing the importance of secure Identity and Access Management (IAM) (Cleven and Winter, 2009). Yet, due to ineffective IAM, e.g. by manually creating, updating, and deleting digital identities and granting and revoking access rights, employees commonly accumulate excessive access rights over time. As a cornerstone of IAM, role-based user management has become the de facto access control model in most large- and medium-sized companies. In this model, users gain permissions on abstract representations of the systems' physical resources by obtaining membership in roles.

The central challenge after setting up a role model is its operative management and strategic maintenance. Operative role management tasks include routine administration duties like user-role assignments according to the given administration model. Strategic role system maintenance, in contrast, focuses on updating and cleansing role misconfigurations, discarding outdated, i.e. no longer needed roles, and defining new roles. Despite the fact that reports like the Ponemon Cyber Crime Study (2012) emphasize the importance of implementing strategic policies and procedures for controlling access control structures, only little research up to now addresses the challenge of role system maintenance. Current research predominantly investigates the optimization of role model structures on a mathematical and technical level rather than providing an applicable process for role maintenance. As a result, current solutions are not applicable in practice when administrators are requested to control and update historically grown role models in order to minimize security vulnerabilities and compliance violations. They do not offer the required process guidance and automation that is mandatory for successfully keeping large-scale role systems up to date in practice.

In order to overcome the existing limitations, this paper introduces the Role Optimization Process Model (ROPM). Following the design science approach (Vaishnavi and Kuechler, 2007), it suggests an iterative process for improving role definitions together with a prototypical tool implementation as the main artefacts of our research. The ROPM considers company-specific quality criteria and integrates expert knowledge. To the best of our knowledge, no such process model has been proposed up to now. It has been designed based on previous academic work as well as on experience gathered during our participation in several industry projects. In order to underline its applicability and automate role optimization activities we extended the role-modeling tool proposed in Fuchs et al. (2013) with ROPM functionality. The tool itself can be connected to application systems and analyses relevant identity data ex-post. It provides standard connectors for widely used application systems like, LDAP services, IAM tools, or SAP environments and additionally offers generic connectivity using standards like SPML or file-based data exchange protocols. Extending an existing tool allowed us to facilitate available functionality (e.g. data import or data visualization) and further evaluate our process model within a real-life project (see Section 4).

The remainder of the paper is structured as follows. In Section 2, an overview of related work is presented before Section 3 proposes the Role Optimization Process Model. An evaluation using real-world data in Section 4 underlines its applicability while Section 5 provides a summary and outlook for future work.

[1] The General Data Protection Regulation currently under discussion by the European Parliament. Expected adoption in 2014.

2 Related Work

One big challenge for enterprises that need to address their security vulnerabilities through insider access is the management of digital identities (Hovav and Berger, 2009). In today's medium to large-sized companies Role-based Access Control (RBAC) has become the de-facto standard for controlling user access to resources. Recent surveys underline the growing importance of RBAC (Fuchs et al., 2011). The RBAC model family published by Sandhu et al. (1996) defines sets of elements, relations, and functions of an RBAC state (Figure 1). In its minimal configuration (Core RBAC), it includes users, roles, objects (OBS), operations (OPS), permissions (PRMS), and a set of sessions. Permission assignments (RPA) assign certain roles to specific permission(s). By obtaining membership in roles via user-role assignments (URA), users gain permissions on representations of the systems' physical resources. Hierarchical RBAC extends Core RBAC by including inheritance relationships among roles (RH) while Constrained RBAC adds dynamic (DSD) and static (SSD) separation of duty relations in order to enforce conflict of interest policies. RBAC as described above was adopted as an ANSI standard in 2004 (ANSI INCITS, 2004).

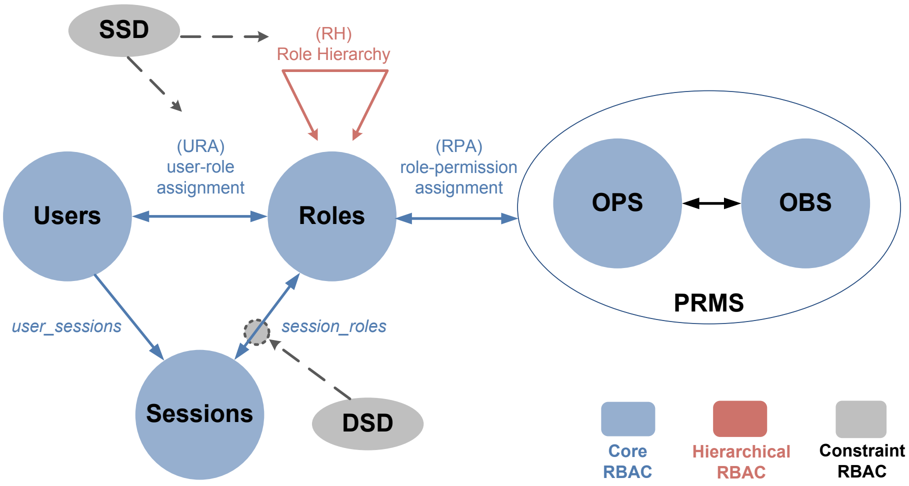


Figure 1: RBAC model family

Defining a valid set of roles has emerged as the initial task during the migration from an identity-based to a role-based user management (Coyne, 1996). In practice, large application systems commonly require the definition of hundreds of roles. Existing approaches for role definition can be characterized by the input data they are based on (Fuchs et al., 2011): Top-down role engineering techniques define roles based on employees' job descriptions, business processes, and organizational structuring. Neumann and Strembeck (2002), e.g., focus on scenario-driven role engineering. Role mining, on the contrary, has developed as a tool-based approach for defining roles by applying association rule mining techniques to access privilege assignments. Molloy et al. (2009) provided an evaluation of existing role mining algorithms. However, as role mining is a mainly technical approach lacking the integration of business semantics, practitioners and researchers have agreed upon the fact that a hybrid combination of role engineering and role mining is required for effective role development in the context of enterprise-wide security management (Fuchs et al., 2011).

After the initial setup of a role system, both, its operative and strategic management, become essential tasks (Fuchs and Müller, 2009). Operative Role Management includes routine administration duties like user account creation or role assignments upon user requests. A number of administration models have been developed in order to address the resulting tasks. Their main objectives include the decentralization of administrative competence, the autonomy of administration, and the control of irregularities. Prominent examples include the ARBAC model family initially proposed by Sandhu et al. (1999) or the SARBAC model family initially proposed by Crampton and Loizou (2003).

Strategic role maintenance, on the contrary, deals with the management of the role definitions. Compared to operative role system management, only a small amount of research work has been conducted in that area. Kern et al. (2002) and Fuchs et al. (2008) both analyzed the lifecycle of role systems in general. Vaidya et al. (2008) focused on role optimization and defined the so-called Minimal Perturbation Role Mining Problem for finding an optimized RBAC state. Nevertheless, their approach is depending on the role quality and struggles with misconfigured roles. In Molloy et al. (2008), the authors investigated the usage of quality metrics like the Weighted Structural Complexity (WSC) for analyzing role system states. Fuchs and Müller (2009) described mechanisms for periodically evaluating a role systems' quality. However, they do not consider the scalability of their approach in large real-world scenarios. Takabi and Joshi (2010) presented the StateMiner algorithm for finding an optimal role hierarchy. StateMiner, however, is restricted to the distribution of permissions via roles and, similar to Vaidya et al. (2008), requires a high input quality of the current RBAC state. Molloy et al. (2010) suggest applying role mining mechanisms and replacing role objects with user objects. Similarly, Giblin et al. (2010) compare a business-driven generated and a tool-mined set of proposed roles. Yet, both approaches do not focus on providing a structured process for role system maintenance. Giblin et al. (2010), for instance, only mention the general need for applying role development mechanisms to maintenance duties.

This paper improves the state of the art by proposing a process model for role optimization, which structures the required activities and provides tool-support. By doing so, it integrates and combines previous findings from the areas of role mining, data cleansing, as well as role maintenance. In contrast to the currently available approaches, it focuses on the comprehensive and tool-supported process of role maintenance instead of only providing a solution to one specific maintenance task.

3 The Role Optimization Process Model

The previous findings underlined the importance of role-based user management in general and role system maintenance in specific. It has been shown that current approaches like Molloy et al. (2010), Giblin et al. (2010), or Takabi and Joshi (2010) do not offer the guidance required when companies aim at ensuring the correctness of role definitions. This part of the paper improves the current situation by introducing the Role Optimization Process Model (Figure 2). ROPM consists of four phases that structure the necessary activities for role system maintenance and provides comprehensive tool-support for automating tasks and fostering human understanding. Its main characteristics are:

- Rating the current role model quality and using the results during later role optimization
- Providing tool support in order to guide human decision making and automate optimization tasks
- Enabling decision makers to integrate a business-oriented perspective
- Minimizing optimization efforts by building on an iterative and selective improvement process

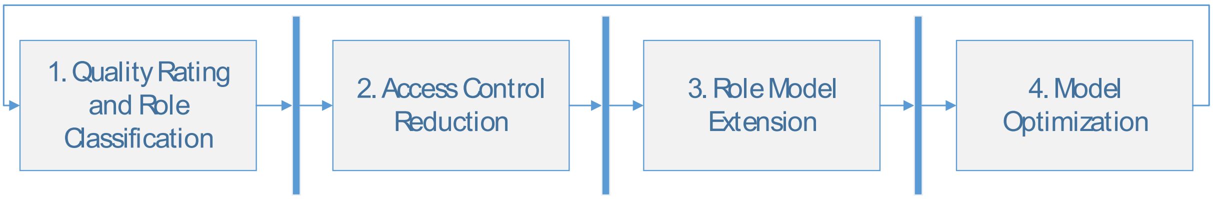

1. Quality Rating and Role Classification
2. Access Control Reduction
3. Role Model Extension
4. Model Optimization

Figure 2: The Role Optimization Process Model

During phase 1 of the ROPM, a quality rating and role classification are carried out based on defined quality metrics. Subsequently, the number of URA, RPA and UPA is reduced by eliminating outdated and erroneous assignments (phase 2). The cleansed data is then investigated for potential URA and RPA extension while remaining UPA are re-modeled on the basis of new roles (phase 3). During phase 4, the optimization of role hierarchies as well as the establishing of preventive control mechanisms for role system control take place. Following previous research findings, the ROPM phases commonly are executed iteratively in order to minimize the resulting disruption of organizational processes (Vaidya et al., 2008). They additionally can be run in multiple instances. Quality measurement activities between phase transitions (lines between the boxes) enable human actors to decide about further process execution.

In the following, we are going to describe the core elements of each ROPM phase. In order to foster the reader's understanding, we employ an artificial example of a typical operations department within a large company, consisting of 50 employees with their respective user accounts in several local applications. Their resource access is managed via seven different business roles (URA, RPA) and several directly assigned permissions (UPA).

3.1 Quality Rating and Role Classification

During phase 1 of the ROPM, metrics setting the general conditions for the role optimization are defined and a quality rating of current roles together with a role classification are executed (Figure 3).

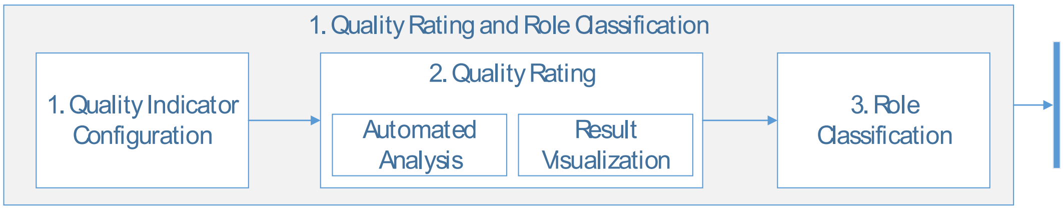


Figure 3: Quality Rating and Role Classification

Prior to the actual role optimization, a human role engineer has to assess available metrics in the given application scenario (Activity 1.1). In general, model-specific and role-specific quality criteria can be differentiated. While the former rate the role model as a whole, the latter focus on the quality of single roles. Metrics like the current number of roles defined and their achieved assignment coverage, the WSC, or expert knowledge about the historic development of the role model are examples for model-specific quality indicators. Role-specific quality criteria like the size of a single role or its similarity to other roles, on the contrary, focus on the inspection of selected roles. Considering the artificial example introduced earlier, overlapping of the defined seven business roles can point at potential for optimizations. Furthermore, available business semantics like an existing classification of role types (i.e. standard roles, expert roles, critical roles) can be considered. Some companies might aim at the definition of fewer but large roles for an increased automation of access control provisioning processes while others might aim at the definition of dedicated expert roles for security reasons.

The quality indicators serve as input for the quality rating of the existing role model (Activity 1.2). Even though a basic rating can be executed manually, this step in practice needs to be supported by an analysis tool. Role mining techniques, for instance, can be used to create a role model by dissolving all existing roles into UPA and compare it to the current role model. The example in Figure 4 compares the assignment coverage of the current role model (lower line) with such a pseudo role configuration (upper line). The x-axis depicts both role types sorted descending according to their size (URA*RPA). The summarized level of assignment coverage (in aggregated % of the total assignments) is displayed on the y-axis. The farther apart both lines are, the lower the quality of the current roles can e.g. be interpreted in case an organization aims at maximizing role sizes and assignment coverage for improving automated IAM processes. Additionally, a slowly increasing current coverage (lower line) reveals a large number of small roles which hardly contribute to the overall assignment coverage. Aiming at minimizing the number of roles, this hints at roles that need to be investigated for deletion.

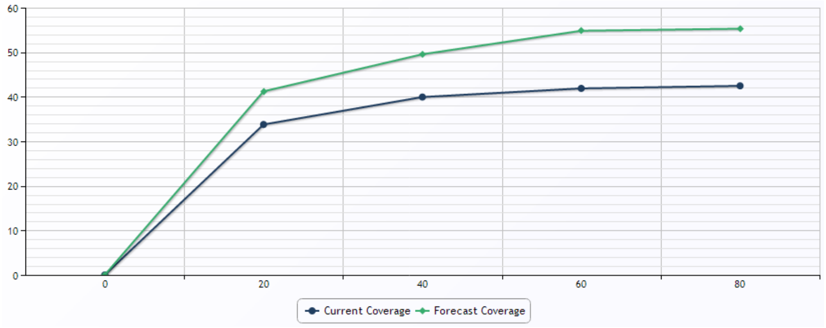


Figure 4: Role model quality rating based on role size

As a last preparatory activity, a classification of existing roles based on the selected quality ratings takes place (Activity 1.3). Human role engineers mark high-quality roles kept for further optimization and differentiate them from roles that should be discarded and re-modeled (commonly low-quality roles). This allows for an iterative role improvement: On the one hand, roles that are an integral part of the current role model are likely to be kept within an improved role model. On the other hand, roles that are potentially outdated or erroneous can be discarded and re-modeled. In the given example, two out of the seven roles might be identified as the standard roles assigned to every employee in the operations department. A human role engineer could decide that they should remain unchanged ant thus need to be excluded from further optimization. In case of existing role hierarchies as introduced in Hierarchical RBAC, a hierarchy deconstruction is recommended. As the discarding of parent roles also affects the RPA of every child role, a flat structure reduces complexity during the subsequent optimization phases. However, the ROPM enables human role engineers to decide whether this decomposition should take place or not. In case hierarchies remain and parent roles are discarded, a child role loses all permission assignments inherited by the respective RPA.

3.2 Access Control Reduction

After the preparatory phase 1, the reduction of existing assignments by revoking unused and excessive privilege assignments takes place (Figure 5). At first (Activity 2.1), the assignments currently managed via roles are reduced by revoking unused or outdated assignments of both, accounts (URA) and permissions (RPA). This needs to be tool-supported due to the potentially high number of assignments.

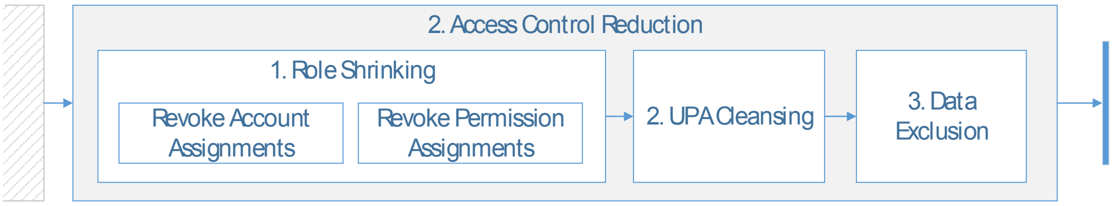

2. Access Control Reduction
1. Role Shrinking
Revoke Account Assignments
Revoke Permission Assignments
2. UPA Cleansing
3. Data Exclusion

Figure 5: Access Control Reduction

Applying automated data analysis allows for a timely interpretation of usage statistics, the identification of empty and orphan roles, or the detection of anomalies within role hierarchies. Additionally, log files containing usage information (e.g. constraint data about usage times of access privileges as introduced by temporal-RBAC approaches (like Bertino et al., 2001) or usage frequencies) can be considered during this activity. Researchers have already underlined the applicability of such approaches in application-specific scenarios (Baumgrass, 2011). Note that exporting log data can be a labor-intensive task, e.g. in case of a centrally managed LDAP directory which controls access to a large number of local file servers where the actual log files are written. In addition to the tool-supported analysis, a manual investigation by role engineers can be carried out in order to identify excessive or untypical role assignments. In large environments, however, this is a time consuming task, which commonly can only be executed to a very limited extent.

One example of tool-supported role shrinking is the application of the so-called Access Grid proposed in (Fuchs et al. 2013) (see Figure 6). It visualizes user accounts representing employees (rows) and their assigned permissions (columns). A colored element indicates an existing assignment of a permission to an account. Additionally, the defined roles can be highlighted manually or automatically, improving human understanding of the given RBAC state. Following the artificial example introduced earlier, Figure 6 depicts two roles (blue and green areas) assigned to five employees each (rows). Mustard coloring highlights direct permission assignments of users that were granted without using roles. Grey highlighting within role areas depicts unused assignments, i.e. the respective users have not used the assigned access privilege during the last period of investigation[2]. In the example, two roles are highlighted. The initial situation reveals anomalies like an employee that has not used a role assignment at all (first row in Figure 6, left side), indicating an inactive employee whose user account should be disabled or revoked. It also highlights RPA that have not been activated by any member of both roles. As a result, these assignments can be removed from the role definitions. The updated role configuration after successful reduction is depicted in the right part of Figure 6.

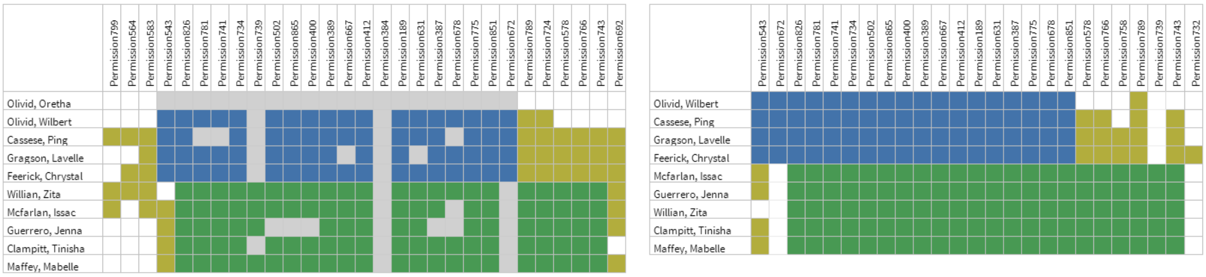


Figure 6: Role shrinking using the Access Grid

Similar to the cleansing of RPA and URA, the existing UPA can be investigated and reduced using the Access Grid in combination with data cleansing technologies (Activity 2.2). This is a mandatory element of every role optimization effort as all remaining UPA are further fed into the consecutive role extension phase of the ROPM. In case erroneous and excessive UPA exist, role modeling is significantly complicated and errors are likely to be transferred into new role definitions.

Concluding the reduction phase, the ROPM allows role engineers to manually exclude certain input data elements from the subsequent role model extension phase (Activity 2.3). This can potentially include all access control components like identities and their user accounts, roles and their hierarchical relationships, or even specific permission assignments. This is important in case certain permissions (e.g. like highly critical) ones shall never be member of roles. Concluding the data reduction activity all remaining assignments act as input for the subsequent steps, either in the form of URA (roles kept in the improved role model) or UPA (discarded roles and existing direct UPA).

[2] Commonly, centrally managed applications allow for such usage monitoring. Project experience shows that the monitoring period should exceed six months in order to capture usage behavior in a representative manner.

3.3 Role Model Extension

After successful model reduction, the expansion of cleansed role definitions and the re-modeling of discarded roles and UPA into new roles takes place (Figure 7). Again, tool-supported detection of potential extensions is eligible at this stage. Even though manual execution in general is possible, it can in practice only act as an addition of the automated proposal of role model extensions. In case several different conflicting role extensions are detected, quality metrics like the highest WSC reduction can be used as decision support mechanism.

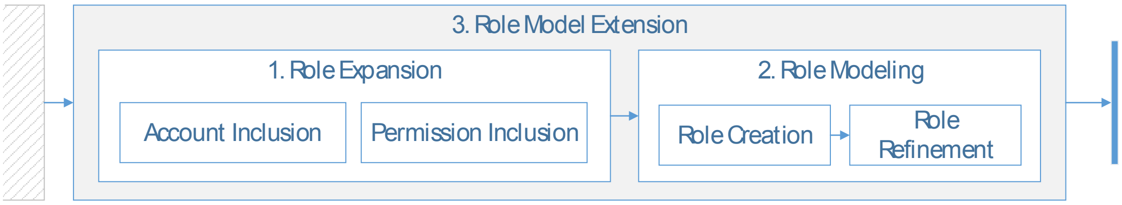


Figure 7: Role Model Extension

During a first activity (Activity 3.1), the cleansed roles are investigated for their potential expansion. An automated analysis can, for instance, highlight user accounts or permissions which can be added to the current set of URA and RPA. Figure 8 highlights an exemplary role (blue colored large area) assigned to 11 employees working in the operations department in the given artificial example. This role can be extended by three permissions (dark green coloring, right side). Additionally, two other employees in the department (lower two rows) might also become role members – even though some of the role's permissions are not assigned to them yet. While this is not desirable in terms of IT security, organizations might grant the assignments in order to achieve a higher degree of automation in their IAM processes. Quality criteria from phase 1 can be facilitated to control such decisions.

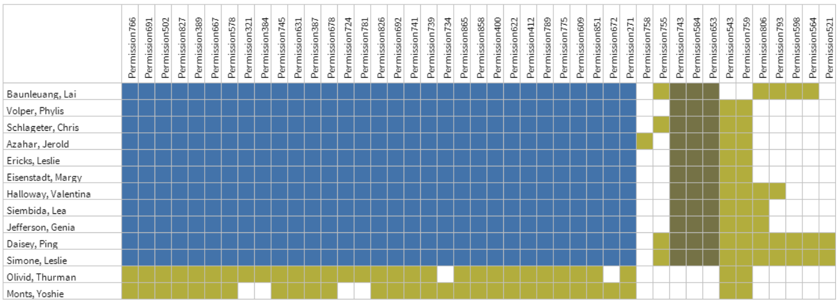


Figure 8: Role expansion using the Access Grid

After the successful extension of roles, the remaining UPA consisting of discarded roles and direct UPA are investigated using standard role development mechanisms (Activity 3.2). Role candidates are generated and investigated by IT- and business experts. In order to further improve detected role candidates, ROPM proposes an automated role refinement comparing the generated role candidates with existing roles as well as previously discarded role definitions. The latter in particular is of interest as discarded roles commonly have been equipped with semantic information like a name, a role owner or any other descriptive information. In case a similar role candidate is identified during the role modeling activity, available semantic annotations of the discarded (but similar) role can be facilitated for an eased definition of the new role. An example would be the assignment of the previous role owner as the initial contact person for role approval.

After successful completion of phase 3, the set of newly defined roles together with the roles excluded from optimization represents new valid role catalogue used as input for phase 4 of the ROPM.

3.4 Model Optimization

Concluding ROPM execution, the set of updated roles and the set of newly generated roles are optimized in terms of model structure and complexity (Figure 9). This embraces the (re-)design of role hierarchies as well as a final model assessment depending on the pre-defined quality criteria. Additionally, the establishment of preventive controls is recommended for reducing the effort of future model optimizations.

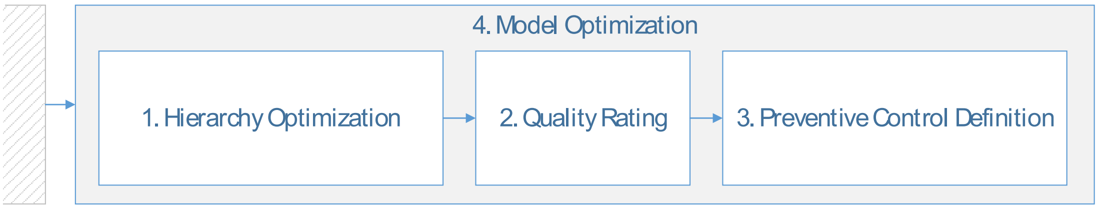


Figure 9: Model Optimization

During the first activity (Activity 4.1), the hierarchy optimization of the gathered sets of roles is carried out. The goal is to merge the two role sets into one optimized role hierarchy. As this challenge has already been thoroughly investigated in literature, the ROPM builds upon available approaches that generate an RBAC state that minimizes a predefined cost measure (e.g. minimizing the number of roles, the number of URA, or the number of RPA). Examples for such approaches include the ones proposed in Zhang et al. (2007), Guo et al. (2008), or Takabi and Joshi (2010). Note that the definition of role hierarchies is an optional step. Despite the advantages of role hierarchies, organizations might skip this activity in order to minimize the role system modeling complexity. Subsequently, a concluding quality rating of the role model takes place (Activity 4.2). Using the metrics applied during previous phases, the final role model is analyzed for suboptimal states. The results of the exemplary optimization process within the operations department could e.g. reveal a higher increased overall role coverage and the reduction of role sizes due to the removal of unused RPA and URA.

Concluding the optimization process, ROPM proposes the establishment of preventive controls for role optimization (Activity 4.3) within a separate control system like the IAM system of an organization. The goal of establishing preventive controls is to reduce the risk of a decreasing role model quality over time. Organizations are likely forced to re-run optimization efforts in case they do not have any established means for periodically controlling role model quality (Fuchs and Müller, 2009). The resulting amount of work can be minimized if preventive controls ensure a timely and appropriate treatment of existing role model quality issues. Examples include tool-based periodic re-certification processes of roles, user accounts or access privilege assignments.

4 Role Optimization in Practice

After having described the ROPM in theory, this Section evaluates its applicability in a real-world scenario, representing a naturalistic ex-post evaluation following the evaluation framework proposed by Pries-Heje et al. (2008). The dataset facilitated stems from the SAP ERP system of a worldwide operating company in the manufacturing industry employing more than 12.000 internal and 4.000 external employees. Note that our analysis is not SAP-specific but the data rather can originate from various other applications like an LDAP directory or proprietary software with a dedicated user management that o. We implemented a conversion tool, which enabled us to automatically extract the required data from the ERP system and import them into our ROPM tool. 12.776 active user accounts and 727 SAP collective roles (from hereinafter referred to as roles) bundling an additional 3.637 SAP single roles (from hereinafter referred to as permissions) were gathered[3]. The number of access privilege assignments sums up to 1.060.757. The ERP system has been introduced about 15 years ago. Since then, a steady extension of its role model has taken place. Current auditing requirements force the company to implement a central IAM infrastructure and cleanse the role definitions. As a result, a role optimization project applying the ROPM was initiated with the following main goals:

- Minimizing excessive UPA and URA by an expert analysis guided by automated quality rating (1)
- Minimizing the number of roles and unused RPA on the basis of usage statistics analysis (2)
- Extending the cleansed roles to increase the degree of automated access privilege provisioning (3)

Requirements (1) and (2) correspond to phase 2 of the ROPM while (3) relates to the role extension phase (phase 3). No definition of new roles was conducted due to the already high role coverage within the system (98%) and the system administrators' decision to rather identify unused and outdated roles. Additionally, no requirements for role model hierarchy definition existed within the project. As a result, the last phase of the ROPM was not executed.

Note that a comparative evaluation of our tool-based approach with manually executing the role optimization process cannot be executed. The main reason is the inapplicability of a manual inspection of the several hundred-thousands of URA, RPA, and UPA. The manual inspection of the given 727 roles including the evaluation of usage statistics as well as the investigation of the mapping state of roles and permissions, e.g., cannot be conducted. Note that even Excel-based approaches applied in role optimization scenarios with a limited focus (e.g. one single department or a small amount of roles) are not capable of analyzing the amount of data given in a reasonable time.

4.1 Quality Rating and Role Classification

At first, company-specific quality criteria were defined by the role engineering experts of our project partner (Activity 1.1). Automated data quality analyses were facilitated for detecting high risk URA and UPA in order to address the first project goal (1). Furthermore, the available usage statistics within the ERP system supported role reduction activities corresponding to the second project goal (2). As an improved degree of automation during role provisioning was one project goal (3), the role coverage metric as introduced in Section 3 additionally was applied during ROPM application. The automated quality rating of the current role model (Activity 1.2) highlighted 1.357 out of the 12.776 active employees with 6.709 potentially erroneous UPA and URA (Figure 10, left side). The role coverage analysis for the 360 largest roles (Figure 10, right side) revealed a high percentage of assignments covered by a very small number of roles (the 20 largest roles cover 66.0% of all assignments[4]). At the same time, a very high number of small roles hardly adds to the assignment coverage (the 367 smallest roles within the system only cover 2% of all assignments). Following the project goal (2), this indicates roles that potentially can be discarded after investigation during phase 2 of the ROPM. The comparison with a pseudo role state generated using role mining techniques was not carried out due to the already high role coverage within the system.

[3] In the following, we facilitate the standard RBAC terminology introduced by Sandhu et al. (1996): As SAP single roles represent the most-fine grained resource object analyzed, containing bundles of operations on SAP-objects, they from hereinafter will be referred to as "permissions" . On the contrary, SAP collective roles assign SAP single roles to user accounts and will therefore be referred to as "roles" following Sandhu et al.'s terminology.

[4] They include the standard roles assigned to every internal and external employee, which grant access to uncritical resources.

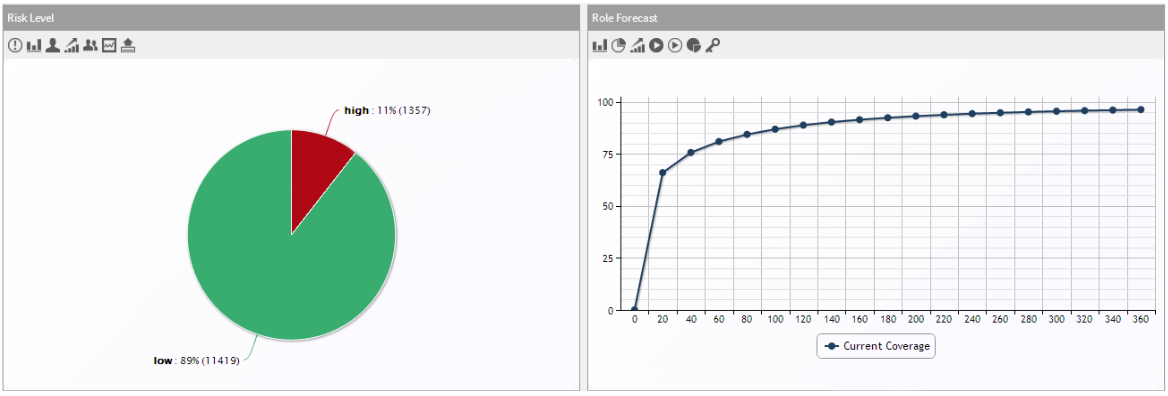


Figure 10: Role model quality rating

The results of the initial data quality assessment gave the human role engineers (in this case the administrators of the ERP system) a comprehensive quality rating of the current system state. Before moving on to the next project phase, they decided to remove the company-specific standard roles assigned to every internal and external employee from further optimization (Activity 1.3). Due to them being not security critical, they were excluded in order to reduce project complexity.

4.2 Access Control Reduction

After successful completion of the first ROPM phase, the reduction of roles was initiated (Activity 2.1). In order to show an appropriate level of detail we are going to focus on the results concerning a specific department consisting of 114 employees assigned to 35 different roles via 263 URA. Additionally, 849 RPA and 115 direct UPA resulting in 6.836 assignments are available. The ROPM tool detected 16 employees with 23 potentially erroneous URA using data cleansing techniques. These URA were investigated by the role engineers and delegated to domain experts in case they were unable to decide about assignment cleansing. Additionally, a manual analysis of all roles for untypical user assignments based on the expert knowledge of the role engineers took place. Due to the naming conventions within the ERP system[5], they were able to cleanse eight additional URA of wrongly assigned roles. In total, 31 out of 263 URA were cleansed in this department (11.8% reduction).

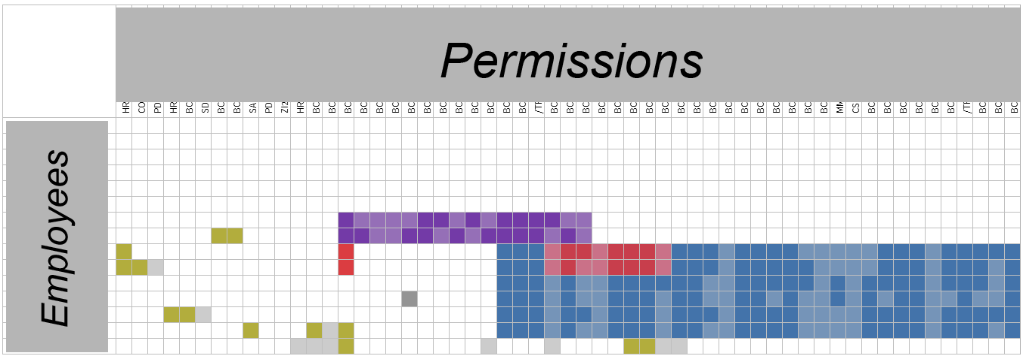


Figure 11: URA reduction and usage statistics analysis (anonymized data)

[5] The naming of each single role and collective role displays their assignment to a functional unit, e.g. FI for Finance roles or PD for Personnel Development roles.

In order to reduce excessive RPA according to the project goal (2), usage statistics from the last six months were presented to the role engineers[6] (Activity 2.1). As aforementioned, available log file information about employees' usage behavior (e.g. concerning time constraints like usage times of access privileges) can be facilitated as additional information during role reduction. In the given use case, however, no detailed log information apart from the statistics of employees concerning their usage of SAP single roles (i.e. permissions) has been available. Figure 11 displays an exemplary part of the department under investigation, highlighting three of the 35 defined roles (blue, red and violet coloring). Unused assignments are displayed using bright coloring within the role areas. It can be seen that several permissions have not been activated by any role member. Additionally, two roles (blue and red coloring) are overlapping to a high degree with only one permission of the smaller role not being included in the larger role (blue coloring). The role engineers thus decided to delete the small role and assigned the uncovered RPA directly as UPA to the respective user accounts. Similar to the URA reduction, untypical UPA have been investigated during the cleansing process (Activity 2.2). However, due to the high role coverage in the system (98%), these direct UPA represent only a very small part of the access control state.

Summing up, the tool-based analysis of usage statistics resulted in the detection of 8.543 unused RPA out of 25.939 RPA for the complete ERP system. Nearly every third permission assignment to a role (32.9%) was affected, resulting in a significant role shrinking increasing the role model quality while at the same time reducing the risk of malicious privilege usage. 674.094 privilege assignments (RPA and UPA) out of the extracted 1.060.757 was highlighted as unused within the ERP system. This high number hints at erroneous role definitions and user assignments where employees inherit access privileges not required for their daily work. Note that this also stems from RPA that are used by several, but not all of the assigned employees (see Figure 11, bright colored elements within the role areas). Those assignments cannot be removed from the role definitions as they are still required by at least one role member.

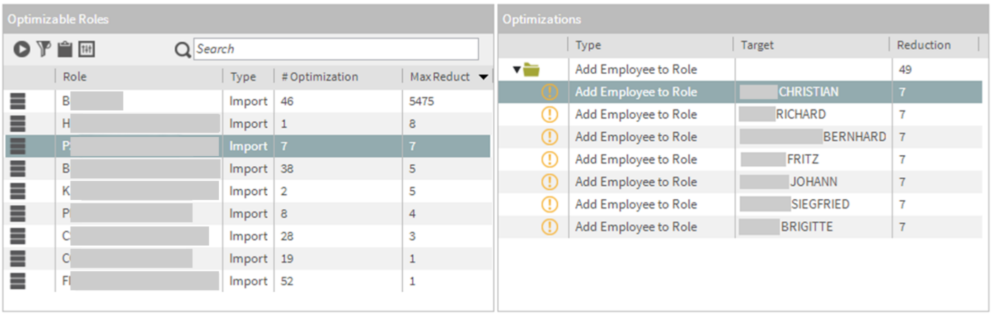
Concluding phase 2 (Activity 2.3), the role engineers decided to exclusively include personal accounts of employees throughout the role extension phase, removing 82 technical accounts from the input data. The main reason for this decision was the project's exclusive focus on personal accounts.

4.3 Role Model Extension

During the third ROPM phase, the existing roles were investigated for potential membership inclusions. Employees directly assigned to permissions, which are managed via roles, were detected (Activity 3.1). The left side of Figure 12 shows the exemplary results within the department under investigation: Nine of the 35 current role definitions were marked for potential extension. The detailed analysis for one selected role on the right side of Figure 12 highlights seven employees which should potentially be assigned to this role. Furthermore, the reduction of the role model complexity (WSC) is depicted using the *MaxReduct* and *Reduction* columns. Optimizing the uppermost role (left side), for instance, results in a significant reduction due to the high number of proposed optimizations (46).

In total, extensions for 101 roles within the ERP system investigated during phase 3 of the ROPM were automatically suggested. Similar to the role cleansing, these suggestions were further analyzed by the human role engineers and optionally by responsible domain experts.

[6] Note that in SAP systems the usage of transactions is recorded. In case one transaction is included in several of a user's roles, all respective roles are marked as used.



Role	Type	# Optimization	Max Reduct
B[redacted]	Import	46	5475
H[redacted]	Import	1	8
P[redacted]	Import	7	7
B[redacted]	Import	38	5
K[redacted]	Import	2	5
P[redacted]	Import	8	4
C[redacted]	Import	28	3
C[redacted]	Import	19	1
Fl[redacted]	Import	52	1

Type	Target	Reduction
Add Employee to Role		49
Add Employee to Role	[redacted]CHRISTIAN	7
Add Employee to Role	[redacted]RICHARD	7
Add Employee to Role	[redacted]BERNHARD	7
Add Employee to Role	[redacted]FRITZ	7
Add Employee to Role	[redacted]JOHANN	7
Add Employee to Role	[redacted]SIEGFRIED	7
Add Employee to Role	[redacted]BRIGITTE	7

Figure 12: Role extension recommendations (anonymized data)

In summary, the naturalistic evaluation based on data from a large industrial ERP system presented in this part of the paper underlined the applicability of the ROPM for structured role system optimization. Based on the ROPM tool implementing quality rating and role optimization functionality, we were able to reveal a large number of erroneous role definitions. In the company, this increased management attention and gave in-depth insight into the current role structures and their quality. By supporting role reduction and role expansion activities, the ROPM was furthermore able to assist the human role engineers and domain experts during their tasks of closing security vulnerabilities and securing compliance within the ERP system.

5 Conclusions and Future Work

Over the last decades, role-based user management has become the de-facto standard in medium- and large-sized enterprises for managing employees' access to protected resources. In this paper, we have underlined the importance of the strategic management of role models in order to keep role definitions up to date and minimize security vulnerabilities and compliance violations. Up to now, only little work has been carried out in that area and no comprehensive process model for structuring the required tasks during role system optimization has been proposed. Our work has closed this existing gap by introducing ROPM, a process model that allows for a guided analysis and improvement of role definitions without disrupting organizational processes. Throughout its four phases, ROPM rates the quality of the existing roles, and subsequently performs role reduction and role extension activities in order to minimize errors in the role configurations and thereby improves the overall role system quality. By offering tool support, it automates various process steps and thus furthermore increases its applicability in practical scenarios. In contrast to other approaches, it integrates an IT-oriented as well as business-related perspective and offers a comprehensive and tool-supported process structuring the tasks during role system maintenance. By presenting results from a naturalistic case study of a large industrial company, we highlighted the applicability of the ROPM in practice.

As future work, we plan to investigate the application of our proposed model during initial role development where a steady control of role definitions can be applied to reduce the project complexity and to increase the quality of the role model. Overlapping roles or misconfigurations might be cleansed before role activation within application systems. Additionally, we aim at investigating the application of ROPM in scenarios with complex role hierarchies. In those cases, interdependencies due to inheritance relationships need to be thoroughly considered during role optimization in order to correctly rate and improve role definition quality.

AcknowledgementThe research leading to these results was supported by the “Bavarian State Ministry of Education, Science and the Arts” as part of the FORSEC research association.

References

ANSI INCITS (2004). American National Standard for Information Technology – Role Based Access Control. ANSI INCITS 359-2004.

Basel III (2011). A global regulatory framework for more resilient banks and banking systems - revised version June 2011. Retrieved December 6, 2013 from http://www.bis.org/publ/bcbs189.pdf.

Baumgrass, A. (2011). Deriving Current State RBAC Models from Event Logs. In Proceedings of the 6th International Conference on Availability, Reliability and Security (ARES), 667-672, Vienna, Austria.

Bertino, E., Bonatti, P. E., Ferrari, E. (2001). TRBAC: A Temporal Role-based Access Control Model. ACM Transactions on Information and System Security (TISSEC), 4, 191-233, New York, NY, USA.

Cleven, A. and Winter, R. (2009). Enterprise, Business-Process and Information Systems Modeling. Regulatory Compliance in Information Systems Research - Literature Analysis and Research Agenda, 29, 174–186.

Coyne, E.J. (1996). Role Engineering. In Proceedings of the 1st ACM Workshop on Role-based Access Control (RBAC), Gaithersburg, MD, USA.

Crampton, J. and Loizou, G. (2003). Administrative Scope: A Foundation for Role-based Administrative Models. ACM Transactions on Information and System Security (TISSEC), 6, 201-231, New York, NY, USA.

European Union (1995). Directive 95/46/EC of the European Parliament and of the Council. Official Journal of the European Communities L (28-31). Retrieved December 6, 2013 from http://www.icm2006.org/imgs/congresos/Directive%2095%2046%20EC.pdf.

Fuchs, L., Meier, S. and Pernul, G. (2013). Managing the Access Grid - A Process View to Minimize Insider Misuse Risks. In 11. Internationale Tagung Wirtschaftsinformatik (WI), Leipzig, Germany.

Fuchs, L. and Müller, C. (2009). Automating Periodic Role-Checks - A Tool-based Approach. In 9. Internationale Tagung Wirtschaftsinformatik (WI), 803-814, Vienna, Austria.

Fuchs, L. and Pernul, G. (2008). ProROLE: A Process-Oriented Lifecycle Model for Role Systems Leveraging Identity Management and Guiding Role Projects. In Proceedings of the 16th European Conference on Information Systems (ECIS), 1322-1333, Galway, Ireland.

Fuchs, L., Pernul, G. and Sandhu, R.S. (2011). Roles in Information Security – A Survey and Classification of the Research Area. Computers & Security 30(8), 748-769.

Giblin, C., Graf, M., Karjoth, G., Wespi, A., Molloy, I., Lobo, J. and Calo, S. B. (2010). Towards an Integrated Approach to Role Engineering. In Proceedings of the 3rd ACM Workshop on Assurable and Usable Security Configuration (SafeConfig), 63-70, Chicago, IL, USA.

Guo, Q.,Vaidya, J. and Atluri, V. (2008). The Role Hierarchy Mining Problem: Discovery of Optimal Role Hierarchies. In Proceedings of the 2008 Annual Computer Security Applications Conference, 237–246, Washington, DC, USA.

Hovav, A. and Berger, R. (2009). Tutorial: Identity Management Systems and Secured Access Control. Communications of the Association of Information Systems (CAIS) 25 (42), 531-570.

Kern A., Kuhlmann, M., Schaad, A. and Moffett, J.D. (2002). Observations on the Role Life-Cycle in the Context of Enterprise Security Management. In Proceedings of the 7th ACM Symposium on Access Control Models and Technologies (SACMAT), 43-51, Monterey, CA, USA.

Molloy, I., Chen, H., Li, T. Wang, Q., Li, N., Bertino, E., Calo, S. and Lobo, J. (2008). Mining Roles with Semantic Meanings. In Proceeding of the 13th ACM Symposium on Access Control Models and Technologies (SACMAT), 21-30, Estes Park, CO, USA.

Molloy, I., Chen, H., Li, T., Wang, Q., Li, N., Bertino, E., Calo, S. and Lobo, J. (2010). Mining Roles with Multiple Objectives. ACM Transactions on Information and System Security (TISSEC), 13 (36), New York, NY, USA.

Molloy, I., Li, N., Li, T., Mao, Z., Wang, Q. and Lobo, J. (2009). Evaluating Role Mining Algorithms. In Proceedings of the 14th ACM Symposium on Access Control Models and Technologies (SACMAT), 95-104, Stresa, Italy.

Neumann, G. and Strembeck, M. (2002). A Scenario-driven Role Engineering Process for Functional RBAC Roles. In Proceedings of the 7th ACM Symposium on Access Control Models and Technologies (SACMAT), 33–42, Monterey, CA, USA.

Ponemon Institute (2012). Cost of Cyber Crime Study: United States. Ponemon Institute. Retrieved December 6, 2013 from http://www.ponemon.org/local/upload/file/ 2012_US_Cost_of_Cyber_Crime_Study_FINAL6%20.pdf.

Pries-Heje, J., Baskerville, R., Venable, J. (2008). Strategies for Design Science Research Evaluation. In Proceedings of the 16th European Conference on Information Systems (ECIS), 255-266. Galway, Ireland.

Sandhu, R.S., Bhamidipati, V. and Munawer, Q. (1999). The ARBAC97 Model for Role-based Administration of Roles. ACM Transactions on Information and System Security (TISSEC), 2, 105-135, New York, NY, USA.

Sandhu R.S., Coyne, E.J., Feinstein, H.L. and Youman, C.E. (1996). Role-Based Access Control Models. IEEE Computer, 29(2), 38-47.

Sarbanes, P.S. and Oxley, M. (2002). Sarbanes-Oxley Act of 2002, also known as the "Public Company Accounting Reform and Investor Protection Act of 2002". Retrieved December 3, 2013 from http://www.sec.gov/about/laws/soa2002.pdf.

Takabi, H. and Joshi, J. B. (2010). StateMiner: An Efficient Similarity-based Approach for Optimal Mining of Role Hierarchy. In Proceedings of the 15th ACM Symposium on Access Control Models and Technologies (SACMAT), 55-64, Pittsburgh, PA, USA.

Vaidya, J., Atluri, V, Guo, Q. and Adam, N. (2008). Migrating to Optimal RBAC with Minimal Perturbation. In Proceedings of the 13th ACM Symposium on Access Control Models and Technologies (SACMAT), 11–20, Estes Park, CO, USA.

Vaishnavi, V.K. and Kuechler, W.J. (2007). Design Science Research Methods and Patterns: Innovating Information and Communication Technology. Auerbach Publications, New York, NY, USA.

Zhang, D., Ramamohanarao, K. and Ebringer, T. (2007). Role Engineering using Graph Optimisation. In Proceedings of the 12th ACM Symposium on Access Control Models and Technologies (SACMAT), 139–144, Sophia Antipolis, France.