유비쿼터스 환경의 데이터베이스 보안을 위한 CSS 설계

이대식* · 윤동식* · 안희학**

요 약

인터넷의 보급과 다운사이징, SI 기법이 등장하면서 기존의 집중식 컴퓨팅은 급격히 분산 컴퓨팅으로 변하고 있다. 또한 분산 컴퓨팅은 유선으로 연결된 네트워크에서 벗어나 유비쿼터스 컴퓨팅으로 빠르게 변화하고 있다. 점차 복잡해지는 이기종 환경에서 응용 프로그램과 운영체제 간에 원만한 통신을 이룰 수 있게 하는 미들웨어로 CORBA가 널리 사용되고 있다. 그러나 지능적이고 다양화되는 공격들(해커, 바이러스, 웜 등) 속에서 분산처리 환경은 절대 안전할 수 없는 것이 현실이다. 본 논문에서는 OMG에서 제시한 CSS를 기반으로 유비쿼터스 환경에 적합한 DB보안 모델을 설계하고 기존 모델과 비교 분석하여 효율성을 제시하고자 한다.

A Design CORBA Security Service for DataBase Security in Ubiquitous Computing

Dae Sik Lee* · Dong Sic Yun* · Heui-Hak Ahn**

ABSTRACT

The spread of Internet and the appear of Downsizing, SI(System Integration) is chaning centralized computing to distributed computing. Also distributed computing is rapidly changing to Ubiquitous computing escape from hard wire connected network. CORBA(Common Object Request Broker Architecture) is a middleware that used for smoothness communication between application program and operation system in a different environment. However distributed computing environment is not safe from the danger, the attack like virus, worm is too intellectual and variety. In this paper, we design a new DB security model and suggest efficiency of it in Ubiquitous environment base on CSS(CORBA Security Service) that present ed from OMG(Object Management Group).

Key words : Ubiquitous Environment, CORBA(Common Object Request Broker Architecture)

* 안동과학대학 사이버테러대응학과

** 관동대학교 컴퓨터학부

1. 서 론

최근 유비쿼터스 환경 구축의 핵심 인프라인 RFID의 도입이 전세계적으로 확산되고 있다. RFID란 초소형 반도체에 식별정보를 넣어 무선 주파수를 이용해 이 칩을 지닌 물체나 동물, 사람 등을 판독, 추적, 관리할 수 있는 기술로, 주차관리, 고속도로 요금징수, 출입 통제, 원격 제어, 재고 관리 등 다양한 분야에 적용이 가능하다[1].

많은 회사들의 개발 참여와 RFID시스템의 판매실적은 이 시장의 중요성을 상기시켜준다. 2000년 전세계 RFID시스템 판매는 약 9억 US$에 이르며 2005년에는 26억 5천만 US$에 이를 것으로 전망하고 있다(Krebs, n.d.). 이것으로 볼때 RFID시장은 이동전화기 및 무선전화기 시장을 포함한 무선기술 분야에서 가장 빨리 성장하는 분야에 해당한다고 할 수 있다[2-4].

RFID 솔루션 형태는 RFID를 미들웨어에서 관리하고 데이터베이스관리시스템(DBMS)과 연동하는 솔루션이 일반적이다[5].

RFID는 원천적으로 보안에 강한 특성이 있다. 그러나 Tag정보를 저장하고 관리하는 DB(서버)는 다양한 취약점을 가질 수 밖에 없다.

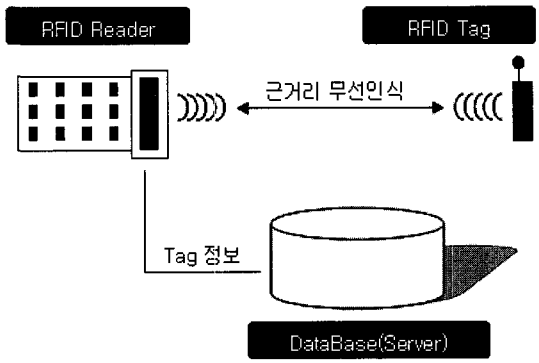
본 논문에서는 유비쿼터스 환경에서 CORBA를 기반으로 자동인식시스템[6-10], 특히 RFID의 Tag정보가 저장되는 DB의 보안을 유지하기 위해 보안 정보를 관리하는 객체를 제안하고 이를 설계하였다.

2. 관련 연구

2.1 유비쿼터스 핵심기술

2.1.1 RFID(Radio Frequency IDentification)

RFID는 (그림 1)과 같이 리더를 통하여 무선 통신에 의해서 접촉하지 않고 태그의 정보를 판독하거나 기록하는 일종의 무선 통신 시스템으로, 크게 안테나가 포함된 리더, 무선 자원을 송수신할 수 있는 안테나, 정보를 저장하고 프로토콜로 데이터를 교환하는 태그(Tag), 서버로 구성된다[4].



(그림 1) RFID 시스템 구성도

RFID는 이동 중에도 인식이 가능하고 장애물의 투과 기능도 가지고 있을 뿐만 아니라, 여러 개의 태그를 동시에 인식할 수 있고, 데이터의 인식속도도 타 매체에 비해 빠른 장점이 있다. RFID가 보편화되기 위한 요구조건으로는 낮은 비용의 생산과, 빠른 인식 속도, 다중 태그의 인식 등의 시스템적 측면과 프라이버시 보호라는 안정성 측면이 동시에 만족되어져야 한다.

2.1.2 바코드 시스템

바코드는 지난 20년간 다른 인식시스템에 비해 성공적으로 고유한 영역을 확보해 왔다. 검은색 바와 흰색 간격이 평행으로 배열된 이진 코드로써 바의 그룹은 미리 정해진 패턴에 따라 배열되며, 지정된 기호로 된 데이터 형태로 표현된다. 넓고 좁은 바가 간격을 두고 있는 연속적인 형태는 숫자와 알파벳으로 해석된다. 바코드는 광학 레이저 스캐너로 판독할 수 있다.

(그림 2) 바코드

2.1.3 광학 문자 인식

OCR(Optical Character Recognition)은 1960년대 최초로 개발되었고, 특수 글자체를 사용하며, 제조, 서비스 및 행정적인 분야, 은행 수표 등에 사용된다.

2.1.4 생체 인식(Biometrics)

체인식은 살아 있는 생명의 신체 측정 절차 혹은 셈(counting)의 과학으로 정의되고 있다. 인식 시스템 맥락으로 볼 때, 생체인식은 비교적 확실하고 개인별 물리적 특성을 확인하는 일반적인 절차에 사용되는 용어이다. 일반적으로 지문(fingerprinting) 및 장문(handprinting), 음성인식 그리고 비교적 덜 알려진 망막(혹은 홍채) 인식이 이에 해당된다.

2.1.5 스마트 카드

스마트 카드는 보조의 컴퓨팅 용량을 가질 수 있는 전자적 저장 장치이며, 편의상 신용카드 크기의 플라스틱 카드에 제작된다. 최초의 스마트 카드는 1984년 선불 전화카드의 형태로 출시되었다. 스마트 카드는 접촉 스프링에 의해 스마트 카드의 접촉면에 전기적으로 연결되는 리더에 가까이 접촉시킨다. 스마트 카드의 접촉 면을 통하여 리더로부터 에너지와 클럭 펄스를 공급 받는다. 리더와 카드 사이의 데이터 이동은 양방향 병렬 인터페이스(I/O port)를 사용한다. 스마트 카드 내부의 기능에 따라 메모리 카드, 마이크로프로세서 카드로 나눌 수 있다.

스마트 카드의 장점은 데이터를 카드 내에 저장하고 의도하지 않은 접근이나 조작되는 것을 보호할 수 있다.

접촉 기반의 스마트 카드의 단점은 접촉 부분이 마모되고 부식되며 오염되기 쉽다.

2.2 CORBA Security Service

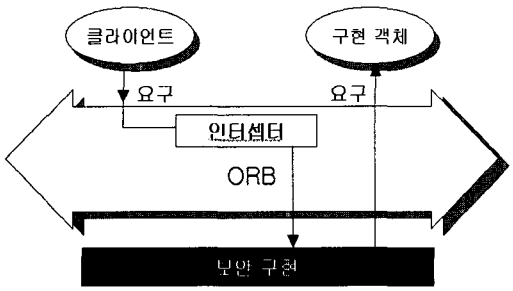
2.2.1 개요

OMG에 의해서 표준으로 정해진 보안 참조 모델은 전체적인 CORBA 보안 프레임워크를 제공하며 사용자가 적절한 수준의 보안 정책을 결정할 수 있게 함으로써 원하는 시스템을 설계할 수 있게 한다.

2.2.2 Security Reference Model

Security Reference Model은 CORBA Security 정책을 어디서, 어떻게 수행하는가를 기술한다. Security 정책은 첫째 객체를 접근할 수 있는 조건, 둘째 사용자 또는 Principal이 누구이며 허용된 일이 무엇이고 그들의 권한을 위임할 수 있는지 여부를 보여주어 행위들에 대해 어떤 책임이 요구되는지 등을 정의한다.

CORBA를 기반으로 한 객체 시스템의 Security Model은 (그림 3)에서 보는 바와 같이 나타낼 수 있다. 모든 객체 접근은 접근 제어 등의 정책을 구현한 보안 구현을 통해서 이루어진다. 이때, ORB를 통해서 전달되는 객체의 요구를 인터셉터(Interceptor)가 가로채서 보안 기능을 거쳐서 서버에 전달되게 한다. CORBA 표준에 정의된 것처럼, 이들 보안 기능들은 그 자체가 도중에 불법으로 변경되거나 절취 되지 않도록 안전성을 유지해야 하며, Security 정책에 의해 요구되는 것들을 정확하게 수행하여야 한다.



(그림 3) 객체 시스템을 위한 Security Model

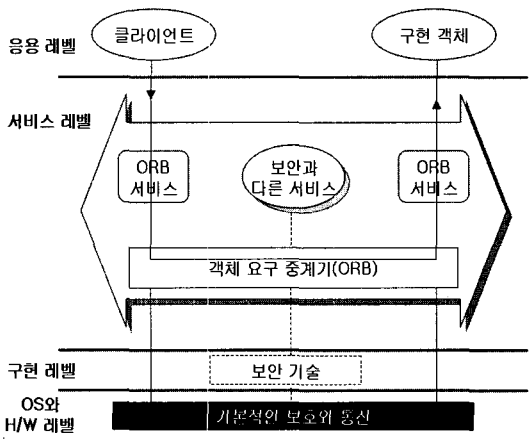
대부분의 응용 객체들은 Security 정책이 어떻게 되어 있는지 또는 내부적으로 어떻게 처리되는지 알지 못한다. 사용자는 응용 클라이언트를 부르기 전에 사용자로서의 자격을 인증 받고, 계속해서 Security 서비스가 자동으로 수행된다. 어떤 응용 객체들은 시스템이 제공하는 Security 정책의 통제를 받고 있지만 그 자신이 Security 기능을 수행하지는 않으며, 어떤 응용 객체는 자신이 정한 Security 정책들을 수행한다.

Security Model은 일반적으로 Security 정책들의 특정 집합을 정의한다. OMA(Object Management Architecture)는 서로 다른 시장의 요구를 만족하는 다양한 Security 정책을 광범위하게 지원해야 하므로 단일 Security Model의 제안을 적합하지 못하고 많은 종류의 정책들을 만들어 낼 수 있는 기본 골격(Framework)을 제공하는 Security Model(또는 Meta-policy)을 정의하는 것이 필요하다. Meta-policy는 Security 구조에서 제공하는 추상화 된 Interface와 가능한 Security 기능들을 정의하고 유연성 있는 지침을 제공하고 있다.

2.2.3 구조적 모델

구조적 모델은 클라이언트에서 구현 객체 접근까지의 주요한 단계를 말한다. 아래와 같이 4개의 레벨로 구성되어 있다.

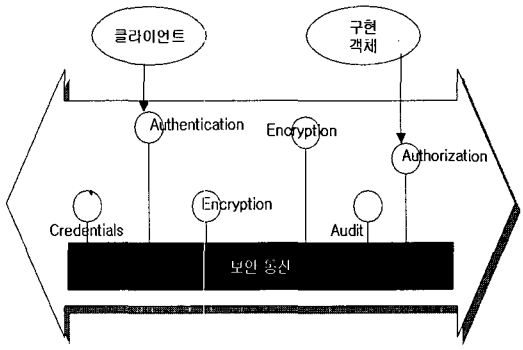
- 응용레벨
- 서비스 레벨
- 구현 레벨
- 운영체제와 하드웨어 레벨



(그림 4) 구조적 모델

2.2.4 CORBA에서의 보안 서비스

CORBA에서 제공하는 객체 보안 서비스(Object Security Service)는 별도의 외부 보안 응용프로그램 없이 ORB 자체에서 보안 서비스를 제공해 준다. 따라서 외부 보안 관리 애플리케이션과의 통신이 없으므로 성능의 향상을 기대할 수 있다.



(그림 5) ORB 내에 구축된 CORBA 보안

3. 새로운 모델 설계

제안하고자 하는 모델은 DB보안을 위한 CORBA Security Service 체제의 정보 관리 객체를 생성하여 안정적으로 운영하기 위한 것이다.

3.1 네트워크 관리 중계 프로토콜

분산컴퓨팅이 진화된 형태인 유비쿼터스 환경 또한 중계 네트워크 관리 프로토콜이 요구되며, 복제된 객체 정보를 패킷으로 묶어 효율적으로 네트워크와 네트워크를 연결할 수 있는 중계 프로토콜을 제안한다.

유비쿼터스 환경 하에서의 네트워크 관리 트래픽이 발생되는 절차는 관리 시스템이 관리 정보를 폴링하고 피 관리 시스템이 이에 응답하는 경우이거나 관리 시스템이 피 관리 시스템의 환경 변수 등을 변경하기 위해서 명령을 보내는 경우이다.

네트워크 관리 트래픽을 효율적으로 처리하기 위해 네트워크의 관리 방식에 의해 발생되는 트래픽의 양을 적정 수준으로 억제하거나 일반 데이터의 발생상태에 따라 조절하는 것이다. 이러한 상황에서 관리 트래픽이 발생되고 전송되는 과정의 특성을 분석하여 관리시스템과 피 관리 시스템 사이에서 관리 정보를 좀더 효율적으로 전달할 수 있는 프로토콜을 사용한다면 더 많은 관리 트래픽을 안정적으로 전송할 수 있게 된다.

3.1.1 정보 관리

네트워크 관리는 이미 개별적인 네트워크 단위의 관리 수준에서 벗어나 여러 회사에서 제조 판매한 여러 종류의 프로토콜, 매체, 통신장비, 응용프로그램들의 네트워크 자원이다.

인터넷의 네트워크 관리는 네트워크 관리 스테이션들이 네트워크 요소들에게 관리 정보를 질의하는 분산 모델이라고 정의할 수 있다. 네트워크 요소를 에이전트(Agent) 또는 피 관리 시스템이라고 한다.

네트워크 관리 스테이션은 먼저 피 관리 노드에 요구 신호를 보낸다. 피 관리 노드는 받은 요청 신호를 SNMP 에이전트에서 처리하여 응답 신호를 네트워크 관리 스테이션에 전해주게 된다.

3.1.2 정보 네트워크 관리

관리 행위를 수행하기 위해서는 먼저 어떤 것을 관리의 대상으로 삼을 것인지 결정하여야 하며 그러한 것을 관리 객체라 한다. 또한 각 관리 객체들에 대해서 다음과 같은 일련의 사항들을 부가적으로 정의하여야 하는데, 이러한 과정을 관리 정보 모델링이라 한다.

첫째, 관리 객체를 어떻게 표현할 것인가?
둘째, 관리 객체를 어떻게 식별할 것인가?
셋째, 관리 객체가 순간 순간 갖고 있는 값을 원격지의 관리 수행자에게 어떤 형태로 전송할 것인가?
넷째, 관리 객체에 대해서 관리 수행자가 어떤 동작을 수행할 수 있는가? 등을 정의하여야 한다.

한 마디로 말해서 관리 수행자에게 효율적인 네트워크 관리를 위해서 취할 수 있는 동작을 제공하는 것이다.

3.1.3 정보 관리 교환 프로토콜

네트워크 관리 프로토콜은 관리 수행자가 시작시키는 네트워크 관리 클라이언트 프로그램과 호스트나 게이트웨이 상에서 수행 중인 네트워크 관리 서버 프로그램간의 통신을 정의한다. 이러한 정의에는 교환되는 메시지의 의미가 포함되며 메시지 이름, 메시지 값을 표현하는 방법 등이 포함된다. 관리 수행자를 인증하기 위해서 두 통신자간의 행정적인 관계를 정의하고 있기도 하다. TCP/IP 인터넷 프로토콜 집합 내에서 SNMP의 관계를 나타낸다.

3.1.4 중계 프로토콜의 설계

중계 프로토콜의 동작 과정, 패킷 형식, 상태 천이도, 프리미티브 등의 설계와 관한 설명과 구현된 프로세스들의 연관 관계, 인터페이스, 접속 설정 및 데이터 전송의 중계 서비스를 설계한다.

• 프로토콜의 동작 과정

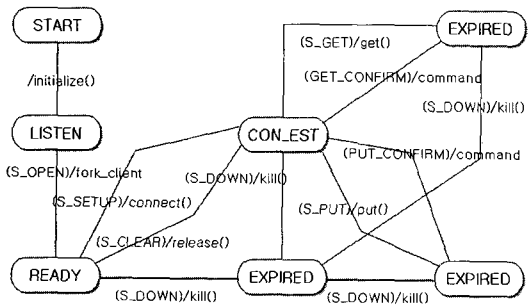
중계 프로토콜의 주요한 특징은 링 형태를 갖는 다중 접속을 구성하고 그러한 접속을 통해서 모든 관리 데이터를 주고받는 점이다.

중계 프로토콜은 이러한 접속을 설정하기 위한 접속 설정 동작, 설정된 접속을 통해서 데이터를 송·수신하는 동작, 설정된 접속을 해제하는 접속해제 동작, 접속 설정 과정이나 데이터 송·수신시에 발생하는 오류를 보고하는 오류보고 동작 등의 동작들로 이루어진다.

• 프로토콜 유한 상태 천이도

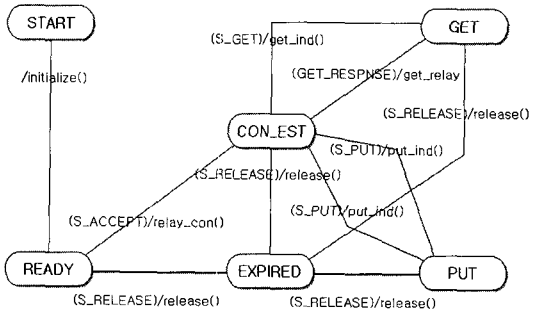
프로토콜의 동작 수행 과정을 유한 상태도를 이용하여 정의하여 보았다. 송신자 역할을 수행하는 부분과 수신자 역할을 수행하는 부분을 나누어 정의하였으며, 각 상태 천이도는 아래 그림과 같다. 각 상태 천이도와 함께 천이 조건을 명시하였으며 천이 할 때 수행하는 출력 사건도 정의하였다. 천이 조건은 조건 만족시 다음 상태로 천이하는 과정에서 수행하여야 할 동작을 기술한다.

• 송신자의 상태 천이



(그림 6) 상태천이도

• 수신자의 상태 천이

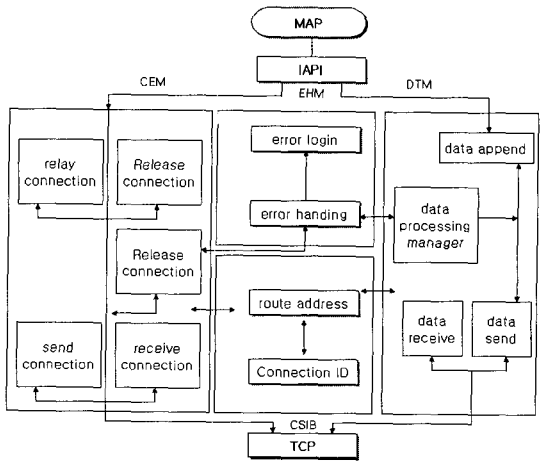


(그림 7) 상태천이도

• 프로토콜 구조도

중계 프로토콜의 전반적인 구조를 아래 (그림 8)과 같이 중계 프로토콜과 응용 프로그램간의 인터페이스는 응용 프로그램의 구현 시 간편한 함수를 제공하는 것이다.

파라미터를 포함하는 프리미티브만을 발행함으로써 응용 프로그램은 서비스의 시작과 끝을 간단히 제어할 수 있다.



(그림 8) protocol structure

접속 설정 모듈은 설정된 접속을 관리하며, 최종적으로 접속을 해제한다. 접속이 설정된 후에 개시되는 데이터 전송 모듈은 관리자에 의해 사

용되어 에이전트들을 폴링하여 관리 데이터를 수집한다.

중계 프로토콜은 접속 설정과 데이터 전송 제어를 포함하고 있기 때문에, 중계를 이용하는 응용 프로그래머들은 그러한 부분을 신경 쓸 필요가 없다. TCP/IP 프로토콜 집합에는 이러한 기능에 관한 표준이 정해져 있지 않기 때문에 상당히 유용하다.

3.2 객체용 프리미티브

프로세스들 간의 메시지 전달의 몇가지 형태는 현재 많은 운영체제의 일부분이 되었으며 몇몇 운영체제에서는 한 프로세스가 다른 특정한 프로세스에게만 메시지를 보낼 수 있도록 메시지 전달을 제한하고 있으나 유닉스 시스템에서는 이러한 제한이 없다. 유닉스 시스템의 메시지 구현방법은 모든 메시지는 커널(Kernel)속에 저장되며, 이와 관련된 메시지 큐 식별자를 가진다.

프로세스들은 임의의 큐에 메시지를 읽고 쓸 수 있고 어떤 프로세스가 큐에 메시지를 쓰기 전에 이 메시지가 도착하기를 기다리는 프로세스가 존재해야 할 필요는 없다. 특히 어떤 프로세스가 큐에 몇 가지 메시지를 쓰고 난 후에 종료해 버리고, 나중에 다른 프로세스가 그 메시지를 읽는 것이 가능하다. 큐 내의 모든 메시지는 형(type) 길이(length) 데이터속성을 가진다.

4. 결 론

본 논문에서는 유비쿼터스 컴퓨팅 환경의 복잡한 이기종(異機種) 환경에서 응용 프로그램과 운영환경간에 원만한 통신을 이룰 수 있게 하는 CORBA Security Service를 이용하여 지능적이고 다양화되는 위험속에서 안정된 컴퓨팅 환경을 설계하였다. 제안하는 메카니즘은 ORB 오퍼레이션, 송신자·수신자와 IDL을 이용한 객체 갱신에 필요한 메소드들이다. 이들은 통신상에서 발생되어지는 병목 현상과 자원의 복제에서 발생되는 오버 헤드를 줄일 수 있는 시스템으로 거듭날 수 있을 것이다.

참 고 문 헌

[1] Klaus Finkenzeller, "RFID-Handbook, 2nd Edition", Wiley & Sons LTD, April 2003.

[2] 이용주 외, "CCCA를 이용한 CORBA기반의 상호 인증 메커니즘", 정보처리학회논문지, Vol. 8, No. 3, 2001.

[3] 표철식 외, "RFID 기술 및 표준화 동향", TTA저널 제95호, 2004.

[4] 최재귀 외, "효율적인 식별 기능을 가진 위조 불가 RFID Tag 가변 ID 방식", 정보처리학회논문지, Vol. 11, No. 4, 2004.

[5] Kathleen Milsted, "Middleware Paradigms for Ubiquitous Computing", France Telecom R&D, 2003.

[6] Object Management Group, "The Common Object Request Broker Architecture and Specification(Revision 2.2)", 1998.

[7] R. Orfali, D. Harkey, "Client/Server Programming with Java and CORBA", Wiley & Sons LTD, 1998.

[8] R. Orfali, D. Harkey, and J. Edwards, "Instant CORBA", New York: Wiley, 1997.

[9] Ryu ki-young, "The Design and Implementation of Security Information Management Object for CORBA Security Service", 충남대 대학원, 2002.

[10] 왕창종 외, "Inside CORBA3 프로그래밍", 대림출판사, 1999.

이 대 식

1995년 관동대학교 전자계산공학과(공학사)
1999년 관동대학교 전자계산공학과(공학석사)
2004년 관동대학교 전자계산공학과(공학박사)
2003년~현재 안동과학대학 사이버테러대응학과 교수

윤 동 식

1992년 관동대학교 전자계산학과(공학사)
1994년 관동대학교 컴퓨터공학과(공학석사)
2000년 관동대학교 컴퓨터공학부(공학박사)
1999년~현재 안동과학대학 사이버테러대응학과 교수

안 희 학

1981년 숭실대학교 전자계산학과(공학사)
1983년 숭실대학교 전자계산학과(공학석사)
1994년 숭실대학교 전자계산학과(공학박사)
1984년~현재 관동대학교 컴퓨터학부 교수