Mining Business-Relevant RBAC States through Decomposition

Alessandro Colantonio[1,2], Roberto Di Pietro[2],
Alberto Ocello[1], and Nino Vincenzo Verde[2]

[1] Engiweb Security, Roma, Italy
{alessandro.colantonio,alberto.ocello}@eng.it
[2] Università di Roma Tre, Roma, Italy
{colanton,dipietro,nverde}@mat.uniroma3.it

Abstract. Role-based access control is widely accepted as a best practice to effectively limit system access to authorized users only. To enhance benefits, the role definition process must count on business requirements. Role mining represents an essential tool for role engineers, but most of the existing techniques cannot elicit roles with an associated clear business meaning. To this end, we propose a methodology where the dataset is decomposed into smaller subsets that are homogeneous from a business perspective. We introduce the ENTRUSTABILITY index that provides, for a given partition, the expected uncertainty in locating homogeneous set of users and permissions that are manageable with the same role. Therefore, by choosing the decomposition with the highest ENTRUSTABILITY value, we most likely identify roles with a clear business meaning. The proposed methodology is rooted on information theory, and experiments on real enterprise data support its effectiveness.

1 Introduction

Among access control models proposed in the literature, *Role-Based Access Control* (RBAC) [1] is presumably the most adopted by large-size organizations. Within an organization, roles are created for various job functions. Permissions to perform certain operations are assigned to specific roles. Members or other system users are assigned particular roles, and through those role assignments acquire the permissions to perform particular system functions. The main benefit of adopting such a model is a simplification of the security policy definition task by business users who have no knowledge of IT systems. Further, use of roles minimizes system administration effort due to the reduced number of relationships required to relate users to permissions [4].

Despite the benefits derived from deploying role-based access control systems, many organizations are reluctant to adopt them, since there are still some important issues that need to be addressed. In particular, roles must be customized to capture the needs and functions of the organization. For this reason, the *role engineering* discipline [9] has been introduced. However, choosing the best way to design a proper set of roles is an open problem. Various approaches to role engineering have been proposed, which are usually classified as: *top-down* and *bottom-up*. The former requires a deep analysis of business processes to identify which access permissions are necessary to carry out

K. Rannenberg, V. Varadharajan, and C. Weber (Eds.): SEC 2010, IFIP AICT 330, pp. 19–30, 2010.

specific tasks. The latter seeks to identify de facto roles embedded in existing access control information. Indeed, companies which plan to go for RBAC usually find themselves with a collection of several legacy and standard security systems on different platforms that provide "conventional" access control [13]. The bottom-up approach has attracted the attention of many researchers, since it can be easily automated [15]. Data mining technology is typically used to discover roles within access control data. For this reason, the term *role mining* is often used as a synonym of bottom-up. However, the slavish application of standard data mining approaches to role engineering might yield roles that are merely a set of permissions, namely with no connection to the business practices. Indeed, organizations are unwilling to deploy roles they cannot bind to a business meaning [4]. For this reason, bottom-up should be used in conjunction with top-down, leading to an *hybrid* approach.

Only few recent works value business requirements in role mining [4,2,11,14]. Their main limitation is to propose theoretical frameworks that are difficult to apply in real cases. For instance, [4,2,14] require analysts to define a measure for the business meaning of roles. However, selecting the measure that fits the needs of an organization is not trivial, and no best practices exist to define it. In [11] the authors offer a probabilistic model: to find a set of roles that approximates the role mining problem (by considering potentially exceptional user-permission assignments); and, to contextually propose roles that likely are meaningful (by taking into account the relevance of each business attribute). However, there is no guarantee that the introduced approximation is licit. In any case, to our knowledge there is no proposal in the current literature to leverage business-related information in existing role mining algorithms. To this end, a possible viable solution may be *to restrict the analysis to sets of data that are homogeneous from an enterprise perspective*. The key observation is that users sharing the same business attributes will essentially perform the same task within the organization. Suppose we know, from a partial or coarse-grained top-down analysis, that a certain set of users perform the same tasks, but the analysis lacks information about which permissions are required to execute these tasks. In this scenario, restricting role mining techniques to these users only—instead of analyzing the organization as a whole—, will ensure that elicited roles are only related to such tasks. Consequently, it will be easier for an analyst to assign a business meaning to the roles suggested by the bottom-up approach. Moreover, elicitation of roles with no business meaning can be avoided by grouping users that perform similar tasks together first, and then analyzing each group separately. Indeed, investigating analogies among groups of users that perform completely different tasks is far from being a good role mining strategy [4]. Partitioning data also introduces benefits in terms of execution time of role mining algorithms. Indeed, most role mining algorithms have a complexity that is not linear compared to the number of users or permissions to analyze [2,20,10]. To apply this divide-and-conquer strategy, a lot of enterprise information can be used. Business processes, workflow tasks, and organization unit trees are just a few examples of business elements that can be leveraged. Notice that very often such information is already available in most companies before starting the role engineering task—for instance within HR systems. When dealing with information from several sources, the main problem is thus ascertaining which information induces the partition that improves the role engineering task the most.

To address all the abovementioned issues, this paper proposes a methodology that helps role engineers to leverage business information during the role mining process. In particular, we propose to divide the access data into smaller subsets that are homogeneous according to a business perspective, instead of performing a single bottom-up analysis on the entire organization. This eases the attribution of business meaning to roles elicited by any existing role mining algorithm and reduces the problem complexity. To select the business information that induces the most suitable partition, an index referred to as "ENTRUSTABILITY" (***ent****ropy-based* ***r****ole* ***us****efulness predic****tability***) is identified. Rooted on information theory, this index measures the expected uncertainty in locating a homogeneous set of users and permissions that can be managed as a whole by a single role. The decomposition with the highest ENTRUSTABILITY value is the one that most likely leads to roles with a clear business meaning. Several examples illustrate the practical implications of the proposed methodology and related tools, which have also been applied on real enterprise data. Results support the quality and viability of the proposal.

The remainder of the paper is organized as follows: Section 2 reports on related work, while Section 3 introduces the background required to formally describe the proposed tools. The ENTRUSTABILITY index and the proposed methodology are introduced and discussed in Section 4, while their viability is demonstrated in Section 5 by testing on real data. Finally, Section 6 provides concluding remarks.

2 Related Work

Role engineering was first illustrated by Coyne [9] from a top-down perspective. Many other authors sought to leverage business information to design roles by adopting a top-down approach such as [17, 16]. These works represent pure top-down approaches—they do not consider existing access permissions. Hence, they do not take into account how the organization actually works. As for the bottom-up approach, Kuhlmann et al. [13] first introduced the term "role mining", trying to apply existing data mining techniques to elicit roles from existing access data. After that, several algorithms explicitly designed for role engineering were proposed [18,20,10,21,11]. Several works prove that the role mining problem is reducible to many other well-known NP-hard problems, such as clique partition, binary matrix factorization, bi-clustering, graph vertex coloring [6] to cite a few. The main limitation of these works is that they do not always lead to meaningful roles from a business perspective. Colantonio et al. [2] first presented an approach to discover roles with business meanings through a role mining algorithm. A cost function is introduced as a metric for evaluating a "good" collection of roles. By minimizing the cost function it is possible to elicit roles that contextually minimize the overall administration effort and fit the needs of an organization from a business perspective. Further improvements of this approach are [4, 3]. A similar approach is provided by Molloy et al. [14], that employs user attributes to provide a measurement of the RBAC state complexity. Frank et al. [11] proposed a probabilistic model to find a set of roles that contextually approximate the role mining problem and that are likely meaningful. However, as stated in the previous section, all these methods are difficult to apply in real cases.

A tool that is widely used in information theory, and employed in this paper, is *entropy*. In data mining, recent works seek to apply the entropy concept to find all subsets of attributes that have low complexity. Heikinheimo et al. [12] proposed to find low-entropy itemsets from binary data in lieu of frequent itemsets. However, this model is not suitable for the role mining problem, since low-entropy sets are symmetric compared to '0' (missing user-permission assignment) and '1' (existing assignment), while roles can be seen as patterns only made up of 1's. Tatti [19] considered the problem of defining the significance of an itemset in terms of the expected frequency. The main goal is to discover different types of biclusters in the presence of noise. Finally, Frank et al. [11] is the only work that seeks to apply the entropy concept to role mining. In particular, they measure the missing information on whether a given permission is granted to a user. In turn, this information is used to extend their probabilistic model to role mining. However, the authors only provide a way to evaluate each business information against single permissions, and the proposed model is not applicable to other role mining algorithms.

3 Background

Before introducing the required formalism used to describe role engineering, we first review some concepts of the ANSI/INCITS RBAC standard [1] needed for the present analysis. For the sake of simplicity, we do not consider sessions, role hierarchies or separation of duties constraints in this paper. In particular, we are only interested in the following entities:

- $PERMS$, $USERS$, and $ROLES$ are the sets of all access permissions, users, and roles, respectively;
- $UA \subseteq USERS \times ROLES$, is the set of all role-user relationships;
- $PA \subseteq PERMS \times ROLES$, is the set of all role-permission relationships.

The following functions are also provided:

- $ass\_users\colon ROLES \to 2^{USERS}$ to identify users assigned to a role. We consider it as derived from UA, that is $ass\_users(r) = \{u \in USERS \mid \langle u, r\rangle \in UA\}$.
- $ass\_perms\colon ROLES \to 2^{PERMS}$ to identify permissions assigned to a role. We consider it as derived from PA, that is $ass\_perms(r) = \{p \in PERMS \mid \langle p, r\rangle \in PA\}$.

In addition to RBAC concepts, this paper introduces other entities required to formally describe the proposed approach. In particular, we define:

- $UP \subseteq USERS \times PERMS$, the existing user-permission assignments to analyze;
- $perms\colon USERS \to 2^{PERMS}$, the function that identifies permissions assigned to a user. Given $u \in USERS$, it is defined as $perms(u) = \{p \in PERMS \mid \langle u, p\rangle \in UP\}$.
- $users\colon PERMS \to 2^{USERS}$, the function that identifies users that have been granted a given permission. Given $p \in PERMS$, it is defined as $users(p) = \{u \in USERS \mid \langle u, p\rangle \in UP\}$.

Having introduced these entities, it is now possible to formally define the main objective of role engineering: given *UP*, *PERMS*, and *USERS*, we are interested in determining the best setting for *ROLES*, *PA*, and *UA* that covers all possible combinations of permissions possessed by users. In this context "best" means that the proposed roles should maximize the advantages offered by adopting RBAC, that is, to simplify access governance, to mitigate the risk of unauthorized access, and to ensure that roles reflect business requirements throughout the enterprise. This can be seen as a multi-objective optimization problem [4, 6]. As for the coverage, there is a need that for each $\langle u, p\rangle \in UP$ at least one role $r \in ROLES$ should exist such that $u \in ass\_users(r)$ and $p \in ass\_perms(r)$.

4 A Divide-and-Conquer Approach

In this section we describe how to condition existing role mining algorithms to craft roles with business meaning and to downsize the problem complexity. By leveraging the observations of Section 1, it is possible to exploit available business information, or top-down analysis results, in order to drive a bottom-up approach. In particular, a business attribute (e.g., organizational units, job titles, applications, tasks, etc.) naturally induces a partition of the user-permission assignment set *UP* to analyze, where each subset is made up of all the assignments that share the same attribute values. When several business attributes are at our disposal, the difficulty arises in the selection of the one that induces a partition for *UP* that simplifies the subsequent mining steps. To this end, for each business information we calculate an index referred to as ENTRUSTABILITY (***ent****ropy-based* ***r****ole* ***us****efulness predic****tability***), which measures the uncertainty in identifying homogeneous sets of users and permissions that can be managed through a single role. The decomposition with the highest ENTRUSTABILITY value is the one that most likely leads to roles with a clear business meaning.

In the following, we first introduce the *pseudo-role* concept (Section 4.1) as a means to identify sets of users and permissions that can be managed by the same role. In turn, we formally introduce the ENTRUSTABILITY index (Section 4.2) to measure how much a partition reduces the uncertainty in locating such sets of users and permissions in each subset of the partition.

4.1 Pseudo-roles

The following definition introduces an important concept of the proposed methodology:

Definition 1. *Given a user-permission assignment* $\langle u, p\rangle \in UP$*, the* pseudo-role generated by $\langle u, p\rangle$ *is a role made up of users* $users(p)$ *and permissions* $perms(u)$.

Pseudo-roles have been introduced for the first time in [5], with the alternative name of "pseudo-biclusters". Moreover, in [6] we discussed pseudo-roles from a graph theory perspective. In particular, in we proposed a mapping between binary matrices and undirected graphs where a pseudo-role represent all the neighbors of a given node. In [6] we also provided efficient algorithms for viable computation of pseudo-roles. In this paper, pseudo-roles will be employed to identify those user-permission assignments that

can be managed together with a given assignment through a single role. Notice that all users $users(p)$ should not necessarily be granted all permissions $perms(u)$—this is the reason for the "pseudo" prefix. Since a pseudo-role $\hat{r}$ is *not* an actual role, with abuse of notation we refer to its users as $ass\_users(\hat{r})$ and to its permissions as $ass\_perms(\hat{r})$. Several user-permission assignments can generate the same pseudo-role. In particular:

Definition 2. *The percentage of user-permission assignments of UP that generates a pseudo-roles $\hat{r}$ is referred to as its* frequency*, defined as:*

$$\varphi(\hat{r}) := \frac{1}{|UP|}\left|\{\langle u, p\rangle \in UP \mid ass\_users(\hat{r}) = users(p) \ \wedge \ ass\_perms(\hat{r}) = perms(u)\}\right| \ .$$

The introduction of the pseudo-roles concept is supported by the following theorem:

Theorem 1. *Given a user-permission assignment $\langle u, p\rangle \in UP$, let $\hat{r}$ be the pseudo-role generated by $\langle u, p\rangle$. Then*

$$UP_{\hat{r}} := (ass\_users(\hat{r}) \times ass\_perms(\hat{r})) \cap UP$$

is the set of all possible user-assignment relationships that can be covered by any role to which $\langle u, p\rangle$ belongs to. Hence, for each possible RBAC state $\langle ROLES, UA, PA\rangle$ that covers the assignments in UP the following holds:

$$\forall r \in ROLES \ : \ u \in ass\_users(r), \ p \in ass\_perms(r) \implies ass\_users(r) \times ass\_perms(r) \subseteq UP_{\hat{r}} \ .$$

Proof. First, we prove that any assignment that can be managed together with $\langle u, p\rangle$ must be within $UP_{\hat{r}}$. Let $\langle u', p'\rangle \in UP$ be an assignment outside the pseudo-role $\hat{r}$, namely $\langle u', p'\rangle \notin UP_{\hat{r}}$. If, by contradiction, $\langle u, p\rangle$ and $\langle u', p'\rangle$ can be managed through the same role r', then by definition all the users $ass\_users(r')$ must have permissions $ass\_perms(r')$ granted. Hence, both the assignments $\langle u', p\rangle$ and $\langle u, p'\rangle$ must exist in UP. But, according to Definition 1, $u' \in ass\_users(\hat{r}) = users(p)$ and $p' \in ass\_perms(\hat{r}) = perms(u)$, that is a contradiction.

Now we prove that any assignment within $UP_{\hat{r}}$ can be managed together with $\langle u, p\rangle$ via a single role. Given $\langle u'', p''\rangle \in UP_{\hat{r}}$, Definition 1 yields $u'' \in ass\_users(\hat{r}) = users(p)$ and $p'' \in ass\_perms(\hat{r}) = perms(u)$. Thus, both the assignments $\langle u'', p\rangle$ and $\langle u, p''\rangle$ exist in UP, completing the proof. □

According to the previous theorem, a pseudo-role groups all user-permission assignments that are manageable through any of the roles that also covers the pseudo-role generators. The pseudo-role frequency indicates the minimum number of assignments covered by the pseudo-role (i.e., the generators) that are manageable through the same role. Consequently, the higher the frequency of a pseudo-role is, the more pseudo-role assignments can be managed by one role. Similarly, the lower the frequency is, the more likely it is that the assignments covered by a pseudo-role cannot be managed by a single role. Therefore, the ideal situation is when pseudo-role frequencies are either close to 1 or close to 0: frequent pseudo-roles circumscribe a portion of assignments that are worth investigating since they likely contain a role for managing most of the assignments; conversely, unfrequent pseudo-roles identify assignment sets that are not worth analyzing.

4.2 Entrustability

Based on the previous observations, we are interested in finding the decomposition that produces pseudo-roles with frequencies either close to 1 or to 0. In the following we show that the *entropy* concept [8] is a natural way to capture these circumstances. Let $\mathcal{A}$ be the set of all values assumed by a given business information—for instance, $\mathcal{A}$ can represent the "job title" information, and one of the actual values $a \in \mathcal{A}$ can be "accountant". Let $\mathcal{P} := \{UP_{a_1}, \ldots, UP_{a_n}\}$ be a n-partition of UP induced by the business information $\mathcal{A}$ such that the number of subsets are $n = |\mathcal{A}|$, each subset is such that $UP_{a_i} \subseteq UP$, the subset indices are $\forall i \in 1, \ldots, n : a_i \in \mathcal{A}$, and the subset are such that $UP = \bigcup_{a \in \mathcal{A}} UP_a$. UP_a indicates all assignments that "satisfy" the attribute value a (e.g., if $\mathcal{A}$ represents the "job title" information, all the assignments where users are "accountant" are one subset). Notice that, according to the previous partition definition, subsets can overlap, namely $|UP_a \cap UP_{a'}| \geq 0$ when users or permissions can be associated to more than one attribute value. Let $\mathcal{R}_a$ be the set of all pseudo-roles that can be generated within the subset UP_a, and $\mathcal{R} := \bigcup_{a \in \mathcal{A}} \mathcal{R}_a \cup \mathcal{R}_*$ where $\mathcal{R}_*$ represents the pseudo-roles belonging to UP before decomposing it. Notice that the same pseudo-role might belong to both $\mathcal{R}_*$ and another set $\mathcal{R}_a$, namely $|\mathcal{R}_* \cap \mathcal{R}_a| \geq 0$, but not necessarily with the same frequencies.

Let $A \in \mathcal{A}$ be the random variable that corresponds to a value of the given business attribute, while the random variable $R \in \mathcal{R}$ denotes a pseudo-role generated by a generic user-permission assignment. Let $\Pr(\hat{r})$ be the empirical probability of a pseudo-role $\hat{r} \in \mathcal{R}$ being generated by an unspecified user-permission assignment. More specifically,

$$\Pr(\hat{r}) := \frac{1}{|UP|} \sum_{\omega \in UP} g(\omega, \hat{r})$$

where

$$g(\omega, \hat{r}) := \begin{cases} 1, & \omega \text{ generates } \hat{r} \text{ in } UP; \\ 0, & \text{otherwise.} \end{cases}$$

Similarly, the empirical probability of a pseudo-role being generated by an unspecified user-permission assignment that "satisfies" the business attribute a is

$$\Pr(\hat{r} \mid A = a) := \frac{1}{|UP_a|} \sum_{\omega \in UP_a} g_a(\omega, \hat{r})$$

where

$$g_a(\omega, \hat{r}) := \begin{cases} 1, & \omega \text{ generates } \hat{r} \text{ in } UP_a; \\ 0, & \text{otherwise.} \end{cases}$$

Notice that, for each attribute value a, when $\hat{r} \in \mathcal{R}_a$, then $\Pr(\hat{r})$ corresponds to the frequency definition. Conversely, if $\hat{r} \in \mathcal{R} \setminus \mathcal{R}_a$, then $\Pr(\hat{r}) = 0$.

As stated before, the natural measure for the information of the random variable R is its entropy $H(R)$. The binary entropy, defined as

$$H(R) := -\sum_{\hat{r} \in \mathcal{R}} \Pr(\hat{r}) \log_2 \Pr(\hat{r})$$

quantifies the missing information on whether the pseudo-role $\hat{r}$ is generated from some unspecified user-permission assignment when considering the set UP as a whole. By convention, $0 \times \log_2 0 = 0$. The conditional entropy is defined as

$$H(R \mid A) := -\sum_{a \in \mathcal{A}} \Pr(a) \sum_{x\hat{r} \in \mathcal{R}} \Pr(\hat{r} \mid A = a) \log_2 \Pr(\hat{r} \mid A = a) \ ,$$

where $\Pr(a) := |UP_a| \,/\, \sum_{a \in \mathcal{A}} |UP_a|$ measures the empirical probability of choosing an assignment that satisfies a. $H(R \mid A)$ quantifies the missing information on whether the pseudo-role $\hat{r}$ is generated from some unspecified user-permission assignment when A is known. The mutual information

$$I(R; A) := H(R) - H(R \mid A)$$

measures how much the knowledge of A changes the information on R. Hence, $I(R; A)$ measures how much the knowledge of the business information A helps us to predict the set of users and permissions that are manageable by the same role within each subset. Since $I(R; A)$ is an absolute measure of the entropy variation, we introduce the following measure for the fraction of missing information removed by the knowledge of A with respect to the entropy $H(R)$ before partition:

$$\textsc{entrustability}(A) := \frac{I(R; A)}{H(R)} = 1 - \frac{H(R \mid A)}{H(R)} \ .$$

By selecting the decomposition with the highest ENTRUSTABILITY value, we choose the decomposition that simplifies the subsequent role mining analysis most. Notice that the previous equations consider one business attribute at a time. Given ℓ business information $\mathcal{A}_1, \ldots, \mathcal{A}_\ell$, it is simple to extend the definition of the ENTRUSTABILITY index by partitioning UP in subsets of assignments that contextually satisfies all business information which has been provided.

5 Results and Discussion

To demonstrate the usefulness of the proposed approach, we show an application to a real case. Our case study has been carried out on a large private organization. Due to space limitation, we only report on a representative organization branch that contained 50 users with 31 granted permissions, resulting in a total of 512 user-permission assignments. We adopted several user and permission attributes at our disposal. In order to protect organization privacy, some names reported in this paper for business attributes are different from the original ones.

According to the proposed approach, we computed the ENTRUSTABILITY index for each available business information. To further demonstrate the reliability of the methodology, we introduced a *control* test. That is, we try to categorize users according to the first character of their surname. Since this categorization does not reflect any access control logic, our methodology reveals that—as expected—partitioning by surname does not help the mining phase. Table 1 reports on the outcome of the analysis—it also specifies whether the attributes were used to partition user-permission assignments by users or

Table 1. ENTRUSTABILITY values of the analyzed business information

Attribute	*User*	*Perm*	ENTRUSTABILITY
Job Title	✓		1.00
Unit	✓		0.93
Cost Center	✓		0.85
Organizational Unit	✓		0.82
Building	✓		0.58
Application		✓	0.49
Division	✓		0.46
Surname	✓		0.02

by permissions. According to the reported values, the "Job Title" information induces the most suitable partition for the attribution of business meaning to roles. As a matter of fact, when ENTRUSTABILITY equals 1, each subset can be managed by just one role. Unsurprisingly, the categorization by surname leads to an ENTRUSTABILITY index that is very close to 0, indicating that the role engineering task does not substantially change its complexity after decomposition.

To better understand the meaning of the ENTRUSTABILITY values obtained from our analysis, Figure 1 depicts user-permission relationships involved with subsets for each partition. In particular, we report on the attribute values that identify each subset, the entropy value $H(R)$ computed for each subset, and a matrix representation of user-permission assignments, where each black cell indicates a user (row) that has a certain permission (column) granted. Figure 1(a) visually demonstrates why the Job Title information leads to a value for ENTRUSTABILITY that equals 1. Indeed, in this case all users sharing the same job title always share the same permission set. Therefore, by creating one role for each subset, we provide roles that can straightforwardly be associated with users whenever they join the organization (and get their job title for the first time) or change their business functions (and thus likely change their job title). Another piece of information that induces a good partition is Unit. As is noted from Figure 1(b), almost all users within each group share the same permission sets. For example, within the unit "Personal Communication Unit" there is one user (the first one) that has an additional permission granted compared to other users of the same unit. For this reason, the identification of roles needed to manage these users requires a little more investigation—hence, leading to a non-zero entropy value, that is, $H(R) = 0.98$. This example also raises another important point: even though the ENTRUSTABILITY value for Job Title is higher than for Unit, the Unit information induces fewer and larger subsets, hence allowing to cover all user-permission relationships with fewer roles. In general, the smaller the subsets, the more likely it is that the ENTRUSTABILITY index is high. However, small subsets reduce the benefits introduced by RBAC in terms of administration effort, due to the limited number of user-permission relationships that can be managed via a single role. Hence, a trade-off should be reached between ENTRUSTABILITY value and subset dimensions. An alternative approach could be to further partition those subsets that have high entropy values by introducing other pieces of business information. In

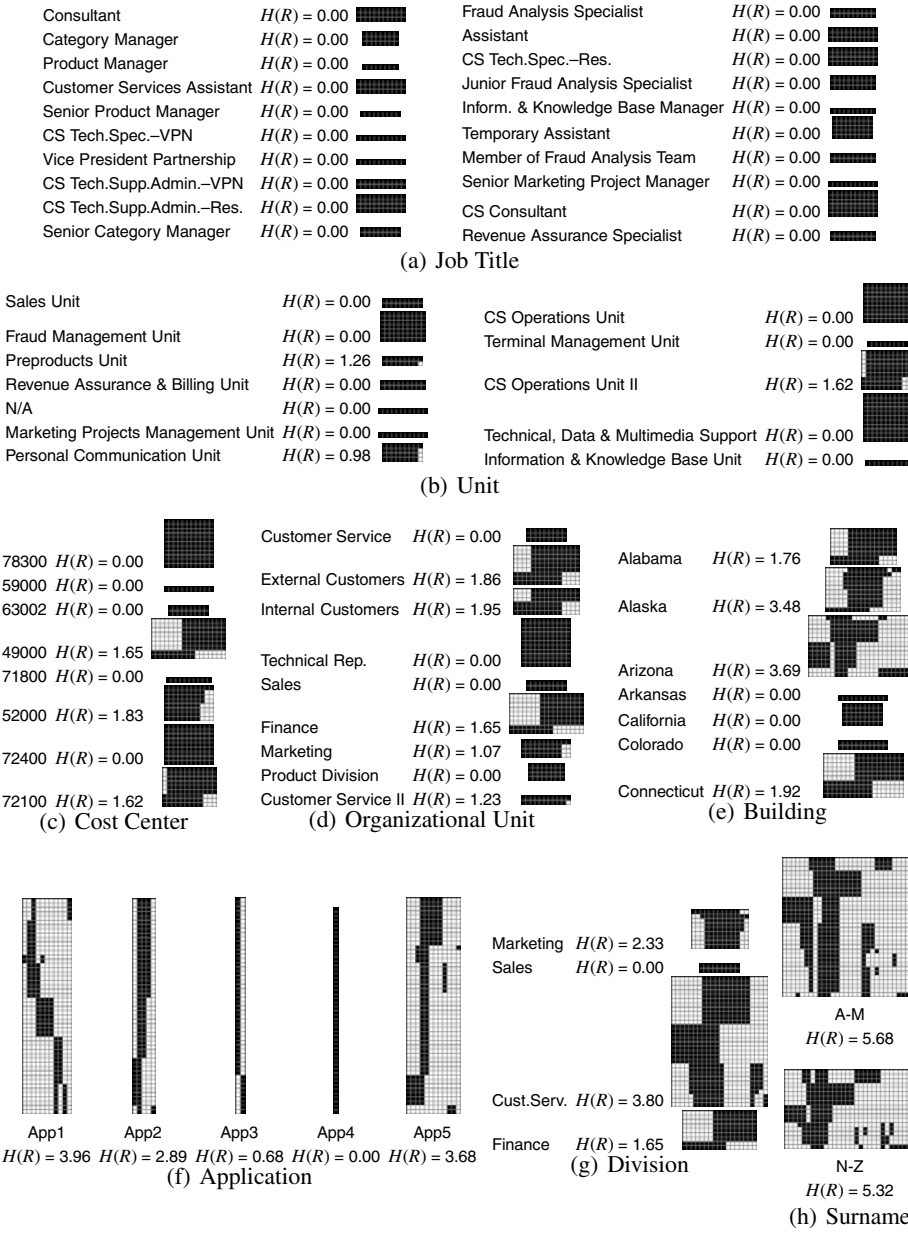

Fig. 1. Graphical representation of user-permission relationships involved with subsets of each partition and corresponding entropy values

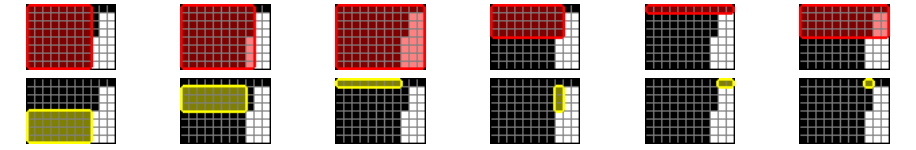

Fig. 2. Pseudo-roles (top figures, highlighted in red) and corresponding user-permission assignment generators (bottom figures, highlighted in yellow)

the previous case, the subset identified by the unit named "CS Operations Unit II" (see Figure 1(b)) involves users with two job titles: if we recursively apply our methodology and divide the unit "CS Operations Unit II" according to the Job Unit information, we will obtain an ENTRUSTABILITY value that equals 1. Hence, obtaining larger roles when compared to the partition by Job Title only.

Figure 1(g) also demonstrates that not every bit of business information improves the role finding task. Although analyzing the data as a whole is obviously more difficult than analyzing smaller subsets, in this case there are still uncertainties regarding the identification of roles. For instance, it is not trivial to assign a meaning to possible roles—without any further information—within the division "Cust.Serv.", namely the division with the highest entropy value. Finally, Figure 1(h) clearly shows that surname information is completely useless. In fact, if we compute the entropy of the entire user-permission assignment, we obtain the value $H(R) = 5.69$. In this case, the entropy values for users "A-M" and "N-Z" are almost the same as before the decomposition.

To conclude, Figure 2 depicts all the pseudo-roles that can be identified in a simple case represented by the cost center named "52000" (from Figure 1(c)), which numbers 8 users, 11 permissions, and 70 user-permission assignments. Each figure from Figure 2(a) to Figure 2(f) shows a different pseudo-role. At the top of each figure, a binary matrix shows all the user-permission assignments covered by the pseudo-role (dark red cells are existing assignments covered by the pseudo-role, light red are non-existing assignments). At the bottom, another matrix shows the assignments that generate the pseudo-role (highlighted in yellow). Notice that when the pseudo-role frequency is high (e.g., Figure 2(a) and Figure 2(b)), it likely contains a role for managing most of the assignments. Conversely, unfrequent pseudo-roles (e.g., Figure 2(e) and Figure 2(f)) identify assignment sets that are not worth investigating due to the reduced number of assignments that can be managed by a single role.

6 Concluding Remarks

This paper describes a methodology that helps role engineers to leverage business information during the role mining process. In particular, we demonstrate that by dividing data into smaller, more homogeneous subsets, it practically leads to the discovery of more meaningful roles from a business perspective, decreasing the risk factor of making errors in managing them. To drive this process, the ENTRUSTABILITY index has been introduced to measure the expected uncertainty in locating homogeneous set of users and permissions that can be managed by a single role. Leveraging this index allows to

identify the decomposition that increases business meaning in elicited roles in subsequent role mining steps, thus simplifying the analysis. The quality of the index is also guaranteed by analysis.

Several examples, developed on real data, illustrate how to apply the tools that implement the proposed methodology, as well as its practical implications. Those results support both the quality and the practicality of the proposal.

References

1. ANSI/INCITS 359-2004, Information Technology – Role Based Access Control (2004)
2. Colantonio, A., Di Pietro, R., Ocello, A.: A cost-driven approach to role engineering. In: Proc. ACM SAC, pp. 2129–2136 (2008)
3. Colantonio, A., Di Pietro, R., Ocello, A.: Leveraging lattices to improve role mining. In: Proc. IFIP SEC, pp. 333–347 (2008)
4. Colantonio, A., Di Pietro, R., Ocello, A., Verde, N.V.: A formal framework to elicit roles with business meaning in RBAC systems. In: Proc. ACM SACMAT, pp. 85–94 (2009)
5. Colantonio, A., Di Pietro, R., Ocello, A., Verde, N.V.: ABBA: Adaptive bicluster-based approach to impute missing values in binary matrices. In: Proc. ACM SAC, pp. 1027–1034 (2010)
6. Colantonio, A., Di Pietro, R., Ocello, A., Verde, N.V.: Taming role mining complexity in RBAC. Computers & Security. In: Challenges for Security, Privacy & Trust (2010)
7. Colantonio, A., Di Pietro, R., Ocello, A., Verde, N.V.: Taming role mining complexity in RBAC. Computers & Security. Challenges for Security, Privacy & Trust (2010)
8. Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley-Interscience, Hoboken (2006)
9. Coyne, E.J.: Role-engineering. In: Proc. ACM RBAC, pp. 15–16 (1995)
10. Ene, A., Horne, W., Milosavljevic, N., Rao, P., Schreiber, R., Tarjan, R.E.: Fast exact and heuristic methods for role minimization problems. In: Proc. ACM SACMAT, pp. 1–10 (2008)
11. Frank, M., Streich, A.P., Basin, D., Buhmann, J.M.: A probabilistic approach to hybrid role mining. In: Proc. ACM CCS, pp. 101–111 (2009)
12. Heikinheimo, H., Vreeken, J., Siebes, A., Mannila, H.: Low-entropy set selection. In: Proc. SIAM SDM, pp. 569–580 (2009)
13. Kuhlmann, M., Shohat, D., Schimpf, G.: Role mining – revealing business roles for security administration using data mining technology. In: Proc. ACM SACMAT, pp. 179–186 (2003)
14. Molloy, I., Chen, H., Li, T., Wang, Q., Li, N., Bertino, E., Calo, S., Lobo, J.: Mining roles with semantic meanings. In: Proc. ACM SACMAT, pp. 21–30 (2008)
15. Molloy, I., Li, N., Li, T., Mao, Z., Wang, Q., Lobo, J.: Evaluating role mining algorithms. In: Proc. ACM SACMAT, pp. 95–104 (2009)
16. Neumann, G., Strembeck, M.: A scenario-driven role engineering process for functional RBAC roles. In: Proc. ACM SACMAT, pp. 33–42 (2002)
17. Röckle, H., Schimpf, G., Weidinger, R.: Process-oriented approach for role-finding to implement role-based security administration in a large industrial organization. In: Proc. ACM RBAC, vol. 3, pp. 103–110 (2000)
18. Schlegelmilch, J., Steffens, U.: Role mining with ORCA. In: Proc. ACM SACMAT, pp. 168–176 (2005)
19. Tatti, N.: Maximum entropy based significance of itemsets. Knowledge and Information Systems 17(1), 57–77 (2008)
20. Vaidya, J., Atluri, V., Warner, J.: RoleMiner: mining roles using subset enumeration. In: Proc. ACM CCS, pp. 144–153 (2006)
21. Zhang, D., Ramamohanarao, K., Ebringer, T.: Role engineering using graph optimisation. In: Proc. ACM SACMAT, pp. 139–144 (2007)