通信情報保護學會誌
第8卷 第3號, 1998. 9

공개키 기반 구조

Overview on Public Key Infrastructure

염 흥 열*, 홍 기 웅**

요 약

인터넷을 통한 상거래가 21세기에 중요한 상거래 수단이 될 전망이다. 인터넷 상거래 시스템은 공개키 암호시스템에 바탕을 두고 실현되고 있다. 공개키 암호 시스템의 폭넓은 사용은 사용자 공개키가 신뢰성이 있어야 안전성을 보장받을 수 있다. 공개키 기반 구조는 이를 위하여 구축되어야 한다. 따라서 안전하고 신뢰성 있게 공개키를 관리하고 공표하기 위한 공개키 기반 구조는 인터넷 전자상거래 시스템뿐만 아니라 정부 기간 통신망에서도 매우 중요한 역할을 수행할 것이다. 공개키 기반 구조는 기본적으로 공개키를 CA의 서명용 비밀키로 서명한 공개키 인증서를 활용한다.

본 고에서는 공개키 기반 구조를 정의하고, 공개키 기반 구조에서 사용될 보안 알고리듬을 살펴보며, X.509 인증서 구조와 종류, 보안 알고리듬 및 보안 정책을 식별하기 위한 OID(Object ID)와 통신 개체를 확인하기 위한 X.500 DN(Distinguished Name)를 정의하고, 믿음의 연결 고리인 인증 경로, 인증 경로의 검증 방안, 새로운 요구사항인 인증 경로의 제한, 인증서 발행 및 발급을 위한 인증서 정책, 그리고 인증서 관리 방법 등을 인터넷 공개키 기반 구조인 PKIX 문서를 중심으로 기술한다. 또한 공개키 기반 구조를 위한 호환성 규격을 제시한다.

1. PKI 개요

1.1 공개키 암호 방식과 PKI

인터넷상에서 존재하는 안전한 파일 전송프로토콜, 안전한 WWW(World Wide Web), 그리고 전자상거래를 위한 대부분의 보안 응용은 공개키 알고리듬에 바탕을 두고 있다. 공개키 알고리듬을 이용하는 사용자는 공개키를 공표하고, 공개키에 대응되는 비밀키를 비밀스럽게 간직하다가 비밀 보안 동작이 수행될 때 이를 사용한다.[1] 공개키 기반 구조는 보안 응용이 공개키 암호 시스템를 사용한다고 전제하에서 구축될 수 있다. 공개키 암호시스템에서의 사용자는 일반적으로 공개키와 비밀키의 쌍을 간섭이 불가능한 스마트카드 형태로 저장한다. 메시지는 메시지 수신자의 공개키로 암호화되고, 메세지 수신자는 자신의 비밀키로 메시지를 복호한다.

* 순천향대학교 공과대학 전기전자공학부

** 한국정보보호 센터

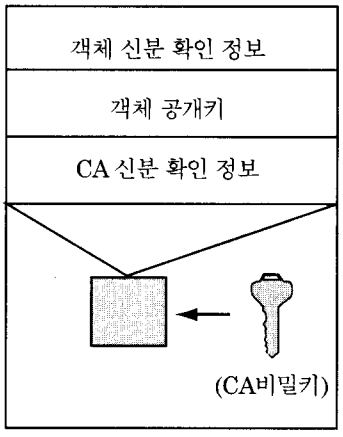

그림 1. 인증서 구조

공개키 인증서(이후 인증서)는 그림 1과 같이 사용자의 공개키와 사용자의 ID을 결합하는 것으로, 특정 개체를 확인하고 특정 활동, 특정 권한, 그리고 특정의 능력을 허가하는 데 이용된다. 인증서는 특정의 권한과 신분을 확인하기 위한 정보로 구성되어 있으며, CA의 서명문이다. 따라서 사용자의 공개키와 사용자의 신분 확인은 CA에 의하여 수행되거나 CA 역할을 대신하는 RA에 의하여 수행된다.[8,9] 인증서는 최종 사용자를 위하여 발행된 인증서와 상호 인증을 위하여 CA를 위하여 발행된 인증서로 구분된다. 인증서는 기본적으로 특정 인증서를 유일하게 확인하기 위하여 CA가 인증서에 부여하는 일련번호(SN: Serial Number), 인증서를 생성하는데 이용되는 서명 알고리듬을 확인하기 위한 서명 알고리듬 ID, 발행한 CA를 확인하기 위한 X.500 이름, 시작 일시에서 종료 일시를 포함하는 인증서 유효기간, 인증서를 발행 받는 인증서 소지자의 X.500 이름, 그리고 인증서 소지자의 암호 공개키를 모아서, 이들을 해쉬한 후 CA의 서명용 비밀키로 서명한 서명문을 구한 후, 원래 정보와 서명문을 연결한 데이터 스트링이다. 지금까지 표준화된 인증서 버전은 X.509 버전 1, X.509 버전 2, 그리고 X.509 버전 3 형태를 갖는다. 또한 응용 분야에 따라서 인증서를 구별하면 X.509 신분 확인용 인증서, X.509 SET(Secure Electronic Transaction) 인증서, PGP(Pretty Good Privacy) 인증서, DNS(Domain Name Server) 인증서, 그리고 SPKI(Simple Public Key Infrastructure) 인증서 등이 있다.

PKI(Public Key Infrastructure)는 사용자 공개키와 사용자 ID를 안전하게 묶는 방법과 공개키를 신뢰성있게 관리하기 위한 수단을 제공하고 있다. 즉, PKI는 공개키 암호 시스템에서 사용하는 공개 정보를 공표하는 시스템이다.[1,3,4,5] PKI는 암호화된 메일, 암호화된 WWW, 안전한 지불 프로토콜 등 다양한 인터넷 보안 응용을 가능케 하기 위한 바탕을 제공한다. PKI가 제공하는 대표적인 보안 서비스는 프라이버시 보호 서비스와 디지털 서명 서비스이다. PKI는 공개키 인증서(Certificate), 인증기관(Certificate Authority), 등록기관(Registration Authority) 등의 요소를 가지며, EE 및 CA의 인증서 관리 서비스, 디렉토리 서비스를 규정하고, 인터넷상에서 거래와 관련된 각 객체에 대한 인증과 신분 확인 기능을 제공한다.

PKI 기능은 두 가지 기능을 포함한다. 하나는 공개키를 개인, 조직, 또는 다른 개체에 묶어주는 인증(Certification) 과정과 인증이 유효하다는 것을 검증하는 유효성 검증 과정으로 구성된다. PKI를 사용하면 신분확인 및 권한, 프라이버시와 기밀성, 무결성, 인증, 부인봉쇄, 액세스 제어, 문서 전송의 증명, 문서의 보관 및 조회 등의 서비스를 제공받을 수 있다. 인증서의 특수한 활용 분야로 전자수표를 사용할 수 있는 권한을 주거나, 은행이 전자수표를 검증하는데 활용된다.

사용자는 자신이 믿는 인증서 발행 인증기관의 공개키를 해당 인증기관에서 발행한 인증서의 유효성을 검증하기 위하여 오프라인으로 인증기관과 연결되어 CA의 공개키를 저장

하고 있어야 한다. 일반적으로 최상의 루트 CA의 공개키는 자산의 서명용 비밀키로 서명한 자가인증서로 확인될 수 있다. 일반적으로 인증 경로는 여러 인증서들의 체인이다. 인증 경로는 믿음의 경로라고 할 수 있다. 인증 경로를 검증할때는 반드시 각 인증기관이 발행한 CRL을 검증하여 인증서의 기한 만기 이전에 폐기되어 있는 상태를 함께 검증해야 한다.

PKI는 인증서에 바탕을 두고 있다. 인증서의 주체 필드(Subject Field)는 X.500 디렉토리의 DN(Distingished Name)으로 특정 개체를 유일하게 확인한다. X.500 디렉토리는 사람의 이름이 주어지면 그 사람에 대한 부가 정보를 얻을 수 있는 전화 번호부와 비슷한 역할을 한다. X.500 디렉토내의 엔트리는 특정인의 주소, 전화번호, 그리고 이름 등의 기본 정보와 그 사람이 속한 조직, 전자 우편 주소, 그리고 그의 직위를 포함하는 부가적인 속성 정보를 제공한다. X.500 엔트리는 실제로 임의의 조직에 속한 특정 개체를 나타내며, 이는 사람과 컴퓨터, 프린터, 회사, 정부, 그리고 나라를 나타낸다. 각 엔트리에는 개체의 공개키를 규정하는 인증서를 포함한다. 각 엔트리는 유일하게 구별 가능한 이름인 DN을 할당받는다. 이 DN의 유일성을 보장하기 위하여 X.500디렉토리는 계층적 구성을 갖는 DIT(Directory Information Tree)로 구성된다. 트리의 각 노드는 하나의 부모 노드와 여러개의 자식 노드로 구성된다. 각 노드는 루트 노드를 제외하고는 모두 RDN(Relative Distinguished Name)을 할당받는다. 루트 노드 아래에는 두 개의 문자로 구성된 나라를 나타내는 RDN이 할당된다. 이는 ISO에 의하여 할당되며, 나라를 나타내는 엔트리 아래에는 나라안의 다양한 조직을 나타내는 정부 조직, 일반 회사, 지방정부, 그리고 연방 정부가 설립한 국영회사 등을 나타내는 엔트리들이 존재한다. 각 엔트리는 유일한 RDN을 할당받는다. 마지막으로 각 조직은 조직이 보유한 종업원 또는 조직이 제어해야 할 다양한 개체들을 나타내는 엔트리들을 포함한다. 여기에도 유일한 RDN 이 할당된다. 만약 다음과 같은 조직을 갖는 홍길동이라는 사람의 속성을 알아보자. 루트 아래에는 한국을 나타내는 C=KR이라는 엔트리가 존재하고, 그 아래에는 순천향대학교를 나타내는 O=SCH 라는 엔트리가 존재하며, 그 아래에는 특정 교수인 홍길동을 나타내는 CN=K.D.Hong 이라는 엔트리가 존재한다. 여기서 C=Country, O=Organization, CN= Common Name을 나타낸다. 최종 엔트리는 다시 그엔트리의 속성 정보인 전화번호, 이름, 전자메일주소, 직위, 그리고 인증서 등이 포함되어 있다.

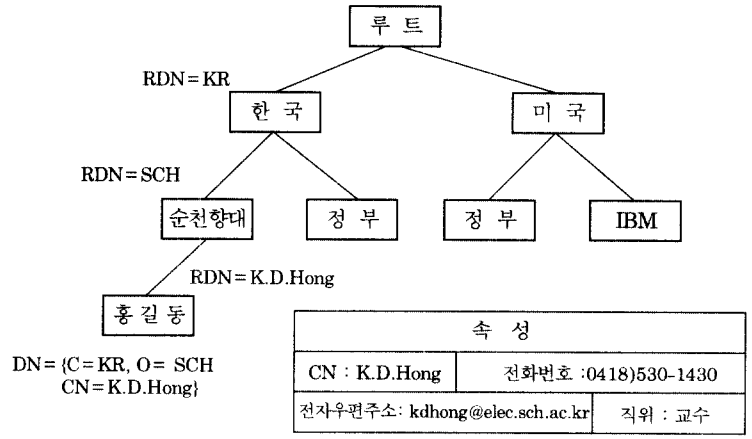

그림 2 X.500 DN

X.509 인증서는 사용되는 보안 알고리듬, 서명 알고리듬, 인증 정책 등의 객체를 확인할 필요가 있다. X.509는 국제적으로 정의된 객체 확인자(OID : Object Identifier)를 사용한다. OID는 정수의 계열로 구성된 수열(Sequence)이라고 할 수 있다. OID는 계층적으로 할당된다. 어떤 회사나 조직도 OID을 할당할 수 있는 기관 역할을 할 수 있다. 예를 들어 다음과 같은 OID를 확인해보자. 한국 정보보호센터가 할당한 전자상거래 정책에 대한 OID는 2-16-40-1-45356-13-15 이다.

PKI를 위하여 이용되어야 할 주요 표준안은 표 1과 같다. 암호 보안 규격은 미국의 암호 모듈 보안 요구사항 등이 있고, 보안 알고리듬은 기존의 대칭형 알고리듬과 공개키 알고리듬을 들 수 있다. 통신 프로토콜은 디렉토리 검색 프로토콜인 LDAP(Lightweight Directory Access Protocol) 과 안전한 메일 관련 표준인 PEM(Privacy Enhanced Mail),[2,6] 그리고 안전한 보안 API(Application Program Interface) 표준인 GSS API 표준 등이 있다. 통신망 프로토콜은 TCP/IP 프로토콜이고, 기반 시스템은 X.500 디렉토리 서비스에 바탕을 두고 있다. PKI 관련 표준은 X.509 v3 인증서 표준과 SPKIM(Simple Public Key Infrastructure Mechanism), PKI 요소 간에 최소 상호 동작 규격을 다룬 NIST 표준 MISPC (Minimum Interoperability Specification for PKI Components), 그리고 IETF PKIX 문서인 여러 문서 등이 있다.[10,11,12,13,14]

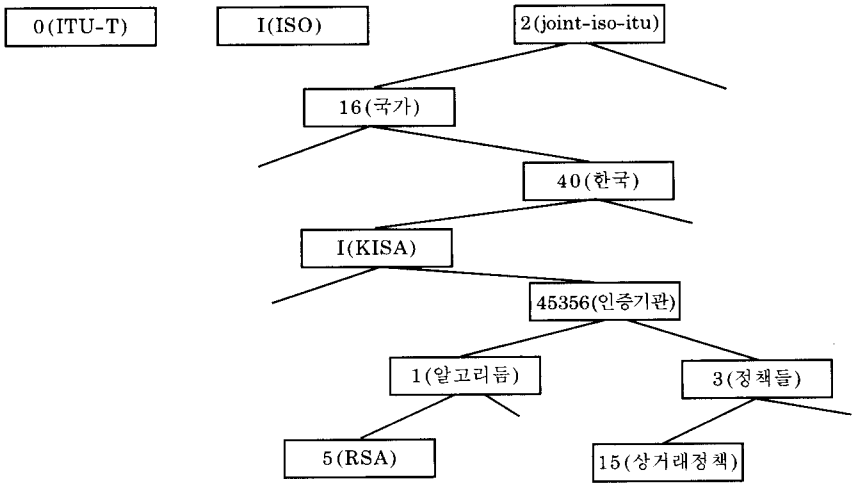

그림 3. OID

표 1 PKI 관련 표준안

종 류	표 준
암호보안	FIPS140-1암호모듈 보안 요구사항
암호 알고리듬	DES, Triple, RSA/MD5, RSA/SHA-1 DSA/SHA, MD2
통신 프로토콜	RFC 17787 LDAP, ISO/IEC 8824/8825 ASN.1 SMIME, RFC 1508 GSSAPI
망	TCP/IP
기반	X.500 Directory 서비스
PKI	X.509 인증서, SPKM, MISPC, IETF PKIX

1.2 PKI를 위한 보안 알고리듬

PKI를 실현하기 위하여 반드시 요구되는 메카니듬은 비밀키 암호 알고리듬, 공개키 암호 알고리듬, 디지털 서명 알고리듬, 그리고 해쉬 알고리듬 등이다. 비밀키 알고리듬은 송신자의 암호키와 수신자의 복호키가 동일한 암호 알고리듬이다.[11] 이 방식은 많은 사용자를 갖는 조직에서 사용하면 공개키 암호 알고리듬에서 요구되지 않았던 키관리를 위한 부가적인 과정이 요구된다. 공개키 알고리듬은 암호키와 수신자의 복호키가 다른 암호시스템이다. 공개키 알고리듬의 대표적인 알고리듬은 Diffie-Hellman 알고리듬과 RSA 알고리듬, 그리고 ElGamal의 알고리듬을 들 수 있다. 공개키 알고리듬은 공개키로부터 사용자의 비밀키의 복구가 계산적으로 매우 어려운 단점이 있다. Alice가 공개키를 공표하면, 임의의 다른 사용자는 Alice의 공개키로 암호문을 생성하여 Alice에 보내고, Alice만이 자신의 비밀키를 이용하여 수신된 암호문을 복구할 수 있는 암호시스템이다. 디지털 서명 알고리듬은 자신의 서명용 공개키를 공표하고, 자신의 서명용 비밀키를 이용하여 메시지에 대응되는 서명문을 생성하여 임의의 사용자에게 보내면, 임의의 사용자는 Alice의 서명용 공개키를 이용하여 Alice의 서명문을 검증한다. 이의 대표적인 알고리듬은 RSA 서명 알고리듬과 미국 표준 서명 알고리듬인 DSA 알고리듬을 들 수 있다. 해쉬 알고리듬은 가변 길이의 메시지를 일정한 길이의 메시지 다이제스트로 변환하는 알고리듬으로서, 충동회피성과 일방향성 등의 특성을 만족해야 한다. 대표적인 해쉬 알고리듬은 SHA 알고리듬과, MDx(x=2,4,5) 알고리듬을 들 수 있다.

공개키 인증서에 직접적으로 이용 가능한 암호 알고리듬은 해쉬 알고리듬과 디지털 서명 알고리듬이다. 일방향 해쉬 알고리듬은 임의의 사용자와 발행자 관련 메시지를 일정 길이의 메시지 다이제스트로 변환하는 기능을 수행한다. 디지털 서명 알고리듬은 해쉬된 결과값을 사용자의 비밀키로 서명하는데 이용되는 알고리듬이다.

1.3 PKI 관련 용어

본 절에서는 PKI를 위한 일반적인 용어를 정의한다.[1,2,12,13,14]

인증서 또는 공개키 인증서(Public Key Certificate)는 객체의 공개키와 사용자 ID를 묶기 위하여, 이들 정보를 CA의 서명용 비밀키로 서명한 디지털 스트링이다. 인증서 소유자(Certificate Holder)는 적법한 인증서를 CA로부터 발급받고 사용하는 주체(Subject)이다.

인증서 사용자는 해당 객체의 인증서를 구하고 인증서를 이용하여 해당 객체의 공개키를 도출하며, 이를 보안 응용에 활용하는 객체이다. 인증서 사용 시스템(Certificate-using System)은 X.509 표준에 정의된 다양한 기능들을 구현한 시스템이다.

인증기관(CA:Certificate Authority)은 최종 객체(EE:End Entity)와 다른 CA 들에게 인증서를 발행해 주는 신뢰성 있는 제삼의 객체이다.

최종 객체(EE:End Entity)는 공개키 인증서 내에 있는 공개키에 대응되는 비밀키를 알고 있으며, 이는 인증서내 주체 필드에 있는 객체이다.

클라이언트 또는 사용자는 인증서를 획득하여 인증서의 유효성을 검증하기 위하여 인증서내 서명문을 검증하고 PKI 시스템을 활용하는 개체이다.

인증 경로는 서명 검증자가 신뢰하는 인증기관에서 특정의 사용자 인증서까지 CA 간의 상호 인증서와 EE의 객체 인증서들을 직렬로 모아둔 인증서 들의 계열이다.

인증서 취소 목록(CRL: Certificate Revocation List)은 인증서의 유효 기간이 경과되지 않았으나 보안키 누설 등으로 인해 폐지된 인증서들의 목록을 저장하고 있다.

delta-CRL는 사용된 인증서의 수가 증가함에 따라 CRL의 크기가 무한대로 커 가는 것을 방지하기 위하여 가장 최근 CRL이 발행된 시점에서 새로운 delta-CRL이 발행된 시점까지의 폐지된 인증서들을 모아둔 목록이다.

디렉토리 서비스(DS: Directory Service)는 모든 사람들에게 공개된 정보를 저장할 수 있는 분산 데이터베이스 서비스이다.

인증 실행 명세서(CPS: Certification Practice Statement)는 CA가 최종객체에게 인증서를 발행하기 위하여 요구되는 일반적 절차를 자세히 규정한 명세서이다.

LDAP(Lightweight Directory Access Protocol)은 인증서 등의 PKI 정보를 모아둔 디렉토리로의 접근 프로토콜로써 LDAP v2가 PKI를 위하여 사용된다.

RA(Registration Authority)는 인증서를 발행하는 CA와 인증서를 발급받는 객체 사이의 중간 매개체 기능을 수행하는 객체이다. CA는 RA를 전적으로 믿고 RA의 인증서 발행 요청을 받아들인다.

Out-of-band는 데이터 등이 전달되는 in-band에 대응되는 개념으로써, 물리적 방법으로 수행되는 트랜잭션들이다. 이는 전달자에 의한 전달, 스마트카드에 의한 전달, 또는 기존의 등기 우편에 의한 전달 방법을 포함한다.

인증서 정책(Certificate Policy)은 인증서를 생성하기 위한 규칙들(Rules)의 집합으로서, 특정의 집단이나 특정의 응용에 적용되며, 인증서 사용자가 수신된 인증서가 특정 응용에 적용 가능한지를 결정하기 위한 판단 기준이 된다. 대표적인 예로는 전자상거래용 인증서 정책과 전자메일용 인증서 정책이 있다.

정책 매핑(Policy Mapping)은 CA가 다른 보안 영역에 속한 CA를 인증할 때 서로의 인증서 정책이 동일함으로 공표하는 것이다. 저장소는 디렉토리 서비스를 포함하지만 디렉토리 서비스로 한정되지는 않는다.

1.4 인터넷 PKIX 표준안[12,13,14]

인터넷 응용에 적용 가능한 PKI를 구축하기 위한 표준안이 PKIX 이다. PKIX 표준안에는 X.509 인증서 및 CRL 프로필, 인증서의 KEA 키 표현, 인증서의 Diffie-Hellman 키 표현 등의 내용을 포함하고 있는 인증서 및 CRL(Certificate Revocation List) 관련 표준안, 모든 사용자의 인증서와 사용자의 폐지된 인증서에 대한 목록을 저장하고 있는 보관소에 대한 액세스 방법을 규정한 LDAP(Lightweight Directory Access Protocol) v2, 인증서 보관소에서 인증서와 CRL을 얻기 위한 FTP(File Transfer Protocol)나 HTTP 프로토콜, EE가 빠르고 효율적으로 인증서의 유효성과 폐지 상태를 판단하게 하는 OCSP(Online Certificate Status Protocol) 등의 운영 프로토콜 관련 표준안, PKI 모델, PSE 환경, 초기등록/인증 과정, POP 과정 등을 포함하는 인증서 관리 프로토콜 표준안, 그리고 인증서 정책과 CPS 프레임워크 등이 있다.

2. PKI 특징

2.1 인증서 및 PKI 구성 요소

PKI 구성 요소는 그림 4와 같이 EE를 위한 인증서를 발급하고 폐지하는 인증기관(CA:Certificate Authority), 인증서를 인증기관으로부터 발급받는 인증서 소지자, 그리고 인증서 소유자와 사용자의 공개키 사이의 연결성을 보장해주는 RA(Registration Authority),

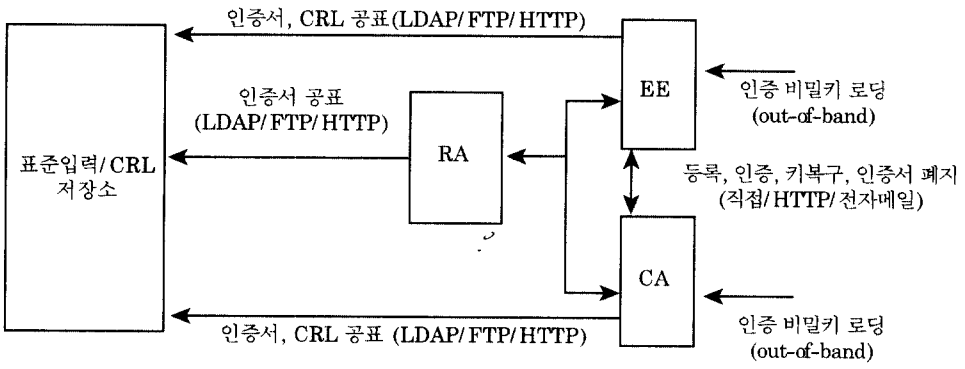

그림 4 PKI 구성 요소

그리고 인증서 및 CRL을 저장하고 공개하는 보관소, 인증 경로의 유효성과 전자서명의 유효성을 검증하는 클라이언트로 구성되어 있다. [12]

인증서의 유효성을 검증하기 위해서는 기본적으로 인증서를 발행한 CA의 서명용 공개키를 미리 인증서를 검증하는 검증자가 오프라인으로 가지고 있다고 가정한다. 수신된 인증서의 유효성을 검증함으로써 인증서 검증자는 인증서 안에 포함되어 있는 사용자의 공개키의 유효성을 확인할 수 있다. 인증서의 유효성은 보통의 서명문의 유효성을 검증하는 절차와 동일한 과정으로 수행된다. 먼저 인증서의 일련번호에서 인증서 소지자 공개키 부분을 포함하는 메시지를 약속된 해쉬 알고리듬으로 해쉬한 결과 값을 얻는다. 그런 후 서명문을 CA 서명용 공개키로 복호하여 구한 해쉬 결과 값을 구한다. 만약 계산된 해쉬 값과 서명문으로부터 복구된 해쉬 값이 동일하면 인증서는 유효한 것으로 판단되며, 만약 두 값이 동일하지 않으면 인증서는 유효하지 않은 것으로 판단하고 추후의 암호학적 동작을 중단한다.

이 세상에 존재하는 모든 EE가 하나의 인증기관에서 인증서를 발급받도록 하는 것은 불가능하다. 따라서 여러 개의 CA가 존재할 수 있다. 그러므로 서로 다른 인증기관간의 상호 인증(Cross-Certification)이 요구된다. 그림 5에 나타난 바와 같이 Alice가 CA X의 공개키를 가지고 있다고 가정하고 CA Y는 CA X의 인증서를 발급해주고, CA Y가 CA Z의 공개키 인증서를 발급하며, CA Z는 Bob 의 공개키 인증서를 발급한다고 가정하자. Alice가 Bob의 공개키를 인증하고 싶을 경우, Alice와 Bob이 공통으로 믿는 믿음의 연결점은 CA Y 이다. 따라서 Alice는 Bob의 공개키의 유효성을 검증하기 위해서는 자신이 믿는 CA Y가 발행한 CA Z의 공개키 인증서내의 CA Z 의 공개키의 유효성을 확인한 후에 CA Z 가 발행한 Bob의 인증서 내의 Bob의 공개키의 유효성을 확인함으로서 Bob의 공개키의 유효성을 확인한다. Alice 는 Bob의 공개키의 유효성을 확인하기 위하해서는 자신이 믿는 CA Y가 발행한 CA Z의 인증서와 CA Z 가 발행한 Bob 의 인증서가 필요하게 된다. 즉, CA Z 의 인증서와 Bob의 인증서의 연결이 인증 경로를 형성하게 된다. 인증 경로는 한마디로 믿음의 연결이라고 할 수 있다. 또한 믿음의 연결을 확립하기 위하여 여러 다른 CA 들간의 상호 인증이 요구하게 된다. 이러한 방식은 기본적으로 Alice가 자신의 루트 인증기관에게 믿음

을 주고 다시 Alice의 인증기관은 다른 인증기관에 믿음을 줌으로서 믿음을 확장시키는 효과를 초래한다. 이와 같은 방법으로 믿음을 확장하는 방법이 인증 경로이다. 믿음을 확장하기 위한 CA 배치는 크게 계층적 구조를 갖는 경우와 네트워크 형의 평면적 구조를 갖는 경우, 그리고 두 방식을 결합한 방식이 이용될 것이다. 일반적으로 나라마다 회사마다 기관마다 다른 인증기관들을 구축하고 인증기관간에 상호 인증을 통하여 믿음을 확장하고, 이를 통하여 인증서의 유효성을 검증하는 검증 기법이 사용될 것이다. CA 계층은 PEM의 경우 최상위 계층이면서 다음 하위 계층인 PCA로 인증서를 발행해 주는 IPRA(Internet Policy Registration Authority), 응용에 따른 인증서 정책을 결정하는 PCA(Policy Certification Authority), 그리고 특정의 그룹 및 조직 구성원에게 인증서를 발행해 주는 CA(Certification Authority) 등으로 구분된다.

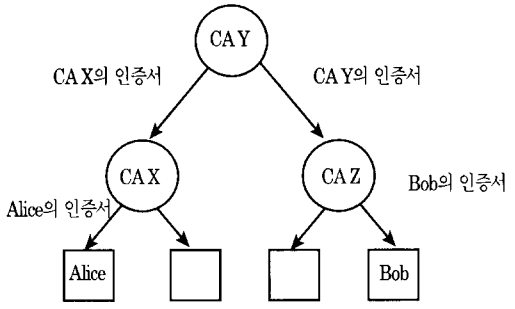

그림 5. 인증 경로

CA가 개체를 위하여 인증서를 발행하여 특정 개체를 보증하고 인증서 사용자가 특정 인증서의 유효성을 검증하면 그 개체는 인증 받았다고 한다. 일반적으로 인증의 유형은 일반적으로 ID 인증(Identity Authentication)과 자격 인증(Credential Authentication) 으로 구분된다. 따라서 인증서도 ID 인증서와 자격 인증서로 분류될 수 있다. ID 인증서는 인증서내의 주체 필드에 X.500 DN이 들어가서 특정 개체의 신원을 인증하는 인증서이다. 자격 인증서는 인증서내의 주체 필드에 특정 개체의 자격 이나 특정 객체로의 액세스 권한을 나타내는 자격 정보가 입력되어 인증서가 발행된다. 자격 인증서 내의 주체 필드에는 특정의 컴퓨터로의 액세스 권한을 나타내는 허용 동작, 특정 회사에서의 지위를 나타내는 지위 정보, 그리고 이 동작이 누구에 의하여 검증 받았다는 것을 나타내는 자격 등의 관련 정보를 포함한다.

확장자(Extension)는 인증서에 적용되는 보안 등급등의 종류와 응용을 지정하는 인증서 정책과 인증서 정책 매핑 필드, X.500 DN 이외에 다른 주소를 나타내는 대체 이름 필드, 객체가 속한 디렉토리의 속성을 기술하는 객체 디렉토리 속성 필드, 특정의 보안 정책이나 이름 부류로 인증 경로를 제한하는 인증 경로 제한 속성 필드 등을 포함하고 있다. 인증서 정책은 인증서에 적용된 규칙들의 집합으로서, 특정의 집단이나 특정의 응용에 적용되며, 인증서 사용자는 이를 이용하여 특정의 인증서가 특정의 응용에 적용 가능한가를 판단하게 한다. 일반적으로 인증서 정책은 OID로 공표되며, 특정의 OID와 정책을 기술하는 문서는 1:1 대응 관계에 있다. 인증서 정책의 대표적인 예는 범용 인증서 정책(General-purpose Policy) 과 상용 등급 인증서 정책(Commecial-graded Policy) 등이 있다. 범용 정책은 전자메일과 같은 일상적인 정보를 보호하기 위한 인증서 정책으로서 암호키쌍이 저가의 소프트웨어로 저장 및 보호되며, 키의 길이도 일반적으로 상용 등급 인증서 정책의 키 길이보다 짧다. 상용 등급 정책은 전자 상거래 및 금융거래를 위한 인증서 정책으로서 암호키 쌍이 암호학적으로 안전한 하드웨어 토큰에 의하여 보호되며, 암호키의 길이도 범용 정책의 암호키 길이보다 짧다. 또한 암호키도 엄격하게 발

급되고 관리된다.

X.500 버전 3 인증서는 기존의 버전 1,2의 인증서에 비해 확장자(Extension)의 개념을 도입한 것이다. 인증 경로는 믿음의 확장이다. 즉 자신이 믿는 루트 CA로부터 통신 상대가 믿는 상대의 루트 CA 로의 믿음의 확장이다. 그런데 만약 인증 경로에 암호학적 취약 노드가 존재하면 공격자는 이곳을 집중적으로 공략하여 사용자의 공개키에 대한 신뢰성을 훼손시킬 수 있다. 만약 Alice의 인증서가 전자상거래용으로 활용되도록 발행되었다고 가정하자. 그리고 Bob의 인증서 역시 전자상거래용으로 발행되었다고 가정하자. 인증 경로가 Alice-CA X-CA Y-CA Z-Bob 으로 인증 경로가 설정되었다고 가정하자. 만약 CA X 는 전자 상거래용 상호 인증서뿐만 아니라 전자 메일용 상호 인증서를 발행한다고 가정하면 일반적으로 전자 메일용 상호인증서는 암호키의 보관 및 관리가 전자상거래용 인증서의 그것보다 현저히 엄격하지 않다. 그런데 만약 인증 경로가 전자메일용 상호 인증서를 통해 설정되었다면 이는 암호학적인 취약 노드가 될 수 있다. 인증 경로의 제한은 특정의 CA로부터 나올 수 있는 인증 경로의 유형을 제한하는 기능을 수행한다. 즉, 인증 경로를 현재의 인증서로부터 특정의 이름 공간, 그리고 특정의 정책하에서 발행된 상호 인증서로 제한하는 기능을 수행한다. 이는 버전 3 인증서에만 존재하는 기능으로서 매우 중요한 의미를 갖는다.

PKI를 위한 암호 보안 API 프리미티브는 암호키의 안전한 실현을 위한 암호키 생성 루틴, 해쉬 결과 값을 얻기 위한 해쉬 루틴, 기밀성을 위한 대칭형/비대칭형 알고리듬 루틴, 외부에서 유입되는 암호키의 설치 및 외부로의 전달을 위한 키 수출/수입 루틴, 하나의 암호키에서 파생되는 여러 암호키를 유도하기 위한 키 유도 루틴, 암호키의 사용을 제어하는 키 사용 제어 루틴, 데이터 무결성 검사 루틴, 데이터 기밀성 루틴, 데이터 서명 루틴 등이 있다.

2.2 인증서 관리

인증서 관리는 일반적으로 인증서를 사용자에 발급하는 CA, 인증서를 발급 받는 EE(End entity), CA 기능을 대신하는 RA(Registration Authority), 그리고 인증서/CRL 보관소 사이에서 이루어진다. CA는 믿을 수 있는 제삼자의 역할을 수행하며, 인증서 사용자에게 인증서를 발급해주고, 발급된 인증서를 공표하며, CRL을 공표하는 개체이다.[13] 인증서와 CRL을 공표하기 위한 프로토콜은 LDAP(Lightweight Directory Access Protocol), FTP, HTTP 프로토콜 중 하나가 이용한다. EE는 CA와 연동하여 인증서를 발급받으며, 인증서 내의 주체 필드내의 응용이나 특정의 사용자를 나타낸다. 인증서/CRL 보관소는 CA나 RA의 요구에 응하여 발급된 인증서 및 CRL를 저장하거나 PKI 개체들의 요구에 응하여 이를 전달하는 기능을 수행하는 개체이다. RA는 CA를 대신하여 특정 사용자의 신분을 확인하고, 특정의 하드웨어 토큰을 분배하며, 특정 사용자의 인증서에 대한 폐지를 보고하고, 특정 사용자에 이름을 할당하는 역할을 수행한다.

인증서 관리 프로토콜은 CA와 CA간의 프로토콜, CA와 클라이언트간의 프로토콜로 구성된다. CA 간의 관리 프로토콜은 주로 상호 인증서의 발행을 위한 프로토콜이고, CA와 클라이언트간의 프로토콜은 EE를 위한 인증서의 발행과 관련된 프로토콜이다. PKI 인증서 관리를 위한 기본 동작은 초기 CRL를 생성하고 다른 CA의 서명용 공개키를 수입하는 동작으로 구성된 CA 초기화 과정, 자신이 믿는 CA의 공개키를 수입하는 EE 초기화 과정, 인증서를 생성하고 초기 등록 및 인증(Initial Registration and Authentication) 과정을 수행하며 CA 키 쌍의 갱신하고 상호 인증을 요구하

고 갱신 등의 동작을 수행하는 인증 과정, 발행된 인증서와 폐지된 CRL 공표 과정, 사용자의 비밀키를 CA에 보관하여 두었다가 사용자의 부주의 등으로 암호키가 손실된 경우 이를 복구하기 위한 키 복구 과정, 발행된 인증서의 비밀키가 누설된 경우 이 인증서의 발행을 취소하기 위한 인증서 폐지 과정으로 구성된다.

인증서 관리 과정 중 가장 중요한 과정은 초기 등록 및 인증 과정으로써 이 과정은 EE가 최초로 CA와 연결되어 자신의 신분을 확인 받으며, EE의 암호키 쌍을 생성하고, 이에 대한 공개키 인증서를 생성하여 전달 받는 과정으로 구성된다. 이 과정은 사용자의 암호키 쌍을 어디서 생성하여 분배한지에 따라 집중화된 방식과 기본 인증 방식으로 구분된다. 두 방식 공히 이 과정동안 기밀성 및 무결성 검사를 위한 암호키는 CA와 EE간에 오프라인으로 사전에 분배가 완료되어 있다고 가정한다. 집중화된 방식에서는 그림 6과 같이 CA가 EE를 위한 암호키 쌍을 생성하여 이에 대한 인증서를 생성하고 그 결과를 EE로 다시 전달하는 인증서 생성 과정을 거친다. 기본 인증 과정에서는 그림 7과 같이 EE가 인증서에 포함될 암호키 쌍을 생성한 후 자신의 공개키를 포함하고 있는 인증서 요구 메시지를 CA로 전송한다. 이를 전달받은 CA는 메시지의 무결성과 유효성을 검증한 후 공개키에 대응되는 공개키 인증서를 생성하여 EE로 전달한다. 이를 수신한 EE는 메시지의 유효성을 검증한 후 자신의 공개키에 대응되는 인증서를 유도한다. 일반적으로 초기 등록 및 인증 방법은 기본 인증 방법이 사용자의 비밀키를 CA로 알리지 않아도 되므로 널리 이용될 수 있으나 사용자에 의한 안전한 암호키 쌍의 생성이 요구된다.

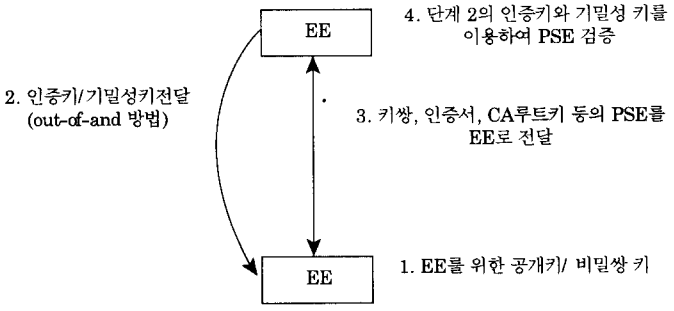

그림 6. 집중방식

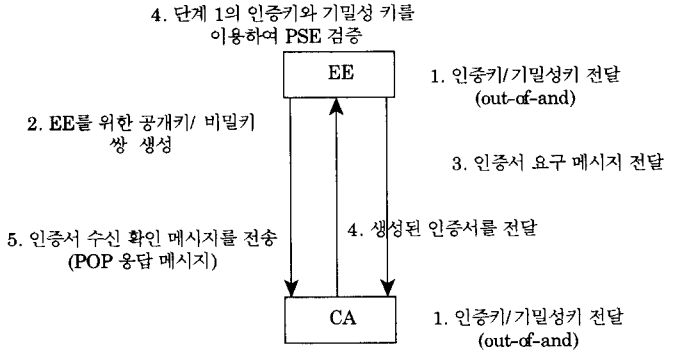

그림 7. 기본 인증 방식

인증서 관리는 하나의 관리 영역에 한정되어 있는 문제가 아니라 여러 관리 영역에 거쳐서 수행된다. 따라서 여러 사용자 및 CA 들 간의 상호 동작이 가능하기 위해서는 단일영역에 적용될 수 있는 관리 기능의 호환성, 다중 영역에서 보장되는 상호 인증 기능의 호환성, 그리고 인증서 및 CRL 의 공표 및 저장을 위한 호환성이 요구된다. 단일 영역에 적용될 호환성을 위하여 규정되어야 사항은 초기 등록 및 인증 방법의 유형, POP(Proof of Possession) 방법, 관리 메시지를 보호하는 방법, 지원되는 관리 기능의 종류, 이용되는 관리 프로토콜, 그리고 관리를 위한 암호 알고리듬의 종류 등이 규정되어야 한다. 초기 등록 및 인증 방법은 집중 인증 방식과 기본 인증 방식이 있으며, 이들 중 하나가 호환성을 위하여 선택되어야 한다.

일반적으로 암호키는 서명용 암호키, 기밀성용 암호키, 그리고 키 일치용 암호키로 구분된다. 따라서 인증서도 이들 세가지 암호키중 하나를 위하여 발행되어야 한다. 만약 기밀성용 공개키에 대한 인증서를 얻기 위해서는 자신의 공개키에 대응되는 비밀키의 소지 사실을 상대에 분명히 입증해야 한다. 자신의 공개키에 대응되는 비밀키를 자신이 알고있다는 것을 확인하기 위한 POP 방식은 자신의 비밀키를 직접 비밀키를 전달하는 방식인 직접 자신의 공개키에 대응되는 비밀키의 소지를 확인하기 위한 방법이며, 이는 인증서 요구 메시지 안에 공개키에 대응되는 비밀키를 삽입하여 전달하는 직접 방법, CA가 난수를 EE의 공개키로 암호화하여 EE로 전달하면 EE는 자신이 비밀스럽게 간직하고 있는 비밀키를 이용하여 난수를 복구하여 이를 CA로 전달하면 CA가 복구된 난수가 자신이 보낸 난수와 일치한가를 검사함으로서 EE가 비밀키의 소지를 확인할 수 있는 직접 POP 방법, CA가 인증서를 EE의 공개키로 암호화하여 전달하고 EE는 자신의 비밀키로 인증서를 복구하여 CA로 전달하면 CA가 복구된 인증서를 검증함으로서 EE의 공개키에 대응되는 비밀키의 소유를 확인하는 난수를 자신의 비밀키로 암호화하여 다시 CA로 전달하면 되돌리고 이용하여 직접 검증 방법, 인증서를 이용한 간접 검증 방법 등이 있다. 이들 중 한 방식이 POP 방식의 호환성을 위하여 선택되어야 한다. 또한 상호 인증을 위한 호환성 규격은 메시지의 기본 형식을 정의하는 상호 인증 메시지 규격, 관리

표 2. 호환성 규격

버 전	3	CRL버전	2
인증국 키 확인자	사 용	초기등록/인증	기본인증
객체 키 확인자	사 용	POP 방법	서명, 암호
키 용도	디지털 서명	초기등록/인증	PasswordBased MAC
인증서 정책	OID로 할당	관리기능	루트 CA 초기화 루트 CA 키갱신 CRL, EE초기화 인증서 요구, 키갱신
객체 대체 이름	RFC822이름		
기본제한	사용	관리메세지 전달	HTTP
해쉬알고리듬	SHA-1	운영프로토콜	LDAP v2
서명 알고리듬	RSA	루트 CA공표	수동분배 스마트 카드 분배
subject 공개키 알고리듬	RSA		

메시지 전달 규격, 그리고 X.509 인증서 규격 등을 포함해야 한다. 인증서 및 CRL 공표 및 저장을 위한 호환성 규격은 상호 인증을 위한 X.509 규격, CRL 규격, 그리고 PKI 정보를 조회하고 관리하기 위한 액세스 요구조건과 보관소에 저장된 PKI 정보에 새로운 정보를 부가하고 삭제하며, 이를 변경하는 절차를 규정한 운영 프로토콜의 규격을 정의해야 한다. 따라서 단일 보안 영역 및 다중 보안 영역의 호환성을 만족하기 위한 호환성 규격 명세서가 정의되어야 하며, 이의 대표적인 규격이 표 2와 같다. 인증기관 키 확인자는 여러개의 암호키 중에서 특정의 암호키를 확인하기 위하여 정의되었다. 기본 제한(Basic Constraint)은 해당 인증서 다음에 올 수 있는 인증 경로의 최대 깊이를 나타내는 값으로써 이 인증서를 발행한 인증 기관이 EE를 위하여 인증서를 발행했는지, 아니면 다른 CA를 위하여 인증서를 발행했는지를 확인할 수 있다.

2.3 PKI 운영 프로토콜(Operational Protocol)

인증서 및 CRL의 상태를 조회하기 위한 프로토콜은 일반적으로 LDAP(Light Directory Access Protocol), FTP(File Transfer Protocol), HTTP(Hypher Text Transfer Protocol), 전자메일 등을 들 수 있다. PKI 운영 프로토콜은 인증서 및 CRL 보관소에 저장해 있는 PKI 정보를 부가하고 삭제하며, 변경하는 절차를 규정한다. 이는 LDAP 프로토콜로 실현되며, LDAP 프로토콜의 동작은 보관소 읽기, 보관소 탐색, 그리고 보관소 내용의 변경 등이 있다. 보관소 읽기는 고객과 서버간에 인증을 수행한 후 프로토콜을 개시하며, 간단한 신분 확인 방식과 고객의 신원을 전달하는 BindRequest 메시지와 특정의 보관소 내용을 읽기 위한 메시지로 구성된 SearchRequest 메시지로 구성된다.[10]

OCSP(Online Certificate Status Protocol)은 그림 8과 같이 EE가 특정의 인증서에 대한 유효성과 폐지 정보를 신속하고 효율적으로 알 수 있게 하기 위하여 인증서 상태에 관한 정보를 조회하는 프로토콜로서, 인증서의 상태를 조회하는 고객, 고객의 요구에 응하여 상태 정보를 전달하는 OCSP 서버, 그리고 일부 인증서들의 상태 정보를 저장하기 위한 CA 로 구성된다. 근본적으로 인증서들에 관한 정보는 인증서를 발행하는 CA에 존재한다. 그러나 인증서의 상태 정보를 CA에서 응답하면 CA는 너무 많은 조회 정보로 간섭받는 문제가 발생한다.

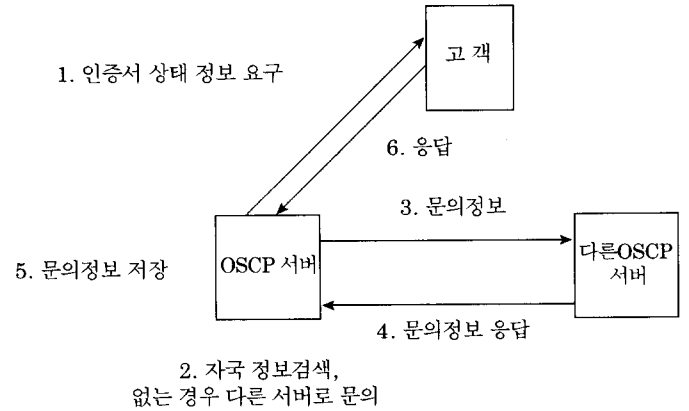

그림 8. OCSP 프로토콜

3. 결 론

기반 구조는 기밀성과 인증성, 그리고 무결성 서비스를 제공하기 위한 인터넷을 통한 상거래 시스템뿐만 아니라, 국가 기간 정보통신망 보안에도 적극 응용될 수 있다. 공개키 기반 구조는 공개키를 공표하고 관리하기 위한 것으로써 이의 구현이 전제되어야 다양한 인터넷상의 전자상거래가 가능할 것이다. PKI는 인증서 규격의 호환성, 관련 구성 요소들간의 호환성, 그리고 상호 인증을 위한 호환성 규격이 먼저 국가적 차원에서 설정되어야 구축될 수 있을 것이다. 본 고의 결과는 인터넷 상거래 및 국가 기간 정보통신망 실현시 적극적으로 활용될 수 있다.

참고문헌

[1] William Stallings, "Network and Internetwork Security," Prentice Hall International Editions, 1995.

[2] Linn, J., "Privacy Enhanced for Internet Electronic Mail: Part I: Message Encryption and Authentication Prccedures," RFC1421, IETF PEM WG, Feb, 1993.

[3] RSA Labatories, PKCS #1 : RSA Encryption, 1993.

[4] RSA Labatories, PKCS #7 : Cryptographic Message Syntax, 1993.

[5] RSA Labatories, S/MIME Message Specification, July, 1997.

[6] J. Galvin, S. Murphy, S. Croker, N. Freed, Security Multiparts for MIME: Multipart/Signed and Multipart/Encrypted, RFC1847, Oct, 1995.

[7] SET(Secure Electronic Transaction) Specification Book 1: Business Description

[8] ITU/ISO Recommendation X.500 Information technology Open Systems Interconnection The directory: Overview of concepts, models, and services, November 1993.

[9] ITU/ISO Recommendation X.509 Information technology Open Systems Interconnection The directory: Authentication framework, November 1993.

[10] Government of Canada, Communications Security Establishment(CSE)," Government of Canada Public Key Infrastructure-White Paper, 1998

[12] Housley, R., Ford, W. and Solo, D., Internet Public Key Infrastructure Part I: X.509 Certificate and CRL Profile, IETF X.509 PKI (PKIX) Working Group (draft)., March, 1998.

[13] Farrell, S., Adams, C. and Ford, W., Internet Public Key Infrastructure Part III: Certificate Management Protocols, IETF X.509 PKI (PKIX) Working Group (draft)., Februrary, 1998.

[14] S.Chokhani, W.Ford, Internet Public Key Infrastructure Certificate policy and Certification Practice Framework, IETF X.509 PKI (PKIX) Working Group (draft)., April, 1998.

□ 著者紹介

염 흥 열

1981년 한양대학교 전자공학과 졸업(학사)
1983년 한양대학교 대학원 전자공학과 졸업(공학석사)
1990년 한양대학교 대학원 전자공학과 졸업 (공학박사)
1982년 12월 ~ 1990년 9월 한국전자통신연구소 선임연구원
1990년 3월 ~ 현재 순천향대학교 공과대학 전기전자공학부 부교수
1997년 3월 ~ 현재 순천향대학교 산업기술 연구소 소장
1997년 3월 ~ 현재 한국통신정보보호학회 총무이사
※ 주관심분야: 암호이론, 부호이론, 이동통신 분야

홍 기 웅

1985년 2월 전남대학교 전자계산학과(학사)
1990년 2월 중앙대학교 대학원 전자계산과(석사)
1994년 4월 정보처리 기술사
1996년 2월 아주대학교 컴퓨터 공학과 (박사)
1985년 9월 ~ 1995년 10월 한국전자통신 연구소 선임연구원
1992년 ~ 1993년 이태리, Alenia Spazio사 Senior Researcher
1995년 10월 ~ 1996년 4월 한국전산원 선임 연구원
1996년 4월 ~ 현재 한국정보보호센터 책임 연구원, 기술기준 팀장
※ 주관심분야: 컴퓨터 • 네트워크 보안, 정보시스템 위험분석 • 평가, 정보보호 표준화