제19회 한국정보처리학회 춘계학술발표대회 논문집 제10권 제1호 (2003. 5)

RBAC 정책을 적용한 마이크로 커널 기반의 안전한 리눅스 커널 분석 및 설계

최병선*, 이성현*, 이원구*, 이희규*,이재광*
*한남대학교 컴퓨터공학과
e-mail:bschoi@netwk.hannam.ac.kr

Analysis and Design Secure Linux on Micro Kernel Applied RBAC Policy

Byoung-Son Choi*, Seoung-Hyeon Lee*, Won-Gu Lee*,
Hea-Gyu Lee*, Jae-Kwang Lee*
*Dept of Computer Engineering, Hannam University

요 약

본 논문에서는 접근 제어 메커니즘을 적용한 마이크로 커널 기반의 안전한 리눅스 커널에 대해 분석 및 설계하였다. 설계된 접근제어 모델은 역할기반 접근제어를 이용하여 권한을 효과적으로 통제하고, 신분 및 규칙기반 접근제어를 이용하여 정보 및 시스템의 비밀성, 무결성, 가용성의 보장 및 시스템의 불법적인 접근을 방지할 수 있다. 리눅스 마이크로 커널 기반 접근제어 모델을 직무, 보안등급, 무결성 등급 및 소유권의 다단계 보안 정책을 기반으로 시스템의 불법적인 접근, 직무기반, 소유권 등의 다단계 보안 정책을 기반으로 하여 시스템의 불법적인 접근을 통제 할 수 있다.

1. 서론

안전한 운영체제(Secure Operating System)란 컴퓨터 운영체제상에 내재된 보안상의 결함으로 인하여 발생 가능한 각종 해킹으로부터 시스템을 보호하기 위하여 기존의 운영체제 내에 보안기능을 통합시킨 보안커널(Security Kernel)을 추가로 이식한 운영체제이다. 보안커널이 이식된 운영체제는 컴퓨터 사용자에 대한 식별 및 인증, 강제적 접근통제, 임의적 접근통제, 재사용 방지, 침입 탐지 등의 보안 기능 요소를 갖추어야 한다.

최근, 보안 커널 연구 개발에 대해 접근 방법은 마이크로 커널 기술을 적용하여 보안 커널을 설계 구현하고 있다[1]. 인터넷은 컴퓨터 네트워크를 통한 해킹 수법이 갈수록 지능, 고도화, 국제화되고 있음에도 불구하고 국가적으로 중요한 비밀 정보를 보안대책 없이 컴퓨터 및 네트워크 시스템을 통해 외부로 유출된다면 위험한 일이 아닐 수 없다. 따라서 국내에서도 정보보호 확립 차원에서 보안 커널을 국산화하고, 안전한 OS개발은 필수적이라 할 것이다. 본 논문에서는 RBAC, DAC, MAC 등의 보안 정책과 보안모델 및 리눅스 OS의 보안 요구 사항이 될 수 있는 마이크로 커널에 관한 내용을 소개하고, RBAC 메커니즘을 적용한 마이크로 커널 기반의 안전한 리눅스 커널을 분석 및 설계한다.

2. 관련연구

2.1 보안 정책(Security Policy)

2.1.1 DAC(Discretionary Access Control) 정책

DAC 정책은 사용자 계정(ID)와 각 파일에 대해 시스템에서 허용된 읽기, 쓰기, 실행 등의 접근 모드를 나타내는 인증을 기반으로 하는 정책을 채택하고 있다. 이 정책에서는 사용자가 어떤 객체에 대한 접근을 요청할 경우 명시되어진 인증을 검사하여 그 사용자가 그 객체를 특정 모드로 접근할 수 있는 경우에는 접근을 허용하며 그 외에는 거부된다. 이러한 정책은 많은 상업적인 운영체제 시스템들과 응용 프로그램들에 융통성 있는 적용이 가능하다는 장점을 가지고 있다. 반면에 하나의 데이터에 접근이 가능한 사용자가 자격이 없는 사용자에게 그 데이터를 전달할 수 있게 되므로 시스템 안전에서 정보의 흐름에 대한 실제적인 보안이 보증되지 못하는 단점을 갖는다[2].

2.1.2 MAC(Mandatory Access Control) 정책

MAC 정책은 DAC 정책에 비해 보다 견고한 접근제어 방법을 제공한다. 이 정책은 각 사용자와 객체에 보안 수준을 할당하여 시스템내의 주체(subject)와 객체(object)를 계층화(Classification)하는 것을 접근제어의 기본 방법으로 삼고 있다. MAC 정책은 군대나 정부 등 보다 강력한 접근제어가 필요한 환경에서 사용되어 왔으며 TS(Top Secret), S(Secret), C(Confidential), U(Unclassified) 등의 보안 수준을 계층적으로 정의하고 있다. 앞서 설명한 것과 같이 MAC은 DAC에 비해 보다 강력한 접근제어를 할 수 있으나 그에반해 대부분의 상업적인 사업들의 요건을 만족시킬 수 있는 유동성이 부족하여 널리 사용되지 않는다[3].

2.1.3 RBAC(Role-Based Access Control) 정책

RBAC은 시스템 안에서 사용자가 현재 실행하고 있는 행동을 기반으로 정보에 대한 사용자의 접근을 제어한다. RBAC에서는 특정 행동에 부여되는 행위들과 책임(responsibility)들의 집합으로써 역할을 정의하고 있으며 사용자들은 자신이 속해있는 역할에 의해서 정보에 대한 접근이 제어된다. RBAC 정책의 특징은 사용자와 정보를 직접적으로 연결하지 않고 사용자들 역할에 연결하고, 역할과 정보객체 사이에 접근 권한을 연결하여 사용자 권한을 논리적으로 나누게 된다. 이러한 논리적 분할은 보안의 관리 문제를 간단하게 해결할 수 있다. RBAC에서는 접근제어가 적용되는 환경에 따라 많은 역할들이 정의된다. 역할의 분할은 적용되는 환경의 계층구조 또는 실행 환경에 의해 구분되어 진다[4].

2.2 보안 모델(Secure OS Model)

2.2.1 레티스(Lattice) 메커니즘

접근제어 매트릭스와 같이 주체와 객체에 직접 접근 권한을 부여하는 대신에, 주체와 객체가 지닌 속성(예:보안등급(Security Level), 범주(Category))를 접근제어에 사용한다. 레티스는 보안 클래스들을 보안의 중요도에 따라 비교우위를 가려서, 이를 선형으로 배열시킨 구조이다. 레티스는 수학적으로 정의하면, 부분 순위 연산자(partial ordering)인 <=을 지닌 집합 L로서, 집합 L의 임의의 두 원소인 a, b가 존재할 때, 최소상한선과 최대 하한선이 집합 L에 포함되어야 한다. 이때, 최소상한선과 최대 하한선은 각각 유일한 값이어야만 하며, 이러한 유일한 값을 이용하여 접근 권한을 결정하는 메커니즘이다.

2.2.2 BLP(Bell-Lapadula) 메커니즘

BLP 모델은 데이터의 접근제어를 통해서 시스템의 비밀성(confidentiality)을 보호하기 위한 모델이다. BLP 모델을 접근제어 매트릭스와 보안수준을 통해서 주체가 객체에 접근하는 것을 통제함으로써 시스템의 비밀성을 보호한다[5].

2.3 리눅스 커널(Linux Kernel)

2.3.1 통합 커널(Monolithic Kernel)

현재 일반적인 리눅스 시스템은 대부분의 시스템 기능들이 단일 주소공간의 커널에 밀집되어 있는 통합(monolithic)커널 형태이다. 보안 통합커널은 통합 커널 안에 보안정책 및 접근통제 메커니즘을 하나의 커널에 구현한 상태이다. 속도가 빠른 장점이 있는 반면, 이러한 형태는 정책 및 메커니즘의 변경이 있을 때 복잡한 구조로 인하여 번거로운 작업이 된다.

2.3.2 마이크로 커널(Micro Kernel)

마이크로 커널은 커널의 핵심적인 부분 이외의 부분은 사용자 영역에서 수행되는 서버가 시스템 기능성을 담당하도록 구현된다. 마이크로 커널에 포함된 기능은 태스크와 쓰레드 관리의 하위 부분, IPC(Interprocess Communication)와 동기화, 메모리 관리의 하위 부분, 최소 디바이스 관리, 시스템 운영중 발생하는 각종 인터럽트 처리 등이다. 파일 시스템, 네트워크 프로토콜, 표준 유닉스 인터페이스 등과 같은 운영체제 서비스의 상위 부분은 사용자 수준의 서버 프로그램으로 구현된다[6]. 각 서버는 IPC를 통해서 상호 동작함으로써 기존의 운영체제 상에서 서비스를 실현한다. 여기서 마이크로 커널은 애플리케이션 또는 시스템 서버와 하드웨어 사이에 메시지를 검증하고 전달하는 역할을 수행한다. 메시지 전달 방식의 통신 기법은 기존 커널 구조에 비해 효율적인 분산 컴퓨팅 환경을 지원한다는 장점을 가진다. 그리고 각 하드웨어에 하나의 마이크로 커널을 구성하게 되면, 다양한 유닉스 서버가 동작 할 수 있으므로 우수한 확장성과 높은 이식성을 제공한다. 대표적인 마이크로 커널에는 2세대 마이크로 커널이라 불리는 미국 MIT에서 개발한 Exokernel과 독일 GMD에서 개발한 L4가 있다[7].

3. 마이크로 커널 기반 안전한 운영체제 설계

3.1 정책 모듈 설계

3.1.1 보안 모델 설계: BLP

아래의 [그림 1]에서 알 수 있듯이, 본 논문에서 제안한 BLP 메커니즘을 적용하여 보다 안전한 시스템에서 정보 흐름의 허용 가능한 경로를 보여주고 있다.

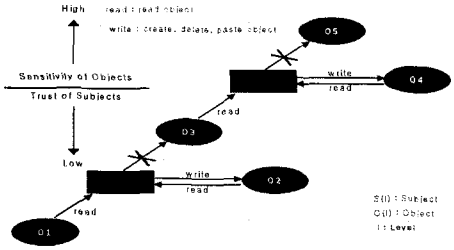
즉 이러한 수정된 메커니즘은 서로 다른 보안 등급을 가지고 있는 데이터를 다루는 시스템에서 불법적인 정보 유출을 막기 위해 필요한 보안 요구 조건에 대해 정의하는데 기본적인 성질은 다음과 같다.

① 주체 S(i)는 객체 O(i)를 오로지 C(S) >=

C(O)일 경우에만 읽을 수 있다.

② 주체 S는 객체 O를 오로지 C(S) <= C(O)일 경우에만 쓸 수 있다.

원래의 BLP 모델은 ②에서와 같이 C(S) <= C(O)일 경우에도 Write 연산을 허용하였다. 그러나, [그림 4]에서 보는바와 같이 Write 연산이란 생성(Create), 붙여쓰기(append), 삭제(Delete) 등의 포괄적인 연산이다. 그러나 낮은 등급의 주체가 더 높은 등급의 객체를 붙여쓰기, 삭제가 가능하다는 것은 현실적으로 보안상 문제가 발생되므로, 쓰기 연산은 제한되어야 한다.



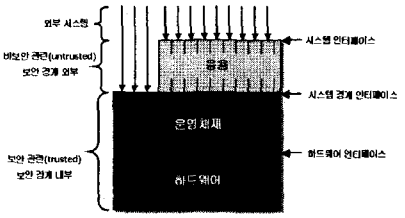
[그림 1] BLP 접근 모델

3.1.2 보안 모델 설계: RBAC 모델

RBAC 시스템은 리눅스 운영체제가 탑재된 PC를 기반으로 RBAC 기법을 인트라넷상에서 이용할 수 있도록 설계한 보안 시스템으로 관리능력을 가지고 있어 권한이 부여된 데이터 관리를 폭넓게 수행할 수 있다. 이는 RBAC 시스템에 관리도구가 있어 관계정보를 일관성 있게 유지하여 주기 때문이다. 이를 위해서는 많은 집합과 함수들이 필요하며, 이러한 집합과 함수를 이용하여 보안 운영체제에 적용할 수 있는 RBAC 모듈은 RBAC96 모델[8]에 기초하여 설계하였다. 보안 운영체제의 여러 구성요소 중에서 관리자가 RBAC을 관리하는데 있어서 가장 중요한 구성요소가 관리도구이며, 역할 간의 관계에 있어 사용자-역할, 역할-역할 관계를 데이터베이스에 저장하고 관리한다. 따라서 이들 관계에 대한 데이터베이스 정보가 일관성 있게 유지되어 있다.

3.2 마이크로 커널 기반 보안 운영체제 설계

전통적인 컴퓨터 시스템의 구조는 [그림 2]와 같이 하드웨어, 운영체제 및 응용프로그램으로 구성된다. 그림에서 각각의 계층은 아래 계층에 있는 facility를 사용한다. 운영체제와 하드웨어는 보안관련으로 보안경계(Security Perimeter) 내부에 위치한다. 응용프로그램은 잘 정의된 시스템 콜을 사용하여 보안경계를 통하여 운영체제에 접근한다. 사용자들은 시스템 외부에 있으며, 운영체제와 직접 통신하거나 응용프로그램을 통하여 시스템에 접근한다.

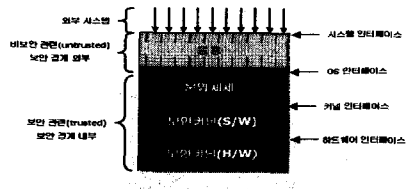


[그림 2] 일반 컴퓨터 시스템 구조

시스템에 대한 보안은 기본적으로 구조를 변경하지 않고 여러 가지 방법으로 개선될 수 있다. 하지만 아주 민감한 정보를 보호하고자 한다면, 강력한 개발 전략과 특별한 시스템 구조가 요구된다. 보안커널 방법은 일반 운영체제에 내재되어 있는 보안 문제점을 해결하기 위하여 운영체제를 설계하는 방법이다. 안전한 운영체제는 커널 수준에서 구현되어야 한다. 커널은 하드웨어와의 인터페이스를 제공하는 운영체제의 핵심이다. 커널에서 제공하는 인터페이스만이 사용자가 메모리와 파일 객체 등을 접근 할 수 있게 한다. 따라서 커널 수준에서 보안 정책을 수행해야 모든 접근이 제어 가능하다. 보안커널은 일반적으로 운영체제와 유사하며, 전통적인 운영체제 설계 개념을 사용한다. 보안커널에 요구되는 하드웨어도 거의 유사이다. 보안커널은 보안경계 내의 모든 주체와 객체를 통제하여야 하며 프로세스, 파일시스템, 메모리 관리, I/O를 위한 자원을 제공하여야 한다.

그리고, 다음과 같은 사항들을 고려하여 커널기반으로 안전한 운영체제를 설계하여야 한다.

- 보안 커널에서의 보안 함수 분리 : 보호 객체에 대한 모든 접근이 보안커널을 통과하도록 해야 하고, 운영체제나 사용자 보호용이 등의 여러 요인으로 보안커널에서 보안 함수를 분리해야 한다.
- 참견의 회피 : 보안 커널은 악의적 혹은 우연한 참견으로 프로세스 수행이 영향받는 것을 막아야 한다.
- 우회경로 회피 : 보안 정책을 강제적으로 적용하여 우회하는 경로가 없어야 하며, 주체의 요청으로 발생하는 보안커널 우회 회피를 관리하도록 설계되어야 한다.
- 보증 제공 : 시스템이 안전하다는 증거를 제공해야 한다.
- 하드웨어 메커니즘 : 메모리를 보호하며, 안전한 I/O 오퍼레이션을 제공해야 한다.
- 복잡도 최소화 : 보안커널을 작고 가볍게 구성해야 한다.



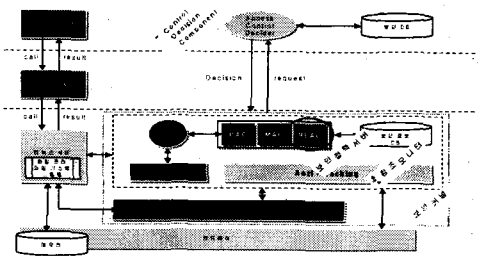
[그림 3] 컴퓨터 시스템의 보안커널

3.2.1 보안 운영체제 설계

실제 마이크로 커널을 설계 시 객체 지향 기법을 적용하여 안전한 운영체제의 기능을 제공하고자 할 때, 마이크로 커널의 정책이나 자료구조를 동적으로 변화시킬 수 있도록 하고 추상을 지원하기 위해 커널의 구조를 인터럽트 처리, 가상 메모리, 프로세스 관리 등으로 나뉘어 시스템 서비스를 응용 프로그래머에게 제공하는 것에 그 목표가 있다. 구현 시 모노리틱(monolithic) 서버를 분석하여 하드웨어 의존적 부분과 독립적 부분을 분리한 후, 하드웨어 의존적인 부분을 마이크로 커널이 대체하여 하드웨어 추상화를 지원할 수 있도록 하고, 나머지 하드웨어 독립적인 부분은 그대로 마이크로 커널 위에서 동작 가능하도록 구성한다. 즉, 기존의 보안 서버가 동작하는 운영체제 환경을 구축함으로써, 안전한 보안 운영체제를 구축한다.

3.2.2 마이크로 커널 기반 보안 서버 설계

본 논문에서 제안한 보안서버는 [그림 4]의 구조도에서 간접적으로 알 수 있듯이, MLS(Multi-level Security), 신분 기반 접근 통제(Identity-based Access Control), 동적 역할 기반 접근 통제(Dynamic Role-based Access Control)와 같은 세 가지 세부 정책의 조합으로 보안 정책을 수행한다. 보안서버에 의해 제공되는 접근 결정은 이 세 가지의 세부 정책을 만족하며, MLS의 구현과 동적인 역할 기반 접근 통제(Dynamic Role-based Access Control)의 지원, 신분 기반 접근 통제(Identity-based Access Control)의 보강 및 지원한다는 측면에서 기존 보안 서버와는 다르다.



[그림 4] 마이크로 커널 기반의 보안 운영체제 구조도

4. 결론 및 향후 연구

인터넷은 컴퓨터 네트워크를 통한 해킹 수법이 갈수록 지능, 고도화, 국제화되고 있음에도 불구하고 국가적으로 중요한 비밀 정보를 보안 대책 없이 컴퓨터 및 네트워크 시스템을 통해 외부로 유출된다면 위험한 일이 아닐 수 없다. 따라서, 본 논문에서는 안전한 운영체제를 위해 리눅스 마이크로 커널에 역할기반 접근제어 메커니즘을 적용하였다. 마이크로 커널에 RBAC을 적용하여 역할에 따른 사용자의 접근 권한을 다양하게 부여할 수 있다. RBAC메커니즘을 이용한 마이크로 커널 기반의 보안 운영체제에 대한 완벽한 설계와 이 설계를 바탕으로 TCSEC B2등급의 보안 운영체제를 구현할 것이며, 이는 다양한 IDS 및 방화벽의 하부 구조에도 이용될 것이다.

이러한 설계 방법은 보안 관련 서버가 마이크로 커널로 설계되어 안전성 측면에서 이점이 있으며, 기존 리눅스 커널의 수정을 최소화하면서, 다른 접근제어 모델을 사용할 때에는 관련 모듈만 교체만 하면 되기 때문에 이미 설정된 보안 정책을 손쉽게 변화 할 수 있다. 후에 통합 커널 방식의 보안 시스템과의 성능 비교가 더 진행되어야 할 것이다.

참고문헌

[1] 박태규, 임연호, "리눅스 커널 기반의 안전한 OS 개발," Proc. of KOSTI 2000, 2000년 12월.

[2] 손득종, "디자인 패턴을 이용한 객체 지향적인 RBAC모델," 석사학위 논문, 아주대학교, 2000년 2월.

[3] J. G. Ko et al. "Design and Implementing for Secure OS based on Linux," Proc. of WISA2000, pp.175-182, Nov. 2000.

[4] 김대중, 김현정, 김정래, 박태규, 조인구, 임연호 "다중등급보안 리눅스 기반의 RBAC 시스템 구현," Proc. of CISC 2001, pp.39-42, 2001년 11월.

[5] 박재경 "Linux에서 BLP 보안모델의 특성구현," 석사학위 논문, 홍익대학교, 1996년 2월.

[6] J Ulfar Erlingsson ,Athanasios Kyparlis "Microkernels," http://os.korea.ac.kr/~yuko/research/microkernel/cornel.htm.

[7] 홍승표, 최용호, 박태규 "L4Linux 다단계 보안을 위한 접근제어 서버 프로토타입 설계 및 구현," Proc. of CISC '99, pp.97-107, 1999년 11월.