情報保護學會論文誌
第20卷 第5號, 2010. 10

내부자의 불법적 정보 유출 차단을 위한 접근통제 모델 설계

엄 정 호,† 박 선 호,‡ 정 태 명
성균관대학교

An Architecture of Access Control Model for Preventing Illegal Information Leakage by Insider

Jung-ho Eom,† Seon-ho Park,‡ Tai M. Chung
Sungkyungkwan Universityty

요 약

본 논문에서는 유비쿼터스 컴퓨팅 환경에서 내부자가 합법적인 권한을 이용하여 불법적인 정보 유출 행위를 차단하는 접근통제 모델 IM-ACM(Insider Misuse-Access Control Model)을 제안하였다. IM-ACM은 상황역할과 개체의 보안속성을 활용하여 보안성을 강화시킨 CA-TRBAC(Context Aware-Task Role Based Access Control)에 오용 모니터 기능을 추가하여 내부자가 데이터를 올바르게 사용하는지 감시한다. 내부자에 의한 정보 유출은 합법적인 접근권한, 접근시스템에 대한 풍부한 지식 등의 내부자의 특성으로 인해 차단하기가 곤란하다. IM-ACM은 CA-TRBAC의 장점인 상황과 보안속성을 이용하여 서로 상이한 보안등급의 객체간 정보 흐름을 방지하고 오용 모니터를 활용하여 내부자의 실제 진행 프로세스를 최근 역할, 직무와 작업 프로세스 패턴 프로파일과 비교하여 내부자의 오용행위를 차단한다.

ABSTRACT

In the paper, we proposed an IM-ACM(Insider Misuse-Access Control Model) for preventing illegal information leakage by insider who exploits his legal rights in the ubiquitous computing environment. The IM-ACM can monitor whether insider uses data rightly using misuse monitor add to CA-TRBAC(Context Aware-Task Role Based Access Control) which permits access authorization according to user role, context role, task and entity's security attributes. It is difficult to prevent information leakage by insider because of access to legal rights, a wealth of knowledge about the system. The IM-ACM can prevent the information flow between objects which have the different security levels using context role and security attributes and prevent an insider misuse by misuse monitor which comparing an insider actual processing behavior to an insider possible work process pattern drawing on the current defined profile of insider's process.

Keywords: IM-ACM, Insider threat, Access Control

I. 서 론

최근 2010 Cyber Security Watch Survey[1]에 의하면 2009년 한 해 동안 발생한 보안사고 중 26%가 내부자에 의해 발생한 것이라고 밝혔다. 오늘날 내부자의 침입 행위는 네트워크 및 기반시설 보안에 가장 심각한 위협으로 부각되고 있다. 또한, 조직의 정보나 데이터 유출은 외부자 침입에 의한 것보다 내부자의 침입에 의해서 발생하는 경우가 많다[2-4]. 유비쿼터스 컴퓨팅 환경[5]에서는 내부자가 언제 어

접수일(2010년 5월 6일), 수정일(1차: 2010년 6월 17일, 2차: 2010년 7월 14일), 게재확정일(2010년 9월 13일)
† 주저자, eomhun@gmail.com
‡ 교신저자, shpark@imtl.skku.ac.kr

디서든지 합법적인 권한으로 데이터에 접근할 수 있기 때문에 정보 유출에 대한 탐지는 더 어려워졌다.

지금까지 내부자에 대한 침입 차단 및 탐지 기법 중에 접근제어 모델이 가장 많이 사용되고 있다. 특히, 역할기반 접근제어(RBAC)[6]나 상황인식 역할기반 접근제어(C-RBAC)[7,8] 모델 등을 이용한 내부자 접근 차단 시스템이 대표적이다. 그러나 RBAC은 상황요소를 반영하지 않았기 때문에 동적인 접근제어가 불가능하고, C-RBAC은 접근하려는 객체간의 보안등급을 고려하지 않아 정보의 기밀성과 무결성을 보장하지 못한다. 또한, 자신이 수행하고 있는 직무와 관련된 객체들에 합법적으로 접근하여 정보를 유출하는 것을 차단하지 못한다.

본 논문에서는 상황정보 기반으로 내부자의 직무, 역할과 보안속성을 적용하여 정보의 기밀성을 확보하는 접근제어 메커니즘(CA-TRBAC)[12]에 내부자의 실제 행동 프로세스를 감시하는 오용행위 모니터 기능을 추가한 IM-ACM(Insider Misuse-Access Control Model)을 제안한다. 본 논문은 2장에서는 관련연구를 소개하고 3장에서는 제안된 접근통제 모델(IM-ACM)을 설명하며, 4장에서는 제안된 모델과 기존의 모델의 비교평가를 서술한다. 마지막으로 5장에서 결론을 맺는다.

II. 관련연구

2.1 내부자 위협

내부자는 내부 네트워크, 서버, 데이터 등의 정보통신체계에 대해 합법적인 접근권한을 가지고 언제든지 컴퓨터 및 네트워크의 구성, 프로그램, 데이터 등에 대한 정보를 열람하거나 변경할 수 있는 고용인을 의미한다[4]. 정식 직원, 임시/계약직 직원, 계약자, 하청업자 등이 포함된다. 이러한 내부자들이 경제적 이익, 이직 등의 동기로 내부 정보시스템에 대해 악의적인 행동을 한다면 외부자에 의한 침입보다 더 큰 피해를 초래할 수 있다. 카네기 멜론 대학의 연구소에서는 내부자 침입[9]이 다음과 같은 유형으로 발생한다고 밝혔다.

- 개인적 이익을 위해 기업의 기밀사항이나 중요한 정보를 변경하거나 유출하는 경우
- 사업 이득이나 외국 정부/조직에게 주기 위해 고객의 거래 및 개인 정보를 유출하는 경우
- 조직 내부의 네트워크, 시스템, 데이터 등을 기술적으로 정교하게 파괴하는 경우

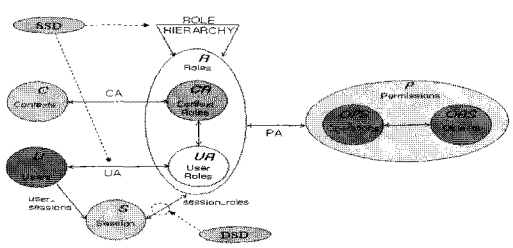
최근에는 네트워크나 시스템보다는 데이터, 전자문서, 고객정보를 대상으로 자신의 권한을 이용하여 데이터에 불법적으로 접근하는 사례가 증가하고 있다. 또한, 자신이 접근 가능한 객체간의 불법적인 정보의 흐름을 발생시켜 중요한 정보를 유출하곤 한다[1]. 본 논문에서는 내부자가 자신의 권한을 이용하여 정보를 유출하는 것을 차단하는 데 목표를 둔다. 내부자는 자신의 역할과 직무에 합법적인 접근권한을 이용하여 서버에 접속한 후 데이터를 파괴, 변경, 유출한다. 대부분 내부자의 시스템 접근 보안정책은 소속, 역할, 직책 등의 그룹별로 접근권한을 부여함으로써 내부자의 데이터에 대한 불법적인 정보 유출을 효과적으로 차단하지 못한다. 또한, 기존의 접근제어 모델도 역할, 직무, 역할-직무, 상황 등의 한 두가지 구성요소로만 접근권한을 승인하고 있어 내부자에 대한 강력한 보안통제가 어렵다.

2.2 접근제어 모델

2.2.1 상황-역할기반 접근제어(C-RBAC)

역할기반 접근제어(RBAC: Role-Based Access Control) 모델[6,10]은 권한을 역할과 연관시켜 사용자들에게 적절한 역할을 할당한 후, 역할에 따라서 권한을 부여함으로써 객체에 대한 접근을 제어한다. 역할은 조직에서 다양한 작업 기능들을 바탕으로 정의되고, 사용자들은 직무에 따른 책임과 자질을 바탕으로 역할을 할당 받기 때문에 권한의 관리를 용이하게 한다. 그러나 RBAC은 상황인식 기술이 적용된 유비쿼터스 컴퓨팅 환경에서 동적인 접근제어를 수행하는 데는 한계가 있다. 이러한 약점을 보완한 접근제어 모델이 상황-역할기반 접근제어(C-RBAC: Context- Role Based Access Control)[11] 모델이다.

C-RBAC 모델은 [그림 1]과 같이 유비쿼터스 컴



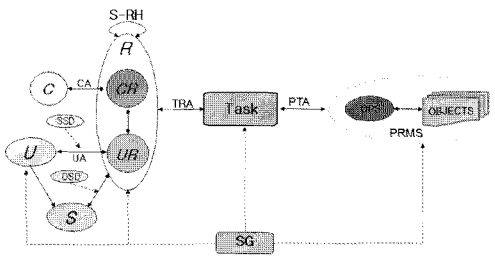
[그림 1] CRBAC 모델

퓨팅 시스템에서 역할기반 접근제어 메커니즘에 관리 도메인 내의 다양한 상황정보들 중 보안에 관련된 요소들을 추상화시킨 상황-역할을 추가하였다. 따라서 역할기반 접근제어의 주체-역할 특성을 그대로 유지하며, 역할 활성화/비활성화, 역할 계층화 등의 특징을 모두 갖는다. 즉, 사용자의 상황 요소들도 고려하여 객체 접근권한 부여에 활용한다. C-RBAC의 구성요소는 다음과 같다.

- U(Users): 접근이 통제되는 개체로서 사용자(user)의 집합 표시
- C(Contexts): 상황정보는 시간, 위치, 온도, CPU 사용량 등으로 C는 상황정보들의 집합
- R(Roles): C-RBAC의 역할은 사용자-역할과 상황-역할을 포함하며, R은 역할들의 집합
- UR(User Roles): 사용자-역할은 RBAC에서의 역할과 같으며, UR은 사용자-역할들의 집합
- CR(Context Roles): 모든 상황정보들 중 보안에 연관되는 정보들을 추상화시킨 개념. CR은 상황-역할들의 집합
- P(Permissions): RBAC에서의 권한과 같은 개념
- S(Sessions): 사용자-역할, 상황-역할의 활성화
- SSD(Static Separation of Duty) : 사용자-역할의 할당 또는 역할의 상속 시에 제약사항 적용
- DSD(Dynamic Separation of Duty) : 사용자-역할의 할당 관계가 아닌 세션이 생성될 때 적용. 즉, 특정 역할에 할당된 사용자라도 DSD에 의해 역할 활성화에 제약을 받게 됨.

2.2.2 상황인식-직무 역할기반 접근제어(CA-TRBAC)

역할기반 상황 인식-직무/역할기반 접근제어(CA-TRBAC: Context Aware-Task Role Based Access Control)[12,14] 모델은 유비쿼터스 컴퓨팅 환경에서 T-RBAC 모델[17]을 기반으로 상황인식 메커니즘과 보안등급 속성을 추가하여 보안정책을 구성하고 접근을 제어한다. CA-TRBAC은 [그림 2]와 같이 접근제어 대상인 사용자역할(UR)과 그 역할들의 집합이 접근제어 개체가 되고 사용자역할의 제약사항과 상황역할(CR)을 확인한 후 역할(R)-직무(T), 직무(T) -객체(O)의 적합한 매핑상태와 사용자-역할-태스크-객체의 보안등급(SG: Security Grade)을 고려하여 접근권한(PRMS)을 결정한다. 권한-직무의 관계는 $PTA(PRMS\,Task\,Assign) \subseteq P(Permission) \times T(Task)$으로 권한과 직무 간의 다대다 할당관계를 나타낸다. 권한은 정보 객체와 접근 모드의 한 쌍으로 정의되며 $P = \{o \in Object, \langle read \text{ or } write \rangle \in Mode | (o, read)\}$ 으로 표현된다.



[그림 2] CA-TRBAC 모델

2.2.3 내부자 통제에 대한 문제점

RBAC 모델은 역할에 의존하여 객체에 대한 접근을 허가하기 때문에 내부자의 상황조건에 따른 접근을 제어할 수 없다. 그래서 내부자가 업무시간 이외에 다른 부서의 PC를 활용하여 객체에 접근한 후 작업을 수행하더라도 통제하지 못한다. 또한, 객체에 대한 보안등급을 고려하지 않기 때문에 내부자에 의해 정보의 기밀성과 무결성을 훼손될 수 있다. 즉, 하위 보안등급의 내부자가 상위 보안등급의 역할에 할당되거나 하위 보안등급의 객체와 상위 보안등급의 객체를 동시에 열람 또는 수정할 수 있기 때문에 정보의 흐름이 발생할 수 있다.

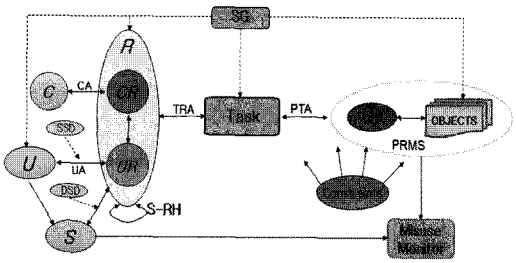
C-RBAC 역시 내부자에 의한 정보의 유출을 차단하는 데는 한계가 있다. 예를 들어, 시스템 관리자가 본인의 계정으로 문서관리 시스템에 접근하여 서로 다른 보안등급의 문서를 동시에 활성화하여 편집할 경우에는 상황정보와 역할만으로 접근을 통제하기가 어렵다. 일반적으로 시스템 관리자에게 문서관리 시스템에 저장되어 있는 데이터 접근을 지속으로 부여하지는 않는다.

CA-TRBAC은 내부자가 자신의 권한을 이용하여 객체에 접근하여 오용하는 것을 효과적으로 차단하지 못한다. 예를 들면, 사용자가 합법적인 권한으로 역할을 활성화하고 직무에 따라 객체에 접근하여 임의적으로 수정하거나 작성하는 것을 차단하지 못한다.

III. 제안된 접근통제 모델

3.1 내부자의 오용행위에 대한 접근통제 모델 (IM-ACM)

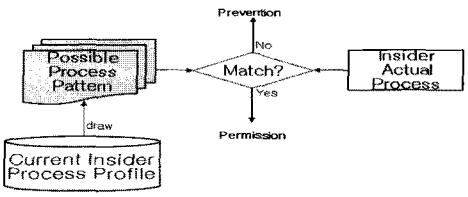
내부자의 부정행위에 대한 접근통제 모델(IM-ACM: Insider Misuse-Access Control Model)은 기존의 접근제어 모델들의 약점을 보완하여 내부자가 자신의 권한을 오용하여 정보나 데이터에 대한 부정행위를 차단하는 모델이다. IM-ACM은 CA-TRBAC을 기반으로 하여 사용자가 실제로 수행하는 프로세스를 감시할 수 있는 오용행위 모니터(Misuse Monitor)[13] 기능을 추가하였다. 이것은 최근에 내부자가 역할과 직무에 따라 수행하고 있는 작업 프로세스 패턴과 실제로 수행하는 프로세스를 비교함으로써 내부자의 오용행위를 차단한다. 내부자의 행동패턴에 대한 프로파일링은 조직에서 수년간 근무를 하였거나 최근 직무에 대해서 오랫동안 수행하였기 때문에 외부자에 비해 보다 더 정확하고 상세하게 작성할 수 있다. [그림 3]은 CA-TRBAC 모델에 오용행위 감시 기능을 추가한 IM-ACM을 나타낸다.



[그림 3] IM-ACM

오용행위 모니터(Misuse Monitor)는 내부자가 소속되어 있는 조직과 부서, 사용하는 운영시스템과 응용체제, 그리고 접근하려는 객체와 활성화된 세션을 통해 내부자의 행동을 감시한다. 또한, 내부자가 현재 수행하고 있는 직무에 따라 역할을 활성화한 후 객체에 접근하여 작업을 수행할 때, 내부자의 객체에 대한 실제 프로세스를 최근 내부자 행동 프로파일에 있는 내부자 가능 작업 프로세스 패턴과 비교하여 객체에 대한 오용행위를 차단한다. 예상되는 내부자의 행동패턴은 가장 최근에 객체에 대한 행동 패턴의 최다 통계로 결정한다. 오용 모니터의 구성은 다음 [그림 4]와 같다.

IM-ACM의 접근제어 정책 규칙은 ⟨⟨*manager, work_available_ctx, development_scheduling-report, preparation_write⟩ and ⟨insider_misuse⟩, Allow⟩*와 같이 구성될 수 있다. "manager"는 사용자-역할을 의미하며, "work_available_ctx"는 상황-역할을 나타낸다. 또한, "development_schedulingreport"는 직무-역할을 의미하며, "preparation_write"는 퍼미션을 나타내는 것으로 개발계획보고서라는 객체에 "write" 권한을 부여한다는 것을 의미한다. 즉, 매니저가 개발계획보고서를 작성하기 위한 상황, 직무의 적합성을 판단하는 것이다. 그리고 "insider_misuse"는 내부자의 오용행위 진위를 판단한다. "insider_misuse"은 트랜잭션이 진행되는 동안에 내부자의 실제 프로세스가 최근 직무와 관련한 내부자의 가능 작업 프로세스 패턴과 상이할 경우에 오용행위로 간주하여 내부자의 행동을 차단하는 것으로 "true, false"로 표시한다. 마지막으로 "Allow"는 허가 비트로서 해당 트랜잭션의 조건을 만족할 경우 권한에 대한 허가를 의미하며, 1비트로 "+" 또는 "-"로 표시한다.



[그림 4] Misuse Monitor의 구성

3.2 IM-ACM의 내부자 오용행위 차단 기법

3.2.1 정보 유출 차단

IM-ACM은 내부자의 역할과 직무를 고려하여 시스템 접속을 허가하고 객체와 역할, 직무의 보안등급을 확인하여 객체에 대한 접근을 허가하며 내부자가 수행하는 실제 프로세스를 감시하여 오용행위를 차단한다. 내부자가 활성화된 역할과 할당된 직무에 따라서 객체에 접근할 때 정보의 흐름[16]이 발생할 수 있다. 즉, 보안등급이 하위인 내부자의 직무에 따라 상위 보안등급의 객체에 "read" 권한을 승인하고 하위 등급의 객체에 "write" 권한을 승인하면 상위 등급에서 하위 등급으로의 정보 흐름(read up, write down)이 발생한다. 반대로, 보안등급이 상위인 내부자의 직무가 하위 등급의 객체에 "read" 권한을 승인하고 상위 등급의 객체에 "write" 권한을 승인하면 하위 등급에서 보안등급으로의 정보 흐름(read down,

write up)이 발생한다. 본 논문에서는 정보의 유출이 하향식 정보의 흐름에 기인한다는 것을 착안하여 하향식 정보 흐름 차단에 중점을 둔다.

보안등급(SG: Security Grade)은 "Secret(S)", "Confidential(C)", "Unclassified(U)"의 집합이며, $SG=\{S, C, U\}$로 표현된다. SG도 역할과 마찬가지로 상속이 가능하다. SG의 상속관계는 다음과 같이 표현할 수 있다.

$$SG_1, SG_2 \in SG, SG_1 > SG_2,$$
$$\forall SG_i \in SG_2 \Rightarrow SG_1 inherits \{SG_i\}$$

우선 내부자가 직무에 따라 객체에 접근할 때 정보의 흐름이 발생하지 않도록 직무와 객체간의 권한 부여를 명확히 해야 한다. 다음 〔표 1〕은 직무와 접근하려는 객체간 보안등급에 따라 접근권한을 부여하는 알고리즘이다.

내부자가 접근하려고 하는 객체들간에 정보의 유출이 발생할 수 있다. 즉, 직무와 상이한 보안등급의 객체에 접근권한을 부여하거나 서로 다른 보안등급의 객체에 접근을 허가할 경우에 객체간의 정보의 흐름이 발생한다. 그리고 두 개의 보안등급이 상이한 객체에 정보 흐름(read up & write down)이 발생하기도 한다. 따라서 직무와 동일한 보안등급의 객체에만 접근하는 권한을 부여하거나 서로 다른 보안등급의 객체에 접근을 허가하되 "read" 권한만 승인하고 상위 보안등급의 객체에는 "write" 권한 승인, 하위 보안등급의 객체에는 "read" 권한만 승인해야 한다. 〔표 2〕는 객체간 정보유출을 차단하기 위한 객체의 보안등급에 따른 접근권한 할당 알고리즘을 보여준다.

우선, 직무 T에 연관된 객체 집합 O를 식별한다. 직무 T에 연관된 객체가 O={o1}일 경우, "read"와 "write" 접근권한을 모두 부여한다. 만약, 직무 T에 포함된 객체 집합이 O={o1, o2}일 경우, 두 개의 객체 {o1, o2}가 서로 같은 보안등급이면 "read"와 "write" 접근권한을 모두 부여할 수 있다. 객체 {o1, o2}가 서로 다른 보안등급일 경우에는 o1과 o2에 모두 "read" 접근권한을 부여한다. 만약 o1 〉 o2 이면 o1에는 "write", o2에는 "read" 접근권한을 부여하고, o1 〈 o2 이면 o1에는 "read", o2에는 "write" 접근권한을 부여한다. 직무 T에 포함된 객체 집합이 O={o1, o2, ..., on}일 경우, 같은 보안등급을 갖는 객체들을 확인한다. 그리고 보안등급이 하나이거나 둘이면 위의 규칙에 따라 접근권한을 부여하면 된다. 그러나 객체들이 Secret(S), Confidential(C)과 Unclassified(U)" 모두 포함하고 있을 때는 IM-ACM에서 제약사항으로 처리한다. 예를 들어 C급의 보안등급을 가진 객체에 접근할 때에 S급 객체와 비교하여 "read" 권한을 갖고 U급 객체와 비교하여 "write"가 갖는다. 이럴 경우에 C급 객체는 "write,

〔표 1〕 직무-객체간 접근권한 할당 알고리즘

```
if(Task's SG < object's SG) {
 prohibit(assign "read" operation to object);
 prohibit(assign "write" operation to object);
}
else if(Task's SG > object's SG) {
 permit(assign "read" operation to object);
 prohibit(assign "write" operation to object);
}
else if(Task's SG == object's SG {
 permit(assign "read or/and write" operation
 to object);
}
```

〔표 2〕 객체간 보안등급에 따른 접근권한 할당 알고리즘

```
loop(All object set) {
 o[]=object set related to t;
}
if(The number of o[] element == 1) {
 permit("read" or "write");
}
if((The number of o[] element == 2) &&
  (o[0] == o1) && (o[1] == o2)) {
  if(o1' SG == o2' SG) {
   permit((o1, o2), "read and write");
    }
    else if(o1' SG > o2's SG) {
         permit(o2, "read");
         permit(o1, "write");
    }
    else if(o1' SG < o2's SG) {
         permit(o1, "read");
         permit(o2, "write");
    }
}
else if((The number of o[] element == n) &&
      (o[0] == o1) && (o[1] == o2) ... &&
      (o[n-1] == on)) {
    check objects's SG;
    if((The number of objects's SG == 1) or
      (The number of objects's SG == 2)) {
       apply above algorithm;
      }
      else if(objects's SG == 3) {
       apply constraints;
      }
}
```

〔표 3〕 내부자 프로세스 프로파일 구성

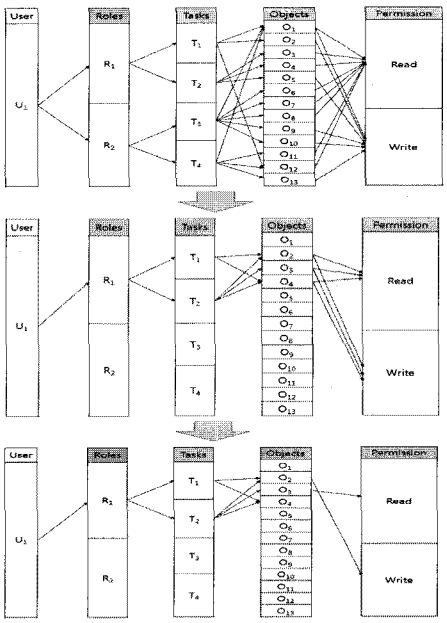
Users	Context			Roles	Tasks	Objects	Work Time(H)	Perm	Work Count
	Location	Time	Resource						
U_1	Office	0900 ~1800	PC	R_1(S)	T_1(C)	O_2(C)	6	W	60
						O_2(C)	2	R	80
						O_4(C)	3	R	20
						O_4(C)	1	W	10
					T_2(S)	O_2(C)	2	R	15
						O_3(S)	5	R	60
						O_3(S)	2	W	40
						O_4(C)	1	W	10
						O_4(C)	2	R	20

read" 권한을 다 갖기 때문에 정보의 흐름이 발생할 수 있다. 본 논문에서는 이런 경우에 제약사항으로 설정하여 "write, read"권한 중에 "read"" 권한만 갖도록 규정한다. 만약 "write" 권한을 수행해야 한다면 상위 보안등급의 객체에 대한 작업을 종료시킨다.

3.2.2 오용행위 차단

합법적인 권한을 가진 내부자는 직무를 수행하기 위해 필요한 역할을 활성화시켜 세션을 수립한다. 내부자가 직무를 위해 역할에 따른 권한을 승인받는다면 내부자가 객체에 접근하여 어떻게 사용하는지 통제할 수 있는 방법이 없다. IM-ACM은 내부자의 객체에 대한 작업 내용을 지속적으로 감시하여 실제 수행하는 프로세스와 최근 내부자의 작업 수행 프로세스 패턴과 비교하여 오용행위를 차단한다.

최근 내부자 프로세스 프로파일은 〔표 3〕과 같이 본 모델의 구성요소를 중심으로 이루어진다. 오용행위 차단 기법이 트랜잭션 즉, 사용자의 프로세스 진행에 따라 각 단계별로 차단하기 때문에 사용자의 상황, 역할, 직무, 객체 등으로 이루어진다. 일차적으로 CA-TRBAC에서 역할, 직무, 객체의 보안등급을 고려하여 접근 가능한 프로세스를 필터링하면 오용행위 모니터는 'Work Count(WC)'를 이용하여 행동 가능 프로세스를 추출한다. WC는 내부자가 최근 일정 기간에 프로세스를 수행했던 횟수를 의미한다. 이 횟수는 프로세스가 수행된 시간(Work Time)과 함께 행동 예상 프로세스 패턴을 추출하는데 사용된다. 프로세스 추출 우선순위는 *Work Rate = 〈Work Count X Work Time(x/24)〉*로 계산된다. 예를 들어 내부자(U_1)이 세션(S_1)을 통해 역할(R_1)을 활성화한 후 직무(T_1)에 관련된 객체(O_2,O_4)에 대한 행동 가능 프로세스 패턴을 추출한다면 Work Rate가 15(60×6/24)로 그 중에 높은 O_2에 쓰기 프로세스를 추천하게 된다. 직무(T_2)에 대한 행동 가능 프로세스



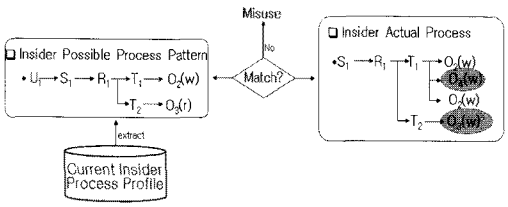
〔그림 5〕 내부자 가능 프로세스 패턴 추출 방법

는 O_3에 읽기 프로세스를 추천한다. 즉, WR값이 높은 것이 우선적으로 추출되게 된다.

IM-ACM은 내부자가 시스템에 로그인하여 작업을 시작할 때까지 다음 과정을 순차적으로 진행한다. 첫째, 내부자가 시스템에 로그인한 역할이 최근에 내부자가 수행했던 작업 프로세스에 포함되어 있는지 확인한다. 둘째, 역할에 따른 직무 및 관련된 객체에 대한 접근 프로세스가 최근 내부자 프로세스 프로파일과 일치하는지 확인한다. 셋째, 접근한 객체에 대한 작업이 최근 작업한 프로세스 프로파일 내의 접근권한과 일치하는지 확인한다. 마지막으로 일련의 프로세스가 다른 프로세스보다 Work Rate가 높은 지를 확인한다. 〔그림 5〕는 내부자 U_1이 역할 R_1으로 시스템에 접속하였을 경우, U_1의 가능 행동 프로세스 패턴을 WC 값에 의해 추출하는 과정을 보여준다.

〔그림 6〕은 내부자가 세션(S_1)을 통해 역할(R_1)을 활성화한 후 직무(T_1)에 관련된 객체(O_2,O_3,O_4)에 접근하여 작업을 수행하는 것을 오용행위 모니터를 통해 감시하여 오용행위를 차단한다.

내부자 실제 행동에서 객체 O_4에 대한 쓰기는 최근 내부자 프로세스 프로파일을 통해 추출한 수행 가능 프로세스 패턴과 다르다는 것을 알 수 있다. 또한, T_2 직무와 연관된 객체 O_3에 대해 쓰기 작업을 수행하는데 가능 프로세스 패턴에는 읽기 작업만 수행할 수 있게 되었다. 오용행위 모니터는 직무에 따라 합법적으



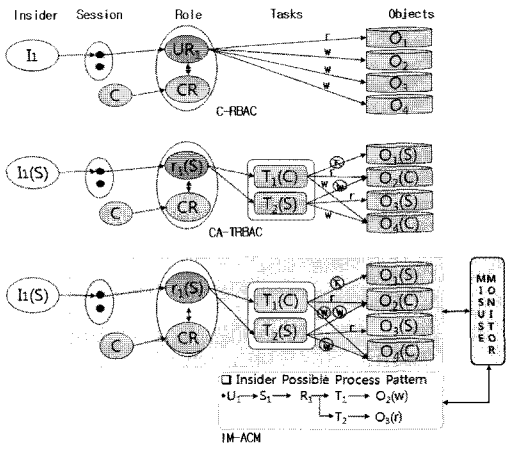
〔그림 6〕 내부자 오용행위 차단 방법

로 객체에 접근할 수 있더라도 최근 프로세스 프로파일에 있는 객체에 대한 작업 기록이 없는 경우나 다른 작업을 수행할 경우에 오용행위로 간주한다.

IV. IM-ACM의 비교 평가

IM-ACM은 기존의 접근제어 모델과는 내부자의 환경을 고려하여 보안속성에 따라 상위 등급의 객체에 대한 정보가 하위 등급의 객체로 이동하지 않도록 하고 오용행위 차단 기능을 추가하여 내부자의 실제 행동과 최근 내부자 행동 프로파일에서 예상 행동 패턴을 추출하여 비교함으로써 오용행위를 차단하는 것에 차별성을 두었다.

〔그림 7〕은 IM-ACM가 C-RBAC과 CA-TRBAC이 차단하지 못하는 내부자의 오용행위를 차단하는 것을 보여준다. CA-TRBAC에서 내부자는 T_1의 보안등급이 C등급이 때문에 상위보안 등급(S)인 O_1에 읽기 권한으로도 접근할 수 없으며, T_2 직무에서는 O_2, O_3, O_4에 각각 읽기, 쓰기, 읽기 권한으로 접근할 수 있으나, O_2에 쓰기 권한을 허용할 경우 상위 보안등급인 O_3의 정보가 유출될 수 있기 때문에 O_2의 쓰기 권한을 차단한다. IM-ACM에서는 T1에 따라 내부자가 O_1, O_2, O_4에 접근할 수 있으나 〔그림 5〕의 최근 내부자 프로세스 프로파일에 저장되어 있는 패턴으로는 O_2에 대한 작업만을 수행하고 있기 때문에 O_4에 대한



〔그림 7〕 접근제어 비교

쓰기 작업 행위는 차단된다. 또한, 최근 내부자의 T_2에 따라 O_3에 대해 읽기 작업을 지속적으로 수행하고 있는데 내부자는 O_4에 대하여 쓰기 작업을 수행하려고 하기 때문에 이 역시 차단된다.

IM-ACM의 최근 내부자 프로세스 프로파일은 내부자가 조직에서 장기간 근무할 경우 그 정확도가 우수하나, 신입사원일 경우에는 프로파일에 대한 정확도가 낮기 때문에 False Positive rate가 높다. 또한, 장기간 근무한 내부자라고 할지라도 새로운 프로젝트를 수행할 경우에는 행동 프로파일이 충분하지 않기 때문에 False Positive가 발생한다. 그래서 충분한 프로세스 프로파일을 확보한 내부자나 지속적으로 수행되고 있는 프로젝트일 경우에 오용행위 모니터를 적용하고 그 이외에는 CA-TRBAC으로만 접근통제 한다.

IM-ACM은 불법적인 정보의 흐름을 차단함으로써 정보의 기밀성을 유지할 수 있으며, CA-TRBAC에서 차단할 수 없는 내부자의 오용행위도 차단한다. 그러나 IM-ACM은 기존의 접근제어모델에 비해 활

〔표 4〕 비교평가 결과

기 준		C-RBAC	CA-TRBAC	IM-ACM
구성 요소	환 경	일반	유비쿼터스	유비쿼터스
	상황인식	적용	적용	적용
	보안등급	-	적용	적용
	접근제어 키	상황, 역할	직무 및 상황	직무 및 상황
	접근모드	불확실	설정(R/W)	설정(R/W)
	역할 상속	모든 권한	부분적 권한	부분적 권한
	접근제어 능력	수동, 능동적	수동, 능동적	수동, 능동적
	활성화 역할 수	2	3	3
정보 흐름		-	read up, write down 불가	read up, write down 불가
오용행위		-	-	대부분 가능(프로파일에 종속)

성화 역할 수가 많고 접근제어 개체 수가 많아 모델 구성측면에서 복잡한 단점을 갖고 있다. [표 4]는 C-RBAC, CA-TRBAC과 IM-ACM을 비교 평가한 결과이다.

V. 결 론

본 논문은 내부자에 의한 불법적인 정보 유출과 오용행위를 능동적으로 차단하기 위해 CA-TRBAC 모델에 오용행위 모니터 기능을 추가한 IM-ACM을 제시하였다. 내부자에 의한 정보 유출은 서로 다른 보안등급의 데이터를 동시에 작업할 때 빈번하게 발생한다. 그러나 기존의 접근제어 모델을 이용할 경우에는 정보의 흐름을 차단하거나 정보의 기밀성을 유지하기가 쉽지 않다. IM-ACM에서는 CA-TRBAC에서 접근제어 개체의 보안속성을 이용하여 정보 유출이 가능케 하는 정보의 흐름을 차단하는 기능으로 정보의 기밀성 훼손을 막는다. 또한, 오용행위 모니터 기능을 추가하여 내부자가 자신의 합법적인 권한으로 객체에 대한 오용행위를 차단한다. 최근 내부자 작업 프로세스 프로파일에서 가장 수행 가능성이 높은 작업 프로세스를 추출하여 내부자의 실제 진행 프로세스와 비교함으로써 오용행위를 차단한다. 그러나 내부자의 업무 경력이 적거나 새로운 프로젝트를 수행하게 될 경우에는 내부자의 작업 프로세스 패턴을 정확하게 프로파일링할 수 없는 단점이 있다.

향후 오용행위 차단율과 정확도를 향상시킬 수 있도록 내부자 작업 프로세스 패턴에 대한 프로파일 방법을 개선할 것이다.

참 고 문 헌

[1] "2010 Cyber Security Watch Survey", CSO magazine, U.S. Secret Service and Carnegie Mellon University&Deloitte, 2009.

[2] Brian M. Bowen, Malek Ben Salem, and Shlomo Hershkop, "Designing Host and Network Sensors to Mitigate the Insider Threat", IEEE The Journal of Security & Privacy, Vol. 7 no. 6, pp. 22-29, Dec. 2009.

[3] Felicia A. Duran, Stephen H. Conrad, Gregory N. Conrad, David P. Duggan, and Edward Bruce Held, "Building a System for Insider Security", IEEE The Journal of Security & Privacy, Vol. 7 no. 6, pp. 30-38, Dec. 2009.

[4] Salvatore J. Stolfo, Steven M. Bellovin, Angelos D. Keromytis, Sara Sinclair, Sean W. Smith, and Shlomo Hershkop, Insider Attack and Cyber Security Beyond the Hacker, Springer, 2008.

[5] Frank Stajano, Security for ubiquitous computing, Wiley, 2002.

[6] David F. Ferraiolo and D. Richard Kuhn, Ramaswamy Chandramouli, Role-Based Access Control, Artech House, 2003.

[7] Antonio Corradi, Rebecca Montanari, and Daniela Tibaldi, "Context-based Access Control for Ubiquitous Service Provisioning", Proceedings of the COMPSAC'04, pp. 444-451, Sep. 2004.

[8] Weili Han, Junjing Zhang, and Xiaobo Yao, "Context-sensitive Access control Model and Implementation", Proceedings of The CIT'05, pp. 757-763, Sep. 2005.

[9] Dawn Cappelli, Andrew Moore, Randall Trzeciak, and Timothy J. Shimeall, Common Sense Guide to Prevention and Detection of insider Threats, SEI Carnegie Mellon, Jan. 2009.

[10] Role Based Access Control, American National Standards Institute, Feb. 2004.

[11] Seon-Ho Park, Young-Ju Han, and Tai-Myoung Chung, "Context-Role Based Access Control for Context-Aware Application", High Performance Computing and Communications 2006, LNCS 4208, pp. 572-580, 2006.

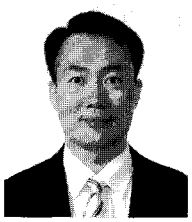
[12] 엄정호, "유비쿼터스 전장 컴퓨팅 환경에서 상황인식과 직무 역할 기반의 접근제어에 관한 연구", 박사학위논문, 성균관대학교, 2008년 2월.

[13] Joon S. Park and Shuyuan Mary Ho, "Composite Role-Based Monitoring For Countering Insider Threats", The 2nd Symposium on Intelligence and Security Informatics 2004, pp. 201-213, 2004.

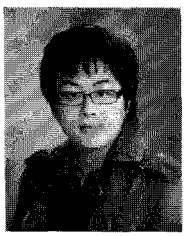
[14] 엄정호, 박선호, 정태명, "NCW 컴퓨팅 환경에서

CA-TRBAC의 접근제어 효율성에 관한 연구", 정보보안논문지, 9(1), pp. 43-53, 2009년 3월.
[15] Robert H. Anderson, Research and Development Initiatives Focused on Preventing, Detecting, and Responding to Insider Misuse of Critical Defense Information Systems, RAND, Aug. 1999.
[16] 임희섭, "군사환경에 과업-역할기반 접근제어 모델을 적용하기 위한 제약조건", 석사학위논문, 서강대학교, 2002년.
[17] Sejong Oh and Seog Park, "Task-Role-Based Access Control Model", Information Systems, Vol. 28, Issue 6, pp. 533-562, Sep. 2003.

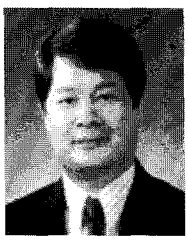
〈著者紹介〉



엄 정 호 (Jung-ho Eom) 정회원
1994년: 공군사관학교 항공공학과(학사)
2003년: 성균관대학교 컴퓨터공학과(석사)
2008년: 성균관대학교 컴퓨터공학과(박사)
1994년~2010년 8월: 대한민국 공군장교
2010년 9월~: 성균관대학교 BK21 연구교수
〈관심분야〉 네트워크 보안, 사이버공격 모델, 접근제어



박 선 호 (Seon-ho Park) 학생회원
2005년: 정보통신공학부 학사, 성균관대학교
2007년: 컴퓨터공학 석사, 성균관대학교
2007년~: 현재 성균관대학교 전자전기컴퓨터공학과 박사과정
〈관심분야〉 유비쿼터스 컴퓨팅, 시스템 보안, 네트워크 보안, 접근제어 모델 및 검증 방법론



정 태 명 (Tai M. Chung) 종신회원
1981년: 연세대학교 전기공학과 졸업(학사)
1984년: University of Illinois Chicago IL, U. S. A. 전자계산학과 졸업(학사)
1987년: University of Illinois Chicago IL, U. S. A. 컴퓨터공학과 졸업(석사)
1995년: Purdue University W. Lafayette, IN, U. S. A. 컴퓨터공학 졸업(박사)
1995년~: 성균관대학교 컴퓨터공학과 정교수
〈관심분야〉 통합보안관리, 네트워크, 무선망