워크플로우 환경에서 요구되는 확장된 RBAC 모델에 관한 연구

최성용[0] 정병수
경희대학교 전자계산공학과
sychoi@jupiter.kyunghee.ac.kr jeong@nms.kyunghee.ac.kr

A study on model of extended RBAC for WorkFlow

Seong-Young Choi[0] Byeong-Soo Jeong
Dept. of Computer Science, Kyunghee University

요 약

오늘날의 컴퓨터 환경은 여러 곳에 산재해 있는 자원의 공유를 위한 분산 환경을 지향하고 있다. 클라이언트-서버 환경은 바로 이런 분산 컴퓨터 환경을 잘 구현한 실제 사례 중의 하나이다. 그리고 이러한 분산환경을 지원하는 근간인 유닉스와 같은 운영체제는 여러 명의 사용자가 여러 업무를 동시에 수행하도록 해주는 멀티 유저, 멀티 태스킹 기능을 지원하고 있다. 이러한 다중 사용자 환경에서 원할한 정보의 공유 및 보호가 이루어지기 위해서는 해당 자원에 접근하기 위한 어떤 규칙이 필요하다. 또한 관리해야 할 조직들의 기능이 다양해지고 그 규모가 커짐으로써 조직의 특성에 맞는 보안 정책의 구현 및 원할한 정보 흐름을 위해 역할기반 접근제어가 접근기술로서 현재 많은 관심의 대상이 되고 있다. 특히 역할기반 접근제어 기술은 현대의 상업용 환경에서 특히 가치가 있는 다른 형태의 정책이다. 본 논문에서는 이러한 역할기반 접근제어를 분석한 결과를 토대로 현재 많은 이슈가 되고 있는 전자상거래 및 워크플로우 환경에 적합한 확장된 역할기반 접근제어 모델에 관하여 연구하고자 한다.

1. 서론

National Institute of Standards and Technology (NIST)의 최근 연구에 의하면 기존의 접근제어의 모델에 비하여 역할기반 접근제어(RBAC;Role Based Access Control) 모델이 상업적이고 행정적인 분야에서 가장 적합하다고 보여주고 있고 이 연구는 많은 조직들이 각 사용자가 조직 내에서 차지하는 역할로서 접근제어 결정에 기반을 둔다는 것을 보여준다[1-4]. 그러나 RBAC 개념의 유용성에도 불구하고 RBAC 모델이 의미하는 것을 정확히 충족시키는 제품은 거의 없다. RBAC 모델은 여러 학자와 시스템 개발자들에 의해 서로 다른 방법으로 해석되고 구현될 수 있으며 다른 접근제어 방법과 같이 사용될 수 있다[4]. RBAC 모델이 큰 조직사이의 정보흐름과 접근제어에는 용이하지만 보안에 있어서는 강제적 접근제어(MAC; Mandatory Access Control) 방법이 더 우수하다. 따라서 본 논문에서는 대규모의 복잡한 조직들로 구성되어 있고 지역이나 전역으로 분산되어 있는 그래서 보안 관리 측면에서의 용이성, 비용 효율성, 유연성 등을 고려해야 하는 WorkFlow 및 EC(E-Commerce) 환경의 특성과 요구사항들을 고려했을 때, 접근제어 모델로서, 융통성 있는 관리를 위해서는 RBAC를 적용하고, 중요한 자료에 있어서는 MLS(Multi-Level Security)의 대표적 접근제어 모델인 수정된 BLP 모델을 적용한다. 즉, RBAC & MLS 결합 모델을 통하여 WorkFlow 및 EC 특성에 맞는 접근제어 모델을 제한한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 전통적인 접근제어 방법과 최근의 상업적이고 행정적인 분야에서 가장 적합한 것으로 알려진 RBAC에 대해 살펴본 후 3장에서는 대규모의 복잡한 조직의 구성과 보안관리 측면에서의 용이성, 비용 효율성, 유연성 등을 고려해야 하는 WorkFlow 및 EC 환경의 특성과 요구사항에 맞는 확장된 RBAC 모델 설명 및 MLS와의 결합 적용원리를 제안하였고 4장에서는 결론 및 향후 연구 과제를 간략히 소개함으로써 끝을 맺는다.

2. 관련 연구

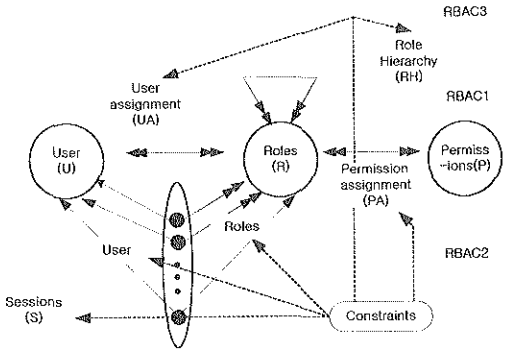
2.1 DAC와 MAC

임의적 접근제어(DAC : Discretionary Access Control)은 접근을 요청한 사용자의 신원(Identification)에 근거를 두고 있다. 이 모델은 대규모 네트워크 환경 하에서는 수많은 사용자와 정보가 존재하기 때문에 접근제어 행렬의 크기가 매우 커지는 단점이 있다. 강제적 접근제어(MAC : Mandatory Access Control)은 정보 사용자와 보안대상 정보의 보안 등급에 따라 접근을 제어하는 방법으로 각각의 사용자와 보안대상 정보에게 보안 등급이 부여된다. 특히 일반적으로 객체라 불리는 보안대상 정보의 보안 등급을 비밀 등급(Classification Level) 주체라 불리는 정보 사용자의 보안 등급을 인가 등급(Clearance Level)이라 한다[5]. 이 모델은 보안 대상 접근에 대한 매우 엄격한 보안 기능을 제공함으로써 오류

정보의 흐름을 통제하는 장점을 제공한다. 그러나 이 모델은 정보의 흐름이 낮은 곳에서 높은 곳으로 흐르게 되므로 비밀성보다는 무결성이 더 중요시되는 상업적인 응용에는 적합하지 못하다.

2.2 RBAC 모델

RBAC의 주요한 목적은 보안 관리와 감사를 용이하게 하자는 것이다. 예를 들어 운영자 역할은 모든 자원들에 접근할 수 있지만 접근 권한을 바꾸지 못한다. 보안 관리자 역할은 권한을 변경할 수 있지만 자원에 접근할 수 없으며 감사자 역할은 감사 파일만 접근할 수 있다.



[그림 1] RBAC 모델

[정의 1] RBAC0 모델은 다음과 같은 요소들을 가진다.

- U(User), R(Role), P(Permission), S(Session)
- PA ⊆ P × R, a many-to-many permission to role assignment relation
- UA ⊆ P × R, a many-to-many permission to role assignment relation
- user : S → U, a function mapping each session s_i to the single user user(s_i) (constant for the session's lifetime),and
- role : S → 2^R a function mapping each session s_i to a set of roles
  role(s_i) ⊆ {r | (user(s_i),r) ∈ UA} (which can change with time) and session s_i has the permissions $U_{r \in roles(si)}$(p | (p,r) ∈ PA}

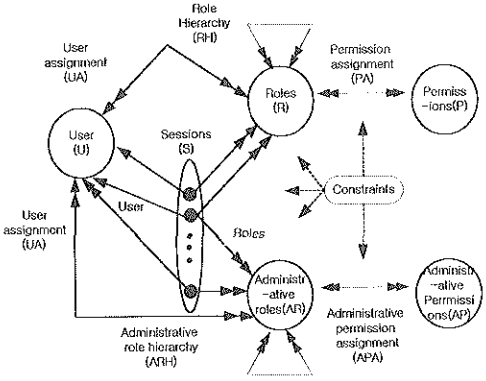
[정의 2] RBAC1 모델은 다음과 같은 요소를 가진다.

- U, R, P, S, PA(Pemission Assignment), UA
- RH ⊆ R × R, is a partial order on R called the role hierarchy or role dominance relation, also written as ≥, and
- role : S → 2^R is modified from RBAC0 to require role(s_i) ⊆ {r | ($\exists r' \geq r$)[(user(s_i),r') ∈ UA]} (which can change with time) and session s_i has the permissions $U_{r \in roles(si)}$(p | ($\exists r'' \geq r$)[p, r'') ∈ PA]}

[정의 3] RBAC2는 RBAC0의 여러 요소들의 값이 허락되는지를 결정하는 제약들의 집합을 요구하는 것을 제외하고는 RBAC0과 같다.

2.2 RBAC 관리 모델

RBAC을 관리하는 예는 Moffet과 Sloman[6]에 의해 보여진다. 이것은 역할 도메인, 소유자, 관리자, 보안관리자에 기반을 둔 정교한 모델을 정의한다. 권한을 하나의 중앙 지점으로부터 대표되거나 제어되지 않고 서로 제



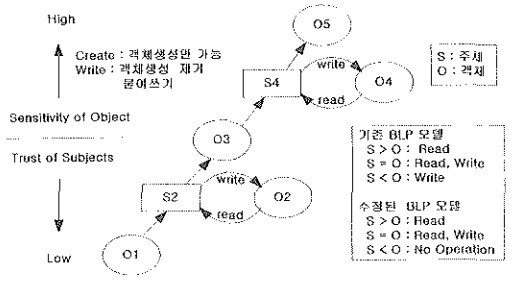
한된 신뢰를 가진 독자적인 관리자 사이에서 협정된다.

[그림 2] RBAC 관리 모델

2.3. 수정된 BLP 모델

BLP 모델은 오직 비밀성만을 취급할 뿐 허가 받지 않은 정보의 수정은 제어하지 않는다. 따라서 허가 받지 않은 수정을 막기 위한 무결성 대책을 고려한 수정된 BLP 모델은 다음과 같다.

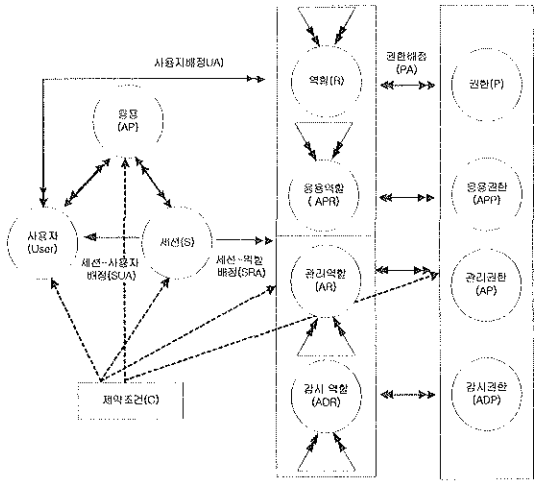
주체보다 높은 등급의 객체에 대한 쓰기(Write-up)를 제한한다. 즉 등급이 서로 같은 경우에만 쓰기가 가능하게 함으로써 BLP의 단점을 보안한다.



[그림 3] 수정된 BLP 모델

3. 제안 모델

3.1 확장된 RBAC 모델



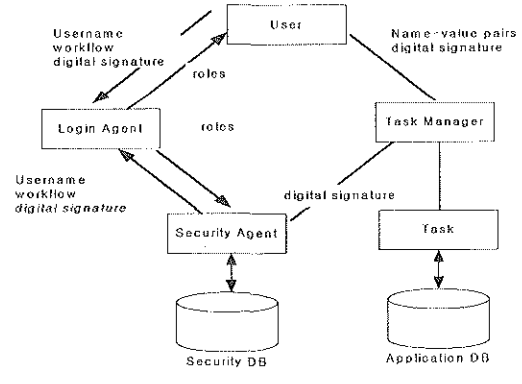
[그림 4] 확장된 RBAC 모델의 구성

3.2 확장된 RBAC와 MLS 결합 원리

워크프로우에서 보안이 더욱더 중요하게 제시되는 영역에서는 상속과 제약을 가진 기본 RBAC를 확장한 모델에 수정된 BLP를 결합시키는 것이 요구된다. 객체를 접근할 때에는 보안영역/레벨의 기준으로 제어되며, RBAC은 접근제어를 위해 사용되고 수정된 MLS는 데이터가 보안영역을 통과할 때 사용된다.

3.3 보안 모델의 구조

최상의 보안을 위해서 조직은 하나의 워크플로우를 여러개의 보안 영역으로 나누고 엄격한 통신제어는 보안영역을 통과할 때 이루어져야 한다. upgrade된 데이터는 필터링의 절차를 가져야 한다. 우선 사용자는 로그인하고 적절한 역할을 선택하는데 이때 로그인 에이전트는 보안 에이전트에 사용자가 역할을 선택하는데 있어 올바르게 할 수 있도록 해주고 사용자는 역할이 허락된 태스크를 실행할 수 있다. 이때 태스크 매니저는 태스크를 실행하기 위해 인가가 되었는지 결정하기 위해 사용자의 디지털 서명과 역할 정보들을 확인 조사해야만 한다.



[그림 5] 보안 모델의 구조

3.4 모델의 기능 평가

본 논문은 기업활동 전반에 걸친 통합된 전산처리 측면과 소비자가 관여된 전자거래의 측면 모두를 고려한 환경으로써 EC 환경에서 요구사항들을 기준으로 모델의 기능을 살펴보면 다음과 같다.

요구 사항	역할 계층	직무 분리	다중 응용	작업 흐름	동적 역할	정보 보호
RBAC 기본모델	○	○	□	□	△	△
RBAC-MLS 모델	○	○	○	○	△	○

○ Tightly △ Loosely □ Not supported

[표1] 모델의 기능 평가

4. 결론 및 향후과제

확장된 RBAC과 MLS 결합 모델은 작업 흐름의 자동화와 함께 일관된 데이터 접근 및 제어를 지원하기 위하여 제안되었다. 하지만 확장된 RBAC과 MLS 결합 모델은 동적인 역할 정의 및 해상을 거의 지원하지 않는다. 즉, 사용자가 자율적으로 다른 문맥 하에서 역할을 변경하는 것을 지원하지 않는다. 이러한 정적인 방법은 역할을 할당하는데 있어 관리상의 부하 문제를 발생시킬 수 있다. 따라서 향후 연구 과제로 이러한 문제들을 해결하기 위한 추가 보완 연구가 이루어져야 할 것이다.

5. 참고문헌

[1] Ravi S. Sandhu, "Role-Based Access Control", Laboratory for Information Security Techology, George Mason Univerity, Sep. 17, 1997.

[2] Ravi S. Sandhu, Edward J. Coyne, HaiL. Feinstein and Charles E. Youman, "Role Based Access Control Models", IEEE Computer, Volume 29, Number 2, Feb. 1996.

[3] David F. Ferraiolo, Janet A. Cugini, and D. Richard Kuhn, "Role-Based Access Control (RBAC) : Features and Motivations", 11th Annnal Computer Security Applications Conference, Dec. 1995.

[4] David F. Ferraiolo, Dennis M. Gilbert, and Nickilyn Lynch, "An Examination of Federal and Commercial Access Control Policy Needs", 16th National Computer Security Conference, Sep. 1993.

[5] Matunda Nyanchama, Sylvia Osborn, "Modeling Mandatory Access Control in Role-Based Security System", Proceedings of the Ninth Annual IFIP TC11 Working Conference on Database Security, pp. 129 - 144, August 1995.

[6] Jonathan D. Moffett and Morris S. Sloman. "Managing Role/Pemission RElationships Using Object Access Types", http://hissa.ncsl.ncist.gov/ rbac.