역할기반 접근통제에서 역할 계층에 따른 접근권한 상속의 표현

이 상 하[†] · 조 인 준[††] · 천 은 홍[†††] · 김 동 규[††††]

요 약

*RBAC(Role Based Access Control)*은 기업이나 정부의 조직구조에서 역할을 기반으로 자원을 보호할 수 있는 기본적인 접근통제모델을 제시하기 때문에 실세계를 반영하는 장점을 가지고 있다. 하지만, *RBAC*은 권한이 역할계층에 따라 상속되기 때문에 상위역할에 속한 사용자에게 권한이 집중되는 현상이 발생하여 최소 권한원칙의 위배로 인한 권한남용의 위험성을 내포하고 있다 반대로 최소 권한원칙을 준수하기 위해 과도한 제약을 가한다면 역할계층의 의미가 없을 수 있다. 거대한 조직에서 상속원칙에 복잡한 제약을 가하는 것은 효율적인 관리 어려움을 더하게 된다. 본 논문에는 *RBAC*의 접근권한 상속에 역방향의 태그 포인터 경로를 새롭게 표현한 상속원칙을 정의함으로써 역할계층의 체계 보존, 비효율적 역할관리 및 권한남용 문제점의 해결책을 제시하였다.

Permission Inheritance Expression with Role Hierarchy of RBAC

Sang-Ha Lee[†] · In-June Jo[††] · Eun-Hong Cheon[†††] · Dong-Kyoo Kim[††††]

ABSTRACT

RBAC(Role Based Access Control) has the advantage that reflects the real world because it presents a basic access control model based on user's role in organizations or governments. But in *RBAC* model, the privileges of the senior roles in these hierarchies are inherited from those of the junior roles, so *RBAC* model has the privileges problem that the senior are given more privileges than they need That is, it tends to infringe the Principle of Least Privilege On the other hand, if we give some excessive constraints on the *RBAC* model without scrupulous care, it may be meaningless property of role hierarchies. Furthermore, such complicated constraints make it more difficult to manage resources and roles in huge enterprise environments.

The purpose of this paper is to solve the problems of roles hierarchies such as inefficient role managements and abuse of privileges by using newly presented the backward tag pointer path expression in the inheritance of privileges

1. 서 론

최근 들어 기업 및 민간정부 조직의 정보 보안체계 구축에 대한 요구가 급증하여 이에 대한 연구가 활발하게 이루어지고 있다. 민간정부나 기업들은 그들의 대부분의 업무를 정보처리 시스템에 의존함에 따라, 비 인가된 주체나, 객체에게 정보자원이 몰법 노출되면 조직운영의 붕괴를 초래할 수 있다. 따라서, 이들 조직의 주관심사 중의 하나는 정보자원에 일관된 규칙의 접근통제 정책을 시행하여 그들이 소유하고 있는 정보자원에 무결성 보호 서비스를 실현하고자 한다[3]. 조직에서 사용자의 의미는 일정한 역할을 부여받고 그

† 정 회 원 : 동서울대 전자통신과 교수
†† 정 회 원 : 배제대학교 컴퓨터공학과 교수
††† 정 회 원 : 우석대학교 정보통신컴퓨터공학부 교수
†††† 정 회 원 : 이주대학교 정보 및 컴퓨터공학부 교수
논문접수 : 2000년 1월 6일, 심사완료 : 2000년 6월 20일

역할에 따라 일관된 접근통제하에서 정보자원을 처리하는 주체이다. 이들은 조직 내에서 역할을 부여받고, 이를 기반으로 조직내의 정보자원의 활용을 통제 받게 된다.

RBAC[1, 6, 7]모델은 상기와 같은 역할을 기반으로 접근통제 서비스를 제공하고자 제시된 모델이다. 이 모델이 기반이 되어 역할과 접근권한을 구성하는 방법에 관한 연구가 활발하게 이루어지고 있다. 여기에서 제시된 문제점은 정의된 특정역할에 접근권한을 부여함에 있어서 정의된 역할이 조직의 실제특성을 완전히 수용하지 못한다는 점이다. 역할계층에서 상위의 역할은 하위의 역할이 갖는 모든 접근권한을 상속받으므로 상위역할에 속한 사용자는 많은 권한을 소유하게 되어 최소권한의 원칙을 위배한다. 따라서 *RBAC* 구성요소에 적절한 제약을 정의하여 이를 해결하고 있다[7]. 하지만, 단순 제약정의는 상위역할의 권한남용 우려가 있고, 너무 과도한 제약은 상위역할로서 권한을 행할 수 없게되어 역할계층의 의미를 전달할 수 있는 문제점을 내포하고 있다. 따라서 접근권한을 역할의 배정시 접근권한의 세밀한 분석을 요한다

역할간의 계층관계는 접근권한의 상속이 이루어질 수 있도록 유지되어야 하고, 역할의 특성에 따라 상호배타적 역할을 설정하거나, 역할간의 관계에 따라 임무분리를 요하는 보안정책이 규정되어야 한다[2]. 즉, 역할에는 한 주체에 의해 동시에 수행되어서는 안 되는 상호 배타적 역할이 있는데[9], 이러한 역할은 사용자가 자신의 접근권한을 위반하지 않으면서 수행되도록 임무분리가 이루어져야 한다.

이럼에도 불구하고, *RBAC* 모델에서 제시한 역할의 접근권한 상속방법 그리고 역할의 배정 및 관리에는 다음과 같은 어려움이 있다. 첫째, 조직의 기능과 특성에 따라 구성되는 역할 계층구조에서 역할의 영역을 구분하는 방법과 역할간의 관계에 따른 접근권한의 표현이 모호한 점이다. 둘째, 역할에 부여되는 접근권한의 분류 및 권한 상속의 표현이 모호하다는 점이다.

이러한 문제점의 부분적인 해결책으로 참고문헌[5, 7]에서는 부분순서 관계(*Partial Ordering relation*)를 정의하여 역할의 상속속성에 따라 역할의 계층을 분류하는 방안을 제시하였다. 이의 특징은 역할에 접근권한 배정 및 수행의 용이성을 목적으로 이루어진 것이다. 이 연구에서는 객체의 기본 접근권한 관계를 방향성 그래프의 특성을 이용하여 모델링하였다[5].

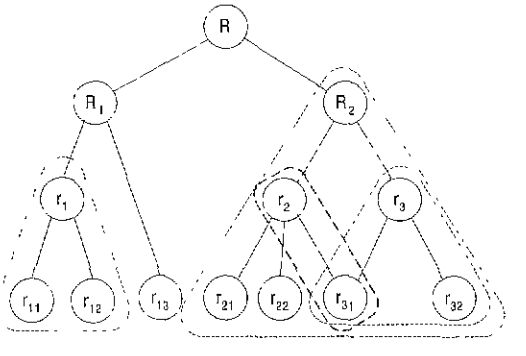
하지만, 이 모델링에 따라 역할기반 접근통제를 실세계에 적용할 경우 사용자가 접근권한내에서 부정조작이 이루어진다는 점과 역할 계층상에서 상위역할이 하위역할의 접근권한을 상속받기 때문에 상위역할은 최소권한의 원칙을 위배하여 하위계층의 역할에게 권한을 남용할 수 있는 위험성을 내포하고 있다.

본 논문에서는 참고문헌[5, 7]에서 제시한 *RBAC*의 역할계층에 순방향과 역방향의 경로를 새롭게 표현하여 상속원칙을 정의함으로써 상기와 같은 문제점의 해결책을 제시하였다 제시된 방안은 *RBAC*에서 제시된 기본적인 역할계층의 유지 및 자연스런 접근권한의 상속이 이루어짐에도 불구하고 권한남용의 위험성을 제거한 새로운 역할계층의 표현이다.

본 논문의 구성은 제안된 방안을 기술하기 위해 제2장에 기 연구된 내용을 요약하였고, 3장에서는 제안된 방안을 기술하고, 분석한 내용을 다루었다. 마지막으로 4장에 결론을 넣었다.

2. 관련 연구

본 장에서는 본 논문에서 제안된 모델과 관련된 연구로 참고문헌[5, 7]에서 제시된 역할상속 및 임무분리 모델을 정의하고, 분석한 내용을 다룬다. 그리고 분석된 내용을 근거로 문제점을 재시한다. 이의 설명을 위해 역할계층에서 역할간의 관계가 (그림 1)과 같은 그래프로 표시된다고 가정한다. 그림에서와 같이 어떤 조직에서 업무의 특성에 따라 각 부서는 역할영역 (r_1, r_{11}, r_{12}), (r_2, r_{21}, r_{22}) 그리고 (r_3, r_{31}, r_{32}) 등으로 구성되는 부 그래프로 표현할 수 있다. 그리고 부서간의 역할의 관련성에 따라 일부역할은 다른 부서



(그림 1) 역할의 계층

의 역할에 연결(r_2, r_{31})되어 정보의 흐름이 발생될 수 있다.

2.1 역할계층의 상속 정의

참고문헌 [5, 7]에 따르면, 역할은 조직의 특성과 배정된 역할을 수행하는 사용자의 책임과 자격에 따라 접근권한이 상·하위 역할간에 상속될 수 있다. 상·하위역할에 있어서 두 역할간의 정보의 흐름을 (→)으로 표현할 때, 임의의 역할집합이 ($r_i \rightarrow r_i$)인 반사, ($r_i \rightarrow r_j$)이고 ($r_j \rightarrow r_i$)이면 $r_i = r_j$인 반대칭, 그리고 ($r_j \rightarrow r_h$)이고 ($r_h \rightarrow r_i$)이면 $r_j \rightarrow r_i$인 천이관계를 각각 나타낸다. 이러한 관계를 모두 만족하는 상·하위역할 관계를 부분순서 관계(≥)역할이라고 하고, $r_i \geq r_j$로 표현한다. 이러한 부분순서 관계의 역할들을 부분순서 집합이라 한다. 역할계층에서 접근권한은 상·하위역할간의 부분순서 관계에 따라 단일상속이 이루어질 수도 있고, 부분순서 관계를 만족하는 다중상속이 이루어질 수 있다[8]. 이들은 다음과 같이 정의된다.

【정의 1】 사용자 $U=\{u_1, \cdots, u_k\}$, 역할 $R=\{r_1, \cdots, r_l\}$ 접근권한 $P=\{pv_1, \cdots, pv_m\}$ 세션 $S=\{s_1, \cdots, s_n\}$, $k, l, m, n \geq 0$라고 하면,

- $U(r_i)$: 역할 (r_i)에 배정된 사용자의 집합,
- $P(r_j)$: 역할 (r_j)에 배정된 접근권한 집합,
- $U(s_i)$: 활성화된 세션 (s_i)을 실제적으로 실행할 수 있는 사용자의 집합,
- $R(s_i)$: 역할 (r_i)에 의해 활성화된 세션 (s_i)가 실제적으로 실행할 수 있는 역할의 집합을 의미한다.

【정의 2】 임의의 역할 (r_j)가 다른 역할 (r_i)로 상속 ($r_j \leq r_i$)될 때, 역할 (r_i)의 접근권한은 $\bigcup_{\forall r_j \leq r_i}\{P(r_j)\}$이다.

【정의 3】 임의의 역할 (r_j)가 다른 역할 (r_i)로 상속 ($r_j \leq r_i$)될 때, 역할 (r_i)에 의해 활성화된 세션 (s_i)은 다음과 같은 의미를 갖는다.

- 사용자 (u_i)에 의해 활성화된 세션 (s_i)의 역할은

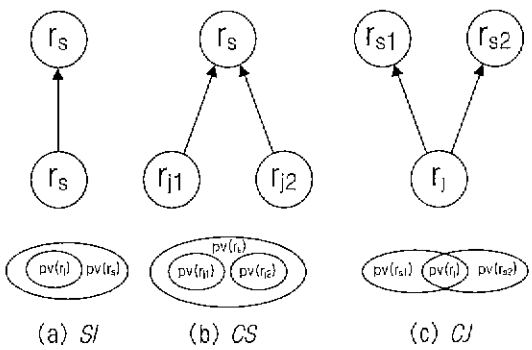
$$R(s_i) \subseteq \{r_i \mid (\forall r_i \geq r_j)[U(s_i) \in U(r_i)]\}$$

where $R : S \rightarrow 2^R$

- 사용자 (u_i)에 의해 활성화된 세션 (s_i)의 접근권한은

$$\bigcup_{r_i \in R(s_i)}\{pv \mid (\forall r_i \geq r_j)[P(r_j)]\}$$

역할의 계층에서 부분순서 관계에 있는 역할은 접근권한의 상속속성에 따라 (그림 2)와 같이 단순 접근권한 상속(*SI : Simple Inheritance*), 공통 상위 접근권한 상속(*CS : Common Senior inheritance*)와 공통 하위 접근권한 상속(*CJ : Common Junior inheritance*)의 세 가지 유형으로 구분하고[5], 다음과 같이 정의한다.



(그림 2) 역할의 구성

【정의 4】 하위역할 (r_j)의 접근권한이 부분순서 관계에 있는 하나의 상위역할 (r_s)로 상속되면 단순 접근권한 상속을 $(r_j, r_s) \in SI$라 하고, 접근권한은 $P(r_j) \subseteq P(r_s)$이다.

【정의 5】 두 개 이상의 하위역할 (r_{j1}, r_{j2})가 부분순서 관계에 있는 하나의 상위역할 (r_s)로 접근권한이 상속되면 공통 상위 접근권한 상속을 $((r_{j1}, r_{j2}), r_s) \in CS$라 하고, 접근권한은 $P(r_{j1}) \cup P(r_{j2}) \subseteq P(r_s)$이다.

【정의 6】 하위역할 (r_j)의 접근권한이 부분순서 관계에 있는 두 개이상의 상위역할 (r_{s_1}, r_{s_2})로 상속되면 공통 하위 접근권한 상속을 $((r_j, (r_{s_1}, r_{s_2})) \in CJ$라 하고, 접근권한은 $P(r_j) \subseteq P(r_{s_1}) \cap P(r_{s_2})$이다

2.2 임무분리의 정의

임무분리는 주어진 특정역할을 여러 계층의 사람이

수행할 경우에 발생한다[5, 9]. 따라서 이러한 역할은 다수의 부분역할로 분리되어야 하고, 이렇게 분리된 부분역할이 접근권한이 허용된 사람에게만 배정함으로써 정보자원의 무결성을 보장할 수 있다. 즉, 분리된 역할이 그 역할을 배정 받은 사람에 의해서만 수행되는 것을 보장해야 한다. 따라서 각 개인은 자신에게 허용된 부분역할만을 수행할 수 있다.

특히, 동일 사용자가 동시에 수행하지 말아야 하는 상호 배타적 접근권한을 갖는 상호 배타적 역할(R_m : *Mutual Exclusive Role*)에 대해서는 사용자가 이를 준수하면서 역할의 접근권한을 수행하도록 하여야 한다[8]. 임무분리는 사용자에게 역할을 배정하거나 사용자가 역할을 수행하는데 있어서 정적 임무분리와 동적 임무분리로 구분할 수 있고, 기본 접근권한을 사용하여 역할을 배정하면 임무분리의 표현 및 관리를 용이하게 할 수 있다.

따라서, 상호 배타적 역할은 상호 배타적 접근권한을 갖는 역할로써, 배정된 접근권한을 사용자가 동시에 수행하는 것을 방지해야 한다. 이에는 하나의 역할에 두 개이상의 상호 배타적 접근권한이 주어지는 경우와 두 개이상의 역할에 상호 배타적 접근권한이 주어져서 수행되어야 하는 경우가 있다.

이 연구에서는 상호 배타적 역할에 대해서 접근권한을 상속할 수 없도록 제약을 두고 있다. 따라서, 이러한 역할은 자신에게 배정된 고유 접근권한 형태로 표현된다. 이들에 대한 설명은 다음과 같이 정의할 수 있다.

【정의 7】 임의의 상호 배타적인 접근권한 (pv_i, pv_j)에 대하여 $\neg(pv_i \in P(r_i) \wedge pv_j \in P(r_j))$일 때 역할 (r_i, r_j)을 상호 배타적 역할 $(r_i, r_j) \in R_m$이라 한다.

【정리 8】 공통 상위 접근권한 상속 $((r_i, r_j), r_s) \in$ CS 관계인 상호 배타적 역할 $(r_i, r_j) \in R_m$의 접근권한은 상속될 수 없다.

【증명】 상호 배타적 역할 (r_i, r_j)의 접근권한이 r_s에 상속될 수 있다고 가정하자. 부분순서 관계에 있는 역할이 $(r_i \preccurlyeq r_s)$와 같이 상속되면 접근권한은 $P(r_i) \subseteq P(r_s)$이다. 따라서 상호 배타적 접근권한 (pv_i)이 $pv_i \in P(r_i)$임에 따라 $pv_i \in P(r_s)$관계가 성립된다. 또한 $(r_j \preccurlyeq r_s)$로 상속될 경우에는 접근권한은 $P(r_j) \subseteq P(r_s)$이다 따라서 상호 배타적 접근권한 (pv_i)가 $pv_j \in P(r_j)$임에 따라 $pv_j \in P(r_s)$관계가 성립된다. 그러므로 (pv_i, pv_j) $\in P(r_s)$가 성립된다. 이는 【정의 7】에 의해서 $pv_i \in P(r_i) \wedge pv_j \in P(r_j)$이 되기 때문에 모순이다. 따라서 *CS* 관계인 상호 배타적 역할의 접근권한은 상속될 수 없다. □

2.3 역할상속 및 임무분리 분석 및 문제점

앞에서 정의한 바와 같이 Sandhu, Matunda Nyanchama [5, 7] 등이 제시한 역할기반 접근통제에서 상속에 의한 최소권한 원칙의 위배 등과 같은 역할상속 및 임무분리의 문제점을 분석하면 다음과 같다.

【분석 1】

[정의 2]에서 역할 (r_j)가 상속 $(r_j \preccurlyeq r_i)$될 때 사용자 역할의 접근권한은 사용자의 역할과 상속이 이루어진 하위역할 각각에 허가된 접근권한의 합집합이다. 따라서, 하위역할에서 상위역할로 접근권한의 상속은 상위역할이 많은 권한을 가진다. 이는 권한남용이 발생할 수 있다.

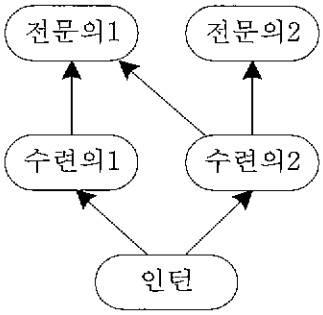
【분석 2】

[정의 3]에서 역할 (r_j)가 상속 $(r_j \preccurlyeq r_i)$될 때, 사용자는 세션을 통해서 대응되는 모든 역할을 활성화할 수 있다. 즉, 사용자가 역할 (r_i)을 통해 세션 (s_i)을 활성화할 경우, 세션 s_i에 의해 활성화될 수 있는 역할은 역할 (r_i)와 상속받은 하위역할 (r_j)들의 집합들로 구성한다. 사용자의 세션 (s_i)이 갖는 접근권한은 역할 (r_i)에 배정한 접근권한과 상속받은 하위역할 (r_j)의 모든 접근권한의 합집합이다. 따라서, 활성화된 세션 (s_i)에서 상위역할이 많은 권한을 가진다.

【분석 3】

[정의 4]에서 단순 접근권한 상속은 (그림 2)의 (a)와 같이 하위역할 (r_j)의 접근권한이 부분순서 관계에 있는 상위역할 (r_s)로 상속되어 하위역할의 접근권한을 수행할 수 있다 이는 상위역할의 접근권한이 하위역할의 접근권한을 수행할 수 있음을 의미한다. 즉, 접근권한이 $P(r_j) = \{pv_1\}$, $P(r_s) = \{pv_2\}$로 주어질 때 $P(r_s) = \{pv_1, pv_2\}$로 된다. 따라서, 상위역할이 많은 접근권한을 가진다.

[예제] 병원 업무에서 (그림 3)과 같이 인턴의 환자에 대한 진료기록의 접근권한이 수련의1과 수련의2에게 상속될 수 있다.



(그림 3) 역할의 상속

【분석 4】

[정의 5]에서 상위역할로 공통 접근권한 상속은 (그림 2)의 (b)와 같이 두 개이상의 하위역할 (r_{j_1}, r_{j_2})의 접근권한이 부분순서 관계에 있는 상위역할 (r_s)로 상속됨을 뜻한다 따라서, 하위역할의 모든 접근권한을 상위역할에서 수행할 수 있다. 다수개의 하위역할이 있을 때 상위역할의 접근권한은 모든 하위역할이 갖는 접근권한의 합집합이다. 이를 일반적인 표기법에 따라 정의하면 $P(r_s) = \bigcup_{r_j \in CS} P(r_j)$이 된다. 예를 들어 하위역할에 배정된 접근권한이 $P(r_{j_1}) = \{pv_1\}$, $P(r_{j_2}) = \{pv_2\}$이고 상위역할에 배정된 접근권한이 $P(r_s) = \{pv_3\}$로 주어질 때, 상위역할이 실행할 수 있는 접근권한은 $P(r_s) = \{pv_1, pv_2, pv_3\}$이다. 따라서, 모든 권한이 상속되어 상위역할은 최고의 권한을 가진다.

[예제] 병원 업무에서 (그림 3)과 같이 환자의 상태에 따라 수련의1과 수련의2의 환자 진료 기록의 접근권한이 전문의1에게 상속될 수 있다.

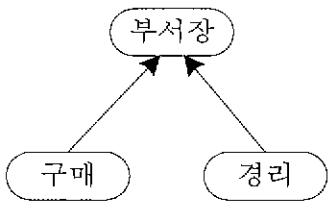
【분석 5】

[정의 6]에서 공통 하위 접근권한 상속은 (그림 2)의 (c)와 같이 하위역할 (r_j)의 접근권한이 두 개이상의 상위역할 (r_{s_1}, r_{s_2})로 상속되어 하위역할의 접근권한을 부분적으로 상위역할에서 수행할 수 있다. 이를 일반적인 표기법에 따라 정의하면 상위역할의 접근권한은 $P(r_s) = \bigcap_{r_j \in CJ} P(r_j)$이 된다. 예를 들어, 상위역할에 배정된 접근권한이 $P(r_{s_1}) = \{pv_1\}$, $P(r_{s_2}) = \{pv_2\}$이고, 하위역할에 배정된 접근권한이 $P(r_j) = \{pv_3, pv_1\}$로 주어질 때. pv_3가 상위역할로 상속될 경우 상위역할의 접근권한은 각각 $P(r_{s_1}) = \{pv_1, pv_3\}$, $P(r_{s_2}) = \{pv_2, pv_3\}$가 된다 따라서, 하위역할의 교집합으로 일부 상속을 받지만 상위역할은 더 큰 권한을 가진다.

[예제] 병원 업무에서 (그림 3)과 같이 환자의 상태에 따라 수련의2의 환자 진료 기록의 접근권한은 전문의1과 전문의2에게 다중 상속될 수 있다.

【분석 6】

[정의 7]에서 예를 들어, 어떤 조직에서 (그림 4)와 같이 구매 부서와 경리부서가 있을 때, 구매 부서의 거래 명세서 발급업무와 경리부서의 비용 지불업무를 상호 배타적 역할이라 한다. 이 역할은 분리되어 수행되어야 하므로, 각 사원이 양쪽의 권한을 동시에 수행할 수 없도록 역할을 분리하여 배정하여야 한다. 그렇지 않으면, 두 권한을 남용하여 실제구매는 이루어지지 않고 비용만 지불될 수 있다. 상호 베타적인 역할을 분리 수행하더라도 상속에 의해 상위역할이 많은 권한을 가질 수 있으므로 상위 사용자가 권한을 남용할 수 있다.



(그림 4) 임무분리의 예

【분석 7】

[정리 8]에서 공통 상위 접근권한 상속에서 상호 배타적인 역할에 대해서 상속하지 못하게 제약을 두는 것은 정당하다. 그러나 역할계층의 표현에 있어서 문제를 야기 시킨다. 즉, 어떤 조직의 역할에 대해서 임무분리를 유지하고자 모든 경우의 패턴을 각각 관리한다는 것이 매우 비효율적이고 관리자의 실수로 상호 배타적인 역할을 찾아내지 못할 경우도 발생한다. 또한 상속을 하지 못하게 하면 조직 역할계층을 유지하는 방법의 모색이 요구된다.

결론적으로 참고문헌 [5, 7]의 [정의 2]~[정의 7]는

상속의 원리를 준수하나 상위역할 권한이 집중되는 문제점을 가지고 있다. 또한, [정리 8]은 상속을 제한시키는 정의이기 때문에 기업조직에 체계적 계층적 역할특성을 반영하는데 문제가 있다. 또한 참고문헌 [5]는 $RBAC_2$[7, 8, 10]모델에 제약조건을 부여한 결과이기 때문에 모든 공유자원 객체에 따라 접근권한을 배정하는데 세심한 고려 없이는 오류를 범하기 쉽고, 관리의 복잡성 때문에 효율을 저하시킨다. 분석된 내용으로부터 문제점을 다음과 같이 3가지로 정리할 수 있다.

(1) 하위역할에서 상위역할로 접근권한의 상속으로 인한 권한남용의 위험성이 있다.
(2) 상호 배타적인 역할의 임무분리를 위해 접근권한 상속을 못하게 함으로 실세계의 조직의 역할 체계를 반영하지 못한다. 즉, 조직 역할계층의 특성을 상실한다.
(3) 역할관리를 효율적으로 할 수 없다.

3. 제안된 방안

본 장에서는 2.3절에 제시된 문제점 해결을 위해 상속의 원리를 유지하면서, 권한남용을 방지할 수 있는 방안을 제시한다. 역할계층은 관련성 있는 역할간의 부분순서 관계로서 정의된 체계를 말한다. 이는 기업조직의 권한과 책임을 정의하는 조직체계와 유사하며, 기업의 권한체제를 모델화 하는데 적합하다. 기업환경에서 행해지는 역할의 흐름은 기능에 따라 다음과 같이 대별할 수 있다.

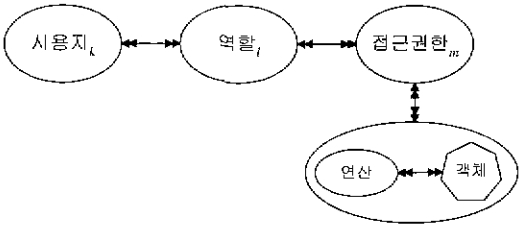
3.1 제안방안의 정의

(1) 관리를 위한 감독기능 - 수직적 관계 - 간접 관계 - 역방향 경로
(2) 자신의 고유 업무기능 - 수평적 관계 - 직접 관계 - 순방향 경로

상기에서 (1)이 의미하는 것은 기업의 조직체계에서 역할이 상위역할과 하위역할로 구성되어 수직적 계층관계가 형성됨을 의미하고 업무의 흐름측면에서 보면 주로 상위역할이 하위역할에 주어진 업무가 감독 및 관리업무임을 뜻한다. 또한 상위역할의 업무처리는 하위역할을 간접적으로 활용하는 간접적인 업무처리 행태를 보인다. 따라서, 상위역할이 하위역할에 대해 업무 감독을 위한 접근권한의 부여 방법은 첫째, 하위역할에서 특정형태로 감독기능을 정의하고, 둘째, 상위역할에서 하위역할을 지시하는 포인터를 생성함으로써 가능하다 이와 같이 접근권한이 부여된 상태에서 상위역할에서 하위역할에 대해 감독 기능을 수행하고자 할 경우에는 상위역할에 정의된 하위역할 포인터를 참조하여 하위역할에 도달하고 도달된 하위역할에서 특정하게 정의된 감독기능만을 수행하도록 제약한다. 따라서 상위역할이 하위역할에 대해 행해지는 감독기능은 상위역할에 정의된 하위계층의 포인터를 따라감으로써 결정되기 때문에 이를 역방향 경로로 정의한다.

그리고 (2)가 의미하는 것은 (1)과 같이 역할 계층간에 이루어지는 업무처리 형태가 아니라 자신의 역할에만 배정된 고유업무 처리 형태를 말한다. 따라서 사용자는 자신의 고유업무를 직접처리하기 때문에 직접관계에 있고 이들 업무는 상하역할 계층과 관계가 없기 때문에 수평관계에 있다. 그리고 자신의 역할 내에 정의된 고유업무만을 처리하기 때문에 이를 순방향 경로로 정의한다.

기존의 방안에서는 상기의 (1)의 경우에 역할계층이 형성된다. 형성된 역할계층에서는 상위역할이 하위역할의 모든 접근권한을 포함하고 있음을 의미한다. 역할간의 접근권한은 상속속성을 갖는다. 이러한 속성을 지닌 접근권한은 하나 이상의 객체에 대해서 특정한 연산을 수행할 수 있도록 승인된 것을 의미한다. 사용자, 역할, 그리고 접근권한 사이의 배정관계는 다대다 (↔) 의미를 내포하여 (그림 5)로 나타낼 수 있다.



(그림 5) 역할 계층의 권한 배정

(그림 5)를 기반으로 본 논문에서 제안된 방안을 [정의 9]~[규칙 1]과 같이 기술한다.

【정의 9】 접근권한 (pv_m)은 연산(op)과 객체 (obj)의 쌍으로 구성된다.

【정의 10】 접근권한의 요소는 2 ″′ ″이다. 여기서,

m은 op의 개수, n은 obj의 개수이다.

여기에서 접근권한 요소를 $pv_m=(op, obj)$로 표현한다. 여기에서 객체(obj)는 시스템의 내적 자원(예, 파일, 디렉토리 등)과 시스템의 외적 자원(예, 프린터, 디스크, CPU 등)으로 구성된다. 연산(op)은 객체(obj)에 행해지는 연산으로 운영체제 측면에서 읽기, 쓰기, 실행으로 표현할 수 있고, 데이터베이스 측면에서 실행, 삽입, 삭제, 선택, 갱신 등을 의미한다.

다음으로 임무분리를 요하는 상호 배타적인 역할, 그리고 *SI, CJ, CS* 형태로 구성되는 역할에 대해서 접근권한 상속제약을 가한다. 이는 역방향(즉, 기업에서 관리감독 기능에 해당함)상속에 제약을 가하고자 한 것으로 다음과 같이 정의된다.

【정의 11】 역할 (r_i, r_j)가 $r_i \geq r_j$일 때, $C(P(r_j))$는 $P(r_j)$에 대한 제약 집합이다.

본 논문에서 제안한 내용을 구체적으로 설명하기 위해 $C(P(r_j))$를 제한적인 집근권한 $M_j(op_{1_j}, \cdots, op_{n_j})$, $n \geq 0$로 표기한다. 여기에서 $M_j(op_{1_j}, \cdots, op_{n_j})$는 역할 (r_i)에게 제한적으로 상속될 접근권한을 나타낸다.

【정의 12】 역할 (r_i, r_j)가 $r_i \geq r_j$인 경우, 다음과 같은 조건을 만족하도록 제약을 가한다.

(1) $P(r_j) \supseteq C(P(r_j))$. 역할 (r_j)에서 정의할 수 있는 제약으로 접근권한 $C(P(r_j))$는 역할 (r_j)가 허용된 접근권한 $P(r_j)$의 부분집합이다.

(2) $r_i \geq r_j$인 경우, 역할 (r_i)에 상속된 역할 (r_j)의 접근권한은 $P(r_i) \supseteq C(P(r_j))$이다. 즉, 역할 (r_i)에 상속된 접근권한은 (1)에 정의된 $C(P(r_j))$이다.

위 [정의 12]은 권한남용 방지를 위해 역할 (r_j)의 모든 접근권한들 중에서 상속이 가능한 접근권한 $C(P(r_j))$를 의미하고 이를 역할 (r_i)에 상속되도록 한 제약 규정이다. (1)에서 $P(r_j)$는 역할 (r_j)에 배정되어 허용된 접근권한의 집합을 의미하고, $C(P(r_j))$는 역할 (r_j)에 허용된 접근권한들 중에서 상속제약을 가한 접근권한의 집합을 의미한다.

【정의 13】 역할 (r_i, r_j)가 $r_i \geq r_j$인 경우, 역할 (r_i)에 제한적으로 상속된 권한을 태그 포인터$\ll r_j \gg$를 사용하여 나타내고, r_i : $\langle obj_{1_j}, \cdots, obj_{n_j} \rangle \ll r_j \gg$로 표기한다.

위 [정의 13]는 상위역할이 하위역할로부터 제한적으로 상속받은 접근권한이 있을 경우, 상위역할에 하위역할을 가리키는 태그 포인터를 생성시키는 규정이다.

역할 r_i와 r_j에 허용된 접근권한은 [정의 9]~[정의 13]에 의해서 다음과 같이 표현할 수 있다. 여기에서 $p_i(op_{1_i}, \cdots, op_{n_i})$ 및 $p_j(op_{1_j}, \cdots, op_{n_j})$는 역할 (r_i, r_j)에 각각 허용된 고유의 접근 권한을 의미한다.

$$r_i \quad \langle obj_{1_i}, \cdots, obj_{n_i} \rangle \ll r_j \gg, \langle obj_{1_i}, \cdots, obj_{n_i} \rangle p_i(op_{1_i}, \cdots, op_{n_i})$$

$$r_j \quad \langle obj_{1_j}, \cdots, obj_{n_j} \rangle p_j(op_{1_j}, \cdots, op_{n_j}) \; M_j(op_{1_j}, \cdots, op_{n_j})$$

【규칙 1】 역할 (r_i, r_j)가 $r_i \geq r_j$인 경우, 역할 (r_i)는 역할 (r_j)로부터 상속된 권한을 다음과 같은 절차에 따라 실행한다.

(1) 역할 (r_i)에 역할 (r_j)의 태그 포인터$\ll r_j \gg$가 존재하면 제한된 상속권한이 있다고 판단하고 (2)를 실행한다. 존재하지 않으면 실행을 정지한다.

(2) 사용자는 역할 (r_j)의 태그 포인터$\ll r_j \gg$를 참조하여 역할 (r_j)에 존재하는 $C(P(r_j))$, 즉, $M_j(op_{1_j}, \cdots, op_{n_j})$의 접근권한만을 실행한다.

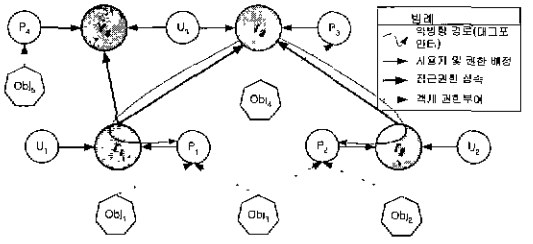
위 [규칙 1]은 본 논문에서 제안한 방안에서 상위역할이 하위역할로부터 상속받은 접근권한을 실행하는 방법을 규정한 것이다.

3.2 제안방안의 적용 예

상기와 같은 정의를 기반으로 제안된 방안을 (그림 6)과 같은 역할계층구조를 예로 들어 설명하면 다음과 같다. 그림에서와 같이 각 역할은 역할간의 경로(즉, 역방향 경로)를 유지한다. (그림 6)에서 제안된 역할계층은 <표 1>과 같은 구조로 각 역할에 상속된 접근권한을 유지한다

<표 1>에서 역할 r_1과 r_2는 고유역할(즉, 순방향역할)권한으로 p_1과 p_2를 가지고 연산(r, w)을 수행함을 나타낸다. 그리고 추가적으로 상위역할의 접근권한 남

용을 방지하기 위해 M_1과 M_2에 상위역할로부터 허용할 수 있는 접근권한(연산(r, w), (r))을 각각 정의하고 있다.



(그림 6) 역할계층 구조 관계

〈표 1〉 역할에 배정된 접근권한 정의

사용자	역할	접 근 권 한
u_1	r_1	$<obj_1, obj_5> p_1(r,w)\ M_1(r,w)\ M_1(r)$
u_2	r_2	$<obj_2, obj_3> p_2(r,w)\ M_2(r)$
u_3	r_3	$<obj_1, obj_3> \ll r_1 \gg$, $<obj_2, obj_3> \ll r_2 \gg$, $<obj_4> p_3(r,w)$
u_3	r_4	$<obj_1, obj_3> \ll r_1 \gg$, $<obj_5> p_4(r,w)$

[범례] $p_x(r,w)$: 역할 r_x의 고유 접근권한을 의미함
$M_x(r,w)$: 역할 r_x가 상위역할에게 허용하는 접근권한 (권한남용 방지용)
$\ll r_x \gg$: 상위역할에 상속된 하위역할을 가리키는 태그 포인터를 의미함

다음으로 상위역할 r_3는 역할 r_1과 r_2로부터 p_1, p_2의 연산을 상속받았다. 따라서 기존의 방안에서 역할 r_3는 r_1과 r_2의 접근권한을 그대로 수행하였다. 하지만, 제안된 방안에서는 <표 1>의 접근권한 항목에 태그 포인터 $\ll r_1 \gg$, $\ll r_2 \gg$을 표기하여 간접관계 상속을 정의하고, [규칙 1]에 따라 p_1, p_2 연산을 직접수행을 하지 않고 태그 포인터 $\ll r_1 \gg$, $\ll r_2 \gg$을 참조하여 역할 r_1과 r_2에 정의된 고유역할 권한(즉, $p_1(r, w)$, $p_2(r, w)$) 혹은 권한남용을 방지하기 위한 권한(즉, $M_1(r, w)$, $M_2(r)$)을 접근통제 정책에 따라 선택적으로 수행한다. 이러한 활성화된 제약을 둠으로써 상위역할 r_3이 수행하는 연산을 하위역할 r_1과 r_2가 확인하여 정의된 연산만을 허용하기 때문에 상위역할에 의한 권한남용을 방지할 수 있다. 이럼에도 불구하고 조직간 역할계층은 그대로 유지된다. 즉, r_3의 태그 포인터를 통해서 r_1과 r_2의 역할계층의 표현이 가능하다.

이와 같은 고안을 바탕으로 (그림 6)의 경로는 다음과 같은 의미를 갖는다. 역 방향경로는 수직계층간의 모든 권한상속을 태그 포인터를 사용하여 간접적으로 유지하고, 권한의 남용을 방지하기 위해 상위역할이 자신에게 상속된 권한을 수행할 경우, 상위역할에 표시된 태그 포인터를 이용하여 하위역할에서 상위역할에 허용된 접근권한만을 수행할 수 있도록 한 경로를 의미한다. 이러한 간편한 역 확인 절차를 부여함으로써 2.3절에서 문제로 제기된 권한 남용을 방지할 수 있다. 따라서, 역 확인 작업에 의해서 고유업무 기능을 모두 수행하는 것이 아니라 [정의 12]에 규정된 임의의 관리기능만 수행하게 할 수도 있다. 즉, 배정 받은 접근권한의 모든 연산을 수행하는 것이 아니라 최소 활성화된 세션으로 하나의 연산만을 수행하게 함으로써 최소권한의 원칙이 유지되면서 효과적인 관리 감독기능이 가능하다.

<표 1>에서 $\ll r_x \gg$는 해당 역할의 활성화 시점에서 하위역할로부터 접근권한을 상속받기 위한 태그 포인터이다. 또한 상호 배타적인 역할이 역할상속의 제약을 준수하면서 역할계층에 자연스럽게 반영된다. 따라서, 상속에 대한 제약을 주는 어려움을 쉽게 해결할 수가 있다.

3.3 제안방안의 특징

(1) 상위역할이 하위역할로부터 상속된 접근권한 수행을 [정의 12]에 정의된 제약에 따라 태그 포인터 $\ll r_x \gg$를 참조하여 하위역할의 M_x에 정의된 접근권한에 대해서만 수행하도록 제약하기 때문에 상위역할에 의해 이루어질 수 있는 권한남용 문제를 해결하였다.

(2) 기 제안된 방안에서는 상호 배타적인 역할에 대해 임무분리가 이루어질 경우, 접근권한 상속을 허용하지 않았다 따라서, 실세계 조직의 역할체계를 접근통제 시스템에 반영할 수 없었다. 본 논문에서 제안된 방안은 각 역할에서 태그 포인터($\ll r_x \gg$)를 이용하여 하위역할의 접근권한 상속을 간접적으로 표기하기 때문에 실세계 조직의 역할계층을 자연스럽게 표현이 가능하다.

(3) 고안된 방안을 역할관리 측면에서 살펴보면, 각 역할의 접근권한 항목에 상위역할이 수행할 수 있는 접근권한을 새롭게 정의하여 추가하고, 수정하고, 삭제하는 행위만을 요구하기 때문에 간편하다. 즉, 기존방안에는 하위역할에서 상위역할로 상속된 접근권한을 상위역할에 직접 표기하였다.

하지만, 제안된 방안에서는 태그 포인터(≪ r_x ≫)를 통해서 간접적으로 접근권한 상속을 표기하였다. 따라서, 기존방안에서 역할에 배정된 접근권한이 변동되면, 이 역할의 상위역할에 상속된 접근권한이 변경되어야 했다. 하지만, 제안된 방안에서는 상위역할의 접근권한에 어떤 변경행위도 필요치 않다. 따라서, 역할에 배정된 접근권한의 관리가 기존의 방안보다 제안된 방안이 우수하다. 다음으로 접근통제 실행의 효율성측면을 살펴보면, 상위역할이 하위역할로부터 상속받은 역할을 수행할 경우, 태그 포인터(≪ r_x ≫)를 참조하여 해당하는 하위역할을 방문하여 직접 권한남용 여부를 확인하여 수행되기 때문에 그 만큼의 연산처리 부담이 수반된다. 하지만, 이는 컴퓨터 내에서 행해지는 행위이기 때문에 성능에는 크게 영향을 미치지 않을 것으로 판단된다.

4. 결 론

역할기반 접근통제는 기업의 조직환경에 적응성이 탁월하기 때문에 기업의 정보자원을 효율적으로 관리하는데 적합하다. 이의 실현을 위해 Ravi S. Sandhu가 4가지 모델로 $RBAC_0$(기본 모델), $RBAC_1$(계층에 따른 상속), $RBAC_2$(접근제약), $RBAC_3$(이전 세 개의 모델의 통합관계를 유지)을 제안하였다. 하지만, 이들 모델에서는 상위역할이 하위역할에 대해 행할 수 있는 권한남용 문제의 해결책을 명시하지 않고 있을 뿐만 아니라, 상호 배타적인 역할에 대해 접근권한의 상속을 허용치 않아 실세계의 역할체계를 반영하여 구현하는데 문제가 있다.

본 논문에서는 상기의 $RBAC$ 모델의 속성을 최대한 유지하면서 구현 시 권한남용문제를 해결하여 최소권한의 원칙을 유지하였다. 이와 더불어 상호 배타적인 역할이 실세계의 역할계층에 자연스럽게 반영되는 방안을 고안하였다. 각 계층간에 M_x라는 제약을 적용할 수 있는 속성을 두어 제약의 상호 배타적인 환경에서 세션의 개념을 적용하였다

이는 각 역할에 허용되는 접근권한 정의 시 상위역할에 허용할 수 있는 접근권한을 추가로 정의하고, 상위역할이 태그 포인터(≪ r_x ≫)를 통해서 추가로 정의된 접근권한에 접근하여 이를 기반으로 상속된 접근권한이 수행할 수 있도록 함으로써 성취된다.

향후 연구분야는 새로운 역할을 객체지향 방법에서 사용하는 클래스로 표현하고, 역할간의 접근권한 상속원칙이 클래스간의 상속성을 기반으로 클래스의 속성을 이용한, 권한남용 방지와 더불어 실세계 조직의 역할체계를 그대로 반영할 수 있는 객체기반 접근통제시스템을 실현하는 분야이다.

참 고 문 헌

[1] David F. Ferraiolo, Janet A. Cugini and D. Richard Kuhu, "Role-Based Access Control(RBAC) : Features and Motivations," 11th Annual Computer Security Application Conference. pp.554-563, Dec. 1995.

[2] D. Richard Kuhn, "Mutual Exclusion of Role as Means of Implementing Separation of Duty in Role-Based Access Control Systems," National Institute of Standards and Technology, Jun. 1996.

[3] Emil Constantin Lupu, "A Role-Based Framework for Distributed Systems Management." University of London, Phd thesis. Jul. 1998

[4] Fang Chen & Ravi Sandhu, "Constraints for Role Based Access Control," In Proceedings 1st ACM Wokrshop RBAC, Sep. 1995.

[5] Matunda Nyanchama, "Commercial Integrity, Roles and Object Orientation," University of Western Ontario, Phd thesis, Sep. 1994..

[6] Ravi Sandhu and Venkata Bhamidipati, "The ARBAC 97 Model for Role-Based Administration of Roles : Preliminary Description and Outline," Proceedings of Second ACM Workshop on Role-Based Access Control, Nov. 6-7, 1997.

[7] Ravi S. Sandhu, Edward J Coyne, Hal L. Feinstein, and Charles E Youman, "Role-Based Access Control Models," IEEE Computer, Vol.29, No.2, pp 38-47, Feb. 1996.

[8] Ravi S. Sandhu, "Role Hierarchies and Constraints for Lattice-Based Access Control," Proc. Fourth European Symposium on Research in Computer Security, Sep. 1996.

[9] Ravi Sandhu, "Separation of duties in Computerized Information Systems," Proc of the IFIP WG11.3 Workshop on Database Security, Sept. 1990.

[10] Serban I. Gavrila and John F. Barkley, "Formal Specification for Role Based Access Control User/Role and Role/Role Relationship Management," NIST. Sep. 1999.

[11] W. A. Jansen, "Inheritance Properties of Role Hierarchies," 21th National Information Systems Security Conference, Oct 1998.

이 상 하

e-mail : shyi@haksan.dsc.ac.kr

1987년 울산대학교 전자계산학과 졸업(공학사)

1991년 아주대학교 대학원 컴퓨터공학과 졸업(공학석사)

1999년 아주대학교 대학원 컴퓨터공학과(박사수료)

1991년~1992년 (주)큐닉스 컴퓨터

1993년~1999년 (주)케이엔아이시스템

2000년~현재 동서울대 전자통신과 전임강사

관심분야 : 정보통신 Security, 네트워크 관리, 분산처리 시스템 보안

조 인 준

e-mail : injune@woonam.paichai.ac.kr

1982년 전남대학교 계산통계산학과 졸업(학사)

1985년 전남대학교 대학원 전자계산학과 졸업(석사)

1999년 아주대학교 대학원 컴퓨터공학과 졸업(박사)

1990년 정보처리 기술사(전산 조직 응용)

1983년~1994년 한국 전자 통신 연구소(선임연구원)

1994년~현재 배재대학교 컴퓨터공학과 교수

관심분야 : 정보통신 Security, 컴퓨터 네트워크(이동컴퓨팅), 전산조직응용

천 은 홍

e-mail : ehcheon@core.woosuk.ac.kr

1981년 광운대학교 응용전자공학과 졸업(공학사)

1985년 아주대학교 대학원 전자공학과 졸업(공학석사)

1998년 아주대학교 대학원 컴퓨터공학과 졸업(공학박사)

1985년~1988년 삼성종합기술원 정보시스템연구소 주임연구원

1998년~2000년 우석대학교 전자계산소 소장

1988년~현재 우석대학교 정보통신컴퓨터공학부 부교수

관심분야 : 컴퓨터 네트워크, 정보보호, 정보통신 보호

김 동 규

e-mail : dkkim@madang.ajou.ac.kr

1973년 서울대학교 공과대학 졸업(학사)

1979년 서울대학교 자연과학대학원 졸업(석사)

1984년 미국 Kansas 주립대 대학원 졸업(전산학 박사, 정보통신 전공)

1981년~1982년 미국 Kansas 주립대 전산학과 교수

1979년~현재 아주대학교 정보 및 컴퓨터공학부 교수, 한국통신학회 상임이사, 한국통신정보보호학회 부회장 역임

관심분야 : 컴퓨터 네트워크, 정보통신 프로토콜 엔지니어링, 정보통신 Security