**PyOrBAC: Security Policy Framework based on OrBAC**

Samer Machara, Francis Abrante de Machara

Telecom SudParis, France

(samer.machara\_marquez@it-sudparis.eu, Francis\_andrea.Abrante\_hernandez@telecom-sudparis.eu )

**ABSTRACT**

This document presents the terminology related to

The objective of this master thesis is to design and implement a security policy framework based on the OrBAC model

**Keywords**. Security, networks.

**INTRODUCTION**

**CHAPTER I. TELECOM SUDPARIS**

**CHAPTER II. STATE OF THE ART**

Before reviewing how the software was designed it is necessary to define some concepts and establish how they are related.

1. **Security**

Nowadays, the security has become to be an extremely important subject around the world, since computer’s systems, passing by airports, schools, etc. In fact a lot of money is invested every year in this subject, and some countries are very concern about this.

But, what is exactly the security? It is possible found several definitions depending of the context, but all of them are very coherent, describing that is a protection mechanism from attacks, damages, robbery or any criminal activity. For example, according to the Merriam Webster dictionary [18] is the quality or state of being secure, understanding by secure, free from fear or distrust. This definition applies in any scope, like for example: national security, job security, computer science security, security of buildings, etc

* 1. **Security in Networks**

The security in the computer science area has become more and more important each day because nowadays the computer systems contain all the information referring to our lives and through them we handled practically all the information of the modern era available in the planet, since the personal information store in the home computers, until the medical information, banking information, enterprise information, governmental information, etc.

At the moment the confidence that has been deposited in the computer systems is immense, which causes that is too important to preserve the information and systems from lost, faults or robberies, with or without intention, because it is well known that sometimes the systems fail for many reasons, may be for a hardware component that is broken, or some file in the operative system that is corrupted, furthermore sometimes appear the scenario where the fail occur because someone execute some command or some action in an incorrect way occasioning a disaster, exist also the failures for natural conditions or natural disasters and of course exist in addition the scenario where some non authorized person try to steal information or sabotage the systems. This last case can be perform from inside the network’s company or outside the network’s company, that’s why although the network’s border is normally the most armored, the internal network must has also efficient security systems.

Here appear in scene the concepts of: confidentiality, reliability, availability and integrity

* 1. **Security Policy**

Due to the fact that organizations must to define very well their security systems in order to guaranty the business operations, a group of specific security definitions is required; these specifications establish the constrains for each particular component that require to be secured.

This set of conditions after one first phase of design; it must pass a series of approval steps starting by the security officer and ending by someone from the business core. Normally before applying some new policy, this one must be tested in a testing platform, thus, it is possible define the impact over the systems in production, over the infrastructure in general and over the user activities; however not always it is possible realize this kind of test. In any case, always a detailed plan must exist, a plan that describe what exactly the policy must to take care of, where the policy must be apply, who must be affected, which equipments are involved, how will be apply and the recovery plan or reverting plan in case that something wrong happen.

* 1. **Access control Policy**

The access control policy contain the relation between users or resources and specific actions, that’s mean that specify who can do what over a resource, understanding by resource whatever data, object, equipment, network, etc. that can be accessed or used in an infrastructure.

Some examples are: who can do log in during the night? who can read some specific document? who can access the Internet during the labor hours?, etc.

<http://publib.boulder.ibm.com/infocenter/wchelp/v5r6/index.jsp?topic=/com.ibm.commerce.admin.doc/concepts/caxaccesspolicy.htm>

1. **OrBAC**

OrBAC (Organization Based Access Control) it is a security policy model created by Alexandre Miege [2] that allow consolidate security policies in a standard way, “defining a structure that allow define complex and flexible policies that match with the systems’ reality” [2]. Its objective is to permit establish and guarantee the configuration policymanagement over the different network components in a centralized way and without concerning about the different implementations.

It was designed in two levels, the concrete level that is composed by tree main components: subject, action and object and the abstract level that is composed by: role, activity and view. And in-between parallel to all the components as a model’s heart we found the organization, which have attached all the policy specifications and in this way the authorizations are associated with organizations, thus it is possible handle several security policies associated with different organizations in the same implementation.

The main difference between Orbac and the other models of access control is the inclusion of the concepts Organization and Context; Organization is represented by any organized group of entities that collaborate with a common objective. And Context represents the conditions that enable o disable the authorizations, these conditions can be setting up by users or environment circumstances.

In this model, the policies are based in an organization’s hierarchy where can exist several levels depending of the design done by the administrator based in the network and company needs, this allow that for each organization its possible define their own components and to set out the boundaries of the policies. Each organization as was mentioned above is transversal to their components, which mean that all the components have direct interaction with it and are directly related as is illustrated in figure 1.

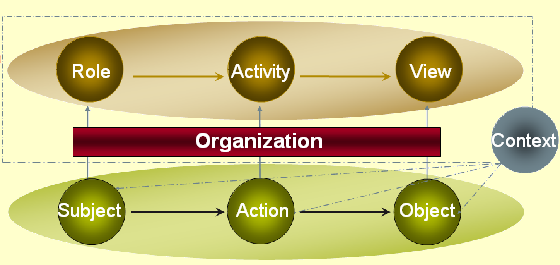


Figure 1 [1] OrBAC Model

**Concrete level components**

Subject: The subject is an active entity that can generate information, give it to the objects and also can change the system status. For example, a process can be a subject because the process it’s capable of produce information. But a subject can be anything in a company that can produce information, as a printer, video security cameras, sensors, etc. However, in order to avoid confusions in the OrBAC environment the Subject is defined only as users and organizations.

Action: The Oxford dictionary [25] defines an action as the process of doing something, typically to achieve an aim. In OrBAC, an action is a modification that can be done over an object, and concretely computer actions like read, write, modify, etc.

Object: The objects are normally passive entities, that only contain or receive information they cannot generate it. But it is possible that in some point an object starts to generate information and in that point it becomes a subject. Nevertheless in the OrBAC context as was already defined, the object can act as everything less like user or organization.

**Abstract level components**

Role: A role is usually a position or function that one subject can perform in a particular situation, this term it is very use in companies, where normally a position is associated with a role, which imply a set of activities and responsibilities that the person with that role must to fulfill. However one person or subject can be assigned to different roles at the same time or in a excluding way, that’s depend of the conflicts and overlapping between activities.

This concept is very useful in security definitions and can be found very often in the different security models without concerning the design. Regarding to OrBAC, a role is a set of authorizations, thus the subjects where the same security rule apply can be logically grouped, in this way the authorization is grant directly to one role instead of a subject one by one.

Activity: As well as we defined the concept of action by the Cambridge dictionary we define also the concept of activity in order to clarify its use. According to him, activity is the work of a group or organization to achieve an aim. In OrBAC, an activity is a task, specifically is an operation that must be done through the use of actions. An activity is a set of actions on which apply the same rule.

View: A view is a set of objects that has something in common, generally the view is composed by objects where the administrator want apply the same policy. The view is defined directly in a policy.

Inasmuch as the subjects are logically grouped in roles, when a role is assigned to a subject automatically that subject has granted all the security definitions that the role has. Similarly, when a specific policy is assigned to an activity, automatically all the actions that the activity is related with, have the same policy. And finally in the same order of ideas, when a specific authorization is given to a view, all the objects that are associated with that view will receive the policy.

Once all the elements are defined it is necessary establishing the relation between them in order to create the structure and define the security schema. Those relations allow the interaction between different components in the model and define how they are grouped as we will see later. For example, one relation can define that one object belong to one specific view in one specific organization, whereas in other one not. Furthermore, is a way for organize the elements in the specification.

And as the model has two levels of components, it is possible define two level of permissions, one abstract and one concrete. In the abstract level are defined those policies that apply to sets, that is to say over roles, activities and views, whereas in the concrete level its possible define the rules with more granularity that means that the policy can be assigned directly over one element. And at this point one important component appear in scene, the inheritance, that’s basically mean that the policies defined in the abstract level, are automatically apply over the concrete components. Evidently when all these elements start to have interaction the conflict arrives, derives of the policies defined in the different levels and assigned to different roles in conjunction with the policies defined directly over the object.

In order to solve that problem the model establish the priority over the concrete level, hence, the explicit policy that is defined over one specific object for grant or deny access to one specific user, has the priority.

Once all these is well designed, one important part is missing, a set of constrains or conditions that must be true for activate one policy. This set is the context, and determine when one policy will be apply. The model define several context types according to space, time,

Although the model describe several components, all of that obeys to a single purpose grant or deny permissions, that at the end become to be a relation between the organization, the role, the view, the activity and the context.

**Relations**

Relevant role:

Empower:

Allow to join the subject in a specific role for one organization

Relevant view:

Use:

Define which objects belong to one view

Relevant activity:

Consider:

Define which actions belong to one activity for one organization

Relevant context:

Hold:

Define that one subject

Permission:

Allow to join the view, the activity and the context

Is permitted:

Is prohibited:

1. **PREVIOUS WORK**

MotorBAC

**CHAPTER III. SOLUTION**

1. **FRAMEWORK SECURITY POLICY**

The OrBAC model define all the security policy structure but does not exist formal specifications about how must be implemented either how must be the interaction between the implementation and the different components to serve, moreover any architecture is proposed. It is in these needs where this project fits.

The main goal of this project is to design and develop an administrative tool based in the OrBAC model, that allow consolidate the security definitions through the different infrastructure’s components in an organization, and to configure automatically those components based in the definitions done in it.

1. **PROPOSED SOLUTION**

The idea was to

OrBAC is a framework for specify security policies

Use a Data Base for store the data and the XML for the communication

Next figure represent the General Architecture of OrBAC implementation, it is a first approach, we describe which are the different components that brings services into the engine

The idea is separate in small modules the different functions that integrate the OrBAC engine, in order to simplify its maintainability and performance.

This engine can interact through API with other systems or different implemented user interfaces

Also describe the interaction among the different components

The communication with external OrBAC components is done thought XML files, the definition of these files was establish in

**Management Service:**

This module is in charge of OrBAC Administration, Security and Configuration.

By Administration Service regards OrBAC database backups and restore, jobs schedule, replication information, etc.

By security give access to the OrBAC modules

By configuration setting the variables that determine the behavior of OrBAC

**Storage Service:**

This module function is store all the elements of OrBAC as well as Logs and Security Politics.

**Connection service:**

It is an API that allows the interaction between OrBAC and graphic interface or remote systems

**Context Manager:**

**Some componets cannot support the context definition so the idea is that this component**

**need**

**Context Agent: take request from the application**

**no estoy claro todavia pero la idea es de que alguien monitoree los diferentes contextos**



OrBAC Architecture

2.1

1. **XML SCHEMA**

For this research, the schema is a structure done in XML which defines what is going to be the template and the parameters in the final file

* 1. **Concepts**

Attributes: apply to all the objects in order to have a parallel way of grouping. Not mandatory

1. **COMPONENT’S MODEL**

When one policy is applied, it is apply over a role. Which are the parameters for the translation It is how is translated the rule in the configuration, how is the mapping between the policy and the different components

1. **MINIMAL POLICY**

Context. the multiple elements defined react against attacks. Define attacks contexts and according to this attacks context some rules are defined to block the attacks. In addition we defined the normal context, how the system reacts when does not attack. And another context where is define the rules when all the possible attacks happen, how to manage the difficult decisions. The solution proposed only do context management and priorities, now, when the minimal context is activated, which are the rules, depend of the particular environment and the security design. Priorities between contexts.

1. **API**

gfg

* 1. **API Pull Mode**

Use for non defined requirements, when the information related with the policy it’s demanded by a none expected component. This API defines the subject, action and object and send the request to the PyOrBAC engine, who will response with a grant or deny access.

Basically, the policy defined previously its contained in a XML file that correspond with the XML Schema manage by PyOrBAC, after that this file go in the policy’s engine.

The possible responses in this case are:

* Permit
* Deny
* No response (In case that does not exist any rule that correspond with the requirement)
* Error
  1. **API Push Mode**

Use for the interaction with specific and well defined tools, in this mode exist totally compatibility between the policies defined in the engine and the specific component.

The complexity in this mode, it’s how configure the different components from the engine, thus it is necessary define how to manage the different contexts in a unique way.

In this scenario when one or several policies change, the engine starts the synchronization with the affected component, in order to maintain the most updated information possible.

1. **INFRAESTRUCTURE**

The infrastructure proposed for test the framework it’s based in a virtual network based on Ubuntu 10.04 LTS and VirtualBox.

**Ubuntu 10.04 LTS**

**VirtualBox**

**OpenLDAP**

**CONCLUSION**

This paper presents

**REFERENCES**

[1] <http://www.orbac.org>

[2] Alexandre Miege, “*Definition of a formal framework for specifying security policies. The Or-BAC model and extensions*”, Ecole Nationale Superieure des Telecommunications, 2005

[3] Nizar Kheir, “*Response policies and counter-measures: Management of service dependancies and intrusion and reaction impacts*”, Ecole Nationale Superieure des Telecommunications de Bretagne, 2010

[4] Organization-based access control. Available online at: <http://en.wikipedia.org/wiki/Organisation-based_access_control>

[5] Anas Abou El Kalam, Yves Deswarte, “*Multi-OrBAC: a New Access Control Model for Distributed, Heterogeneous and Collaborative Systems*”, ENSIB, Universite de Toulouse. Available online at: <http://homepages.laas.fr/deswarte/Publications/06427.pdf>

[6] Nizar Kheir,Herve Debar, Frederic Cuppens, Nora Cuppens-Boulahia and Jouni Viinikka, “*A Service Dependency Modeling Framework for Policy-based Response Enforcement*”, DIMVA 09 : 6th international conference on Detection of Intrusions, Malware and Vulnerability Assessment, Italy, 2009. . Available online at: <http://hal.archives-ouvertes.fr/docs/00/43/33/02/PDF/DIMVA09_-_Final_copy.pdf>

[7] Nasser B, Laborde R, Benzekri A, Barrere F, Kamel M, “*Dynamic creation of inter-organizational grid Virtual Organizations*”, IRIT Laboratory, SIERA team, Paul Sabatier University, Toulouse, FRANCE, 2005. Available online at: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1572251>

[8] Ahmed A. Hassan, Waleed M. Bahgat, “*A framework for translating a high level security policy into low level security mechanisms*”, Journal of ELECTRICAL ENGINEERING, VOL. 61, NO. 1, 2010. Available online at: <http://iris.elf.stuba.sk/jeeec/data/pdf/1_110-3.pdf>

[9] Ahmad A. Hassan, Waleed M. Bahgat, Abdel Fatah Ibrahim, “*EOrBAe Based Network Security Management Toolkit*”, IEEE, 2009. Available online at: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5010537>

[10] Frederic Cuppens, Nora Cuppens-Boulahia, Alexandre Miege, “*Inheritance hierarchies in the Or-BAC model and application in a network environment*”, Available online at: <http://orbac.org/publi/OrBAC/OrBacHierachies.pdf>

[11] Anas Abou El Kalam, Salem Benferhat, Alexandre Miege, Rania El Baida, Frederic Cuppens, Claire Saurel, Philippe Balbiani, Yves Deswarte, Gilles Trouessin, “*Organization based access control*”, IEEE, 2003. Available online at: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1206966>

[12] F. Cuppens, N. Cuppens-Boulahia, T. Sans and A. Miège “*A formal approach to specify and deploy a network security policy*”, Second Workshop on Formal Aspects in Security and Trust (FAST), 2004.

[13] Tejeddine Mouelhi, Yves Le Traon, Benoit Baudry, “*Transforming and Selecting Functional Test Cases for Security Policy Testing*”, International Conference on Software Testing Verification and Validation International Conference on Software Testing Verification and Validation, 2009. Available online at: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4815349>

[14] Lei Qing, Zhao Huan, “*Research on Dynamic Authorization in Workflow of Virtual Enterprise*”, IEEE, 2007. Available online at: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4341263>

[15] Frederic Cuppens, Nora Cuppens-Boulahia, Celine Coma, “*MotOrBAC : un outil d’administration et de simulation de politiques de securite*”, GET/ENST Bretagne

[16] Frederic Cuppens, Alexandre Miege, “*Administration Model for Or-BAC*”, GET/ENST Bretagne, ENST

[17] Narhimene Boustia, Aicha Mokhtari, “*Representation and reasoning on ORBAC: Description Logic with Defaults and Exceptions Approach*”, Third International Conference on Availability, Reliability and Security, 2008

[18] Merriam Webster dictionary http://www.merriam-webster.com

[19] <http://www.sun.com/blueprints/1201/secpolicy.pdf>

[20] <http://wordpress.org/>

[21] <http://www.modsecurity.org/>

[22] <http://en.wikipedia.org/wiki/Security-Enhanced_Linux>

[23] <http://www.netfilter.org/>

[23] <http://publib.boulder.ibm.com/infocenter/>

[24] <http://dictionary.cambridge.org>

[25] <http://oxforddictionaries.com>