通信情報保護學會誌
第8卷 第2號, 1998. 6

운영체제 보안 기술 동향

김 학 범*, 오 경 희*, 권 현 조*, 구 자 동*, 홍 기 웅*

요 약

본 고에서는 운영체제 보안에 대한 개념과 이를 위한 보안커널의 구현 전략과 방법을 다루었다. 현재 이 분야의 세계적인 추세와 동향을 파악하고자 미국 정부에서 주도하는 안전한 운영체제 개발 중 활발하게 활동중인 Synergy 연구 프로그램 및 DTOS 프로토타입을 소개하였다. 또한 민간 업체에서 개발되어 사용되는 제품 현황과 미국의 TCSEC과 유럽의 ITSEC에 근거하여 평가된 운영체제 보안 제품 목록을 소개하였다.

1. 서 론

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.[1] 보안커널이 이식된 운영체제는 컴퓨터 사용자에 대한 식별 및 인증, 강제적 접근통제, 임의적 접근통제, 재사용 방지, 침입 탐지 등의 보안 기능 요소를 갖추어야 한다((그림 1) 참조).[2]

인터넷과 같은 네트워크 환경에서 유닉스가 가지는 "개방성"은 중요한 특성이지만 컴퓨터 내의 정보보호를 향상시키기 위한 도구는 현재의 표준 유닉스에서는 매우 부족한 실정이다. 한 통계자료에 의하면 성공한 네트워크 공격의 약 8%가 유닉스 시스템 자체의 취약점을 공격함으로 해서 이루어졌다고 한다. 이에, 기존 유닉스 시스템의 취약점을 보완하는 패치버전이나 업그레이드를 통한 임시방편적인 방법보다는 원천적으로 새로운 안전한 유닉스 운영체제의 필요성이 대두되고 있다.

본 고에서는 안전한 운영체제와 보안커널 개념을 소개하고, 미국 정부에서 주도하고 있는 안전한 운영체제 개발 현황을 소개하고, 민간 업체에서 개발되어 상용제품으로 사용되고 있는 제품 현황 및 현재까지 TCSEC(Trusted Computer System Evaluation Criteria)[3]과 ITSEC(Information Technology Security Evaluation Criteria)[4]로 평가된 제품목록을 소개하였다.

*한국정보보호센터

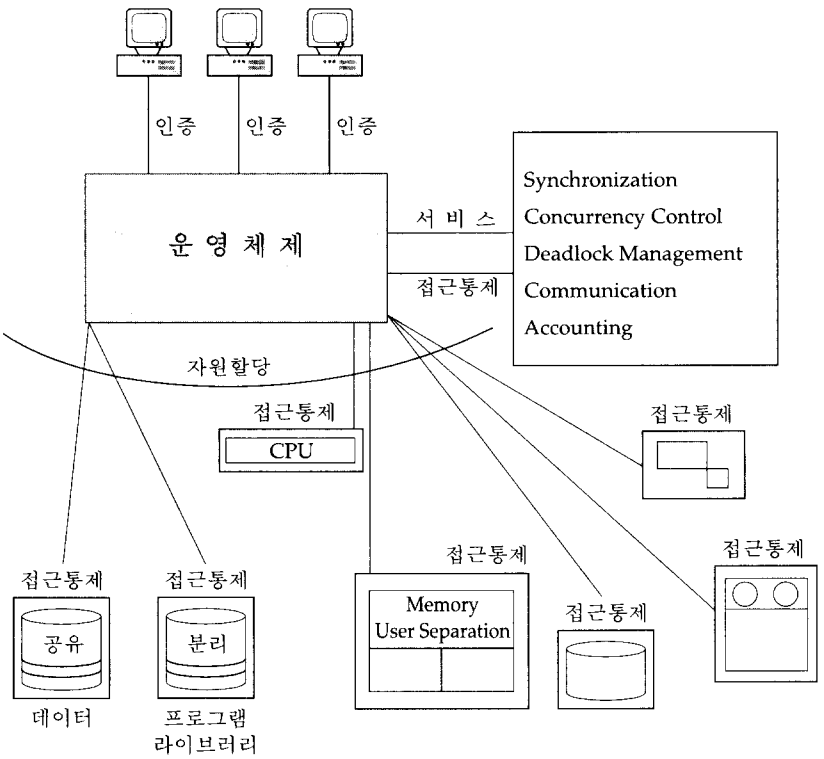


그림 1. 운영체제 보안 기능

2. 안전한 운영체제 및 보안커널 개념

2.1 컴퓨터 시스템의 구조

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.

2.2 보안커널

시스템에 대한 보안은 기본적으로 구조를 변경하지 않고 여러가지 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.[5]

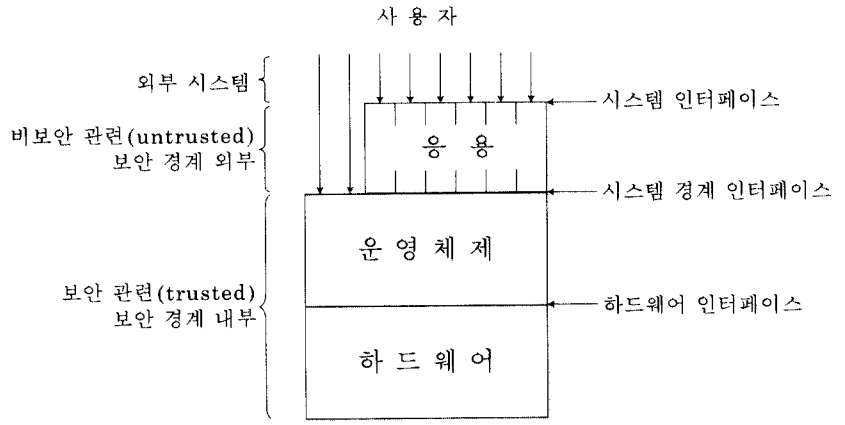


그림 2. 일반 컴퓨터 시스템 구조

2.3 보안커널 구현 전략

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.

주어진 컴퓨터 하드웨어 상의 안전하지 않은 운영체제(ISOS, INsecure Operating System)에 보안커널을 구현하는 방법은 (그림 4)와 같이 세가지 방법으로 나눌 수 있다.[5]

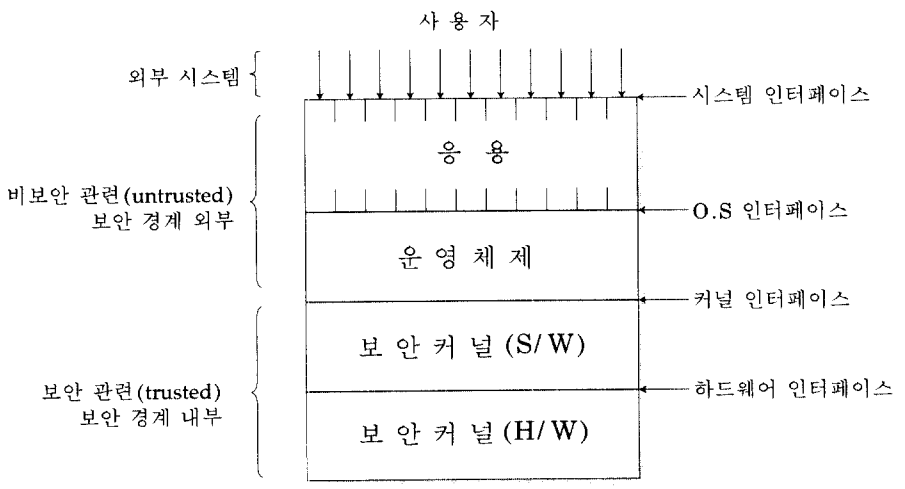


그림 3. 컴퓨터 시스템의 보안 커널

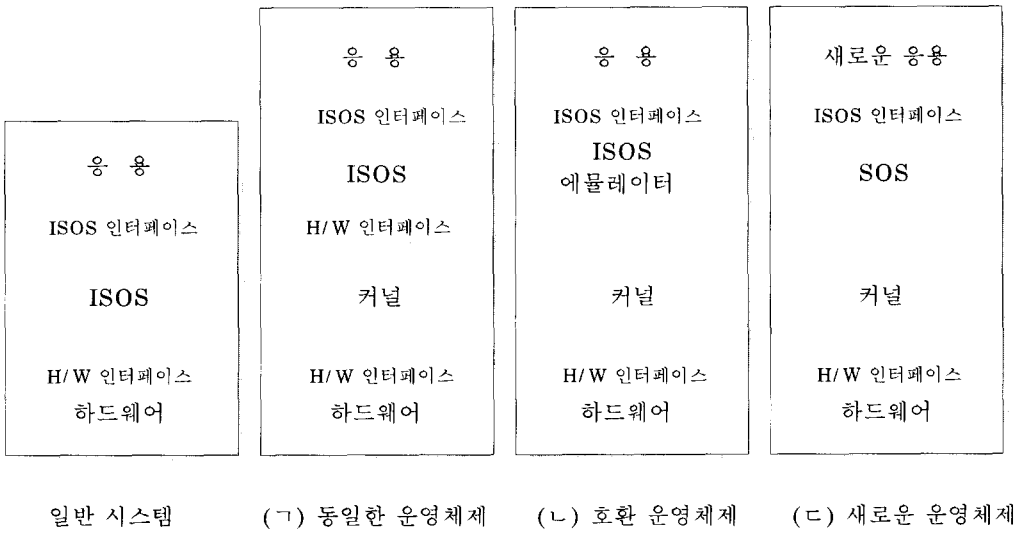


그림 4. 보안커널 구현방법

(1) 동일한(Identical) 운영체제

실행코드의 호환성을 제공하여 기존의 응용을 대체하는 방법이지만 기존의 운영체제 변경시(Upgrade 등) 새로운 배포 문제가 발생하여 더 많은 제약이 발생한다.

가상기계(Virtual Machine) 모니터 개념을 적용하며, 3개의 도메인(커널 도메인, 운영체제 도메인, 응용 도메인)으로 구성된 구조를 갖는다. 이 시스템의 구조는 (그림 5)과 같다.

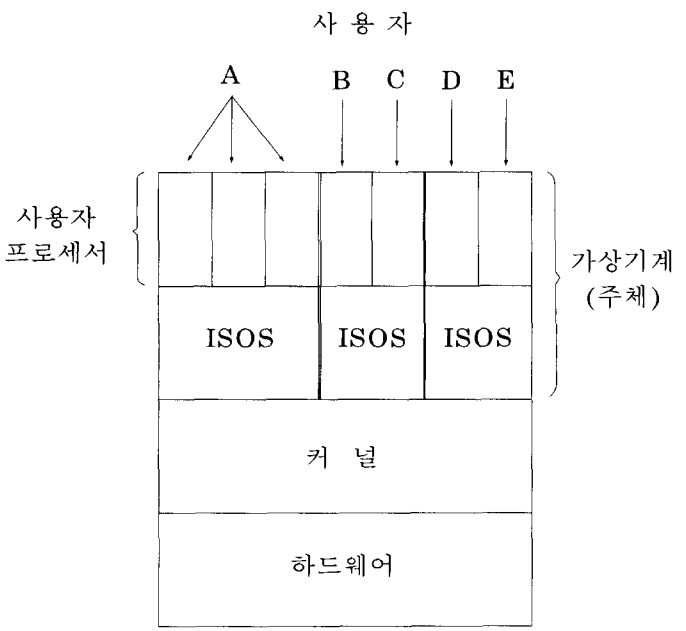


그림 5. 가상기계 모니터 방법

이 구조에서 커널은 다중의 가상기계를 지원하는데, 각각은 원래 운영체제의 복사본을 실행시킨다. 각각의 운영체제는 다중의 응용과 사용자 서비스가 가능하고, 커널은 가상기계간의 수직적인 분리를 시행하며, 커널이 보조메모리 상의 파일시스템을 관리할 수 있다. 그 방법의 예로는 각 가상기계마다 디스크의 비밀영역(private area)을 할당하는 것이다. 이 방법은 기존 운영체제의 소스코드를 가능한 많이 사용하는 것이다. 예로써 IBM VM/370, KVM 등을 들 수 있다.

(2) 호환(Compatible) 운영체제

기존의 ISOS를 완전히 재설계하는 방법으로 기존의 모든 응용이 지원되어야 한다. 해결 방법은 기존의 ISOS 인터페이스와 같게 에뮬레이터를 구현하므로써 응용프로그램이 실제 ISOS와 ISOS 에뮬레이터를 분별할 수 없도록 하는 것이며, ISOS 에뮬레이터와 커널 사이의 인터페이스 정의에는 제약이 없다. 이는 또한 커널이 외부에서 동작하며, 사용자 프로그램 보다 특권(privilege)을 갖는 하나의 프로그램으로서 존재하는 운영체제 에뮬레이터를 구현하는 방법이다.

예로서는 UNIX 에뮬레이터를 보안커널의 최상위에서 실행시키는 KSOS를 들 수 있으나 비용상의 제한으로 제대로 구현되지는 못하였다.

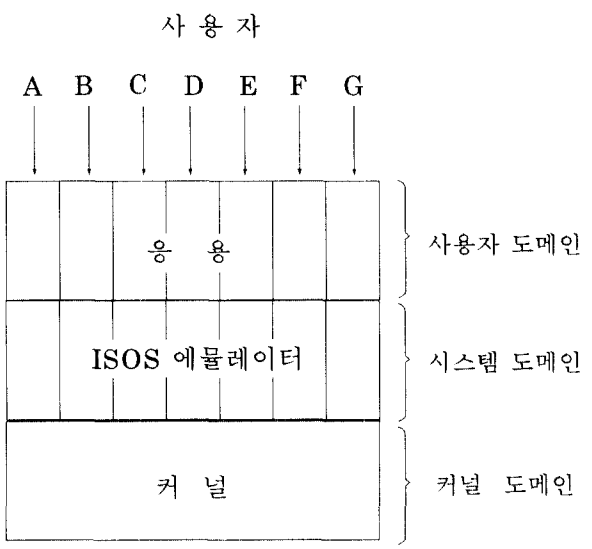


그림 6. 운영체제 에뮬레이터

운영체제 에뮬레이터의 구조는 (그림 6)과 같이 가상기계와 유사한 3 도메인 구조이며, 하드웨어와 커널간, 커널과 ISOS 에뮬레이터 간의 인터페이스를 설계하는 방법으로 인터페이스를 위한 기능을 분할한다. 이는 원래의 운영체제 소스코드를 그대로 사용하여 에뮬레이터를 구현한다는 이점이 있다.

그러나 이 구조는 일부 기능들이 새로운 보안정책(Security Policy)에 대하여 안전하지 않으며 에뮬레이터 될 수 없다는 것과 일부 기능들은 안전하기는 하나 에뮬레이트 하기가 매우 어렵다는 문제점이 있다.

기존의 운영체제를 커널화하는 프로젝트를 수행하기 전에 에뮬레이트할 기능에 대한 세

심한 분석과 함께 정책을 위반하는가를 정확히 결정해야 한다.

(3) 새로운 운영체제

커널과 운영체제를 새롭게 설계하므로써 기존의 ISOS와 응용에 제약이 없게 하는 방법으로, 시스템 구조는 에뮬레이션의 경우와 유사하다. 구현 예로서는 SCOMP를 들 수 있다.

구조는 커널의 통제하에서 각 프로세스 내에서 운영체제 코드의 부분이 (그림 6)과 유사하지만, 운영체제가 구현되는 공유의 형태에 따른 제약 때문에 호환성이 없다. 따라서 보안정책에 따라 정보를 공유하기 위해 커널 프리미티브를 사용하기 위해서는 파일 시스템과 운영체제의 내부 데이터베이스를 설계하는 것이 더 나은 방법일 것이다.

3. 미국 정부의 안전한 운영체제 개발 현황

미국 정부에서는 정부기관 및 군사 기관들의 컴퓨터 네트워크에 사용할 목적으로 안전한 유닉스 운영체제 개발을 사업화하여 4~5년 전부터 타당성 검토를 마치고 현재 프로토타입이 개발된 상태에 있다.

NSA(National Security Agency)에서 NIST(National Institute of Standards and Technology), DISA(Defence Information Security Agency), ARPA(Advanced Research Projects Agency) 등의 기관과 공동으로 추진중인 Synergy 연구 프로그램(Synergy research program)은 미국의 "High Performance Computing and Communications(HPCC) : Advancing the Frontiers of Information Technology" 프로그램 중의 하나이다. 1993년 11월 23일 시행령 12881에 의해 클링턴 정부가 만든 NSTC(National Science and Technology Council)의 CCIC(Committee on Computing, Information, and Communications)의 일부인 HPCC는 정보기술, 컴퓨팅, 통신 및 정보기반구조를 이끌어 21세기에도 자국 국민들의 생산성을 유지할 수 있도록 경쟁력을 유지하고 확장시키기 위한 것이다[6].

Synergy 연구 프로그램의 연도별 주요 성과는 [표 1]과 같다[7].

표 1. Synergy 연구 프로그램의 연도별 성과

년 도	내용
1995년	◦초기 보급용 Synergy 프로토타입 개발 완료 ◦공동 연구를 위해 NIST 및 각 대학들에 배포 ◦네트워크 보안 서비스 프로토타입의 초기 단계 구현(네트워크를 통한 두 호스트간의 기본적인 접근통제) ◦인터넷 보안 관련 키관리 프로토콜을 Synergy 네트워크 보안서버에 탑재 ◦직무-기반(role-based) 접근통제 구현을 시작
1996년	◦두 번째 Synergy 소프트웨어 보급 ◦보안, 보증(assurance), 생산성, 가용성 그리고 분산 및 실시간 운영체제 환경에 대한 이식성을 높임 ◦보안관리와 Synergy component가 개선된 버전 개발 및 데모 ◦지속적인 Synergy 구조에 대한 연구와 연구결과의 성공적인 기술 이전 ◦기업 및 정부기관의 보안 요구사항을 만족시킬 수 있는 Synergy 시스템 개발을 위해 상용 생산업체들과의 공동 연구(DTOS)

1997년	◦네트워크 보안관리 추가 구현 ◦완전한 암호 서버 프로토타입의 데모 ◦윈도우 시스템을 사용한 인터페이스 개선 ◦DTOS 프로토타입 구현 완료('97.6)

3.1 Synergy 설계 원칙

Synergy 연구 프로그램의 설계 원칙은 [표 2]와 같다

표 2. Synergy 연구 프로그램의 설계 원칙

원 칙	내 용
유연성 (Flexibility)	◦위협 환경에 따라 구성가능한 보안 서비스 ◦메커니즘에 독립적인 보안 서비스 ◦일반 보안 API 내에 캡슐화된 보안 서비스/메커니즘
투명성 (Transparency)	◦운영체제 기반구조에 통합된 보안 ◦자동적으로 시작되는 보안 서비스 ◦자동화된 보안관리 기능
모듈성 (Modularity)	◦독립적으로 평가될 수 있는 모듈로 분해된 구조 ◦모듈간에 잘정의된 인터페이스 및 상호독립성 ◦모듈의 구성에 대해 판단할 수 있는 능력
계층적 보안 (Layered Security)	◦가능한 많은 위협을 처리할 수 있는 기본 시스템 구성요소 ◦다른 TCB 구성요소상에 최소한으로 증명된 의무

3.2 Synergy 구조

Synergy의 구조는 (그림 7)과 같이 마이크로커널(Microkernel) 기반구조이며, 보안정책 서버(Security policy server), 암호 서브시스템(Cryptographic Subsystem), 인증 서브시스템(Authentication Subsystem)으로 구성된 보안 서비스 서브시스템으로 구성되어 있으며, 구성가능한 네트워크 및 파일 서비스 구조를 가지고 있다.

마이크로커널 기반구조를 사용하는 이유는 위에서 언급한 Synergy 설계원칙을 만족시키기 위한 것이다. 메커니즘과 정책간의 분리를 강조함으로써 유연성을 제공할 수 있다. 또한 마이크로커널에 의해 많은 위협을 투명하게 처리할 수 있다. 또한, orthogonal 서버로 운영체제를 분해함으로서 모듈성을 제공하며, 마이크로커널이 운영체제 서버의 활동들을 제한함으로서 계층화된 보안을 제공할 수 있다.

보안정책 서버는 강제적 접근통제(MAC, Mandatory Access Control)[8][9]를 제공하기 위한 것이며, MAC의 세부 항목(마이크로커널, 파일 서비스, 네트워크 서비스, 응용)으로부터

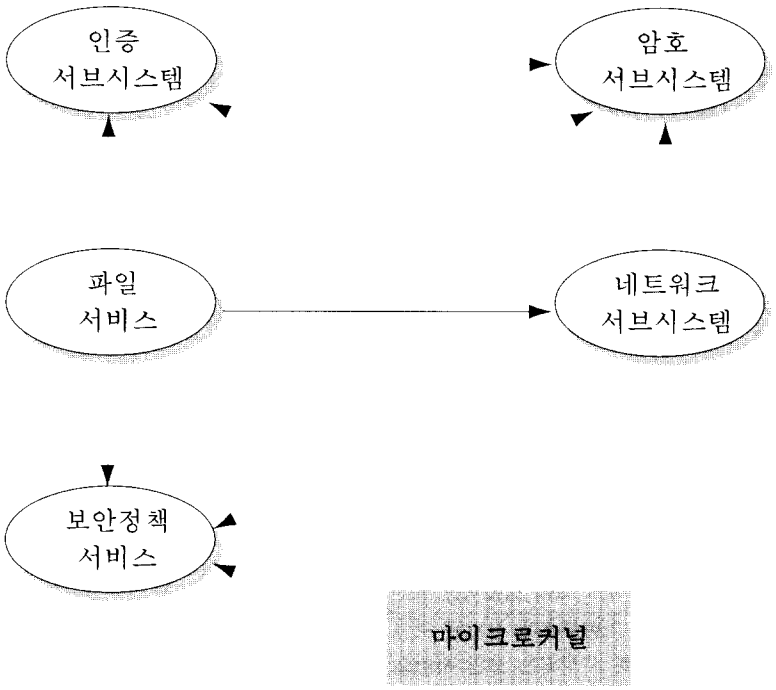


그림 7. Synergy 구조

정책 시행자를 분리시킴으로서 MAC 정책이 변경되었을 경우, 변경된 부분만을 재평가함으로써 비용을 줄일 수 있다.

암호 서브시스템은 암호기능을 제공하기 위한 것으로 암호사용 정책과 암호메커니즘에 대한 내용을 다룬다. 암호알고리즘의 세부 항목(파일 서비스, 네트워크 서비스, 응용)으로부터 클라이언트를 분리시킴으로서 암호 기능의 변경시에 비용을 줄일 수 있다.

4. DTOS

DTOS(Distributed Trusted Operating System)는 NSA의 Synergy 프로그램 중의 일부로 강력하고 유연성 있는 보안 통제 제공을 목적으로 만든 운영체제의 프로토타입으로 1997년 6월에 개발·완료되었다[10].

4.1 목적

현재의 Synergy 노력은 차세대에 상용 운영체제에 강력한 보안 메커니즘을 포함하도록 운영체제 개발자들을 독려하는 장기적인 전략 중의 하나이다.

DTOS 프로그램의 주목적은 Synergy 프로그램의 기본적인 요구사항을 만족시키기 위한 것으로 Synergy 구조내에서의 최하위수준의 소프트웨어 구성요소에 대한 프로토타입을 개발하고, 보안 요구사항을 만족시키기 위한 보증 문서와 증거의 제공 즉, Synergy 구조를 따르는 전체 시스템의 보안에 대한 증거를 어떻게 다룰 것인가를 연구하는 것이다.

4.2 프로토타입의 목적

DTOS는 High assurance multilevel secure

환경에 사용하기 위해 B3 등급을 목표로 설계된 DTMach 프로그램(그림 8)을 대체하는 것으로 안전한 분산 운영체제를 위한 고수준의 설계를 하는 것이다. DTOS 프로토타입 개발은 거의 마이크로커널과 보안서버의 설계로 집중된다.

마이크로커널은 기존의 Mach3.0을 개선하는 방향으로 설계되었다. 이는 기존의 마이크로커널에서 요구하는 보안에 대한 필요성이 크게 다르지 않다는 것을 나타낸다.

Mach 마이크로커널을 선택한 이유는 하나의 통제 메커니즘으로 거의 모든 시스템 오퍼레이션을 통제할 수 있으며, 널리 사용되는 플랫폼이라는 점과 함께 유닉스 응용을 지원하며, 소스코드를 별도의 동의 절차 없이 사용할 수 있었던 등등 다양하다.

보안서버에는 시스템의 보안정책이 포함되는데 보안정책의 주 기능은 마이크로커널과 같은 다른 서비스를 제공하는 구성요소에 의해 시행되는 보안결정(security decision)을 제공하는 것이다.

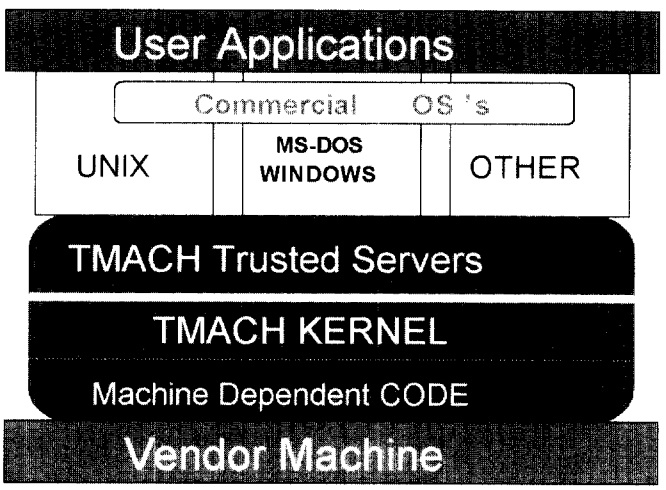


그림 8. DTMach 구조

프로토타입의 설계 목적은 정책의 유연성을 제공하고 Mach와의 호환성을 유지하며 Mach 커널과 유사한 성능을 갖는 시스템을 구현하는 것이다. DTOS는 위와 같은 요구사항을 충족하며 기존의 Mach에서의 여러 가지 문제점을 개선했다. Mach의 문제점과 이를 해결한 DTOS에 대한 비교 내용은 [표 3]과 같다.

표 3. Mach의 문제점과 DTOS의 개선 방향

구 분	내 용	세 부 내 용
Mach의 문제점	포트 권한 전송에 대한 제한된 통제	◦전송에 대한 적법성 여부 판단에 대한 메커니즘 제공 결여 ◦통제는 중간 대리자를 요구
	객체 관련 서비스의 통제 결여	◦포트권한에 대한 소유는 객체 동작에 대한 모든 권한을 소유하는 결과 발생 ◦다중 포트에 대한 부분적인 서비스를 요구
	송신 식별자의 결여	◦다중 메시지 전송자에 대한 식별이 불가능 ◦다중 지역 인증은 사용자 레벨의 인증 프로토콜을 요구
	메시지 탈취에 대한 보안 부족	◦신뢰성 있는 경로 메커니즘의 제공이 없음 ◦커널에서 메시지 수신자 지정을 명시하는 메커니즘이 없음
DTOS 개 선 방 향	커널 객체에 대한 보안 정보의 결합	◦보안식별자는 작업, 포트, 메모리공간, 장치를 결합 ◦특정 보안식별자를 갖는 객체 생성 가능 ◦객체로부터 보안식별자 도출 가능 ◦타겟 작업 식별자로부터 객체 보안식별자 도출 가능
	개선된 객체 서비스 통제	◦커널 객체 동작에 대해서 접근벡터를 기반으로 통제 ◦수신자는 송신측의 접근통제 벡터 정보의 획득 가능
	보안정책서버와의 상호작용	◦커널은 subject SID와 object SID를 제공함으로써 접근벡터 요구 가능 ◦접근벡터 캐쉬 관리 기능 추가(flush, wire 등)
	IPC 통제에 대한확장 기능	◦수신자는 송신 작업으로부터 보안식별자 도출 가능 ◦송신자는 수신자 지정을 명기할 수 있는 기능 ◦접근벡터에 대한 IPC 허가여부 검사 가능

4.3 프로토타입 개요

1) 구조

DTOS의 접근방법은 모든 커널 오퍼레이션을 통제하는 일반적인 메커니즘을 제공하므로서 커널 엔티티를 통하여 정의된 많은 정책을 허용하는 것이다. 이 방법의 이점은 상위 계층에서 보안 특성이 침투당하더라도 방어하는 다른 계층을 제공할 수 있다는 것이다. 예를 들면, KeyKOS/KeySAFE 및 TrustedMach는 서버계층의 단일 보안 유지가 실패하면 침입자는 금지된 많은 오퍼레이션을 수행할 수 있는 기회를 제공한다. 이것은 전체 보안을 파괴되는 결과를 초래할 수도 있다[10].

이 이외에도 커널기반구조를 채택하므로서 얻을 수 있는 이점으로는 안전한 서버와 응용 내에서 보안 기능을 쉽게 개발할 수 있으며, KeyKOS/KeySAFE나 TrustedMach처럼 특정 보안정책에 국한되지 않고 일반적인 모든 커널 엔티티의 보호를 위한 단일 모델을 제공할 수 있다.

보안정책의 유연성과 localization의 지원을 위해 DTOS 구조는 보안정책 시행과 보안정책 의사결정 간에 엄격한 분리를 제공한다. 그리고 DTOS 커널은 모든 커널 내의 동작을 클라이언트의 접근통제에 대한 보안서버로부터 허가 여부를 참조하여 보안정책을 시행한다.

2) 커널 확장

보안을 위한 커널 확장의 주목적은 보안서버가 모든 커널 동작에 대한 타당성 여부를

결정하는 기능을 제공하는 것이다. 그러나 보안커널은 클라이언트나 객체의 실제 신분에 대한 정보를 보안서버에게 요구하지 않고 대신에 각 클라이언트와 객체에 대한 보안식별자(security identifier, SID)를 사용한다.

DTOS 커널에 대한 주요 확장 내용은 다음과 같다:

- 커널은 보안식별자와 커널 객체간의 관계를 관리한다.
- 커널은 각각의 커널 동작에 대한 접근통제와 커널 동작에 대한 허가 여부를 결정하기 위해 정보를 가져오는 보안서버와의 인터페이스를 제공한다.
- 커널은 보안서버로부터 하드웨어 기반 메모리 접근통제를 정의하기 위해 보안결정을 사용한다.

DTOS의 상세한 커널 확장 내용은 [표 4]와 같다.

표 4. 확장된 커널 내용

내 용	구 현
보안식별자 관리 (SID Management)	◦Mach 인터페이스와 차별화된 객체 생성 인터페이스 기능 ◦기본적으로 SID는 변경 불가능하게 관리 ◦새로운 객체에 대한 클라이언트의 SID 명명 기능
커널 접근통제 (Kernel Access Control)	◦모든 커널 동작에 대한 접근통제가 가능한 단일 모델 제공 ◦클라이언트와 접근되어지는 객체간에 단일 허가 여부 Check 기능(Initial permission checks) ◦커널 내부에서 수행되어지는 중에 허가 여부를 판단해야 하는 경우에 필요한 접근통제 기능(Deferred permission checks)
메모리 접근통제 (Memory Access Controls)	◦메모리의 읽기, 쓰기, 실행에 대한 접근통제 기능 지원
타스크 생성 인터페이스 (Task Creation Interface)	◦타스크가 다른 타스크를 생성하는 인터페이스 지원 ◦생성자는 하부 타스크를 새롭게 초기화 할 수 있는 인터페이스 지원
IPC 확장	◦수신자가 수신된 메시지를 통해 송신자의 SID를 도출 가능 ◦송신자가 송신 메시지에 대한 SID를 명명할 수 있는 기능
DTOS 커널 구현시 발생하는 장애 (Obstacles)	◦시스템 변화에 따른 기존 응용프로그램의 이식성 ◦확장 보완된 시스템 변화에 대한 자세한 지식 습득의 어려움 ◦하위 레벨에서의 동작 루틴의 부재 ◦구현의 기본이 되는 Mach code의 비합리적인 구조 ◦소스 코드의 변경 유무에 대한 식별의 어려움 ◦Mach 코드의 버그로 인한 DTOS 구현시에 발생하는 버그 수정의 어려움 ◦Mach 시스템 자체에서 발생하는 문제점과 추가 보완된 기능에서 발생하는 문제점 구별의 어려움 ◦시스템 변경으로 생기는 예상치 못한 부작용 ◦가정으로 시작한 요구사항들의 기능 테스트 ◦Desired Robustness of the Prototype

3) 보안서버 설계 및 구현

보안서버는 DTOS 시스템 내의 보안결정을 하는 구성요소로서 사용자 공간에서 일반적인 Mach 타스크처럼 수행된다. 보안서버의 기능은 보안식별자와 보안 속성간의 매핑을 제공하고, 특별히 요청된 접근 권한이 두 보안식별자 간에 주어질 수 있는가에 대한 판단을 하므로서 보안결정에 대한 응답을 처리한다.

DTOS 프로토타입은 MLS(Multilevel Secure System)와 Type Enforcement 보안정책을 구현하였다. 이는 요구되는 다양한 정책을 쉽게 적용할 수 있도록 모듈화된 설계를 하였다. 보안서버의 설계와 구현을 위해서 고려되어야 할 주요 요소는 다음과 같다.

- 보안식별자의 해독 : 보안서버는 보안식별자와 보안 속성간의 매핑 관계를 유지 한다. 따라서 보안식별자와 보안 속성간에 변형시키기 위한 인터페이스가 제공되 어져야 한다. 보안커널은 명명된 보안 속성에 대해서는 알 필요가 없기 때문이다.

보안식별자는 다음과 같은 보안 속성 필드와 연결된다.

- 분류자(Classifier) : 정책과는 독립적인 필드
- 시행 속성 타입(domain or type)
- 계층적인 보안 레벨 (top secret, secret, confidential, unclassified)
- MLS 정책을 위한 계층 레벨 변수의 범주 집합
- 인증 문맥 (사용자 이름 등)

초기에 보안식별자는 보안 속성과 직접적으로 연결이 되었으나 최근 매핑 구조는 보안서버내에서만 해석 가능하게 되었다. 이것은 보안서버 이외의 응용 개발자들 에 의한 불법적인 사용에 대한 위협 요소를 제거하기 위한 것이다.

- 보안결정 생성 : 보안서버의 핵심은 보안결정 생성 기능이다. 보안서버의 기본형은 속성들의 상호 동작에 대해서 정의된 Hard-code 로직과 명명된 속성간에 상호 작용과 특성을 허락하는 보안 데이터베이스 등의 조합을 기반으로 보안결정을 생성한다.

- 안전한 데이터베이스 : 보안정책은 안전한 데이터베이스에 저장되어 있다. 그러나 관리자가 거대한 보안 데이터베이스를 관리하기 위한 도구가 매우 부적절하거나 부족하다. 보안 데이터베이스 사용 도구의 부족은 커널 동작에 대한 높은 수준의 통제를 제공하는 DTOS를 관리할 수 없게 되는 문제가 발생한다.

- 사용자 레벨의 정책 시행 : DTOS 커널 입장에서는 보안정책은 클라이언트 보안식별자와 객체 식별자간에 허가 여부를 판단하는 접근 권한의 집합이다. 이것은 매우 간단한 인터페이스로 처리 가능하지만 상위 계층의 클라이언트에 의해 요구 되는 보안정책 결정은 언제나 단순하지 않다. 상위 계층의 보안정책 제안은 동작을 수행할것인지의 여부를 판단하는 것 대신에 어떻게 수행할 것인가에 대한 정의를 담고 있다. 즉 선택된 암호 알고리즘 또는 레이블된 출력 데이터 등 다양한 선택이 가능하다.

4) 커널 및 보안서버 인터페이스

커널과 보안서버와의 상호작용은 커널이 통제점(control point)에 도달했을 경우에 발생하는데 이 시점이 보안결정을 위하여 보안서버

를 질의하는 것이다.

커널이 통제점을 만나 보안결정을 요구하게 되면, 먼저 내부 캐쉬를 살펴본 후에 캐쉬에 없다면 security fault를 내고 보안서버로부터의 결정을 요청하게 된다. 커널과 보안서버에 캐싱 메커니즘을 지원하기 위해서는 프로토타입 개발과정이 복잡해진다.

이를 해결하기 위해서 [표 5]와 같은 내용을 구현하였으며 개념도는 (그림 9)와 같다.

표 5. 커널 및 보안서버 인터페이스

내 용	커널에 추가되는기능	기능 및 세부내용
캐쉬 최적화 (Cache Optimization)	◦ Access Vectors	◦ 접근벡터를 가장 간단하고 강력하게 최적화 하는 기법 ◦ 보안커널 내에서는 (보안식별자, 보안식별자)로 구성
	◦ Elimination of Redundancies	◦ 보안식별자는 MID(Mandatory identifier), AID(authentication identifier)로 구성 ◦ AID 값은 접근벡터와는 독립적으로 캐쉬 내에서 제거함으로 캐쉬의 효율 증가
	◦ Second Level Cache	◦ 수행하는 작업은 자신의 보안식별자와 포트 위치를 소유하고 있으므로 포트 위치 정보로 캐쉬된 접근벡터로 빠르게 접근하기 위한 캐쉬를 첨가
캐싱 및 정책의 유연성 (Cashing and Policy Flexibility	◦ Cache Flushing	◦ 가장 간단한 방법 ◦ 커널이 캐쉬에 있는 접근 벡터 내용을 일정 시간 후에 Flushing하는 인터페이스 제공 ◦ 캐쉬의 부분 혹은 전체를 Flushing할 수 있는 기능 제공 ◦ 보안서버로부터 온 메시지의 순서제어 기능 추가 필요
	◦ Cache Control Vector	◦ 보안서버의 정책 변화에 대한 커널 내의 캐쉬 Flushing이 허용 시간 내에 가능하도록 하는 요구사항 충족 ◦ 캐시 조절 벡터 비트는 현재 캐쉬된 보안정책과 보안서버내의 보안정책과의 동일성 여부를 판단하기 위해 사용
	◦ Timeout Value	◦ 캐쉬된 내용의 유효성 여부를 보장하기 위한 수단 ◦ 보안서버가 절대 시간을 설정하여 그 이후에는 유효하지 않음을 나타내어 커널이 이 시간을 이용하여 캐쉬내의 접근벡터의 유효성 여부 판단 ◦ 정기적인 보안서버와 커널간의 통신을 이루는 수단
	◦ Adding Wired Cache Entries	◦ 커널 내의 캐쉬 메모리의 효율적 사용을 목적으로 함 ◦ 재사용 여부가 희박한 엔트리에 대한 제거 ◦ DeadLock 방지 수단 ◦ 새로운 벡터에 대한 표시 기능이 추가된 벡터 기능으로 제공
	◦ Changing Security Server	◦ 보안정책 변화로 보안서버의 변경에 따르는 커널에서의 문제를 해결 ◦ 새로운 보안서버와 연결되면 기존의 캐쉬 내용을 Flushing하고 Dead Lock을 방지하기 위해서 제안한Wired 벡터 등을 사용
보안결정의 감사 (Audit of Security Decisions)	◦ 캐쉬된 보안결정 정보는 보안서버에서 감사를 할 수 없는 문제점 발생 ◦ Notification vector를 추가하여 접근벡터의 해당 비트가 보안결정을 위해 사용될 경우 감사 기록을 생성하는 메커니즘 제공	

표 5. 커널 및 보안서버 인터페이스(계속)

내 용	커널에 추가되는기능	기능 및 세부내용
인터페이스 상의 반영 (Reflections on the Interface)	◦Design Alternative (물리적인 관점)	◦시스템 설계를 위해 고려되는 주요 요소 - 커널의 보안정책 시행 요소 - 보안결정 생성에 대한 요소 - 단일 보안정책 데이터를 보관하는 보안 정책 DB
		◦대안 A - 보안결정 생성자와 보안정책 데이터 베이스가 모두 사용자 작업 공간에 존재
		◦대안 B - 보안정책 결정 기능의 일부분이 커널의 한부분으로 존재(정책에 독립적인 설계 가능) - 모든 정책 의존적인 보안정책 결정 기능과 보안 데이터베이스는 사용자 작업 공간에 존재 - 커널에 구현되는 부분은 접근벡터 캐쉬와 캐쉬내 엔트리의 유효성을 판단하는 부분
		◦대안 C - 정책 의존적인 결정 기능의 일부분이 커널 주소 공간에서 실행 - 커널 정책 실행자는 각각의 특정한 보안정책에 관여할 경우마다 새로 연결하여 시행
	◦Tradeoffs	◦성능(Performance) - Mach 커널과 동등한 성능을 유지 - 대안 A : 커널로부터 사용자 작업 공간으로의 외부호출로 인한 잦은 문맥교환과 IPC로 Overhead 증가, DTOS 설계의 선택에서 제외 - 대안 B : 캐쉬 기능이 없으면 A와 유사한 성능을 나타냄 - B > C > A (성능순)
		◦정책 유연성(Policy Flexibility)
		◦코드의 복잡성(Code Complexity) - C > B > C (코드 복잡도)
		◦보증(Assurance) - 커널과 보안서버는 보안에 가장 민감한 요소임
		◦디버깅의 용이(Ease of Debugging) - 마이크로 커널 기반 시스템의 장점은 모듈 및 인터페이스에 독립적으로 수행함으로 디버깅이 용이 - A > B > C (디버깅 용이도)
		◦장치 접근(Device Access) - 보안정책 데이터베이스는 일관된 정책 정보를 저장하는 공간으로 사용 - 대안 C3은 성능에 중점하여 보안에 치명적인 결점
		◦User Space Client Access - 대안 A : 클라이언트와 보안서버간의 인터페이스는 커널과 보안서버간의 인터페이스와 동일, 각 Permission 검사는 한쌍의 IPC 메시지를 요구 - 대안 B : 클라이언트가 캐쉬 기능을 사용하지 않으면 각 인터페이스를 지나는 정보를 대부분 무시, 캐쉬 기능을 사용한다면 모든 클라이언트를 수용할 수 있는 표준 캐쉬 모듈이 제공되어야 함 - 대안 C : 구현적인 측면에서 가장 유연함, 단순히 보안정책 결정 요구를 하는 커널 요구에 대해서 보안결정 생성자에게 첨가시킴으로 해결
		◦관리적인 클라이언트 접근 - 보안정책 데이터베이스를 관리하는 클라이언트 존재 - 보안 기능을 갖는 물리적인 인터페이스로 요구되는 기능 충족 - 위에서 제안한 3가지 대안이 각각의 정책 명기

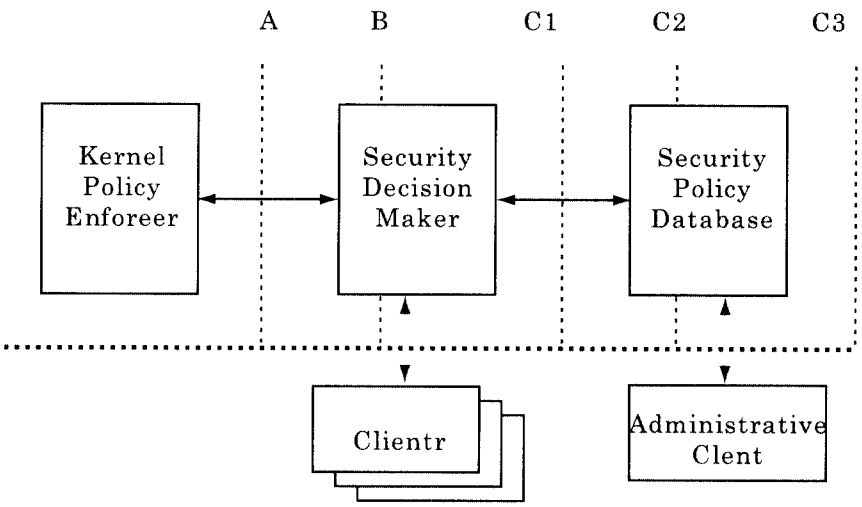


그림 9. 커널 및 보안서버 인터페이스

4.4 향후 계획

Synergy 프로그램의 최종 목표는 아직 정해지지 않았지만, DTOS 프로토타입이Mach 코드 기반으로 개발되었기 때문에 높은 보증을 제공하기에는 아직 충분하지 않다. DTOS 프로토타입은 현재 더 진행되고 있지는 않지만 다른 Synergy 노력들이 이와는 독립적으로 진행되고 있다. 차세대 Synergy 마이크로커널은 Synergy 연구 사이트에서 배포될 예정인데 이 마이크로커널은 운영체제에서 Mach의 논리적인 계승자인 Utah 대학에서 개발중인 Fluke 마이크로커널[11]에 기반을 둘 것으로 예상하고 있다.

5. 민간업체의 안전한 운영체제 개발 현황

안전한 운영체제는 정보기관이나 국방관련 기관 뿐만아니라 금융기관, 정부기관, 병원, 정보통신 회사 등 정보보호가 중요시되는 기관을 주요 시장으로 하고 있으며 각급 기관들이 인터넷 등을 통한 외부 사이트와의 접속으로 인해 보안 위협 및 안전한 운영체제의 필요성이 대두되고 있다.

민간업체의 안전한 운영체제 개발 현황을 주요 업체별로 살펴보면 [표 6]과 같다[12].

표 6. 민간업체의 안전한 운영체제 제품 개발 현황

업체명	제 품 명	특 징
IBM	MVS/SP VERSION 5 RELEASE 2.2 with RACF Version2 Release 2	◦B1 등급
	ES/9000 PM/SM(Processor Resource /System Manager)	◦ITSEC E4 등급 ◦ES/9000 9021 & 9121
	AIX 4.1	◦C2 security enhancement ◦접근통제 리스트와 개선된 ◦패스워드 보호 포함
Digital Equipment	VMS V5.4	◦C2 등급 ◦원격인증과 suspicious usage monitor가 없음
	ULTRIX MLS+	◦multilevel secure operating system ◦B1 등급
HP	HP-UX BLS	◦HP UNIX의 secure 버전 ◦HP 9000 시스템에 사용 ◦B1 등급
	HP-UX Version 10	◦standard U.S DoD C2 등급 기능
	MPE/iX Release G.03.04	◦C2 등급의 proprietary midrange operating system
ICL	VME/High Security Option(HSO)	◦B1 등급 ◦39 시리즈 메인프레임에 사용
Unisys	OS 1100 Security Release	◦B1 등급 ◦2200 메인프레임에 사용
TIS(Trusted Information System)	Trusted Mach(TMach(R))	◦B3/E5 등급 목표 ◦다중 운영체제 인터페이스 제공
	Trusted XENIX 3.0/4.0	◦B2 등급
SCO	SCO UNIX	◦B1 등급
OSF	OSF/1	◦B1 등급 ◦Kerberos 원격인증의 pioneer
Bull HN	Multics System	◦B1 등급 ◦1996년에 취소
	Securics B1+	◦B1 등급 ◦Bull HN XPS-100 서버상에서 구동
Sun Micr-osystem	Sun Solaris 2.4	◦E2 등급
Microsoft	Windows NT Version 3.5	◦C2 등급

6. 안전한 운영체제 제품 평가 현황

미국의 경우 국방부, NBS(National Bureau of Standards, NIST의 전신) 및 MITRE 등을 중심으로 안전한 컴퓨터시스템의 구축 및 평가 등에 관한 연구를 지속한 결과 1983년 "Orange Book"으로 불리는 안전한 컴퓨터시스템 평가기준인 TCSEC(Trusted Computer Security Evaluation Criteria) 초안이 제정되었고, 1985년에 미 국방부 표준(DoD 5200.28-

STD)으로 채택되었다. 국방부는 안전 신뢰성이 입증된 컴퓨터 시스템을 국방부 및 정부기관에 보급하기 위하여 TCSEC을 6가지 등급(C1, C2, B1, B2, B3, A1)으로 분류하여 각 기관별 특성에 맞는 컴퓨터 시스템을 도입 운영하도록 권고하고 있다. [표 8]은 TPEP(Trusted Product Evaluation Program) 평가제도[13]를 통하여 현재까지 미국의 TCSEC으로 평가받은 안전한 운영체제 제품의 목록이다.

ITSEC은 영국, 독일, 프랑스 및 네덜란드 등 자국의 정보보호시스템 평가기준을 제정하여 시행하던 4개국이 평가제품의 상호 인증 및 평가기준이 상이함에 따른 정보보호 제품의 평가에 소요되는 시간, 인력 및 소요 비용을 절감하기 위하여 제정한 "Harmonized Criteria" 이다[14]. 1991년에 ITSEC v1.2를 제정하였으며 현재도 이 버전으로 평가를 수행하고 있다. ITSEC은 TCSEC과는 달리 단일 기준으로 모든 정보보호 제품을 평가하고자 하였다. [표 7]은 현재까지 유럽에서 ITSEC으로 평가된 운영체제 제품목록이다.

표 7. 미국의 안전한 운영체제 제품 평가목록

업체명	등급	특 징	날 짜
SCOMP STOP Release 2.1	A1	Honeywell Information Systems Inc.	'84.12.24
XTS-200 STOP 3.1 E	B3	Wang Government Services, Inc.	'92.5.27
XTS-200 STOP 3.2 E	B3	Wang Government Services, Inc.	'94.3.9
XTS-200 STOP 4.1	B3	Wang Government Services, Inc.	'95.5.30
XTS-200 STOP 4.1a	B3	Wang Government Services, Inc.	'95.10.31
Multics MR11.0	B2	Honeywell Information System(HIS)	'85.9.1
Trusted XENIX 3.0	B2	Trusted Information Systems, Inc.	'92.4.8
Trusted XENIX 4.0	B2	Trusted Information Systems, Inc.	'93.9.17
XTS-200 STOP 3.1E	B2	HFSI	'92.5.27
UTS/MLS, Version 2.1.5+	B1	Amdahl Corporation	'94.1.7
SEVMS VAX Version 6.0	B1	Digital Equipment Corporation	'94.6.30
SEVMS VAX Version 6.1	B1	Digital Equipment Corporation	'94.7.14
SEVMS VAX and Alpha Version 6.1	B1	Digital Equipment Corporation	'96.10.24
ULTRIX MLS+ Version 2.1 on VAX Station 3100	B1	Digital Equipment Corporation	'96.4.18
CA-ACF2 Release 6.1 with CA-ACF2 MAC	B1	Computer Associates International Inc	'98.3.23
CX/SX Release 6.1.1	B1	Harris Corporation	'93.9.15
CX/SX Release 6.2.1	B1	Harris Corporation	'95.9.18
HP-UX BLS, Release 8.04	B1	Hewlett Packard Corporation	'93.9.21
HP-UX BLS, Release 9.09+	B1	Hewlett Packard Corporation	'94.12.1
Trusted IRIX/B release 4.0.5EPL	B1	Silicon Graphics Inc.	'95.2.6

표 7. 미국의 안전한 운영체제 제품 평가목록(계속)

업체명	등급	특 징	날 짜
OS 1100 Security Release I	B1	Unisys Corporation	'89.9.27
OS 1100/2200 Security Release SB3R6	B1	Unisys Corporation	'91.4.5
OS 1100/2200 Security Release SB3R8	B1	Unisys Corporation	'91.11.5
OS 1100/2200 Security Release SB4R2	B1	Unisys Corporation	'92.10.7
OS 1100/2200 Security Release SB4R7	B1	Unisys Corporation	'94.4.20
Unisys OS 1100 Security Release 1	B1	Unisys Corporation	'89.9.27
System V/MLS version 1.1.2	B1	AT&T	'89.9.7
AOS/VS II. Release 3.01	C2	Data General Corporation	'94.6.1
AOS/VS II. Release 3.10	C2	Data General Corporation	'94.12.5
CA-ACF2 MVS Release 6.1	C2	Computer Associates International Inc.	'98.3.23
OpenVMS VAX Version 6.0	C2	Digital Equipment Corporation	'94.6.30
OpenVMS VAX Version 6.1	C2	Digital Equipment Corporation	'95.7.14
OpenVMS VAX and Alpha Version 6.1	C2	Digital Equipment Corporation	'96.10.24
AS/400 with OS/400 V2R3M0	C2	International Business Machine	'95.10.5
AS/400 with OS/400 V2R3M5	C2	International Business Machine	'96.9.11
AS/400 V3R2 Feature Code 1920 Version 2 hardware	C2	International Business Machine	'97.10.3
Windows NT Version 3.5	C2	Microsoft Corporation	'95.7.31
Guadian-90 w/Safeguard S00.01	C2	Tandem Computers Inc.	'93.6.14
TOP SECRET Version 3.0	C2	CGA Software Products Group Inc.	'85.4.2
UTX/32S Release 1.0	C2	Gould Inc.	'86.12.31
VAX/VMS+	C2	Digital Equipment Corporation	'88.5.9
Primos	C2	Prime Computer Inc.	'88.6.24
MPE V/E Release G.03.04	C2	Hewlett Packard	'88.10.5
AOS/VS Version 7.60	C2	Data General Corporation	'89.2.22
SVS/OS CAP 1.0	C2	Wang Laboratories Inc.	'90.9.28
ConvexOS/Secure V10.0	C2	CONVEX Computer Cooperation	'92.5.29
Trusdted OS/32 with OS/32 MTM Release 08.03-3S	C2	Concurrent Computer Cooperation	'92.10.1
Tandem Guardian 90 with Safeguard S01.00	C2	Tandem Computers Incorporated	'93.6.14
Access Control Facility2(ACF2) Release 3.1.3	C2	SKK Inc.	'84.8.3
Resource Access Control Facility (RACF) Version 1 Realease 5	C1	International Business Machine	'84.7.23

표 8. 유럽의 안전한 운영체제 제품 평가목록

업체명	등급	특 징	날 짜
영 국			
IBM PR/SM ES/9000	E4	IBM United Kingdom Ltd.	'95. 9
BEST-X/C2(Bull Enhanced Security Technology)	E3	Bull Information Systems Ltd.	'97. 6
BEST-X/B1(Bull Enhanced Security Technology)	E3	Bull Information Systems Ltd.	'97. 4
SECURICS Version 7.7.0.1	E3	Bull Information Systems Ltd.	'94. 9
Sequent DYNIX PTX Unix	E3	Sequent Computer Systems Ltd.	'97. 2
Hewlett Packard Ltd.	E3	HP-UX Version 10.10	'97. 1
Maxion/OS	E3	Concurrent Computer Corporation Ltd.	'96. 12
Argus B1/CMW and C2/TMW Release 1.2	E3	Argus Systems Group Inc.	'96. 12
Microsoft Windows NT Workstation 3.51	E3	Microsoft Ltd.	'96. 10
DEC MLS+ CMW V3.1A	E3	Digital Equipment Co Ltd.	'96. 10
Sun Trusted Solaris 1.2 ITSEC(E)	E3	Sun Microsystems Ltd.	'95. 11
VME O/S with HSO Version SV294	E3	International Computers Ltd.	'94. 9
VME O/S with GSO Version SV294	E3	Interantional Computers Ltd.	'94. 9
ICL Unix Version 7 level 5	E2	Fujitsu-ICL Computers Ltd.	'96. 1
Sun Solaris 2.4SE	E2	Sun Microsystems Ltd.	'95. 11
독 일			
Reliant UNIX 5.43 with AUDIT 2.0	E3	Siemens Nixdorf	'97. 4
AIX V4.2	E3	IBM Deutschland	'97. 4
GUARDIAN 90 Version C20 withSAFEGUARD Version C22L	E3	Tandem Computers GmbH	'93. 10
SINIX V5.42/AUDIT V1.0	E2	Siemens Nixdorf	'95. 10

7. 결 론

본 고에서는 운영체제 보안에 대한 기술 동향과 활발하게 진행중인 미국의 안전한 운영체제 현황을 분석, 소개하였다. 일반적으로 널리 사용되는 안전하지 못한 운영체제에 보안 기능을 추가하기 위한 다양한 전략이 개발 중인데 기존 응용 계층과의 호환을 최대한 수용하기 위하여 커널 계층만의 수정을 요구하는 Identical 운영체제와 운영체제까지의 변경을 요구하는 호환 운영체제, 그리고 응용 계층, 운영체제 계층, 커널 계층까지 모두 보안에 대한 요구사항을 충족하기 위해 새로 설계되는 새로운 구조의 안전한 운영체제로 나뉘어진다. 이러한 전략은 운영체제의 보안 수준에 타당하게 선택·적용되어져야 하는데 현재는 기존의 운영체제를 충분히 수용하는 Identical 운영체제 구현 전략이 많이 사용되고 있는 실정이다. 하지만 강력한 신뢰성을 보장하기 위한 새

로운 운영체제 전략이 바람직하다고 볼 수 있다.

현재 보안 분야를 선도하고 있는 미국에서는 군사 및 정부에서 안전한 컴퓨터 네트워크 사용을 목적으로 Synergy 연구 프로그램을 강력하게 추진하고 있으며 이것은 자국의 이익 및 통신 신뢰성을 극대화하기 위한 범 국가적인 일로 수행되어지고 있다.

현재 차세대 상용 운영체제에서 강력한 보안 메커니즘을 포함하도록 하기 위한 DTOS라는 프로토타입까지 개발된 상태이다. 네트워크 기술의 발달과 자국의 정보 자산 보호에 민감해지고 있는 세계적인 추세에 따라 자신의 시스템 보안에 대한 요구사항은 날로 증가하고 있다. 따라서 다양한 운영체제 보안 제품이 개발되었거나 개발 중에 있으며, 미국과 유럽 등에서 이에 대한 평가와 함께 평가된 제품을 사용하고 있다. 이처럼 안전한 운영체제 보안 시스템 개발과 이에 대한 평가는 일부 선진 국가에서만이 이루어지고 있는 실정이다. 국내의 경우 운영체제 보안에 대한 연구 개발은 미비하나 향후 중점적으로 추진되어야 할 분야이다.

참 고 문 헌

[1] 한국정보보호센터, *전산망정보보호 - 접근통제기술*, 1996. 12.

[2] Charles P. Pereeger, *Security In Computing*, Prentice Hall, 2nd Ed., 1997.

[3] National Computer Security Center, *Department of Defense Trusted Computer System Evaluation Criteria*, DOD 5200.28-STD, Dec., 1985.

[4] France, Germany, The Netherlands, and The United Kingdom, *Information Technology Security Evaluation Criteria*, Version 1.2, Jun., 1991.

[5] Morrie Gasser, *Building a Secure Computer System*, Van Nostrand Reinhold Company Inc., 1988.

[6] http://www.hpcc.gov/

[7] http://www.ccic.gov/pubs/blue97/nsa/secureos.html, Secure Operating System Development.

[8] 이철원, 홍기용, 김학범, 오경희, 심주걸, "다중등급 보안정책을 지원하는 침입차단시스템의 설계" 제7회 통신 정보 합동학술 대회(JCCI '97) 논문집, pp. 59 ~ 63, 1997. 4.

[9] C. W. Lee, K. Y. Hong, K. H. Oh, H. B. Kim, J. G. Sim, "Firewall System for Multilevel Security Environment", JWISC Fall Conference Proceedings, pp. 147 ~ 152, Oct., 1997.

[10] Secure Computing Corporation, *DTOS Lessons Learned Report*, 27. June, 1997.

[11] http://www.cs.utah.edu/~sds/synergy/microkernel/fluke/ fluke.html

[12] Gery Herman, "Operating Systems Security for midrange and Large Computer: Overview", DATAPRO, Dec. 12. 1996.

[13] http://www.radium.ncsc.mil/tpep/process/procedures.html, Overview of the Trusted product Evaluation Program(TPEP).

[14] 홍기용, "정보보호시스템 평가와 인증제도", 제2회 정보보호심포지움(SIS'97), pp. 123 ~ 169, 1997. 6. 24.

□ 著者紹介

김 학 범

1988년 2월 경기대학교 전자계산학과(학사)
1990년 2월 중앙대학교 대학원 전자계산학과(석사)
1996년 3월 - 현재 아주대학교 대학원 컴퓨터공학과 박사과정 재학중
1991년 10월 - 1996년 6월 한국전산원 주임연구원
1996년 7월 - 현재 한국정보보호센터 선임연구원

※ 주관심분야 : 컴퓨터 · 네트워크 보안, 정보보호시스템 평가체계, 정보보호 표준화

오 경 희

1988년 2월 서강대학교 전자계산학과(학사)
1992년 2월 한국과학기술원 전자계산학과(석사)
1995년 11월 CISA
1992년 10월 - 1996년 12월 한국통신 멀티미디어연구소 전임연구원
1996년 12월 - 현재 한국정보보호센터 주임연구원

※ 주관심분야 : 정보보호시스템 평가체계, 정보시스템 감사, 위험분석 및 관리

권 현 조

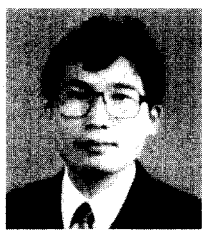
1997년 2월 성균관대학교 정보공학과(학사)
1998년 3월 - 현재 성균관대학교 정보통신대학원 재학중
1997년 1월 - 1997년 7월 (주)나라계전 연구소, 연구원
1997년 7월 - 현재 한국정보보호센터 연구원

※ 주관심분야 : 정보보호시스템 평가체계, 네트워크 보안, 전자서명

구 자 동

1996년 2월 아주대학교 컴퓨터공학과(학사)
1998년 2월 아주대학교 대학원 컴퓨터(석사)
1998년 3월 - 1998년 7월 삼성전자 정보통신 개발센타
1998년 7월 - 현재 한국정보보호센터 연구원

※ 주관심분야 : 정보보호시스템 평가체계, 이동통신 보안, 침입탐지시스템



홍 기 웅

1985년 2월 전남대학교 전자계산학과(학사)
1990년 2월 중앙대학교 대학원 전자계산과(석사)
1994년 4월 정보처리기술사
1996년 2월 아주대학교 컴퓨터공학과(박사)
1985년 9월 - 1995년 10월 한국전자통신연구소 선임연구원
1992년 - 1993년 이태리, Alenia Spazio사 Senior Researcher
1995년 10월 - 1996년 4월 한국전산원 선임연구원
1996년 4월 - 현재 한국정보보호센터 책임연구원, 평가체계팀장/기술표준팀장

※ 주관심분야 : 컴퓨터 · 네트워크 보안, 정보시스템 위험분석 · 평가, 정보보호 표준화