SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks 2012; **5**:847–862
Published online 25 February 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.314

SPECIAL ISSUE PAPER

Minimizing insider misuse through secure Identity Management

Ludwig Fuchs*[1,2]* and Günther Pernul[2]

[1] Nexis GmbH, Prüfeninger Str. 94a, D-93049 Regensburg, Germany
[2] Department of Information Systems, University of Regensburg, Universitätsstraße 31, D-93053 Regensburg, Germany

ABSTRACT

To avoid insider computer misuse, identity, and authorization data referring to the legitimate users of systems must be properly organized, constantly and systematically analyzed, and evaluated. In order to support this, structured and secure Identity Management is required. A comprehensive methodology supporting Identity Management within organizations has been developed, including gathering of identity data spread among different applications, systematic cleansing of user account data in order to detect semantic as well as syntactic errors, grouping of privileges and access rights, and semiautomatic engineering of user roles. The focus of this paper is on the cleansing of identity and account data leading to feedback where insider misuse due to existing privileges which go beyond the scope of the users' current need-to-know may occur. The paper in detail presents used data cleansing mechanisms and underlines their applicability in two real-world case studies. Copyright © 2011 John Wiley & Sons, Ltd.

KEYWORDS

insider misuse; Identity Management; user account data; quality of identity data; role-based access control

***Correspondence**

Ludwig Fuchs, Nexis GmbH, Prüfeninger Str. 94a, D-93049 Regensburg, Germany.
E-mail: ludwig.fuchs@nexis-secure.de

1. INTRODUCTION

Insider threats represent the historical cause of the majority of incidents with the authorized, non-technical employee being the typical potential threat to Information Security [1]. Several publications like [2] investigate the motivation and impact of those attacks in detail. Widely cited surveys like the BERR[†] Information Security Breaches Survey 2008 [3], or the CSI[‡] Computer Crime and Security Survey 2008 [4], furthermore, underline that insiders like employees are one of the biggest threats to data security.

Insiders misusing a system commonly act by using their own user accounts and perform within the range of their assigned privileges and access rights but abuse their current job functions, which became possible because of inadequate and manual user management [5]. In such an environment misuse might remain undetected and invisible because current detection methods mainly rely on rule-breaking behavior which would not be the case here. In 2006, Cappelli *et al*. [6] investigated this problem and point out possible ways of solving it.

During their lifetime in the organization employees are not statically assigned to a certain job but migrate between different job functions. Each change often goes along with new and additional access privileges to the IT resources. An additional risk arises from the fact that identity and authorization data is usually spread among several applications (identity silos) and in the case an employee leaves the organization is not completely revoked. The situation described above is sometimes referred to as the so called 'identity chaos.' It describes a situation in which users have multiple identities, passwords, and accounts spread across a variety of security domains (networks, applications, computers, and/or computing devices). Given these factors, it is not surprising that the Aberdeen Group[§] states that only 17 per cent of companies claim they do not have orphaned accounts (accounts with access that should have

[†] Department for Business, Enterprise & Regulatory Reform, formerly Department of Trade and Industry.

[‡] Computer Security Institute.

[§] Aberdeen Group, *Identity and Access Management Critical to Operations and Security*, March 2007; Copyright © 2007, Aberdeen Group.

been revoked). Some organizations take more than 30 days to de-provision accounts and others have no defined processes for de-provisioning or means to discover orphaned accounts at all.

Related to preventing insider misuse is the aspect of evaluating the compliance of IT with laws and regulations. Under this umbrella, organizations are increasingly forced to control, manage, and audit their Identity Management processes. Among the most known drivers are the U.S. Sarbanes-Oxley Act (SOX) of 2002 [7,8], Basel II [9], the German BSI Grundschutz [10], the Directive 95/46/EC of the European Parliament [11], ISO IT Security standards (such as ISO 27002), and own regulations large organizations use for their internal audits.

This paper is concerned with the risk of system misuse by over-authorized insiders to whom the capability of accessing one or many components of the IT system has been legitimately given. It is an extension of the work published in Ref. [12]. In order to fight the identity chaos and the risk of insider misuse we propose a methodology for structured Identity Management consisting of (a) gathering of identity data spread among different security domains, (b) systematic cleansing of identity and account data in order to increase their quality and detect orphaned accounts, (c) grouping of privileges and access rights based on job functions and organizational structure, and (d) semiautomatic engineering of user roles. We give a general overview of the methodology but have a focus on the data cleansing and detection of the orphaned accounts phases.

The paper is structured as follows: Section 2 contains the general overview, Sections 3 and 4 have a focus on syntactic and semantic data cleansing and Sections 5 and 6 contain the evaluation of our methodology by performing case studies with account data of companies. Section 7 contains the conclusion and future work.

2. GENERAL OVERVIEW OF THE CONTROLE METHODOLOGY

ContROLE is a methodology and corresponding tool set supporting a structured Identity Management process. The process consists of six different phases (Figure 1). Early stages are concerned with gathering identity and account information as well as information about the organizational structure of the enterprise. They are followed by data cleansing, aiming at detecting inconsistencies, syntactic and semantic errors, and orphaned accounts. Final phases are concerned with mining and grouping access characteristics, relating them to typical job functions, and with suggesting user roles. The methodology may be applied as a whole (leading to a role catalog) or only partly, for example, data cleansing process steps only. The six phases and the respective quality measurement (QM) and execution decision (ED) steps during phase transition allow users of the methodology to move back to previous phases or within phases in an incremental and iterative fashion.

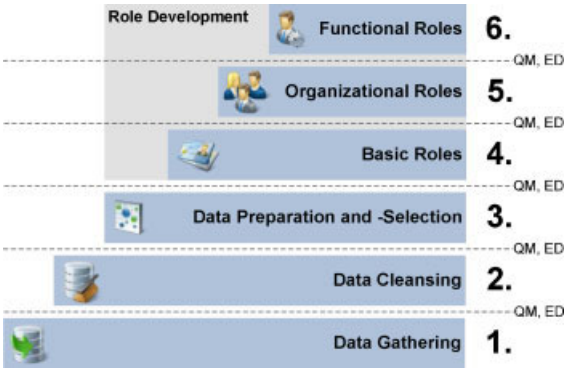

Figure 1. Structured Identity Management process according to the contROLE methodology.

Applying the methodology has high potential for reducing the risk of insider misuse. The earlier phases help to get a better understanding where identity and account data is spread in the organization and in what aspect security policies in different domains differ. Making security officers and CIOs aware of this is an important part of mitigating the risk of insider misuse. Analyzing and cleansing of existing identity and account data is central to reducing the risk. It significantly contributes to increasing the quality of identity and account information by pointing to syntactic and semantic errors in directories, orphaned accounts, or existing privileges which might not be necessary to perform the job. Also structuring the user population according to typical user roles has significant benefit to hinder insider misuse.

The following is a short description of each of the contROLE process steps. Data gathering is concerned with the compilation of a consistent information repository representing the basis for further data cleansing, data preparation, and role development. Identity and account information can be spread over several security domains and hidden in LDAP directories, meta-directories, authorization lists, and tables or embedded within different applications. Molloy *et al*. [13] recently provided a compilation of different dimensions of available input data based on existing electronically stored identity information within the organizations' IT systems. A minimal configuration of input data consists of user permission information, in other words, the set of users, the set of permissions, and the binary user-permission-assignments (UPA). Commonly, extensional user attribute data is available. That is, a user's job title, hierarchy elements, or locations are given. Additionally, organizational structure of an enterprise is related to the access controls as it is commonly stored within databases. Often, also permission parameter information is provided, for example, a number of permissions that may concern the same target system. In some systems, additionally permission update information that records how the access control state has evolved in the past is available. Permission usage information, at last, could be available in the form of logs showing which permissions are used at what time.

Security Comm. Networks 2012; **5**:847–862 © 2011 John Wiley & Sons, Ltd.
DOI: 10.1002/sec

After input data has been cleansed (Sections 3 and 4), the process moves on to Data Preparation and Selection. Its goal is to automatically generate additional knowledge about the cleansed input data by using classification and clustering technologies. In order to arrive at a suitable role catalogue, it is mandatory to allow choosing the users, rights, and/or organizational units to be included in the role development process. Phases 4–6 are devoted to the actual Role Development. The outcome is a set of roles of a certain type: Basic roles bundle common access privileges within organizational elements, organizational roles represent job positions while functional roles represent common task bundles of employees. The roles are stepwise derived, coming from more general ones, such as basic roles to very specific ones. The methodology supports iterative role development through integration of role mining and role engineering in various loops. While mining is concerned with analyzing patterns in user account information, role engineering follows a top–down approach and considers input data concerning the organizational structure of the enterprise.

More information about the contROLE methodology can be collected at www.nexis-secure.com. Different aspects are already published, i.e., the general process of in-house Identity Management [14], tool support for structured Identity Management [15], the process of semi-automated generation of roles [16] the impact of the data cleansing phase on data quality, and insider threat [12]. This work extents the work in Ref. [12] focusing on the different aspects and new mechanisms for semantic cleansing of identity and account data in order to reduce insider misuse (Sections 3 and 4).

3. SYNTACTIC CLEANSING OF IDENTITY AND ACCOUNT DATA

There is a common agreement that identity data as stored in the access controls (e.g., access control lists) tends to be incomplete, noisy, and inconsistent [17]. Analyzing and cleansing of this data is, therefore, central to reducing the risk of insider misuse. After the initial data gathering, the contROLE methodology, therefore, continues with syntactic and semantic data cleansing. From the previous phase it assumes the existence of a central information repository built from existing identity and account data as well as data concerning the organizational structure of the enterprise. After errors have been detected and cleansed, the updated data is written back to the originating sources.

Syntactic checks aim at revealing errors regarding the input data entities while semantic checks try to identify inconsistencies in the relationships among those entities. Additionally, the methodology provides a Policy Checking Engine (PCE) which allows organizations to validate if existing security policies are correctly applied. All activities mentioned are supported by a computerized tool. This section describes syntactic data cleansing and the PCE before Section 4 focuses on the semantic data analysis.

3.1. Syntactic analysis of identity and account data

Syntactic analysis follows the process described in Figure 2 (gray shading represents optional tasks). It aims to detect invalid data like misspelled attribute values, duplicate or similar datasets, incomplete datasets with missing- or null-values, and violations of referential integrity constraints.

In the case valid value lists have been provided a consistency check can optionally be carried out to ensure the correctness of the datasets corresponding to the employees, permissions, and hierarchical structure. Actual values which are not included in the valid value lists are highlighted. The consistency check includes a distance metric similar to the Levenshtein distance [18] in order to propose a valid value for an erroneous entry. As an example consider a misspelled name of an employee. Instead of deleting the respective dataset the correct employee name should be proposed. In the case no correct value can be proposed, the consistency check by default assigns a null-value, marking datasets for further investigation. The same holds for predefined null-values included in valid value lists.

A duplicate check identifies identical datasets while the similarity check reveals misspellings. The latter is commonly applied in case the consistency check has been skipped. Again, distance metrics are computed and used for the detection of errors and the proposal of correct values for misspelled datasets. Finally, a check for missing values reveals incomplete datasets. Depending on the general policy, these datasets could be deleted. More likely, however, the missing values are replaced with a valid null-value. This allows for the later treatment during the semantic cleansing.

3.2. Policy Checking Engine

Besides the semi-automated syntactic and semantic data investigation the contROLE methodology allows for the

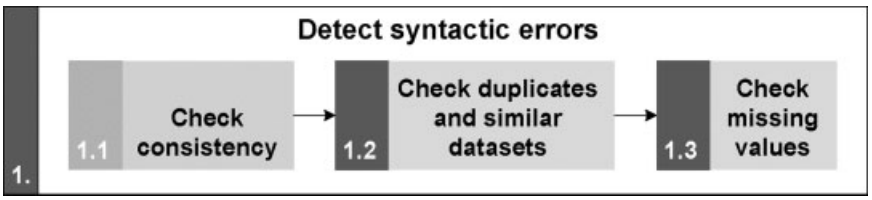

Figure 2. ContROLE process steps for syntactic analysis.

consideration of available security – or IdM policies during a policy checking activity. Since the upcoming of governmental regulatory requirements an increasing demand for accountability urges organizations to improve compliance practices. IdM is regarded as an essential part for achieving compliance in the area of user management [19].

However, up to now there has been hardly an integration of IdM practices with the aforementioned controlling practices required by the governance frameworks and certification processes. One goal of the contROLE methodology, therefore, is the automatic integration as well as re-engineering (in other words, the critical investigation and optimization) of security policies concerning Identity Management.

During a first step users of the methodology can define security policies within a dedicated tool or import existing policies (Figure 3). After the import or definition activity contROLE checks the enforcement of the policies on the basis of the existing and precleansed identity data. Policy checking can be executed iteratively and periodically for all organizational units or only for preselected parts of the company. In order to support a revision process the contROLE toolset generates a reporting overview of all detected policy violations for technical users of the methodology. It suggests sending the identified policy violations that cannot be resolved by technical experts for reviewing by business professionals. Using their feedback of those technical and managerial experts, the policy violations on the one hand can be resolved and, on the other hand, the policies themselves can be investigated and re-engineered if necessary.

The methodology allows for the checking of different security policy types. One major security policy type is Separation of Duty (SoD) constraints limiting the relationships of users to permissions [20,21]. SoD constraints can be defined for permissions and predefined roles. Hence,

Table I. Separation of Duty constraints.

SoD rule	Constraint
SoD001	Mutual_Exclusion {FR:Cost Accounting, FR:Business Management}
SoD002	Min_Assigned_Employees {FR:Project Controlling} = 2

contROLE allows users of the methodology to specify bundles of permissions or business roles which must not be assigned to a certain number of employees or one single employee at a time. Note that up to now this checking is restricted to static SoD constraints and not dealing with dynamic SoD.

As an example, a departmental manager might define two constraints (Table I). Firstly, the two roles Cost Accounting and Business Management must not be carried out by one single person (rule SoD001). Besides the SoD constraints, contROLE is able to include cardinality constraints on identity related data during the policy checking process. The departmental manager in the example above might, secondly, require that Project Controlling is carried out by at least two employees (rule SoD002 in Table I), in other words, no single employee alone must be responsible for executing the related task bundles.

Our main research focus concerning the PCE up to now was on the manual definition of security policies within the contROLE toolset. Providing standardized data exchange interfaces is currently under development. We are implementing XML-, OLE/COM-, or text-based interfaces for the inclusion of existing security policies (Figure 4).

We are furthermore extending the PCE to additionally be able to include digitally available business processes or UML diagrams annotated with security requirements or actors carrying out task bundles. Usable approaches

Figure 3. ContROLE process steps for policy checking.

Figure 4. Automatic inclusion of operational structures and policies.

Security Comm. Networks 2012; **5**:847–862 © 2011 John Wiley & Sons, Ltd.
DOI: 10.1002/sec

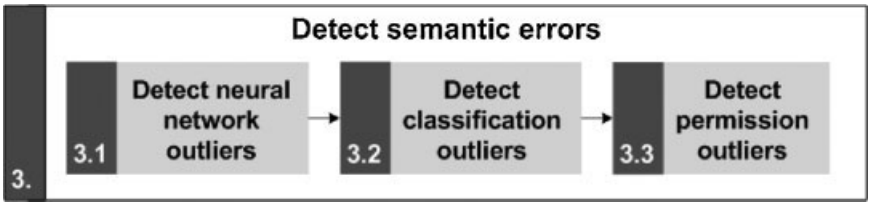

Figure 5. ContROLE process steps for semantic analysis.

that focus on the automatic extraction of roles or security-relevant information from organizational structures have been proposed in Ref. [22] and recently in the context of Identity Management also in Ref. [23].

4. SEMANTIC CLEANSING OF IDENTITY AND ACCOUNT DATA

In addition to the syntactic checks, the contROLE methodology provides functionality for semantic analysis of the input data, which, in specific, is important for the detection of potential insider threat risks. Technologies used by the contROLE toolset to identify semantic errors are statistical analysis, clustering, classification, mining, and artificial neural networks. While syntactic checks might be fully automatable, semantic checks cannot be processed without human intervention. Therefore, results need to be visualized appropriately and be sent to a domain expert for approval.

Focusing on the relationship between permissions, employees, and organizational hierarchy elements (OHE), semantic error detection is used to detect different types of outliers. An outlier in general is understood as an object exhibiting alternative behavior in a dataset, i.e., a data point that does not conform to the general patterns characterizing the dataset [17]. The contROLE methodology is able to detect the following outliers: (a) employees with authorizations not matching their job functions (e.g., over-authorized employees), employees with atypically assigned permissions or attributes, employees with valid but incorrectly assigned attribute values and (b) permissions no longer in use but still assigned to employees and permissions used by nearly all employees within an organizational unit. We call type (a) 'employee outliers' and type (b) 'permission outliers'.

Semantic analysis follows the process model described in Figure 5. While the permission outlier checks reveal potentially erroneous user-permission assignments for deletion, neural networks, and employee classification techniques highlight attribute values or permissions of employees which might be subject to re-assignment.

4.1. Detect neural networks outliers

The first type of semantic checks detects outliers in the form of identity and account data of employees with atypically assigned permissions or attributes using self-organizing maps (SOMs) as proposed by Kohonen [24]. Identity and account information from a specific element of the organizational hierarchy or even the whole input dataset can be selected for investigation. Before the detection of semantic anomalies can be conducted, the underlying SOMs have to be parameterized and trained. During training, the SOM groups employees according to the similarity of their assigned permissions. Similar users are located close to each other whereby employees with different access rights are located on different parts of the map.

The example shown in Figure 6 illustrates the detection of outliers by using SOMs. It depicts part of a SOM used

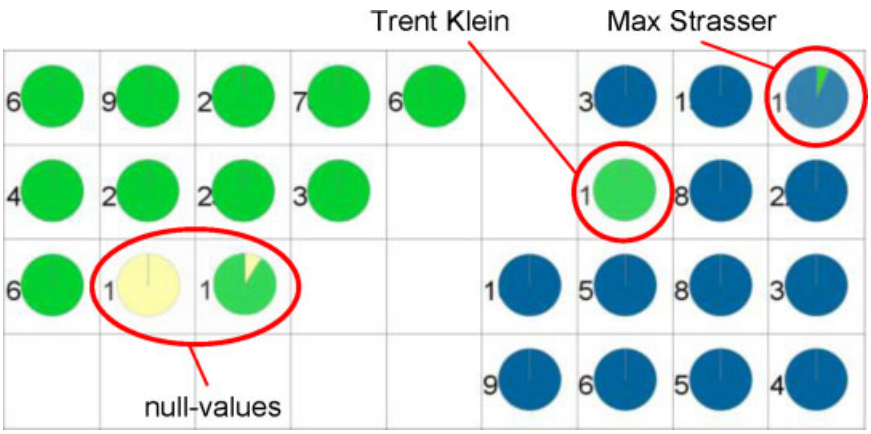

Figure 6. Employee outlier detection using SOMs.

to classify identity and account information of employees based on the attribute 'assignment to OHE', i.e., every employee is classified according to his aggregated hierarchy level (HL) assignment. Consider the employees Trent Klein and Max Strasser. Assume both employees have been re-assigned to the Support department (blue colored). Max Strasser's old access privileges from the Infrastructure department (green colored) have been correctly de-provisioned. However, his departmental attribute has not been changed. He therefore is visualized as outlier within a group of Support department employees. On the contrary, Trent Klein's old access privileges have not been revoked and his OHE assignment has not been updated yet. Trent thus is located in between the green and blue employee groups. Additionally, every null-value is considered to be an outlier (see yellow colored pie charts in Figure 6).

After detection, a decision about the treatment of the candidate errors needs to be made. For proposing a correct attribute value, the methodology analyzes all identity and account data located on a suspicious node and its direct neighborhood (NL-1) and selects the element with the highest similarity to the identified outlier. The non-aggregated class information of this user is proposed as correct value. In the example above Max Strasser has been identified as a member of the Support department (HL-1). If this OHE has five sub-departments, Max Strasser could potentially be assigned to either of them. Thus, in the example the employees assigned to the same node and the employees located on the three surrounding nodes are analyzed for their similarity to Max Strasser. In case the winner unit is assigned to a Support Billing department (HL-2), this value is proposed as correct value for Max Strasser.

Even though the contROLE methodology supports a manual review of the trained SOMs, it incorporates an algorithm (SOMParsing) which we developed to automatically investigate neighborhood levels of each node (NL, see Figure 7) for their predominant class information by analyzing the set of employees allocated to the NL. Imagine a situation where 10 employees constitute the NL-1 of a given node. If nine of those employees are assigned to class A, while one of them is assigned to class B, A is considered the predominant class of NL-1 with 90 per cent weighting.

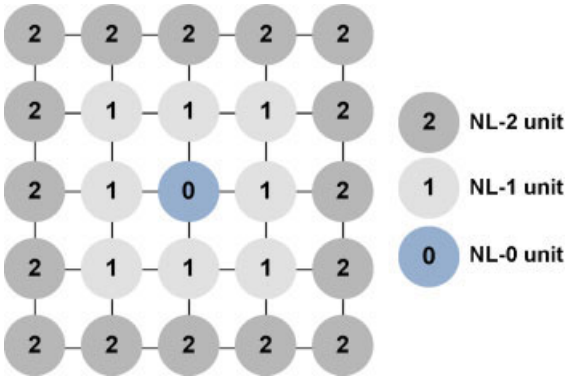

Figure 7. Neighborhood levels on a SOM.

Table II describes four checks providing a heuristic to extract suspicious node elements from a given SOM. Even though the algorithm does not claim to extract all potential outliers, contROLE suggests an iterative application in order to detect a high percentage of the outliers on the map. Example visualizations for each check are given in Figure 8.

The different checks are used during the detection of erroneous datasets. Check 1 investigates ambiguous nodes with more than one class information value assigned on which the predominant class A represents more than x per cent of all node members. Members not assigned to this class (yellow coloring in the left SOM in Figure 8) are marked as outliers if the node is surrounded by nodes exclusively

Table II. SOMParsing checking criteria.

Check	NL-0	NL-1	NL-2
1	A (>x per cent; <100 per cent)	A	Not considered
2	A (>x per cent; <100 per cent)	A (>y per cent)	Not considered
3	B	A (>x per cent)	A (>y per cent)

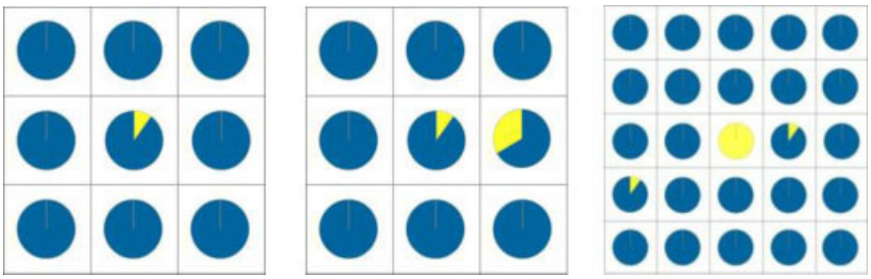

Figure 8. SOMParsing Checks 1-3 (from left to right).

Table III. Contingency table of employees and permissions.

	Permissions						
GlobalID	*P*1	*P*2	*P*3	*P*4	*P*5	*P*6	*P*7
CS1910	1	1	0	1	1	1	0
LL1012	1	1	1	1	1	1	0
JA1210	1	1	1	0	0	0	1
MR0120	1	1	1	0	0	0	1
LS1050	1	1	1	1	0	1	1
RS1201	1	1	0	1	0	1	1
DH2323	1	1	0	1	0	1	1

representing the class information A (blue coloring) on NL-1.

Check 2 is essentially a less restrictive version of Check 1. It weakens the detection policy and thus might likewise lead to an increased false-positive rate. In contrast to Check 1, Check 2 does not require all NL-1 nodes being assigned to the same predominant class A. Check 3 considers homogenous nodes with a predominant class B assigned on NL-0 (yellow coloring in the right SOM in Figure 8) that are surrounded by nodes with the predominant class A on NL-1 and NL-2. In this case, again, thresholds for the predominant classes need to be defined.

Note that the respective threshold for the application of the checks has to be manually set. In order to support this process, contROLE suggests the visual analysis of trained maps prior to the automated investigation. The subjective map quality can hint at the suitability of a more restrictive or loose parsing parameter.

4.2. Detect classification outliers

During a second process step of the semantic data analysis, employees are automatically grouped according to their similarity on the basis of a heuristic classification mechanism. The goal is to identify groups of similar employees that are not fully homogenous. These groups can be used for detecting employees with superfluous permissions assigned. The classification process is divided into three steps. Firstly, the similarity matrices of the investigated employees are calculated on the basis of the given UPAs. Secondly, the heuristic classification is executed for defining employee classes and all employees are assigned to exactly one of the predefined classes according to their maximum similarity. Thirdly, the result quality, i.e., the heterogeneity of the defined classes, is calculated. The process is presented in more detail in the following subsections.

4.2.1. Derive similarity matrices.

After the scope of the employee classification (e.g., the investigated department) has been defined, the similarity matrices are derived based on a similarity coefficient (the Jaccard coefficient [25] is used in the following). A simplified example is given in Table III.

In this example seven employees are either assigned (1) or not assigned (0) to seven different permissions (P1–P7). For the calculation common permissions (P1, P2) are excluded as they do not affect the classification process. Assume that permission P3 is an outdated permission no longer in use by Linda Loner (LL1012).

She and Chris Summer (CS1910), for example, share three permissions (P4, P5, and P6) while P7 is assigned to none of them and P3 is only assigned to Linda Loner. The similarity value is calculated pair-wise according to the Jaccard coefficient for all employees resulting in the symmetric similarity matrix depicted in Table IV.

4.2.2. Heuristic classification.

After the calculation of the similarity matrices, the actual classification is carried out by using a heuristic classification algorithm. The procedure can be divided into two steps: At first class centers are defined on the basis of a given upper bound and secondly the employees are assigned to these classes in a disjunctive and exhaustive manner.

In more detail, a starting point for the classification an employee with the most similarity values above a predefined threshold is selected as reference user of the first employee class. Successively the representative of a second class C_2 is chosen on the basis of the minimal similarity to the representative of the previously defined class. For the selection of the representative of the remaining classes the lowest maximal similarity to the existing classes is facilitated. As a stop criterion the previously used threshold is considered, arguing that in case every remaining object is similar to an already existing class, it is not feasible to define further classes.

Table IV. Similarity matrix of employees.

GlobalID	CS1910	LL1012	JA1210	MR0120	LS1050	RS1201	DH2323
CS1910	1	0.75	0	0	0.4	0.5	0.50
LL1012		1	0.2	0.2	0.6	0.4	0.4
JA1210			1	1	0.5	0.25	0.25
MR0120				1	0.5	0.25	0.25
LS1050					1	0.75	0.75
RS1201						1	1
DH2323							1

Table V. Lowest maximal similarity selection.

GlobalID	RS1201	JA1210	Max s(*i*,*j*)
CS1910	0.5	0	0.5
LL1012	0.4	0.2	0.4
MR0120	0.25	1	1
LS1050	0.75	0.5	0.75
DH2323	1	0.25	1

Continuing the example introduced above, the employees RS1201, LS1050, as well as DH2323 have three similarity values equal or above a predefined threshold of 0.7 (Table IV). The algorithm suggests choosing one of those employees as representative of the first employee class C_1 (e.g., RS1201). The employees JA1210 and MR0120 have the lowest similarity value (0.25) in respect to RS1201. Again, one of them (e.g., JA1210) is randomly selected as representative of the second employee class.

LL1012 has the lowest maximal similarity to the existing class representatives (0.4 in Table V). This object is thus selected as representative for a third class. All other employees are too similar to the existing class representatives, so no further classes are created. Finally, unclassified employees are assigned to the defined classes according to the maximal similarity, leading to the following class definition

$$C_1 = \{\text{RS1201}, \text{LS1050}, \text{DH2323}\}$$

$$C_2 = \{\text{JA1210}, \text{MR0120}\}, C_3 = \{\text{LL1012}, \text{CS1910}\}.$$

Note, that after the definition of those three classes, the creation of a fourth class is not feasible: CS1910 and LS1050 are the objects with the lowest maximum similarity to the existing class representatives; however, their values are already 0.75, in other words, above the threshold of 0.7.

4.2.3. Calculate result quality.

As a last step of the employee classification algorithm, the overall quality of the results is calculated. In the context of data quality analysis the classification results are considered the better the more homogenous classes are in terms of the included objects. The contROLE methodology facilitates the average distance of employees within a certain class C_i for gathering a classification quality value. Classes with a high-homogeneity value which is below 1 hint at two types of outliers. Firstly, a small number of class members might be assigned to permissions which have not been correctly de-provisioned. Secondly, some class members might be missing permissions that are typical for the respective class, leading to an increase of the heterogeneity value.

The class heterogeneity can be calculated according to

$$\text{Heterogeneity}(C_i) = \frac{2}{|C_i| \times (|C_i| - 1|)} \times \sum_{i,j \in C_i} d(i, j)$$

with $d(i,j)$ being the distance between two class members i and j. In respect to class C_3 the heterogeneity is 0.25. Linda Loner and Chris Summer (LL1012, CS1910) are both grouped into one class but Linda's excessive permission P3, which she no longer needs for her work, decreases the result quality of the class definition. In real-life examples employees have a larger number of permissions, resulting in a more fine-grained result quality differentiation. Thus, appropriate thresholds for highlighting permission outliers can be defined and the employee classification can be used for detecting them.

4.3. Detect permission outliers

The last process step during semantic data analysis deals with outliers concerning the distribution of single permissions among the hierarchy elements of the organization, carried out by a (a) rare permission check and the inverse (b) common permission check. Both checks are split into a detection and refinement phase. After the initial detection of possible outliers a crosschecking reduces the amount of outliers that are communicated to domain experts without actually being an error (false-positive rate). As an example, Figure 9 visualizes the candidate permission outliers detected by both checks for a hierarchy unit with 500 users.

A rare permission is defined as an access privilege that is only assigned up to a certain percentage of employees within the organization (lower bound parameter). In the example three permissions are marked as rare permissions (12, 3, and 89). These permissions could be local or individual permissions needed for specialist tasks. However, they

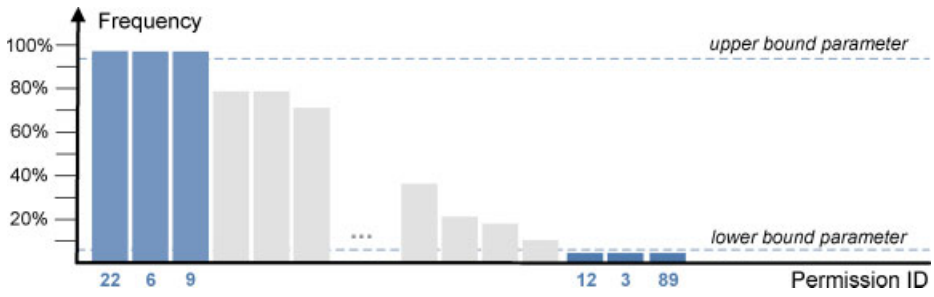

Figure 9. Rare permission and common permission detection.

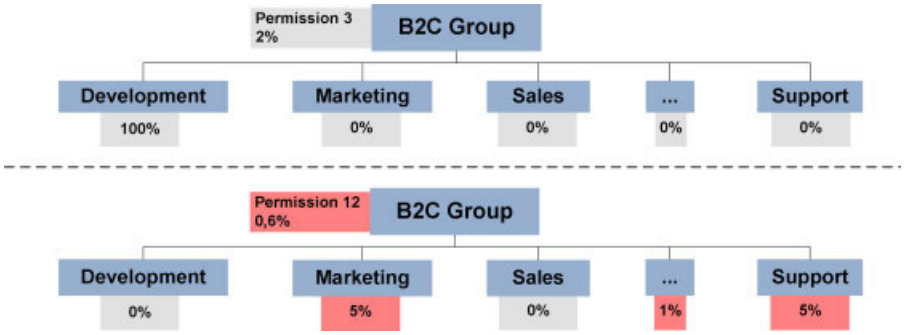

Figure 10. Permission outlier refinement.

could likewise be no longer in use but have not been deprovisioned correctly or be orphaned accounts referring to employees who may not work in the company any more.

Therefore a refinement step investigates every organizational unit with at least one employee assigned to a suspicious permission. The percentage of employees assigned to the respective permission in these hierarchy elements needs to be below a predefined threshold in order to mark the dataset as outlier.

Figure 10 illustrates the refinement of the candidate outlier permissions 3 and 12 from the example above. Permission 3 is assigned to 10 employees (2 per cent of 500 employees). It is exclusively used in the Development department (10 employees). Thus permission 3 is likely no outlier but related to specialist tasks. In contrast, the lower part of the picture shows the crosschecking of permission 12 which is assigned to three employees spread across the whole organization. This permission thus might be considered as an outlier that needs to be further investigated by a domain expert. The common permission check is reverse to the rare permission check. It investigates permissions that are assigned to a very high percentage of the employees (permissions 22, 6, and 9 in Figure 9). The goal is to identify missing user-permission assignments. The most critical stage in applying both aforementioned checks is the parameterization of the bounds for detecting potential errors as well as the refinement threshold. One indicator could be the average number of employees per investigated organizational unit in case of a low standard deviation. The same holds for the refinement threshold which could be defined depending on the average employee count in the different departments.

In order to cleanse the semantic errors and the potentially unresolved syntactic errors, the results have to be sent for review to a human domain expert. The errors are bundled according to elements of the organizational hierarchy and users together with the proposed correct values. The domain expert can then accept this proposal or alter the data. By exposing the correct input data elements to the productive systems in place, the quality of the identity and account data has been advanced and the risks for security breaches and system misuse by insiders has been considerably minimized in a timely manner.

In general, applying syntactic and semantic cleansing of identity and account data has high potential for reducing the risk of insider misuse. It significantly contributes to increasing the quality of identity and account information by pointing to errors in directories, outdated privileges, orphaned accounts, or existing privileges which might not be necessary to perform the job. Analyzing and cleansing account information is also a prerequisite for structuring the user population according to typical user roles. Having proper knowledge of potential roles is rudimental for role-based access controls [17], which also has significant benefit to hinder insider misuse.

5. CASE STUDY SEMICOND

After the data cleansing mechanisms applied during the execution of the contROLE methodology have been presented, the paper continues with their evaluation in a real-world application scenario, using a complex and potentially erroneous dataset provided by a large cooperation operating in many countries all over the world.

5.1. Data gathering

The input data used (from hereinafter called Access Controls following the terminology of Molloy et al. [13]) originate from the Identity Management repository (Microsoft Active Directory) of a large industrial organization. The company, from hereinafter called SemiC, operates worldwide with about 30 000 employees. For this application scenario the Active Directory domain Asia-Pacific including 8115 employees and their memberships in 7533 different groups is provided. In the following, every group is treated as permission. The SemiC access controls include the employees, their assigned department, location and the group memberships (see extract in Table VI).

Table VII sums up the relevant statistics for the provided Access Controls. During the initial data import duplicate datasets already have been excluded reducing the UPA (user-permission assignments) from 151062 to 150329.

Table VI. Access Controls extract SemiC.

```
Accountname;Location;Department;Permission
roN1w;Malacca;SCMY IT;CN = AP-SemiCEmployees-G
roN1w;Malacca;SCMY IT;CN = MKZ-OU-Users-G
roECn;unspec;SCWU IT MFG;CN = AP-SemiCEmployees-G
roECn;unspec;SCWU IT MFG;CN = WUX-ITCoordinator-G
row6r;Singapore;SCAP IT;CN = SeC-Employees-SG-U
[. . . ]
```

Table VII. Access Controls statistics SemiC.

Access Controls element	Total
Employees	8115
Permissions	7533
Hierarchy elements	1527 (1486 line-, 41 geographic hierarchy)
UPA	151062 (after import 150329)

OrgUnit ID	OrgUnit Name	Proposed Value	Distance	Postpone	Delete
39	Dresden	-	5.0	☐	☐
70	349282	-	6.0	☐	☐
83	San Jose	-	5.0	☐	☐
224	Neubiberg	-	6.0	☐	☐
637	Melbourne	-	6.0	☐	☐
876	Regensburg	-	7.0	☐	☐
953	Grasbrunn	-	6.0	☐	☐
960	Nagoya-shi	Nagoya	4.0	☐	☐
1333	Xian	Xi'an	1.0	☐	☐
1375	Shingawa-ku	Shinagawa-ku	1.0	☐	☐
1378	Sinagpore	Singapore	2.0	☐	☐
1384	Pioneer St. Mandaluyong	-	17.0	☐	☐
9999	Hsinchu	Hsin-Chu	2.0	☐	☐

OK

Figure 11. ContROLE consistency check results.

5.2. Syntactic data cleansing

Executing the similarity checks introduced earlier is not reasonable as the employee names were randomized and the organizational unit names are abbreviations. Only for the location attribute can suggestions according to the Levenshtein distance [18] be made. Additionally, all missing location and department values of users have been set to the valid null-value UNSPECIFIED for further investigation (missing value check).

In our scenario a list of valid organizational units for the line – and the geographic hierarchy has been provided. Checking the consistency of the line organization marked 263 out of 1486 hierarchy elements as invalid. One reason for this high number might be the large amount of organizational restructuring within SemiC which took place over the last years. Old organizational units might not have been de-provisioned correctly and thus still exist in the directory environment.

Besides the line organization, the consistency of the geographic structure of SemiC has been analyzed (Figure 11). Several datasets with a location value from other regions than Asia are identified together with cryptically named locations. The related datasets (12 employees, 238 UPA) might represent accounts of employees that have been re-assigned to a new site while their location attribute has not been updated. Thus, the null-value UNSPECIFIED is assigned to affected employees and the related UPA are further investigated during the semantic data analysis. Secondly, several datasets with a misspelled location attribute have been revealed (e.g., Xi'an instead of Xian; Levenshtein distance 1.0).

Syntactic data checking revealed that a total of 263 out of 1486 organizational units in the line organization and 15 out of 41 OHE in the geographical hierarchy have been identified as erroneous. Carrying out the referential integrity check revealed 32 employees with two or more assigned hierarchy elements of the same OHE type. This small number of violations might be the result of previous consolidation efforts of the Identity Management team within SemiC. In total the previous syntactic data cleansing efforts reduced the number of UPA included in the Access Controls from 150329 to 146584 and the total number of employees from 8115 to 7576. In terms of insider threat the results show a large number of active user accounts with invalid attribute assignments. The related access rights represent major security holes for insider attacks.

5.3. Semantic data cleansing

In the following, the different aspects of semantic data cleansing are described in more detail, beginning with the identification of employee outliers, and continuing with the detection of permission outliers using the previously presented techniques.

5.3.1. Identify employee outliers.

Employee outlier detection is carried out for the pre-cleansed dataset on the basis of a semi-automatic SOM analysis. Figure 12 presents the SOM visualization of the geographic hierarchy of SemiC[‖]. It can be seen that employees working in the same locations are in general located near to each other (same coloring). However, areas where users from different locations are located close to each other are also visible (centre part of Figure 12). These areas either hint at permission bundles (and thus roles) that are valid throughout several locations or could represent erroneous data elements.

For deciding about which of the employees are considered as potential outliers for attribute value re-assignment, our tool allows for different threshold levels during the

[‖] Note that the lattice numbers are only partly visible as they are overlaid by the pie charts. The depicted map, however, visualizes all 7576 remaining employees in the input data.

Security Comm. Networks 2012; **5**:847–862 © 2011 John Wiley & Sons, Ltd.
DOI: 10.1002/sec

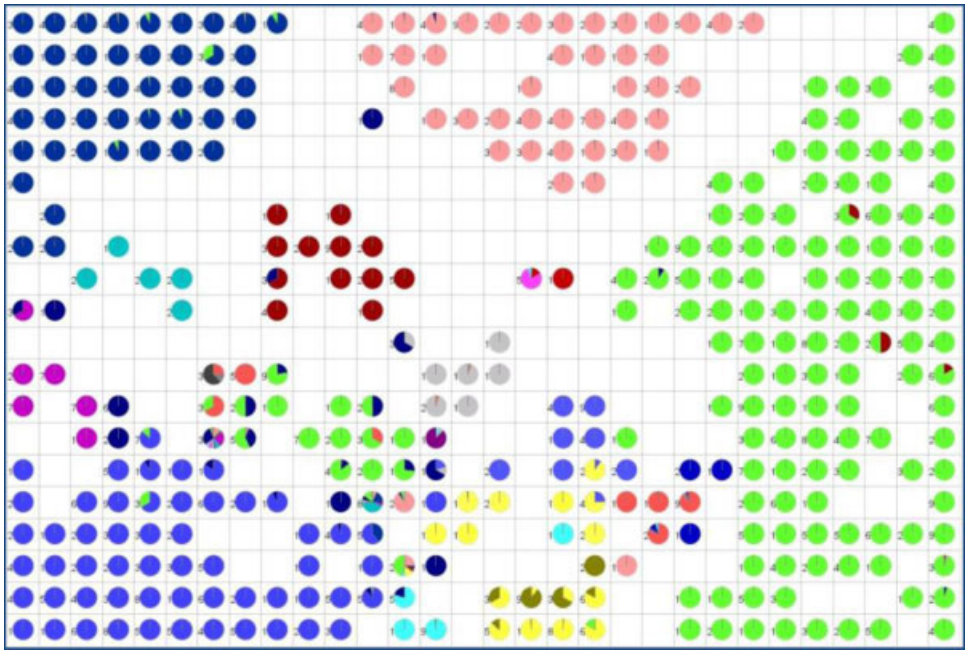

Figure 12. SOM analysis of the SemiC geographical hierarchy.

analysis steps. In the given example, error detection and data cleansing is carried out within seven iterations, using the manual map investigation, and industry partner feedback as abort criteria. During the first iteration nodes with more than 75 per cent, but less than 100 per cent of the node members being assigned to the predominant node–class are marked as outliers. The refinement step validates whether their first neighborhood level is also dominated by the same class by more than 75 per cent. If this condition is true the node is cleansed. During the consecutive iterations these threshold values are adapted based on the manual map investigation in order to identify remaining outliers. The last iteration involves a manual selection of suspicious datasets not identified previously.

Focusing on the upper left part of Figure 12, Figure 13 shows the effects of the aforementioned cleansing process using the SOMParsing algorithm iteratively. On the left side several outliers with the location attribute Singapore (green) are depicted in the group of users working in Kulim. The contROLE toolset thus extracts these datasets and proposes Kulim (blue) as correct location attribute value. Remember that such employees with wrongly assigned location attributes negatively influence the consecutive role development process. Additionally, note that the cleansing process described requires human interaction in order to finally decide if a suspicious dataset is erroneous or not. The contROLE toolset allows for the integration of business know-how for acquiring high-quality results.

5.3.2. Identify permission outliers.

Subsequent to the employee outlier detection the common permission analysis reveals potentially missing UPA. Executing it with the exemplary upper bound of 0.95 (line organization) resulted in 175 UPA of this type. Additionally, the rare permission check is executed for the level-1

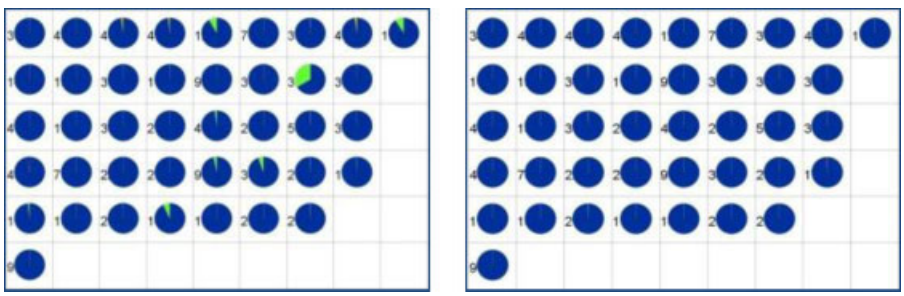

Figure 13. Employee outlier cleansing with contROLE (erroneous *vs.* cleansed data).

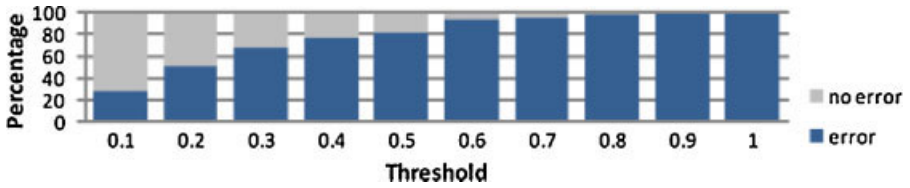

Figure 14. Rare permission check refinement results.

line organization departments in order to reveal permissions which are likely to be no longer needed but existing due to incorrect de-provisioning processes.

At first, the respective threshold needs to be thoroughly set. If 5 per cent or less of the employees within a level-1 department, e.g., are assigned to a certain permission, the check considers 43630 of the remaining 146759 UPA (29.7 per cent) suspicious. However, in order to minimize the false positive rate the restrictive bound of 0.01 has been used in the following, highlighting 20288 potentially erroneous UPA (13.8 per cent of the total UPA).

Consecutively, the refinement loop excludes UPA assigned to more than a certain percentage of the employees within a non-aggregated department. Figure 14 depicts the percentage of the 20288 suspicious UPA considered erroneous after the refinement loop (blue coloring) in relation to the excluded UPA (gray coloring) depending on the used refinement threshold. It can be seen that a high-refinement parameter leads to all 20288 suspicious datasets being considered erroneous while a low parameter excludes a high percentage of them from further investigation. During our evaluation process a restrictive refinement parameter of 0.1 was applied in the following in order to minimize the false-positive rate (5852 candidate errors).

5.4. Data cleansing impact

This section briefly sums up the impact of the data cleansing efforts for reducing insider misuse, retrospectively underlining the importance of this contROLE phase (Table VIII.). Overall, an average reduction of the input data elements within the SemiC Access Controls of about 12.75 per cent has been achieved carrying out the described checks. The result refinement in shows that the number of permissions even could be reduced by 18.31 per cent. This underlines the large number of potentially outdated but still accessible permissions within the provided input data. The results moreover underline that the high reduction of permissions only has little impact on the existing UPA as most excluded permissions are individual permissions.

Note that these statistics to not depict the numerous re-assignments of attribute values carried out during the semantic data cleansing. Additionally, executing the employee classification checks would have led to even higher error detection and data cleansing rates. However, the investigation of employee classes was out of scope of the case study and thus not executed.

Table IX. Access Controls statistics RetComp.

Access Controls element	Total
Employees	2393
Permissions	881
Hierarchy elements	431 (337 line organization)
UPA	76665

6. CASE STUDY RETCOMP

In order to show the applicability of our methodology and toolset we carried out a second case study dealing with a medium-sized organization in the retail sector.

6.1. Data gathering

The input data used in the second case study originate from the IdM repository (Microsoft Active Directory) of the medium-sized company, from hereinafter called RetComp. The company operates in large parts of Europe with more than 2000 employees. The company-wide Active Directory includes nearly 900 different permissions (AD groups, see Table IX).

Every group is treated as permission in the remainder. In the following the one central department of RetComp (Purchasing: 279 employees, 252 permissions, 11221 UPA)

Table VIII. Data cleansing impacts on the SemiC data.

Access Controls element	Raw input	After cleansing	Reduction (per cent)
Employees	8115	7576	6.64
Permissions	7533	6154	18.31
Hierarchy elements	1527	1232	19.32
UPA	151062	140907	6.72

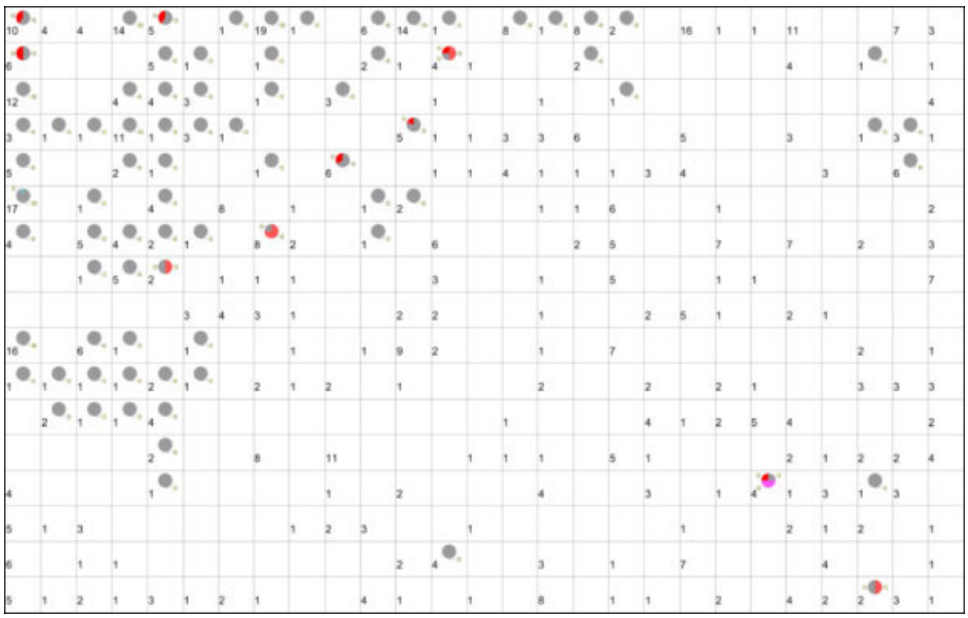

Figure 15. Employee outliers in the purchasing department.

is investigated in detail, focusing on the semantic anomalies. The syntactic data analysis results are not shown in detail at this point, as the goal of the analysis was the discovery of employees with untypical access rights and employees with excessive permissions which are no longer needed and represent a threat for security.

6.2. Semantic data cleansing

In the following the employee outlier detection using SOMs as well as the clustering of employee groups and the statistical analysis of permission assignments have been carried out in order to extract semantic data anomalies.

6.2.1. Identify employee outliers.

At first, a neural network analysis was carried out for the department. Note that for readability reasons not the whole neural network but just the essential parts for this interpretation are shown.

The map in Figure 15 shows the members of the department grouped in the left upper corner in gray color. The remainder of the map depicts employees in other departments of RetComp (uncolored). One can see that most of the employees within Purchasing represent a homogenous group with similar access rights typical for that part of the company. However, the analysis extracted several outliers representing employees that are members of Purchasing but have untypical access rights. Overall, a total of 8 per cent of the employees in Purchasing (22 out of 279) were marked as outliers. A detailed analysis integrating expert knowledge from Identity Management representatives of RetComp revealed that most of the marked outliers are identities used by trainees. Those trainees typically were reassigned to different departments over time and accumulated excessive permissions, making them a potential source for insider threats.

6.2.2. Cluster analysis.

In addition to the neural network investigation, a cluster analysis was executed to detect employees with untypical permissions. It on the one hand underlined the homogeneous permission structures within the department Purchasing and, therefore, pointed out its suitability for later role development. Nevertheless, several anomalies were identified.

The following example (Figure 16), e.g., reveals a sub-department with 12 employees from which 11 with 36 permissions in average have been grouped in an employee class with a class similarity of more than 90 per cent. The

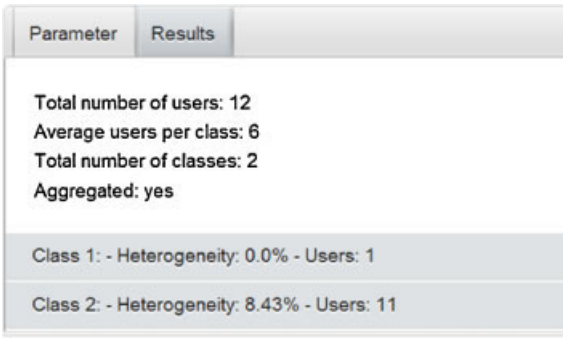

Figure 16. Cluster analysis of a sub-department of purchasing.

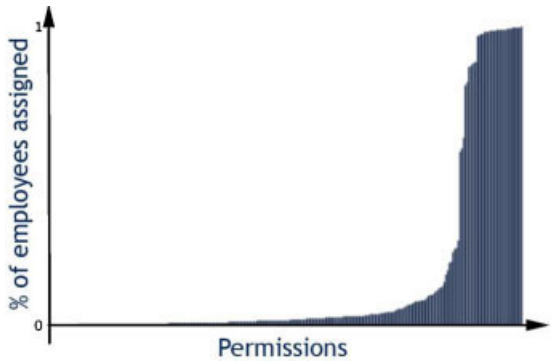

Figure 17. Permission distribution in the purchasing department.

second employee class has only one member (the departmental head) which has different access rights assigned. Several employees of the large class, however, had additional permissions assigned and one employee was missing a permission that the other 10 employees were assigned to. Even though the employees were still grouped, the class heterogeneity is not null, leading to the necessity of further investigation.

Overall, the results of the cluster analysis have underlined its applicability to detect erroneous identity data, employee accounts, and permissions. Again, one core aspect of the cleansing process is the manual re-investigation of domain experts after the detection of potential outliers. Due to the semantic background knowledge needed a fully automated data cleansing process is not feasible.

6.2.3. Identify permission outliers.

Besides the neural network analysis and the employee classification, an investigation of the access rights structures was carried out to detect common and rare permissions. In the Purchasing department 51 of the 252 different permissions (20 per cent) are assigned to only one employee. Refining the results with the permission spreading algorithm reduced the error rate to 13 per cent. Nevertheless, this high number of permissions is currently re-investigated for appropriate provisioning. With this re-investigation, the Identity Management representatives of RetComp aim at reducing administrative overhead by reducing the total number of permissions used by employees as well as license costs for permission usage (e.g., Operating System licenses, database and development software licenses, etc.).

Figure 17 visualizes the permission distribution, showing the large number of different permissions (x-axis) which are assigned to a very small number of employees (y-axis) on the left side. On the right side the permissions assigned to nearly every employee are depicted. The analysis revealed 26 of 252 permissions as common permissions that can be bundled within a departmental role. The detailed analysis in Figure 18 shows that several access rights are assigned to more than 99 per cent of the employees in Purchasing. The representatives of RetComp, therefore, needed to re-investigate the missing assignments and correct those potential errors. One example of missing assignments might be Microsoft Office licenses (276 out of 279 employees) or Microsoft Windows XP Service Pack licenses (277 out of 279 employees).

6.3. Data cleansing impact

This section briefly sums up the impact of the semantic data cleansing efforts for reducing insider misuse, retrospectively underlining the importance of this contROLE phase within the purchasing department of the RetComp. Firstly, 26 common permissions (out of 252 permissions in

Detailed Results

PERMISSION ID	PERMISSION NAME	PERCENTAGE	COUNT
44	CN=XP_LocalAdminAutoIt,	100%	279
42	CN=XP_Silverlight,	100%	278
12	CN=XP_ADRegistrierung,	100%	278
45	CN=XP_SQLNet_11g,	100%	278
47	CN=XP_DeviceWatch,	100%	278
53	CN=XP_USBdelete,	100%	278
43	CN=XP_BiosUpdate,	99%	277
54	CN=XP_SP3,OU=	99%	277
55	CN=XP_Sophos50,	99%	277
36	CN=XP_Office2007,	99%	276
21	CN=XP_Pb10user,OU=	99%	276
20	CN=XP_Pb10user_9731,	99%	276
19	CN=XP_Sql10g,	99%	276
18	CN=XP_ZFDAgent,	99%	276
13	CN=XP_ADClient,	99%	276
34	CN=XP_Java2RE160,	99%	275

Figure 18. Common permissions in the purchasing department.

total) have been bundled in one departmental role, reducing the UPA from 11221 to 4948 (−55 per cent). This departmental role is now used for automatically (de-)provisioning a large amount of the required access rights. Secondly, the neural network analysis and the cluster analysis led to the cleansing of numerous wrongly assigned permissions. Note that employees that have been considered outliers in the neural network analysis could also have been detected as suspicious during the cluster analysis.

7. CONCLUSION AND FUTURE WORK

Effectively administrating employees' access to sensitive applications and data in order to reduce the risk of insider misuse 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. This paper has shown that not carefully maintained user accounts of individuals that have a trusted relationship with organizations – namely (former) employees, contractors, or consultants – represent a typical potential threat for Information Security. Those authorized non-technical insiders directly interact with an organization's Information Systems, have some type of authority on those systems, and know about security processes and how to circumvent them. Because detection methods mainly rely on rule-breaking behavior, a misuse performed by those types of users is very difficult to detect. Making security officers and CIOs aware of these threats is an important part of mitigating the risk of insider misuse.

During their lifetime in the organization, employees usually develop a personal career and migrate between different jobs and assignments. Each change implies new duties and responsibilities which in general come along with new and additional obligations and access privileges. As a consequence, many users possess more access privileges than necessary to perform their actual job, permissions exist which are not used anymore, or accounts are still valid for which users that already have left the organization.

This paper dealt with the risk of system misuse due to bad quality of the identity and account data. In order to encounter the related risk of insider misuse, we proposed the methodology contROLE for structured Identity Management including syntactic and systematic cleansing of account data. It was shown that cleansing of identity and account data results in a considerable increase of data quality. User accounts and permissions which did not reflect the current job function of the employees, orphaned accounts, inconsistencies, errors, and permissions no longer needed could be detected and resolved. We gave a general overview of the methodology, background information on the cleansing algorithms we are using and a report about the results gathered from two real-life application cases.

For future work we are currently developing and evaluating additional semantic data cleansing checks which aim at identifying employees and departments with untypical and excessive permissions assigned in specific departments. This for instance includes detection mechanisms for employees with excessive permissions in single departments. Additionally, we are extending the contROLE PCE with standardized data exchange interfaces. Our goal is furthermore to enable the contROLE methodology to be able to include digitally available business processes or UML diagrams annotated with security requirements during the checking of security policies concerning digital identities and account data.

ACKNOWLEDGEMENTS

The tool supporting the presented methodology extends an open-source implementation of SOMs developed in the SOMLib Digital Library Project by the Information & Software Engineering Group at the Vienna University of Technology.

REFERENCES

1. Peltier TR. Information Security Policies, Procedures, and Standards: Guidelines for Effective Information Security Management, 1st edn. Auerbach Publications: Boca Raton, FL, USA, 2001.
2. Bidgoli H. Handbook of Information Security, Information Warfare, Social, Legal, and International Issues and Security Foundations (Handbook of Information Security), 1st edn. John Wiley & Sons, Inc.: New York, NY, USA, 2006.
3. Department for Business, Enterprise & Regulatory Reform: *Information Security Breaches Survey*, 2008. www.pwc.co.uk/pdf/BERR_ISBS_2008(sml).pdf. Retrieved: 27th July 2009.
4. Richardson R. Computer Crime & Security Survey 2008. Computer Security Institute: San Francisco, CA, USA, 2008; i.cmpnet.com/v2.gocsi.com/pdf/CSIsurvey2008.pdf. (Retrieved: 27th July 2009).
5. Dhillon G. *Violation of Safeguards by Trusted Personnel and Understanding Related Information Security Concerns*. In: *Computers & Security* 20 (2001) No. 2, pp. 165–172.
6. Cappelli D, Moore A, Shimeall T, Trzeciak R. *Common Sense Guide to Prevention and Detection of Insider Threats*. Carnegie Mellon University, CyLab. Available at: www.cylab.cmu.edu/files/pdfs/CERT/CommonSenseInsiderThreatsV2.1-1-070118-1.pdf 2006.
7. Sarbanes PS, Oxley M. *Sarbanes-Oxley Act of 2002* (Pub.L. No. 107–204, 116 Stat. 745), frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=107_cong_bills&docid=f:h3763enr.tst.pdf 2002.

8. Volino L, Gessner GH, Kermis GF. *Sarbanes-Oxley Links IT to Corporate Compliance*. In: Proc. of the 10th Americas Conference on Information Systems, New York, New York (2004).
9. Bank for International Settlements BIS: *Basel II: International Convergence of Capital Measurement and Capital Standards: A Revised Framework – Comprehensive Version*. Available at: www.bis.org/publ/bcbs128.pdf 2006.
10. Federal Office for Information Security (BSI): *IT-Grundschutz*. Available at: www.bsi.bund.de/english/gshb/index.htm 2004.
11. European Union: *Directive 95/46/EC of the European Parliament and of the Council. Official Journal of the European Communities of 23 November 1995 No L. 281 p. 31*. Available at: www.cdt.org/privacy/eudirective/EU_Directive_.html 1995.
12. Fuchs L, Pernul G. *Reducing the Risk of Insider Misuse by Revising Identity Management and User Account Data*. 2nd Int. Workshop on Managing Insider Security Threats, 2010, Morioka, Iwate, Japan. 2010.
13. Molloy I, Chen H, LI T, *et al*. *Mining Roles with Semantic Meanings*. In: Proc. of the 13th ACM Symposium on Access Control Models and Technologies (SACMAT'08), pp. 21–30, Estes Park, CO, USA, 2008.
14. Broser C, Fuchs L, Pernul G. *Different Approaches to in-house Identity Management*. In: Proc of the 4th International Conference on Availability, Reliability and Security (ARES 2009), IEEE Computer Society, Fukuoka, Japan.
15. Fuchs L, Müller C. *Automating Periodic Role-Checks: A Tool-based Approach*. In: Proc. Business Services: Konzepte, Technologien, Anwendungen. 9, Internationale Tagung Wirtschaftsinformatik 2009, Vienna, Austria.
16. Fuchs L, Pernul G. *HyDRo – Hybrid Development of Roles*. In: Proc. 4th Int. Conf. on Information Systems Security (ICISS 2008), Hyderabad, India, LNCS 5352, Springer, Berlin.
17. Zhu X, Wu X. *Class noise vs. attribute noise: A quantitative study of their impacts*. In: *Artificial Intelligence Review* 22 (2004) No. 3, pp. 177–210.
18. Levenshtein VI. *Binary Codes Capable of Correcting Deletions, Insertions, and Reversals*. In: Doklady Akademii Nauk SSSR 163 (1965) No. 4, pp. 845–848.
19. Pohlman M. Oracle Identity Management: Governance, Risk, and Compliance Architecture, 3rd ed. Auerbach Publications: Boca Raton, FL, USA, 2008.
20. Sandhu RS, Coyne EJ, Feinstein HL, Youman CE. *Role-based access control models*. In: *IEEE Computer* 19 (1996) No. 2, pp. 38–47.
21. Ferraiolo D, Kuhn R, Chandramouli R. Role-Based Access Control, 2nd ed. Artech House: Boston, MA, USA, 2007.
22. Mendling J, Strembeck M, Stermsek G, Neumann G. *An Approach to Extract RBAC Models from BPEL4WS Processes*. In: Proceedings of the 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'04), pp. 81–86, Washington, DC, USA, 2004.
23. Klarl H, Wolff C, Emig C. *Identity Management in Business Process Modelling: A Model-Driven Approach*. In: 9. Internationale Tagung Wirtschaftsinformatik, Business Services: Konzepte, Technologien, Anwendungen (WI'09), Vienna, Austria, 2009, 161–170.
24. Kohonen T. *Self-organized Formation of Topologically Correct Feature Maps*. In: Biological Cypernetics 43 (1982) No. 1, pp. 59–69.
25. Jaccard P. Etude comparative de la distribution orale dans une portion des alpes et des jura. *Bulletin del la Socieete Vaudoise des Sciences Naturelles* 1901; **37**: 547–579.