Analysis of Various RBAC and ABAC Based Access Control Models with Their Extension

Prajapati Barkha, Gurucharansingh Sahani
Student, Assistant Professor,
Computer Engineering Department, Sardar Vallabhbhai patel Institute of Technology, Vasad, India

Abstract: **In access control role based access control and attribute based access control are two most popular and widely used models. Both have their limitation which is complimentary to each other. Numbers of research are carried out that integrate the RBAC and ABAC model. There is need to develop the model that overcomes the limitation of both RBAC and ABAC. In this paper we are describing different types of integration of RBAC and ABAC as well as the various extensions of RBAC and ABAC. Also describing how this extension is advent ages over the pure RBAC and ABAC. Finally we compare different model by taking some of properties of RBAC and ABAC.**

Index Terms: RBAC, ABAC, RABAC, PFP

I. INTRODUCTION

RBAC is the standard and most important access control model and has been a most important research topic since last two decades. Role Based Access Control model provides a great way to fulfill the access control needs. An access control policy is a statement which specifies the rules about who can access the resources and how much access is given to each user. In RBAC main Focus is on Role. Main idea behind the RBAC is that a role is an intermediate module between users and permissions. In RBAC roles are assigned to the users (many-to-many assignments) and permissions are associates with each roles (many-to-many assignments), and thus indirectly assigns users to permissions.[8] Recently there has been rising concern about the limitations of RBAC, RBAC has certain limitation which has been met by researchers in two different Ways. First researchers have meticulously and creatively extended RBAC in number of directions. Also they integrate the RBAC with other access control model. Second they tries to develop a more general model, especially attribute-based access Control (ABAC), ABAC have the benefits of DAC, MAC and RBAC and also overcome the limitation of this model [9]. RBAC is used in situation where access is depends on role of the user in organization but it does not suitable in the situation where the contextual attribute are taken in to the consideration while making the access decision. Another limitation of RBAC is that if size of organization is large then in order to provide the finer grained access we have to define so many roles which leads to the role explosion problem. Also in RBAC permission is formed using the object identifier this is not fit in the situation where there is large number of object and it leads to role to permission explosion problem. Attribute-Based Access Control (ABAC) can be used as alternative to RBAC to overcome the limitation of RBAC. ABAC is more flexible than RBAC because it easily accommodates the contextual attribute while forming the permission. However, ABAC also has its own limitation such as it is complex than RBAC in term of policy review or policy modification visualization is difficult because there is no role so if we want to modify the policy then it becomes so difficult to determine that which group of user is affected by the modified policy. As discussed above, both RBAC and ABAC have their advantages as well as disadvantages. Both have features that are complimentary so that integration of RBAC and ABAC is become an important research topic. [10]

II. RELATED WORK

There are number of existing system that integrates the ABAC and RBAC in many different ways. We will discuss the each system in detailed.

Hui Qi et al [1] propose a model in which they preserve the full RBAC model. They use the ABAC as a constraint on user to role assignment and role to permission assignment. They dynamically adjusting the role that the user is associated with and the permission that the role is associated with by attribute based access control rules.

The new model can be expressed as:

$$U \xrightarrow{A_1,\ldots\ldots,A_n} R \xrightarrow{A'_1,\ldots\ldots,A'_m} P \qquad \ldots\ldots\ldots\text{equation (1)}$$

Ai and Aj in (l) are the constraint attributes of U –> R and R -> P respectively. They are not directly involved in U - > Rand R - > P mappings. Instead, they are used to filter the association relationships after the establishment of U - > R and R - > P mappings. As shown in figure 1 ABAC is used as constraint on user to role assignment and role to permission assignment.

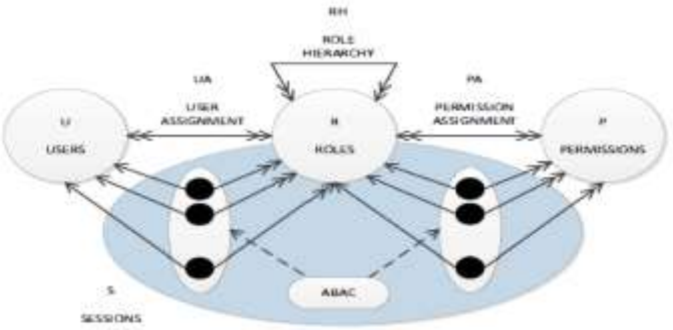


Figure 1: RBAC-ABAC Model [1]

Since full RBAC model is preserved it is simple to manage and Audit. It also provides fine grained access control because all dynamic and static attribute is considered during access control. The model is context aware because during role to permission assignment they taken the dynamic attribute such as time, location etc. in to account. Multi domain implementation of this model is somewhat difficult because first it is based on RBAC model second attribute they used as constrained during user to role and role to permission mapping May different in different domain that makes the filtering of user to role and role to permission assignment a difficult task. This model solves role explosion of RBAC and the access control rule explosion of ABAC. For example, if there are n attributes in the system, RBAC and ABAC will respectively generate 2^n roles and 2^n access control rules, while the new model will divide n attributes into x static attributes and n - x dynamic attributes, and then sets up 2^x roles for x static attributes and sets up 2^n-x access control rules for n - x dynamic attributes, so that the total will be reduced to 2^x + 2^n-x.but if no. of static attribute and no. of dynamic attribute is more than no. of role and rule is again increases hence there is still a problem of role and rule explosion.

Lawrence Kerr et al [2] represents a combined MAC and ABAC model. They take the classification, clearance and compartment of the MAC model as mandatory attribute in ABAC model. By taking classification, clearance and compartment as mandatory attribute they combine the traditional MAC model in ABAC model while preserving the flexibility of ABAC model. The classification of objects or clearance of subjects, as well as compartments is treated as attributes. This model has all the advantages of ABAC such as fine grained, context aware as well maintaining the basic principles of a MAC model. Since it also suffer from the limitation of ABAC such as difficult to audit, hard to visualize the policy modification.

Li Ma et al [3] propose a model in which they overcome the permission explosion problem of RBAC by using task as constraint. By assigning the appropriate role to the user RBAC provides the least privileges. However it is difficult to find the minimum number of role set. Also role hierarchy and inheritance leads to permission explosion problem because due to role hierarchy the senior role inherent the permission of junior role so they use the permission of their junior role and it may happen that they misuse their permission and hence principle of least privilege is not satisfied. To solve this problem task is taken as a kind of constraints introduces to the RBAC model. Activation of role should not be decided by user discretionarily but by the constraint of specific task assigned to user. They define four class: Class P (Private), Class S (Supervision), Class W (Workflow) and Class A (Approval for activity), where Class P means the access rights for the tasks are not inherited by senior roles; tasks in Class S are inherited by senior role and it is related to management or supervision; tasks in Class W are not inherited to senior roles and belong to a business process; and Class A has characteristics of Class S and Class W. Here as shown in figure 2 T stands for the set of tasks; $TR \subseteq T \times R$, stands for the set of many-to-many assignment relation between tasks and roles, and one task can be completed by multiple roles, and also one role can do many tasks; $TP \subseteq T \times P$, stands for the set of inclusion relation between tasks and permissions, which is a many-to-many relation, i.e. one task contains multiple permissions, and one permission can be contained by many tasks; TC stands for the set of task constraints to RH and PA relations; $TH \subseteq T \times T$, is a inclusion relation between tasks, for example a task t can be decomposed to two sub-tasks t1 and t2, then (t, t1), (t, t2)∈TH . Hence in this model task is used as constraint on role hierarchy also in role to permission assignment etc.

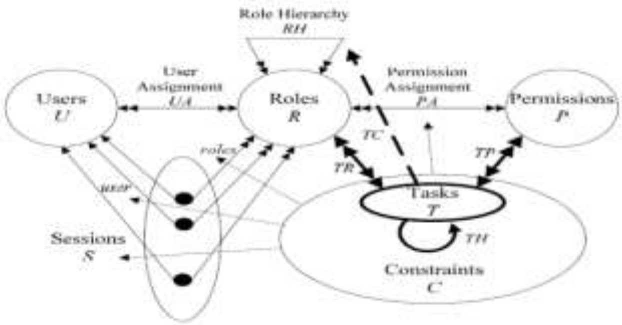


Figure 2: Task constrained RBAC model [3]

Since the model is based on RBAC so it has advantages of RBAC such as simple to manage ,easy auditing ,easy to visualize the policy modification because by changing the policy it is easily understood its effect on role and on user. But it suffers from the problem of role explosion because in order to provide fine grained access control they have to define large number of role. The model only restricts the role to permission explosion problem by using task as constraints but not role explosion problem.

Wenrong Zeng et al [4] introduce content based access control for content centric information sharing. The model is an extension to RBAC. In this model the access control decision is based on the content. In CBAC, each user has a RBAC rule to access the large set of data while this access is restricted by CBAC rule. CBAC rule provide the restriction on base set of data and allow to access only on some subset of data according to content similarity. Their goal is to build an access control system that taken in the account the textual content of data while making the access control decision. In this model they propose two phases 1) initial phase is authorization phase, in this phase a user has an access on some set of record which is called base set. 2) Second phase is content-based authorization phase, in this phase CBAC rule is applied which expand the base set. In which authorizations are depends on content similarity between requested records and base set records. The model is based on RBAC so it has advantages of RBAC such as easy auditing, easy to visualize the policy modification also associate the limitation of RBAC such as role explosion problem, provide coarse grained authorization. It has taken into an account the content information but fails to hold contextual condition during authorization. Law enforcement agency FBI require both content aware as well as context aware access control because access to the specific content is also depend on specific context so in such situation the model is not fit.

Xin Jin1 et al [5] propose a first model that integrate roles and attributes using the role centric methodology. This model extends the RBAC model with permission filtering policy. They use PFP to find out the available set of permission which contained user and object attribute as shown in figure 3.

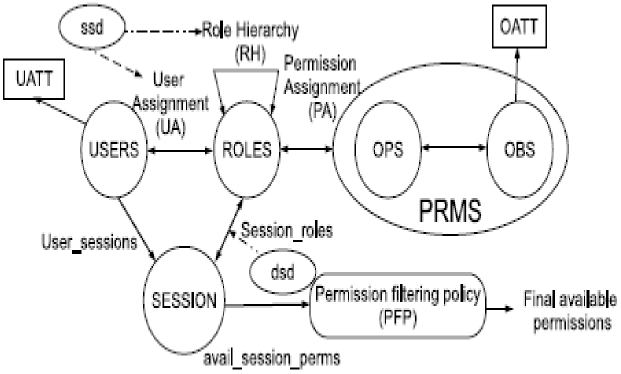


Figure 3: RABAC Model [5]

In figure 4 the maximum permission set available in a session is represented by the available session permission function. Filtering policy provide the constrained on these available permission set. There is set of filter function {F1,F2,F3…..Fn} for the purpose of providing a constraint. Each filter function is a Boolean expression contained user and object attribute. There is a Target Filter function that maps each data object to a subset of this filter functions. This mapping is based on the attributes expression which contained the object attribute called a condition which is used to determine whether each filter function is applicable or not. The applicable filter functions are invoked one by one for each of the permissions in available session permission. If any of the functions return FALSE, the permission is blocked and removed from the available permission set for this session. And At the end of this process, we get the final available permission set.

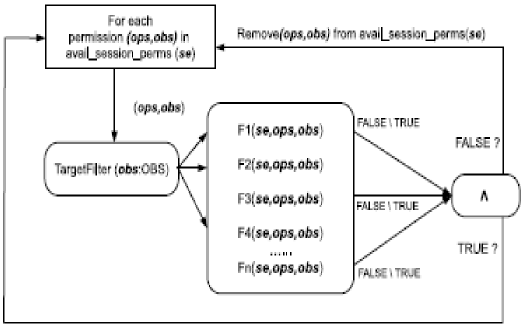


Figure 4: Permission Filtering Process [5]

Jingwei Huang et al [6] developed a model in two levels 1) aboveground 2) underground. The aboveground level is simple and standard RBAC model extended with environmental constraint. In underground level they use the attribute based policy to explicitly represent RBAC model hence creating RBAC model in aboveground. They use ABAC policy to automatically assign user to role and role to permission as shown in figure 5.

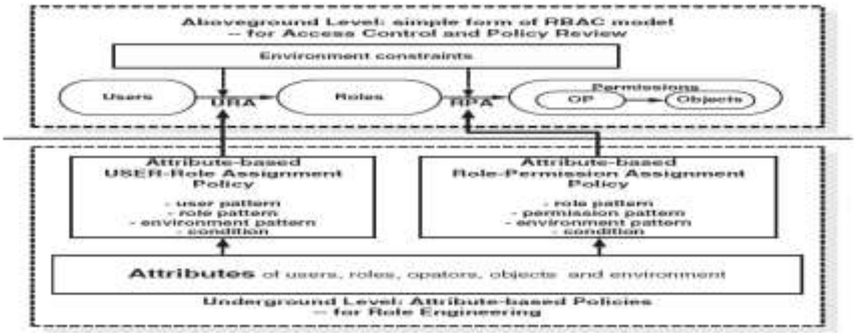


Figure 5: A two-layered framework integrating Attribute-based policies into RBAC [6]

The model provide fine grained access control but using large number of role hence there is role explosion problem and it is somewhat complex because for user to role and role to permission assignment they have to define the attribute based policy. They also consider the environmental condition so it is context aware .Auditing and policy visualization is difficult because role is assigned to user based on policy so that role can be change on changing policy and it is not clear that what set of user will be effected by changing in policy.

Qasim Mahmood Rajpoot et al [7] in their approach provides fine-grained access control mechanism that not only suitable for applications where access to resources is controlled by contents of the resources in the policy but it also takes contextual information into account while making the access control decisions. Their solution has the following key features: a) it allows to make context-aware access control decisions by associating conditions with permissions that are used to verify whether the required contextual information holds or not when a decision is made, b) it offers a content-based authorization system while keeping the approach role-oriented, in order to retain the advantages offered by RBAC. They achieve this by allowing specifying permissions using attributes of the objects rather than using only identifier. The entities in figure 6 such as users, roles, objects and operations have the same semantics as in RBAC. Users and objects in this model are associated with attributes too. They also incorporate the environment attribute to support the situation where contextual attribute are required in access control decision. The dotted-box in Figure 6 represents the modules of the architectural design to enforce this model.

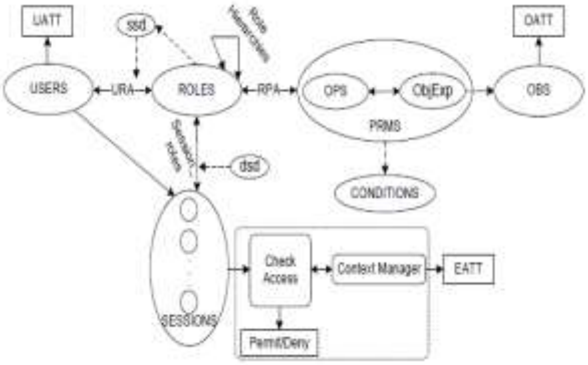


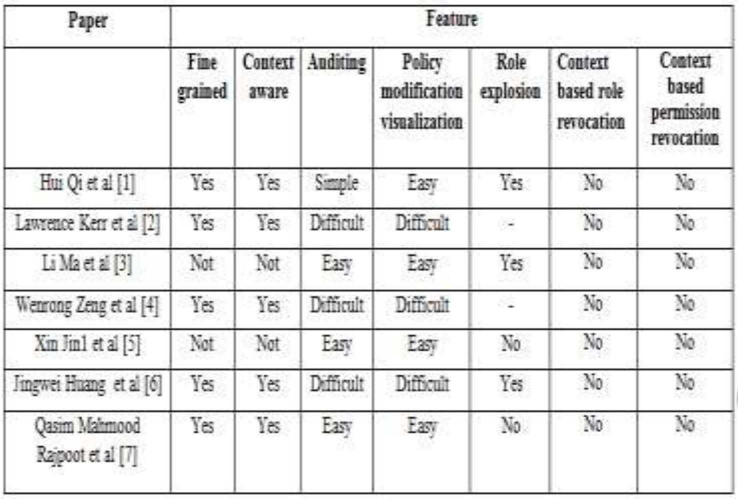
Figure 6: Attributes enhanced role-based access control (AERBAC) model [7]

Using the proposed approach, they provide fine-grained access control mechanism without creating a large number of roles. The model is context aware because it takes environmental condition while performing access control. It simpler to audit what permissions may be granted to a user because of being role-centric. It is relatively easy to visualize what the impact of adding is or removing a policy since policy specification is at the level of role. Therefore, a change in policy can affect only those users who are assigned to a role being modified.

III. COMPARATIVE ANALYSIS

As described earlier each model has their limitation and advantages .We are comparing above model by taking complimentary property of ABAC and RBAC. Here we are considering six properties fine grained, context aware, auditing, policy modification visualization, Role explosion, multi domain implementation for comparisons of different model described above. Table 1 shows which paper support which properties and which not.

Table 1: comparisons of different model



Paper	Feature						
	Fine grained	Context aware	Auditing	Policy modification visualization	Role explosion	Context based role revocation	Context based permission revocation
Hui Qi et al [1]	Yes	Yes	Simple	Easy	Yes	No	No
Lawrence Kerr et al [2]	Yes	Yes	Difficult	Difficult	-	No	No
Li Ma et al [3]	Not	Not	Easy	Easy	Yes	No	No
Wenrong Zeng et al [4]	Yes	Yes	Difficult	Difficult	-	No	No
Xin Jin1 et al [5]	Not	Not	Easy	Easy	No	No	No
Jingwei Huang et al [6]	Yes	Yes	Difficult	Difficult	Yes	No	No
Qasim Mahmood Rajpoot et al [7]	Yes	Yes	Easy	Easy	No	No	No

IV. CONCLUSION

Attribute-based access control (ABAC) and role-based access control (RBAC) are now a days the two most popular and widely used access control models. Yet, they both have known limitations and other features complimentary to each other. Number extension is carried out for both RBAC and ABAC but no one overcome the limitation of each other fully. Hence there is need of developing the model that overcomes the limitation of both ABAC and RBAC.

V. REFERENCES

[1] Hui Qi, Hongxin Mat, Jinqing Li and Xiaoqiang Di " Access Control Model Based on Role and Attribute and Its Applications on Space-Ground IntegrationNetworks" IEEE 2015.
[2] Lawrence Kerr, Jim Alves-Foss "Combining Mandatory and Attribute-based Access Control" IEEE 2016
[3] Li Ma, Yanjie Zhou, and Wei Duan "Extended RBAC Model with Task-Constraint Rules" Springer-Verlag Berlin Heidelberg 2014
[4] Wenrong Zeng, Yuhao Yang, and Bo Luo" Content-Based Access Control: Use Data Content to Assist Access Control for Large-Scale Content-Centric Databases" IEEE International Conference on Big Data 2014
[5] Xin Jin, Ravi Sandhu, and Ram Krishnan" RABAC: Role-Centric Attribute-Based Access Control" Springer-Verlag Berlin Heidelberg 2012
[6] Jingwei Huang, David M. Nicol, Rakesh Bobba and Jun Ho Huh" A Framework Integrating Attribute-based Policies into Role-Based Access Control" SACMAT'12, June 20–22, 2012
[7] Qasim Mahmood Rajpoot, Christian Damsgaard Jensen and Ram Krishnan" Attributes Enhanced Role-Based Access Control Model" Proceedings of the 12th International Conference on Trust, Privacy and Security in Digital Business (TrustBus'15). (pp. 3-17). Springer.

[8] Dipmala Salunke, Anilkumar Upadhyay, Amol Sarwade, Vaibhav Marde, Sachin Kandekar ” A survey paper on Role Based Access Control” International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 3, March 2013
[9] Xin Jin, Ram Krishnan and Ravi Sandhu” A Unied Attribute-Based Access Control Model Covering DAC, MAC and RBAC”
[10] Qasim Mahmood Rajpoot, Christian Damsgaard Jensen and Ram Krishnan” Integrating Attributes into Role-Based Access Control”