안전한 운영체제를 위한
객체확장 역할기반 접근제어 모델

강현수[O] 허 신
한양대학교 컴퓨터공학과
{hskang[O], shinheu}@cse.hanyang.ac.kr

Object-extended Role-Based Access Control model for Secure Operating Systems

Hyunsu Kang[0] Shin Heu
Dept. of Computer Science and Engineering. Hanyang University

요 약

역할기반 접근제어모델은 기업이나 정부 등의 조직체계를 반영하는데 적합한 접근제어 모델로 컴퓨터의 자원을 접근하는 데에도 유용하게 사용될 수 있으며, 임의접근제어나 강제접근제어 방식의 단점을 대안할 수 있는 방법이기도 하다. 본 논문에서는 기존의 역할기반 접근제어 방식에서 사용자에게만 할당하던 역할을 객체에게도 할당하여 객체의 관리와 보안성을 향상시켰다. 객체의 특성에 따라 사용자들이 접근할 수 있는 권한이 비슷한 경우가 많기 때문에, 이를 기반으로 하여 객체를 역할의 단위로 관리할 경우 보다 효율적인 객체 관리를 할 수 있고, 동일한 보안 등급을 갖는 객체들이 하나의 역할로 그룹지어지기 때문에 보안 관리 측면에서도 효율적이다 할 수 있다.

1. 서 론

안전한 운영체제를 위하여 현재까지의 대부분의 운영체제들은 임의접근제어(DAC: Discretionary Access Control) 정책에 강제접근제어(MAC: Mandatory Access Control) 정책을 추가하여 운영체제의 보안 능력을 향상시키는 경향이 많다. 임의접근제어 정책은 사용자들에게 유연한 보안 설정을 제공하나 그만큼 보안에 취약하고, 강제접근제어 정책은 관리자에 의해 모든 것이 제어되는 중앙 집중 방식으로 강력한 보안성을 제공하나 지나치게 엄격한 통제로 인하여 그만큼 유연성이 떨어진다는 단점을 가진다. 역할기반 접근제어(RBAC: Role-Based Access Control)는 기업이나 정부 등의 조직체계를 반영하는데 적합한 접근제어 모델로 컴퓨터의 자원을 접근하는 데에도 유용하게 사용될 수 있으며, 임의접근제어나 강제접근제어 방식의 단점을 대안할 수 있는 방법이기도 하다.

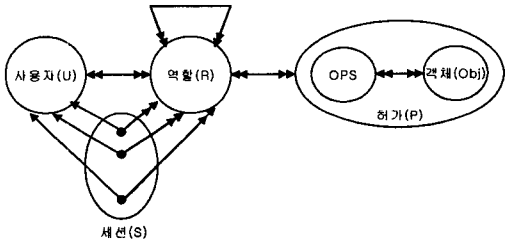
본 논문에서는 기존의 역할기반접근제어 정책을 살펴보고, 기존 정책에서 사용자를 역할로 구분하여 처리하던 방식을 객체에도 적용하여 객체의 효율적인 관리와 향상된 보안 방식을 제공한다.

이를 위한 본 논문의 구성은 다음과 같다. 2장에서는 기존의 역할기반 접근제어 모델에 대해 설명하며, 3장에서는 객체에 역할을 추가한 제안된 모델을, 마지막으로 4장에서 결론을 맺는다.

2. 역할기반 접근제어 모델

역할기반접근제어 정책은 사용자의 역할에 기반을 둔 접근통제방식이다. 관리자가 회사나 조직에서 수행해야 할 관련된 기능들을 역할(role)로 구분 짓고, 그 역할에 따라 수행할 수 있는 기능이 제한된다. 따라서 사용자는 각자의 역량과 임무에 맞는 역할에 할당되어 작업을 수행할 수 있다[1][2][3][4].

<그림 1>은 역할기반접근제어 정책을 나타내고 있다.



<그림 1> 역할기반 접근제어 모델

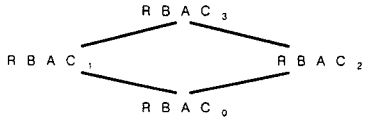
여기서 2중 화살표는 다대다(many-to-many) 방식으로 할당 가능함을 의미한다. 예를 들어 사용자 한 명이 여러 역할에 속할 수도 있고, 하나의 역할에 여러 명의 사용자가 배정될 수도 있다.

역할에 할당된 사용자는 세션(session)을 활성화시킴으로 역할이 가질 수 있는 접근제어 조건에 맞추어 객체

에 접근할 수 있다. 세션이란 사용자가 할당되어 있는 역할의 집합 중에서 활성화된 역할들과 사용자와의 맵핑을 의미한다.

역할기반 접근제어 모델은 4가지 형태로 나누어진다.

가장 기본이 되는 $RBAC_0$ 모델은 사용자와 역할, 객체, 허가 등의 최소한의 요구 사항만을 가지고 정의된다. $RBAC_0$에 사용자 역할의 상속성(role hierarchies)을 추가한 $RBAC_1$ 모델이 있으며, $RBAC_0$ 모델에 역할의 상호배제(mutual exclusion), 역할 배정 인원 수(cardinality) 등의 여러 가지 제약점(constraints)을 추가한 $RBAC_2$ 모델이 있다. $RBAC_3$ 모델은 $RBAC_0$ 및 $RBAC_1$과 $RBAC_2$를 통합하는 모델이다. <그림 2>에서 서로의 관계를 나타내고 있다.



<그림 2> 역할기반 접근제어 모델간의 관계

3. 제안된 객체확장 역할기반 접근제어 모델

기존의 역할기반 접근제어 방식은 사용자나 주체만을 역할에 할당함으로 사용자 개개인에 대한 관리는 간소화 시켰지만, 객체에 대해서는 그렇지 못하다는 단점이 있다. 객체의 수가 많을수록 역할과 허가의 관계가 복잡해지며 이 정보를 저장하기 위한 오버헤드가 커지게 된다.

또한 대부분의 운영체제의 경우 객체에 대한 접근 방식이 일정하게 지정되어져 있다. 예를 들어 로그 파일이나 설정 파일과 같은 시스템 관리 목적의 많은 파일들은 일반 사용자들은 읽기만 가능하거나 어떤 접근도 불가능한 경우가 많다. 반면에 관리자의 경우는 읽기뿐만 아니라 쓰기 접근도 가능하게 되어 있다. 홈페이지 문서의 경우는 운영체제에 계정을 가진 허가된 사용자뿐만 아니라 외부의 어느 누구에게도 접근 권한을 주는 경우가 많다.

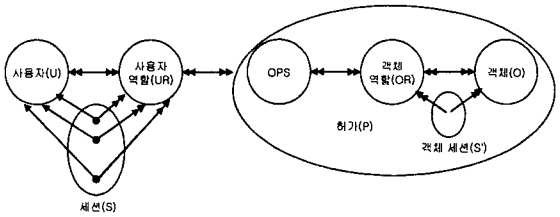
따라서 본 논문에서는 동일한 기능이나 보안 가치를 지닌 객체에도 역할 개념을 도입하여 보다 효율적으로 역할 관리를 수행할 수 있게 한다.

기존 연구 중 [5]와 [6]과 같은 연구에서도 객체에 역할을 부여하는 방식을 적용하고 있다. [5]에서는 객체를 음악 파일, 그림 파일, 실행 파일 등의 객체의 형식에 맞추어 주로 할당하고 있으며, [6]에서는 사용자에게 할당될 수 있는 역할을 객체에게도 할당하며 여기에 역할에 대한 보안 등급을 고려하여 확장 시키고 있다.

3.1 객체확장 역할기반 접근제어 모델

<그림 3>은 역할기반 접근제어 모델 중 가장 기본적인 모델인 $RBAC_0$에 객체에 역할을 부여한 제안된 모델을 도식화하고 있다.

객체의 역할은 객체의 종류나 사용 빈도, 크기 등 객체의 특성에 따라 다양하게 부여될 수 있지만, 본 논문에서는 보안과 관련된 특성에 따라 역할을 분류한다. 즉 로그 파일, 사용자나 역할 설정 파일, 시스템 설정 파일, 네트워크 설정 파일, 연결된 하드웨어 등과 같은 형태로 역할을 분류한다.



<그림 3> 제안된 객체확장 역할기반 접근제어 모델

기존의 사용자에게만 할당되던 역할(R)을 사용자 측면의 역할(UR)과 객체 측면의 역할(OR)로 구분하여, 사용자 역할에 속해 있는 사용자들은 객체 역할에 속해 있는 객체들을 허가(P)에 따라 사용할 수 있다.

이 때 사용자와 사용자 역할 사이의 세션(S)의 개념처럼 객체 역할에 속한 객체가 활성화 될 때 객체 세션(S')이 생성된다. 사용자 세션의 경우는 한 명의 사용자에 대해 여러 개의 역할 맵핑이 가능하지만, 역할 세션의 경우는 객체 역할에 속한 여러 개의 객체가 동시에 활성화가 가능하다.

이를 [14]에 기반하여 다음과 같이 정형화된 표기법으로 나타낼 수 있다.

- $UA \subseteq U \times UR$
- $assigned\_users: (ur: UR) \rightarrow 2^U$
  $assigned\_users(ur) = \{u \in U \mid (u, ur) \in UA\}$
- $OA \subseteq O \times OR$
- $assigned\_objs: (or: OR) \rightarrow 2^O$
  $assigned\_objs(or) = \{o \in O \mid (o, or) \in OA\}$
- $P \subseteq 2^{(OPS \times OR \times O)}$
- $PA \subseteq P \times UR$
- $assigned\_permissions: (ur: UR) \rightarrow 2^P$
  $assigned\_permissions(ur) = \{p \in P \mid (p, ur) \in PA\}$
- $Op(p: P) \rightarrow \{op \subseteq OPS\}$
- $Or(p: P) \rightarrow \{or \subseteq OR\}$
- $Obj(p: P) \rightarrow \{obj \subseteq O\}$
- $session\_users(s: S) \rightarrow U$

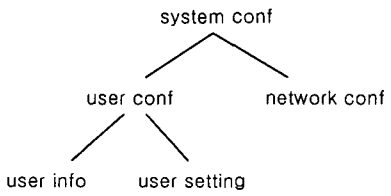
- $session\_roles(s: S) \rightarrow 2^{UR}$
  $session\_roles(s) \subseteq$
  $\{ ur \in UR \mid (session\_users(s), ur) \in UA \}$
- $session'\_objs(s': S') \rightarrow 2^{O}$
- $session'\_roles(s': S') \rightarrow 2^{OR}$
  $session'\_roles(s') \subseteq$
  $\{ or \in OR \mid (session'\_objs(s'), or) \in OA \}$

각각의 할당된 정보와 함수들을 통하여 접근 제어 정보를 확인할 수 있다.

3.2 객체 역할의 상속

$RBAC_1$에서의 사용자 역할의 상속처럼 객체의 역할 간에도 상속의 개념이 존재한다.

<그림 4>에서는 시스템 설정을 위해 필요한 몇 가지 객체를 간략히 역할로 구성한 예이다.



<그림 4> 객체 역할의 상속 예제

가장 포괄적인 형태의 역할이 상속 그래프의 위쪽에 위치하고 있다.

user conf 역할은 user info와 user setting 역할을 상속받아 이 두 역할이 가질 수 있는 속성을 모두 가질 수 있다.

일반 사용자는 user info 역할에 해당되는 객체에 접근하여 사용자들의 정보를 읽을 수는 있지만, user setting이나 이를 상속하는 user conf 등의 역할에 속하는 객체에는 접근을 할 수가 없다. 예를 들어 유닉스 시스템의 경우 user info 역할에 속하는 /etc/passwd 파일의 내용은 일반 사용자들도 읽을 수 있으나 쓰기나 수정 등은 금지되어 있다. 하지만 사용자가 시스템 관리자의 역할로 user info를 상속받는 user conf 역할로 접근하였을 때 /etc/passwd 파일에 쓰기가 가능해진다.

4. 결론 및 향후 과제

본 논문에서는 기존의 역할기반 접근제어 방식에서 사용자에게만 할당하던 역할을 객체에게도 할당하여 객체의 관리와 보안성을 향상시켰다. 객체의 특성에 따라 사용자들이 접근할 수 있는 권한이 비슷한 경우가 많기 때문에, 이를 기반으로 하여 객체를 역할의 단위로 관리할 경우 보다 효율적인 객체 관리를 할 수 있고, 동일한 보안 등급을 갖는 객체들이 하나의 역할로 그룹지어지기 때문에 보안 관리 측면에서도 효율적이다 할 수 있다.

향후 본 논문에서 제안한 객체확장 역할기반 접근제어 모델을 실제 시스템에 구현하여 적용함으로 향상된 성능을 입증할 필요가 있다. 또한 객체 역할의 상속과 역할의 상호배제 등에 대하여 보다 심도 있는 연구가 필요하다.

5. 참고 문헌

[1] D. Ferraiolo and R. Kuhn, "Role-Based Access Control," *Proc. of 15th National Computer Security Conference*, pp.554-563, 1992.

[2] Ravi S. Sandhu and Edward J. Coyne, "Role-Based Access Control Models," *Computer*, Vol.29, Issue 2, pp.38-47, Feb. 1996.

[3] D. Ferraiolo, J. Cugini, and D. R. Kuhn, "Role-Based Access Control(RBAC): Features and Motivations," *Proc. of 11th Annual Computer Security Application Conference*, pp.241-248, Dec. 1995.

[4] NIST, *RBAC Standard*, April, 2003.

[5] Matthew J. Moyer and Mustaque Ahamad, "Generalized Role-Based Access Control," *21st Information Conference on Distributed Computing Systems*, pp.391-398, 2001.

[6] 김학범, 홍기용, 김동규, "다단계 보안통제가 가능한 확장된 역할기반 접근통제 모델", 한국정보처리학회 논문지, 제7권 제6호, pp.1886-1902, 2000.