

## 안전한 직무 기반 접근 제어에 대한 연구

이 호\*

### A Study on Secure Role-Based Access Control

Ho Lee\*

#### 요 약

본 논문에서는 실제적인 접근 제어 시스템에 적용이 가능하도록 기존의 검증된 접근 제어 메커니즘을 적용하여 보안성, 무결성 및 흐름제어 보안 기능을 제공하며 직무 중심 조직의 접근 제어 요구를 용이하게 수용할 수 있는 안전한 직무 기반 접근 제어 모델을 제안한다. 제안한 모델은 주로 인터넷 상에서 웹을 기반으로 하는 응용 시스템에 적용할 수 있는 간단하면서도 안전한 접근 제어 모델이다.

#### Abstract

In the paper, is proposed a secure role-based access control model that not only has s functions such as security, integrity and flow control, but also can easily meet access requirements of role-based social organizations. The proposed role-based access control mod designed based on proven existing rule-based access control mechanisms in order to be app real access control systems. The model proposed in the paper is simple and secure. It can be used for the web-based application systems working on the Internet.

---

\* 송호 대학 정보 산업 계열 부교수

식별자로서 여기서는 특정한 직무이다. 각 객체 식별자와 이에 따른 보안성 등급, 무결성 등급 및 소유권자가 객체의 접근 제어 정보에 명시된다(7)(표2 참조).

## I. 서론

직무 기반 접근 제어는 신분 기반 및 규칙 기반 접근 제어(1,2,3)의 특성을 포함하고 있는 방법으로서, 개별 신분이 아닌 자신의 직무에 따라 접근할 수 있는 정보가 결정되고 사용할 수 있는 정보의 한계가 정해진다.

규칙 기반 접근 제어가 신분 기반 접근 제어의 완전한 대체 방법이 아니듯이 직무 기반 접근 제어(4)도 신분 기반 접근 제어와 규칙 기반 접근 제어의 병합이 아닌 상호 보완적인 방법이다. 따라서 이러한 세 가지 기존의 접근 제어 기술에 근간을 둔 새로운 접근 제어 모델의 설계가 가능하다.

본 논문에서는 실제적인 접근 제어 시스템에 적용이 가능 하도록 검증된 접근 제어 메커니즘을 수용하여 보안성, 무결성 및 흐름제어 보안 기능을 제공하며 직무 중심 조직의 접근 제어 요구를 용이하게 수용할 수 있는 안전한 직무 기반 접근 제어 모델을 제안한다.

## II. 안전한 직무 기반 접근 제어 모델 설계

### 2.1 접근 제어 정보의 설계

주체에 있어서, 직무는 실체가 수행할 수 있는 역할을 말한다. 직무는 응용 분야에 따라 고유한 직무를 정의할 수 있으며 각각의 직무는 서로 다른 권한을 부여 받을 수 있다. 각 직무 식별자와 이에 따른 보안성 등급, 무결성 등급이 주체의 접근 제어 정보에 명시된다(표1 참조).

객체에 있어서, 식별자는 시스템에서 주체가 객체에 접근을 시도할 때 객체를 식별할 수 있도록 하는 식별자이다. 소유권자는 객체의 소유권을 가지는 실체를 나타내는

표 1. 주체 접근 제어 정보  
Table. 1 Subjects' access control info.

직무 식별자	보안성 등급	무결성 등급
r1	Top Secret	Crucial
r2	Secret	Very Important
r3	Secret	Important
r4	Confidential	Important

표 2. 객체 접근 제어 정보  
Table. 2 Objects' access control info.

객체 식별자	보안성 등급	무결성 등급	소유권자
o1	Top Secret	Crucial	r1
o2	Secret	Important	r2
o3	Secret	Very Important	r3
o4	Confidential	Important	r2

### 2.2 접근 제어 규칙 표기법 정의

집합 표현에서의 대소문자는 집합과 원소를 나타내며, 함수 표현에서의 매개 변수는 실체를 나타낸다.

- S : 접근 제어의 주체 집합, s S
- O : 접근 제어의 객체 집합, o O
- P : 접근 허가 집합, P=c, r, w, x, d p ∈ P.  
(c :create, r :read, w :write, x :execute, d :delete)

- S\_Level(a) : 보안성 등급 함수
- I\_Level(a) : 무결성 등급 함수
- permit(role, a) : role의 a에 대한 접근 권한 검색 함수
- owner(a) : 소유권자 함수
- get\_ACI(a) : ACI 요구 함수
- exist\_ACI(a) : ACI에 대해서 a의 존재 여부 확인 함수

### 2.3 접근 제어 규칙 설계(6,7)

규칙 1 : 주체(s)가 객체(o)에 대하여 수행하고자 하는 접근 허가(p)가 있을 때 주체와 객체는 ACI에 존재해야 하며, 주체가 객체에 대해 요구한 접근 허가가 부여될 수 있는 접근 제어 정보가 명기되어 있어야 한다.

```

role_acr(s, o, p)
{
  if ( exist_ACI(s) and exist_ACI(o) ) then
    role ← get_ACI(s)
    if ( p == permit(role, o) ) then
      return TRUE
    else if
      return FALSE
    endif
  else if
    return FALSE
  endif
}

```

규칙 2 : 보안성 보장을 위하여 주체의 보안성 등급이 객체의 보안성 등급을 지배할 때는 read와 execute 권한을 허용하고, 주체의 보안성 등급이 객체의 보안성 등급과 동일할 때는 create와 delete권한을 허용한다. 무결성 보장을 위해서는 주체의 무결성 등급과 객체의 무결성 등급이 일치할 때만 create, write, execute 및 delete 권 허용한다.

```

rule_acr(s, o, p)
{
  case p = 'c'
    if ( S_Level(s) == S_Level(o) and
        I_Level(s) == I_Level(o) )
      then
        return TRUE
      endif
  case p = 'r'
    if ( S_Level(s) ≥ S_Level(o) and
        I_Level(o) ≥ I_Level(s) )
      then
        return TRUE

```

```

      endif
    case p = 'w'
      if ( s == owner(o) and
          S_Level(s) == S_Level(o) and
          I_Level(s) == I_Level(o) )
        then
          return TRUE
        endif
    case p = 'x'
      if ( S_Level(s) ≥ S_Level(o) and
          I_Level(s) == I_Level(o) )
        then
          return TRUE
        endif
    case p = 'd'
      if ( s == owner(o) and
          S_Level(s) == S_Level(o) and
          I_Level(s) == I_Level(o) )
        then
          return TRUE
        endif
    otherwise
      return FALSE
}

```

규칙 3 : 상위의 보안성 등급을 가진 객체가 하위의 보안성 등급을 가진 객체로 상위 보안성 등급의 정보를 불법적으로 전달해 주는 것을 방지하기 위해서 흐름제어가 필요한데, 주체(s)가 한 객체가 소유한 정보(o1)를 다른 객체(o2)로 전달하기 위한 동작을 하기 위해서는 주체의 두 객체에 대한 보안성 등급이 지배 관계에 있어야 하고 두 객체간에는 보안성 등급과 무결성 등급이 일치해야만 한다.

```

flow_acr(s, o1, o2)
{
  if ( s == owner(o1) and
      S_Level(s) ≥ S_Level(o1) and
      S_Level(s) ≥ S_Level(o2) and
      S_Level(o1) == S_Level(o2) and
      I_Level(o1) == I_Level(o2) )
    then

```

```

return TRUE
else if
return FALSE
endif
}
    
```

2.4 접근 제어 규칙 적용 모델 설계

앞에서 설계한 접근 제어 규칙을 적용하는 모델을 그림1에서와 같이 구성한다(7).

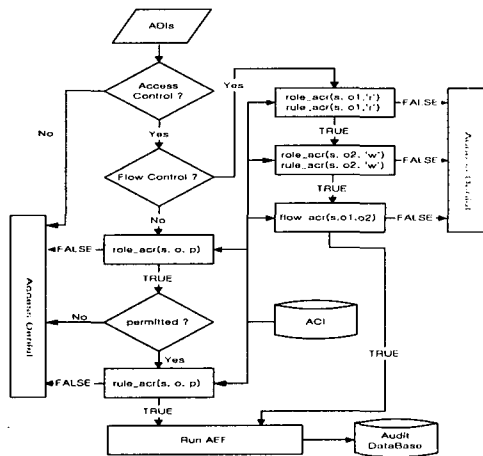


그림 1. 접근 제어 규칙 적용 모델  
Fig. 1 Model of applying access control rules

III. 안전한 직무 기반 접근 제어 모델의 특징

본 논문에서 제안한 안전한 직무 기반 접근 제어 모델 (SRM)은 전통적으로 UNIX와 같이 사용자를 접근 제어의 주체로 하는 운영 체제나 최근의 Windows 2000처럼 사용자와 그룹을 접근 제어의 주체로 하는 운영 체제들이 사용하는 접근 제어 방법과 비교하여 다음의 특징을 갖는다.

첫째, UNIX와 Windows 2000은 파일 시스템이 제공하는 기존의 ACL 등과 같은 접근 허가 정보를 이용하여 정적인(static) 접근 제어를 수행하지만 SRM은 run-time

접근 제어 규칙을 적용함으로써 접근 허가를 결정하는 동적인(dynamic) 접근 제어 방법을 사용한다.

둘째, UNIX와 Windows 2000의 접근 제어 리스(ACL)나 보안 서술자(SD)와 같은 접근 제어 정보는 많은 기억 공간을 필요로 하지만(5) SRM은 접근 제어 규칙 적용의 근거가 되는 주체와 객체의 최소한의 보안 정보만(주체와 객체의 ACI만) 유지하면 되므로 많은 기억 공간을 필요로 하지 않는다.

셋째, Windows 2000 조차도, 최근의 접근 제어 기술서 그 관리의 편리성 때문에 인정을 받고있는 직무 기반 접근 제어의 모델 측면에서 보면, 직무 기반 접근 제어의 특성 중에 하나인 상위 계층으로의 상속을 반영치 못하고 있으나, SRM은 통상적인 직무 기반 접근 제어의 메커니즘으로 동작하므로 상위 계층의 보안 정보가 하위 계층의 보안 정보를 상속할 수 있다.

넷째, UNIX나 Windows 2000 운영체제의 접근 제어 모드는 파일 시스템에 근거하여 결정되지만 SRM의 경우는 모델이 가지는 접근 제어 규칙에 따라 접근 제어 모드가 정해진다.

다섯째, UNIX나 Windows 2000 운영체제의 접근 제어는 범용성을 위하여 일반적인 컴퓨팅 환경에서 객체에 대한 접근 제어 요구를 수용할 수 있도록 되어있는데 비하여, SRM은 웹 기반 응용 시스템이라는 웹 서버 상에서 동작하는 시스템의 접근 제어를 목적으로 설계된 특수 목적의 접근 제어 모델이다.

여섯째, SRM은 UNIX나 Windows 2000의 경우와 다르게, 보안이 취약한 웹을 기반으로 동작하는 응용 시스템을 위해 엄격한 보안 규칙을 적용하는 안전한 접근 모델이므로, 일반적인 파일 시스템의 기능을 이용하여 객체별로 복잡한 접근 제어를 수행하는 방법과 비교하면 간단하지만 확실한 보안 체계를 제공한다고 할 수 있다.

IV. 접근 제어 모델의 비교

여기서는 SRM의 우수성을 입증하기 위해서 앞장의 안전한 직무 기반 접근 제어 모델의 특징 중에서 접근 제어 정보의 측면에서 Windows 2000의 경우와 비교하고자

한다.

4.1 주체의 접근 제어 정보

주체의 접근 제어 정보를 표시하기 위하여 SRM은 주체의 ACI를 사용하고 Windows 2000은 액세스 토큰을 용한다(5). 표3과 표4는 각각 하나의 주체를 위한 ACI 액세스 토큰의 구조를 보여준다. SRM의 경우에는 사용자가 복수의 직무를 부여받는 경우도 있지만 주체 ACI를 기준으로 한다면 모든 직무는 ACI에 하나의 엔트리로서 표시된다. Windows 2000의 경우에는 사용자 별로 복의 그룹 보안 식별자 및 특권이 부여될 수 있으므로 주체의 접근 제어 정보를 표시하기 위하여 SRM의 경우보다는 많은 양의 기억 공간과 복잡한 데이터 구조가 필요하다는 것을 알 수 있다. Windows 2000의 경우에 사용자가 여러 그룹에 속하는 경우에는 그에 따라 특권도 늘어나기 때문에 필요한 정보를 저장하기 위한 기억 공간이 그 만큼 더 요구되며 따라서 SRM의 경우와는 다르게 가변적인 크기의 데이터 구조를 사용해야 한다는 사실을 알 수 있다.

표 3. 주체의 ACI  
Table. 3 Subject's ACI

직무 식별자	보안성 등급	무결성 등급
r1	Top Secret	Crucial

표 4. 주체의 액세스 토큰  
Table. 4 Subject's access token

User SID	Group SIDs	Privileges
Jane Use	Administrators Power Users Service Operators Users	Start Service Load Device Driver Shut Down

4.2 객체의 접근 제어 정보

객체의 접근 제어 정보를 표시하기 위하여 SRM의 경우는 표5에서처럼 객체의 ACI를 사용하는데 Window 2000은 그림2에서처럼 보안 서술자를 사용한다(5).

표 5. 객체의 ACI  
Table. 5 Object's ACI

객체 식별자	보안성 등급	무결성 등급	소유권자
o1	Top Secret	Crucial	r1

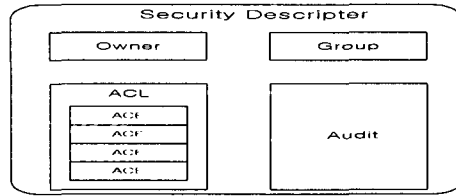


그림 2. 객체의 보안 서술자  
Fig. 2 Object's security descriptor

SRM의 경우에는 표5에서 보여 주듯이 모든 객체가 동일한 구조의 접근 제어 정보를 갖는다. 반면에 Windows 2000의 경우에는 그림2에서 보듯이 SRM의 경우보다는 아주 복잡한 데이터 구조를 가진다. 그림에서 보안 서술자를 구성하는 구성 요소 중에 ACL이 있는데 이 ACL은 다시 임의 개수의 ACE로 구성된다. 그림2는 네 개의 ACE로 이루어진 ACL의 예를 보여준다. ACE는 객체를 접근할 수 있는 접근 허가가 주어진 주체, 접근 허가 여부 및 접근 허가 종류 등을 나타내는 정보를 포함하고 있다. 이 경우에 ACE는 사용자 또는 그룹과 같이 접근 허가를 부여 받는 주체별로 한 엔트리씩 존재한다. 즉 접근 허가를 부여 받은 사용자 및 그룹의 총 숫자 만큼의 ACE가 ACL에 존재한다는 것을 의미한다. 이 경우에도 주체의 접근 제어 정보에서와 같이, 접근 제어 정보를 구성하는 요소 정보를 표시하는 정보의 표시 단위 및 데이터 구조 등이 구현 방법에 따라 다를 수 있기 때문에 정확한 정량적인 비교는 어렵고, 객체의 접근 제어 정보의 양이 SRM의 경우는 항상 일정한 크기이므로 이것을 1로 보았을 때 Windows 2000의 경우에는 대략 객체 접근이 허가된 주체 전체(r, 여기서 r은 ACE의 총 개수와 동일함)의 k(SRM의 경우를 1로 보았을 때 k > 1) 배수(multiplication)의 접근 제어 정보 저장을 위한 기억 공간(S)이 필요하다. 이를 수식으로 나타내면 S = kr (r = m + n, 사용자 수, n은 그룹 수) 이다. 이 경우에 보안 서술자의 소유자, 그룹 및 감사 정보 저장 공간은 계산

에 산입하지 않았다. 이러한 비교치가 증명하듯이 객체의 접근 제어에 필요한 정보량에 있어서 SRM이 Windows 2000 보다 탁월한 기억 공간의 효율성을 갖고 있다고 할 수 있다.

## V. 결론

본 논문에서는 기존의 접근 제어 모델들이 서로 다른 보안 기능과 특성을 가지고 있으면서 동시에 상호 보완적인 요소들도 가지고 있다는 사실에 착안하여 보안 기능을 만족 시킬 수 있으면서도 효율적인 관리가 가능한 안전한 접근 제어 모델(SRM)을 제안하였다. 보안성 무결성 및 흐름제어가 가능한 안전한 접근 제어 모델 제안을 위하여 접근 제어 규칙을 정의하고 이를 토대로 하여 접근 제어 규칙 적용 모델을 설계하였다. SRM은 직무에 접근 허가를 할당하는 한 방법으로서 보안 속성 정보에 근거하여 보안 규칙을 적용하는 간단하면서도 안전한 방법이다.

아울러, 본 논문에서는 윈도우즈 2000의 접근 제어 방법과 비교하였을 때 SRM이 가지고 있는 특징을 제시하고 SRM의 접근 제어 정보 저장에 위한 기억 공간의 탁월한 효율성도 입증하였다.

Techniques", Prentice Hall, pp. 149-176, 1994.

- [4] Ravi S. Sandhu, Edward J. Coyne, "Role-Access Control Models", IEEE Computer, 38-47, Feb., 1996.
- [5] Michael M. Swift, "Improving the Granular Access Control in Windows NT", In Proc. of ACM RBAC 2001, pp. 87-96, 2001.
- [6] 이호, 정진욱, "안전한 인터넷 사용을 위한 접근 제어 메커니즘 설계", 한국 OA 학회 논문지, Vol. 5 No. 3, pp. 84-90, 2000.
- [7] 이호, 정진욱, "직무-기반 접근 제어 모델 설계", 한국 OA 학회 논문지, Vol. 6 No. pp. 60-66, 2001.

## 저자 소개



이 호

1989 벨기에 V.U.B. 대학원 (공학 석사)

2002. 2 성균관 대학교 대학원 박사 과정 졸업 예정 (공학 박사)

1982-1991 한국전자통신연구원 선임 연구원

현재 송호 대학 정보 산업 계열 부교수

## 참고 문헌

- [1] ISO, "OSI - Security Frameworks in Systems - Part 1: Security Framework Overview", ISO/IEC DIS 10181-1, 1993.
- [2] ISO, "OSI - Security Frameworks in Systems - Part 3: Access Control", ISO/IEC DIS 10181-3, 1993.
- [3] Warwick Ford, "Computer Communication Security - Principles, Standard Protocols