UNIVERSITY OF TARTU
Institute of Computer Science
Software Engineering Curriculum

Darwin Sivalingapandi

Comparison and Alignment of Access Control Models

Master's Thesis (30 ECTS)

Supervisor(s): Raimundas Matulevičius

Tartu 2017

Comparison and Alignment of Access Control Models

Abstract:

Security system policies are implemented in the computer through access control mechanism. The primary controls that the access control mechanism possesses are confidentiality, integrity, and security. Access control mechanism can be applied through any of the access control models. It is is a way of protecting information or resources from the unauthorized user to provide access to authorized user. There exist different access control models in which all models are not adequate for today's digital environment. So, the problem arises in difficulty faced to choose which access control model suits well for a particular type of multi-user infrastructure with various access needs. Access control model differs according to the environment. There is an environment which grants access to its users within a particular network and for an environment which has users, who switch dynamically between different networks to access resources. Hence, determining the right model for the efficient use of resources and network is difficult, unless, there is a way to implement the needed concepts in our existing model as to make our new flexible model.

Access control standards for managing different access privileges are complex to understand. With the emerging technologies, components of a system are getting updated, so, it will be a challenge to find out the suitable and flexible access control model that matches the system. Even though there are different access control model available, there is the real problem in finding out the needed access control mechanism which can be employed for the improvement of our new access control model for the efficient use of the resources to be accessed securely.

The solution is to understand the weak and strong features of access control model by comparing different models and aligning their best features to compose into a flexible access control model. It is achieved with the help of systematic survey, where a group of audience validated that access control model can be understood and compared with three main components, subject, policy and object with underlying principles, meta-models and examples of three different access control model.

Analytical comparison of different access control model is drawn from a report of how the audience deals with it at various cases that were analyzed. This survey helped to receive the opinion of different people realistically, such that this empirical way of conducting experiments concludes with the way for finding strong and weak factors. Finally, all the strong factors are aligned to form a new flexible access control model.

The result helps to compare, study and implement a suitable and necessary access control system. It also makes us think in a way how a new access control system can be analyzed and compared with the existing ones. This research work result can be used for further research in future for the potential enhancement of newer access control models.

Keywords:

Access Control, Meta-model, Analytical Comparison

CERCS:P170 omputer science, numerical analysis, systems, control

Juurdepääsu Poliitikate Võrdlus Ja Joondamine

Lühikokkuvõte:

Turvasüsteemipoliitikat rakendatakse arvutis juurdepääsu kontrollimehhanismi kaudu. Juurdepääsu kontrollmehhanismi peamised kontrollid on konfidentsiaalsus, terviklikkus ja turvalisus. Juurdepääsu kontrollmehhanismi saab rakendada mis tahes juurdepääsu kontrollmudelite kaudu. See on viis, kuidas volitamata kasutaja eest teavet või ressursse kaitsta, tagades juurdepääsu ainult volitatud kasutajale. On olemas erinevad juurdepääsu kontrollimudelid, kus kõik mudelid pole tänapäeva digitaalkeskkonnale piisavad. Seega tekib probleem probleemi lahendamisel, milline juurdepääsukontrolli mudel sobib teatud tüüpi mitme kasutaja infrastruktuuriga, millel on erinevad juurdepääsuvajadused. Juurdepääsu kontrollmudel erineb vastavalt keskkonnale. On olemas keskkond, mis annab juurdepääsu oma kasutajatele konkreetses võrgustikus ja keskkonnas, kus on kasutajaid, kes vahetavad ressursse pääsemiseks dünaamiliselt erinevate võrkude vahel. Seega on ressursside ja võrgu tõhusa kasutamise õige mudeli määramine keeruline, välja arvatud juhul, kui on võimalik olemasolevas mudelis kasutada vajalikke kontseptsioone, et muuta meie uus mudel paindlikumaks.

Juurdepääsu kontrollimise standardid erinevate juurdepääsupõhiste õiguste haldamiseks on keerukad. Tekkivate tehnoloogiatega muutuvad süsteemi komponendid ajakohastatuks, seega on väljakutse leida sobiv ja paindlik juhtimismudel, mis vastab süsteemile. Isegi kui saadaval on erinevad juurdepääsukontrolli mudelid, on tõeline probleem leidmaks vajalikku juurdepääsu kontrollimehhanismi, mida saab kasutada meie uue juurdepääsukontrolli mudeli täiustamiseks, et turvaliselt juurde pääsedes ressursse tõhusalt kasutada.

Lahenduseks on mõista juurdepääsu kontrollimudeli nõrku ja tugevaid omadusi, võrrelda erinevaid mudeleid ja viia nende parimad omadused kokku paindliku juurdepääsu kontrollimudeli koostamiseks. See saavutatakse süstemaatilise küsitluse abil, kus osalejad kinnitavad, et juurdepääsu kontrollimudelit saab mõista ja võrrelda kolme põhikomponendi, subjekti, poliitika ja objektiga, mille aluseks olevad põhimõtted, metamudelid ja kolme erineva juurdepääsu kontrollimudeli näited.

Erinevate juurdepääsukontrolli mudelite analüütiline võrdlus põhineb aruandel selle kohta, kuidas kasutajad sellega erinevatel juhtudel tegelevad. See uuring aitas saada erinevate inimeste arvamust reaalselt nii, et see empiiriline katsete läbiviimise viis suudaks leida tugevaid ja nõrgemaid tegureid. Lõpuks viiakse kõik tugevad tegureid kooskõlla uue paindliku juurdepääsukontrolli mudeli loomisega.

Tulemus aitab võrrelda, uurida ja rakendada sobivat ja vajalikku juurdepääsu kontrollisüsteemile. See paneb meid mõtlema ka sellele, kuidas saab uut juurdepääsu kontrollisüsteemi analüüsida ja võrrelda olemasolevatega. Reaalajas vaatajaskonna abil saab selle väljund olla realistlik. Seda uurimustöö tulemust saab kasutada juurdepääsu kontrolli mudelite edasiseks täiustamiseks.

Võtmesõnad:

Juurdepääsukontroll, Metamudel, Analüütiline võrdlus, Joondamine

CERCS:P170 - Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaatjuhtimisteooria)

Table of Contents

1 Introduction

1.1 Motivation

The motive of this research is to derive criteria from comparing different access control models and understanding its strong and weak features. Access control models are capable of protecting the resources and grant access to the subjects when needed. Considered access control models for this research are Discretionary Access Control, Mandatory Access Control, Risk adaptive access control, Attribute based access control and Role based access control.

This research includes a survey of the access control models including a systematic analysis of their key principles with meta-models and examples. Systematic analytical comparison and empirical comparison of the selected access control models are done and validated with the help of a survey. The main problem here is there is no general way of understanding an access control model easily. Also, there is no common flexible model which fits all possible existing access control model.

1.2 Scope

The scope of the research work is to define comparison criteria for access control models. Existing works like [27] tells about the reference model by **combining roles from RBAC and attributes from ABAC** to become a scalable, flexible, auditable and understandable access control model, but, it fails to give a general comparison criterion or a general unified structure of an access control model. [26] suggests considering **three abstractions in an access control model,** such as, policies, models and mechanisms. Policies describe how access is managed and who will access in what condition. Mechanisms enforce policies and translate user's access request. Models connect policies and mechanisms. This paper describes only one of the component of access control model, policy in detail, but, it fails to describe other components like subject and object specifications and its other related components in much detail. [24] explains the **need for a unified model for modeling authorization in an enterprise context**. It proposes a unified model which is claimed to be flexible with suitable configurations. So, this paper shows the study of few access control models and its components and they unified it based upon the similar components, but, they fail to focus more on the structure of three **main building** blocks (Subject, Policy and Object) of an access control model, as, this is the base of any access control model.

1.3 Research Statements

Interest in this research topic urged to offer a solution for a research question "**How can we compare an access control model and align it with best suitable concepts?**" this can be answered with the following research questions. RQ1:"**What are the most important access control models?"** helped to find three main access control model to conduct this research. As each access control model has its key concept and each of them is different from other access control model, that provided the flexibility in conducting this research. RQ2:"**How do we define comparison criteria for an access control model?",** This gives the essential key elements for comparison. RQ3:**" How can anyone understand an access control model with the help of**

three building blocks?". Provides a solution through deriving comparison criteria and it is validated by a group of audience. RQ4:**"How the conceptual model is constructed into a unified model?",** Shows how the unified model is constructed with the best access control concepts. RQ5:**" How can we verify our constructed unified model?".** Finally, the constructed unified model is validated here. RQ6:**" What is the proof that this verified model is consistent and reliable?".** SRQ6.1: **What are the limitations this unified model overcome?** The answer to this question helped to develop an example to see the fitness of constructed model.

1.4 Explanation of Contribution

This paper starts with introducing what an access control model, why it is needed is and how it is functioning. For this study, I took three types of access control models, namely, RBAC, ABAC & RAdAC, because of its varied concepts. With this, I will explain its functional mechanism and discuss its specific characteristics in detail with a **real-time example (University of Tartu Study Information System** [28]**)**. Validation of its characteristics will be carried out in the form of a survey. After this validation, a unified and flexible Metamodel will be constructed based upon the key concepts of three access control models which are followed by a **validation against another unified model** [24]. Conclusions will be made based on the validation.

1.5 Structure of Work

The paper is organized as follows: **Section 2** presents and describes what are the most important access control models and what are its key concepts. **Section 3** gives every detail about the three main building blocks of an access control model which helps in defining comparison criteria and leads to the construction of the unified model. In **Section 4**, shows the validation of three main building blocks derived in the previous section. **Section 5**, provides the flexible unified model, constructed with best building blocks. tells us the survey findings. **Section 6** consists of validation process of the constructed unified model. **Section 7** has an example of a unified model and this research paper is then concluded in **section 8**.

2 Overview of Recent Access Control Model

Objective of the section is the result of finding an answer to the below question

Q: What are the most important access control models?

For there are different access control models, which have been designed by control policies that formalize them. Models included in this research are Mandatory access control [23], Discretionary access control [22], Risk-Adaptive Access Control [4], Role-Based Access Control [12], and Attribute-based Access Control [16] as these are considered to be the main access control models, which leads to the development of other hybrid models.

2.1 Mandatory Access Control

It implements security controls on access objects. It is generally based on sensitivity labels. Sensitivity labels are the labels that have two components, classification (level) and compartments (categories) [12]. Classifications are standardized with the measure of trust. The level of data can be classified accordingly in any of the following category SECRET, TOP SECRET, CONFIDENTIAL or UNCLASSIFIED. These four categories are the degree of protection. Industries have their own level of classification, hence, they are not standardized. Compartments represent categorization into certain groups according to the nature of the object. Max Label [23], Confidential, and Public are the three types of compartment an object can have. Confidential has three exclusive sub-compartments as follows Restricted, need to know and Internal Use Only. Public and Max Label don't have any sub-compartments. MAC policy [23] uses this sensitivity label to look for access control permissions. MAC [23] is associated with multilevel security model of Bell-LaPadula confidentiality model [23]. It was named after the person who found this model. This model assigns security labels to subjects and objects by using security properties [23].

2.2 Discretionary Access Control

It is an authorization based user-centric access control. Every system has its object, every object has its owner called subjects. Subjects are responsible for determining access control policies for their objects. Owner creates an object and determines access privilege to the users the owner wants to assign. Easy to implement in applications that are in distributed network. This mechanism allows an entity that performs an action in a system to gain control over the access rights without the pre-specified rules [12]. The file owner grants permissions on file for those users if the control over the access rights is perfectly managed. It is a technique for confining access to objects and the group it belongs. Controls are flexible in the view that certain subjects can transfer their access permission to the other subject permissions. Entities are known as subjects. These entities cause the flow of information between objects. Subjects can be users, process or devices. Generally, they are users who perform any operations on files. When the user assigns a job to perform in a system can be treated as a subject or the process that takes the attribute of the user can also be treated as a subject. If a process initiates a sub process it can be treated as objects. Normally files are called as objects if a file is split up into single pages, it will also be called as objects. In the same way, directories will also join as objects. Generally, a file is the common way of representing segments,

files, etc. Discretionary access control looks through the access information along with its files. So that hardware associated will be treated as files along with its access control information. Control permissions associated with the file actuate what all the subjects can access or cannot access to do any operations as permitted. Consistency in DAC implementation leads to less complexity in handling its objects. Objects protected by DAC depends on the capability of the system that contains its directories, devices and communication channels. The general-purpose operating system provides DAC with a user-friendly interface that displays a list of objects protected by DAC [22]

2.3 Risk-Adaptive Access Control

Risk Based Access control system has the capacity to adapt according to the increasing need for secure sharing. Which also resolves the pitfalls of traditional access methods**.** Whenever a subject requests for access RAdAC calculates the security risk and operational need of each request and provides or deny access. Hence it provides access dynamically based on a modern real-time paradigm [1]. Access control systems other than RAdAC does not have the flexibility and decision-making system based on the various situational factors. Such situational factors are characteristics of people, characteristics of IT components, characteristics of objects, environmental factors, situational factors, and heuristics.

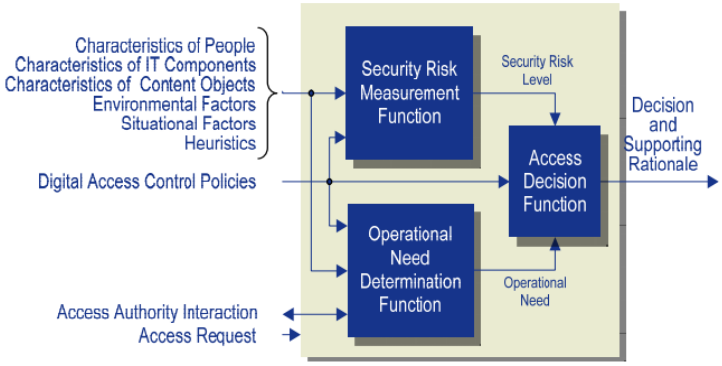


Figure 1. RAdAC Functional View [2]

2.3.1 RAdAC Meta Model

RAdAC system allows a subject to initiate access request through the IT component, which can be a computer, tablet, mobile device, etc., Each access request towards the resource goes through an RAdAC decision system. This is to determine the security risk and operational need. Once it is done it will be compared with the access policy to determine if the access can be granted for the requested resource. The access request of each and every user is treated as homogeneous.

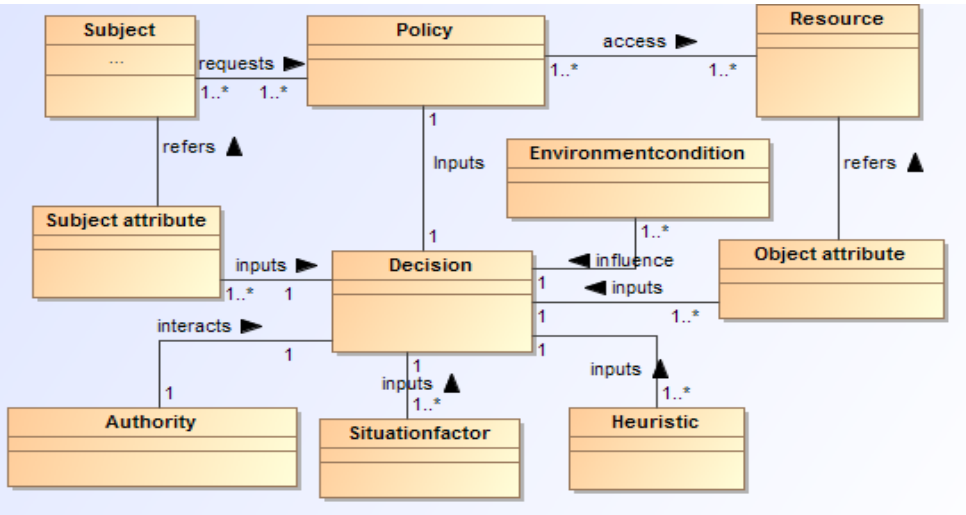


Figure 2. RAdAC Meta Model.

Subject: The subject is the user, who requests for access the resource. A User is defined as an entity that is requesting access. Although the subject of a user can be extended to include machines, networks, or intelligent autonomous agents, for simplicity we limit a subject to a human being. We assume that the subjects and other components are defined and represented by their attributes. Examples of subject attributes include subject's identity, subject's security clearance, and so on.

Subject Attribute: It is defined as a characteristic of the subject it can be a computing device through which a subject is requesting access. Examples, where attributes of IT component are used in making access control decisions are a user's access to online banking account, is restricted to a pre-registered mobile device or an organization's policy allow VPN access to employees from a device registered in its domain

Policy: Highly robust infrastructure must exist to provide the access control policy enforcement needed to support RAdAC based decisions. At a high level, this infrastructure element acts as a repository for access control policies in a machine-readable format is served. Determining policies are always challenging. Policies are the representation of policy maker's intent of sharing information under certain conditions and translate it into machine-readable format statements.

Decision: RAdAC decision can be defined based on the security constraints and operational need defined in this section. Its Purpose is to find out the operational need and security level of the access request to access the resource. There are a lot of factors to captures the security risk from the access request. Situational factors determine under which conditions an access decision can be made and heuristics factor is defined to capture the access decisions that are made and also it provides input to the decision function. Under certain conditions, this RAdAC model allows for to override by an approved authority for the security risk at decision function.

Object attribute: It is defined as a characteristic of the object. Which can be a resource component or a device, A device here is defined as a computing device through which the access request of the subject access the resource. Sufficient knowledge of the information and security robustness of a platform is called as an object attribute.

Object: Access is authorized or restricted to objects, depending on the labeling on the resource and the user's credentials. A resource is an object (e.g., a file, a resource in the system). Multiple users can share common resources.) A resource can be a tuple in a database, a table, a file, a service.

Access Request: Access control policies are enforced through a mechanism that translates a user's access request. It is a request to access the information or data or some valuable resource in a safeguarded network. This request is initiated by the user or subject to gain access to the resources. It is then sent to access control mechanism for authentication. Access requests not satisfying the given patterns of access control policies are identified and it undergoes the process of risk measurement and operational need function [5].

Characteristics of People: This is one of the risk categories, its purpose is to assess the trustworthy of people who are requesting for access. Factors that help to determine the characteristics of people are role, rank, clearance level, access level, previous violations, and educational level. The role is the position of authority of the people in any organization. Rank determines the risk associated with the people's position in an organization. For security measure, its likelihood of occurrence decreases as the people's rank increases. Clearance level is inversely proportional to the risk value. The higher the clearance, lower the risk value. Access level defines the access level of the user. Risks are lower if the resource owner gives access to the user. If not or when the access was granted by the third party, risks are higher. In terms of previous violations risk factor increases. If there are no violations in the past it won't lower the risk level. Educational level of the user either increase or decrease the risk factor based on the amount of security related training the user received. If the user received security related training, possibilities of committing a security violation are very less. If not higher possibilities for the occurrence of security violation due to negligent action [4].

Characteristics of IT Components: IT components are the components that involve during the whole information transaction. The purpose of assessing risk factor here is to know how safe the data will be transit. The value of risk will be low whenever the level of protection is higher. Factors that help to determine the characteristic of IT components are the type of machines used, the traveling distance of the information, the level of encryption used to protect the information, type of network, authentication type of the user request, the connection type of the information, and type of application [4].

Situationfactors: It is also called as situational factors security risk lies in the surrounding situation when information transaction happens. Risk factors to be considered here are specific mission role of the user who requests access, time sensitivity of the information, transaction type, auditable or non-auditable, audience size. Its purpose is to identify the security risk that revolves around the transaction, not the associated data or the user [4].

Environmencondition: In the other way it is also called as environmental factors are the factors associated with the environment surrounding the transaction. It analyses the increased likelihood of how the surrounding environment can be exploited during the transaction. Its purpose is to assess the risk category of the transaction with regards to the environment. Factors considered for assessment are the current location of the requesting user and the data source [4].

Characteristics of the Information Requested: This characteristic is to assess the risk factors involved in the requested information itself. Its purpose is to analyze the sensitivity of the

information. To find out the risk factors following are to be considered classification level of the data, encryption level required for access, network classification level for data transmission, permission level set in the data, time sensitivity of the data [4].

H*euristics:* This helps to improve access control policies to grant decisions. Like, various access decisions in the past help to improve the decisions. This is used to find the risk category in the transaction based on similar transactions that have occurred before. It helps to fine-tune access control policies, improve future access decisions and to predict the future behavior. Risk knowledge and trust level are the factors help to assess the risk factor of heuristics [4].

Authority Human intervention or access authority intervention is allowed to change access policies and to address the operational need for a category of users under certain conditions. Access authority is an authoritative external source, who can be an administrator or a supervisor of approving authority to override security risk when necessary. Through this RAdAC allows operational need to override the measured security risk [6].

Policy: It is the specification of the rules of access control policies of information objects under different conditions. This is the place where we have to define the degree of operational need required to override the security risk to set it as acceptable. It should be defined in the way that the policy has the relative weight of each factor like personal risk, IT component risk, environmental risk, situational risk, heuristic risk and risks in accessing the requested information. To make RAdAC more successful it is important to implement digital control policies effectively [4].

Security Risk Determination Function: RAdAC implements security risk measurement function, unlike traditional access control models. It converts the measured security factors into a positive or negative decision towards the access request. This security risk measurement function provides a numerical measure of the decision-making process. Here risks are grouped from different factors. Strong digital policies are needed to determine total security measurement [4].

Table 1.Risk Measurement

Numerical Measure	**Explanation**
1	Low Risk
2	Average Risk
3	High Risk

Operational Need Determination Function: This function determines the operational need of the access request and supports the access decision. It helps to give some quantifiable measure to compare with the access policies defined. Digital policies have the requirement for determining acceptable operational need. There are situations that security risk factors can be unacceptable but because of the operational need measure, the access could be granted. But it is necessary to show how operational need outweighs the security risk measure [4].

Table 2. Operational Need Measurement

Numerical Measure	Explanation
1	Low Operational Need
2	High Operational Need
3	Very High Operational Need

Final Access Decision Function: The final access decision functions determine the decision whether to grant decision or not. It gets the input from security risk measurement function and operational need determination function and compares with the digital access policies which are already preset according to the requirements. If the access policies are satisfied then the access is granted else not [4].

2.3.2 RADAC Notational Process Model with Example

Let's consider the example of University of Tartu study information system. It is a system to manage all the information of students studying at the University of Tartu. It can be accessed either with the student user name and password or with an Estonian national ID card. It enables the students to use proactively anytime to know information about the courses and study programs available, provides personalized study timetable for students registered their courses, manages up-to-date information about the student. Helps to register for courses and examinations. Also, it has the information about the academic progress of every student. Factors under which RAdAC can be implemented in OIS as follows. Unexpected situations when there is a need for flexibility in access control policy or violation of security policies in which the access is legitimate. Specific situations can be when a student has to register for a course at different faculty which has the nearest deadline and the person responsible for this registration is in absence. Valuation of coursework submitted by students by another professor when the responsible professor for that coursework is away from work. When there is a decision to be taken for stipend or Erasmus mobility or Erasmus scholarship, the responsible person is away from work due to unexpected sickness.

Basic components needed to apply RAdAC in OIS are requestor characteristics, component characteristic, heuristics, local and global situational factors, environmental characteristics and characteristic of the requested data or information. When any of the above-mentioned factors arise, RAdAC takes certain functions as inputs and quantify them into certain measures to measure the security risk and operational need from the access request.

In the case of the considered example suppose the program coordinator has to give a decision for a stipend in the form of a list of eligible students. Program coordinator should have full access towards the stipend rules and documents also the list of students who applied. So the person will click on gain total access at stipend page which is the access request for RAdAC system.

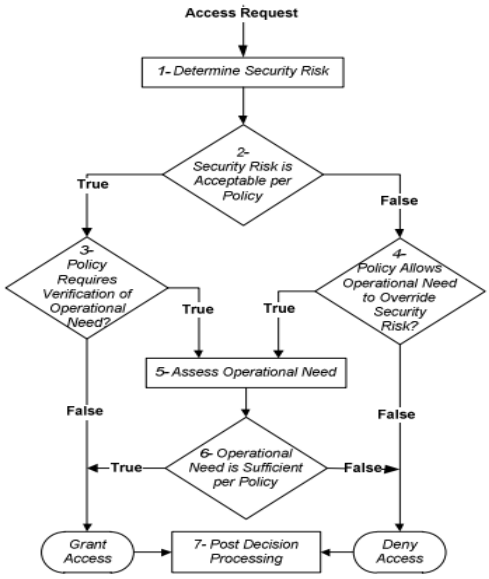


Figure 3. RAdAC Notational View [4]

Step 1 - Determining Security Risk or Security Risk Measurement Function: Security risk plays an important role in granting a decision towards the access request. From the access request made, several external factors are determined and quantified to determine the level of risk. Risk level is determined by the characteristic of people, characteristic of IT Components, situational factors, environmental factors, characteristics of information requested and heuristics. It outputs the quantitative indication of the risk level composed of each area described above [5].

Through the access request of the above example, risk factors are determined as follows. As the requestor is the program coordinator and whose level of the profession has high standards in dealing with students. Who is also closely related to the person (who actually respond for granting stipend decisions) at dean's office. Hence the classification level and encryption level of the requesting data is marked as high (3) because those factors need high-security measures, which is not easily accessible for anyone other than who has permission. Program coordinator characteristics such as role, rank, clearance level, access level, machine type, applications, connection type, authentication type, transaction type, network classification level and permission level are marked as average (2) in terms of security measure, as he is a highly ranked role in the university than compared to others. Rest of the factors such as risk knowledge, trust level, mission role, time sensitivity of the situation, audience size, auditable factor, location, operational environment threat level, and perishable factor are marked as low (1), as those factors cannot demand high-security risk for the program coordinator. Finally, the measured security risk factors are added totally and quantified in terms of percent. Hence for the considered example the measure comes to 51.86%. This result is carried forward to the next step.

Table 3. Risk Measurement Calculation

Risk Factors Considered	Maximum	Actual Security Measure
Characteristics of People		
Role	3	**2**
Rank	3	**2**
Clearance Level	3	**2**
Access Level	3	**2**
Previous Violations	3	**1**
Educational Level	3	**1**
Characteristics of IT Components		
Machine Type	3	**2**
Application	3	**2**
Connection Type	3	**2**
Authentication Type	3	**2**
Network	3	**1**
Encryption Level	3	**1**
Distance from requestor to Source	3	**1**
Heuristics		
Risk Knowledge	3	**1**
Trust Level	3	**1**
Situational Factors		
Specific mission role	3	**1**
Time Sensitivity	3	**1**
Transaction type	3	**2**
Auditable or Non Auditable	3	**1**
Audience Size	3	**1**
Environmental factors		
Current Location	3	**1**
Operational environment threat level	3	**1**
Characteristics of Information Requested		
Classification Level	3	**3**
Encryption Level	3	**3**
Network classification level	3	**2**
Permission Level	3	**2**
Perishable or Non Perishable	3	**1**
Sum Total	81	**42 (51.86 % of 81)**

Step 2 – Comparison of Security Risk or Security Risk Measurement Function against Policy: In this step, it compares the security risk obtained from the previous step with the implemented policies. This is the place to identify the acceptable level of risk for the requested data being accessed. By comparing it with the policies implemented it says whether the security risk is acceptable or not [5]. Hence it compares the measured security risk 51.86% with the policy. Here policy categorizes the result in any of the following. 0% – 49.99% as low, 49.99% – 69.99% as high, 69.99% – 100% as very high. So only low category is set to accept in this level, remaining is not acceptable per policy. According to the result, obtained process flow goes to the step 4.

Step 3 – Policy for Verifying Operational Need: At this step the resulted security risk measure is been accepted and seems ok in terms of security risk. This step is to find out whether the user's request has to be analyzed for operational need. By comparing it with the access policy, may results need in determine the operational need? In this case, it undergoes other steps, if not access with being granted towards the access request [5].

Step 4 - Policy for Overriding Security Risk or Security Risk Measurement Function: This step will be executed only when the security risk was not acceptable, but the request seeker might have the operational need to access the information irrespective of the risk. Here the policy should be capable of demonstrating how the operational need can outweigh the security risk. Here factors like user trustworthiness, user location, IT components, and situational factors can be analyzed again. After the analysis, if the override is allowed further processing is necessary to see if the operational need outweighs the security risk. Access will be denied if the override is not allowed [5]. So the considered example results in 51.86% in security measure and here it verifies with the policy if it allows operational need to overrides the security measure. The policy is set as follows, high (50% to 69.99%) security measure is allowed to override but very high (70% to 100%) is not allowed to override the security risk. According to the policy, it allows this security risk measure to override the operational need, in case if operational need verification results positive. Hence for the outputted result, it allows overriding the security measure.

Step 5 – Assess Operational Need: At this step, the examination is done to determine the operational need to access the object. The policy has to be implemented in such a way that it would specify different requirements for determining operational need because of both which security risks are acceptable and which security risks were not acceptable, undergoes this process. Requestor's location, rank or any discretionary factors might be used to determine the operational need [5].

Operational need for the considered example can be assessed as follows. As the requestor is the program coordinator whose operational need for the factors like transaction type and audience size is considered to be low (1), because whose transaction type will be usual and there is no more than 1 in audience size. In the case of factors like clearance level, risk knowledge, time sensitivity, and classification level are marked as high (2). But in the case of role, rank, trust level and specific mission role it is marked to be very high, as these are the factors that demand high operational needs. Once all the factors are marked it will be totally summed up and it will be converted to a percentage. So the output will be 73.33%.

Table 4. Operational Need Measurement

Factors Considered for Operational Need Measure	Maximum	Actual Security Measure
Characteristics of People		
Role	3	**3**
Rank	3	**3**
Clearance Level	3	**2**
Heuristics		
Risk Knowledge	3	**2**
Trust Level	3	**3**
Situational Factors		
Specific mission role	3	**3**
Time Sensitivity	3	**2**
Transaction type	3	**1**
Audience Size	3	**1**
Characteristics of Information Requested		
Classification Level	3	**2**
Sum Total	30	22 (**73.33 %** of 30)

Step 6 – Comparing Operational Need against Policy: After determining the operational need of the request with the necessary factors, this step comes into action. It compares the determined operational need with the access policy which was already defined to see if it is sufficient enough to grant access, if not the access will be denied. And the policy required to compare at this step should be capable enough to determine sufficient operational need in both stressed and unstressed security conditions [5].

Here the policy is set to accept those operational need measure which has the output more than 70%. In this case, it can override the security risk measure. In the example the output of the operational need measure is 73.33%, which is evaluated by the policy to be positive. Hence the access will be granted.

Step 7 – Post-Decision Processing: Once the request is been processed by granting the access or by denying, its result is stored for post-decision processing. It helps to improvise the decision processing of RAdAC decision engine [4]. Results made will be available for the resource or information owners to help them in improving their access control policies [5].

2.4 Role Based Access Control

RBAC is a simplified access control model to define security access to different systems. Because of its cost effective measures and simple administration. The basic concept of RBAC is users can access an object or resource by getting assigned to roles, permissions are also assigned to roles. Permissions are acquired by users only by being members of roles. The user can have many roles, the single role can be assigned many permissions [7] [12].

2.4.1 Core Components of RBAC

User: User can be defined as a human being. This can also include intelligent autonomous agents such as robots, computers, etc. But to keep it simple, a user in this RBAC model is considered as human being. Users are assigned to roles [7].

Role: Roles are the user groups or can be defined as a job or job title in an organization. Permissions of the users are associated with roles. Roles describe the level of access of the user [7].

Permission: permission is consent or approval of access to one or more objects in the system. Permissions hold the access control to perform any actions. Authorization, access right, and privilege are the terms of the other that represents a permission. Permission helps to perform some action in the system [7].

ActionType: Action type is the four different type of actions used to perform operations on the object. Those actions are Insert, read, write and execute operations [7].

Resource: Resource is also called as objects. The object can be defined as data. Every system in access control model protects its object. Hence, the access control system protects directories, files, ports, and devices with action type [7].

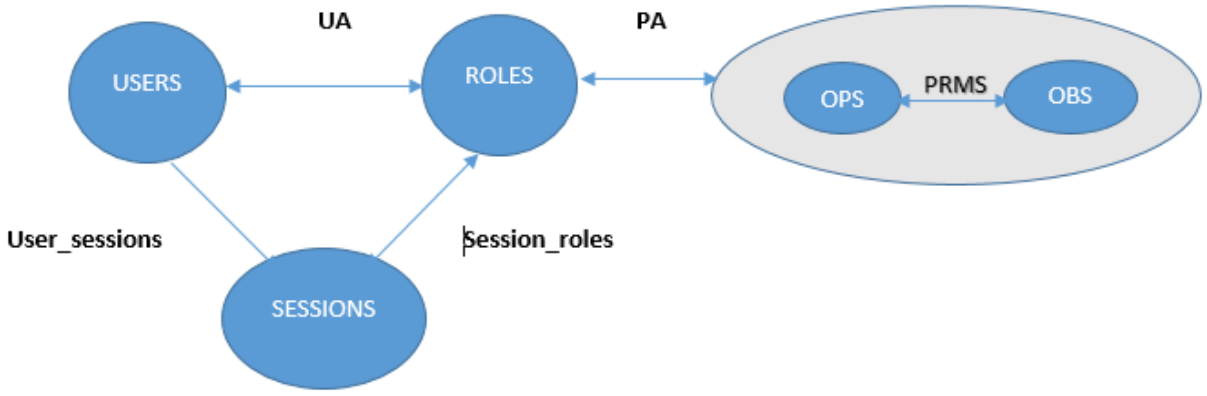


Figure 4. RBAC [7]

2.4.2 RBAC MetaModel

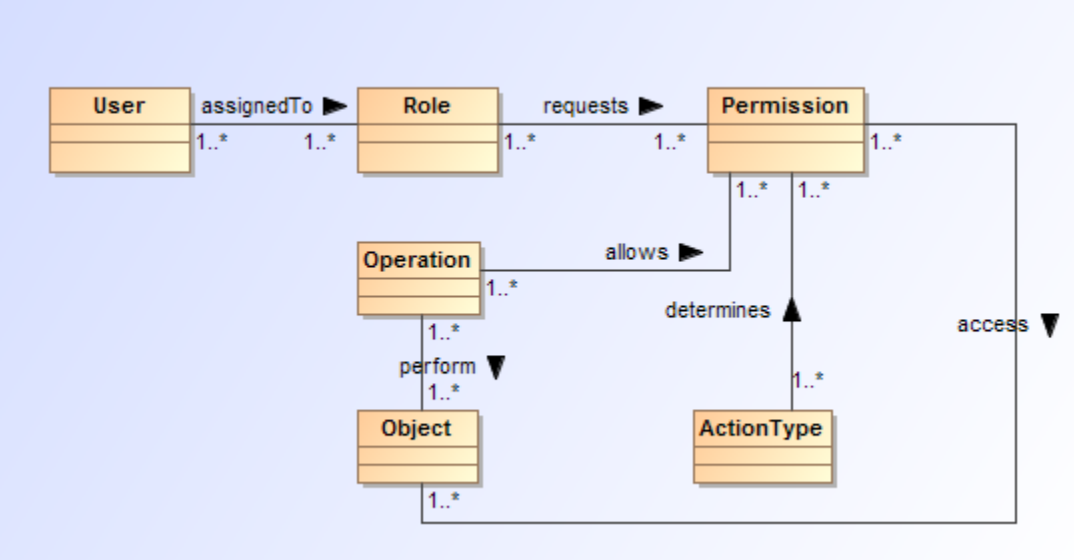


Figure 5. RBAC Meta Model, adapted from [12]

The above representation of the RBAC Meta model and its relation between the components are explained as follows [13].

Users the humans are often represented as a user, but sometimes the user can also be systems and it extends to networks or intelligent autonomous agents. Roles are the job functions defined for an organization. Sometimes it also refers to the responsibilities of the particular role in an organization. Permissions are the organizational consent to perform specific operations. Action type is the elements used to classify permissions. Here every Action Type represents a class of security-relevant operations. Resource type defines all action types available for a particular meta-model type. Resource set represents a user defined a set of model elements used to define permissions and authorization constraints. Protected resources are expressed using the standard

UML elements called model elements. Authorization constraint expresses a precondition as a part of the access control policy. It imposed to every call to an operation of a resource. This precondition usually depends on the dynamic state, the current call, or the environment. The authorization constraint is attached either directly or via permissions, to a particular model element representing a protected resource.

2.4.3 RBAC Example

It is possible to develop the RBAC model using security modeling language. In this section of RBAC example, Secure UML[14] [15] has been used to illustrate the principles of RBAC. The abstract syntax of Secure UML is shown in Figure 6. Secure UML has annotated UML based models with the necessary information for access control. This is based on RBAC with additional authorization constraints. SecureUML[14] [15] defines a vocabulary for roles, role permissions and user-role assignments. SecureUML is well suited for business analysis and different technologies. It is used here to cover access control aspects of RBAC. The semantics of Secure

UML is defined using informal transformation rules. A role in the access control model is transformed into a single element of type security-role or securml.role. The user is transformed into security-user or securml.user. Permission is transformed into security-permission or securml.permission. The resource is transformed into security-resource or securml.resource. To represent any actions we use security-actions or securml.action in SecureUML model. These action includes Insert, Update, Select and Delete[12]

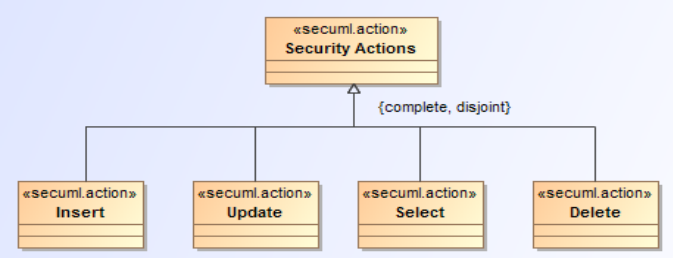


Figure 6 : Security Actions of SecureUML Model [12]

Let's assume the system as Study Information System of the University of Tartu, which provides information about courses, study programs, allow students to register for courses, exams and apply for a scholarship at the University of Tartu. For this example, we consider the situation as the department coordinator needs all of his eligible Students to apply for the newly introduced scholarship by submitting required information through Study Information System of the University of Tartu. Let's say that there are three users Andrew, Mark, and John, who play different roles in the considered example. We also consider that a resource (i.e, ScholarshipApplication), which characterize name and academic results of the student to be secured. This restricts by changing the value of attributes into a state, by defining resource ScholarshipApplication for the role Coordinator and role Student.

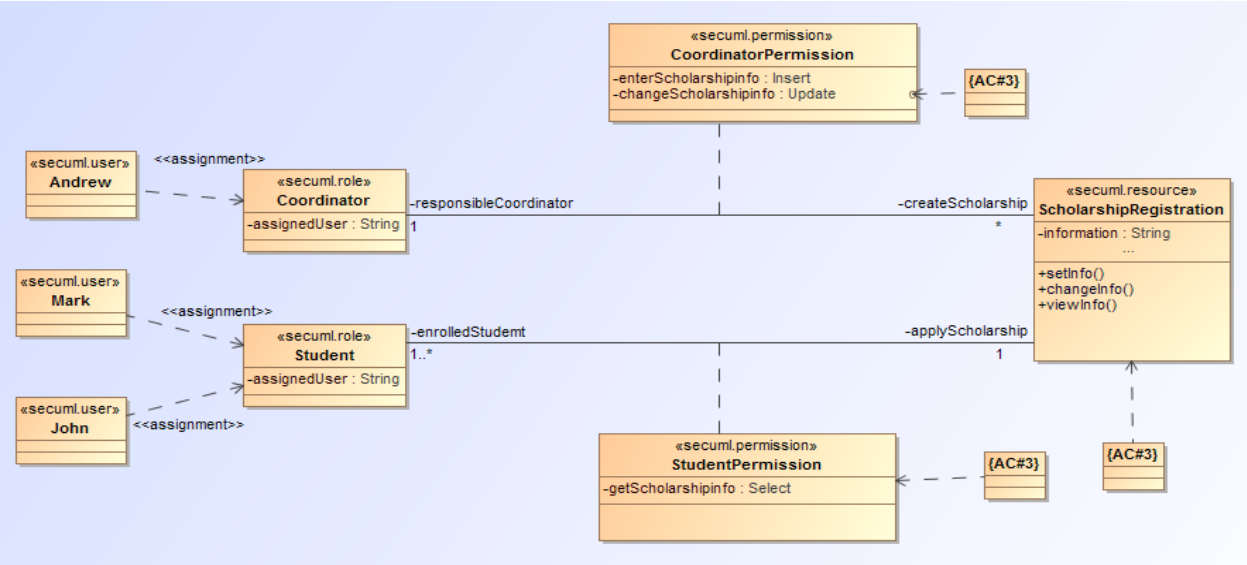


Figure 7: RBAC Example

Association class CoordinatorPermission characterises two actions for Coordinator: (i) action enterRegistrationDetails (of the type Insert) defines that Coordinator can enter information by executing operation setInfo() at class ScholarshipApplication, and (ii) action changeInfo(of the type Update) allows to change the information of the ScholarshipApplication by executing operation changeInfo(). We also define authorization constraints to strengthen permissions.

AC#1:

```
Context ScholarshipApplication :: setDateTime ( ) : void

      pre : self . responsibleCoordinator . assignedUser  ->

            exists ( i | i . assignedUser = ' Andrew ' )
```

Authorisation constraint AC#1 means that operation setInfo() can be executed by user Andrew, assigned to a role Coordinator

AC#2:

```
Context ScholarshipApplication :: changeDateTime ( ) : void

     pre : self . responsibleCoordinator . assignedUser  ->

            exists ( i | i . assignedUser = ' Andrew ' )
```

Authorisation constraint AC#2 defines the restriction for operation changeInfo(). Association class StudentPermission defines the restriction for role Student. It defines an action getScholarshipInfo() of type Read says that only Student can view info in the ScholarshipApplication. To enforce this permission as authorization constraint AC#3 is defined.

AC#3:

```
Context ScholarshipApplication :: viewDateTime ( ) : void

     pre : self . enrolledStudent ->

            exists ( p1 | p1 . assignedUser = ' Mark ' ) and

     self . enrolledStudent ->

            exists ( p1 | p1 . assignedUser = ' John ' ) and

     self . enrolledStudent -> size = 2
```

Authorisation constraint AC#3 defines that only users Mark and John of assigned role Student can execute an operation viewInfo(). In this example security actions like Update, Insert and Select data are defined. An action Delete is not used in this example.

2.5 Attribute Based Access Control

Attribute Based Access Control (ABAC) [16] provides access on the basis of attributes. Normally access control is based on the user identity who requests to perform some operation on an object. This is done either directly or through user groups or assigned roles. This approach has been noted that it. It is been noted that it is burdensome to associate the request for the user, roles or groups. Because they are insufficient in matching with the real world policies. Hence when we look for a solution it can be carried out easily through the attributes of the object, attributes of the user, conditions of the environment that are recognized globally for the policies. This concept is called ABAC. ABAC controls access to its objects by evaluating the rules against the attributes of subject and objects, the operations it deals with, and the environment it dealt with. With ABAC it is easy to implement both Discretionary Access Control (DAC) and Mandatory Access Control (MAC). ABAC has the possibility of allowing. It is called Discretionary Access Control (DAC) because the access control is based on the discretion of the owner. It is also called Mandatory Access Control (MAC) because the system authorizes which subjects can access the data objects. This model is based upon the security labels [16].

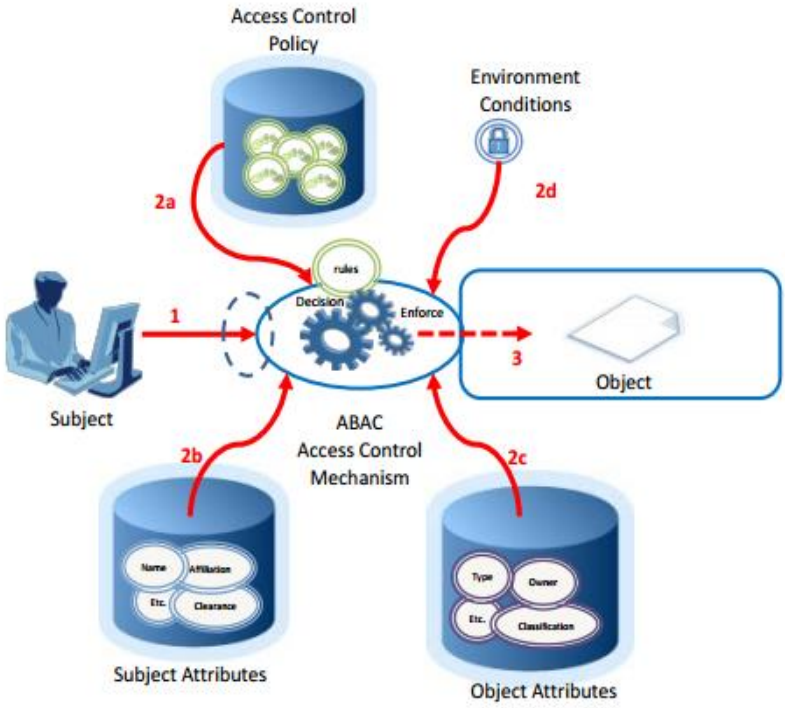


Figure 8: ABAC Basic Mechanism [16]

Attributes: An attribute is a function which takes user or subject as an entity. It is a building block to define access control rules and to describe access. Access is granted to users with the help of policies. These policies combine the attributes together. Attributes are evaluated against the policies and access to objects are granted based on the evaluation [16].

Subject: The subject is an entity that sends a request to perform an operation. Which is mostly referred to as humans. As they are the one who creates to perform some actions in the system. It can also be referred as the requestor. Sometimes a non-person entity (NPE) can also act as a subject. A non-entity person (NPE) is an autonomous service that performs its operation with authority. Generally, every operation performed by the computer is either done by a person (human) or NPE is called as the subject. Which requests access to an object [16].

Subject Attributes: These are the attributes associated with the subjects or concerning the person or actor being evaluated. Each subject is assigned with a finite set of subject attribute functions. Subject attributes are created by the user and constrained with different policies for security architects. This can be explained as the value of the subject attributes can be inherited from its corresponding user [16].

Rules: Rules are the constraints. Based on the evaluation of functions it returns either true or false. Constraints are configured through policy languages. One can apply a constraint during the creation of subject and object, and also during the modification of subject and object [16].

Enforce: After evaluating rule relationship ABAC enforce relationships between attributes and environmental conditions. During enforcement flexibility in secure sharing and protecting information of specific resources in ABAC is been maintained [16].

Object: An object is the resource of the system that needs to be protected, for which the access control mechanism is managed by ABAC. They can be created by a subject on behalf of its user [16].

Object Attributes: Object attributes can also be referred to as resource attributes, which are attributes of the target or object being affected. While creating an object, values for its attributes may be set by the user through the subject [16].

2.5.1 ABAC MetaModel

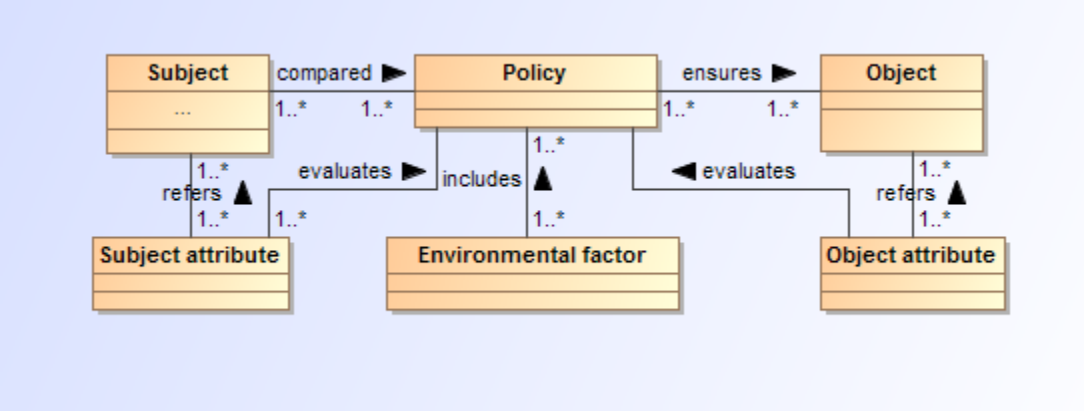


Figure 9: ABAC Metamodel

When the subject is requesting access to the object, subject attributes are compared with the access control policy along with environmental conditions. Then it ensures if the subject can access the requested object.

Subject: Subject or user class is characterized by its own attributes called as its identity. The subject class takes action on a resource. Each subject attribute defines the character of the subject. Such attributes are name, organization, title, project, and so on. Here subject's role is considered to be an attribute [20].

Subject attribute: It is defined as a characteristic of the subject it can be a computing device through which a subject is requesting access. Examples, where attributes of IT component are used in making access control decisions are a user's access to online banking account, is restricted to a pre-registered mobile device or an organization's policy allow VPN access to employees from a device registered in its domain

Environmental Condition: Environmental condition class is the inputs derived from the external factors. This can also be ignored in access control policies. These describe the operational, situational and technical environment with its environmental attributes. Such as date and time [20].

Policy: Whenever the objects are created, its policy rule of what subject attribute can access the object will be derived at first. So when the subject is trying to access the particular object in the form of attributes derived from subject and object, subject attribute matches with the existing policies. When the match evaluates to true access is granted through access control policy class [20].

Object: Object or resource class is also characterized by its own attributes called as its identity. Object class allows the access of subject when policy evaluates to true. An object attribute is often extracted from the metadata of the resource. For example, object attributes can be the title, subject, author, project, etc., [20].

Object attribute: It is defined as a characteristic of the object. Which can be a resource component a device, A device here is defined as a computing device through which the access request of the subject access the resource. Sufficient knowledge of the information assurance capabilities and security robustness of a computing platform, as well as the risk associated with the environment in which it resides, will be required to determine the security risk of allowing access from that computing platform to the specific resource requested.

2.5.2 ABAC Example

Let's consider our Study Information System of the University of Tartu, which provides course related information to the student and the University staffs in which consider that it is implemented with ABAC. When SIS is live with ABAC implementation consider the following scenario as an example of ABAC implementation. This SIS system has students from a different department, they can be considered as Subject. Every Subject has its own attributes called subject attributes. Likewise, objects in the SIS has object attributes and environmental condition attributes. Object attributes can be any objects which the student or staff can able to access, can be a subject information, exam information, personal data, academics data, academics result, class data, etc., Environmental attributes can be a date, time or any other external factors other than subject and object. Which acts as a constraint in accessing different resources.

In our considered example of ABAC implemented the system, the student as a Subject, who has Subject attributes as software engineering department, with active student status is trying to access

the Object of the list of students in his department, which also has an Object attribute as software engineering department. Environmental condition can be the ongoing current academic year.

So here the policy can be defined as follows with a subject attribute, object attribute and environmental conditions like a subject which is from software engineering department with active student status can only access the object with attribute software engineering department under the environmental condition as a current active academic year. Hence for those subjects whichever has the respective attributes as described in the policy can only access its assigned object.

If the subject has to access the list of students from a different department (different object) like computer science instead of software engineering, the same subject has to have its respective attribute as computer science department. Only then according to the newly assigned policy this object would be accessible. This can be clearly depicted in the below diagram in which terms are mentioned very general for a better understanding of this concept.

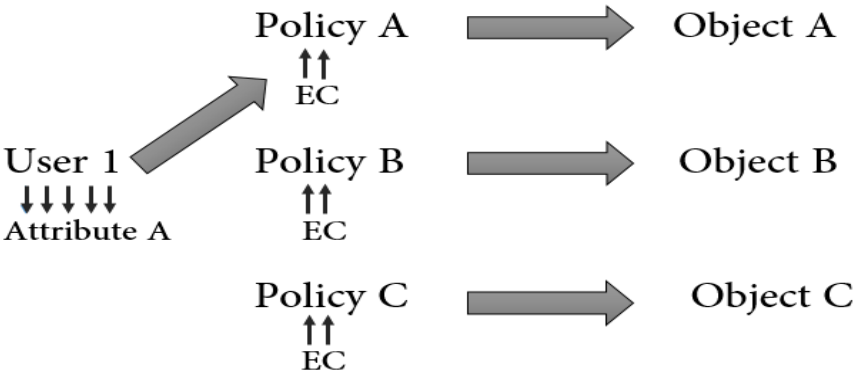


Figure 10: ABAC Example

2.6 Summary

In this chapter we discussed the important access control models in detail. From those models, we took three most important Risk-Adaptive Access Control, Role based access control, attribute based access control for further research. We also described its metamodels with corresponding examples. For those examples, we considered Study Information System from the University of Tartu. Hence this chapter leads to the further research in identifying the main building blocks of an access control model.

3 Comparison of Access Control Model

Aim of this section is the result of finding an answer to the below question

Q: How do we define comparison criterias for an access control model?

This section defines comparison criterias based on the fact that it should be common in every accesss control model, at the same time, it has its own unique key concept which will be different or similar that other access control model. Based on that an access control model can be divided in to three main components **Subject, Policy and Object**. Subsections below will be brief about these three main components.

3.1 Comparison of Subject

In order to understand the subject context, will start with a question: **"How to identify a component as a subject component?**" for this question, first of all, will refresh the subject term, what a subject is. *The subject is the one who initiates the flow of access control request*. Whichever the component associated with the component requests for access is called as subject component and all the subject component comes under the subject boundary context.

After that the subject components are identified, now we need to know the answer to another question: **How does a subject help to compare access control model?** As you see from the access control models discussed, the subject is not the same in all aspects. The boundary of the subject and its behavior differs according to its access control model.

For instance, in RBAC, User and Role are together called as the subject. So, User is the one who requests for access, as the user is connected to the role, based upon the role the access is restricted with the help of permission. So, the subject boundary of RBAC ends at Role.

In ABAC, Subject and Subject attribute are called as Subject. Here when the subject requests access, its attributes are evaluated by the policy. The subject boundary of ABAC ends at subject attribute.

RAdAC's [4] subject boundary is little bigger than other access control model. As it has to calculate the risk, Situational factor and heuristic are included in the subject boundary. with subject and subject attribute. If it is usual access control model only subject and its attributes will be enough to be called as the subject. As it is RAdAC[4], due to the need of risk calculation it considers situational factor and Heuristic as the subject. Along with it, Access authority is also included in Subject boundary because access authority can override an access decision under the specific condition when there is a need to attest to a subject's need. Hence in RAdAC[4] Subject, Subject attribute, Situational factor, Heuristic and Access authority are included in the Subject context. Coming back to the research question, it can be answered as from the above discussion it is clearly known that subject is not the same in various access control model. One can understand it by comparing the subject context boundary of various access control model(For example with RBAC, ABAC and RAdAC[4]). By comparing, it can be found that which of the subject component suits well and efficient for the environment we consider. Also if we need to add any other factor from other access control model that may seem useful in our subject context.

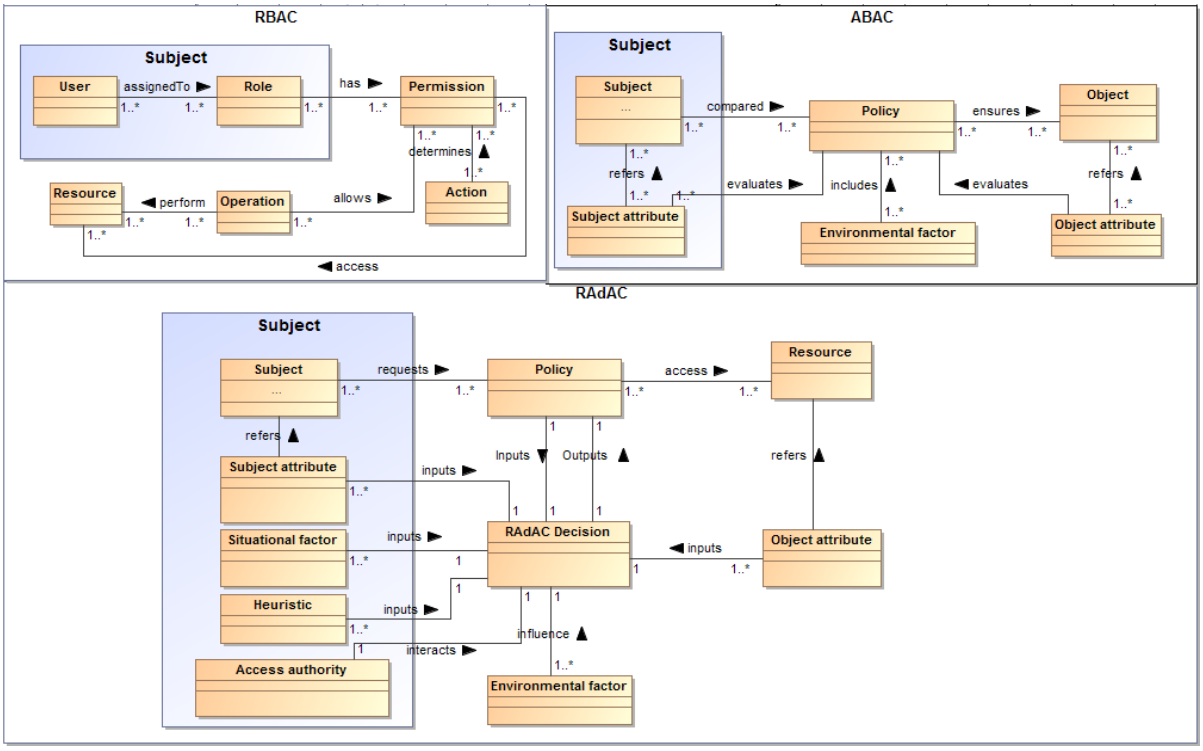


Figure 11: Subject Comparison

3.2 Comparison of Policy

Comparison of policy will start with a research question RQ:"How to identify a component as policy component?" The answer is simple, can be explained as follows. The policy is the condition or a rule that evaluates the access request. It restricts the access request of a subject, which is not authorized to access the object. If a component helps in filtering the access requests that seems to act as a condition or a rule. So this way it is easy to identify if it is a policy component. A policy also has the following attributes: A scope, mechanism, an action, and a triggering event or condition, the rule set of policies must be centrally defined. As the question for first research question is obtained. Now will find out, RQ: "How does the policy help to compare access control model?" for this analysis, let us start figuring out the policy and its behavior of three different access control model one by one. In RBAC access control requests are filtered out with the help of Permission. Policy component of RBAC has two entities Permission and Action. Whereas the permissions available to the role are the union of actions and operations. Action determines what to perform with the operation. So what the user can do is determined by this Action entity. Whatever the action is allowed by the user to perform are listed as Security Action (Fig). This security action is mapped to suit any actions of real-time. For example, if there is a user who has to give permission for copying a file, this will be achieved through security action Select (Ref Fig.). Permissions are assigned to roles, Single permission can be assigned too many roles, and single role can have much permission. So this is about the policy context of RBAC.In ABAC policy is also called as rules. Rules are the constraints. Which evaluates Subject attributes and object attributes. ABAC policy uses certain keywords it can be of our own choice based on the implementation logic. Use of keywords is to evaluate attributes effectively to fulfill the ultimate

goal. So with the help of keywords, it is possible to combine two or many attributes to create a policy in a more customizable way. This provides a way to create policies towards the targeted subject or object. Normally policies are expressed in natural language, machine understandable format. Policy representation is still an active topic of research. So far policies are used to represent in Rei, XACML, OWL formats [25]. Policy representation is chosen based on the flexible construct a language provides for the actual need. So, this is about the policy context of RBAC. The key concept of RAdAC lies in the dynamic management of policies, as administrator changes the limit of security risk and operational need. Policies, used by both the Security Risk Function and the Operational Need Function, may consist of simple if/then statements. Administrators manages rules in the policy system. Network server makes policy decisions in response to a request from a user wanting to access a resource[28].

This helps to derive the answer for the RQ, which is policy is also not the same in different access control model. This can be understood by comparing the policy context of different access control model. After the discussion of policies of three different access control model (RBAC, ABAC & RAdAC) it is easy to understand that policies of the different access control model are not the same and each has its own merits and demerits. Based on our needs, we can compare just like above and understand its merits from a different model and employ those in the new paradigm which we are going to construct. For instance, policy context of any access control model can be replaced by desired and suitable policy context for the considered environment. Alternatively, even other factors can also be added if felt that would be helpful.

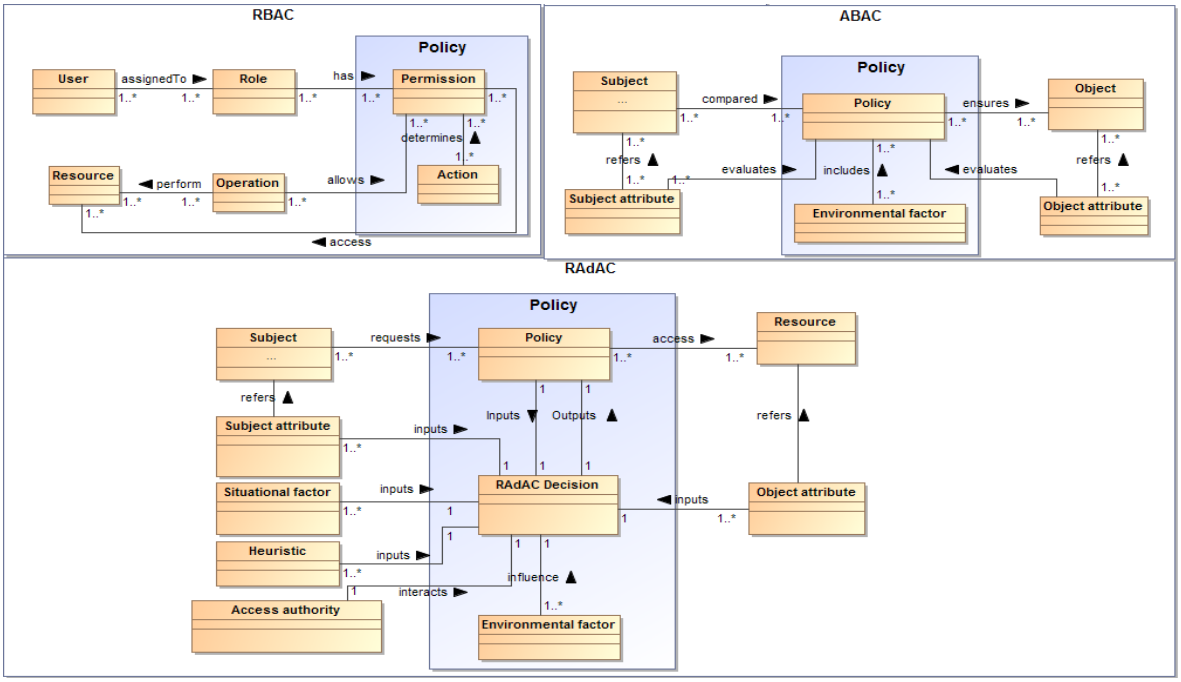


Figure 12: Policy Comparison

3.3 Comparison of Object

To discuss the object comparison and its behaviors, will start with the same as other RQ:"How to identify a component as object component?". There can be many objects in a system. The object is a target of every subject in which the allowed subjects are given a chance to perform permitted actions in their place. The object is one of the purposes that access control model has a purpose. As the answer to the first research question is already found, there is a more RQ that has to be answered, RQ: "How does the object help to compare access control model?". The object of different access control model shares the same purpose, but the form of representation may differ within different access control model. Hence by comparing it helps us to know about its strong and weak features between the objects of different access control model. Thereby we can add any concepts within its boundary if we found it as beneficial.

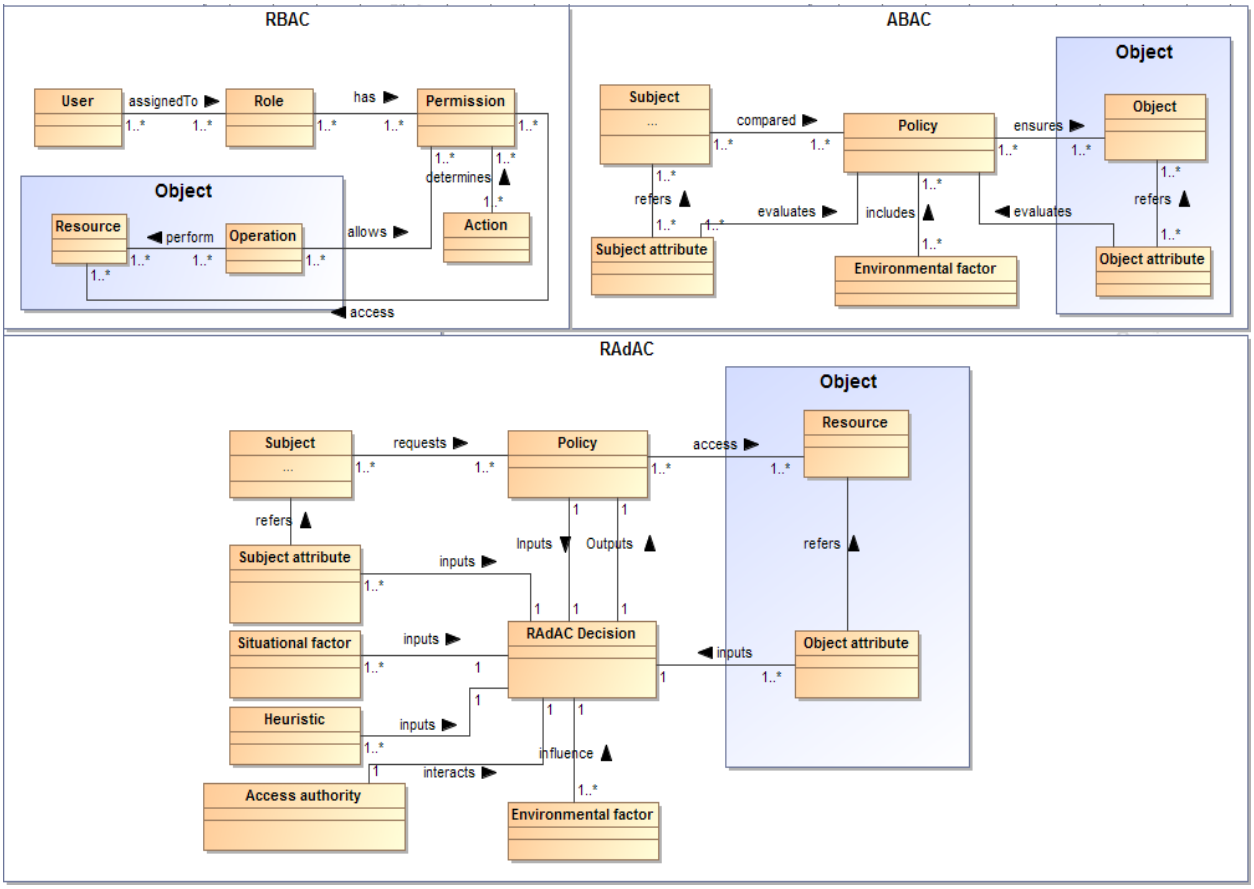


Figure 13: Object Comparison

3.4 Summary

This chapter provides the comparison criteria of an access control model, and it shows how the three essential models are compared with that criteria. This chapter which says how different access control models can be compared in a similar way

4 Survey Analysis and Findings

Objective of this section is the result of finding an answer to the below question

Q: How can anyone understand an access control model with the help of three building blocks?

This section acts as a proof of understanding the concepts of any access control model with the help of three main components **(Subject, Policy & Object).** We introduced three access control models for this survey, they are RBAC [12], ABAC [16], and RAdAC [4] to conceptualize and to help to compare the strong and weak features based upon the response of 53 audiences. This is done as a systematic survey by explaining the underlying principles, meta-models, examples to the audience group and by capturing their experience with it. Intention is to make empirical comparison for selected access control model by conduction a survey explaining the audience about three main components of an access control model which is Subject, Policy, and Object

4.1 Problem Statement

There exists many access control model. To implement the right access control model for our systems Firstly one needs to know its strong and weak factors also one has to understand the policy standards for a different model and if that is suitable for the current system in which it is going to be implemented. So that the resource can be used efficiently and securely. These days with the advancement of science and technology it is not easy to figure out the suitable access control model that matches the system. To compare its core component to match the system in need, this analysis is done Goal

4.2 Goal

The main goal of this survey is to make the audience understand these three main access control model with the help of three building blocks. This helps to validate and generalize the comparison criteria of different access control model.

4.3 Experiment Planning

This is done as a systematic survey by explaining the underlying principles, meta-models, examples of different access control models to the audience and by capturing their experience with it. There by it outputs the real-time experience of the different audience. Which is the one of the possible solution to find out the strong and weak characteristics of different access control models Analytical comparison of different access control models is drawn from a report of how the audience deals with it at different cases that have to be analyzed. This way gives the opinion of different people in a realistic way.

4.4 Experiment Operation

The survey was done with a group of 53 students from the University of Tartu. Have presented three access control models (RBAC [12], ABAC [16] & RAdAC [4]) by focusing on the basic components of **Subject, Object, and Policy**. This presentation session was followed by a task and questionnaire session. The audience had 3 tasks for three different access control models in which they were given an example model and asked to find out its Subjects, Objects, Policies and to run a simulation in that model representation. Through our tasks audience easily understood the main

concepts of this three access control model. But the same time some places in the task lacked some logical representation which made the audience to go wrong with the answers.

4.5 Data Analysis

Have received 53 responses for the task. Some of the responses are mixed responses while others are very particular. Table 5 shows the response count for the various category. Hence have to eliminate the components which show the minimum count which is marked in gray color also in the table at the interpretation of results it is been mentioned clearly about total minimum count as a response.

Ease of Learning: It tells how quickly the system can be learned by the various groups of users when access control models are compared. This quality helps for easy memorability. Which provides the ease of adoption from one model to the another if one model is been followed from time to time

Task efficiency: It tells how quickly the system can do a task efficiently when access control models are compared. This quality helps to know the performance of access control model when compared. It shows the ability of a model to achieve its performance.

Ease of remembering: It tells how quickly the system can be remembered for the occasional user when access control models are compared. This quality helps the user or administrator to remember its concepts and predict the outcome of the actual model. Hence this helps to know if the model of the system is functioning as designed.

Subjective satisfaction: It tells how quickly the system adapts to satisfy user needs when access control models are compared. This quality helps the user to know about the system if it adapts to the varying needs from time to time under certain conditions.

Understandability: It tells how easy is to understand the system functionality when access control models are compared. This quality helps the user to understand better about the model. So that the user can know under which situation they would be in need of such model.

Table 5. Questionnaire response count

	Access Control Model		
	RBAC	**ABAC**	**RAdAC**
Ease of Learning: How quickly the system can be learned by the various groups of users?			
Which model *Subject* is easy to learn?	36	22	16
Which model *Policy/Permission* is easy to learn?	30	22	14
Which model *Object* is easy to learn?	31	23	18
Which model is easy to learn?	32	20	15
Task efficiency: How quickly the system can do a task efficiently?			
Which model *Subject* is efficient for frequent use?	34	13	15
Which model *Policy/Permission* is efficient for frequent use?	27	21	11
Which model *Object* is efficient for frequent use?	27	17	15
Which model is efficient for frequent use?	30	16	11
Ease of remembering: How quickly the system can be remembered for occasional user?			
Which model *Subject* is easy to remember for occasional use?	24	21	19
Which model *Policy/Permission* is easy to remember for occasional use?	24	17	16
Which model *Object* is easy to remember for occasional use?	23	22	20
Which model is easy to remember for occasional use?	23	20	15
Subjective satisfaction: How quickly the system adapts to satisfy user needs?			
Which model *Subject* is satisfied with most of all occasions?	29	21	6
Which model *Policy/Permission* is satisfied with the system?	29	13	16
Which model *Object* is satisfied with the system?	31	16	13
Which model is satisfied with the system?	35	15	8
Understandability: How easy is to understand the system functionality?			
Which model *Subject* is easy to understand?	28	23	20
Which model *Policy/Permission* is easy to understand?	26	19	21
Which model *Object* is easy to understand?	29	17	20
Which model is easy to understand?	28	19	16

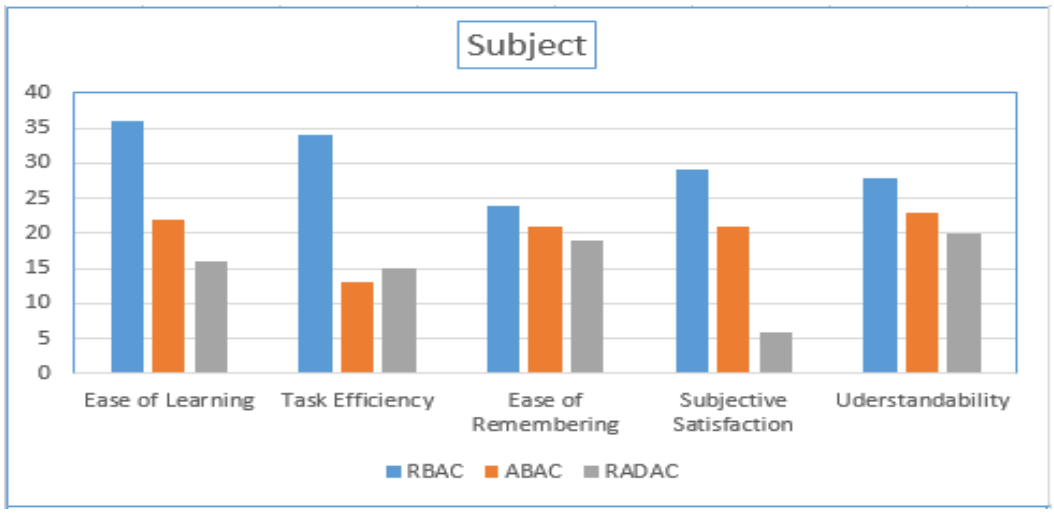


Figure 15: Survey outcome for Subject

Here subjects of three access control model are taken in to account from the survey response and represented in histogram for further analysis. We can see that RBAC ranks highest in all the five category of learning. Among the five-different category (Ease of Learning, Task Efficiency, Ease of Remembering, Subjective Satisfaction, Understandability). Ease of Learning ranks top from audience. Then if we look closer in to it, RBAC's subject outnumbers other access control model subject in Ease of Learning. This could be because the audience had a general idea about RBAC even before this survey session, which affected the survey scale gradually. But on the other case if we try to know the least ranking model from the above representation, RAdAC's subject ranks low in four categories (Ease of Learning, Ease of Remembering, Subjective Satisfaction, Understandability). The reason could be because of risk and operational need factors. ABAC's subject stays in the average scale in four categories (Ease of Learning, Ease of Remembering, Subjective Satisfaction, Understandability). This is possible be ABAC takes the Role from RABC and considers it as a subject attribute.

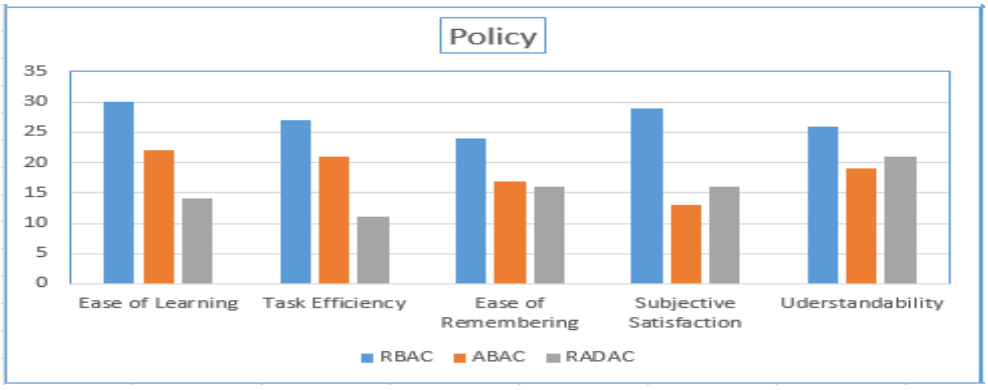


Figure 16: Survey outcome for Policy

Here the survey response of policy from three different access control models (RBAC [24], ABAC [16] & RAdAC [4]) is represented in the above analysis. Here we can see that RBAC ranks highest in all the five categories of learning. Among the five different categories (Ease of Learning, Task Efficiency, Ease of Remembering, Subjective Satisfaction, Understandability). Ease of Learning ranks top from the audience. Then if we look closer into it, RBAC's policy, it outnumbers other access control model policy concepts in

Ease of Learning. This could be because the audience had a general idea about RBAC even before this survey session, which affected the survey scale gradually. But in the other case, if we try to know the least ranking model from the above representation, RAdAC's policy ranks low in four categories (Ease of Learning, Ease of Remembering, Subjective Satisfaction, Understandability). The reason could be because of risk and operational need factors. ABAC's policy stays in the average scale in four categories (Ease of Learning, Ease of Remembering, Subjective Satisfaction, Understandability). This is possible because ABAC policy only evaluates both subject and object attributes.

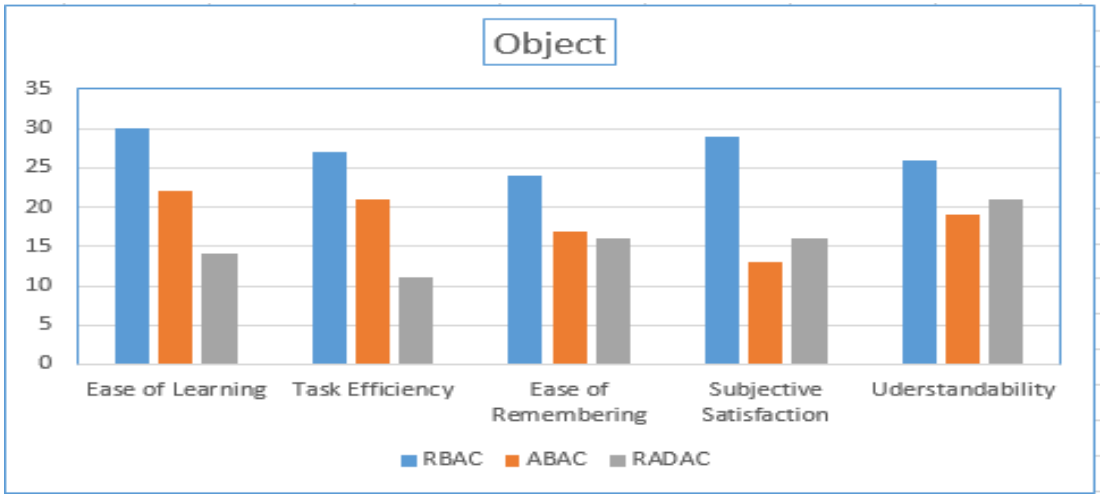


Figure 17: Survey outcome for Object

From this above derived graphical representation, RBAC has most of the audience response as usual. RBAC object ranks first in the section Ease of Learning. Which shows how quickly objects can be learned by the various groups of users. Lowest of them all is ranked by RAdAC object. The same RBAC that is been in a higher rank in all three-context scored low in Ease of Remembering with total no of the response of 24 (among RBAC). Hence the audience couldn't remember its object compared to other four qualities.

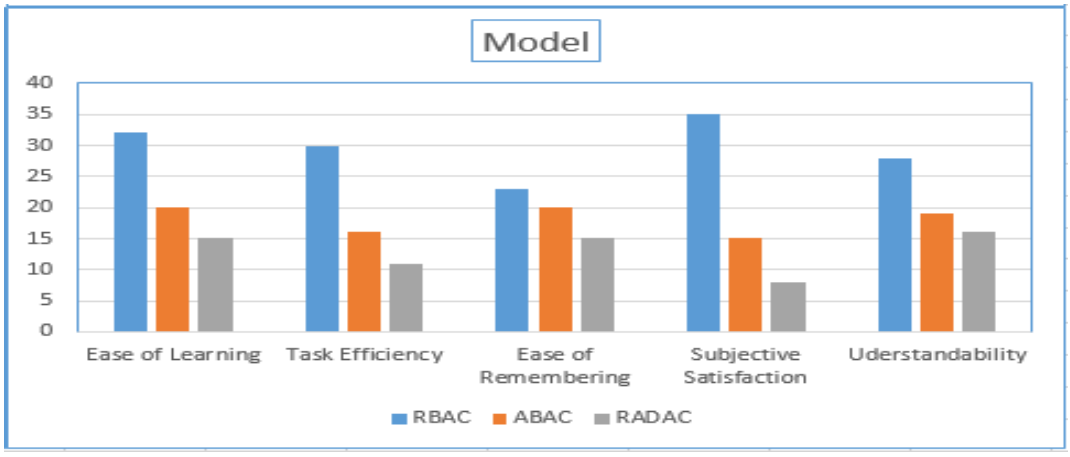


Figure 18: Survey outcome for Model

This histogram for model proves again that RBAC is the model that ranked first with the audience and majority of the audience considered that RBAC model is been chosen by most of the people for subjective satisfaction, that means RBAC model adopts to satisfy user needs quickly than the other two models compared. If we take ABAC model and see why can't it be the best in subjective satisfaction, satisfying the user need quickly is not the main purpose? As models like ABAC & RAdAC are designed to satisfy administrator needs, it is no wonder that RBAC model has become the audience choice.

4.6 Interpretation of Results

Results can be interpreted as the below table from the total responses received. By considering the above table all grayed out columns are counted to display the minimum points received from the audience.

Table 6. Concepts to be eliminated

	RBAC	ABAC	RAdAC
Subject	0	1	4
Policy	0	2	3
Object	0	1	4
Model	0	0	5

Hence from this result based upon the audience response collected it is been evident that RBAC model is convenient for the whole group to follow and figure out and it made them feel efficient than other access control models.

4.7 Conclusion

Upon observation of results processed from the audience response, it can be concluded that RAdAC is not convenient for most of the audience though it is claimed to be an advance access control method. It is being proved that RBAC is a simple classic of all times based upon the audience responses. Apart from this result what matters most was that the audience enjoyed learning different access control model with the help of three main building blocks

4.8 Summary

In this chapter, we validated our three main building blocks (subject, policy and object) with a group of audience. This is to see if the audience can understand the access control model based upon the building blocks and the resulted survey shows the level of understanding of this audience group is more inclined towards RBAC. Also, the audience group were asked to finish up three different task exercises(Appendix) to show their understanding level. In the end, our survey results were positive.

5 Unified Meta Model

Objective of this section is the result of finding an answer for the below question

Q: How the conceptual model is constructed in to a unified model?

The Unified metamodel is the model built after the comparison of main components of three main access control models. All those concepts are discussed above are combined and made in to a unified model as follows.

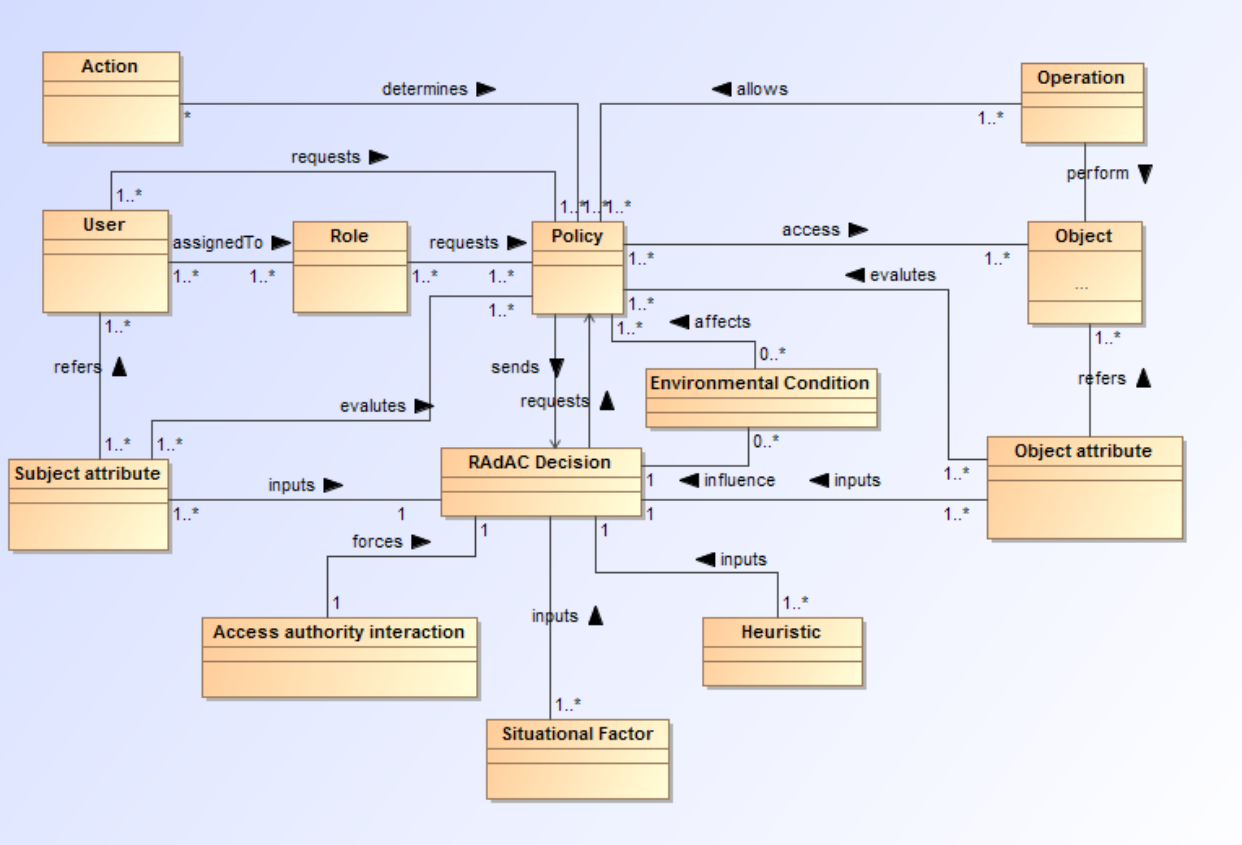


Figure 14: Unified Access Control Model

Subject: Users can also be subjects, they request for access to perform necessary actions on the object. Their characteristics are termed as attributes. Sometimes when granting access it is necessary to validate user attributes.

Subject Attribute: Attributes associated to the subject. It is defined as a characteristic of subject it can be a computing device through which a subject is requesting access. Examples where attributes of IT component are used in making access control decisions are a user's access to on-line banking account is restricted to a pre-registered mobile device or an organization's policy allow VPN access to employees from a device registered in its domain

Role: In organization users are assigned to roles according to their nature of work. One user can be assigned multiple roles.

Policy: Policies are essential plans of action which guide decisions so that we might arrive at logical outcomes. They enable administrators to control and evaluate who can access information, how long to retain information and how effectively individuals are complying with the policies themselves. Policy-based management aims to support dynamic adaptability of behavior by changing policy during the process. Place where evaluation of policies are done. Considered factors or attributes are evaluated against the predefined rules. For those which are passed will be granted access. Predefined rules are the policies defined as certain threshold, which is evaluated against the factors to determine if access can be granted.

RAdAC Decision: Policy decision point is a component of a policy based access control that makes to determine the access control requests based upon the available information and security policies

Object: Object is the resource, which every user wants to access. But only those which passes the policy validations can access it. Sometimes even the object attributes plays a vital role in terms of determining the access condition.

Object Attribute: It is defined as a characteristic of the object. Which can be a resource component a device, A device here is defined as a computing device through which the access request of the subject access the resource.

*Action Type: A*ction represents how it performs an operation. There are four main actions, which are inserted, update, select, delete. If there are any other actions requested, it should fall under in any of this main four actions.

Operation: Operation is some tasks, which represents what it does with the help of actions. Now let's discuss the main functionalities or specific behavior of an access control model over another to see how does it suit well while drawing conclusion for a unified model

Environmental Condition: Environmental condition is the same as factors associated with the environment surrounding the transaction. It analyses the increased likelihood of how the surrounding environment can be exploited during the transaction. Its purpose is to assess the risk category of the transaction with regards to the environment. Factors considered for assessment are the current location of the requesting user and the data source [4].

Access Authority Interaction: Human intervention is allowed to change access policies and to address the operational need for a category of users under certain conditions. Access authority is an authoritative external source, which can be an administrator or a supervisor of approving authority to override security risk when necessary. Through this RAdAC allows operational need to override the measured security risk [6].

Situational Factor: In situational factors security risk lies in the surrounding situation when information transaction happens. Risk factors to be considered here are specific mission role of the user who requests access, time sensitivity of the information, transaction type, auditable or non-auditable, audience size. Its purpose is to identify the security risk that revolves only around the transaction, not the associated data or the user [4]

Heuristic: This helps to improve access control policies to grant decisions. Like, various access decisions in the past help to improve the decisions. This is used to find the risk category in the transaction based on similar transactions that have occurred before. It helps to fine-tune access control policies, improve future access decisions and to predict the future behavior. Risk knowledge and trust level are the factors help to assess the risk factor of heuristics [4].

5.1 Summary

This chapter shows the construction of conceptual model to a unified model in the form of meta model. Hence from this chapter we can say that the identifying main building block of an access control model helps to align the best concepts to form a best suitable unified model.

6 Model for Validation

The goal of this validation chapter is to get an answer to the question:

Q: How can we verify our constructed unified model?

First of all, we will start answering with a small introduction about [24]. Which will also include a brief reason for why we are considering this model [24] to validate our proposed model. Followed by, validation of three main components. In order to validate the unified metamodel completely, validation is done by comparing each building blocks of unified model, building blocks are the representation of three main components in this paper.

6.1 Unified Metamodel for Validation

Model to be validated against the proposed unified model is taken from [24]. This model [24] addresses the **challenge in the flexibility** of making unified conceptual modeling scenarios of authorization according to well-known access control models. This study also presents a number of existing access control models in terms of conceptual modeling then proposes a unified modeling. Mainly it illustrates four different ways of modeling authorization using DAC, BLP and Biba, Brewer-Nash (Chinese wall), RBAC, ABAC, RAdAC and TBAC [24].

We consider this model [24] because it was designed to **support enterprise architecture** for the management of IT. This model also supports the variety of individual models used these days. As a unified model, it is capable of expressing its configurations in a flexible manner.

The first model that takes into consideration is Discretionary Access Control (DAC) in which this model is based on the identity of the subject and access rule stating what the subject is allowed to do. Its example is a typical windows system [24]. Then the other model of consideration is Mandatory Access Control (MAC) which is synonymous with the term lattice-based access control as its security levels are structured as a lattice. The Bell-LaPadula (BLP) model [24] and Biba model [24] regulates access to objects like DAC, and security labels denotes security levels for classification of objects and clearance of subjects. Both models consider two modes of access reading and modification. Biba additionally considers invocation which can consequentially be viewed as a modification under the invoked subject's clearance. However, while BLP addresses confidentiality, Biba addresses integrity.

In BLP, reading an object is allowed to a subject if the subject's clearance is equal or higher than the object's classification, and writing in an object is allowed if it is equal or lower. In Biba [24], reading an object is allowed if the subject's clearance is equal or lower than the object's classification, and writing (and invocation) in an object is allowed if it is equal or higher [24]. Role-based access control is technically a nondiscretionary model, in which access are granted to subjects based on the roles they are assigned for a specific session. Attribute-based access control is one of the more recent models, its major advantages over DAC [24], MAC [24], and RBAC [24], are far greater expressiveness, richness, greater precision [24]. Risk-adaptive and token-based access control have been new in recent times. As it evaluates risk dynamically it is considered to be on top of ABAC [24].

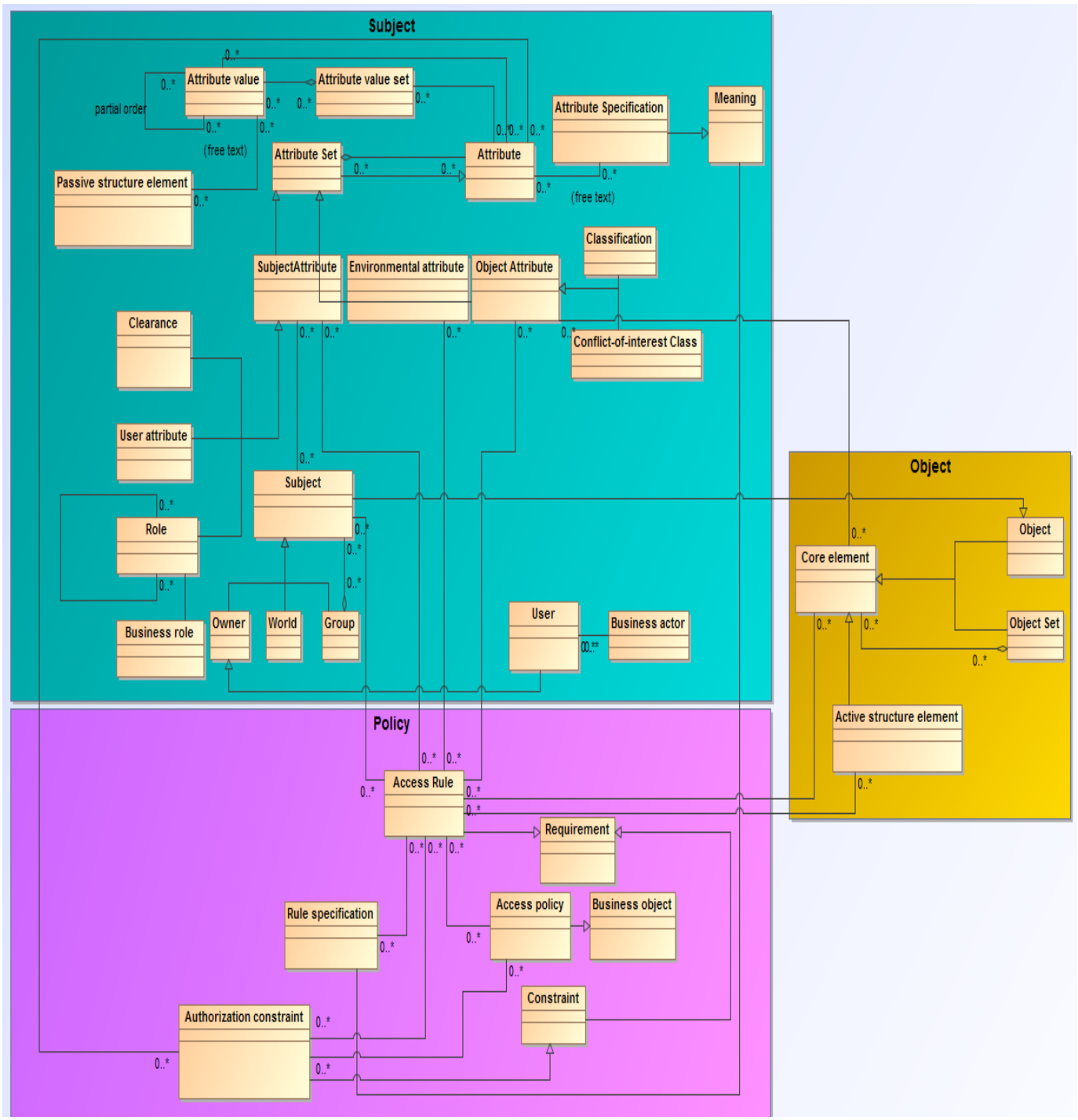


Figure 19: Unified model for modeling authorization [24]

6.2 Validation of Unified Model

6.2.1 Subject Validation

Unified model [24] has more classes for a subject compared to the proposed unified model. Also, it has more details than the proposed unified metamodel in this research. There are three main classes, **Subject attribute**, **Environmental Attribute** and an **Object attribute.** In which subject attribute has been formed as a generalized class for three other classes **Clearance**, **User attribute**, and **Role.** Same way **Object attribute** also formed as a generalized class by introducing two other classes **Classification and Conflict-of-interest Class** as its child class. within the subject boundary subject classes of that unified model are **Subject** is being generalized into three different classes **Owner**, The **World,** and **Group**.

Whereas while talking about the attribute in the subject boundary, class **Attribute Set** is to represent the set of attributes from ABAC and tokens from TBAC. Each Attribute set comprises of its own **Attribute**, hence it is mentioned as aggregation relationship. Other classes of Attribute could have simply modeled as an attribute rather than mentioning implicitly but in order to retain the clarity that it is been derived from its actual model instead of representing it commonly as an attribute. Every **Attribute** has its **Attribute value set** and every **Attribute value set** has its own **Attribute value,** it is represented in a simple association. But the relation between Attribute value and Attribute value set is in composition. All attributes have certain specifications called **Attribute Specification,** which has a different meaning for every attribute specification**.**

When we compare the subject boundary of this model [24] with our proposed unified model, both of its differences and similarities can be seen. First of all, lets start to validate with the similarities, both model has **Subject** which is the main component of this boundary. The subject can be also called as **User**. **Role** and **subject attribute** are the same in both of its unified models. **The situational factor** can be related to being **Environmental attribute**, but in order to preserve its originality better to keep it separate as a **Situational factor**. Apart from other existing classes, our model has two new classes, **Access authority interaction,** and **Heuristic.** Other classes in the model are

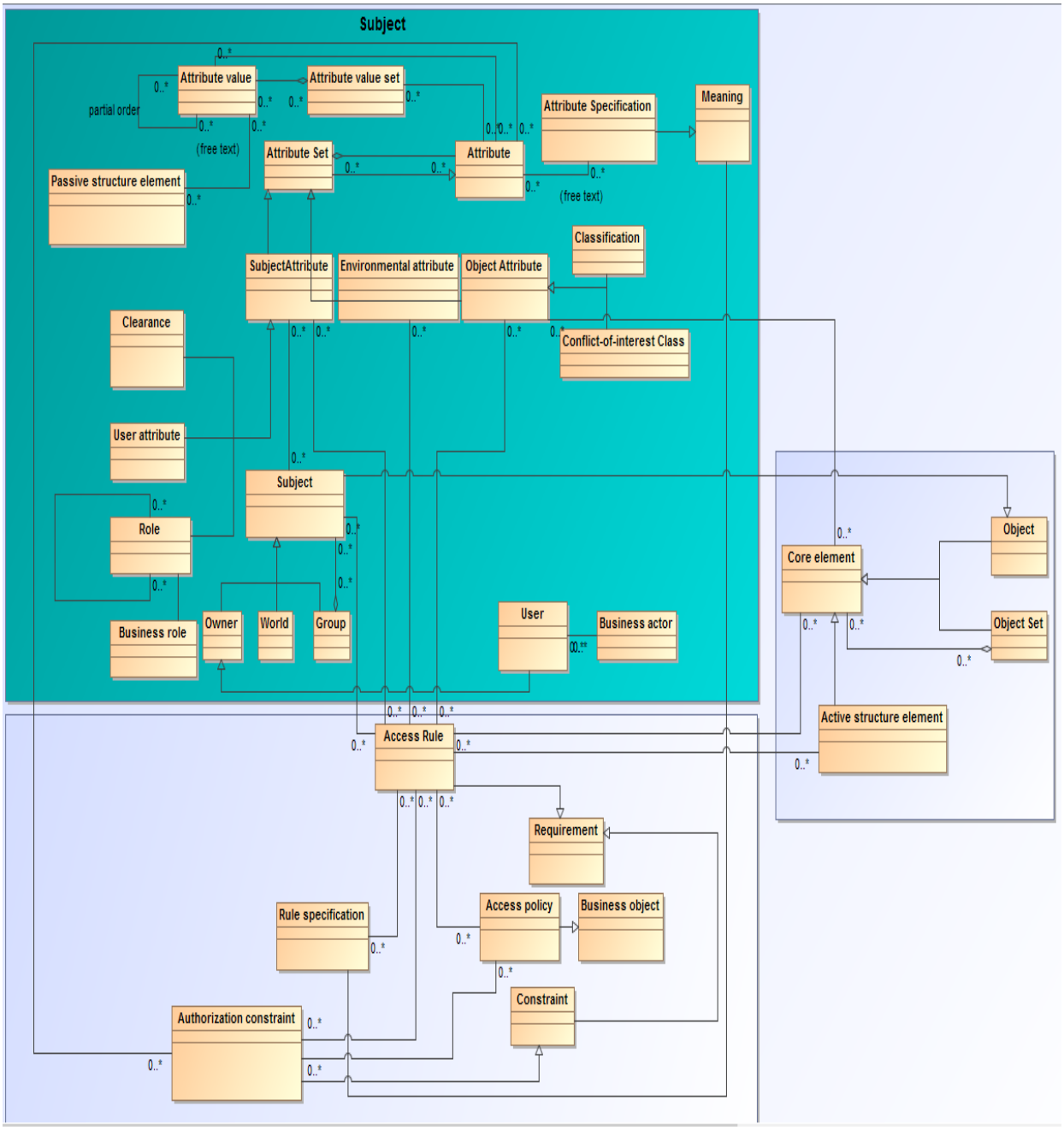


Figure 20: Subject Validation with Korman, M., Langerstorm, R., Ekstedt , M model [24]

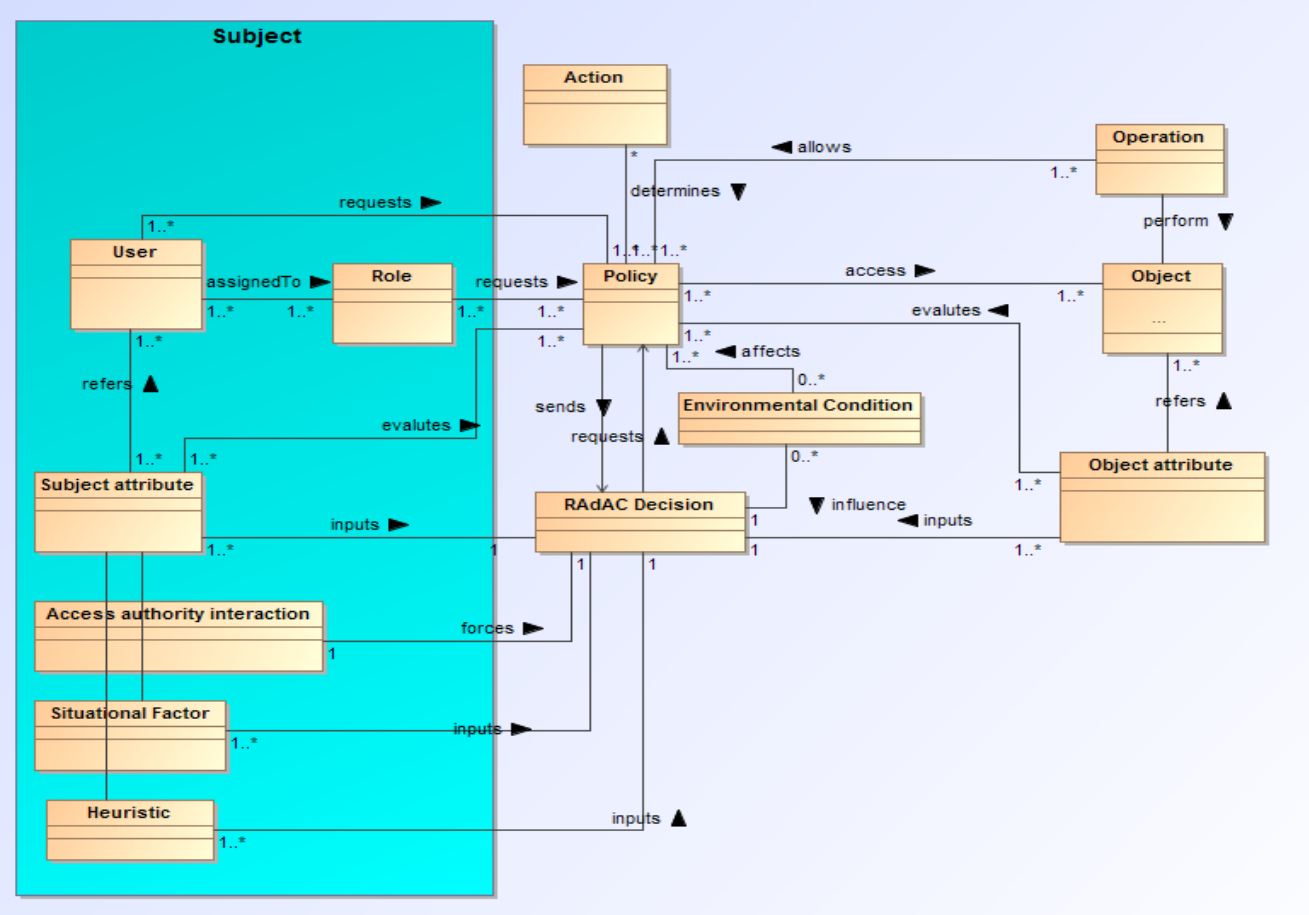


Figure 21: Subject Validation of proposed metamodel

6.2.2 Policy Validation

Policy boundary of the unified model [24] has **Access rule** as the main class, from which **Rule specification**, **Authorization constraint**, **Access policy** and **Requirement** are originated and treated as an entity. These are the classes which support the main entity, Access rule in some specific ways. Also, all these classes mentioned here has been supported by few more additional classes like Constraints, which is derived from the class **Authorization constraint**. Also, **Business object** is from access policy.

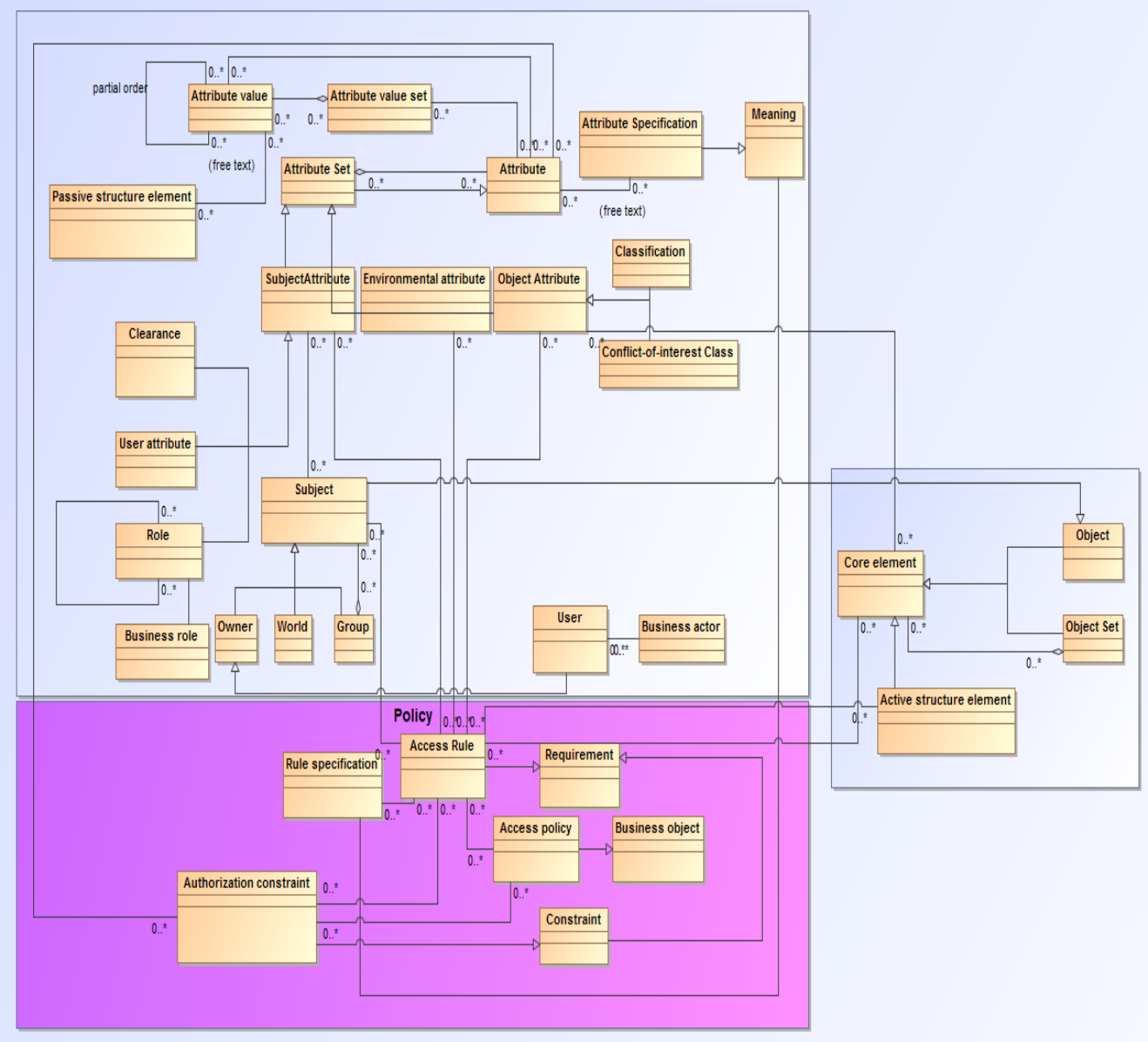


Figure 22: Policy Validation with Korman, M., Langerstorm, R., Ekstedt , M model [24]

Policy validation of constructed unified model against the published unified model [24] can be done by comparing the similarities and dissimilarities of the model against the proposed model in this paper. When we compare the policy boundary of both unified models few things can be noted that it has main class **Access rule** in [24] and Policy in our newly proposed unified model. Other classes in these models have not matched one another, though few may have similar functionalities like **Rule specification, Authorization constraint & Access policy**.

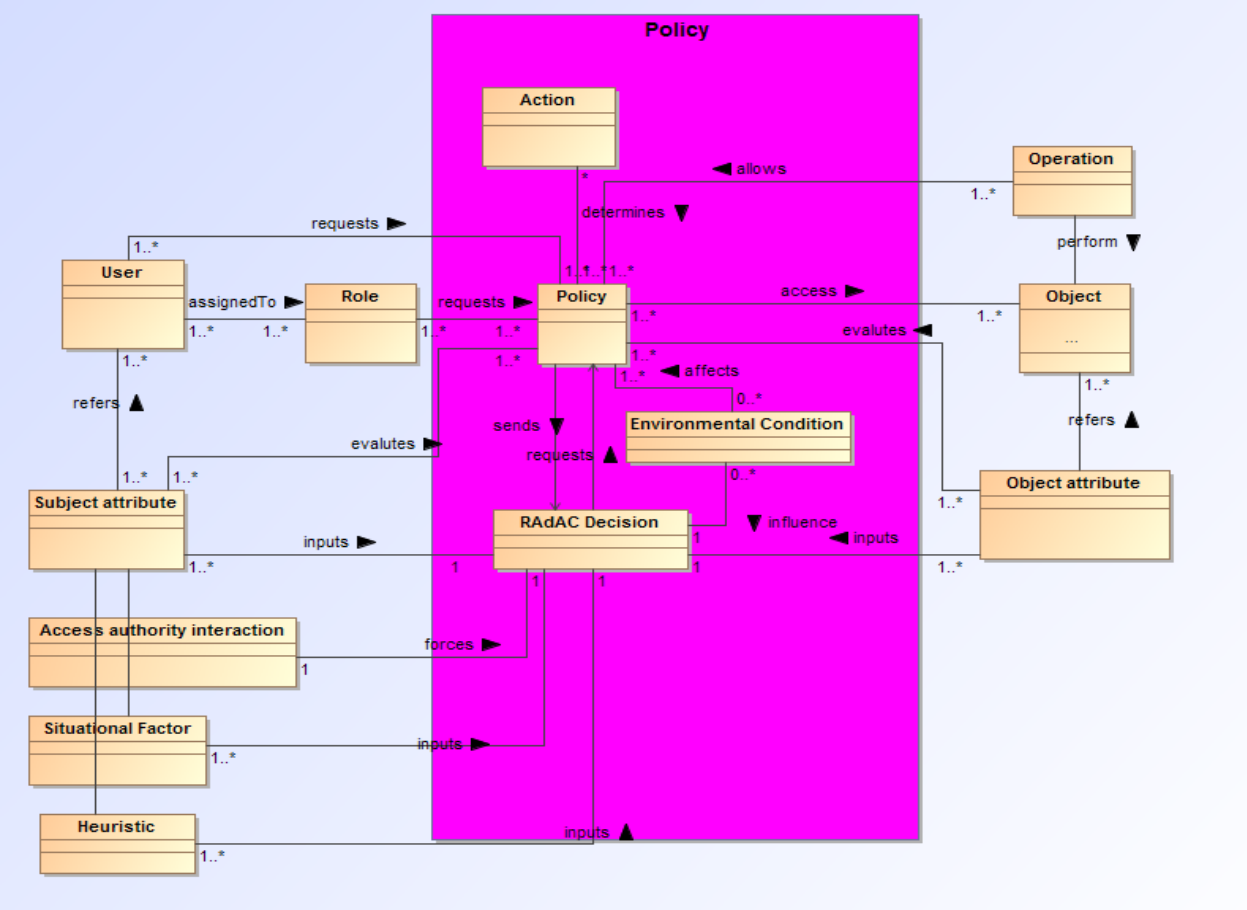


Figure 23: Policy Validation of proposed metamodel

6.2.3 Object Validation

Object validation of constructed unified model against the published unified model [24] is done by comparing the similarities and dissimilarities of both the models. In the unified model of "Modeling Authorization in Enterprise-wide Contexts" [24] main class **Object** is derived from the **Core element,** hence object is the child class of core element class. Core element class also has an aggregation relationship with **Object set** class, which shows that core element is comprised of the object set. Also, the core element class has a child class of **Active structure element** which has an association relationship with Access Rule, the main class of policy boundary.

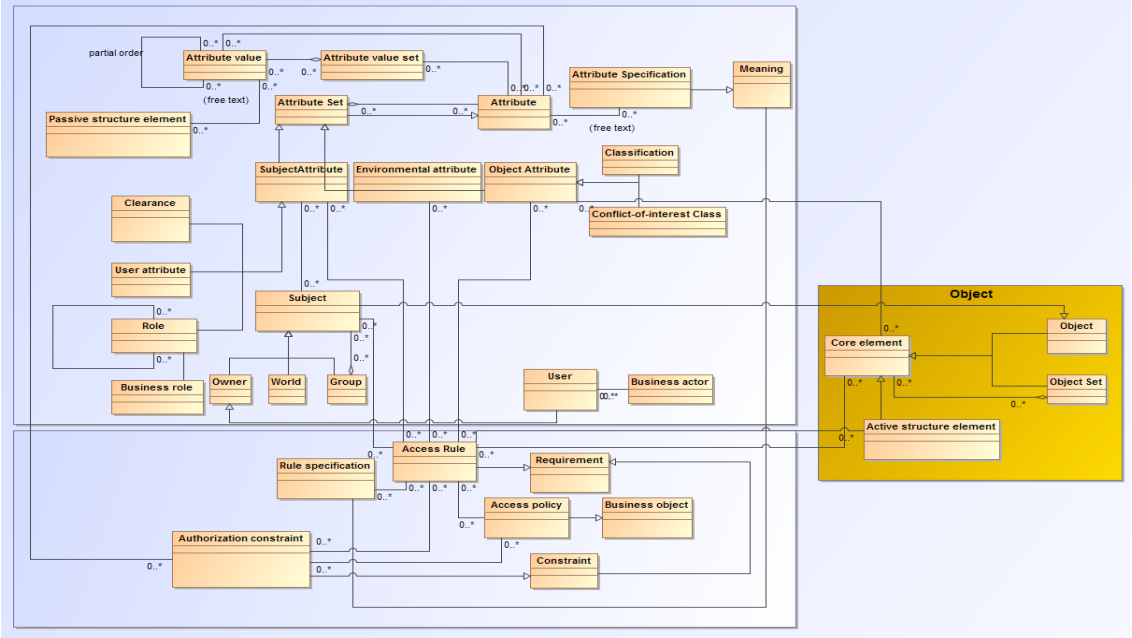


Figure 24: Object Validation with Korman, M., Langerstorm, R., Ekstedt , M model [24]

If we compare the object contexts of both unified model, it can be easily known that this boundary has only a few classes but not similar. Although it does the same job in both the unified model. But the difference is that in our model there is a special class **Operation,** which shows the available list of operations to access the object. Other than that other classes like Object attribute and object are the classes which function same.

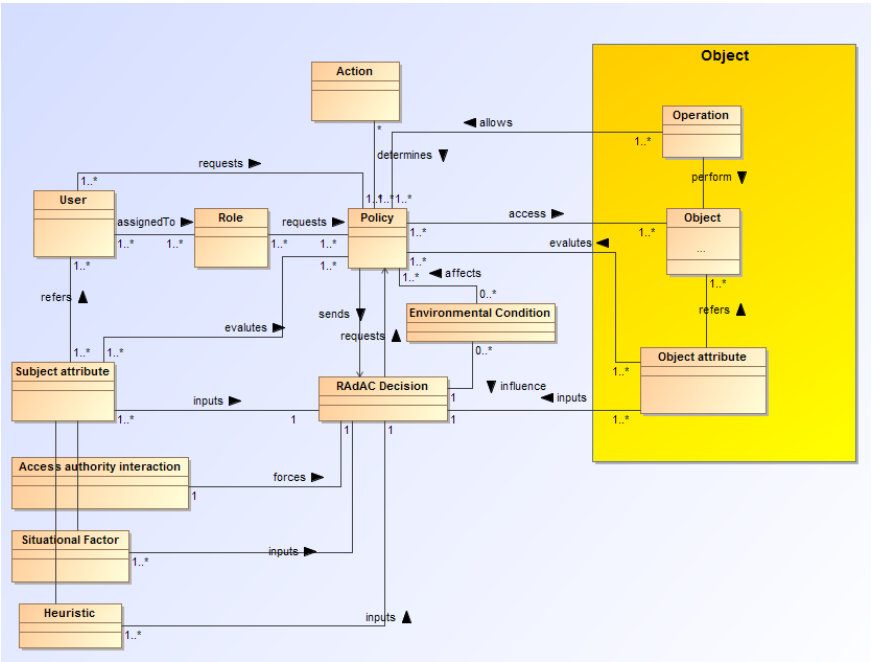


Figure 25: Object Validation of proposed metamodel

In my model, I took three different models as RBAC is of the standard model and most widely used this has been employed extensively from 70's and 80's. Second model ABAC is on the verge of large-scale adoption and the third model RAdAC is of more recent and sophisticated risk-adaptive variants. From these three models, I came with a simplified unified model based on the analysis of underlying principles of three different models. And it addresses the how to find its strong and weak factors. As far as RBAC is concerned strong factor is Role and there is no such weak factor in RBAC which is clearly exposed from the audience responses. In ABAC the strongest factor is its attribute, both subject and object attribute as it tries to include most of all the basic components in general even role can be included in the subject attribute but for the purpose of different access control model and its main concepts to be seen role has been mentioned separately. In RAdAC basically, its risk adaptability feature is considered to be the strongest but in terms of audience response, RAdAC is the one who got very low responses for almost every factor.

6.3 Summary

In this chapter, we validated our unified model with [24]. Our validation shows how our unified model varies and overcome the limitations of [24]. Here we compared the building blocks of both the access control model, can be seen from fig (20 - 25). This comparison helped us to achieve the results as we expected.

7 Example for Verified Model

Objective of the section is the result of finding an answer to the below question

Q: What is the proof that this verified model is consistent and reliable?

The consistency of our verified model can be proved with the help of an example designed as follows

7.1 Process of Usage

This section describes the process flow of how this unified model can be used as an example. Here we can see the importance of the order in which the process executes. The general overview of this process is as follows, the access request is handled by three different models one by one at the same time unless it is either granted or denied.

When the user requests for accessing a resource, based on its role it will look for its permission with the help of RBAC [7] model at first. As RBAC's *permission* [7] cannot be not formalized according to the dynamic aspects certain requests cannot be handled. In that case, the request will be sent to next access control model of our unified model for handling. For this reason, ABAC [16] is placed here so that it can handle the requests with environmental conditions like dates or values. Here ABAC [16] tries to evaluate subject and object attributes with the help of *policy*. Up to this level, most of the requests will be processed. Permissions and Policies in RBAC [7] & ABAC[16] are predefined. Which is not flexible according to the various situational and environmental factors. If any of the factors of the subject is not defined, then the access cannot be handled in RBAC [7] & ABAC [16]. Hence to overcome this situation we included RAdAC [4] model in the third level. When the request triggers RAdAC model, it calculates security risk and operational need based on the real-time factors, and it decides whether the access can be granted or denied.

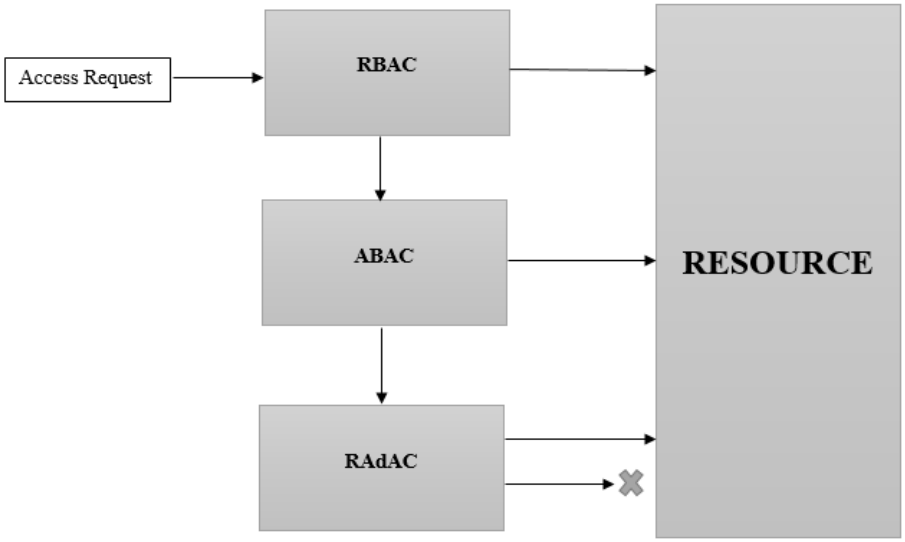


Figure 26: Process of usage representation

The unified model has been already verified in the above sections. Mix up of three models, first RBAC [7], second ABAC [16] and finally RAdAC[4], so the access request first goes through RBAC[7] and the assigned job functions will work according to different roles. Three diagrams of Access request RBAC[7], ABAC[16] & RAdAC[4]. First, the requests will be going through this process so that it will first the process check will happen based on roles, as through this it will be easy to handle the access requests. if there is no permission through the pre-assigned permission for the particular role for which the request is been given.

Different access control models combined to form a unified model. Unified model can be accomplished in any number of ways. But in order to bring out the best and flexible unified model which serves almost every purpose for an enterprise architecture. In order to use this unified model efficiently in an enterprise context.

7.2 Example

An example of our unified model will be based on the process explained in section 7.1. Our unified model provides more flexibility and it overcomes the limitations of traditional and recent access control model.

Q: What are the limitations this unified model overcome?

This model overcomes the limitation of RBAC, 'role explosion' [30], which can be explained through an example of a large enterprise organization: If one employee requires many applications or services with two roles, then the number of roles being managed for employees in an organization will be many. Also, it is not wise to implement RBAC for all users, as authorizations are based on the roles rather than skill sets which define the specialty of an employee.

Though ABAC [16] helps to overcome the limitations of RBAC [7] with the help of *attributes*, in the other way it introduces a limitation of handling its own attributes and the effort of defining policies. Also, ABAC [16] has to manage the full set of access rules, which is not easier than defining and managing roles in RBAC [7]. Hence the ultimate solution can only be obtained by combining RBAC [7], ABAC [16] & RAdAC [4] to synergize the advantages of each and to overcome its disadvantages by one another.

Considering all these limitations there was always a need for a 'newer' and 'better' access control solution. This solution is provided by our unified access control model. To understand the purpose of our unified model here is an example with the help of Study Information System [18].

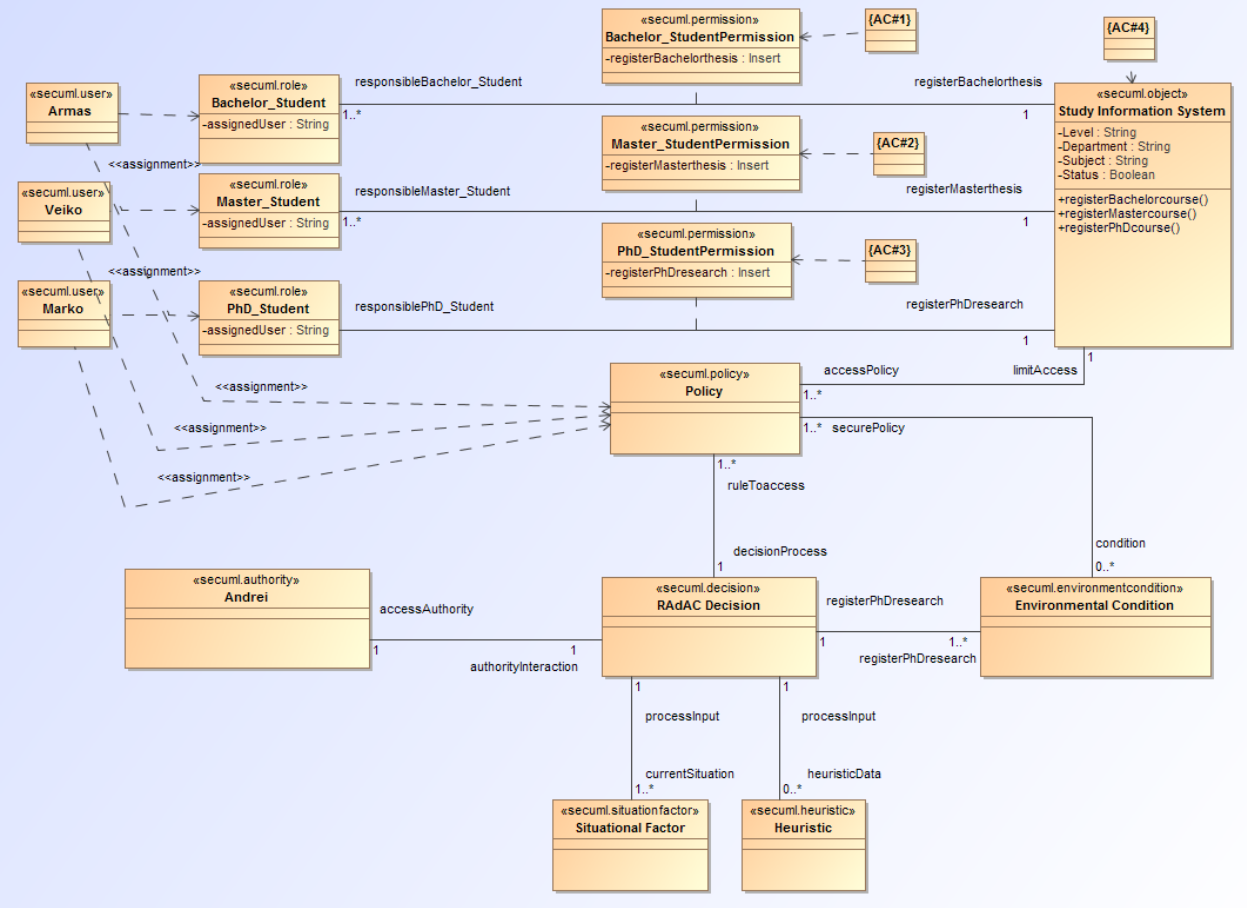


Figure 27: Example of unified model

There are three main use cases in this example

1. Access based on Roles

2. Access based on Attributes

3. Access based on Risk & Operational Need

Access based on Roles:

This is the first process that takes place whenever the subject is requesting for access. The entities listed in *Table 7* are the active entities in this process of access based on **Roles**

Table 7: Security Risk Measure Table

Entities	Description of Entity
User (Subject)	Who requests for access
Role	User is assigned to *Role* & *Permissions* are granted to user through the assigned *Role*
Permission	Provides access to its roles to perform some actions in an object
Object (Resource)	A resource in which a subject wants to perform some operation
Authorisation Constraints	Helps to strengthen permission

Here there are three users Armas, Veiko & Marko with roles Bachelor_Student, Master_Student & PhD_Student. Each user has particular permissions to perform an operation on the resource

So every user has one operation to perform through RBAC, Armas has a permission to perform **operation**, *registerBachelorcourse()* using the **action**, *registerBachelorthesis : Insert* . Veiko has a permission to perform **operation**, *registerMastercourse()* using the **action**, *registerMasterthesis : Insert* and Marko has a permission to perform **operation**, *registerPhDcourse()* using the **action**, *registerPhDresearch : Insert*. If it doesn't suit none of the permissions listed then attribute avaluation takes place. Which is explained in the following as another process.

Access based on Attributes

This is the second process that takes place whenever the access request does not find a suitable **permission** through **RBAC** [7]. In this process of access based on attributes the entities listed in *Table 8* are the active entities.

Table 8: Security Risk Measure Table

Entities	Description of Entity
User (Subject)	Who requests for access & doesn't have appropriate *permission* through their assigned r*oles*
Subject Attribute	Qualities and characteristics of *subject*
Policy	Rules that *grant/deny* access of an subject towards the object
Object	A resource in which a subject wants to perform some operation
Object Attribute	Qualities and characteristics of *object*

Environmentcondition	An *attribute* to influence access decisions

In this process as the user Marko is a teaching assistant for master's level subject, systems modelling he has requested for write and update access to the object file of courseRegister, which is inside the object Study Information System [28]. This is represented in the following **Figure 28**. It explains that the subject (user) Marko has subject attributes as Role: PhD Student, Level: PhD, Department: SE, Course: Systems Modelling, Status: Active. Object attributes are Course: Systems Modelling, Department: SE. Level: Masters, Status: Active. There is this Policy which evaluates both subject and object attributes, which says that the subject can access to Write and Update if the value of attributes are same in Department & Course for subject and object. If there is any other request requested by subjects then it will be handled in the final process based on risks and operational need.

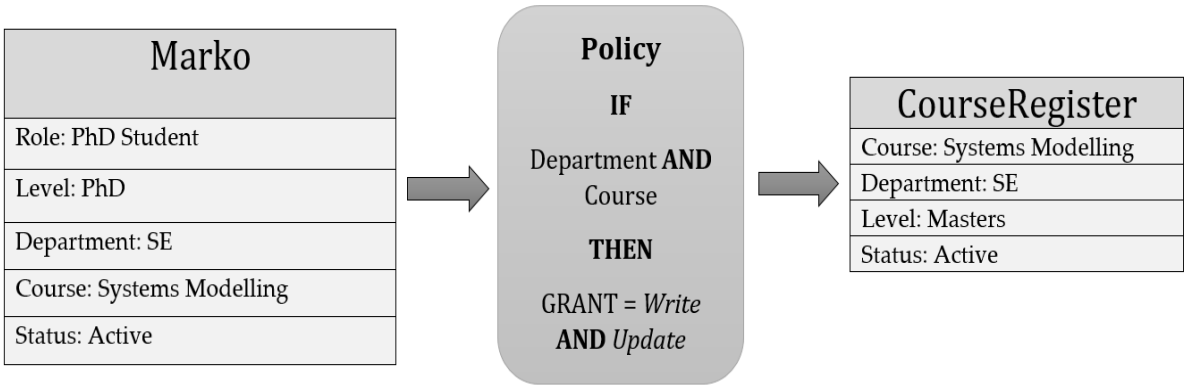


Figure 28: ABAC Model process in unified model example

Access based on Risk & Operational Need

This is the final process that takes place whenever the access request does not find a suitable **policy** through **ABAC** [16]. In this process, access will be granted based on risk and operational need. The entities listed in **Table 9** are the active entities of this process.

Table 9: Security Risk Measure Table

Entities	Description of Entity
User (Subject)	Who requests for access & doesn't have appropriate *permission / policy* through their *roles / attributes*
Decision	Helps in making decision by quantifying the risks and operational need through available attributes in real time
Environmentcondition	Detects how the surrounding environment can be exploited

Situationfactor	Identifies the security risk lies in the surrounding situation
Heuristic	Helps to grant decision by accessing the past access history
Authority	supervisor of approving authority to override security risk
Object	A resource in which a subject wants to perform some operation

For this process let us consider that the user Marko is requesting for write access to enter and update the grade sheet of the subject (systems modelling) he is working as a teaching assistant. As soon as he requests for access, it comes to the third process (RAdAC [4]) for processing as it cannot find any match in permission (RBAC [4]) and policies (ABAC [16]). It undergoes through the decision process as it is the first step of this process. Where risks and operational needs are calculated based on the metrices. **Table 10** helps to quantify the metrices of different factors that the subject and object is evolved.

Table 10: Subject Level Consideration for Security Risk & Operational Need measure

Subject	Level of Expertise
Armas - Bachelor Student	level of expertise is very low in risk compared to other *subjects*
Veiko - Master Student	level of expertise is Low in risk compared to other *subjects*
Marko - PhD Student	level of expertise is Average in risk compared to other *subjects*
Professor (Not Mentioned in example figure)	level of expertise is High in risk compared to other *subjects*
Department Coordinator (Not Mentioned in example figure)	level of expertise is Very High in risk compared to other *subjects*

In the above **Table 10** there are two more subjects mentioned apart from the subjects represented in the **Figure 27**. Though the main subjects are the three users Armas, Veiko and Marko, this two users Professor and Department Coordinator has to be mentioned in order to quantify the requested subject appropriately. Decision process of **RAdAC**[4] is very well explained through the flow chart process in **Figure 29**. Here Marko's request to grant write and update access to the grade sheet of subject, systems modelling is the input access request. When the request is in **Determine Security Risk** Flow **Table 11** will be processed. **Table 10** helps to understand how different factors are quantified for calculating security risk. **Table 12** show the quantified measures for calculating operational need. The final result of Marko's access request can be seen in **Figure 30**. In which it shows that the user Marko is granted access to this access request. Hence this ends up the example of our unified model.

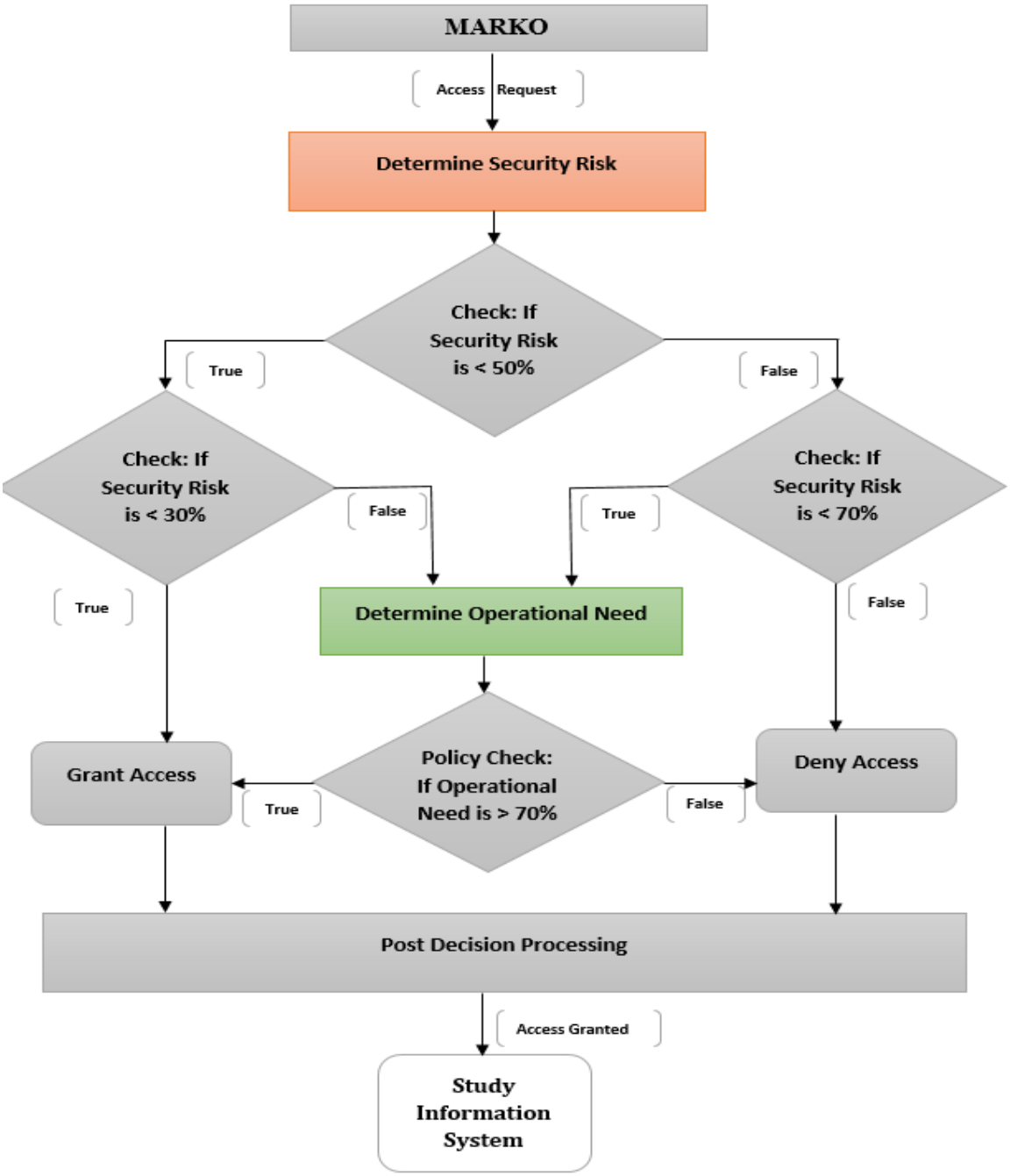


Figure 29: RAdAC functional model in unified model example

Table 11: RAdAC security risk measure for unified model example

Risk Factors Considered	Maximum	Actual Security Measure
Characteristics of People		
Role	3	**2**
Rank	3	**2**
Clearance Level	3	**2**
Access Level	3	**2**
Previous Violations	3	**1**
Educational Level	3	**1**
Characteristics of IT Components		
Machine Type	3	**1**
Application	3	**1**
Connection Type	3	**1**
Authentication Type	3	**1**
Network	3	**1**
Encryption Level	3	**1**
Distance from requestor to Source	3	**1**
Heuristics		
Risk Knowledge	3	**1**
Trust Level	3	**1**
Situational Factors		
Specific mission role	3	**1**
Time Sensitivity	3	**1**
Transaction type	3	**2**
Auditable or Non-Auditable	3	**1**
Audience Size	3	**1**
Environmental factors		
Current Location	3	**1**
Operational environment threat level	3	**1**
Characteristics of Information Requested		
Classification Level	3	**3**
Encryption Level	3	**3**
Network classification level	3	**2**
Permission Level	3	**2**
Perishable or Non-Perishable	3	**1**
Sum Total	81	**38 (46.91 % of 81)**

Table 12: RAdAC operational need measure for unified model example

Factors Considered for Operational Need Measure	Maximum	Actual Security Measure
Characteristics of People		
Role	3	**3**
Rank	3	**3**
Clearance Level	3	**2**
Heuristics		
Risk Knowledge	3	**2**
Trust Level	3	**3**
Situational Factors		
Specific mission role	3	**2**
Time Sensitivity	3	**2**
Transaction type	3	**2**
Audience Size	3	**3**
Characteristics of Information Requested		
Classification Level	3	**2**
Sum Total	30	**24 (80% of 30)**

Table 13: Security Risk Measure Table

Numerical Measure	Security Risk
0% – 29.99%	Very Low
30% – 49.99%	Low
50% – 69.99%	High
70% – 100%	Very High

Table 14: Operational Need Measure Table

Numerical Measure	Operational Need
0% – 49.99%	Low
50% – 69.99%	High
70% – 100%	Very High

Figure 30: Decision upon MARKO's access request

7.3 Summary

This chapter acts as a proof of the unified model created in the previous chapter. It shows how our unified model concepts can be applied in real time how it overcomes the limitation of other access control models created so far. This chapter starts with the process of usage to show how unified model concepts are applied to create an example.

8 Conclusion

8.1 Limitations

Though various studies are being done in this thesis, proposed unified model is a basic architecture in which enhancement can be done to improve according to the various enterprise structure and needs. Factors of RAdAC [4] model can be included more for security needs, this is lacking in this research. To comprehend the advancement of science and technology users with both good and bad motive are on rise. So, the system should be capable of identifying right users anytime. Which it does of course but for the enterprise context it has to be the best that is good. It all starts with the best understanding of subject environment and policies which allow it

Finally, there are not so many unified model concepts are available as literature. So, the validation was done with the help of so only one validation was done. Security threats in case of misconfiguration are not discussed. Maintenance of unified models like cost and all is not discussed. Application of this metamodel over time is possible if the factors are well defined and cover a range of elements used to authenticate and verify.

8.2 Answer to Research Questions

Here we will briefly discuss the research questions

Q1. What are the most important access control models?

We have defined some important access control models and from those we took only three access control models, RBAC, ABAC and RAdAC. Section 2 is the literature review of these models. These three are the most important access control models because each model has its own key concepts which make those as a standalone model and lot of other models were also derived from these models which made them as super models. Based on that these three are called as the most important access control models.

Q2. How do we define comparison criteria for an access control model?

We have defined three main components **(Subject, Policy & Object)** of an access control model in **Section 3**. Based on that different access control models are compared with the behavior and its interaction with one another. Three key elements are for an access control are **Subject, Policy & Object.** As these three elements define the nature of an access control model these are the key elements

Q3. How can anyone understand an access control model with the help of three building blocks?

Section 4 clearly answers this question through survey analysis. This survey is conducted by a group of audience and they understood all the three access control model concepts (RBAC, ABAC & RAdAC) clearly. This was observed through the given exercises.

Q4. How is the conceptual model constructed into a unified model?

Based upon the key elements defined in **Section 3** suitable needs are grouped and matched with the respective components of our unified model from **Section 4**. As it is designed to suit most of

the needs it will make the best fit, it is also designed in the way to be **flexible** so that new concepts can be added to this model as well. So, we present our unified model to be your suitable access control model. Key concepts are figured out separately as a module with the help of its **three main building blocks (Subject, Policy & Object)**. By aligning all the key concept modules, a unified model is constructed. Same is been discussed in **Section 5** of how to make a unified model through alignment.

Q5. How can we verify our constructed unified model?

Our unified model can be verified with the help of validating it against a similar research model. This is clearly explained in **Section 6** We know that not all the access control models are flexible. Making an access control model flexible

Q6. What is the proof that this verified model is consistent and reliable?

We know that not all the access control models are flexible. Making an access control model flexible will be the challenge. But it is possible with the guidance of **Section 6 & 7**. **Section 6** conveys what all the limitations it overcomes and its consistency against other models. **Section 7** shows the practical way through an example that how this unified model is consistent and reliable. Also, to add up unified model constructed in this thesis can be modified or changed based upon anyone's need.

8.3 Summary

This metamodel provides an approach to simplify the specification and implementation of different concepts of subject, policy, and object in an access control model. It enables the model administrators to concentrate on the model they wanted rather than compromising their organizational set up according to some specific access control model or implementing a model that cannot be utilized completely to its vast extent. That's because of the distinct concepts in access control model implemented and the set up on the organization that implemented this model.

With this currently proposed, model administrator don't have to worry about distinct concepts like RBAC, ABAC, RAdAC, it surpasses each model mentioned here by implementing all of its features as a Unified Model

8.4 Future Work

As access control model deals with the security feature of an enterprise set up there will be always a place to improve or enrich according to the complexity of an enterprise. This unified metamodel provides a basic through this research set up but hasn't implemented in development. Hence it can be extended for development in order to build a basic implementation set up. This should give way for further research enriching the concepts of a unified metamodel with further research. Also, this unified meta-model helps in such a way that the person who is going to implement such set up can extend with this as a foundation also it depends upon the goals of such people.

9 References

[1] Fathy M., Bahgat M., & Yehia A. (2013). Security access control research trends. In 6th Joint IFIP Wireless and Mobile Networking Conference (WMNC),1–6.

[2] McGraw R. W. (2009). Risk Adaptable Access Control (RAdAC). Information Assurance Architecture and Systems Security Engineering Group National Security Agency.

[3] Kandala, S., Sandhu, R., & Bhamidipati, V. (2011). An Attribute Based Framework for Risk-Adaptive Access Control Models. Sixth International Conference on Availability, Reliability and Security.

[4] Britton, D. W., Brown, I. A., (2007). A Security Risk Measurement for the RAdAC Model. Thesis

[5] McGraw, R. W., (2009): Risk-Adaptable Access Control (RAdAC). Unpublished work

[6] Farroha, B., & Farroha, D. (2012). Challenges of "Operationalizing" dynamic system access control: Transitioning from ABAC to RAdAC. IEEE International Systems Conference (SysCon).

[7] Sandhu, R. S., Coyne, E. J., Feinstein, H. L., & Youman, C. E. (1996). Role-based access control models. IEEE Computer 29(2), (pp. 38-47)

[8] Li, M., & Fan, B. (2012). The modeling of RBAC model based on UML and XACML. 2012 International Conference on Systems and Informatics (ICSAI)

[9] Saidani, O., & Nurcan, S. (2006). FORBAC: A Flexible Organisation and Role-Based Access Control Model for Secure Information Systems.

[10] Chadwick, D. W., & Otenko, A., (2003). The PERMIS X.509 Role Based Privilege Management Infrastructure. Future Gener. Comput. Syst., 19(2), 277–289.

[11] Chandran S.M., Joshi J.B.D. (2005) LoT-RBAC: A Location and Time-Based RBAC Model. Web Information Systems Engineering – WISE. Lecture Notes in Computer Science, vol 3806. Springer, Berlin, Heidelberg

[12] Matulevičius, R., (2016). Fundamentals of Secure System Modelling. Chapter 10. Unpublished draft

[13] Ferraiolo, D., Sandhu, R., Gavrila, S., Kuhn, D.R., R., C., (2001). Proposed NIST standard for rolebased access control. ACM Transactions on Information and System Security 4(3), 224–274

[14] Basin, D., Doser, J., Lodderstedt, T., (2006). Model Driven Security: from UML Models to Access Control Infrastructure. ACM Transactions on Software Engineering and Methodology (TOSEM) 15(1), 39–91

[15] Lodderstedt, T., Basin, D., Doser, J., (2002). SecureUML: A UML-based Modeling Language for Model-driven Security. In: Proceedings of the 5th International Conference on The Unified Modeling Language. vol. 2460, pp. 426–441. Springer-Verlag

[16] Hu, V. C., Ferraiolo, D., Kuhn, R., Friedman, A. R., Lang, A. J., Cogdell, M. M., Schnitzer, A., Sandlin, K., Miller, R., Scarfone, Karen., (2013). Guide to Attribute Based Access Control (ABAC) Definition and Considerations (Draft). NIST Special Publication on Computer Security.

[17] Gajeli, P. B., Yalagi, P. S., (2016). A Survey on Access Control Models and Encryption Schemes for Cloud Storage System. International Journal of Engineering Research and Management (IJERM). Volume-03, Issue-10.

[18] Karp A. H., Haury H., Davis M. H., (2009). From ABAC to ZBAC: The Evolution of Access Control Models, HP Laboratories, HPL-2009-30

[19] Sinnema R. R., (2013). Secure Cloud Development. Musings on the Art and Craft of Creating Secure Software in the Cloud Era. Blog of XACML. Retrived May 08th, 2013, from https://remonsinnema.com/tag/radac/

[20]. Yuan E., Tong J., (2005). Attributed based access control (ABAC) for Web services. Web Services, 2005. ICWS 2005. Proceedings. 2005 IEEE International Conference. Booz Allen Hamilton Inc., McLean, VA, USA.

[21]. Verma, S., Singh, M., Kumar, S., (2012). Comparative analysis of Role Base and Attribute Base Access Control Model in Semantic Web. International Journal of Computer Applications. Volume 46, No.18

[22]. Downs, D. D., Rub, J. R., Kung, K. C., Jordan, C. S., (1985). Issues in Discretionary Access Control. IEEE Symposium on Security and Privacy (pp. 208-208). Oakland, CA.

[23] Ausanka-Crues, R. Methods for Access Control:Advances and Limitations. Harvey Mudd College. Claremont, California.

[24] Korman, M., Langerstorm, R., Ekstedt , M., (2015). Modelling Authorization in Enterprise-wide Contexts. Practice of Enterprise Modelling (PoEM) Short and Doctoral Consortium Papers (pp. 81-90). Stockholm, Sweden:KTH Royal Institute of Technology.

[25] Sharma, N. K., & Joshi, A. (2016). Representing Attribute Based Access Control Policies in OWL. Tenth International Conference on Semantic Computing (ICSC) (pp. 333-336). Delhi: Indian Institute of Technology.

[26] Hu, V. C., Ferraiolo, D. F., Kuhn, D. R., (2006). Assesment of Access Control Systems. NIST Publication on Computer Security. Gaithersburg, MD.

[27] Coyne, E., Weil, T. R., (2013). ABAC and RBAC: Scalable, Flexible, and Auditable Access Management. Published by the IEEE Computer Society.

[28] University of Tartu, Study Information Systematic (SIS). from http://www.ut.ee/en/studies/study-regulations/system.

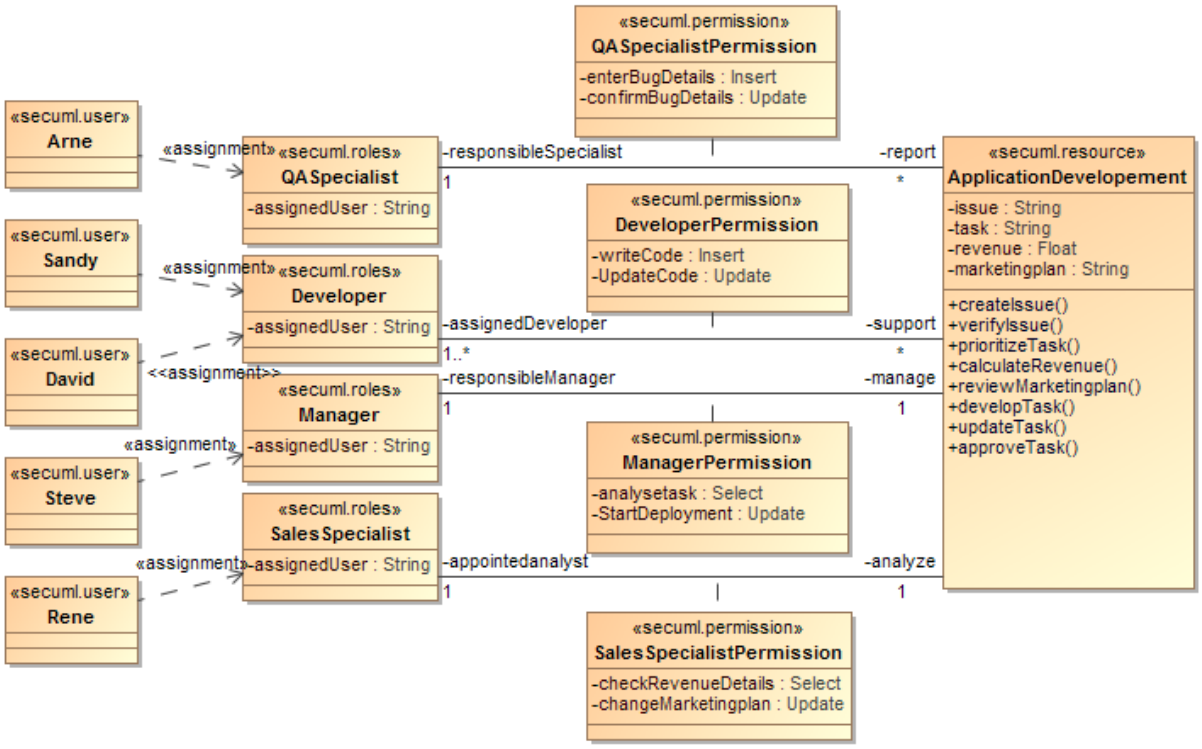
[29] Yavatkar, R., Pendarakis, D., and Guerin, R. (2000), A Framework for Policy-based Admission Control. Published by RFC Editor from United States.

[30] Elliott, A. A., Knight, G. S. (2010). Role Explosion: Acknowledging the Problem. Proceedings of the 2010 International Conference on Software Engineering Research & Practice, SERP 2010, Las Vegas, Nevada, USA.

Appendix

I RBAC Task

RBAC Model in SecureUML Representation



1.1 Choose the subject(s) of RBAC from above representation:

a. ApplicationDevelopement

b. Steve - Manager

c. QASpecialistPermission

d. David - Developer

e. Sandy - Developer

f. SalesSpecialistPermission

g. DeveloperPermission

h. Rene - SalesSpecialist

i. ManagerPermission

j. Arne - QA Specialist

1.2 Choose the permission(s) / policies of RBAC from above representation:

a. ApplicationDevelopement

b. Steve - Manager

c. QASpecialistPermission

f. SalesSpecialistPermission

g. DeveloperPermission

h. Rene - SalesSpecialist

d. David - Developer

e. Sandy - Developer

i. ManagerPermission

j. Arne - QA Specialist

1.3 Choose the Object(s) of RBAC from above representation:

a. ApplicationDevelopement

b. Steve - Manager

c. QASpecialistPermission

d. David - Developer

e. Sandy - Developer

f. SalesSpecialistPermission

g. DeveloperPermission

h. Rene - SalesSpecialist

i. ManagerPermission

j. Arne - QA Specialist

1.4 By considering the above RBAC model determine which user will able to run which operations to access the resource(s)?

a. Arne - QASpecialist - createIssue() - enterBugDetails : Insert

b. Sandy - Developer - prioritizeTask() - analyseTask : Select

c. Steve - Manager - prioritizeTask() - analyseTask : Select

d. Rene - SalesSpecialist - calculateRevenue() - checkRevenueDetails : Select

e. Rene - SalesSpecialist - reviewMarketingPlan() - changeMarketingplan : Update

f. Sandy - Developer - developTask() - writeCode : Insert

g. Sandy - Developer - updateTask() - UpdateCode : Update

h. Steve - Manager - approveTask() - StartDeployment : Update

i. David - Developer - developTask() - writeCode : Insert

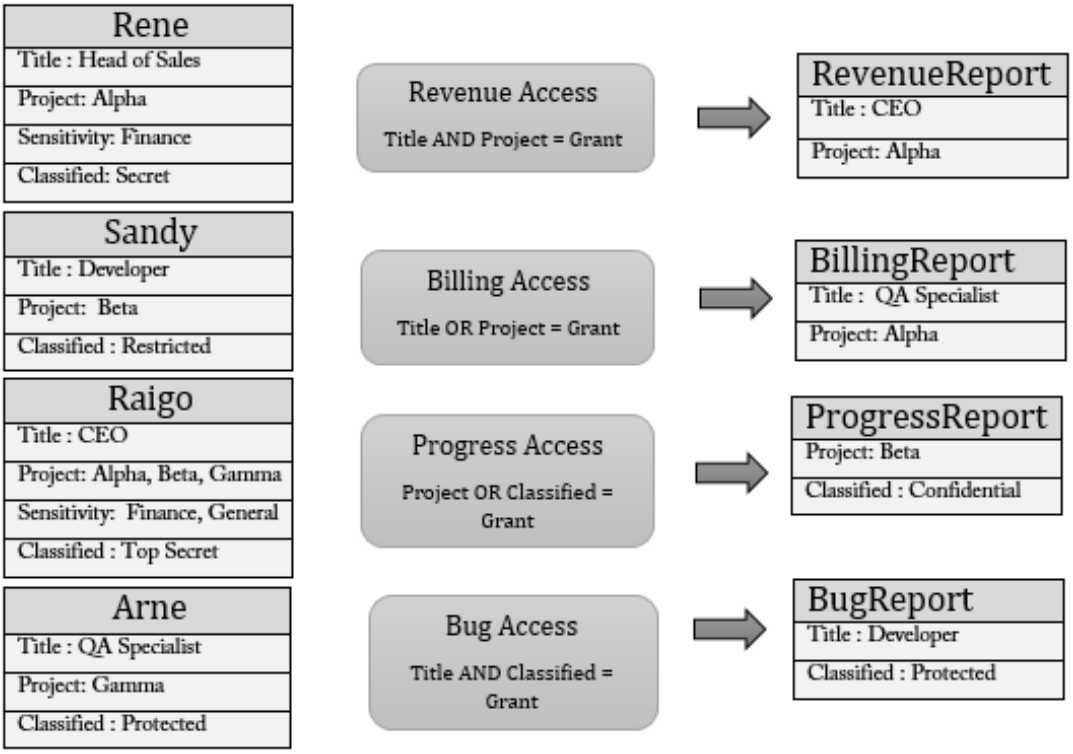
j. David - Developer - updateTask() - UpdateCode : Update

k. David - Developer - prioritizeTask() - analyseTask : Select

l. Arne - QASpecialist - verifyIssue() - confirmBugDetails : Update

II ABAC Task

ABAC Model Representation



2.1 Choose the subject(s) of ABAC from above representation:

a. Bug Access **e.** Raigo **i.** Revenue Report

b. Billing Access **f.** Progress Access **j.** Sandy

c. Revenue Access **g.** Billing Report **k.** Progress Report

d. Arne **h.** Rene **l.** Bug Report

2.2 Choose the permission(s) / policies of ABAC from above representation:

a. Bug Access **e.** Raigo **i.** Revenue Report

b. Billing Access **f.** Progress Access **j.** Sandy

c. Revenue Access **g.** Billing Report **k.** Progress Report

d. Arne **h.** Rene **l.** Bug Report

2.3 Choose the Object(s) of ABAC from above representation:

a. Bug Access	**e.** Raigo	**i.** Revenue Report
b. Billing Access	**f.** Progress Access	**j.** Sandy
c. Revenue Access	**g.** Billing Report	**k.** Progress Report
d. Arne	**h.** Rene	**l.** Bug Report

2.4 By considering the above ABAC model determine which user(s) will able to access which resource(s) ?

a. Rene - can access – Revenue Report

b. Raigo – can access – Billing Report

c. Sandy - can access – Bug Report

d. Arne - can access – Progress Report

e. Rene & Arne - can access - Bug Report

f. Raigo & Sandy - can access – Progress Report

g. Arne - can access - Bug Report

h. Raigo, Sandy & Arne - can access – Billing Report

i. Arne, Raigo & Rene - can access – Billing Report

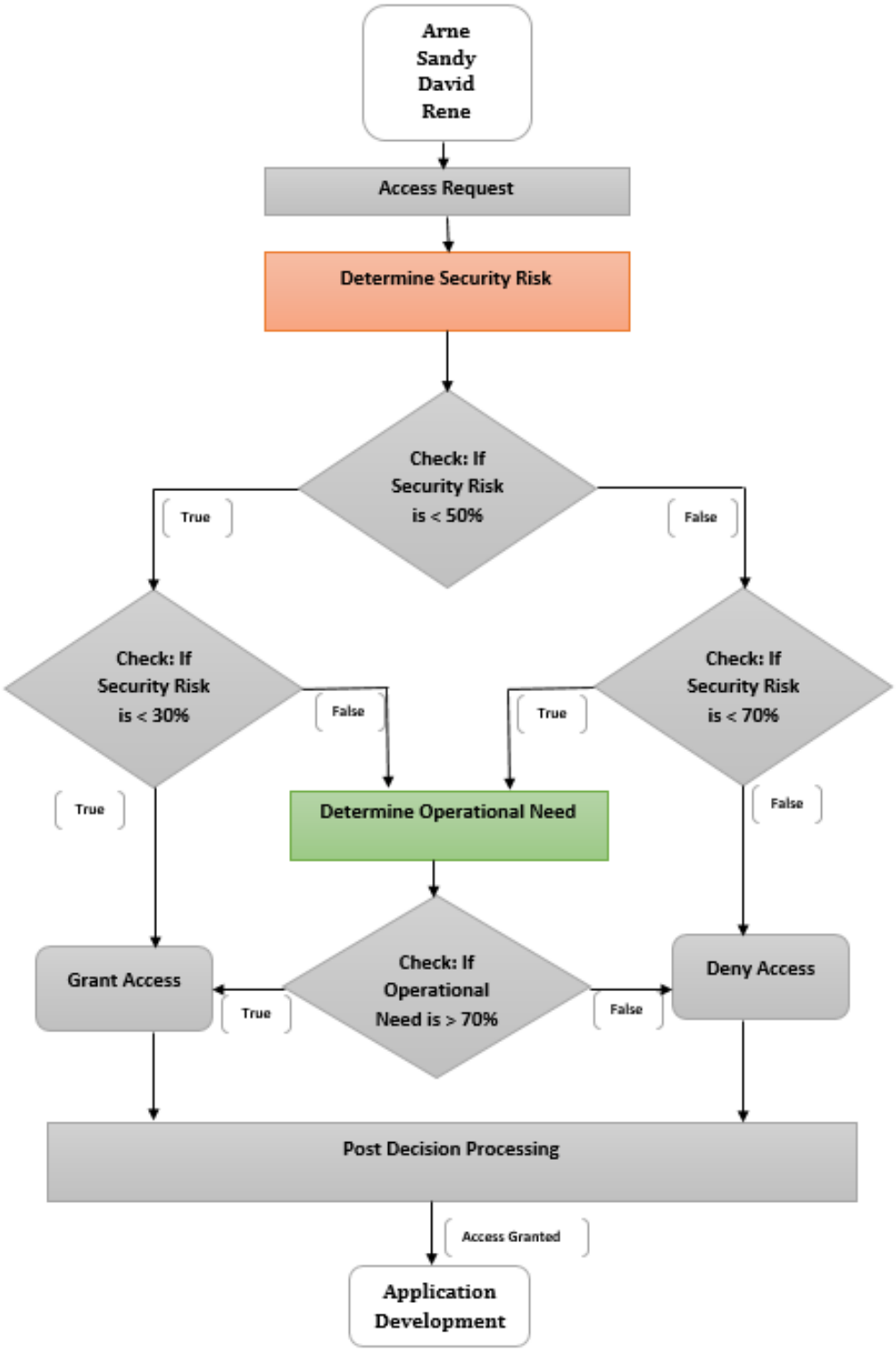
j. Sandy & Arne - can access – Bug Report

k. No One - can access – Bug Report

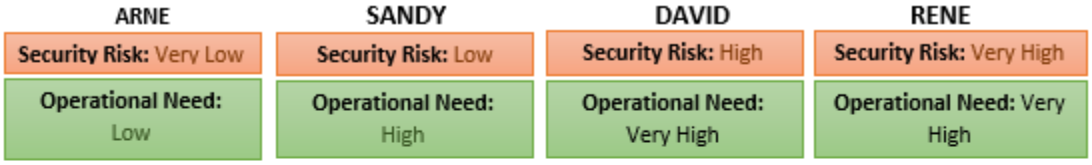
l. Raigo & Rene - can access – Revenue Report

III RAdAC Task

RAdAC Process Representation



Different Types of user with their quality measure



Security Risk Measure Table

Numerical Measure	Security Risk
0% – 29.99%	Very Low
30% – 49.99%	Low
50% – 69.99%	High
70% – 100%	Very High

Operational Need Measure

Numerical Measure	Operational Need
0% – 49.99%	Low
50% – 69.99%	High
70% - 100%	Very High

3.1 Choose the subject(s) of RAdAC from its process representation:

a. Check: If Security Risk is < 30%
b. Deny Access
c. Check: If Operational Need is > 70%
d. Determine Security Risk
e. Rene
f. Post Decision Processing
g. Grant Access
h. Arne
i. David
j. Sandy
k. Check: If Security Risk is < 50%
l. Application Development
m. Check: If Security Risk is < 70%
n. Access Request
o. Determine Operational Need
p. Permit Access

3.2 Choose the permission(s) / policies of RAdAC from its process representation:

a. Check: If Security Risk is < 30%
b. Deny Access
c. Check: If Operational Need is > 70%
d. Determine Security Risk
e. Rene
f. Post Decision Processing
g. Grant Access
h. Arne
i. David
j. Sandy
k. Check: If Security Risk is < 50%
l. Application Development
m. Check: If Security Risk is < 70%
n. Access Request
o. Determine Operational Need
p. Permit Access

3.3 Choose the Object(s) of RAdAC from its process representation:

a. Check: If Security Risk is < 30%

b. Deny Access

c. Check: If Operational Need is > 70%

d. Determine Security Risk

e. Rene

f. Post Decision Processing

g. Grant Access

h. Arne

i. David

j. Sandy

k. Check: If Security Risk is < 50%

l. Application Development

m. Check: If Security Risk is < 70%

n. Access Request

o. Determine Operational Need

p. Permit Access

3.4 By considering the above ABAC model determine which user(s) will able to access which resource(s) ?

a. Rene - can access – Application Development

b. David - can access – Application Development

c. Sandy - can access – Application Development

d. Arne - can access – Application Development

Questionnaire on basis of three access control model task, Select how strongly you feel the following:

	Access Control Model		
	RBAC	**ABAC**	**RAdAC**
Ease of Learning: How quickly the system can be learned by the various groups of users?			
Which model *Subject* is easy to learn?			
Which model *Policy/Permission* is easy to learn?			
Which model *Object* is easy to learn?			
Which model is easy to learn?			
Task efficiency: How quickly the system can do a task efficiently?			
Which model *Subject* is efficient for frequent use?			
Which model *Policy/Permission* is efficient for frequent use?			
Which model *Object* is efficient for frequent use?			
Which model is efficient for frequent use?			
Ease of remembering: How quickly the system can be remembered for occasional user?			
Which model *Subject* is easy to remember for occasional use?			
Which model *Policy/Permission* is easy to remember for occasional use?			
Which model *Object* is easy to remember for occasional use?			
Which model is easy to remember for occasional use?			
Subjective satisfaction: How quickly the system adopts to satisfy user needs?			
Which model *Subject* is satisfied with most of all occasions?			
Which model *Policy/Permission* is satisfied with the system?			
Which model *Object* is satisfied with the system?			
Which model is satisfied with the system?			
Understandability: How easy is to understand the system functionality?			
Which model *Subject* is easy to understand?			
Which model *Policy/Permission* is easy to understand?			
Which model *Object* is easy to understand?			
Which model is easy to understand?			

IV License

Non-exclusive licence to reproduce thesis and make thesis public

I, **Darwin**,

(author's name)

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:

   1.1. reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and

   1.2. make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright,

of my thesis

Comparison and Alignment of Access Control Models,

(title of thesis)

supervised by Raimundas Matulevicius,

(supervisor's name)

2. I am aware of the fact that the author retains these rights.

3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, **14.08.2017**