□특집□

분산 객체 컴퓨팅의 정보 보호 기술

김 경 범† 최 락 만†† 송 영 기††† 인 소 란††††

◈ 목 차 ◈

1. 서 론

고성능의 시스템들을 고속의 통신망으로 연결하여 활용하는 환경이 성숙됨에 따라 분산 컴퓨팅이 자연스럽게 정보 처리의 한 형태로 자리잡게 되었다. 분산 컴퓨팅은 통신망으로 연결된 시스템 자원들의 상호 협동, 성능 향상, 신뢰성과 가용성 확보, 확장성과 이식성, 성능 효과 등의 이점을 가진다고 알려져 있다. 그러나, 실제 분산 응용 개발자들은 상당한 어려움을 겪고 있다. 그 이유는 자원의 분산으로 인하여 생기는 복잡성 등을 효율적으로 해결할 수 있는 방법이 미약하기 때문이다. 그러나 객체 지향 기법을 이용할 때, 이러한 단점들을 극복할 수 있다[1].

분산 객체 컴퓨팅은 분산 시스템과 객체 지향 기법의 장점을 결합함으로써 문제를 최소화하기 위하여 나타난 형태이다. 분산 객체 컴퓨팅을 언급할 때, 객체가 무엇이며, 분산 컴퓨팅의 어떠한 기술들이 객체와 객체 사이의 상호 작용에 이용되고 여러 다른 회사 제품들 간의 동작을 어떻게 할 것인가가 중요한 문제가 된다. 이를 해결하기 위하여 표준화된 객체 모델이 있어야 하고 분산된 객체들간에 통신 메커니즘들이 만들어져야 한다.

이러한 작업들은 현재 크게 나누어 OMG(Object Management Group), Microsoft사 등에서 이루어지는데, OMG는 여러 업체들이 모여 만든 컨소시움으로 객체 모델의 표준인 CORBA(Common Object Request Broker Architecture) 규격[2]을 만들었으며, Microsoft 사는 COM(Component Object Model)이라는 독자적인 모델을 만들었다.

한편 정보 보호 기술은 이제까지 국가적인 목적으로 개발되고 활용되던 환경에서 점차로 상용으로 활용되기 시작하고 있다. 특히 인터넷의 급속한 발전과 이를 상용 업무에 활용하고자 할 경우에는 정보 보호 기술을 절대적으로 활용하여야 한다.

정보 보호의 목적은 자료의 기밀성, 무결성 등을 보장하고 인증된 사용자에게 자원을 활용할 수 있는 가용성을 제공하는 것이다. 분산 컴퓨팅 환경에서 정보 보호는 더욱 더 중요한 위치를 차

† 정회원 한국전자통신연구원 책임연구원
†† 정회원 : 한국전자통신연구원 책임연구원
††† 정회원 : 한국전자통신연구원 책임연구원
†††† 정회원 : 한국전자통신연구원 책임연구원

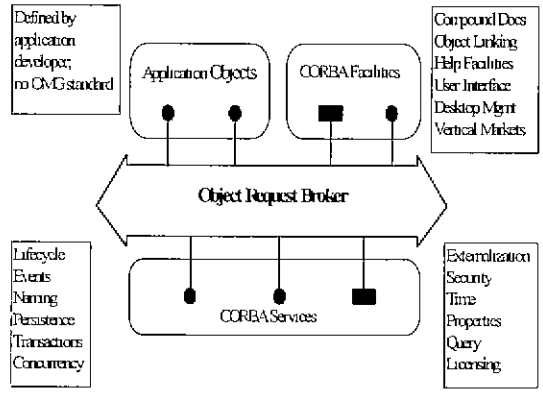
지하게 된다. 자원이 통신망 상에 분산되어 있으면서도 사용자가 원하는 일을 만족하기 위하여 자원들 간에 서로 긴밀하게 협동하여야 하기 때문이다. 또한 분산 컴퓨팅의 장점인 자원에 대한 투명성을 유지하면서도 성능을 저하시키지 않아야 하는 조건을 만족하여야 한다.

여기에서는 분산 객체 컴퓨팅과 이 환경에서 자원을 안전하게 보호할 수 있는 정보 보호 기술을 OMG에서 만든 security 규격을 기반으로 하여 살펴본다. 2절에서는 분산 객체 컴퓨팅의 커다란 흐름인 CORBA를 살펴보고, 3절에서는 정보 보호 기술의 일반적인 기술들과 이 기술들이 분산 환경에서 어떻게 변화되고 적용되어야 하는지를 설명한다. 4절에서는 현재 분산 객체 컴퓨팅 환경의 보호 기술 표준으로 제안된 CORBA security에 관하여 조사하고 5절에서는 이기종 분산 환경이 될 것으로 예상되는 초고속정보통신 기반의 정보 보호 플랫폼 기술에 관하여 소개하고 결론을 맺는다.

2. 분산 객체 컴퓨팅

OMG는 1989년에 분산 객체 컴퓨팅 환경에서 응용을 개발하기 위한 표준을 만드는 목적으로 만들어진 소프트웨어 컨소시움이다. 현재 약 700여 개의 멤버들이 참여하고 있으며, 이 멤버들은 OMG가 제기하는 RFI (Request For Information)이나 RFP (Request For Proposal)에 대하여 기술적인 응답을 하며 이를 통하여 만들어지는 규격을 OMG는 수용하게 된다. 한편 OMG는 실제 구현을 하지 않고 필요한 인터페이스만을 정의한다.

OMG는 객체 모델(Object model)과 참조 모델(Reference model)로 이루어진 OMA (Object Management Architecture)라는 구조를 제시하였다. 객체 모델은 분산 환경에서 분산되어 있는 객체들을 나타내는 방법을 설명하고, 참조 모델은 이 객체들 사이의 상호 작용을 언급하고 있다. OMA는 클라이언트와 객체들 사이에 통신을 책임지고 있는 객체 요구 중개자 (ORB: Object Request Broker), Object service, Common Facility, 그리고 기타 인터페이스로 구성되어 있다. (그림 1)은 OMA 참조 모델을 나타낸다.



(그림 1) OMA(Object Management Architecture)

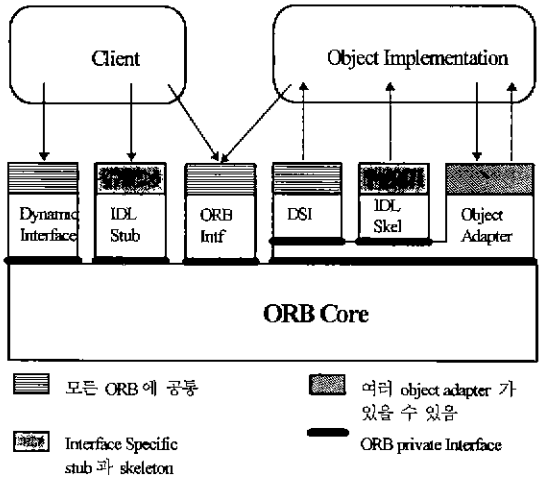
ORB는 ORB는 객체들이 투명성 있게 요구를 하고 그 응답을 받을 수 있도록 해 주는 소프트웨어 구성 요소이다. 즉, 모든 통신은 ORB를 통하여 일어나고, 객체 서비스들은 그 위치에 상관없이 서비스될 수 있다.

CORBA Service는 많은 분산 객체 프로그램들이 사용하는 인터페이스로 영역에 종속되지 않으며, 객체를 이름을 이용하여 찾을 수 있도록 해 주는 Naming 서비스, 속성에 따라 객체를 찾을 수 있도록 해 주는 Trading 서비스를 비롯하여 여러 가지 서비스들이 정의되어 있다.

CORBA Facility는 최종 사용자들이 이용할 수 있는 인터페이스들로 문서 모델에서 객체를 교환하고 표현할 수 있도록 해 주는 DDCF(Distributed Document Component Facility) 등이 있다. 응용 객체들에 관해서 OMG에서는 현재 표준화 하고

있지는 않으나 앞으로는 이 작업도 행해질 것으로 예상된다.

CORBA 규격은 OMA의 ORB 구성 요소에 대한 인터페이스와 특징들을 정의하고 있다. 현재 CORBA 2.0이 발표되어 있는 데, 그 주요 내용으로는 ORB core, OMG IDL (Interface Definition Language), Interface Repository, Language Mapping, Stub과 Skeleton, Dynamic Invocation, Object Adapter, Inter-ORB protocol 등이 있다. (그림 2)는 CORBA를 나타내는 그림이다.



(그림 2) Common Object Request Broker Architecture

현재 ORB 제품은 여러 회사에서 발표하였고 CORBA service 중 일부들을 개발하고 있다. (표 1)에서 (표 3)은 여러 제품들의 특징을 비교한 표이다.

3. 정보 보호 기술

분산 객체 컴퓨팅에서 적용되는 정보 보호 기술은 기존의 환경에서 사용하는 기술들이 적용된다. 그러나 분산 환경이 대규모의 이종 시스템들로 구성되는 환경이므로 확장성과 상호 운용성을 고려하여야 한다.

정보 보호의 기본적인 기능은 여러 가지 위협들에 대하여 자원을 보호할 수 있는 서비스를 제공하는 것이다 이 서비스를 제공하기 위하여 여러 가지 메커니즘들이 포함되어야 한다.

위협 요소들은 승인 받은 사용자인 체 하는 위장, 비인가 된 자원에 접근하려 하는 것, 자료의 수정, 수행한 동작에 대한 부인, 서비스 제공의 거부 등이다. 이러한 위협 요소들에 대하여 인증 서비스, 인가 서비스, 기밀성 서비스, 무결성 서비스, 부인 봉쇄 서비스 등이 행해진다 이 서비스들은 암호와 관련된 기술들과 접근 제어 기술을 통하여 이루어진다.

가장 중요한 서비스는 인식과 인증이다. 이는 서비스를 이용하고자 하는 주체를 검증하는 것이다. 허가는 요구자가 어느 특정한 서비스를 이용할 수 있는지를 결정하는 것이고 보안 감사는 사용자들의 동작에 대한 사항을 조사하는 것이다. 부인 봉쇄는 메시지의 발송에 대한 증명과 수신에 대한 증명으로 구분된다. 발송에 대한 부인 봉쇄는 메시지를 처음 보낸 사람의 발송 사실에 대한 부인을 막기 위해서 부인 봉쇄 서비스가 발신 측 위에 생성 증거를 만들어 두었다가 분쟁 발생시 사용하고, 수신에 대한 부인 봉쇄는 수신 사실을 부인하는 것을 막기 위해서 수신 측 위에 수신 증거를 만들어 놓았다가 이용한다. 이를 위해 하나 이상의 신뢰성 있는 제삼자(Trusted Third Parties)가 필요하다. (표 4)는 이러한 위협들과 이에 대응하는 방법들을 나타낸 것이다.

(표 1) CORBA Vendor 플랫폼

Vendor	Sol	HPUX	AIX	DEC	Linux	SGI	NT	W95	OS/2	Mac	VMS	MVS	other
Expersoft	Y	Y	Y				Y	Y					
Sun	Y						Y	Y					
IONA	Y	Y	Y	Y		Y	Y	Y	Y	Y	Y	Y	Y
Visigenic	Y	Y	Y	Y			Y	Y					Y
DEC	Y	Y	Y	Y			Y	Y			Y	Y	
ICL	Y	Y	Y	+			Y	Y	Y		Y		Y
HP	Y	Y					Y						
IBM			Y				Y	Y	Y			Y	Y
Chorus	Y				Y		Y	Y					Y
OOT	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y
Gemstone	Y	Y	Y				Y						
Prism													
Electra													
U Colorado	Y	Y		Y		Y							
Xerox	Y	Y	Y	Y	Y	Y	Y	Y					Y
BBN	Y	Y											
SNI	Y						Y	Y					
TRW	Y	Y	Y	Y		Y					Y		Y

주) (Source: http://www.vex.net/~ben/platmatrix.html)

Y: 가용 (표준) +: 곧 지원 - : 지원되지 않음

#: 표준 아님 ?: **Unknown**

(표 2) ORB Core 기능표

Vendor	IDL	C++	C	St	Ada	IIOP	DCE	DII	DSI	IR	Java	OLE
Expersoft	Y	Y	-	Y	-	Y	-	Y	?	?	-	Y
Sun	Y	Y	Y	-	-	Y	-	Y	Y	Y	Y	1Q97
IONA	Y	Y	-	Y	Y	Y	?	Y	Y	Y	Y	Y
Visigenic	Y	Y	-	Y	-	Y	-	Y	?	?	Y	?
DEC	Y	Y	Y	?	?	Y	Y	Y	?	Y	Y	Y
ICL	Y	Y	Y	-	-	Y	-	?	?	?	Y	Y
HP	Y	Y	?	?	?	Y	Y	Y	-	-	?	Y
IBM	Y	Y	Y	Y	Y	Y	?	Y	?	?	?	Y
Chorus	Y	Y	-	-	-	+	-	Y	?	?	-	?
OOT	Y	Y	Y	-	-	+	-	'97	-	-	-	-
Gemstone	Y	-	-	Y	-	Y	?	?	?	?	-	?
Prism	Y	Y	?	?	?	?	Y	?	?	?	?	?
Electra	Y	Y	-	-	-	Y	-	Y	-	-	Y	-
U Colorado	Y	Y	Y	-	Y	Y	-	-	-	-	Y	-
Xerox	Y	Y	Y	-	-	Y	-	-	-	-	+	-
BBN	Y	Y	Y	?	?	+	-	Y	?	?	?	?
SNI	Y	Y	-	-	-	Y	-	Y	Y	Y	-	-
TRW	Y	Y	?	?	Y	?	Y	?	?	?	?	?
Tandem	**IBM** 참조											
ILOG	**IONA** 참조											

(Source: http://www.vex.net/~ben/orbmatrix.html)

(표 3) CORBAservices 기능표

Vendor	Nm	Lf	Ev	Tr	Id	Ri	Cc	Ex	Po	Tx	Qr	Tm	Pr	Cm	Sc	Li
Expersoft	Y		+													
Sun	Y	Y	Y		Y	Y							Y			
IONA	Y	?	Y		?	?	?		?	Y						
Visigenic																
DEC																
ICL			Y	Y	?	?	?		?	Y					Y	
HP	Y	Y	Y													
IBM	Y	Y	Y		Y	?	Y	Y	Y	Y						
Chorus	#															
OOT	Y	+						Y							#	
Gemstone	Y	Y	Y						Y	Y						
Prism																
Electra	Y	Y	Y													
U Colorado																
Xerox	#	#														
BBN	Y	Y							Y							
SNI	Y	Y	Y		Y	Y							Y			
TRW	Y															
Tandem	IBM 참조															
ILOG	IONA 참조															

(Source: http://www.vex.net/~ben/orbmatrix.html)

Nm: Naming
ID: Identity
Po: Persistent Objects
Pr: Properties

Lf: Life cycle
Ri: Relationship
Tx: Transactions
Cm: Configuration Management

Ev: Event
Cc: Concurrency
Qr: Query
Sc: Security

Tr: Trading
Ex: Externalization
Tm: Time
Li: Licensing

(표 4) 위협과 대응 방안

위 협	대응 방안
위장(Masquerading)	인증 (Authentication)
도청(Eavesdropping)	암호화 (Encryption)
중간 개입	Digital Signature
Address Spoofing	방화벽 (Firewall)
Data Diddling	암호화한 message digest
Dictionary Attack	강력한 패스워드
Replay Attack	Time stamping, sequence numbering
서비스 거절	Authentication, service filtering

한편 암호 관련 기술은 가장 중요한 기술로 많이 연구되어 왔으며, 현재 외국에서는 수출 전략 상품의 핵심 기술로 활용하고 있다. 암호화는 자료를 적절한 복호화 키를 가지고 있지 않은 사용자들이 읽을 수 없도록 변환하는 것으로 암호/복호 방식 등은 사용자들이 선택하는 security 모델 또는 정책에 따라 달라진다.

암호 키는 자료를 보호하기 위하여 암호화나 디지털 서명 등이 사용하는 숫자로 된 값이다. 이 키를 관리하는 것은 사용자들이 요구할 때 키를 제공할 수 있게 하는 것이다. 암호 키 모델에는 대칭키 방식과 비대칭키 방식이 있다. 대칭 키 방식은 통신하고자 하는 대상자들끼리만 키를 알고 이 키

를 사용하여 암호/복호화 한다. 비대칭 키 방식은 수학적으로 연관된 한 쌍의 키를 만들어 낸 후, 하나는 공개하고 (공개 키), 하나는 개인이 간직하여 (비밀 키) 자료의 송,수신에 활용하는 것이다.

4. CORBA Security[3]

4.1 CORBA Security 규격

1989년에 OMG가 만들어져서 분산 객체 컴퓨팅에 관한 규격을 만든 후, 여러 가지 객체 서비스에 대한 연구를 계속 해 왔다. 그 중 하나가 security 서비스인데 1994년에 RFP를 만든 후, 1995년 말 1.0판을 제정하였고 1996년 7월에 이를 수용하였다. 그리고 7월에 개정 작업을 거친 후 1996년 11월에 CORBAservice로 문서화 되었다. 이 작업에 참여한 회사는 AT&T GIS Co., Digital, Expersoft, Groupe Bull, HP, IBM, ICL, Novell, Siemens Nixdorf, Sunsoft, Tandem, Tivoli system 등이다. 또한 CSI (Common Secure Interoperability) 규격은 1996년 6월에 만들어졌으며 이는 안전한 상호 운용성을 제공하기 위한 규격이다.

CORBA security는 CORBA 시스템에서 정보 보호 기능을 어떻게 제공할 것인가 하는 데 대한 인터페이스를 정의하고 그 방법을 규격화한 것이다. 이는 단순성, 분산 객체들 사이에 일관된 정보 보호 서비스의 제공, 확장성과 유용성, 적당한 정보 보호 정책을 사용할 수 있는 융통성, 응용의 이식성 유지 등을 목표로 하고 있다. 또한 특정한 정보 보호 기술에 종속되지 않도록 하고, 상호 운용성을 제공하며, 객체 지향적으로 TCSEC(Trusted Computer System Evaluation Criteria), ITSEC(Information Technology Security Evaluation Criteria)과 같은 표준 정보 보호 평가 기준을 따라야 한다고 명시하고 있다.

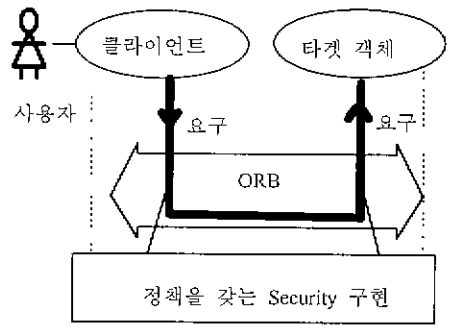
CORBA Security는 security unaware application과 security aware application을 지원할 수 있고, 부인 봉쇄, security 서비스의 대체 기능, 상호 운용성 등을 옵션으로 하고 있다.

현재 IONA, HP 등에서 CORBA 규격을 따르는 상용 ORB 제품들을 이미 판매하고 있으며 97년 상반기에는 ORB core에 security 규격에 따라 정보 보호 기능을 갖는 제품이 제공될 것으로 보인다.

4.2 Security 모델

Security 참조 모델(Security Reference Model)은 CORBA security의 전체 프레임워크를 보여주는 개념 모델로, 여러 등급의 다양한 security 정책들을 유연성 있게 정의하도록 해 주고 security 구조를 구성하는데 필요한 기본 골격을 제공해 준다.

Security 정책은 i) 객체를 접근할 수 있는 조건, ii) 사용자 또는 principal 인증에 대한 정보, iii) 객체간 통신의 안전성 품질, iv) security 관련 행위들에 대해 어떤 책임이 요구되는지 등을 정의한다. Security 정책에는 접근 제어 정책, 감사 정책, 인증 정책, security 호출 정책(Security Invocation Policy), 부인 봉쇄 정책(Non-Repudiation policy), 위임 정책(Delegation Policy) 등이 있다. (그림 3)은 security 참조 모델을 나타낸다.



(그림 3) Security 참조 모델

Principal은 시스템에 등록되어 있는 인증 가능

한 사람이나 시스템 개체를 말하며, 초기 principal은 시스템 사용 활동을 시작하는 주체로서 사용자의 패스워드나 시스템 개체의 long term key와 같은 인증 정보에 의한 방법으로 인증될 수 있다. 초기 principal은 적어도 하나 이상의 고유 식별자를 갖게 된다. 이 식별자는 principal의 활동을 추적하거나 보호되어 있는 객체에 대한 접근 권한을 부여하거나 메시지를 생성한 개체를 구별하거나 시스템을 누가 사용할 수 있는지를 구별하는데 이용된다. Principal은 어떤 객체를 접근할 수 있는지를 정의하는 데 사용되는 권한 속성을 갖고 있으며 접근 제어 정책에 따라서 다양한 속성이 가능하다.

객체 호출에 필요한 안전성 기능은 security 정책에 따라 다를 수 있는데 기본적으로 다음과 같은 기능들을 포함한다.

- 클라이언트와 타겟 객체 간에 상호 신뢰를 갖도록 security 관계를 확립
- 클라이언트가 접근 제어 정책에 의거 객체에 대한 동작 자격이 있는지 여부 결정
- 호출에 대한 감사 기능
- 보호 품질 정책에 따라 요구와 회신에 대한 변경이나 도청 방지

접근 제어에서는 다양한 접근 제어 security 정책을 반영할 수 있도록 객체 호출 접근 제어 정책과 응용 접근 제어 정책의 두 계층만을 갖는 형태의 프레임워크를 제안하였다.

객체 호출 접근 제어 정책은 클라이언트가 타겟 객체에 대한 동작을 호출할 수 있는지 여부를 판단해 주며 응용이 security 정책을 알고 있는지 여부에 관계없이 ORB와 ORB에서 제공하는 security 서비스에 의해 자동적으로 수행된다. 응용 객체에 대한 접근 제어 정책은 클라이언트 또는 타겟 객체 내의 security 정책을 잘 알고 있는 응용에 의해 수행된다. 클라이언트는 객체 접근 정책에 허용된 경우에 한해서 접근 여부 결정 기능을 통해 타겟에 있는 객체에 대한 동작을 호출할 수 있다. 클라이언트 측의 접근 여부 결정 기능은 클라이언트가 호출할 수 있는 조건을 정의하고 타겟 측의 접근 여부 결정 기능은 호출을 받아들일 수 있는 조건을 정의한다. 객체 호출을 위한 접근 정책은 접근 제어 규칙들을 접근 여부 결정 기능에 구현해 놓고, 이 기능은 principal의 권한 속성, 타겟의 제어 속성 유효 기간, 수행되고 있는 동작과 데이터, context 등과 같은 정보들을 조사하여 허용된 접근인지를 결정한다.

Security 감사는 security 관련 이벤트들의 세부 사항을 기록함으로써 안전성 침해를 찾아내는 것을 도와 준다. 감사 정책은 어떤 이벤트들을 어떤 환경하에서 감시해야 하는가를 규정하며 시스템 감사 정책과 응용 감사 정책의 두 가지 범주로 나눌 수 있다. 감사 기능은 응용 층이나 ORB에서 이루어질 수 있고 클라이언트와 타겟에서 각기 수행된다.

부인 봉쇄는 사용자나 principal들의 활동을 추적해 볼 수 있도록 해주는 서비스로서, 부인한 사건이나 행동에 대해 반박할 수 없는 증거를 생성해 주고 이들을 증명하는 일을 수행한다. 부인 봉쇄 서비스는 객체 호출 시 자동으로 수행되는 것이 아니고 응용의 제어하에 수행되며 이 서비스 내용을 잘 아는 응용에 한해 사용이 가능하다. 증거물이 유효한 것인지를 증명해 주는 신뢰성 있는 authority로부터 time stamp를 포함한 증거물을 받는 것과 같은 여러 활동이 적용된다.

ORB 상호 작용 구조에서 영역은 어떤 공통 특성이나 규칙들이 공개적으로 존재하고 이들의 영향이 미치는 구별된 범위를 말한다. security에 관련된 영역의 형태는 ORB 기술 영역과는 별도로 security의 정책, 환경, 기술 등 세가지 영역이 있다.

Security 관리는 앞에서 살펴본 영역별로 이들

내에 존재하는 객체들에 대해 수행된다. security 정책 영역에 대한 관리 기능에는 영역 자체를 생성하고 없애는 기능, 영역간의 객체 이동을 포함한 영역 내 객체들에 대한 관리 기능, 어떤 정책을 어느 영역에 적용할 것인지에 대한 세부적인 security 정책 관리 등이 있다. 본 규격은 security 정책의 관리에 초점을 맞추고 있으며 이는 일반적으로 security의 공통 기능 중의 일부로 간주된다. security 정책 관리에 대한 프레임워크와 특정 타입의 정책을 어떻게 시행할 것인가에 대한 세부 사항을 언급하고 있는데, 예를 들면 메시지에 대한 기본 보호 정도를 명시하기 위한 인터페이스, 신임장 위임 정책, 감사 이벤트 등이 기술된다. 접근 제어 정책과 같이 영역에 따라 다른 경우에는 세부적으로 정의하지 않고 대신 영역에 대한 표준 접근 정책을 수립한다.

4.3 Security 구조와 기능

CORBA Security 구조는 앞 절에서 언급한 모델을 어떻게 구현하는 가를 나타낸다.

CORBA security 모델은 여러 관점에서 볼 수 있다. 기업의 관리자는 정보 시스템을 포함한 기업 정보를 여러 가지 위협 요소들로부터 적정한 비용으로 보호하기 위한 조치를 취한다. 이를 위해 정보 자산에 대한 위험과 대응 방안에 따른 비용을 평가하여, security 관리자가 구현하고 관리할 수 있도록 security 정책을 규정한다.

최종 사용자는 시스템의 인증 대상으로 고유 identity 및 권한 속성과 일의 역할, 소속 그룹 등을 갖는다. 사용자는 자기의 업무를 수행하기 위해 객체를 호출하고, 사용자의 권한은 그가 무엇을 접근할 수 있는가를 결정하는데 사용되며, audit identity는 시스템에 행한 개인의 행위에 대한 책임성 규명에 사용된다. 사용자는 업무 수행을 위해 어떤 객체를 사용해야 하는가에 대해 구체적으로 모를 수 있다는 점이 고려되어야 한다.

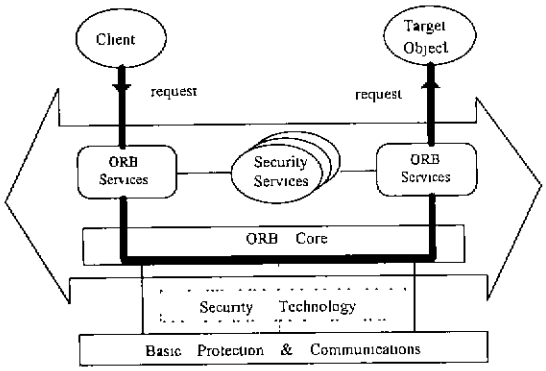
상용 업무들을 처리하는 방법에만 관심이 있는 응용 개발자에는 security unaware와 security aware형이 있을 수 있다. security unaware한 응용에 대해서는 최소한의 정보 보호 서비스가 시스템에 의해 자동으로 제공되어야 하며, security aware한 응용에 대해서는 필요에 따라 추가적인 정보 보호 조치를 취할 수 있도록 security 기능들이 제공되어야 한다. 여기에는 객체 호출 시 보호 정도의 지정, 다른 security 기능들과 독립적인 감사, 일반적인 정보 보호 인터페이스를 통한 인증, 권한 속성의 취급과 같은 서비스 등이 있다.

관리자는 영역의 생성과 관리를 담당하고, 그 영역 내에 존재하는 객체들에 적용될 security 정책의 관리와 사용자의 권한 속성 및 사용자 권리의 위임 조정 등의 역할을 수행하며, 복수의 관리자에 의한 시스템 분담 관리도 고려되어야 한다. 이 기능들은 Common Facilities의 일부로 포함할 수 있다.

Secure object system 개발자는 ORB, Object Services, Common Facilities와 이들이 필요로 하는 security 기능을 제공하는 security 서비스들을 적절히 조합하여 구성할 수 있다. ORB 구현자는 객체 호출에 ORB security 서비스를 이용하며, 객체간 interference 방지를 위해 protection boundaries 기법을 이용하기도 한다. Object Service와 Common Facilities 개발자는 특별한 security 요구 사항이 없는 경우 ORB에서 제공하는 security 서비스를 이용하여 처리할 수 있다. Security Services 구현자는 응용 측면에서 필요로 하는 security를 위해 ORB 및 기타 부가적인 security 서비스를 제공하여야 한다. 이와 같은 security 서비스는 security 정책과 여러 종류의 security 객체를 이용함으로써 제공된다. security 객체들은 필요한 모든 security 기능들을 자체적으로 제공할 수 있지만, 이미 존

재하는 외부 보안 서비스를 호출하여 제공할 수도 있다. security 객체와 외부 보안 서비스간의 연결에는 generic API가 유용하게 이용될 수 있다. CORBA Security 규격에서는 개발자, 관리자, 사용자 측면에서 이용하는 인터페이스들을 정의하고 있다.

CORBA에서 object invocation시 security 서비스를 제공하기 위한 구조 모델은 그림과 같이 응용, 특정 security 기술에 독립적인 security 서비스, 특정 security 기술 처리, 기본적인 보호와 통신 등의 요소로 구성된다. ((그림4) 참조)



(그림 4) Structural model의 구성

특정 security 기술에 독립적인 security 서비스 구성 요소는 ORB core와 ORB 서비스, security 서비스, security 정책 객체들로 구성된다.

ORB core는 객체의 기본적인 표현과 서비스 요구에 대한 기본적인 통신을 제공하는 ORB 핵심 요소로서, 클라이언트가 타겟 객체의 동작을 호출하는데 필요한 최소한의 기능을 제공한다. ORB 서비스는 ORB core에서 제공되는 기본 기능을 기반으로 하여 안전한 연계, 메시지 보호, 접근 제어 및 감사와 같은 높은 수준의 ORB 환경을 응용 구성 요소에게 제공한다.

ORB security 서비스에서는 secure invocation과 접근 제어를 처리한다. ORB security 서비스와 응용들은 인증, 접근 제어, 감사, 부인 봉쇄 및 secure invocation을 위해 Object Security Service를 부를 수 있다. 이들 Object Security Services는 security 기능 대체 옵션을 형성하며, 이들은 정보 보호 기술을 구현하기 위해 다시 외부 정보 보호 서비스들을 호출하기도 한다.

Secure 연계 설정을 위한 키 생성/관리 및 데이터 기밀성 또는 무결성을 제공을 위한 메시지 보호와 같은 서비스를 Object Security Service에게 제공하는 security기술은 기존의 security 구성 요소(외부 보안 서비스)를 사용하여 제공할 수 있다. 이러한 security 기술은 운영체제에 의해 제공 가능하지만, 다 기종 분산 환경의 수요 증대에 따라 분산화 된 형태로 제공될 수도 있다. 본 모델에서는 다양한 security 기술을 수용하고, GSS-API와 같은 generic security interface를 이용 함으로서 얻을 수 있는 장점을 최대한 활용하기 위해서 security 기술 요소를 별도의 층으로 둔다.

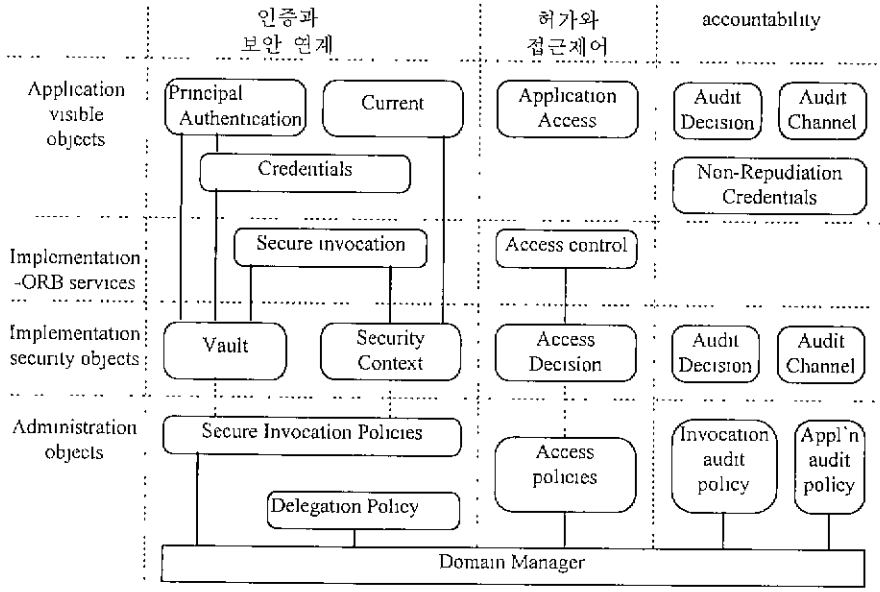
외부 보안 서비스와의 상호작용 시 표준화된 generic security interface를 사용하면 security 메커니즘의 교체가 용이할 뿐 아니라, 기존의 검증된 메커니즘들을 활용할 수 있다는 장점이 있다.

(그림 5)는 CORBA Security service를 구성하는 객체들과 기능들을 나타낸 것이다.

4.4 상호 운용성 지원

4.4.1 CORBA Security의 상호운용성 모델

분산 객체 컴퓨팅은 기본적으로 이기종 환경에서 운용되므로 상호 운용성을 지원하는 것이 대단히 중요하다. OMG에서는 상호 운용성을 제공하기 위하여 Inter-ORB Protocol을 규격화 하였다. 또한 CORBA Security 규격은 공동의 정보 보호 기술을 이용하고 CORBA 2 상호 운용성 규격을 따르는 ORB들이 안전하게 상호 운용할 수 있게 하



(그림 5) Security Object Model

는 모델과 안전성 있는 상호 운용을 위한 통신 규약에 필요한 다음 사항들을 정의하고 있다.

- CORBA 2 상호 운용 객체 참조(Interoperable Object Reference)에 있는tags의 규격
- 라이언트와 타겟 간의 정보 보호 연계 설정과 무결성과 기밀성을 유지하기 위해 CORBA 2 GIOP (General Inter-ORB Protocol) 메시지의 보호를 지원하기 위한 정보 보호 상호 운용 프로토콜
- DCE-CIOP 프로토콜을 사용할 때 정보 보호 사항

현재 규격에 있는 정보 보호 프로토콜에서는 정보 보호 연계를 설정하기 위하여 교환되는 정보 보호 토큰의 내용들과 메시지의 무결성을 위한 무결성 봉인 및 메시지의 기밀성을 위해 사용되는 암호화에 관한 세부 사항은 정의하고 있지 않다. 이런 것들은 사용 중인 특정 정보 보호 메커니즘에 따라 다르다.

한편 ORB 사이에 안전한 상호 운용을 지원하기 위하여 SECIOP (Secure Common Inter-ORB Protocol)이라는 것이 있는 데, 이는 GIOP와 IIOP 사이에 위치하여 security object context를 설정하고 메시지 교환을 보호하는 역할을 수행한다.

4.4.2 CSI (Common Secure Interoperability) [4]

CSI 규격은 GIOP/IIOP를 이용하여 상호 운용성을 제공하고 안전성을 제공하고자 하는 규격으로 표준 security 메커니즘과 이와 연관된 암호 알고리즘, SECIOP 프로토콜 메시지와 IOR security tag에 관한 상세한 사항, 상호 운용될 때 지원되는 security 기능 등을 정의하고 있다.

CSI는 세 등급으로 기능을 구분하였다. Level 0는 identity를 기본으로 하는 정책을 가지고 있으며 위임은 지원하지 않는다. 이는 기존의 시스템을 wrapping하는 데 이용될 수 있다. Level 1은 identity를 기본으로 하는 정책을 가지고 있으며 단순한 위임을 지원하며 방화벽으로 보호되는 사

무실 환경에서 활용된다. Level 2는 identity와 권한 기반의 정책을 가지고 조절된 위임을 갖는다. 이는 대규모의 기업 내 통신망에서 활용된다.

Security 메커니즘은 암호 형식, 특히 키 분배 방식에 따라 다르다. CSI 규격에서는 비밀 키, 공개 키, hybrid 키 분배 방식을 정의하고 있다. 이 방식들은 CORBA security의 기능들을 지원하는 데 이용된다.

한편 SECIOP 메시지에 들어가는 토큰을 정의하고 있는 데 SPKM(Simple Public Key Mechanism) 프로토콜, GSS Kerberos 프로토콜, CSI-ECMA 프로토콜 등의 세가지 프로토콜이 정의되어 있다.

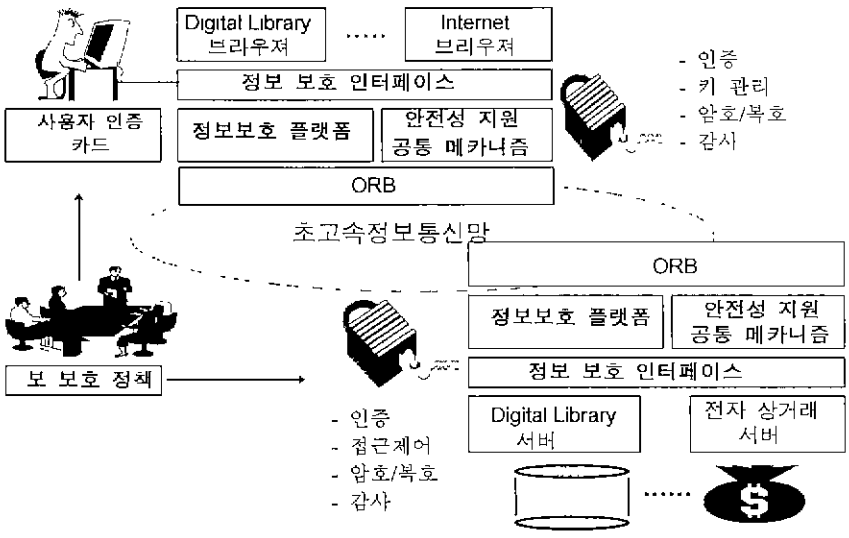
5. 정보 보호 플랫폼 기술 개발

이 절에서는 분산 객체 컴퓨팅 환경에서 정보 보호 기술의 한 예로 한국전자통신연구원에서 개발 중인 초고속정보통신 환경의 정보 보호 플랫폼에 관하여 기술한다.

5.1 초고속정보통신망과 정보 보호

디지털 라이브러리나 전자 상거래와 같은 응용 서비스들이 초고속정보통신망 위에서 제공되고 정보의 송.수신이 분산 객체 환경을 통해 이루어지는 경우, 정보의 분산 처리는 ORB를 통해 수행되고 ORB를 기반으로 한 정보 보호 플랫폼이 제공되며 정보 보호 메커니즘의 일부는 플랫폼 외부에서 지원 받는 것이 가능하다. 정보 보호 정책 관리자는 같은 정보 보호 영역 내의 제반 정보 보호 대책을 수립하고 키 분배 방식, 암호 알고리즘, 접근 권한 설정 등의 정보 보호 정책을 결정하여 제공한다. (그림 6)은 이러한 환경을 보여 주고 있다.

정보 보호 플랫폼은 초고속정보통신망을 통해 제공될 다양한 정보 통신 서비스의 안전성을 제고하기 위한 환경을 제공해주는 개방형 분산 컴퓨팅 환경에 적합한 플랫폼으로서 분산 환경에서 클라이언트와 서버간의 상호 인증, 접근 제어, 데이터 기밀성/무결성 유지, 보안 감사 등의 정보 보



(그림 6) 정보 보호 플랫폼 운용 환경

호 서비스를 지원하는 것을 목표로 한다. 이를 위해 정보 보호 플랫폼이 갖추어야 하는 기능 및 구조상의 요구 사항을 분석하고, CORBA, SESAME, POSIX, DCE 등 분산 시스템의 정보 보호 규격 및 서비스를 비교 검토한 후, 안전한 초고속정보통신망 구축에 적합한 참조 모델을 선정하여 정보 보호 플랫폼의 기본 구조를 작성하였다.

정보 보호 플랫폼은 구조적 측면에서 정보 보호 통제 경로를 우회할 수 없도록 강제성을 가져야 하며, 시스템 크기나 정보 보호 정책 영역이 다른 경우에도 응용이 가능하도록 확장성과 상호 연동성이 있어야 한다. 또한 기업마다 원하는 정보 보호 수준을 쉽게 설정할 수 있도록 융통성을 제공하고 시스템에 따라 최적의 정보 보호 메커니즘이나 알고리즘을 선택하여 사용할 수 있도록 정보 보호 메커니즘으로부터 독립성을 가져야 한다. 아울러 플랫폼은 향후 정보 보호 기술의 발전 추세를 고려하여 유지 보수성과 재사용성이 높아야 한다[5].

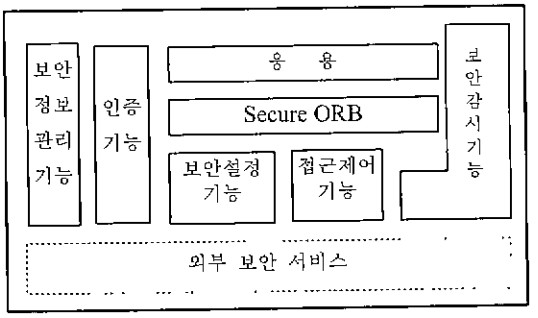
정보 보호 플랫폼이 제공해야 하는 기본적인 기능으로는 접근 주체(사용자, 시스템, 응용 객체 등)에 대한 식별과 인증 기능, 데이터나 오퍼레이션에 대한 사용 권한 판단 기능, 데이터의 기밀성과 무결성 유지 기능, 정보 보호 관련 행위에 대한 추적과 책임 규명을 위한 보안 감사 기능, 정보 보호 정책의 설정과 정보 보호 정보의 관리를 위한 정보 보호 관리 기능 등이 있다.

구조적인 면의 특성을 살펴 보면, CORBA는 구조적 모델과 객체 모델을 기반으로 한 정보 보호 구조를 제시하고 ORB에서 반드시 정보 보호 검사를 거치도록 하였으며 객체 단위로 교체가 가능토록 하여 외부 기능을 유연성 있게 활용할 수 있도록 하였다. DCE는 인증, 접근 제어 중심의 정보 보호 서버 구조를 갖고 있으며, POSIX는 정보 보호 서비스 중심의 계층적 프레임워크를 제시함으로써 범용성을 강조하고 있으며, SESAME는 Kerberos의 키 메커니즘을 반영한 정보 보호 서버 중심의 구조를 갖고 있다.

CORBA는 객체 지향 개념을 기반으로 하고 있기 때문에 다른 분산 시스템에 비해 확장성이 우수하고 적용하고자 하는 응용 시스템이 계속 변하거나 확장될 경우에 유리하며 정보 보호 모델이 특정 메커니즘이나 알고리즘에 종속되지 않도록 되어 있어 범용성이 뛰어나다. 따라서 초고속정보통신망의 정보 보호 플랫폼 기반 구조로 CORBA를 선택하였다.

5.2 플랫폼 구조 및 기능

정보 보호 플랫폼은 Secure ORB를 중심으로 응용에게 정보 보호 서비스를 제공하기 위한 인증, 정보 보호 설정, 접근 제어, 보안 감사, 정보 보호 정보 관리와 같은 기능들을 수행하며 (그림 7)과 같이 구성된다. 각 기능들은 필요에 따라 기술 종속적인 하위 정보 보호 기능을 처리하기 위해 외부 정보 보호 서비스를 이용할 수 있다[6].



(그림 7) 정보 보호 플랫폼의 논리적 구조

인증 기능은 접근 주체의 신분을 확인하고, 인증 결과로 접근 주체의 역할에 따라 해당하는 인증서를 만들어 준다. Secure ORB는 클라이언트와 서버간의 서비스 요구 및 응답을 정보 보호 정책에 따라 안전하게 전달하기 위한 기본 메커니즘을 제공한다. Secure ORB는 ORB core와 정보

보호 문맥 설정 및 서비스 요구문에 대한 암호화를 처리하는 메커니즘인 안전 호출 인터셉터와 서비스 요구 시 접근 권한 보유 여부를 검사하는 메커니즘인 접근 제어 인터셉터로 구성된다. 실제 정보 보호 서비스는 이들 인터셉터를 통해 정보 보호 설정 기능과 접근 제어 기능 등 서비스의 종류에 따라 해당 기능을 호출함으로써 실현된다.

보안 설정 기능은 클라이언트와 서버간에 안전한 정보교환을 위해 정보 보호 문맥을 설정하고, 정보 보호 문맥을 기반으로 클라이언트와 서버간에 주고 받는 내용의 무결성 또는 기밀성을 유지하기 위한 암호/복호화 등을 처리한다. 접근 제어 기능은 개시자가 타겟의 특정 오퍼레이션이나 기능에 접근할 수 있는지를 접근 제어 규칙과 입력 데이터(개시자의 권한 속성, 타겟의 제어 속성, 행위 및 문맥 정보)를 이용하여 판정하여 접근 허용 여부를 결정한다. 보안 정보 관리 기능은 사용자의 인증 정보 및 보안 속성 정보, 권한 속성 정보, 보안 영역에 있는 모든 객체들에 적용되는 보안 정책 정보 등을 추가 삭제 변경한다. 보안 감사 기능은 시스템의 안전성을 저해할 수 있는 이벤트가 발생하면 책임 규명을 위해 그 세부 내용에 대한 기록을 유지하고, 필요에 따라 시스템 관리자에게 통보하기도 한다. 외부 보안 서비스는 정보 보호 관련 기술 종속적인 부분으로, 인증, 보안 설정, 접근 제어, 보안 감사 등 각 기능에 대해 하위 수준의 정보 보호 서비스를 제공해 준다.

정보 보호 플랫폼은 클라이언트와 서버 형태를 취한다. 클라이언트 머신은 클라이언트의 정보 보호 정책에 따라 사용자에 대한 인증과 서버에 대한 서비스 요구를 안전하게 처리한다. 클라이언트 머신은 secure ORB를 중심으로 클라이언트 응용, 인증 기능, 정보 보호 설정 기능, 정보 보호 캐쉬, 외부 정보 보호 서비스 접속 기능 등으로 구성된다. 서버 측 정보 보호 플랫폼은 서버의 정보 보호 정책에 따라 서버 측 정보 보호 문맥을 만든 후 클라이언트가 서비스를 요구할 권한을 가지고 있는지 판단하고, 권한을 가지고 있으면 서비스 요구를 처리하여 그 결과를 클라이언트에게 안전하게 전해준다. 서버 측 정보 보호 플랫폼은 secure ORB를 중심으로 타겟 응용, 보안 설정 기능, 접근 제어 기능, 응용 정보 보호 기능, 정보 보호 캐쉬, 외부 보안 서비스 접속 기능 등으로 구성된다. 클라이언트와 서버 머신을 구성하는 인증 기능과 보안 설정 기능, 접근 제어 기능, 응용 정보 보호 기능 등은 필요에 따라 외부 정보 보호 서비스를 사용할 수 있다.

6. 결 론

이제까지 앞으로 중요한 정보 처리 형태의 하나로 주목되는 분산 객체 컴퓨팅 기술과 이러한 환경에서 필요한 정보 보호 기술에 대하여 살펴보았다. 그리고 현재 진행 중인 연구 개발 과제의 내용을 소개함으로써 정보 보호 기술을 활용할 수 있는 예로 삼는 데 도움을 주고자 하였다.

분산 객체 컴퓨팅은 21세기에 중요한 정보 처리 형태로 주목 받고 있다. 이는 분산 시스템의 장점과 객체 지향 기법의 장점을 결합하는 것으로 응용 서비스 개발자들의 부담을 덜어주고 재사용을 가능하게 한다.

정보 보호 기술은 정보 처리에서 필수적인 기술로 사용자들이 이용하는 정보의 안전성을 확보함으로써 정보의 신뢰성을 높이는 기반 기술이다. 분산 환경에서 이 두 기술의 결합은 필수적이며, 또한 앞으로 미래도 밝다고 볼 수 있다. 또한 정보 보호 기술은 현재 외국에서도 전략적으로 이용하고 있어 우리가 독자적인 기술을 확보하여 활용하는 것이 중요하다.

본 논문에서 기술한 분산 객체 환경의 정보 보호 기술은 기본적으로 OMG에서 표준화한 CORBA Security규격을 따른다. 이 규격은 인터페이스들을 정의하고 있으며 상세한 메커니즘들은 시스템을 구성하는 개발자들에게 맡겨 놓았다. 따라서 사용자, 개발자들은 각자의 요구 사항에 맞는 정보 보호 메커니즘들을 선택하여 규격에 명시되어 있는 인터페이스들과 결합함으로써 분산 객체 환경에서도 정보 보호 기능을 충분히 활용할 수 있을 것이다.

참고문헌

[1] Steve Vinoski, CORBA: Integrating Diverse Applications Within Distributed Heterogeneous Environments, IEEE Communications Magazine, Vol.14, No.2, February 1997.

[2] Object Management Group, *The Common Object Request Broker: Architecture and Specification*, 2.0 ed., July 1995.

[3] Object Management Group, *CORBA Security Specification*, December 1995.

[4] Object Management Group, *Common Secure Interoperability Specification*, June 1996.

[5] 최락만, 송영기, 김경범, "초고속 정보통신서비스용 보안 플랫폼의 요구 분석", 한국통신정보보호학회 학술 대회, pp.305-pp.314, 1996.11.

[6] 김경범, "초고속정보통신기반 안전성 기술 개발", HSN97, pp.29-pp.37, 1997. 1.

김 경 범

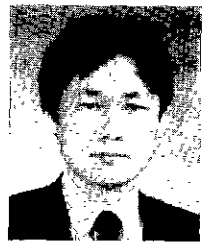
1981년 인하대학교 전자공학과 졸업 (학사)
1983년 인하대학교 대학원 전지공학과 졸업 (석사)
1983년-현재 한국전자통신연구원 책임연구원, 전자계산기 기술사

관심분야 : 정보통신 시큐리티, 분산시스템, 컴퓨터통신

최 락 만

1977년 한양대학교 전자공학과 졸업 (학사)
1987년 한양대학교 산업대학원 전산학과 졸업 (석사)
1977년-현재 한국전자통신연구원 책임연구원

관심분야 : 정보통신 시큐리티, 소프트웨어 공학, 멀티미디어



송 영 기

1977년 서울대학교 계산통계학과 졸업 (학사)
1981년 한국과학기술원 전산학과 졸업 (석사)
1977년-1985년 국방과학연구소 선임연구원
1985년-현재 한국전자통신연구원 책임연구원
1993년-1994년 미국 텍사스 알링톤 대학 방문 연구원

관심분야 : 정보통신 시큐리티, 소프트웨어 공학, 객체지향 소프트웨어 개발환경

인 소 란

1978년 홍익대학교 전산학과 졸업 (학사)
1982년 홍익대학교 대학원 전산학과 졸업 (석사)
1991년 홍익대학교 대학원 전산학과 졸업 (박사)

1978년-현재 한국전자통신연구원 책임연구원, 소프트웨어 공학 연구실장

관심분야 : 소프트웨어 공학, 정보 보안, 프로토콜 공학, 분산시스템