

Proposed Scenario-Driven Flexible Role-Based Access Control (fRBAC) Model: Role Reengineering, Policy Compliance and Amendment

Bodong Xu
bxu@emich.edu
College of Technology
Eastern Michigan University
Ypsilanti, MI 48197, USA

Table of Contents:

ABSTRACT.....	3
INTRODUCTION	4
PROPOSED SCENARIO	5
PREMINARY RESEARCH: fRBAC MODEL.....	7
• Definitions in RBAC.....	7
• Definitions in fRBAC	7
• fMW	9
• Processes	10
• Role Reengineering: Temporal and Permanent	11
• Location Based.....	11
• User Conflict.....	11
CONCLUSION.....	12
PROPOSED FUTURE RESEARCHES	13
• <i>cvS</i> Calculation and Determination.....	13
• Delegation approaches in Role Reengineering	13
• Implementation of Role Reengineering	13
REFERENCES	14

ABSTRACT

Role-Based Access Control (RBAC) is a family of policy-neutral and flexible access control models that are sufficiently powerful to simulate Discretionary Access Control (DAC) and Mandatory Access Control (MAC). In Role-based Access Control (RBAC), permissions are associated with roles and users are made members of those roles. RBAC's motivation is to simplify administration of authorization. This paper discusses a scenario in which an absence of a user triggers the developed Flexible Role-Based Access Control (fRBAC), providing a temporary solution to the scenario, which further leads to potential permanent policy compliance and/or amendment. Permission may then be granted to the alternative user performing that role's duties, through Role Reengineering. Delegation approaches, including Permission-Based Delegation Model (PBDM), Delegation of Authority Model (DAM), Cascaded Delegation, and a Law-governed interaction mechanism are being mentioned and designated to future research topics in this paper. Scenario-Driven Role Reengineering, a Temporal RBAC concept, and Location-based Access Control (LBAC) techniques are integrated into the proposed fRBAC model.

INTRODUCTION

In order to tackle the solution of providing flexibility within current RBAC, under the scenario of the need for a substitute user to perform the functional duties of an absentee user, and to keep potential compromises of accessed Objects to a minimum, numerous delegation approaches are mentioned and designated to future research topics. These approaches include Permission-Based Delegation Model (PBDM), Delegation of Authority Model (DAM), Cascaded Delegation, and a Law-governed interaction mechanism. The fRBAC model is used to derive efficient implementation of accessibility, with certain restrictions, to otherwise inaccessible Objects for certain users. Those users must demonstrate a current need, under the special circumstance of absence of the user who is permitted access to the requested Objects, with the least compromise to the Objects' integrity and confidentiality. Policy compliance and enforcement would remain operational while the granted temporary accessibility is active. The model is based on the idea of incrementally maintaining the result of the user's credibility, where a new method is introduced and used for systematically deriving increment rules. Numerous variants of efficient implementations are precisely calculated with their complexities as well as for a straightforward implementation based on the specification. This proposed fRBAC integrates Scenario-Driven Role Reengineering, a Temporal RBAC concept, and Location-Based Access Control (LBAC) techniques.

PROPOSED SCENARIO

Mike is a Computer Science student at the College of Arts and Sciences (CoAS). Nancy is the Computer Science (CS) department secretary; she has the access to override the restrictions on all the CS courses. Bonnie is the CoAS secretary; she has the access to override the restriction on all the courses offered within the college, including the CS department and other departments such as the Math department. Mike now needs to take a Math course in order to fulfill his program of study. Normally, Mike should ask Bonnie to give him an override to register for the course. The course starts later today, and Mike is a conscientious student who would not and cannot miss one class. Unfortunately, Bonnie called in sick today. Mike then goes to seek Nancy's help. Under ordinary circumstances, Nancy would not be able to help Mike.

The proposed fRBAC is then triggered by this special scenario to provide flexibility and solve the existing issue.

Nancy is very nice, and willing to help Mike. Nancy goes to the University's Banner system by using her office computer. Nancy then attempts to give Mike an override. The Banner system, which uses RBAC, immediately realizes that Nancy doesn't have permission to access this Object. Under the fRBAC model, the system would then realize that the only user who has those permissions is Bonnie (the Math department secretary has the access to override only for Math students but not students from another department), and Bonnie is absent for the day because the system knows that she did not log in. The Banner system then checks if Nancy's attempt to access the override is initialized from her assigned workstation. Normally the system would not check if Nancy is using her assigned workstation, she can even access the Banner system from home. But this is different, because Nancy is trying to access something to which she doesn't have permission. The system then prompts a warning, asking Nancy: "Are you sure that you want to be granted access to this override?" Nancy answers "Yes, I want to have this permission." The system then further considers other factors of the scenario, and then finally makes a decision that gives permission to Nancy on a temporary basis. The system launches an application, and opens up a user interface. Nancy then continues her request for Mike's override.

Nancy can then successfully override the restriction of the course and Mike is happily registered for his Math class. It all seems the same to Nancy, except the system asked for her confirmation. What Nancy didn't know is that the latest launched application quarantined her attempt to gain access within a secured area. Nancy did not actually have full access to this part of the system. Nancy only exchanged information through middleware. The middleware checked and filtered Nancy's request, launched a request of access itself, gained the access to the Objects, checked and filtered the data it got from the Banner system again, and then passed it along to Nancy. Luckily, Nancy didn't do anything wrong, otherwise the middleware would limit the transaction of the information.

Jenny heard from Mike that Nancy was able to help, so she asked Nancy for the same kind of help; then Belinda, then Henry, and Karl. Each time, the system (actually the middleware) asked Nancy to confirm that she wanted to grant an override. When Karl asked Nancy the same thing, although Nancy did not notice, the system, after five compliant interactions by Nancy, did not ask for her confirmation on “Are you sure?”

PREMINARY RESEARCH: PROPOSED fRBAC MODEL

• Definitions in RBAC

Subject = A person or automated agent, interchangeable with User in this scenario

- Nancy

Role = Job function or title which defines an authority level

- Nancy's Role: Secretary of Computer Science Department in different matters
- Bonnie's Role: Secretary of College of Arts and Sciences in different matters

Permissions = An approval of a mode of access to a resource

- Permission of Nancy's Role in course override matter: override all CS courses
- Permission of Bonnie's Role in course override matter: override all CoAS courses including CS and Math

Object = Data or information

- Access to override the courses

• Definitions in fRBAC

$f_vR: f_vR \in [0, 1]$

The flexibility value of a Role. Each Role has its own predetermined f_vR value, which never changes. Generally, a Role associated with more permissions of access would have a lower value, and vice versa. In this scenario, the f_vR of the Role as a secretary of CoAS is lower than of the Role as a secretary of CS.

$cvS: cvS \in [0, 1]$

The credibility value of a Subject. cvS increases/decreases associated with the Subject's activities, behaviors, and credibility over time. In this scenario, Nancy's cvS increases over time when she helps Mike, Jenny, Belinda, Henry, and Karl, because of her numerous activities, good behaviors and credibility. The increase of her cvS is one of the factors that triggers the middleware to no longer ask her for confirmation. The detail method of calculating cvS is not discussed in this paper and will become a future research topic.

$f_vS: f_vS \in [0, 1]$

The flexibility value of a Subject. Each Subject can have multiple Roles. Generally, the more Roles that a Subject is associated with, the higher value his f_vS becomes. f_vS is also associated with cvS .

$$f_vS = cvS \cdot (1 - f_vR_1 \cdot f_vR_2 \cdot f_vR_3 \dots f_vR_{n-1} \cdot f_vR_n)$$

$fvO: fvO \in (0, 1]$

The flexibility value of an Object. Each Object has its own predetermined fvO value, which never changes. The more sensitive the Object, the higher the value fvO becomes. In the scenario, the Object overriding the course has the fvO value of 0.5. An Object staff's Social Security Number would be less than 0.5. fvO cannot be 0. An Object with the $fvO = 0$ would make such an Object absolutely inaccessible.

$SLi: SLi \in [0, 1]$

Security Level index. Each system has its predetermined SLi . The higher the SLi , the more secure the system becomes. In the scenario, the Banner system would have a $SLi = 0.5$. A pharmaceutical company might have a $SLi = 0.2$ because of the higher security requirements for intellectual property protection. In the case of $SLi = 0$ such as in a military system, the RBAC then simulates a MAC. The idea of fRBAC is to provide better information availability while minimizing potential possibility of compromise of information integrity and confidentiality. SLi is the key to balance such tradeoffs.

$As: As \in \{0, 1\}$

Absentee switch. In the scenario, $As = 1$ because Bonnie called in sick. Otherwise $As = 0$. This would ensure that only when no active (logged in) user can gain access to the Object, the possibility of fRBAC granting access to less privileged users then becomes real. This also solves the issue of Role redundancy. Again in the scenario, if Bonnie decides to come back and log in to the Banner system, then $As = 0$, and it is no longer possible for Nancy to access the override privilege. Mike would have to ask Bonnie for the override.

$Ls: Ls \in \{0, 1\}$

Location switch. In the scenario, it is only when Nancy attempts to access the Banner from her assigned workstation that $Ls = 0$. She is not able to access the Banner system from any other physical location. This is another measurement to mitigate the potential compromise of information integrity and confidentiality.

$fv: fv \in [0, 1]$

fv is the flexibility value of the scenario, to determine whether fRBAC will grant access to the Subject and on what level. This will be discussed in the paper later.

$$fv = As \cdot (1 - Ls) \cdot SLi \cdot fvS \cdot fvO, \text{ or}$$

$$fv = As \cdot (1 - Ls) \cdot SLi \cdot cvS(1 - fvR_1 \cdot fvR_2 \cdot fvR_3 \dots fvR_{n-1} \cdot fvR_n) \cdot fvO$$

FV: $FV \in [0, 1]$

FV is a pre-set threshold value for each system. The more secure a system, the higher the *FV* will be to be assigned.

RrTm: *RrTm* is a non-negative integer. $RrTm \in \{0, Z^+\}$

Role Reengineering Timer. Each Role has its own pre-set *RrTm*, which represents the time period between Role Assignment and Role Revocation, in seconds. *RrTm* decreases by 1 for each second time cycle.

RrPs: $RrPs \in \{0, 1\}$

Role Reengineering Permanent Switch. *RrPs* has an initial value of 0.

RrT: $RrT \in (0, \infty)$

RrT is the value of the Role Reengineering Trigger.

$$RrT = ((RrTm \neq 0) \text{ OR } (RrPs)) \cdot (fv/FV), \text{ or}$$

$$RrT = ((RrTm \neq 0) \text{ OR } (RrPs)) \cdot ((As \cdot (1-Ls) \cdot SLi \cdot cvS(1-fvR_1 \cdot fvR_2 \cdot fvR_3 \dots \\ \dots fvR_{n-1} \cdot fvR_n) \cdot fvO)/FV)$$

when $RrT \geq 1$, trigger or keepalive Role Reengineering

when $RrT < 1$, drop or nokeepalive Role Reengineering

RrPV: $RrPV \in (1, \infty)$

Role Reengineering Permanent threshold Value. *RrPV* has a pre-set value for each Subject.

when $RrT \geq RrPV$, let $RrPs = 1$

when $RrT < RrPV$, let $RrPs = 0$

fMW:

The middleware in the fRBAC model.

- **fMW**

A specific scenario triggers the fRBAC to launch its fMW in order to handle a series of actions, which can provide flexibility in Access Control. The fMW then monitors the specific User's activities, continually auditing his flexible access and accumulating the calculation of

fv, to later decide on policy compliance and/or amendment. The processes used by the fMW include Role Reengineering, Role Assignment, and Role Revocation. These processes deal with issues such as separation of duty and least privilege. The processes are also intended to solve Role conflicts and privilege conflicts, by determining policy compliance and/or amendment. However, due to the limitations of the author's knowledge in these matters, this paper refers to them as future research topics.

- **Processes**

S attempts to access O^+ (*S* is a Subject, O^+ is an Object which *S* does not have permission to access)

fRBAC determines that *S* does not have access to O^+ , then launches **fMW**

fMW gathers and collects the values of variants, then calculates

$$RrT = ((RrTm \neq 0) \text{ OR } (RrPs)) \cdot ((As \cdot (1 - Ls) \cdot SLi \cdot cvS(1 - fvR_1 \cdot fvR_2 \cdot fvR_3 \dots fvR_{n-1} \cdot fvR_n) \cdot fvO) / FV)$$

when $RrT < 1$, denies permission to the Subject, fRBAC ends

when $RrT \geq 1$, initiates Role Reengineering, grants permission to the Subject.

For each second (a pre-determined time unit in place of second should be more accurate here),

updates cvS , $RrPs$, n , and $fvS = cvS \cdot (1 - fvR_1 \cdot fvR_2 \cdot fvR_3 \dots fvR_{n-1} \cdot fvR_n)$, etc.

when $RrPs = 1$, fRBAC ends, Role Reengineering becomes permanent.

recalculates RrT

when $RrT \geq 1$, keepalive Role Reengineering

when $RrT < 1$, revoke permission of the Subject, fRBAC ends

- **Role Reengineering: Temporal and Permanent**

When under certain conditions, the fMW decides to initiate the Role Reengineering, the processes include:

- Allocate Roles with adequate permission to access the Object
- In the event of such Roles' nonexistence within current Security Policy, such Roles must be created associated with *RrPs* to be assigned to the Subject. When fMW revokes the permission to the Subject, fMW also discards such Roles. When *RrPs* = 1, Role Reengineering become permanent. In this process, the Security Policy has been amended.
- In the event of fMW being able to locate existing Roles within current Security Policy to be assigned to the Subject, no new Roles are being created, the Security Policy has only been recompiled.
- Assign allocated Roles to the subject with the possible consequences of Role Revocation and Role Assignment becoming permanent.
- *RrTm* determines the length of the temporal period of the Role Reengineering. Role Assignment starts the countdown of *RrTm*, and *RrTm* terminates Role Reengineering and initiates Role Revocation along with other factors. This implements Role Reengineering on a temporary basis.
- *RrPs* is the switch in the event of considering all the factors such as *cvS*, *fvS*, *As*, *Ls*, to be able to enable Role Assignment to become permanent.

- **Location Based**

Ls allows the possibility of Location-based control. A decision can be made by the System Administrator to overwrite *Ls*, by analyzing system security requirements, in conjunction with Location-based sensitivity. $Ls \in \{0, 1\}$ can be further enhanced to $Ls \in [0, 1]$, which reflects the physical and/or logical distance, such as hop counts and number of segments, from the location where the initial attempt occurs.

- **User Conflict**

User conflict defines that a pair of users should not be assigned to the same role. *As* ensures that only in the event of a user's absence, fRBAC is then able to assign permission to a less privileged user. *As* also addresses the issue of Role redundancy.

CONCLUSION

RBAC is a model and framework for controlling user access to resources, based on their roles. RBAC continues to be an evolving body of knowledge, with its scalability and inheritance. This allows for further research, such as the fRBAC that is introduced by this paper. This paper demonstrates a scenario-driven RBAC model, integrated with Role Reengineering, a Temporal RBAC concept, and a Location-Based Access Control (LBAC) technique, implemented by a proposed fMW, in furtherance of precisely calculating numerous variant factors with their complexities. It is this paper's conclusion that the fRBAC model's intention has the ability to balance information confidentiality, integrity, and availability tradeoffs, by minimizing potential compromises of security, while still providing flexibility in Role-Based Access Control.

PROPOSED FUTURE RESEARCH

- ***cvS* Calculation and Determination**

The fRBAC model is based on the idea of incrementally maintaining the result of the user's credibility. The *cvS* value increases/decreases associated with the Subject's activities and behaviors, calculating credibility over time. A precise and accurate methodology in calculating and determining *cvS* will further complete and improve fRBAC modeling.

- **Delegation approaches in Role Reengineering**

Delegation denotes that a user can give all or a portion of his authority to someone else. When a user is absent, his job functions need to be maintained by others. This requires that another user be delegated the authority to perform the absent user's functions. The question of whether fRBAC can assign the role of such a user and, if so, the implementation methodology of such delegation remains for future research.

- **Implementation of Role Reengineering**

The implementation in fRBAC of allocating, selecting, creating Roles to be assigned, the revocation of the session and discarding of the temporal Role, and decision making of Role Assignment becoming permanent, reflecting Policy Compliance and Amendment, would all call upon future research on Artificial Intelligence.

REFERENCES

- Andreas Schaad, J. M. (2001). The role-based access control system of a European bank: a case study and discussion. *ACM Workshop on Role Based Access Control, Proceedings of the sixth ACM symposium on Access control models and technologies* (pp. 3-9). Chantilly, Virginia, United States: ACM.
- Chunxiao Ye, Y. F. (2004). An attribute-based-delegation-model. *ACM International Conference Proceeding Series; Vol. 85, Proceedings of the 3rd international conference on Information security* (pp. 220-221). Shanghai, China: ACM.
- Claudio A. Ardagna, M. C. (2006). Supporting location-based conditions in access control policies . *Conference on Computer and Communications Security, Proceedings of the 2006 ACM Symposium on Information, computer and communications security* (pp. 212-222). New York, NY, USA: ACM.
- Elisa Bertino, P. A. (2001). TRBAC: A temporal role-based access control model. *ACM Transactions on Information and System Security* , 4 (3), 191-233.
- Emin Gün Sirer, K. W. (2002). An access control language for web services. *Symposium on Access Control Models and Technologies, Proceedings of the seventh ACM symposium on Access control models and technologies* (pp. 23-30). Monterey, California, USA : ACM.
- Ferraiolo, D. F. (2001). An argument for the role-based access control model. *ACM Workshop on Role Based Access Control, Proceedings of the sixth ACM symposium on Access control models and technologies* (pp. 142-143). Chantilly, Virginia, United States: ACM.
- Gustaf Neumann, M. S. (2002). A scenario-driven role engineering process for functional RBAC roles. *Symposium on Access Control Models and Technologies, Proceedings of the seventh ACM symposium on Access control models and technologies* (pp. 33-42). Monterey, California, USA : ACM.
- He Wang, S. L. (2006). Delegation in the Role Graph Model. *Symposium on Access Control Models and Technologies, Proceedings of the eleventh ACM symposium on Access control models and technologies* (pp. 91-100). Lake Tahoe, California, USA : ACM.
- Jacques Wainer, A. K. (2005). A fine-grained, controllable, user-to-user delegation method in RBAC. *Symposium on Access Control Models and Technologies, Proceedings of the tenth ACM symposium on Access control models and technologies* (pp. 59-66). Stockholm, Sweden : ACM.
- Jing Jin, G.-J. A. (2006). Role-based access management for ad-hoc collaborative sharing . *SACMAT '06: Proceedings of the eleventh ACM symposium on Access control models and technologies* (pp. 200-209). New York, NY, USA: ACM.

- Joon S. Park, K. P. (2004). A composite rbac approach for large, complex organizations . *Symposium on Access Control Models and Technologies, Proceedings of the ninth ACM symposium on Access control models and technologies* (pp. 163-172). Yorktown Heights, New York, USA: ACM.
- M. A. C. Dekker, J. G. (2007). Extended privilege inheritance in RBAC. *Conference on Computer and Communications Security, Proceedings of the 2nd ACM symposium on Information, computer and communications security* (pp. 383-385). Singapore : ACM.
- Maria Luisa Damiani, E. B. (2007). GEO-RBAC: A spatially aware RBAC. *ACM Transactions on Information and System Security (TISSEC)* , 10 (1), Article No.2.
- Ravi S Sandhu, E. J. (1996). Role-Based Access Control Models. *IEEE Computer* , 29 (2), 38-47.
- Roberto Tamassia, D. Y. (2004). Role-based cascaded delegation. *Symposium on Access Control Models and Technologies, Proceedings of the ninth ACM symposium on Access control models and technologies* (pp. 146-155). Yorktown Heights, New York, USA: ACM.
- Steve Barker, P. J. (2003). Flexible access control policy specification with constraint logic programming. *ACM Transactions on Information and System Security (TISSEC)* , 6 (4), 501-546.
- Trent Jaeger, J. E. (2001). Practical Safety in Flexible Access. *ACM Transactions on Information and System Security (TISSEC)* , 4 (2), 1580190.
- Tuan-Anh Nguyen, L. S. (2007). Flexible and Manageable Delegation of Authority in RBAC. *AINAW, Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops - Volume 02* (pp. 453-458). Washington, DC, USA: IEEE Computer Society.
- Xinwen Zhang, S. O. (2003). PBDM: a flexible delegation model in RBAC. *Symposium on Access Control Models and Technologies, Proceedings of the eighth ACM symposium on Access control models and technologies* (pp. 149-157). Como, Italy : ACM.
- Xinwen Zhang, Y. L. (2005). An Attribute-Based Access Matrix Model. *Symposium on Applied Computing, Proceedings of the 2005 ACM symposium on Applied computing* (pp. 359-363). Santa Fe, New Mexico: ACM.
- Xuhui Ao, N. H. (2004). On the role of roles: from role-based to role-sensitive access control. *Symposium on Access Control Models and Technologies, Proceedings of the ninth ACM symposium on Access control models and technologies* (pp. 51-60). Yorktown Heights, New York, USA : ACM.