Chapter 2 Literature Survey

2.1 Introduction

The previous chapter introduced the motivation of the work, a brief description of the related work, and the need to extend the existing models. This chapter is a detailed literature survey about the related database access control models, explaining reasons of selecting these models for developing the proposed model along with this thesis. In section 2.2, a detailed survey about Role Based Access Control model (RBAC) [11] is discussed, followed by Task Based Access Control model (TBAC) in section 2.3. An integrated model Task Role based access control model is illustrated in section 2.4. Another contribution to the RBAC model is the Purpose based access control model (PBAC) presented in section 2.5. In section 2.6, CPBAC model [14] is explained. CPBAC is based on the mature and widely used RBAC model [11]. In section 2.7 a detailed survey about MD-TRBAC model [15] is established followed by a survey about PBFW model [16]. The need to extend the current work is elaborated in section 2.9. The scope of the work is outlined in section 2.10. Finally the chapter is concluded in section 2.11.

2.2 Role based access control model (RBAC)

2.2.1. Introduction

There was a need for regulating access to resources based on users' roles within an enterprise. It is preliminary design to satisfy the need of simplifying the authorization management and directly presenting access control policies. In RBAC system administrators create roles according to the organization and the business needs. Each role has some privileges to regulate access to computers or specific resource. Roles may represent a job description or specific user task. Roles are assigned to permissions and then users are assigned to roles on the basis of their specific job description and qualifications. Users can be easily assigned and reassigned from roles as need change rather than assigning users to permission directly as shown in Figure 2-1 and Figure 2-2 [11] [20].

Figure 2-1 Role-User-Permission relationship

One example of the RBAC system implementation as in college system, Roles can be represented as Teacher role, staff role, and student role. Each user to the system is assigned to one of these roles. Each role is assigned some permissions; Teacher role has access to students' records, and Staff role has access to classroom assignments.

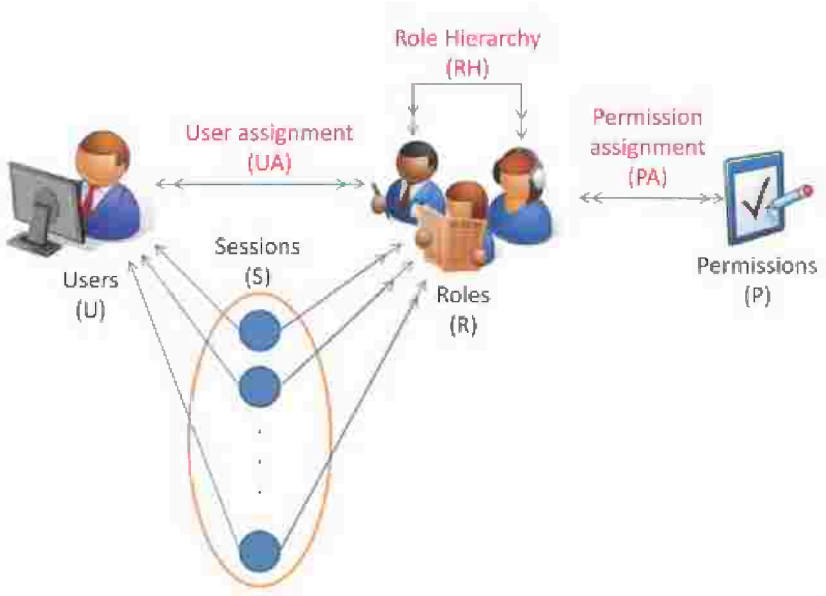

Figure 2-2 Role based access control model permission assignment

2.2.2. RBAC Model Definitions and Principles

Definition 2.2.2.1. (Role).

It is a job function within the organization that describes the users' authorities and responsibilities in the organization or the company.

Definition 2.2.2.2. (Resource).

Resource may be database object, computers, or a specific process.

Definition 2.2.2.3. (Administrative role).

A role that includes permission to modify a set of roles, users, or to modify the user assigned permissions.

Principle2.2.2.1. (Least privilege) [11].

Only those permissions required for the tasks performed by the user, in the organization, are assigned to the role.

Principle2.2.2.2. (Separation of duty) [11].

Invocation of mutually exclusive roles can be required to complete a sensitive task, such as requiring an accounting clerk and an account manager to participate in issuing a check.

The key issue is that the RBAC model cannot enforce the way these principle are applied.

2.2.3. RBAC Database Model

The following Figure 2-3 shows the RBAC database model followed by the description of its core elements.

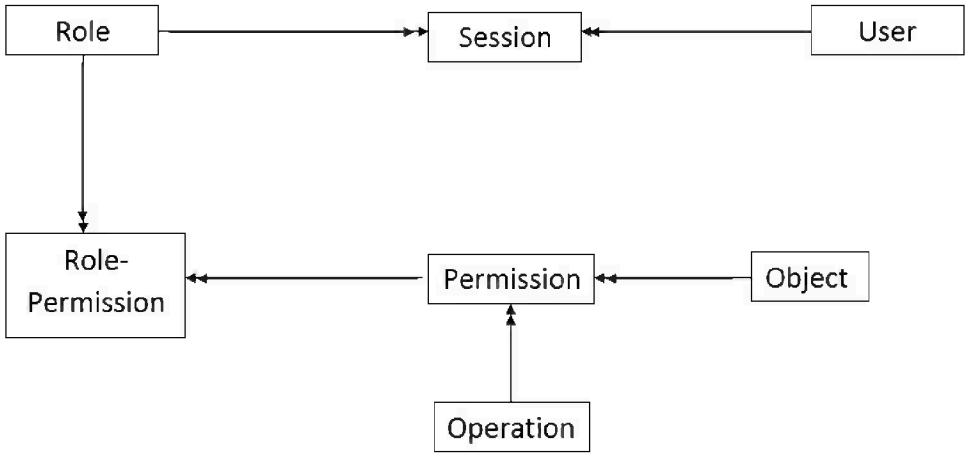

Figure 2-3 RBAC Database Model

Role: is a job function or job title within the organization associated with its authority and responsibility.

User: any person who interacts directly with a computer system.

Session: it is a mapping between a user and its active assigned roles.

Role-Permission: is the relation between roles, and permissions.

Object: database object.

Operation: there are many operations, for example: query, add, delete, modify, and so on.

Permission: it is granting or revoking access to a specific resource or computer.

2.3 Task based access control model (TBAC)

2.3.1. Introduction

TBAC model access controls from a task-oriented perspective than the traditional subject-object one. TBAC differs from traditional access controls and security models in many respects. Instead of having a system-centric view of security, TBAC approaches security modeling and enforcement at the application and enterprise level. Secondly TBAC lays the foundation for "active" security models. By active security models, there is active runtime management of security as tasks progress to completion. Permissions are constantly monitored and activated and deactivated in accordance with emerging context associated with progressing tasks (such as in workflows) as shown in Figure 2-4 [18].

One example to TBAC as in College system. Each user (User1) has a task (Update students 'grades) to do. (User2) has the same (User1) task (Update students' data). (User3) has a different task (Add new students to the system).

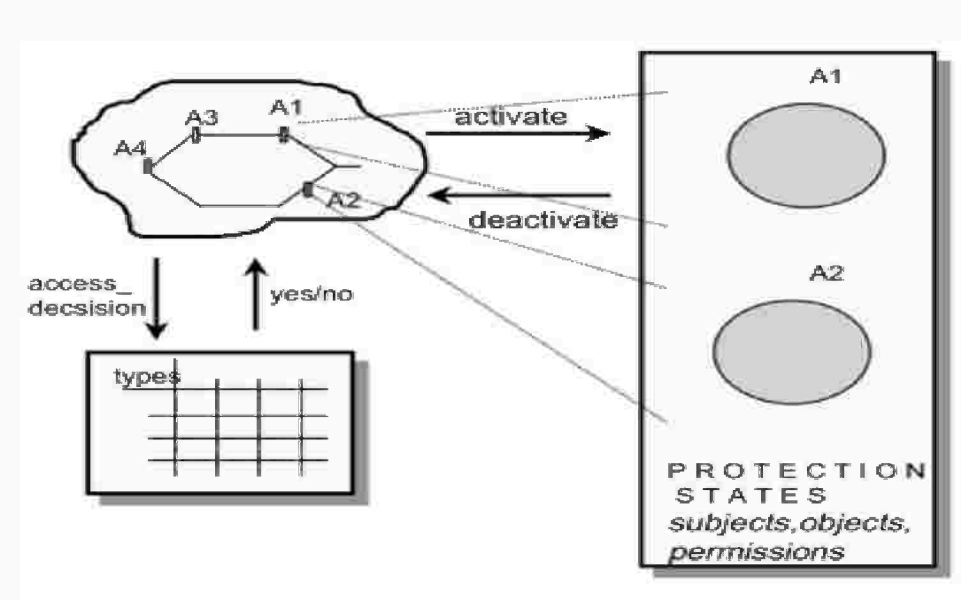

Figure 2-4 Task based access control model

2.3.2. TBAC Model Definitions

Definition 2.3.2.1. (Workflow) [21, 22, 23].

A kind of business process, which can be fully or partially automated, and it can also be transferred and implemented among executors based on a series of process, rules, documents, information or tasks. Workflow management systems make it possible that a workflow could be analyzed, simulated, designed, enact, controlled and monitored.

Definition 2.3.2.2. (Task) [18].

Task is the logical unit in the workflow. It do functional operations based on the own permission. It can also be the activities of the workflow or the composition of several activities.

Definition 2.3.2.3. (Dynamic permission management) [18].

Permissions are constantly monitored and activated and deactivated in accordance with emerging context associated with progressing tasks (such as in workflows). Permission has a lifetime associated with it during, which it is considered valid. Granting, usage tracking, and revoking of permissions are automated and coordinated with the progression of the various tasks. Without active authorization management, permissions will in most cases be "turned on" too early or too late and will probably remain "on" long after the workflow tasks have terminated. This opens up vulnerabilities in systems.

2.3.3. TBAC Database Model

The following Figure 2-5 shows the TBAC database model followed by the description of its core elements.

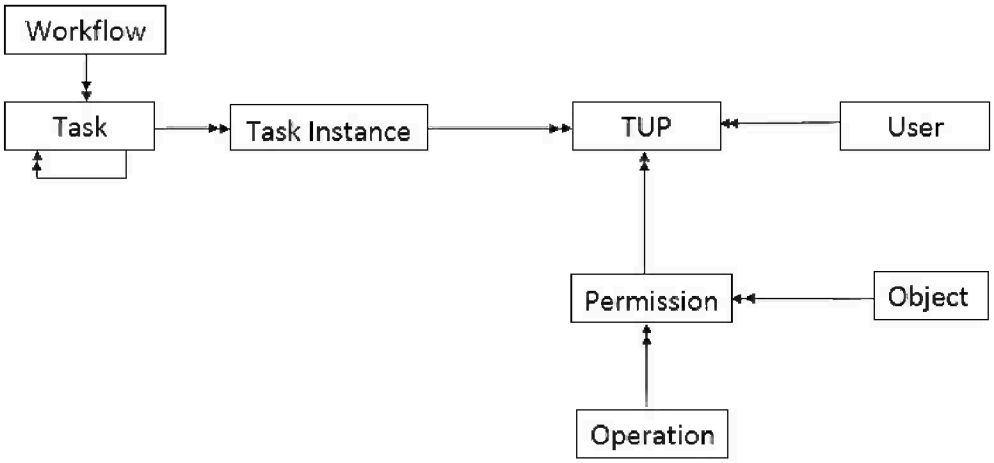

Figure 2-5 TBAC Database Model

Task: activities in the system.

Workflow: group of some tasks.

Tasks-Instance: the task instance is a dynamic concept in workflow system and also is an instance of operational task and task execution. Each task includes five status: static status, active status, suspended status, termination status, and failed status.

TUP: is the relation between users, tasks, and permissions.

Object: database object.

Operation: there are many operations, for example: query, add, delete, modify, and so on.

Permission: it will grant one or more data in computer system by someway in the range of user access permissions.

2.4 Task Role based access control model (TRBAC)

2.4.1. Introduction

TRBAC model has strong descriptive capability. It inherits the intuitionistic characteristic of RBAC model and the dynamic characteristic of TBAC model. As in some systems some abstracting ways of RBAC are needed to divide and describe some activities connected with access control in the system. At the same time, some descriptive ways of TBAC are also needed to describe the dynamic characteristics in the system.

As an implementation example in a college systems. Roles can be represented as Teacher role, staff role, and student role. Tasks can be represented by the jobs each role does. For instance Task1: update student grades, Task2: classroom assignment. Each task is given the right permission to access the resources, like database tables, according to its description. Then each task in the system is assigned to one of these roles.

2.4.2. TRBAC Model Definitions

Definition 2.4.2.1. (Workflow) [21, 22].

A kind of business process, which can be fully or partially automated, and it can also be transferred and implemented among executors based on a series of process, rules, documents, information or tasks.

Definition 2.4.2.2. (Task) [18].

Task is the logical unit in the workflow. It do functional operations based on the own permission. It can also be the activities of the workflow or the composition of several activities.

Principle2.4.2.1. (Least privilege) [11].

Only those permission required for the tasks performed by the user in the organization are assigned to the role.

2.4.3. TRBAC Database Model

The following Figure 2-6 shows the TRBAC database model followed by the description of its core elements.

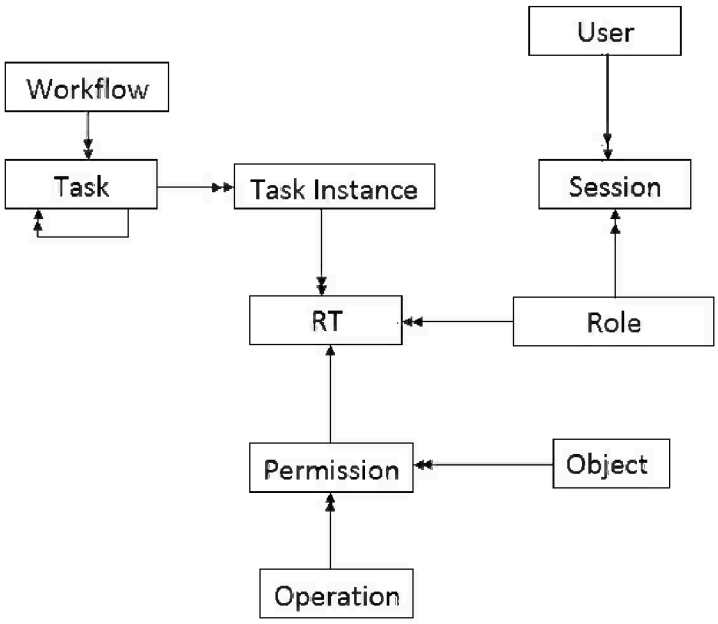

Figure 2-6 TRBAC Database Model

Role: is a job function or job title within the organization associated with its authority and responsibility.

User: it can independently access the computer data and resource and maybe person or application program.

Session: is the mapping between the user and the activated subset of the roles the user assigned to.

Task: activities in the system.

Tasks-Instance: task instance is a dynamic concept in the workflow system and also in an instance of operational task and task execution. Each task includes five status: static status, active status, suspended status, termination status, and failed status. The task instance state migration is shown in figure 2.7 [15].

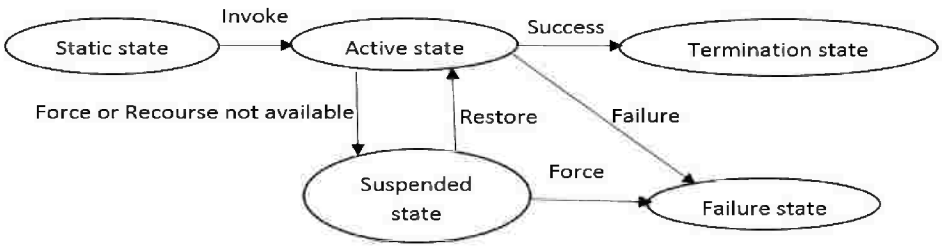

Figure 2-7 Task instance state migration

Workflow: group of some business processes.

RT: is the relation between roles, tasks, and permissions.

Object: database object.

Operation: there are many operations, for example: query, add, delete, modify, and so on.

Permission: it will grant one or more data in computer system by someway in the range of user access permissions.

2.5 Purpose based access control model (PBAC)

2.5.1. Introduction

As privacy becomes a major concern for both consumers and enterprises, many research efforts have been devoted to the development of privacy protecting technology. In this model 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 the privacy policies for the data include or imply the purpose for accessing the data. The purposes associated with data and thus regulating data accesses are Intended Purposes, and the purposes for accessing data are Access Purposes. Intended purposes can be viewed as brief summaries of privacy policies for data, stating for which purposes data can be accessed. When an access to data is requested, the access purpose is checked against the intended purposes for the data. In this model intended purposes support both positive and negative privacy policies. An intended purpose consists of two components: Allowed Intended Purposes and Prohibited Intended Purposes. This structure provides greater flexibility to the access control model [20].

In order to illustrate the model, suppose that we wish to allow some users in the Staff role to access data for the purpose of "Service-Update". Under the "Staff" role, there are two descendant roles: "Analysts Staff" and "Writers Staff". "Analysts Staff" are the users who analyze the student advertisements and prepare the contents of emails, and "Writers Staff" are the users who write and send out emails to students. These two roles are defined based on the access permission assignments in that the permissions to access the student profiles are exclusively assigned to the "Analysts Staff" role while the permissions to access the email addresses of the students are exclusively assigned to the "Writers Staff" role. However, as we want to assign the access purpose only to the users who are responsible for the specific task of sending out updated service information, neither the definition of "Analysts Staff" or "Writers Staff" matches our intention. Moreover, assigning the access purpose to the "Staff" role is not desirable as it will allow all the users with the "Staff" role to access data with the access purpose. An alternative solution is to split the "Staff" role or the "Analysts Role" and the "Writers" roles into more specific roles. However, this method requires the reconstruction of both the user assignments and the permission assignments for the modified roles. Authorizing the access purpose on an individual basis is not an elegant solution either, as it does not utilize the existing role hierarchy and thus is not scalable. In order to address these issues, a new concept is introduced, which is the Purpose concept.

2.5.2. PBAC Model Definitions

Definition 2.5.2.1. (Purpose and Purpose Tree) [17].

A purpose describes the intentions for data collection and data access. A set of purposes, denoted as ω, is organized in a tree structure, referred to as Purpose Tree and denoted as Ω, where each node represents a purpose in ω and each edge represents a hierarchical relation between two purposes. Let r_i ,r_j , be two purposes in Ω. We say that r_i is an ancestor of r_j (or r_j is a descendent of r_i) if there exists a downward path from r_i to r_j in Ω.

Figure 2.8 is an example of purpose tree, where each node represents a purpose in ω and each edge represents a hierarchical relation between two purposes.

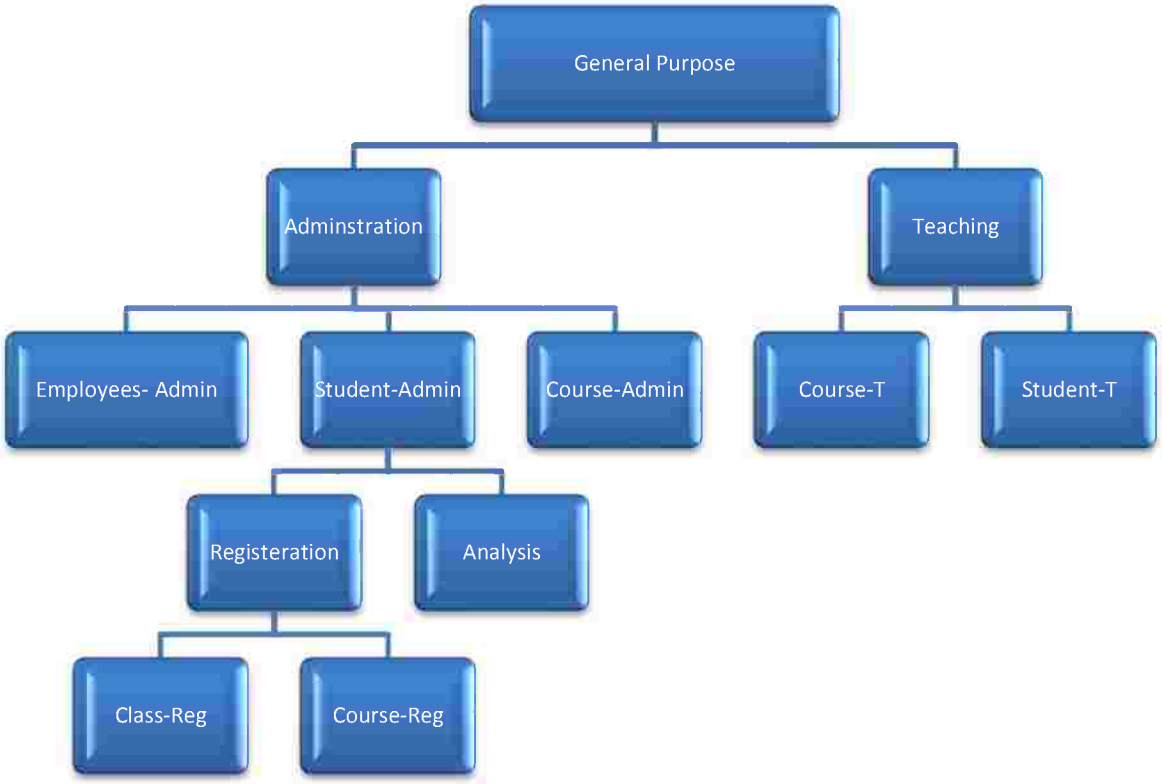

Figure 2-8 Purpose tree

Definition 2.5.2.2. (Access Purpose).

An access purpose is the reason for accessing data objects, and it must be determined by system when data access is requested. So access purpose specifies the purpose for which a given data element is accessed.

Definition 2.5.2.3. (Intended Purpose).

Intended purpose is the purpose associated with the data and regulating data access. It is divided into two types: Allowed Intended Purposes (AIP), and Prohibited Intended Purpose (PIP).

Allowed Intended Purpose allows the data to be used without any restrictions for certain purposes. Prohibited Intended Purpose strictly disallowing accessing the data under certain purpose.

Definition 2.5.2.4. (Access Purpose Compliance).

Let Ω be a purpose tree. Let IP = {AIP, PIP} and AP be an intended purpose and an access purpose defined over Ω, respectively. AP is said to be compliant with IP according to Ω, denoted as $AP \rightarrow_{\Omega} IP$, if and only if the following two conditions are satisfied.

1. $AP \notin PIP^{\updownarrow}$.
2. $AP \in PIP^{\downarrow}$.

Where $PIP^{\updownarrow}$: is the set of all nodes that are either ancestors or descendants of nodes in the purpose tree, and $PIP^{\downarrow}$: is the set of all nodes that are descendants of nodes in purpose.

Definition 2.5.2.5. (Role Attribute).

It is the pre-assigned, specific descriptions associated with each role.

Definition 2.5.2.6. (System Attribute).

It is a system information that affects the authorization like the system state, time, and specific access machine.

Definition 2.5.2.6. (Conditional Role).

It utilizes both the role attributes and the system attributes to describe a specific set of users in a particular system environment

2.5.3. PBAC Database Model

The following Figure 2-9 shows the PBAC database model followed by the description of its core elements.

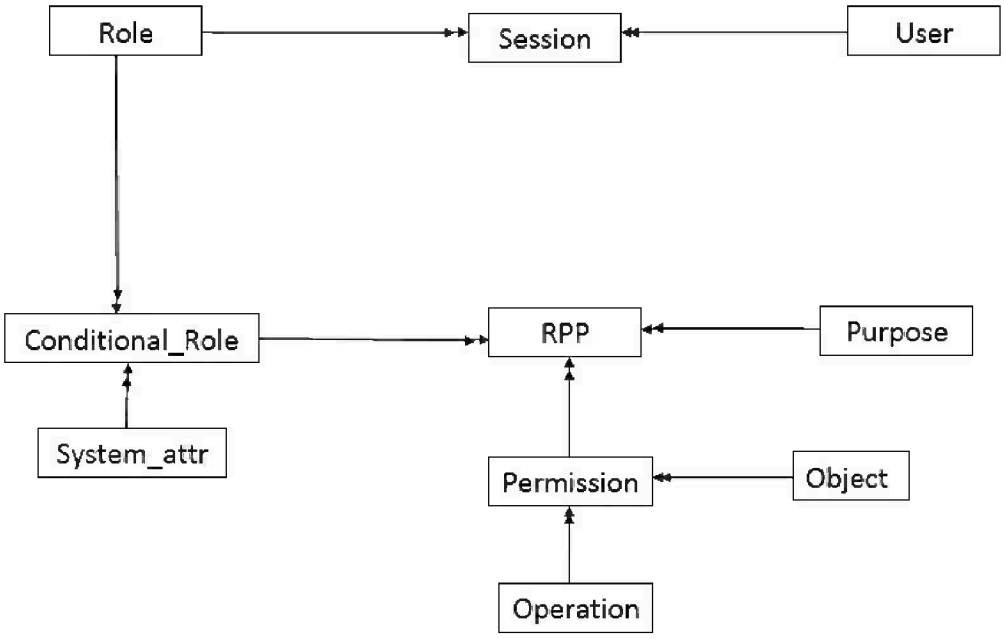

Figure 2-9 PBAC Database Model

Role: is a job function or job title within the organization associated with its authority and responsibility.

User: it can independently access the computer data and resource and maybe person or application program.

Session: is the mapping between the user and the activated subset of the roles the user assigned to.

Conditional Role: conditions on roles, like system conditions.

System_attribute: system condition on roles, roles may be activated only in a specific time interval, or activated for users only logged in from specific machines.

Purpose: is the reason for accessing the data.

RPP: is the relation between roles, purposes, and permissions.

Object: database object.

Operation: there are many operations, for example: query, add, delete, modify, and so on.

Permission: it will grant one or more data in computer system by some way in the range of user access permissions.

2.6 Conditional Purpose based access control model (CPBAC)

2.6.1. Introduction

CPBAC allows users to use some data for certain purpose with conditions. More information from data providers can be extracted while at the same time assuring privacy that maximizes the usability of consumers' data. During the data collection procedure customers are informed about the purposes of enterprises. Customers then decide whether their information could be used or not for a certain purpose and certain conditions. That means data providers are given an option of using their data with certain purposes and certain conditions. Using conditions allows more information to be extracted while assuring the same customer privacy [7].

The basic concept of CPBAC is based on purposes. Permissions are assigned on the combination of conditional roles and conditional purposes.

In order to recognize the model clearly, consider the hypothetical database1 in table 2-1.

Table 2-1 Hypothetical database1

Name	*Age*	*NameIP*	*AgeIP*
Alice	*24*	*<{M},{A}>*	*<{A},{M}>*
Nora	*45*	*<{M},{A}>*	*<{A},{M}>*

A= {Admin purpose}, M= {Marketing purpose}, IP = {Intended purpose} =< {Allowed intended purpose}, {Prohibited intended purpose}>

If we take a query "SELECT name FROM Table 2-1 FOR Marketing Purpose" and the results from this query is "Alice, Nora" and if we have another query-"SELECT name, age FROM Table 2-1FOR Marketing Purpose" it returns nothing because prohibited intended purposes override the allowed intended purposes.

This model protects privacy of consumers as it considers customers' requirements but it occurs more information loss. So "Is it possible to extract information from PIP at least conditionally?"

Consider the following Hypothetical database2 and Intended purpose hypothetical database2 in table 2-2 and table 2-3, respectively.

Table 2-2 Hypothetical database 2

Name	*Age*	*NameIP*	*AgeIP*
Alice	*24*	*<{M},{M},{A}>*	*<{A},{M},{M}>*
Nora	*45*	*<{M},{M},{A}>*	*<{A},{M},{M}>*

A= {Admin purpose}, M= {Marketing purpose}, IP = {Intended purpose} =< {Allowed intended purpose}, {Conditional Intended purpose}, {Prohibited intended purpose}>

If we run the second query, "SELECT name, age FROM Table 2-2 FOR Marketing Purpose", the result would be "(Alice, 20-30), (Nora, 40-50)". So we managed to get more data without revealing the consumer privacy, and this data can be used in many applications like data mining applications.

Table 2-3 Hypothetical database Intended purposes representation

	Name	Age
AIP CIP PIP	Alice A *	24 20-30 *
AIP CIP PIP	Nora N *	45 40-50 *

IP = {Intended purpose} =< {Allowed intended purpose}, {Conditional Intended purpose}, {Prohibited intended purpose}>

2.6.2. CPBAC Model Definitions

P3P defines purpose as "the reason (s) for data collection and use" and specifies a set of purposes [13].

Definition 2.6.2.1. (Purpose and Purpose Tree)[17].

A purpose describes the intentions for data collection and data access. A set of purposes, denoted as ω, is organized in a tree structure, referred to as Purpose Tree and denoted as Ω, where each node represents a purpose in ω and each edge represents a hierarchical relation between two purposes. Let r_i, r_j , be two purposes in Ω. We say that r_i is an ancestor of r_j (or r_j is a descendent of r_i) if there exists downward path from r_i to r_j in Ω.

Figure 2-10 is an example of purpose tree, where each node represents a purpose in ω and each edge represents a hierarchical relation between two purposes.

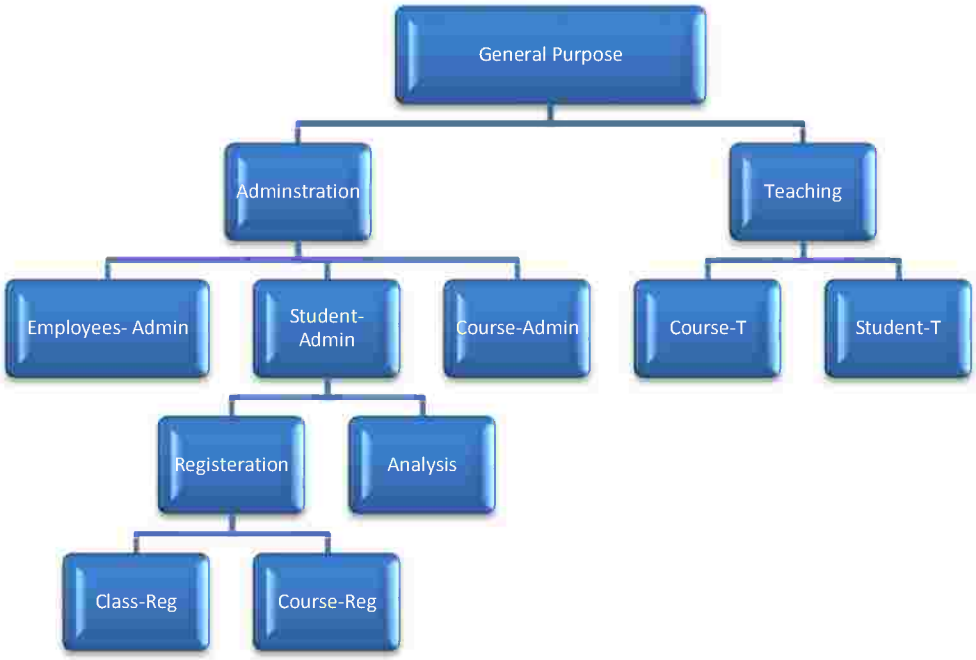

Figure 2-10 Purpose tree

Definition 2.6.2.2. (Access Purpose).

An access purpose is the reason for accessing data objects, and it must be determined by system when data access is requested. So access purpose specifies the purpose for which a given data element is accessed.

Definition 2.6.2.3. (Intended Purpose).

Intended purpose is the purpose associated with the data and regulating data access. It is divided into three types: Allowed Intended Purposes (AIP), Conditional intended purpose (CIP), and Prohibited Intended Purpose (PIP).

Allowed Intended Purpose allows the data to be used without any restrictions for certain purposes. Conditional Intended Purpose allows the data to be used for certain purposes under some conditions. Prohibited Intended Purpose strictly disallowing accessing the data under certain purpose.

The set of purposes implied by IP, denoted by IP* and the set of conditional purposes, denoted by IP_C^* are defined to be $AIP^{\downarrow}$- $PIP^{\updownarrow}$ and $CIP^{\downarrow}$-$PIP^{\updownarrow}$ respectively, where:

- $R^{\downarrow}$: is the set of all nodes that are descendants of nodes in R, including nodes in R themselves, $R^{\uparrow}$: is the set of all nodes that are ancestors of nodes in R, including nodes in R themselves,
- $R^{\updownarrow}$: is the set of all nodes that are either ancestors or descendants of nodes in R, that is, $R^{\updownarrow}=R^{\uparrow} \cup R^{\downarrow}$.

Definition 2.6.2.4. (Access Purpose Compliance).

Let Ω be a purpose tree. Let IP = {AIP, CIP, PIP} and AP be an intended purpose and an access purpose defined over Ω, respectively. AP is said to be compliant with IP according to Ω, denoted as $AP \rightarrow_{\Omega} IP$, if and only if the following two conditions are satisfied:

1. $AP \notin PIP^{\updownarrow}$.
2. $AP \in PIP^{\downarrow}$.

Where $PIP^{\updownarrow}$: is the set of all nodes that are either ancestors or descendants of nodes in the purpose tree, and $PIP^{\downarrow}$: is the set of all nodes that are descendants of nodes in purpose.

Definition 2.6.2.5. (Conditional Access Purpose Compliance).

Let Ω be purposed tree. Let IP = {AIP, CIP, PIP} and AP be an intended purpose and an access purpose defined over Ω, respectively. AP is said to be conditionally compliant with IP according to Ω, denoted as $AP_c \rightarrow_{\Omega} IP$, if and only if $AP \rightarrow IP_C^*$.

2.6.3. CPBAC Database Model

The following Figure 2-11 represents the CPBAC database model followed by the description of its core elements.

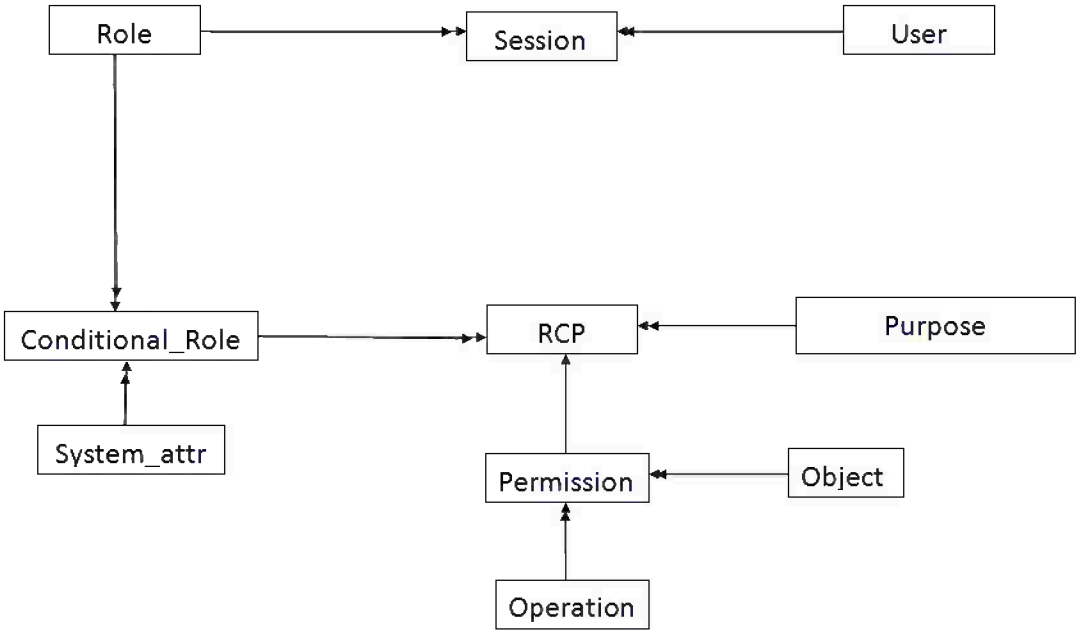

Figure 2-11 CPBAC Database model

Role: is a job function or job title within the organization associated with its authority and responsibility.

User: it can independently access the computer data and resource and maybe person or application program.

Session: is the mapping between the user and the activated subset of the roles the user assigned to.

Conditional Role: conditions on roles, like system conditions.

System_attribute: system condition on roles, roles may be activated only in a specific time interval, or activated for users only logged in from specific machines.

Purpose: is the reason for accessing the data.

RCP: is the relation between roles, purposes, and permissions.

Object: database object.

Operation: there are many operations, for example: query, add, delete, modify, and so on.

Permission: it will grant one or more data in computer system by some way in the range of user access permissions.

2.7 Access control model based on Multi-Role and Task (MD-TRBAC)

2.7.1.Introduction

MD-TRBAC combines the advantages of TRBAC model, and adds a new concept of action scope between the user and the assigned roles by removing the role of inheritance in the traditional model 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 the user permission changes dynamically [21].

As stated above MD-TRBAC model applies scope inheritance; in traditional TRBAC model, the high role inherits total permissions from the low role. But in the actual application, high role can only inherit some permission or just inherit no permissions. As an implementation is the school hierarchy as shown in figure 2-12. The school hierarchy is divided into multiple domains, each domain contains several roles, and each role only inherit the permissions form the domains it assigned to. For example Employee1 role in College2 domain can inherit permissions from descendant roles in StaffRoom1 domain and Course1 domain, otherwise Employee2 role in College2 role inherits nothing [21].

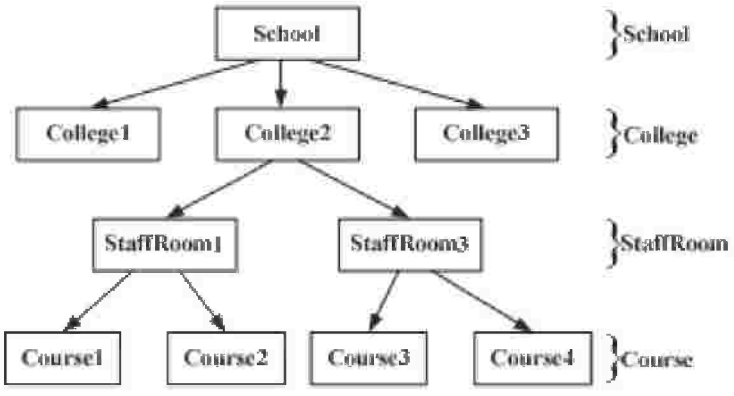

Figure 2-12 Scope instance figure

2.7.2. MD-TRBAC Model Definitions

Definition 2.7.2.1. (Workflow) [21, 22].

A kind of business process, which can be fully or partially automated, and it can also be transferred and implemented among executors based on a series of process, rules, documents, information or tasks.

Definition 2.7.2.2. (Task) [18, 21].

Task is the logical unit in the workflow. It do functional operations based on the own permission. It can also be the activities of the workflow or the composition of several activities.

Definition 2.7.2.3. (Task Classification) [18, 21].

The task can be divided into management tasks, monitoring tasks, and data manipulation tasks according to the their function. Management tasks manage task assignment of low-level user in user group and assignees tasks. Monitoring tasks are used to check the activity status of active task. Data manipulation tasks are used for doing access operation to bottom data. Task can be divided into workflow tasks and non-workflow tasks in the process of task execution. The workflow role also has the permission of obtaining non-workflow tasks and those non-workflow tasks do not require activation and the static authorized method is adopted.

Definition 2.7.2.4. (Dynamic permission management) [18].

Permissions are constantly monitored and activated and deactivated in accordance with emerging context associated with progressing tasks such as in workflow systems. Permission has a lifetime associated with it during, which it is considered valid. Granting, usage tracking, and revoking of permissions are automated and coordinated with the progression of the various tasks. Without active authorization management, permissions will in most cases be "turned on" too early or too late and will probably remain "on" long after the workflow tasks have terminated. This opens up vulnerabilities in systems.

Definition 2.7.2.7. (Domain) [21].

Domain allows users to have the system boundary access permission and do not inherit all permission according to their assigned roles. Only inherit permissions from roles inside their domains. Domain provides a flexible way to divide thousands of objects. The domain administrator can divide the

domain according to function responsibilities, object type, the geographical location of object. The mainly problem domain can solve is decentralized authority management and grading authorized. Both user-role and permission-role management can realize decentralized management by giving administrator distribution scope appropriately as shown in Figure 2-13.

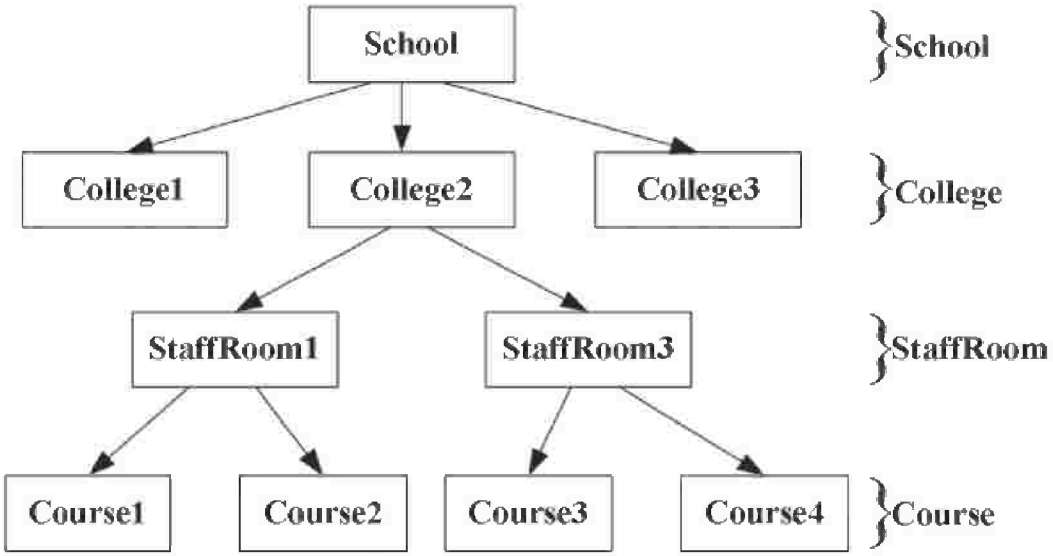

Figure 2-13 MD-TRBAC domain hierarchy example

2.7.3. MD-TRBAC Database Model

The following Figure 2-14 depicts the MD-TRBAC database model followed by the description of its core elements.

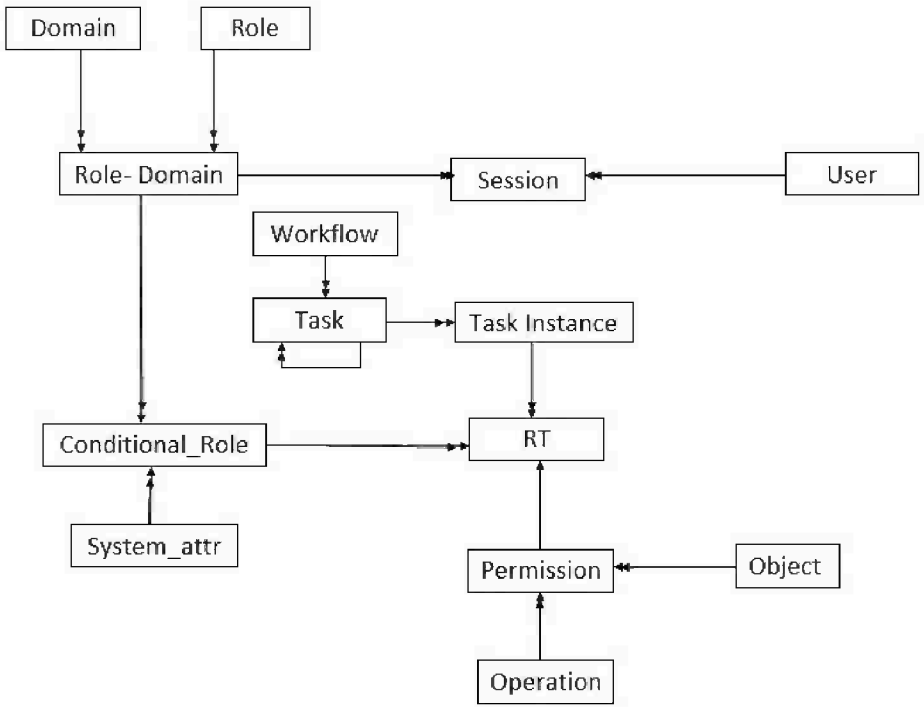

Figure 2-14 MD-TRBAC Database model

Domain: roles-scope, system boundary access permission.

Role: is a job function or job title within the organization associated with its authority and responsibility.

User: it can independently access the computer data and resource and maybe person or application program.

Session: is the mapping between the user and the activated subset of the roles the user assigned to.

Conditional Role: conditions on roles, like system conditions.

System_attribute: system condition on roles, roles may be activated only in a specific time interval, or activated for users only logged in from specific machines.

Task: activities in the system.

Tasks-Instance: the task instance is a dynamic concept in workflow system and also in an instance of operational task and task execution. Each task includes five status: static status, active status, suspended status, termination status, and failed status. The task instance state migration is shown in figure 2.7 [15].

Workflow: group of some business processes.

RT: is the relation between roles, tasks, and permissions.

Object: database object.

Operation: there are many operations, for example: query, add, delete, modify, and so on.

Permission: it will grant one or more data in computer system by some way in the range of user access permissions.

2.8. Flexible Policy based access control model for Workflow Management Systems (PBFW)

2.8.1. Introduction

The security of the workflows is summarized to authentication, authorization, access control, audit, data privacy, data integrity, non-repudiation, security management and administration as described in the Workflow Security Considerations paper published by WFMC [1].

Access control is one of the important technologies, which should ensure that the data in the system could only be accessed by the users who are legal and authorized. In the context of workflow systems, it may operate at the level of: (a) log-on to the workflow service, and (b) access to undertake particular activities or work items according to functional role and/or data sensitivity [24].

In the PBFW model, the notions and advantages of RBAC [11] and TBAC [18] are adopted. As a result, PBFW meets the dynamic and flexible requirements, such as Separation of duty policy (SoD), least privilege constraints, dynamic access control that meets the workflow needs [12].

In order to understand how Separation of duty policy is implemented in the PBFW model, consider we need to implement the following tasks in a company; Task1 (Request promotion) and Task2 (Approve promotion). These two tasks could not be assigned to the same role and user because they are conflicting entities. So there is a need for a policy to apply that, which is the Separation of Duty using policies. Separation of duty policy is supported by means 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 and authority for an action or task over multiple

people, thereby raising the risk involved in committing a fraudulent act by requiring the involvement of more than one individual.

2.8.2. PBFW Model Definitions

Definition 2.8.2.1. (Workflow).

Workflows mainly consist of users, operations and objects. Users deal with the objects through operations, as shown in Figure 2-15. Users want to operate some information objects for their business activities. The final goal of access control in workflow is to judge whether an access request to an object from a specific user is valid or not. Figure 2-15 shows the related factors in the workflow systems [19].

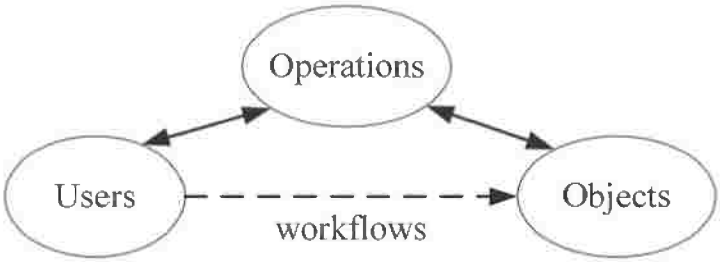

Figure 2-15 Related factors in workflows

Definition 2.8.2.2. (Least Privilege Principle).

User can only work with the minimum necessary set of permissions of a task when he is performing the task, and the set of permissions will be withdrawn as soon as the task is finished.

Definition 2.8.2.3. (Separation of Duty).

As a security principle, SoD has long been recognized for its wide application in business, industry, and government. Separation of Duty is a security principle used to formulate multi-person control policies, requiring 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 and authority for an action or task over multiple people, thereby raising the risk involved in committing a fraudulent act by requiring the involvement of more than one individual [25][26].

2.8.3. PBFW Principles

Principle 2.8.3.1. Different roles with different permissions

Principle 2.8.3.2. No ring in Role hierarchy relation, as shown in Figure 2.16 (a).

Principle2.8.3. 3.Same permissions cannot be assigned to different roles in a RH relation, as shown in Figure 2.16 (b).

Principle 2.8.3.4. Do not assign roles in a RH to a same user, as shown in Figure 2.16 (c).

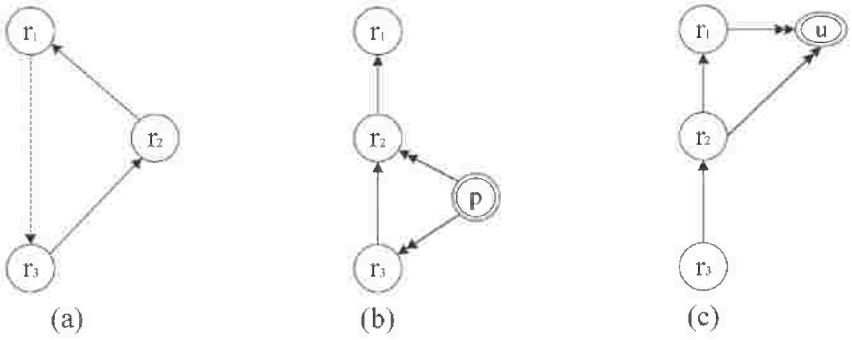

Figure 2-16 Authorization principles

Principle 2.8.3.5. Conflicting roles should not be assigned to a user in a workflow [23] [27].

Principle 2.8.3.6. Conflicting permissions should not be assigned to a task in a workflow.

Principle 2.8.3.7. Conflicting tasks definitions should not be assigned to a role in a workflow_instance

2.8.4. PBFW Database Model

The following Figure 2-17 shows the PBFW database model followed by the description of its core elements.

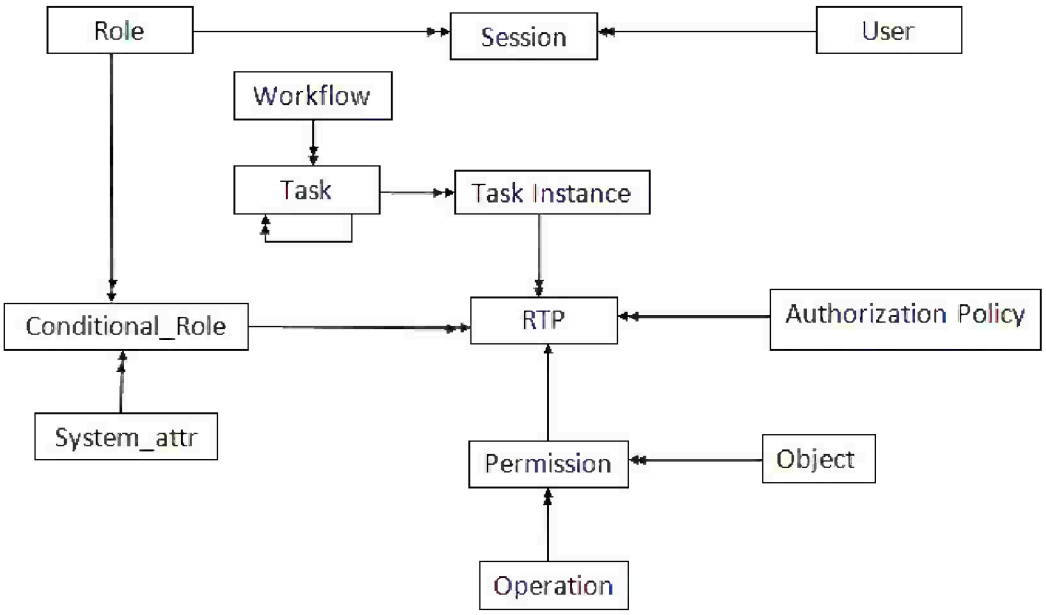

Figure 2-17 PBFW Database Model

Role: is a job function or job title within the organization associated with its authority and responsibility.

User: it can independently access the computer data and resource and maybe person or application program.

Session: is the mapping between the user and the activated subset of the roles the user assigned to.

Conditional Role: conditions on roles, like system conditions.

System_attribute: system condition on roles, roles may be activated only in a specific time interval, or activated for users only logged in from specific machines.

Task: activities in the system.

Tasks-Instance: the task instance is a dynamic concept in workflow system and also in an instance of operational task and task execution. Each task includes five status: static status, active status, suspended status, termination status, and failed status [15].

Workflow: group of some business processes.

Authorization policies: a set of authorization policies.

RTP: is the relation between roles, tasks, policies, and permissions.

Object: database object.

Operation: there are many operations, for example: query, add, delete, modify, and so on.

Permission: it will grant one or more data in computer system by some way in the range of user access permissions.

2.9. The need to extend the related work

A summary of the related work taxonomy, described above, could be illustrated by revisiting the taxonomy depicted in Figure 1-1 as shown in Figure 2-18.

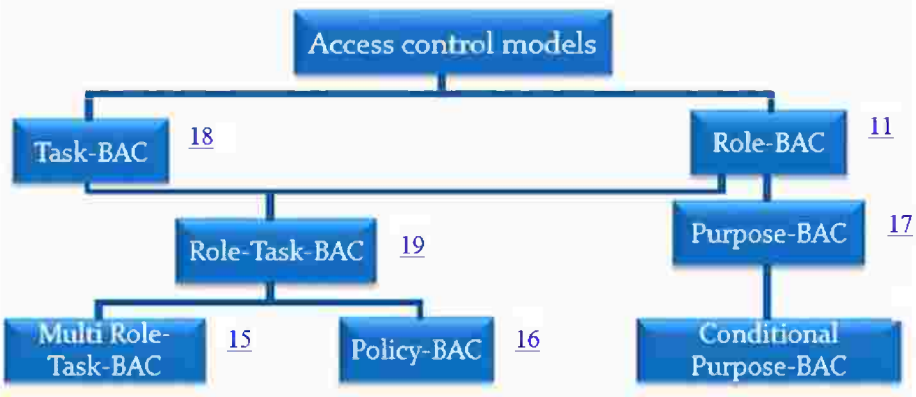

Figure 2-18 Related work taxonomy

Close study of related work reveals.

1. The most powerful access control model is the **CPBAC.** However, it has static permission assignment and does not fit the workflow and enterprises' needs. Therefore, it is useful to combine the advantages of the **CPBAC** model, MD-**TBAC** model and **PBFW** model**.** The

proposed model will have a reliable data management by using the conditional access purpose concept. In addition achieving scalability and meeting the workflow environment requirements.

2. The following features are needed to be implemented in the access control system. As summarized in table 2.4.

    A. Task dependency.

    B. Dynamic permissions management.

    C. Using data conditionally.

    D. Dynamic Separation of Duty using policies.

    E. Scope inheritance.

3. There is also a need to implement the proposed model over different environment like data streams, distributed systems, and XML databases.

4. **Table 2-4 Existing database access control models main characteristics**

Features	CBBAC model	MD-TRBAC model	PBFW model
Task dependency	X	✓	✓
Dynamic permission management	X	✓	✓
Using data conditionally	✓	X	X
Dynamic Separation of duty policy	X	X	✓
Scope inheritance	X	✓	X

2.10. Scope of Work

The scope of this thesis is to implement and study in detail the first and the second mentioned extensions. The last extension is left for future work. A case study will be presented as a proof of concept to simulate the implementation process and evaluate the proposed model.

2.11. Conclusion

This chapter presented a background of most common important database access control models, namely: RBAC, TBAC, TRBAC, PBAC, CPBAC, MD-TRBAC, and PFBW models. The need for extending this models was established. The novel proposed model, with the new desirable features, is described in the next chapter.