Reducing the Risk of Insider Misuse by Revising Identity Management and User Account Data

Ludwig Fuchs
Nexis GmbH, www.nexis-secure.com
Department of Information Systems
University of Regensburg, Germany
Ludwig.Fuchs@wiwi.uni-r.de

Günther Pernul
Department of Information Systems
University of Regensburg, Germany
Guenther.Pernul@wiwi.uni-r.de

Abstract

To avoid insider computer misuse, identity and authorization data referring to the legitimate users of the systems must be properly organized and constantly and systematically analyzed and evaluated. In order to support this, a methodology for structured Identity Management has been developed. This methodology includes 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. Each of the steps involved includes quality criteria and comprehensive tool support. The focus of this paper is on the data cleansing phase leading to feedback where insider misuse may occur due to existing privileges which go beyond the scope of the users' current need-to-know.

1 Introduction

Insider misuse of computing resources takes on a variety of forms. Looking at the problem at a first glance, one might only think of the malicious insider who attempts to harm the organization or acts in a purposeful way that threatens the organization's interests. Sometimes, nevertheless, there is a fine line between malicious intent and mere misuse. Insider misuse does not necessarily need to be malicious to pose a threat to the organization. For example, consider an employee downloading music or video to a desktop computer would not usually be doing so with intent to harm the organization. However, if the files being downloaded are pirated or if a peer-to-peer file-sharing program is used, the employee is putting the organization at risk or may even through the P2P-software inadvertently give outsiders access to confidential files stored on the computer. Additionally, of course, there is also the insider who accidentally misuses the system because of wrong use or an error.

To summarize, there is no general way of IT misuse by insiders. We are confronted with a range of potential scenarios. Insiders 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 authorizations. It such an environment the 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. investigated this problem and possible ways of solving it in [2].

According to a recent survey conducted by the American Society for Industrial Security [1], current and former employees and on-site contractors with authorized access to facilities and networks pose the most significant risk to intellectual property, such as research data, customer files, and financial information. In general, employees are over-authorized causing a high threat to IT misuse by insiders. During their lifetime in the organization employees are not statically assigned to a certain job but migrate between different job functions and assignments. Each change implies new duties and responsibilities which often go along with new and additional access privileges to the IT resources. Over the time, most employees mainly acquire access rights which only very rarely are dispensed later even when they are no longer

[1] http://www.asisonline.org/

needed. 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 deleted from all the directory systems. The situation described above is sometimes referred to as the so called "identity chaos". The term 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). To further complicate matters, each security domain may be subject to different rules, allow access to different security levels and passwords and accounts may expire after a certain time or may not expire at all. Given these factors, it is not surprising that the Aberdeen Group [2] states that only 17 percent of companies claim they do not have orphaned accounts (accounts with access that should have been revoked). Some organizations take more than 30 days to decommission accounts and others have no defined processes for decommissioning nor means to discover if orphaned accounts even exist.

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 [11, 13], Basel II [4], the German BSI Grundschutz [3], the Directive 95/46/EC of the European Parliament [12], 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. In order to fight the identity chaos and the risk of insider misuse which comes along 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, section 3 has a focus on data cleansing and section 4 contains the evaluation of our methodology by performing a case study with account data of a real company. Section 5 contains the conclusion and future work.

2 General Overview of contROLE

ContROLE is a methodology and corresponding tool set supporting a structured Identity Management process. The process consists of six different phases (see Figure 1) which roughly can be grouped into three major functional units. Early phases are concerned with gathering identity and account information as well as information about the organizational structure of the enterprise. It is 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 which may serve as basis for role-based access controls. The methodology may be applied as a whole (leading to a role catalog) or only partly, for example, data cleansing process steps only.

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

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

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. Users may be assigned to certain roles but in a particular point in time may only play one role. Playing the role will restrict them to only those privileges and accounts which are necessary to perform the job functions corresponding to the role.

Some more information about contROLE can be collected at www.nexis-secure.com. Different aspects of the methodology are already published, i.e. the general process of in-house Identity Management [1], tool support for structured Identity Management [5], and the process of semi-automated generation of roles [6]. The following is a short description of each of the contROLE process steps (see Figure 1). Cleansing of identity and account data as the major aspect of preventing insider misuse will be described in more detail in Section 3 of this paper.

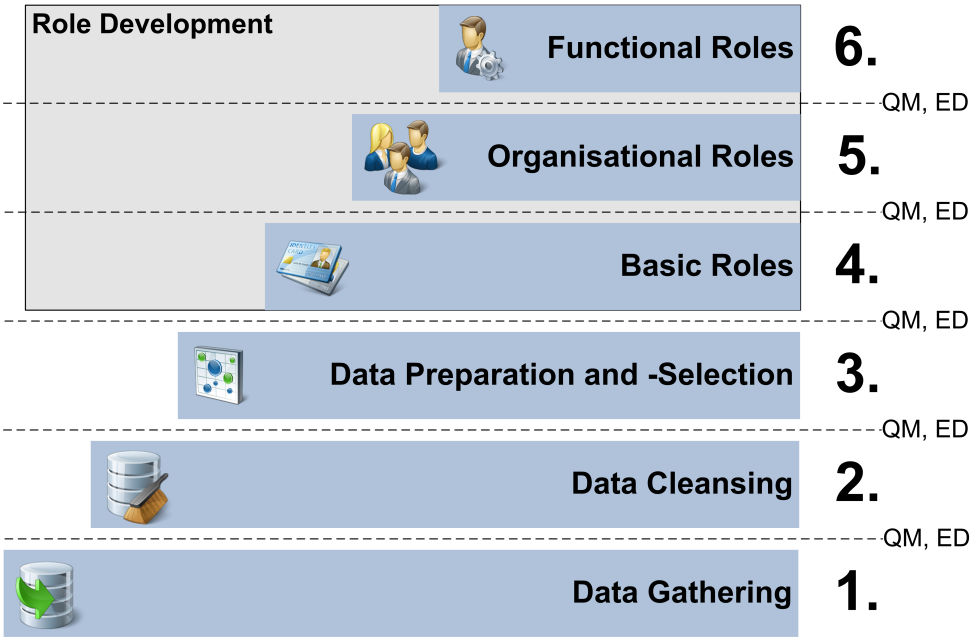

Figure 1: Structured Identity Management process according to contROLE

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. Besides the mandatory identity information, data from the organizational structure of the enterprise is optional but highly desired. It might be available in forms of defined job positions, task bundles, or already existing local role definitions within certain departments.

After input data has been cleansed (see Section 3), the process moves on to *Data Preparation and Selection*. 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. During this phase contROLE develops several parameters which offer a first feedback whether proper input data is available so that role development might be successful. Examples of parameters are number of privileges held only by a small number of users, comparison of different departments, amount of cleansed input data, or even more complex such as the grade of interdependencies between hierarchical elements in the organization.

In the case feedback is not sufficient the methodology suggests revising the earlier process steps. 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, such as functional roles. 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.

3 Cleansing Identity and Account Data

Analyzing and cleansing of existing identity and account data is central to reducing the risk of insider misuse. From the previous phase contROLE 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. As usual with many types of real-world data [14] identity and account information also tends to be incomplete, noisy, and inconsistent. The main goal of this phase is to increase the data quality, detect errors and inconsistencies, orphaned accounts, or authorizations which go beyond to what the employee needs to know to perform his work. Syntactic checks aim at revealing errors regarding the input data entities while semantic checks try to identify inconsistencies in the relationships among those entities. 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. After errors have been detected and cleansed, the updated data is written back to the originating sources.

3.1 Syntactic analysis of identity and account data

Syntactic analysis follows the process described in Figure 2 (light grey 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.

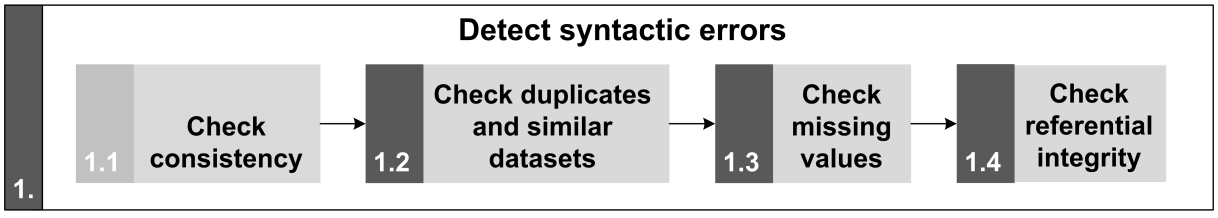

Figure 2: contROLE process steps for syntactic analysis

Consistency check. 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 hierarchy elements of the organization. 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 [8] 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 (if the entry analyzed does not have any semantic meaning, for instance, abbreviated organizational unit names), the consistency check by default assigns a null-value,

marking the respective datasets for further investigation. The same holds for predefined null-values included in the valid value lists.

Duplicate and similarity check. The 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 used for the detection of errors and the proposal of correct values for misspelled datasets.

Missing value check. The 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.

Referential integrity check. Referential integrity checks investigate the relationships among the datasets. Depending on predefined policies, several restrictions could exist, for example the policy that every employee needs to be assigned to exactly one component within at least one hierarchy type of the organization. In the case empty assignments are determined the dataset is marked for further handling during the semantic checks or manual validation by a domain expert.

3.2 Semantic analysis of identity and account data

In addition to the syntactic checks, contROLE provides further functionality for semantic analysis of the input data. Focusing on the relationship between permissions, employees, and organizational hierarchy elements, semantic error detection is used to detect (a) employees with authorizations which do not match their job functions (for example, 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 also follows a process model which is described in Figure 3. While the permission outlier checks reveal potentially erroneous user-permission assignments for deletion, the employee outlier checks highlight attribute values of employees which might be subject to re-assignment. Technologies used by the contROLE toolset to identify semantic errors are statistical analysis, clustering, mining and artificial neural networks. In order to explain a semantic error to the domain expert data needs not only be analyzed but also properly visualized. For detecting employee outliers we have very good experiences with self-organizing maps (SOMs) as proposed by Kohonen [7]. Before the detection can be conducted, the underlying SOMs have to be parameterized and trained. For example, identity and account information from a specific element of the organizational hierarchy or even the whole input dataset can be selected for investigation. 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. A typical classifier often selected is the attribute used to assign an employee to an organizational unit.

Detect employee outliers

The first type of semantic checks focuses on the detection of identity and account data of employees with atypically assigned permissions or attributes. contROLE is able to detect the following outliers: Wrong user attribute values, null-values in accounts, wrong permission assignments. It also suggests correct attribute values for the detected datasets and passes this information for manual inspection to a domain expert. In selected cases errors might also be cleansed without any human interaction. As an example to illustrate how detection of employee outliers and attribute cleansing work by using SOMs consider Figure 4. The example gives part of a SOM used to classify identity and account information of employees based on the attribute ”assignment to organizational hierarchy element (OHE)”, i.e. every employee is

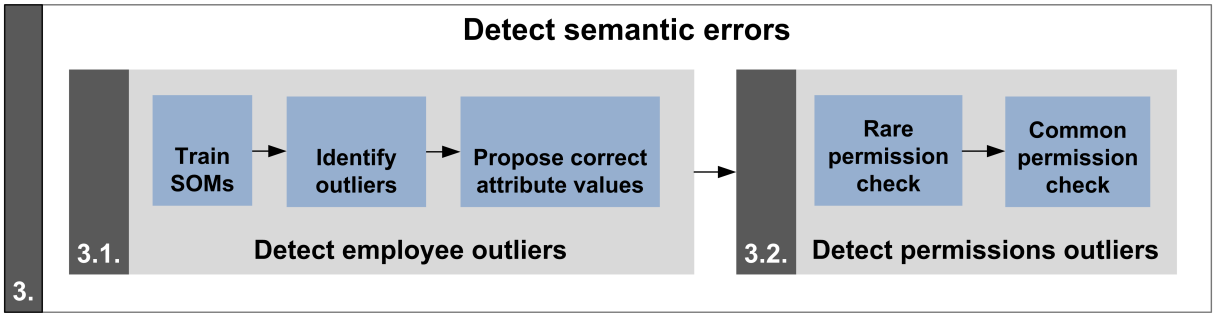

Figure 3: contROLE process steps for semantic analysis

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 4).

After detection, it needs to be decided about treatment of the candidate errors. For proposing a correct attribute value, contROLE analyses 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 (hierarchy level 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 OHE of Max Strasser. Note that in several cases automatic re-assignment of the respective attribute value is possible, for example, in the case of null-values. However, the integration of domain expert knowledge in order to optimize the results is highly recommended.

Detect permission outliers

The second type of semantic checks 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 cross-checking reduces the amount of outliers that are communicated to domain experts without actually being an error (false-positive rate). As an example, Figure 5 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 above 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 could likewise be no longer in use but have not been de-provisioned correctly or 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

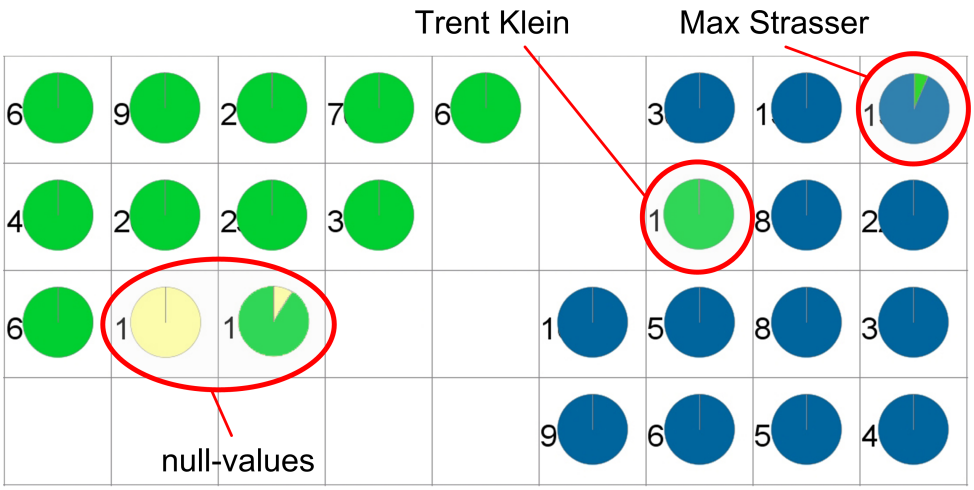

Figure 4: Employee outlier detection using SOMs

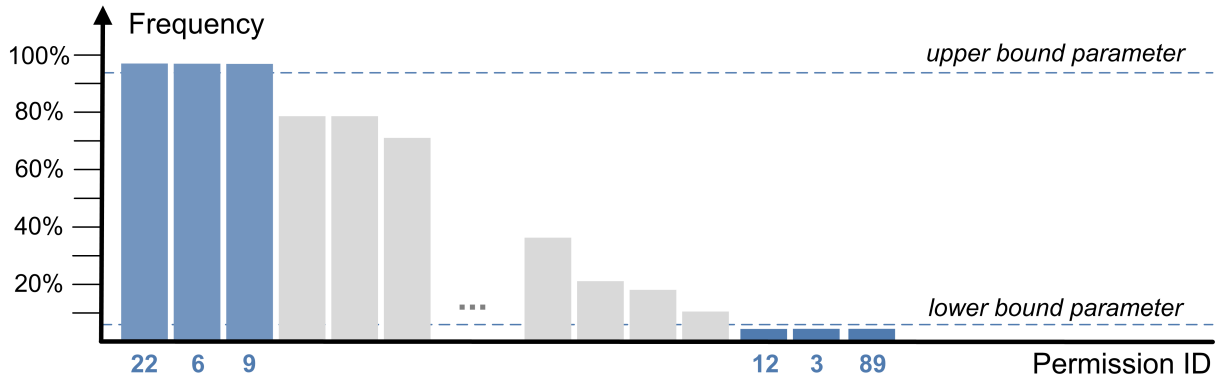

Figure 5: Rare permission and common permission detection

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 6 illustrates the refinement of the candidate outlier permissions 3 and 12 from the example above. Permission 3 is assigned to 10 employees (2% 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 cross-checking of permission 12 which is assigned to 3 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 somehow 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 the example in Figure 5). 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 on the basis of the average employee numbers in the different departments. In order to cleanse the semantic errors and the potentially unresolved syntactic

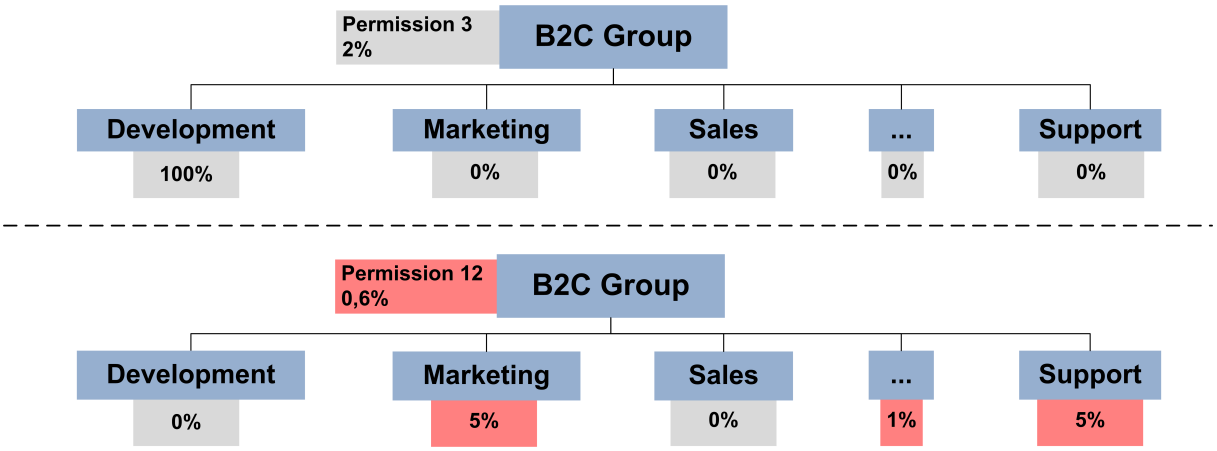

Figure 6: Permission outlier refinement

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. After the errors have been iteratively resolved and cleansed, the reconciliation flow to the target systems takes place. 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 [10], which also has significant benefit to hinder insider misuse.

4 Case Study

After the data cleansing mechanisms applied during the execution of contROLE have been presented, the paper continues with their evaluation in a naturalistic application scenario, using a large, complex, and potentially erroneous dataset.

4.1 Data gathering

The input data used (from hereinafter called Access Controls following the terminology of Molloy et al. [9]) originate from the Identity Management repository (Microsoft Active Directory) of a large industrial organization. The company, from hereinafter called SemiC, operates worldwide with about 30000 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 Listing 1).

Table 1 sums up the relevant statistics for the provided Access Controls. During the initial data import the *duplicate check* has been executed and duplicate datasets already have been excluded reducing the UPA (user-permission assignments) from 151062 to 150329.

```
Accountname;Location;Department;Permission
rorZ0cBq0rc5U;Singapore;SCAP OP SC MIT IN FAB;CN=SIN-OU-Users-G
roN1w2ZAbwwVg;Malacca;SCMY IT;CN=AP-SemiCEmployees-G
roN1w2ZAbwwVg;Malacca;SCMY IT;CN=MKZ-OU-Users-G
roECnrbpybF0w;unspecified;SCWU IT MFG;CN=AP-SemiCEmployees-G
roECnrbpybF0w;unspecified;SCWU IT MFG;CN=WUX-ITCoordinator-G
roECnrbpybF0w;unspecified;SCWU IT MFG;CN=WUX-FAB-Standard-Users-G
row6reFFa8y7o;Singapore;SCAP IT CBS HR;CN=SeC-Employees-SG-U
[...]
```

Listing 1: Access Controls extract SemiC

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

Table 1: Access Controls statistics SemiC

4.2 Syntactic data cleansing

Executing the *similarity checks* introduced earlier is not reasonable as the employee names are random strings and the organizational unit names are abbreviations. Only for the location attribute can suggestions according to the Levenshtein distance [8] 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 given scenario a list of valid organizational units for the line- and the geographic hierarchy has been provided. Thus, the departments included in the Access Controls have been compared with these values. 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 few years. Old organizational units might not have been de-provisioned correctly and thus still exist in the Active Directory environment, possibly resulting in insider abuse of information systems.

Besides the line organization, the consistency of the geographic structure of SemiC has been analyzed (see Figure 7). 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. contROLE proposes a correct value from the list of valid values (e.g. *Xi'an* instead of *Xian*; Levenshtein distance 1.0). Further analysis of this typing mistake reveals that ten employees are assigned to the misspelled location *Xian* while 300 employees are assigned to *Xi'an*. Thus, the erroneous location can automatically be renamed.

Effects of the syntactic data cleansing efforts

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

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 7: contROLE consistency check results

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 IdM 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.

4.3 Semantic data cleansing

Identify employee outliers

Employee outlier detection is carried out for the pre-cleansed dataset on the basis of a semi-automatic SOM analysis. ContROLE parses trained SOMs and stores potential outliers in a database table together with the node coordinates. Figure 8 presents the SOM visualisation of the geographic hierarchy of SemiC [3]. It can be seen that employees working in the same locations are in general clustered and 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 8). 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 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%, but less than

[3]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.

100% 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%. 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.

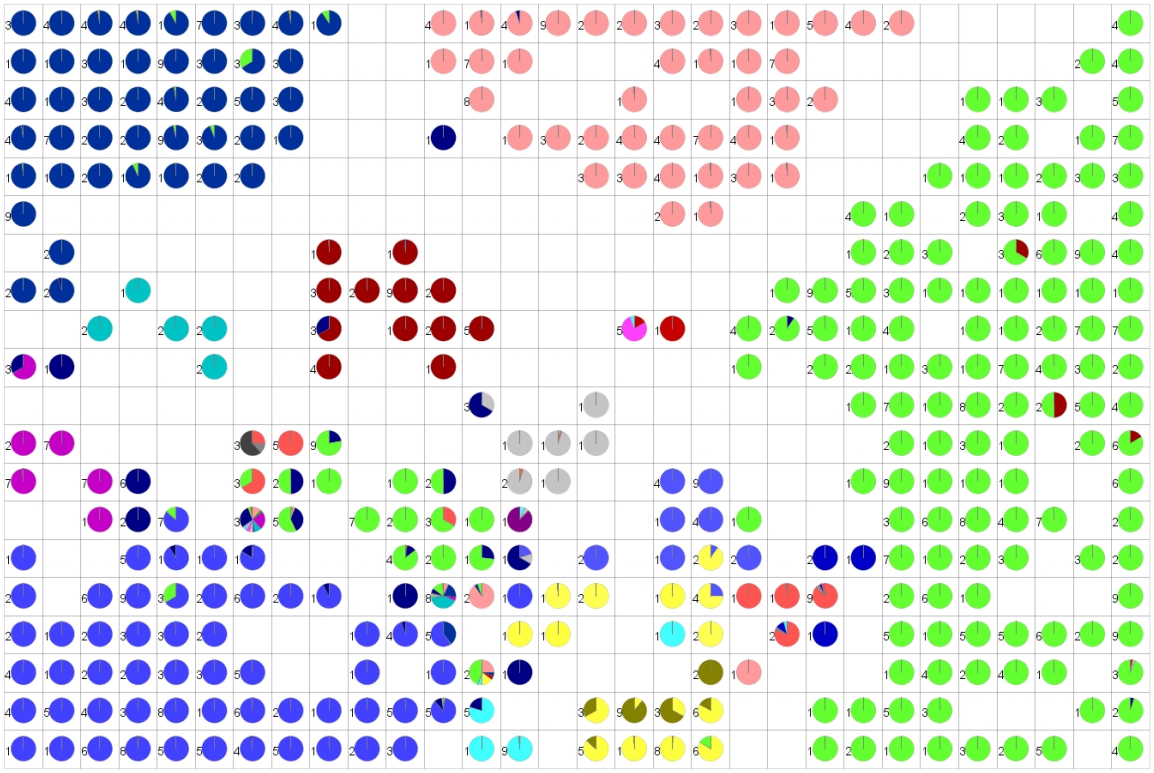

Figure 8: SOM analysis of the SemiC geographical hierarchy

Focusing on the upper left part of Figure 8, Figure 9 shows the effects of the aforementioned cleansing process. On the left side several outliers with the location attribute *Singapore* (light grey colouring) are depicted in the group of users working in *Kulim* (dark grey colouring). contROLE thus extracts these datasets and proposes *Kulim* 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. contROLE allows for the integration of business know-how for acquiring high-quality results.

The employee outlier detection described above resulted in a total of 340 attribute re-assignments for the remaining 7576 employees for the geographic hierarchy. The same process carried out for the line organization resulted in 153 attribute re-assignments to the proposed correct value.

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. contROLE suggests sending those potential outliers to the respective domain experts for approval.

The contROLE *rare permission check* is executed for the 16 provided level-1 line organization de-

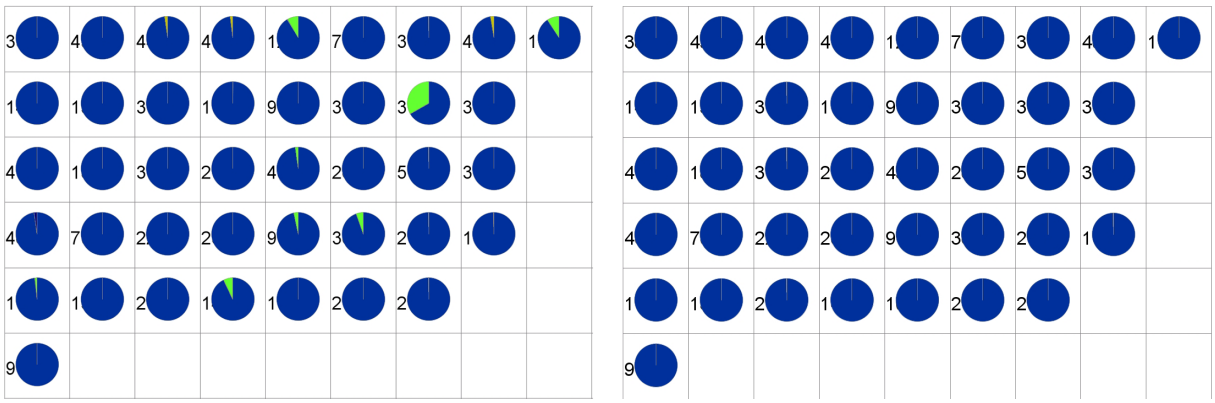

Figure 9: Employee outlier cleansing with contROLE (erroneous vs. cleansed data)

partments 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% 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%) 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% 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 10 depicts the percentage of the 20288 suspicious UPA considered erroneous after the refinement loop (blue coloring) in relation to the excluded UPA (grey 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).

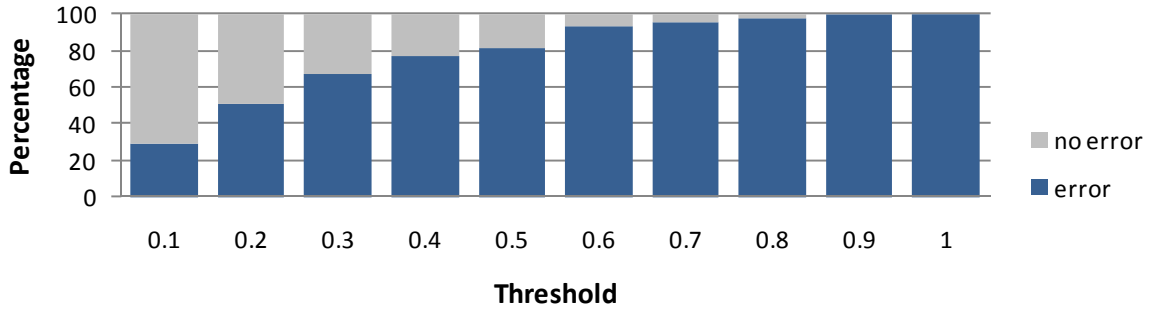

Figure 10: Rare permission check refinement results

4.4 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. Overall, an average reduction of the input data elements within the SemiC Access Controls of about 12.75% has been achieved. The result refinement in Table 2 shows that the number of permissions even could be reduced by 18.31%. 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.

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

Table 2: Data cleansing impacts on the SemiC identity data

5 Conclusion and Future Work

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. Over the time, access rights are mainly acquired and only rarely dispensed later even when they are no longer needed. As a consequence, many users possess more access privileges than are necessary to perform their actual job, permissions exist which are not used anymore, or accounts are still valid for which users already have left the organization. This situation is a risk to the threat of insider IT misuse because in most cases misuse takes place by insiders using their own user accounts and performing within the range of the totality of currently assigned privileges. Because detection methods mainly rely on rule-breaking behavior, this type of misuse 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.

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 a 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 and 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 a real-life application case.

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. In contrast to the currently existing contROLE data cleansing mechanisms, these checks investigate non-aggregated user and departmental data and examine the permission assignments within single organizational hierarchy elements.

5.1 Acknowledgments

The tool supporting the presented methodology uses on 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] L. Pernul G. Broser, C. Fuchs. Different Approaches to in-house Identity Management. In *Proc of the 4th Int. Conference on Availability, Reliability and Security (ARES)*, Fukuoka, Japan, 2009. IEEE Computer Society.

[2] D. Cappelli, A. Moore, T. Shimeall, and R. Trzeciak. Common Sense Guide to Prevention and Detection of Insider Threats. Technical report, Carnegie Mellon University Cylab, 2006.

[3] Federal Office for Information Security (BSI):. IT-Grundschutz. Available: http://www.bsi.bund.de/english/gshb/index.htm, 2004.

[4] Bank for International Settlements BIS. Basel II: International Convergence of Capital Measurement and Capital Standards: A Revised Framework - Comprehensive Version. Available: http://www.bis.org/publ/bcbs128.pdf, 2006.

[5] C. Fuchs, L. Mueller. Automating Periodic Role-Checks: A Tool-based Approach. In *Proc. Business Services: Konzepte, Technologien, Anwendungen. 9. Internationale Tagung Wirtschaftsinformatik.*, Vienna, Austria, 2009.

[6] G. Fuchs, L. Pernul. HyDRo - Hybrid Development of Roles. In *Proc. 4th Int. Conf. on Information Systems Security (ICISS), Hyderabad, India, LNCS 5352*, Berlin, Germany, 2008. Springer.

[7] Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69, 1982.

[8] V. I. Levenshtein. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Doklady Akademii Nauk SSSR 163*, 163(4):845–848, 1965.

[9] I. Molloy, H. Chen, T. LI, Q. Wang, N. LI, E. Bertino, S.B. Calo, and J. Lobo. Mining Roles with Semantic Meanings. In *Proc. of the 13th ACM Symposium on Access Control Models and Technologies (SACMAT)*, pages 21–30, Estes Park, CO, USA, 2008.

[10] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, and Charles E. Youman. Role-Based Access Control Models. *IEEE Computer*, 19(2):38–47, 1996.

[11] P. S. Sarbanes. Sarbanes-Oxley Act of 2002 (Pub. L. No. 107-204, 116 Stat. 745). Available: http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=107_cong_bills&docid=f:h3763enr.tst.pdf, 2002.

[12] European Union. Directive 95/46/EC of the European Parliament and of the Council. Official Journal of the European Communities of 23th November 1995 No L. 281 p. 31. Available: http://www.cdt.org/privacy/eudirective/EU_Directive_.html, 1995.

[13] L. Volino, G. H. Gessner, and G. F. Kermis. Sarbanes-Oxley Links IT to Corporate Compliance. In *Proc. of the 10th Americas Conference on Information Systems, New York*, 2004.

[14] Xingquan Zhu and Xindong Wu. Class Noise vs. Attribute Noise: A Quantitative Study of their Impacts. *Artificial Intelligence Review*, 22(3):177–210, 2004.