Alexandria Engineering Journal (2017) 56, 459–468

HOSTED BY

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

ORIGINAL ARTICLE

Role-task conditional-purpose policy model for privacy preserving data publishing

Rana Elgendy, Amr Morad *, Hicham G. Elmongui, Ayman Khalafallah, Mohamed S. Abougabal

Computer and Systems Engineering, Alexandria University, Alexandria, Egypt

Received 28 February 2017; accepted 27 May 2017
Available online 23 July 2017

KEYWORDS

Database security;
Access control;
Data publishing;
Anonymization

Abstract Privacy becomes a major concern for both consumers and enterprises; therefore many research efforts have been devoted to the development of privacy preserving technology. The challenge in data privacy is to share the data while assuring the protection of personal information. Data privacy includes assuring protection for both insider ad outsider threats even if the data is published. Access control can help to protect the data from outsider threats. Access control is defined as the process of mediating every request to resources and data maintained by a system and determining whether the request should be granted or denied. This can be enforced by a mechanism implementing regulations established by a security policy. In this paper, we present privacy preserving data publishing model based on integration of CPBAC, MD-TRBAC, PBFW, protection against database administrator technique inspired from oracle vault technique and benefits of anonymization technique to protect data when being published using k-anonymity. The proposed model meets the requirements of workflow and non-workflow system in enterprise environment. It is based on the characteristics of the conditional purposes, conditional roles, tasks, and policies. It guarantees the protection against insider threats such as database administrator. Finally it assures needed protection in case of publishing the data.

1. Introduction

Many enterprises would collect customers' data, such as personal information, financial or medical data in order to provide better service [1]. Since the occurrences of deceptive crimes and sensitive personal information disclosure happened frequently, privacy protection has been taken much attention by companies, consumers, and researchers [1]. Victims may receive annoying advertisements and reluctant marketing tricks in addition to face the threat of life and property [2].

Because of these threats, individuals are becoming frightened of sharing their businesses and transactions online, so organizations are losing large amount of potential profits. Therefore organizations pay attention to the management of private data [2].

* Corresponding author.
E-mail addresses: rana_elgendy@yahoo.com (R. Elgendy), amr.morad20@gmail.com (A. Morad), elmongui@alexu.edu.eg (H.G. Elmongui), ayman.khalafallah@alexu.edu.eg (A. Khalafallah), mohmed.abougabal@alexu.edu.eg (M.S. Abougabal).
Peer review under responsibility of Faculty of Engineering, Alexandria University.

http://dx.doi.org/10.1016/j.aej.2017.05.029

A major requirement of any information management system is to protect resources and data against unauthorized disclosure, called secrecy, and unauthorized or improper modifications, called integrity, while at the same time ensuring their availability to the users, means that no denial-of-service occurs. Enforcing such protection requires that every access to a system and its resources have to be under control and only authorized access requests are granted. This process is called access control [3].

Significant research efforts have been done toward achieving the perfect privacy preserving data publishing model. Many different types of database access control models have been developed to protect against outsider threats. Recent research also has been conducted on the privacy protection in the context of both workflow and non-workflow systems. A workflow system is defined as the orchestration of a set of activities involving coordinated execution of multiple tasks done by different processing entities [4]. Workflow systems guarantee the management of the flow of work such that the work is done by the proper person at the right time. This ensures a global integration between all the entities in the business process framework. Workflow systems also support resource allocation and dynamically adapt to workload changes [5].

In this paper, we provide a solution for privacy preservation against insider and outsider threats. This solution assures privacy in case the data are published. Our solution represents integration in some existing privacy preserving models; namely, (1) the CPRBAC access control model [6], (2) the MD-TRBAC access control model [7], (3) the PBFW access control model [8], some concepts from oracle vault technique [9], and (4) k-anonymity [10]. This integration would result in the benefits of these protection techniques. Therefore, the proposed privacy preserving data publishing model would inherit, from CPRBAC [6], the role-based access control and the task-based access control. It would also support workflow systems, as MD-TRBAC [7], and would have access control policies to enhance user privacy, as in PBFW [8]. It guarantees the protection against insider threats by adopting some concepts from oracle vault technique [9]. The model also would guarantee, from k-anonymity [10], that the individuals who are the subjects of the data cannot be re-identified while the data remain practically useful when the data are published. The model meets the particular requirement of the workflow systems such as the notion of a task life cycle, the dynamic access control, the separation of duty principle [11], and active permission assignment. In addition the new model adapts the notion of conditional purpose [6] which provides more reliable data management because more information can be extracted while assuring the user's privacy.

2. Related work

Several works have been done toward privacy protection technology. Enterprises have to develop a secure privacy protection model that ensures accessing the customers' data while at the same time assuring privacy for their sensitive data.

Role-based access control (RBAC) [12] has been widely used in database management systems and operating systems products because of its significant impact on access control systems. Following RBAC, Task-based access control (TBAC) [13] mainly focused on task-oriented perspective; therefore it approaches security modeling and enforcement at the application/enterprise level. A combination between RBAC and TBC, called Task Role-based access control (TRBAC) [14], which inherits the intuitionistic characteristic of RBAC model and the dynamic characteristic of TBAC model, is considered good step toward privacy protection access control models.

Purpose based access control (PBAC) [15] and conditional purpose based access control (CPBAC) [16] are considered a landmark toward privacy protection. The basic concept of both models is purposes. Purposes [15] describe the intentions for data collection and data access. Permissions are assigned on the combination of conditional roles and purposes. Role is defined as a job title or job function within the organization associated with its authority. Roles are organized in a role hierarchy to facilitate the administration tasks [15]. Purposes support both positive and negative privacy policies. In both models, purpose information associated with a given data element specifies the intended use of the data element. An access to a specific data item is allowed if the purposes allowed by privacy policies include or imply the purpose for accessing the data. An intended purpose consists of three components: Allowed Intended Purpose, Conditional Intended Purpose, and Prohibited Intended Purpose [15]. This structure provides greater flexibility to the access control model. Conditional purpose allows users to use some data for certain purpose with conditions [16]. More information from data providers can be extracted while at the same time assuring privacy. This maximizes the usability of consumers' data. The main drawback of the model is that it has a static permission assignment which means that the permission assignment process is not automated and does not change by the progression of a task. Permissions will in most cases manually be "turned on" too early or too late and will probably remain "on" long after the tasks have terminated. Another drawback is that there is no scope for the permission inheritance in the role hierarchy means that the parent role inherits total permissions from the child roles. This leads to vulnerabilities in the system, as the data may be misused [15,16].

Flexible Policy Based Access Control Model for Workflow Management Systems (PBFWs) [8] presented a great approach for enforcing privacy policy in workflow environments [8]. It has authorization policies to support dynamic separation of duty to prevent illegal data access [11]. The advantages of RBAC [12] and TBAC [13] are adopted; therefore PBFW meets the dynamic and flexible requirements, such as Separation of Duty policy (SoD), and dynamic access control that meets the workflow needs [11]. Separation of duty means that at least two different people are responsible for the completion of a task or set of related tasks [11]. The purpose of this principle is to discourage fraud by spreading the responsibility and authority for an action or task over multiple people, thereby raising the risk involved in committing a fraudulent act [11]. Also the model dynamically manages permissions as authorizations progress to completion [8]. The main drawback of the system is that it does not know the notion of the purposes and conditional purposes, which cause more information loss [8].

One of the remarkable contributions to the privacy protection is the Access Control Model Based on Multi-Role and Task (MD-TRBAC) [7]. This model addresses some distinct problems in the Conditional Purpose Dynamic Role-based

access control model (CPRBAC) [6]. MD-TRBAC introduced a new concept called permission inheritance scope. Permission inheritance scope is an action scope between the user and the assigned role [7]. MD-TRBC removes the role inheritance in the traditional RBAC [12] and classifies the roles and tasks according to the actual needs. Using the action scope ensures the system security, and reduces the complexity of the access policy. MD-TRBAC also uses dynamic permission management; this means that when the user performs operations, his permissions change dynamically. The major disadvantage about the model is that it leads to information loss because users are not allowed to get more information conditionally [7].

To achieve basic properties of privacy preserving, the model has to prevent the database administrator from accessing sensitive data if he is unauthorized to. One of the primitive methods of preventing database administrator is activity monitoring [17]. Real-time database activity monitoring can be done, either by analyzing protocol traffic (SQL) over the network, or by observing local database activity on each server using software agents, or both. Analysis can be performed to identify known exploits or policy breaches. This helps to build a normal pattern used for detection of anomalous activity that could be indicative of intrusion [17]. The main drawback for this is that it requires complex analysis for that database actions audit.

Another method for guaranteeing the protection against database administrator is encryption [18]. This encryption capability is designed into the application itself. Organizations will not have to add another solution for encrypting data across the network. By the time the database receives the data, it has already been encrypted and then stored in the database in this encrypted state. Communication from the client to the application needs an additional solution for encryption purposes. The major disadvantage of this scheme is that we should have significant changes in both the application layer and the database layer [18].

Oracle vault [9] can be considered one of most important techniques for the protection against database administrator unauthorized access. Oracle Database Vault restricts access to specific areas in an Oracle database from any user, including users who have administrative access. For example, administrative access to employee salaries, or other sensitive information can be restricted. The main oracle vault components are: realms, command roles, factors, rule sets and secure application roles. Oracle Database Vault protects against insider threats by using realms, factors, and command rules [9]. Combined, these provide powerful security tools to help secure access to databases and sensitive information. Rules and factors can be combined to control the conditions under which commands in the database are allowed to execute, and to control access to data protected by a realm. For example, rules and factors can be created to control access to data based on IP addresses or the time of day. This can prevent unauthorized access to the application data and access to the database by unauthorized applications [9].

The privacy preserving model needs to guarantee privacy in case the data are published to protect users' sensitive data such as medical or financial records. Among various approaches addressing this issue, the k-*anonymity* model [19] has recently drawn significant attention in the research community. In the k-anonymity model [10], privacy protection is achieved by ensuring that every record in a released dataset is indistinguishable from at least $(k-1)$ other records within the dataset. Thus, every respondent included in the dataset correspond to at least k records in a k-anonymous dataset, and the risk of record identification is guaranteed to be at most 1/k. This means that the released data guarantees that the individuals who are the subjects of the data cannot be re-identified while the data remain practically useful [19]. Many algorithms are used to implement k-anonymity such as Datafly algorithm [19], incognito algorithm [20] and Mondrian algorithm [21].

The aim of proposed model is to combine the advantage of the most powerful access control models to achieve perfect protection against inside and outside threats. The main concern is to preserve privacy of individuals as well as extracting more information and have an automated solution to meet the workflow environment requirements. The proposed model will combine the advantages of the CPBAC [6] model along with the advantages of the PBFW [8] model and MD-TRBAC [7] model. It will have a reliable data management by using the conditional access purpose concept in addition to achieving scalability and meeting the workflow environment requirement. The proposed model will also provide, inspired by Oracle vault [9] protection techniques, the needed protection against unauthorized access of the database administrator. It will guarantee the privacy of the sensitive data by integrating the advantages of the k-anonymity model [10] (see Table 1).

3. Proposed access control model

In this section, we present our novel model, which is an improved model that meets the particular requirements of workflow and non-workflow system in enterprise environment. It meets also requirements of protection against database administrator unauthorized access as well as needed protection in case of publishing the data. It is based on the characteristics of the conditional purposes, conditional roles, tasks, and policies. The database Entity Relationship Diagram (ERD) schema is shown in Fig. 1.

3.1. Features of the proposed privacy preserving model

The proposed model allows more information from data providers to be extracted while at the same time assuring privacy that maximizes the usability of consumers' data. The model also provides protection against database administrator (DBA) and necessary protection when the data is published. The proposed model has the following features.

- **The proposed model allows using data conditionally** to release certain information for certain purpose. This is done either by removing field (for example: name or id) or through generalization.
  This information is then stored in the database along with the collected data. Access to the data is tightly governed according to the data providers' requirements. Using the data conditionally, data providers feel more comfortable to release their data. It allows more information from data providers to be extracted while at the same time assuring privacy that maximizes the usability of customers' data.

Table 1 Proposed model vs. state of the art models.

Features	CPBAC	MD-TRBAC	PBFW	Proposed model
Task dependency	X	✓	✓	✓
Dynamic permission management	X	✓	✓	✓
Using data conditionally	✓	X	X	✓
Dynamic separation of duty	X	X	✓	✓
Scope inheritance	X	✓	X	✓
Database administrator protection	X	X	X	✓
Protection when publishing	X	X	X	✓

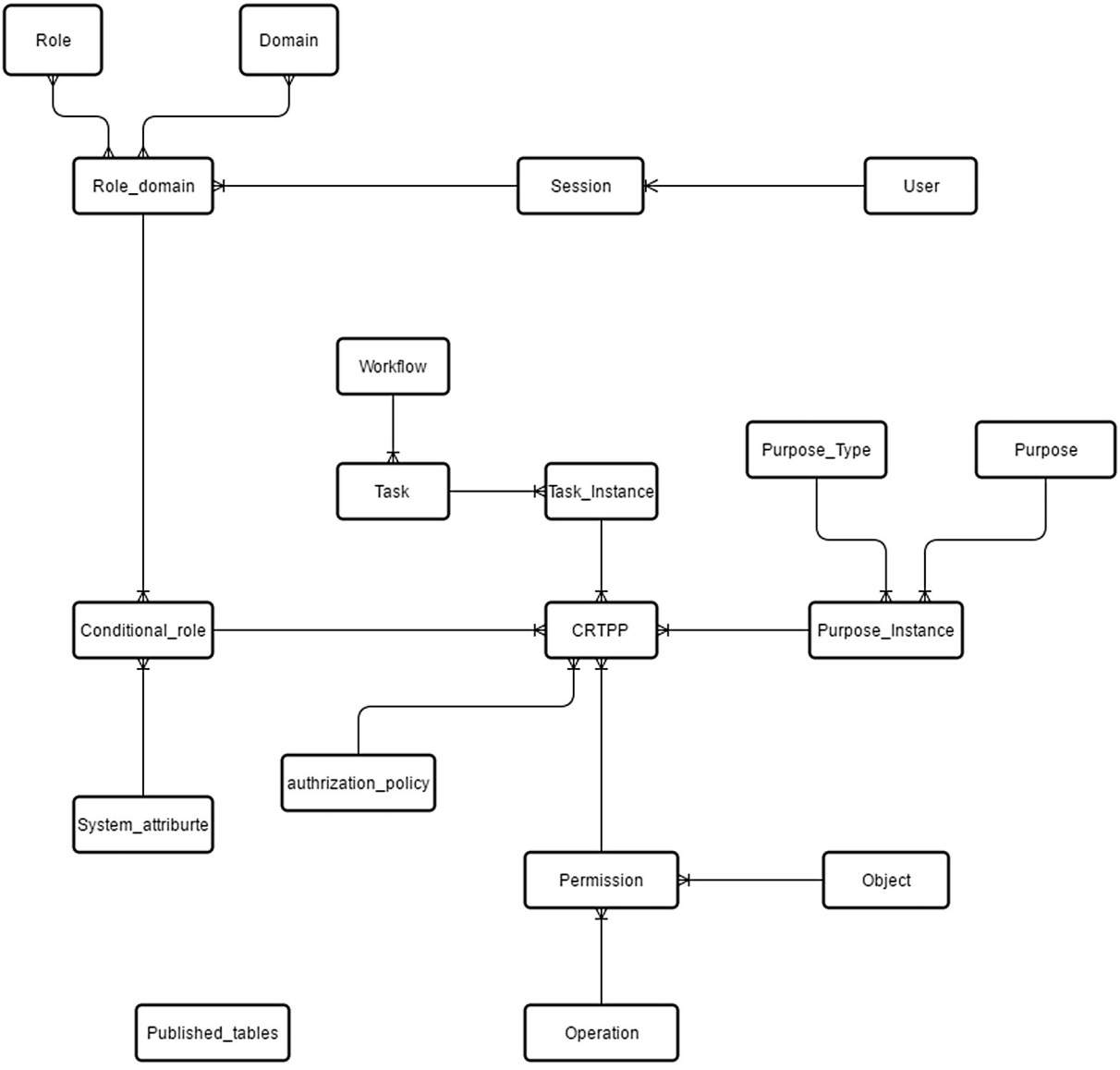

Figure 1 Proposed access control model ERD.

- **The proposed model supports workflow and non-workflow systems.** The proposed model has an active security model, which means that it has active runtime management of tasks progression to completion and permissions assigned to tasks.
- **The proposed model enables automated Permission assignment and revoking.** The proposed model should allow granting, usage tracking, and revoking of permissions automatically and coordinated with the progression of the tasks. Without active authorization management, permissions will in most cases be ''turned on'' too early or too late.

- **The proposed model supports domain inheritance not role hierarchy inheritance.** The proposed model removes role of inheritance in the traditional model by using domains, classifies the roles and tasks according to the actual needs, ensures the system security, and reduces the complexity of the access policy. Domain inheritance means that roles would inherit permissions from their domain, not from their role hierarchy.
  The proposed model uses Static and dynamic authorization. The proposed model uses policies in static authorization and dynamic authorization; also it applies the Separation of Duty principle. Both Static Separation of Duty (SSD) and Dynamic Separation of Duty (DSD) are enforced to prevent information being misused and prevent fraudulent activities [11].
- **The proposed model supports non-centralized management.** Non-centralized management assigns permissions to multiple managers to share permissions and simplify the work task of permission management. Despite of having one administrator responsible for permission assignment in the parent role, each manager has a separate management and must be familiar with the system and is qualified to judge who needs information.
- **The proposed model supports protection against database administrator.** The proposed model follows the techniques used in oracle vault [9] to prevent database administrator from accessing sensitive data. Oracle vault controls the access to specific tables, relations or views in the database from specific users even if these users have administrative access. Inspired by the oracle vault technique, the proposed model adds new domain for the administrator. A new role is created and associated to this newly created domain. This role will be assigned to administrative permissions only. Not all permissions will be assigned to this role. Only permissions needed for the admin to accomplish his tasks will be assigned to this domain. Moreover, Factors for the administrator can be added to determine the IP address which can access certain tables. Factors can help to restrict administrative permissions in specific machines, networks or places. This can help to prohibit fraud. Generally factors are named variable or attribute that can be recognized and secured such as IP address or session user [9]. Command roles can be added and be assigned to the appropriate domain for the database administrator if he is authorized to control how users can execute almost any SQL statement. Command role help to create new table or drop a table. These permissions are usually assigned to the administrator.
- **The proposed model supports protection when data are published.** Integrating the proposed model with k-anonymity [22] enables the data to be published safely. Users can now participate with their sensitive data without being afraid that once the data are published, their sensitive data such as medical records or financial data are not going to be exposed. The entities or tables which are supposed to be published are stored in order to run the appropriate k-anonymity algorithm. The algorithm will generalize Quasi-identifier attributes so the sensitive information can be published. Using Mondrian algorithm, all we need to do is to specify the Quasi-identifier attributes and then the algorithm will guarantee the data protection when the data are published [2,23].

3.2. Proposed privacy preserving model definitions

- **Privacy Preserving Data Publishing:** Privacy Preserving Data Publishing (PPDP) has become an area of interest for researchers and practitioners. The objective of PPDP techniques is to modify the data by making it less specific, in such a way that the individuals' privacy is protected. This aims to retain the usefulness of the anonymized data. The essence of PPDP is to produce datasets that have good utility for a variety of tasks because usually all the potential usage scenarios for the data are unknown at the time of publication [19].
- **Domain:** Domain allows users to have the system boundary access permissions and does not inherit all permission according to their assigned roles. It allows inheriting only permissions from roles inside their domains. Thus domain provides a flexible way to divide thousands of objects. The domain administrator can divide the domain according to function responsibilities or object type. The main problem that domain solves is decentralized authority management [14].
- **Domain inheritance:** Roles inside each domain have a hierarchy. Domain inheritance means that roles would inherit permissions from their domain, not from their role hierarchy [14].
- **Separation of duty:** Separation of Duty is a security principle used to formulate multi-person control policies. It requires that two or more different people be responsible for the completion of a task or set of related tasks. The purpose of this principle is to discourage fraud by spreading the responsibility for an action or task over multiple people. Thus raising the risk involved in committing a fraudulent act by requiring the involvement of more than one individual. Consider we need to implement the following tasks in a company; Task1 (Request promotion) and Task2 (Approve promotion). According to SoD [11], these two tasks could not be assigned to the same role and user [8].
- **Static and dynamic separation of duty:** Compliance with static separation requirements can be determined simply by the assignment of individuals to roles. The more difficult case is dynamic separation of duty where compliance with requirements can only be determined during system operation. The objective behind dynamic separation of duty is to allow more flexibility in operations [8].

3.3. Proposed privacy preserving data model

Fig. 1 shows the proposed data model satisfying the features mentioned in 3.1. In this model, roles are assigned to domains instances through "role_domain". Each "role_domain" is assigned to a conditional role. A workflow has some tasks, and each task instance is assigned to both conditional roles and purposes. Each purpose has a type. Access rights are associated with conditional roles, tasks instances and purposes instances through authorization policies. "Published_tables" are used to specify which tables will be published and to indicate the Quasi-identifiers for this table. Quasi-identifiers (QIDs) are set of attributes used to re-identify the record such as gender, data of birth or ZIP code.

The data model entities are:

- **Role:** Role is a job within organization associated with its responsibility. For example, in faculty organization we may have professor role, TA role and secretary role.
- **Session:** It is defined as the mapping between the user and the activated subset of the roles the user assigned to.
- **User:** User refers to object which can access the computer data and resource. It maybe person or application program.
- **Conditional_Role:** It refers to conditions on roles, like system conditions.
- **Domain:** Domains are system boundary access permission or roles-scope.
- **System Attribute:** It refers to system condition on roles. For example, roles may be activated only in a specific time interval, or activated for users only logged in from specific machines.
- **Task:** Tasks are activities or business processes.
- **Workflow:** It is a group of some business processes.
- **Task Instance:** It is a dynamic concept in workflow system. It is defined as an instance of operational task or task execution. Each task includes five statuses: static status, active status, suspended status, termination status, and failed status [7].
- **Object:** It refers to database objects like database tables, table columns, and table rows.
- **Operation:** there are many different database operations like query, add, delete, modify, and so on.
- **Purpose:** Purpose refers to the reason for accessing data. As mentioned before purpose can be divided into intended purpose and access purpose. Intended purpose, in turn, can be divided into allowed intended purpose or prohibited intended purpose.
- **Purpose Type:** It can be "allowed intended purpose", "prohibited intended purpose", or "conditional intended purpose".
- **Authorization Policies:** They are the restrictions by the system. For instance, approving write-offs roles should not be assigned to the same user.
- **CRTPP:** This represents a mapping between conditional role, permissions, authorization policy, purpose instance and task instance.
- **Purpose instance**: the combination of the purpose and its type.
- **Permission:** it will grant or deny one or more data in computer system by some way in the range of user access permissions.
- **Role_domain:** It is a combination between of the roles and the domains.
- **Published_tables:** This is used to list the tables, along with the Quasi-identifiers used in this table, that need to satisfy k-anonymity when the data is published.

The following table shows a comparison among the proposed model and state of the art models.

3.4. How proposed model achieves these features

SQL queries illustrated below are considered proof of concepts for the model. These can be optimized in multiple ways.

- Using data conditionally.

Given the database schema in Fig. 1, the following SQL query shows the implementation of this feature

```
Select distinct perm.id, perm.name from permission perm
Join object obj on (perm.object_id = obj.id)
Join operation oper on (oper.id = perm.operation_id)
Join crtpp on (perm.id = crtpp.permission_id)
Join conditional_role cr on (crtpp.conditional_role_id = cr.id)
Join system_attribute sattr on (sattr.system_attribute_id = cr.
system_attribute_id)
Join role_domain rd on (rd.id = cr.role_domain_id)
Join role role on (role.id = rd.role_id)
Join domain dom on (dom.id = rd.domain_id)
Join session sess on (sess.role_domain_id = rd.id)
Join purpose_instance pi on (pi.id = crtpp.purpose_instance_id)
Join purpose_type pt on (pt.id = pi.purpose_type_id)
Join purpose p on (p.id = pi.purpose_id)
Where (obj.id = object_id) and (oper.id = operation_id) and
(dom.id = domain_id) and (sattr.system_attribute_id =
sys_attribute_id) and (sess.user_id = user_id) and (pt.type_name
=purpose_type);
```

Where: object_id, purpose_id, operation_id, domain_id, sys_attribute_id, user_id, purpose_type are given.

This query filters on the type_name attribute of the purpose_type and on the given id attributes of the relations (object, purpose, operation, domain, system_attribute, user) and then projects on the id attribute of the permission relation to retrieve the distinct permissions of the given user. It would either return the permission of the user given these conditions or return empty set which means that this user is not authorized to do any operations on the data.

- **The proposed model supports workflow and non-workflow systems**. Given the database schema in Fig. 1, SQL query can be written similar to query given in conditional data usage section. The query filters on the given id attributes of the relations (object, task, operation, domain, system_attribute, user, workflow, task) and then projects on the id attribute of the permission relation to retrieve the distinct permissions of the given user. It would either return the permission of the user given these conditions or return empty set which means that this user is not authorized to do any operations on the data.
- **The proposed model enables automated Permission assignment and revoking**. As described in Section 3.1, the proposed model understands the notion of automated permissions. The model will guarantee automated handling for the permissions. The permissions will be turned on when the task is activated and turned off when the task is done.
- **The proposed model supports domain inheritance not role hierarchy inheritance**. Given the database schema in Fig. 1, SQL query can be written similar to query given in conditional data usage section. The query filters given domain attribute of the role and on the given id attributes of the relations (object, operation, domain, system_attribute, user) and then projects on the id attribute of the

permission relation to retrieve the distinct permissions of the given user. It would either return the permission of the user given these conditions or return empty set which means that this user is not authorized to do any operations on the data.

- **The proposed model uses Static and dynamic authorization**. Given the database schema in Fig. 1, SQL query can be written similar to query given in conditional data usage section. The query filters given the authorization policy and on the given id attributes of the relations (object, operation, domain, system_attribute, user, authorization_policy) and then projects on the id attribute of the permission relation to retrieve the distinct permissions of the given user. It would either return the permission of the user given these conditions or return empty set which means that this user is not authorized to do any operations on the data.
- **The proposed model supports non-centralized management.** As illustrated in Section 3.1, the proposed model allows the permissions to be shared across multiple managers. It is not necessary that the database administrator is the only one authorized to grant/revoke permissions.
- **The proposed model supports protection against database administrator.** The proposed model follows the techniques used in oracle vault [9] to prevent database administrator from accessing sensitive data. Inspired by the oracle vault technique, the proposed model adds "admin" domain to the "domains" table given in the database schema in Fig. 1. A new role called "admin" role is created and associated to this newly created domain. This role will be assigned to administrative permissions only. Not all permissions will be assigned to this role. Only permissions needed for the admin to accomplish his tasks will be assigned to this domain. In the proposed model, we will add new domain for administrator and a new role to the "roles" table. A mapping between the new domain in "domains" table and new role in "roles" table is saved in "domain_roles" table.
- **The proposed model supports protection when data are published.** Integrating the proposed model with k-anonymity [22] enables the data to be published safely. The proposed model achieves this by adding a new entity called "pub lished_tables" to the proposed model in Fig. 1. This indicates which table(s)/entity(ies) are going to be published and specifies which attributes are used to form Quasi-identifiers [19]. Mondrian algorithm [23] is used to achieve k-anonymity in this case.

4. Example

In this section, an example is presented toward the attempt of covering most of introduced concepts. A college system, illustrated in Figs. 2 and 3, will be implemented using the proposed model. The system will be divided into registration process and management process.

The registration process contains the following four tasks

- **Task1.1:** College staff provides course registration forms for students who only had met the prerequisites for the course.
- **Task1.2:** After the end of the course registration period, if the number of the registered students did not meet the minimum number required to open the course, then the staff has to make an announcement to drop the course. Otherwise; college staff sends a request to the Professor to confirm the completion of the course registration process.
- **Task1.3:** Professor confirms that the registration process is complete.
- **Task1.4:** College staff assigns a classroom for the course.

The management process contains the following tasks

- **Task2.1:** The database administrator adds/removes professors and college staff.
- **Task2.2:** Staff Manager modifies the salary for professors and for staff members. Database administrator does not have the authority to do such modifications.

The college system ERD is illustrated in Fig. 3. A student may register to multiple courses. A professor may teach multiple courses, and several professors may teach the same course. A college staff assigns a classroom for each course.

According to the proposed model, roles will be organized in domains. Each role will only inherit permissions form the domains it assigned to. Applying the pervious aspect leads us to the college roles hierarchy and domains (group of roles) hierarchy in Table 2. Domain names get the prefix "d", whereas role names get the suffix "r".

The responsibility of the different roles is described as:

- **rStaffA:** Responsible for (1) providing course registration forms for students, (2) sending requests to professors to confirm the completion of the registration process, or (3) sending a request to the Staff Manager to make an

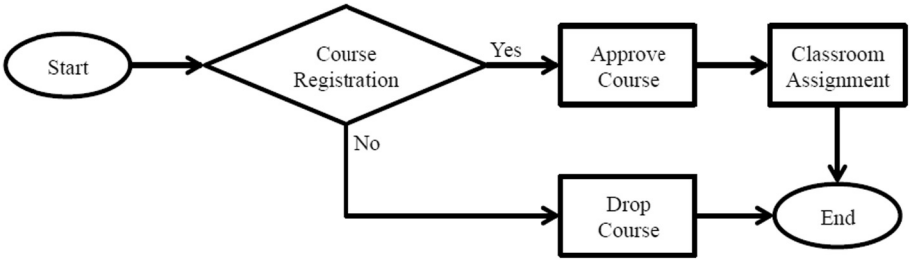

Figure 2 Course registration flow.

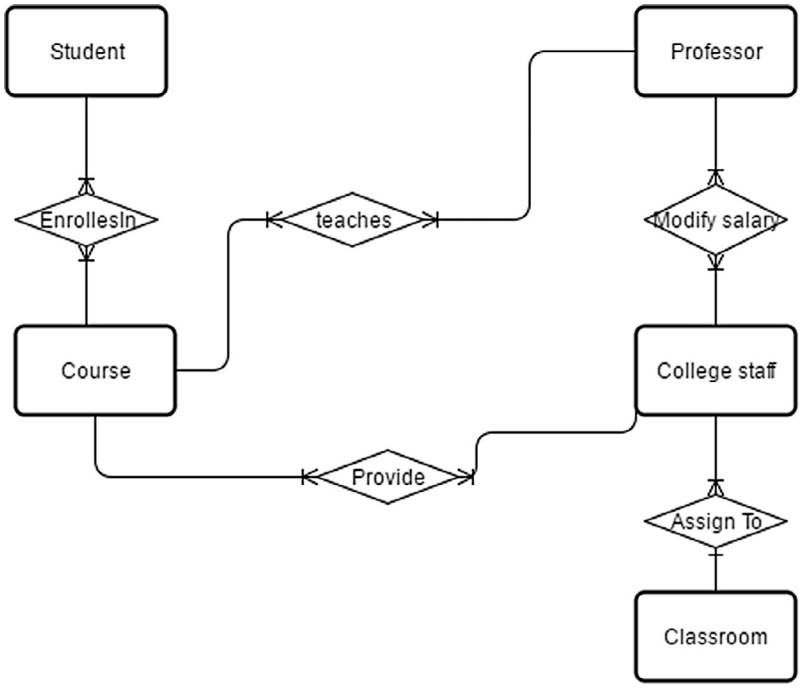

Figure 3 College system ERD.

Table 2 College domain hierarchy.

Domain	Assigned roles
dStaff	rStaffManager, rStaffA, rStaffB
dTeach	rProfessor
dCourse	rStaffA, rProfessor
dAdmin	rDatabaseAdmin
dFinancial	rStaffManager

Table 3 Domains and permissions assignment.

Domain	Permission assignment (database tables)
dStaff	Course, Classroom, *conditional purpose on* EnrollsIn
dTeach	Student, Course, Professor, Teaches
dCourse	Student, Course, EnrollsIn
dAdmin	*conditional purpose on* College_staff, *conditional purpose on* Professor
dFinancial	*conditional purpose on* College_staff, *conditional purpose on* Professor

announcement to cancel the course if the number of registered students is less than the minimum required to open the course.

- **rStaffB:** Responsible for classroom assignments.
- **rStaffManager:** Responsible for making an announcement to cancel the course because the number of registered student did not meet the minimum number required to open the course. He is also responsible for modifying the salary of the professor and staff members.
- **rProfessor:** Manages teaching, writing tests and other teaching assignments.
- **rDatabaseAdmin:** Adds and/or deletes college staff members and professors.

The permissions given to each domain are shown in Table 3. As a result of the domain and roles hierarchy, each role will be assigned to some domains and will gain only needed permissions as shown in Table 4.

According to the registration process requirements and the proposed model, Task1.4 (Classroom assignment) will not be active until Task1.3 (Professor confirms the registration process) is terminated; applying the dynamic permission assignment concept. Moreover, in the traditional RBAC model, the higher role inherits total permissions from the lower role. But in the proposed model, roles only inherit the domain permissions it assigned to. Applying role inheritance scope, rStaffManager role is assigned to domain dStaff and dFinancial so it will only inherit dStaff and dFinancial permissions. Similarly, in the traditional RBAC [12]; rStaffManager will inherit all its descendant permissions, and accordingly rStaffManager will inherit dTeach permissions and dStaff permissions, which may lead to information misuse. In traditional RBAC, roles are organized in a form of tree (Role hierarchy). rStaffManager role is senior to rStaffA and rStaffB roles. This means that rStaffManager role inherits permissions of both rStaffA and rStaffB roles' permissions. Accordingly rStaffManager will gain access to tables that it is not supposed to have such as Teaches, professor, Student and full access to EnrollsIn. This may lead to information misuse.

In Task1.4 (Classroom assignment), rStaffB has a conditional access purpose on that task, which is to get the count of the registered students to locate a suitable classroom, rStaffB role does not need a full access on the student database table, all it need is the count of the students. For example, If rStaffB wants to modify EnrollsIn table, then he will specify access purpose of modification. As shown in Fig. 1 (Proposed Access Control Model ERD), authorization access entity will be passed this access purpose along with this role. According

Table 4 Roles and permissions assignment.

Role	Assigned domain	Permission assignment (database tables)
rStaffA	dStaff, dTeach	Course, Classroom, Student, Professor, Teaches, EnrollsIn, *conditional purpose on* EnrollsIn
rStaffB	dStaff	Course, Classroom, *conditional purpose on* EnrollsIn
rStaffManager	dStaff, dFinancial	Course, Classroom, *conditional purpose on* professor, *conditional purpose on* college_staff, *conditional purpose on* EnrollsIn
rProfessor	dTeach, dCourse	Course, Professor, Student, EnrollsIn, Teaches
rDatabaseAdmin	dAdmin	*conditional purpose on* college_staff, *conditional purpose on* professor

to the authorization policy [11], this type of purpose will be rejected for this user as he does not have the sufficient privileges to do so.

Staff Manager is the only one allowed to modify the salary of the professor and college staff. As the database administrator is assigned to the admin domain and this domain does not have the permission of viewing the salaries, the admin will be able neither to view nor modify the salary of the professor or college staff. This way the model protects against database administrator.

In Task 2.1, the database administrator needs to add or remove professors and college staff. He should not be able to access the sensitive data for professors or college staff such as financial data. The admin is assigned to dAdmin domain which gives him conditional purpose access on professors and college_staff tables. This conditional access will allow him either to add or remove records without being able to expose their sensitive data.

For Task 2.2, the staff manager will have also conditional purpose access for both professors and college_staff to modify the salaries for them. Only staff manager will be able to access this sensitive data and as shown even the database administrator will not be able to expose this data.

Finally we need to make sure that the published data will keep the protection specially we have here sensitive (financial data) data for professors and college staff, "professor" and "college_staff" will be added along with the Quasi-identifiers for each of them (for example: IDs and date of birth) in "pub lished_tables" entity in the proposed model schema given in Fig. 1. The sensitive attribute that we have to protect in these tables is the "salary". When the publish request is fired for the professors and/or college staff data, the Mondrian algorithm will be applied to satisfy k-anonymity. It will apply the Mondrian algorithm [21] to make sure that each released data record will be indistinguishable from other k records, therefore satisfying the k-anonymity model [22] and protecting the released data from being re-identified.

4.1. Discussion

After the simulation of the implementation of the registration process above on the proposed model, the following aspects and features are discussed.

- The proposed model applies the dynamic permission assignment concept; the permissions would be authorized to the user when he needs them not too early or too late.
- Role inheritance scope is applied by the proposed model. Roles will only inherit permissions needed to complete the tasks, and will not inherit its descendants permissions as applied in the RBAC model.
- The proposed model makes a full use of the data without violating the privacy by applying the notion of the conditional purposes. Conditional purposes provide extracting more information from the data while at the same time assuring privacy that maximizes the usability of consumers' data.
- After each task completion, its permissions will be revoked automatically, which provides more security to the system.
- The model protects against database administrator unauthorized access.
- When the data are published, the tables and corresponding Quasi-identifier attributes loaded in "Published_tables" table in the model ERD schema, Fig. 1, will be used along with Mondrian algorithm to make sure that the released data will satisfy the k-anonymity [21] properties.

5. Conclusion

In this paper we presented Role-Task Conditional Purpose Policy based protection model for privacy preserving data publishing. As shown throughout the paper, this model will combine advantages of workflow and on-workflow systems along with characteristics of the conditional purposes, conditional roles, tasks, and policies. The model will guarantee privacy preserving data publishing meaning that it will meet: (1) protection against database administrator unauthorized access to data, and (2) published data will be secured meaning that each released data record will be indistinguishable from k other records. The estimated cost for the needed queries will be affordable and such queries can be cached so this model will guarantee privacy preserving in relatively low cost.

Acknowledgments

The authors acknowledge the very useful comments of the reviewers.

References

[1] B.S. Babu, N. Jayashree, P. Venkataram, Performance analysis of Steiner tree-based decentralization mechanism (STDM) for privacy protection in wireless sensor networks, Int. J. Network Secur. 15 (5) (2013) 321–330.
[2] W.S. Juang, J.L. Wu, Efficient user authentication and key agreement with user privacy protection, Int. J. Network Secur. 7 (1) (2008) 120–129.
[3] Pierangela Samarati1, Sabrina De Capitani di Vimercati, Access Control:Policies, Models, and Mechanisms, lecture notes In Computer Science, vol. 2171, 2000, pp. 137–196.
[4] Workflow Management Coalition Specification. Workflow Security Considerations. Technical Report WFMC-TC-1019, The Workflow Management Coalition, 1998.
[5] Workflow Management Coalition Specification. Terminology & Glossary. Technical Report WFMC-TC-1011, The Workflow Management Coalition, 1999.

[6] Md. Enamul Kabir, Hua Wang, Elisa Bertino, A conditional purpose-based access control model with dynamic roles, Expert Syst. Appl. 38 (3) (2011) 1482–1489.
[7] Jing-Mei Li, Bin Wang, Nan Ding, and Shengnan Jin. Access Control Model Based on Multi-Role and Task. In Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011 2nd International Conference on, pages 2756–2759, Aug 2011.
[8] Gang Ma, Kehe Wu, Tong Zhang, and Wei Li. A Flexible Policy-Based Access Control Model for Workflow Management Systems, in: Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on, vol. 2, pp. 533–537, June 2011.
[9] Introducing oracle database vault, in: Oracle documentation. [Online]. Available: < https://docs.oracle.com/cloud/latest/db121/DVADM/dvintro.htm > (last visited: September 14, 2016).
[10] Latanya Sweeney, Achieving k-anonymity privacy protection using generalization and suppression, Int. J. Uncertain., Fuzzin. Knowl.-Based Syst. 10 (05) (2002) 571–588.
[11] Richard Simon and Mary Ellen Zurko. Separation of Duty in Role-based Environments. In Proceedings of the 10th IEEE Workshop on Computer Security Foundations, CSFW '97, pages 183–. IEEE Computer Society, 1997.
[12] R.S. Sandhu, E.J. Coyne, H.L. Feinstein, C.E. Youman, Role-based access control models, IEEE Comput. 29 (2) (1996) 38–47.
[13] R.K. Thomas, R.S. Sandhu, Task based Authorization Controls (TBAC): A Family of Models for Active and Enterprise oriented Authorization Management, IFIP International Federation for Information Processing, 1997.
[14] Xiangning Zhou, Zhaolong Wang, An Access Control Model of Workflow System Integrating RBAC and TBAC, IFIP International Federation for Information Processing, October 2007, pp. 246–251.
[15] Ji Won Byun, Elisa Bertino and Ninghui Li, Purpose-based Access Control for Privacy Protection in Relational Database Systems, Technical Report 2004–52, Purdue University, 2004.
[16] Md. Enamul Kabir, Hua Wang, Conditional Purpose Based Access Control Model for Privacy Protection. In Proceedings of the Twentieth Australasian Conference on Australasian Database - Volume 92, ADC'09, pages 135–142. Australian Computer Society Inc, 2009.
[17] Seema Kedar, Database Management System, 1st ed., Technical Publications Pune, 2009.
[18] Tanya Baccam, Transparent Data Encryption: New Technologies and Best Practices for Database Encryption, SANS institute, April 2010.
[19] Latanya Sweeney, K-anonymity: a model for protecting privacy, Int. J. Uncertain., Fuzzin. Knowledge-Based Syst. 10 (05) (2002) 557–570.
[20] K. LeFevre, D.J. DeWitt, R. Ramakrishnan, Incognito: Efficient Full-domain K-Anonymity, In Proceedings of the 2005ACMSIGMOD International Conference on Management of Data, SIGMOD'05, pp. 49–60, 2005.
[21] LeFevre Kristen, David J. DeWitt, Raghu Ramakrishnan, Mondrian multidimensional k-anonymity, 22nd International Conference on Data Engineering (ICDE'06). IEEE, 2006.
[22] L. Sweeney, Achieving k-anonymity privacy protection using generalization and suppression. Int. J. Uncertain., Fuzzin. Knowl.-based Syst., vol. 10(5), 2002.
[23] Ayala-Rivera, Vanessa, et al. A Systematic Comparison and Evaluation of k-Anonymization Algorithms for Practitioners, Transactions on Data Privacy 7.3 (2014) 337–370.