Access Control Protocol for Cloud Systems Based On the Model TOrBAC

Mustapha Ben Saidi, Abderrahim Marzouk

Abstract--The challenge that arises by the arrival of cloud computing is to carefully control the data that are no longer in possession of the company alone, but may be in the hands of third parties (TTP). Managing user trust is a major concern related to the management of migrated data in a Cloud. Dealing with this issue, our paper contributes to this process by defining a security policy based on trust, followed by the description of a security protocol for a TTP monitor attempts to violations of this policy by users of an organization's cloud. This protocol is based on ordered policies established by the AS and assigned to each user during its connections to the cloud.

Keywords --- Security, Cloud, Access Control, TOrBAC, OrBAC.

I. INTRODUCTION

Cloud computing is a general concept that incorporates internet based (cloud) development, use and storage of computer technology. For example Google Apps, provides common business applications online that are accessed from a web browser, while the software and date are stored on the servers and cached temporarily on clients, tablet computers, notebooks, wall computers handles, sensors, monitors etc. In this context, as more and more information on individuals and companies is placed in the cloud while the Cloud is actually a fairly new and emergent technology with several open areas mainly related to security: remote storage, data dispersion, multi-location, isolation, risk exposure, data lost, abuse and malicious use, non-secure API, account or service diversion, etc.

Privacy, trust and access control are hence some of the most important security concepts met in Cloud systems.

In particular, access control is of vital importance in a Cloud environment since it is concerned with allowing a user to access a number of Cloud resources: who has access to what, when, how and under which conditions? An extensive research has been done in the area of access control in collaborative systems but few works are really dedicated to the cloud computing. Further examination is thus necessary, especially duo to this domain specificities and to the partial or weak fulfilment of security requirements in the Cloud.

Manuscript received on November, 2012.

Mustapha Ben Saidi. FST Settat University Hassan 1 Settat Departement of Mathematics and computers sciences Lab. MAI; Morocco.

Abderrahim Marzouk. FST University Hassan 1er Settat Departement of Mathematics and computers sciences; Lab MAI Morocco.

While current and emerging applications become more complex, particularly in the context of cloud, most security policies and existing models consider only a yes / no answer to requests for access to information or a service. Therefore, modeling, formalization and implementation permissions and prohibitions do not cover all the needs of all possible scenarios, particularly in the context of the cloud. In our recent work, we extended policies and access control models by the notion of "recommendation", in addition to permissions, prohibitions and obligations [8, 9]. This notion of recommendation is interesting but not sufficient in the context of the cloud.

It would be interesting to build a monitoring protocol access each subject (actor, user, process, etc.) depending on the model TOrBAC [1]. This model is based on an index of confidence decrease in real time based on malicious actions (violating the security policy).

In this article, we recall the fundamental principle of TOrBAC model and we announce our idea. In Section IV we define weighted actions. Then, in the fifth section, we set up group security policy ordered, and we explain our new protocol detailed in the sixth section. Finally, the last section presents our conclusions and perspectives.

II. TOrBAC

The main idea of our TOrBAC is to define a confidence index for each connected. This index will be initialized by the AS at T0. The user will be controlled and penalized following each violation or attempted violation of an action. But the rest of the entities in the model TOrBAC are exactly those OrBAC[10]. We can say that a model TOrBAC is an extended version of OrBAC with confidence index.

Basically, TOrBAC based on the OrBAC model and the confidence index. The latter is based on the following parameters:

- Trust T0:

The security manager assigns a confidence level T0 connect user about when creating accounts and sessions. The connected subject must ensure that T0 is constant during the connection because the value of the initialization may depend on the change curve of the confidence index initialized by T0; where T0 = confidence level affected by the security manager.

- Number of malicious attempt NMA "Number of malicious actions":

Management of malicious actions within the cloud is very important. Indeed, the implementation of the coefficient allows NMA attempts to control violations. Sanctions generated by incrementing the NMA are within the heart of our access control model TOrBAC. This parameter is an integer initialized to zero when creating the account, it is incremented (by 1) after each non-compliance with the security policies (e.g. malicious attempts). Obviously, after each attempted rape of policies, T0 decreases by a positive step. This sanction is not related only to NMA but also to the

frequency of connection and disconnection. Hence, there is a need to introduce metrics in this direction.

- Connection counter NC

The frequency of a user logs, indicates more information on the identity of the connected, when we compare this number with the normal average of these needs. This is an integer initialized with zero and incremented (by 1) after each connection. This number can bring several information that facilitates in their turn the trust management in as a broad environment such as Cloud. This counter is still very useful when combined with that of the disconnection.

- Counter disconnection ND

This is necessarily an integer less than or equal to N in normal cases. It counts the number of closures correct session; its importance is that to compare it with NC, so for a user who meets the security policy, the NC is equal to ND or NC=ND+1 in or if he is offline. In other words, if the NC > ND + 1 + K, where K> 0, then we can deduce that the system has already forced the disconnection of this user K times, after a period of idle connection. This behavior deserves punishment naturally; hence there is interest in including it in the calculation of our confidence level.

- Duration of passive connection DPC "Duration of passive connection"

The passivity of a session is normally not recommended in the cloud environment by touching the confidentiality of data to which it is entitled access. This is an index that reflects the carelessness of the user. This behavior can affect the confidentiality of information because it opens a window through this session, through which a person can do a consultation. This coefficient will link the logon necessarily to a continuous activity and legal identity connected. The penalty generated by this behavior is translated via the number of times or is forced via the disconnection of the session. Note NDPC as an integer that will be a part of the definition of our confidence index.

III. OVERVIEW

We propose in this paper a new protocol which allows an organization wishing to delegate a TTP "Trusted Third Party" [6] monitoring the activity of users on a cloud as shown in Figure 1. This protocol is based on both the concept of confidence index [1], and an order on the security policy assigned to a user. It is assumed that each user has initially a capital of trust and can have a succession of policies during its connections to the Cloud, he starting from the policy maximum to minimum. Each attempted violation of an action not permitted, this capital falls to take a threshold which depends on each user.

Typically, any action on an object is provided with a weight [2], which is a real number between 0 and 1. This weight may vary during the user activity on the Cloud or by reducing or increasing in a predefined manner by administrator system. And action could change state after raping her. Such action weight 0.7 can be transformed into an obligation, while a 0.4 weight action can, in turn, become a prohibition after one or more violations. All of these shares belong to a security policies P which is an element of the policy space can **P**.

Following the policy assigned to a user, it starts with a maximum policies and decreases always to a minimum policies. When a user reaches a minimum when the TTP policy assigns public policy that will keep for the rest of its business on the cloud, unless the AS comes to assign another policy with more rights (Figure 1).

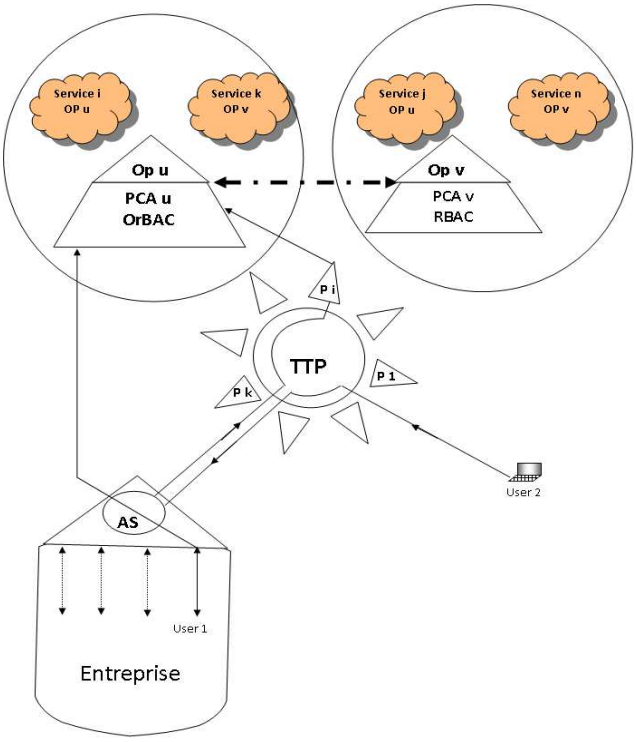


Figure 1: Access control architecture for trust.

Op: Operator.
PCA: Policy Control Access
AS: Administrator System
Pi: policy i
RBAC: Role based access control
OrBAC: Organization based Access

IV. WEIGHTED ACTIONS

We have seen in [1] that the shares weighted by a weight between 0 and 1 are either obligations or prohibitions or permissions, or recommendations [2]. These four types have limitations for critical information systems. In this article, we will share all the recommendations into two separate parts which are formed by the terms of weight respectively between 0 and 0.5 and is called pre-Prohibition and the weight between 0.5 and 1 and is called pre-Obligation. Then we distinguish in this article five types of weighted actions:

- The Prohibitions are actions zero weight.
- The Obligations are actions of weight 1.
- The Permissions are actions weight 0.5.
- The pre-Prohibition (or actions not recommended) actions that are weight between 0 and 0.5.
- The pre-Obligation (or recommendation) are actions in weight between 0.5 and 1.

Pre-Prohibition (resp. pre-Obligation) becomes prohibition (resp. Obligation) after a number of violations of the security policy.

Examples:

- Let α1 be the action “It is recommended that the user s1saves the file f1.doc in the working directory” with a weight of 0.6. The system tolerates s1 violates this rule (e.g. do the opposite) a number of times. Each violation increases the weight of a certain amount up to the value 1 and turn into an obligation.
- Let α2 be the action “It is recommended that the user s2 writes to the file f2.doc” with weight 0. 4 Whenever s2

can do this a number of times, and each time the weight decreases to a value of 0 and thus become a Prohibition.

When a user s performs an action α of an object o, then we will specify whether a Obligation(s, α, o), Prohibition(s, α, o), Permission(s, α, o), pre-Prohibition(s, α, o, w) or pre-Obligation(s, α, o, w) where w is the weight of action α.

We assume in the sequel that we have always:

i) ∀s∀α∀o∀w Pre-Prohibition (s, α, o, w) ===> Pre-Prohibition (s, α, o, w') with 0 < w'≤ w.

ii) ∀s ∀α∀o∀w Pre-Obligation (s, α, o, w) ===> Pre-Obligation (s, α, o, w'*) with* $w \leq w' < 1$.

V. SECURITY POLICY

A. *Definition:*

Security policy a set of weighted actions associated or assigned to a user or group of users. In the following, we denote by w(α, P) the weight of the action α belonging to the policy P.

B. *Order in the set of security policies:*

Definition 1:

Let P1 and P2 be two security policies. We say that P2<P1 holds if and only if:

- P1 and P2 contain the same weighted actions.
- Either there is a pre-Obligation α belonging to P1 (thus P2) such that w (α, P1) <w (α, P2), or there is a pre-Prohibition α belonging to P1 (thus P2) such that w (α, P1)> w (α, P2).

C. *Switching policies :*

Definition 2:

We say that a user S switches from a policies P to policies, denoted Switch (s, P, P') if:

- *P and P' contain the same actions.*
- *s violates a pre-Obligation or pre-Prohibition α belonging to P.*
- *P' is obtained from P by changing the weight of the action α.*
- *P and P 'are assigned successively to s by TTP.*

Corollary:

Switch (s, P, P') ===> P' < P.

Proposition 1: Let *P* be the set of policies assigned to a user during his various connections to the cloud. If *P* contains a security policy P containing only permissions, obligations and prohibitions, then P is minimal in *P* with respect to the order relation "<" on the set of security policies.

Proof: Suppose that there exists a policies *P'* belonging to *P* and containing the same actions as *P* but with different weights. Then P' contains at least one pre-Obligation or pre-Prohibition α.

Hence two cases only are possible:

Case 1: If α is a pre-Prohibition then w *(α, P)>w (α, P')=0.*

Case 2: If α is a pre-Obligation then we have *w (α, P') <w (α, P) = 1.*

In both cases, we obtain P <P '.

In the remainder of this article, we denote by PMIN(s) a minimal security policies assigned to a user s, by PMAX(s) a maximum security policies granted to the user at the first connection to the cloud and by PPUB the public security policies granted to any user who has only the permissions set by the AS.

VI.BUILDING A PROTOCOL FOR MONITORING ANY SECURITY POLICY

A. *Principal of the monitoring Protocol*

Note first that only the actions of different weights of 0.5 could be violated. To switch any user to a minimum policy after a series of violations, our proposed monitoring protocol consists of the following four rules:

Rule 1: For any violation of a prohibition or obligation, the weight remains constant for all connections unlike Confidence Index "section 2" drop which a fixed amount in advance by the AS.

Rule 2: For any violation of a pre-Prohibition, its weight and the index of user confidence down by amounts set by the AS. Such actions are transformed into prohibitions after a finite number of violations because their weight will eventually become zero**.**

Rule 3: For any violation of a pre-Obligation, weight undergoes an increase and the confidence index of the user experiences a decrease (on a scale set by the AS). Such actions are transformed into Obligation after a finite number of violations because their weight will eventually become equal to 1**.**

Rule 4: If the user reaches a minimum policy or if its confidence index reached a threshold set by the AS, and then automatically switch to public policy PPUB.

It follows that Protocol-Algorithm, with consists in applying the precedents rules.

B. *Algorithm*

Our protocol is performed according to the following algorithm:

Initialization

Assign each user his initial capital of trust and policy maximum PMA: s← PMA.

Process:

While (p≠PPUB) do

{

For any violation of a Prohibition or Obligation apply Rule1.

For any violation of a Pre-Prohibition apply Rule2.

For any violation of a Pre-Obligation apply Rule3.}

By construction, we know that the above algorithm will terminates eventually, since there is a finite strictly decreasing sequence for < of policies.

C. *Role of TTP in the context of Cloud :*

TTP must ensure (in real time, execution time) in respect of the security policy assigned to each connected. It applies the monitoring protocol described above for each connected user. The latter sees each violation, common security policy switch to a new or stricter policy finds its confidence index down. A user S can then pass on his connections a strictly decreasing security policies contains the same weighted *actions (P1, P2, …, Pk) with P1 = PMAX>P2>…..…> Pk =PMIN and Switch(s, Pi, Pi+1).*

D. *UML Modeling*

1- *Classes diagram*

To complement the UML model presented in [1] there now includes an entity "Violation" which can represent

attempts violations shares weighted by the users. Then introduced a new association "Violation" of the type (Subject, Action, Object). Violation(s, α, o) means:

User s violates action α on object o if and only if the action α is an obligation to object o and s has not met, or the action α is a prohibition for the object o and tried to perform either action α is a pre-Prohibition or pre-Obligation and tried to make its negation (or its inverse).

The associations *Assigns, Control and Modify* keep the same meaning as in [1]. Here, we redefine the association *Modify* using association *Switch*:

Modify (TTP, P, s) <==> Control(TTP ,s)∧

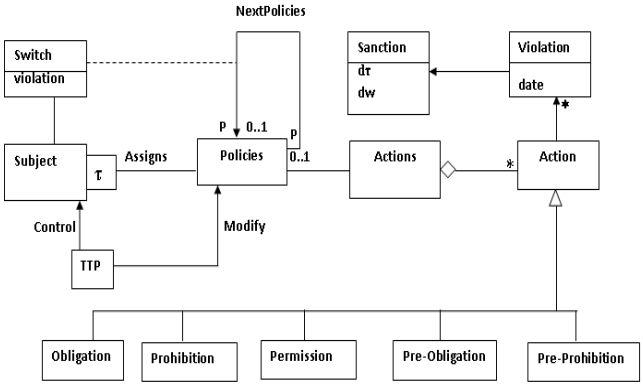
s.indexOfonfidence ( ) drop ∧

If ∃ P' such that P'≠ PMIN ∧

Switch (s, P, P') ∧

Assigns (TTP, s, P').

This results in the following UML diagram:



We see that:

- The link between policies P and his next policies P'(P'<P) is represented by the association "NextPolicies" (a policies has at most one following policies).
- Relationship "Switch" is modeled as an association class connected to the class 'Subject'.
- For each violation by a subject is a sanction characterized by two attributes dτ and dw respectively which represents the amount set by the AS to be subtracted from the current confidence index of the user and the amount to add or to subtract the weight of action violated.

2- The state diagram of a weighted action

An action can go through the following states: Permitted, Prohibited, Obliged, Pre-Prohibited and Pre-Obliged. Is represented in Figure 2 below, the various states of action and possible transitions between them:

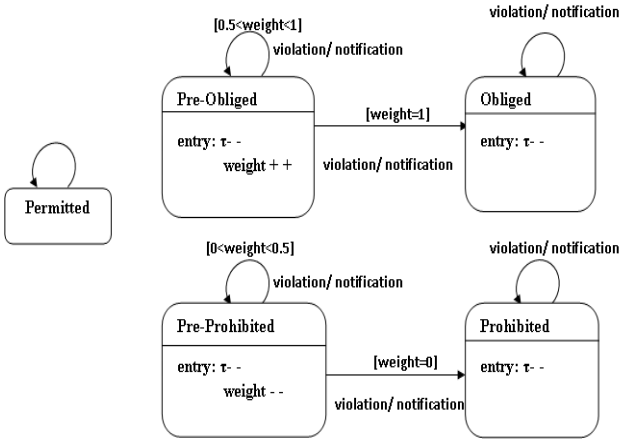


Figure 2: UML diagram transitions actions

We see that:

- Each violation confidence index decreases and the TTP is notified of the violation.
- After each violation of trust capital decreases until it reaches a threshold for each user, in which case the user is assigned public security policy PPUB.

E. Axiomes :

1. ∀s∀α∀o∀w
   Violation(s, α, o) <==>
   ¬Obligation(s, α, o) ∧
   Prohibition(s, α, o) ∧
   ¬Pre-Prohibition(s, α, o,w) ∧
   ¬Pre-Obligation(s, α, o, w)
2. ∀s∀α∀o
   Violation(s, α, o) ===>
   Confidence index s decline ∧
   The weight of α is modified.
3. ∀s∀α∀o
   ¬Obligation (s, α, o) ===>
   Violation(s, α, o) ∧
   Confidence index of s decline.
4. ∀s∀α∀o
   Prohibition (s, α, o) ===>
   Violation(s, α, o) ∧
   Confidence index of s decline.
5. ∀s∀α∀o∀w
   ¬ Pre-Prohibition (s, α, o,w) ===>
   Violation(s, α, o) ∧
   Confidence index of s decline ∧
   The weight of α decrease.
6. ∀s∀α∀o∀w
   ¬Pre-Obligation(s, α, o,w) ===>
   Violation(s, α, o) ∧
   Confidence index of s decline ∧
   The weight of α increase.

VII. CONCLUSION

In this article, we could dissect the broad topic of Cloud Computing and the security issue that is related. There are other areas of research that relate to the multiple challenges of the cloud. The goal of these efforts is to push scientific customers enjoy a new era where computing is a tool finally becoming ready and available on charge demand for use.

Given that cloud computing is a fusion of computers and telecommunications on both the technical and the business model, security research will therefore be active on all levels and therefore problems of information security continue to emerge from this vast environment.

In this paper we propose a security policy based trust TOrBAC adapted to the cloud and a process of real-time control of policy violations.

In perspective, this new idea can review the notification mechanism in critical information systems. We intend to develop a comprehensive model for application systems wishing to integrate computing clouds. Among the open slopes model also TOrBAC the updates authentication algorithms.

REFERENCES

[1] M. Ben Saidi, A. Abou El Kalam, A. Marzouk, TOrBAC: A Trust Organization Based Access Control model for Cloud computing systems, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-4, September 2012

[2] A. Abou El Kalam, "A Research Challenge in Modeling Access Control Policies: Modeling Recommendations", IEEE International Conference on Research Challenges in Information Science, 3-6 Jun 2008, Marrakech, Morocco.

[3] www.ORBAC.org

[4] Livre Cloud : Cloud Computing - 2ème éd - Une rupture décisive pour l'informatique d'entreprise : Guillaume PLOUIN.

[5] Commentary : Cloud computing a security problem or solution? P.G. Dorey a, *, A. Leite b a Royal Holloway, University of London, UK b KPMG LLP, UK

[6] **M. Ben Saidi, A. Abou El Kalam , A. Marzouk** Politique de contrôle d'accès au Cloud Computing Recommandation à base de confiance, JNS2, IEEE Xplore 2012. 0.1109/JNS2.2012.6249249 .

[7] Recherche Portio pour Colt TelecomGroup, 2009

[8] N. Essaouini, A. Abou El Kalam, A. Ait ouahman, "Modeling Security Policies with Recommendations", International Journal of Computer Science and Network Security (IJCSNS), octobre 2011, ISSN : 1738-7906, Volume Number: Vol.11, No.10, http://paper.ijcsns.org/07_book/201111/20111121.pdf .

[9] N. Essaouini, A. Abou El Kalam, A. Ait ouahman, "Access Control Policy: A Framework to Enforce Recommendations", International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 2 (5) , 2011, 2452-2463 , Septembre 2011, ISSN : 0975-9646, Pages: 2452-2463, disponible à http://www.ijcsit.com/docs/Volume%202/vol2issue5/ijcsit20110205128.pdf .

[10] A. Abou El Kalam, P. Balbiani, S. Benferhat, F. Cuppens, Y. Deswarte, R. El-Baida, A. Miège, C. Saurel, G. Trouessin "Organization-Based Access Control", 4th International Workshop on Policies for Distributed Systems and Networks (Policy'03), Côme, Italie, 4-6 june 2003, IEEE Computer Society Press, pp. 120-131.

[11] http://www.ijcsit.com/docs/Volume%202/vol2issue5/ijcsit20110205128.pdf .

[12] A. Abou El Kalam, P. Balbiani, S. Benferhat, F. Cuppens, Y. Deswarte, R. El-Baida, A. Miège, C. Saurel, G. Trouessin "Organization-Based Access Control", 4th International Workshop on Policies for Distributed Systems and Networks (Policy'03), Côme, Italie, 4-6 june 2003, IEEE Computer Society Press, pp. 120-131.

Mustapha Ben Saidi researcher security of information. In University Hassan I Morocco.holds a degree in higher education and telecommunications networks. Member of the association AMAN Morocco. His current research interests are Software Engineering, Software Security and Software Process Modeling adapted to Cloud computing.

Dr. Abderrahim Marzouk received his Ph.D (Computer Science) from University of Cean (France) in 1995. He has more than 15 years of experience in teaching Computer Science, JEE Technology and Web Applications. His current research interests are Software Engineering, Software Security and Software Process Modeling(UML, XML, OWL).