알기 쉬운

정보보호개론 3판

흥미로운 암호 기술의 세계

INFORMATION SECURITY and CRYPTOGRAPHY

INFINITYBOOKS
인피니티북스

CHAPTER 11 인증서

Section 01
인증서

1.1 인증서란 무엇인가?

1.2 인증서를 사용하는 시나리오

1.1 인증서란 무엇인가?

- **공개 키 인증서**(public-key certificate; PKC)
  - 이름이나 소속, 메일 주소 등의 개인 정보
  - 당사자의 공개 키가 기재
  - **인증기관**(CA; certification authority, certifying authority)의 개인 키로 디지털 서명
  - 인증기관은 국제적인 조직, 정부, 일반기업, 개인 등이 만들 수 있음
  - 인증기관 : verisign.com, 한국전자인증(Crosscert) 등등

1.2 인증서를 사용하는 시나리오(I)

1) 밥이 키 쌍(공개키,개인키)을 작성한다
   - 키쌍을 만드는 업무를 인증기관에 의뢰하는 경우도 있다
2) 밥은 인증기관 트렌트에 자신의 공개 키를 등록한다
   - 트렌트는 밥의 공개키가 맞는지 확인한다
3) 인증기관 트렌트는 밥의 공개 키에 자신의 개인 키로 디지털 서명을 해서 인증서를 작성한다
   - 트렌트는 미리 자신의 공개키와 개인키를 준비해둔다
4) 앨리스는 인증기관 트렌트의 디지털 서명이 되어 있는 밥의 공개 키(인증서)를 입수한다
5) 앨리스는 인증기관 트렌트의 공개 키를 사용해서 디지털 서명을 검증하고, 밥의 공개 키가 맞다는 것을 확인한다
6) 앨리스는 밥의 공개 키로 메시지를 암호화해서 밥에게 송신한다
7) 밥은 암호문을 자신의 개인키로 복호화해서 앨리스의 메시지를 읽는다

인증기관 트렌트를 이용해서 앨리스가 밥에게 암호문을 보내는 예

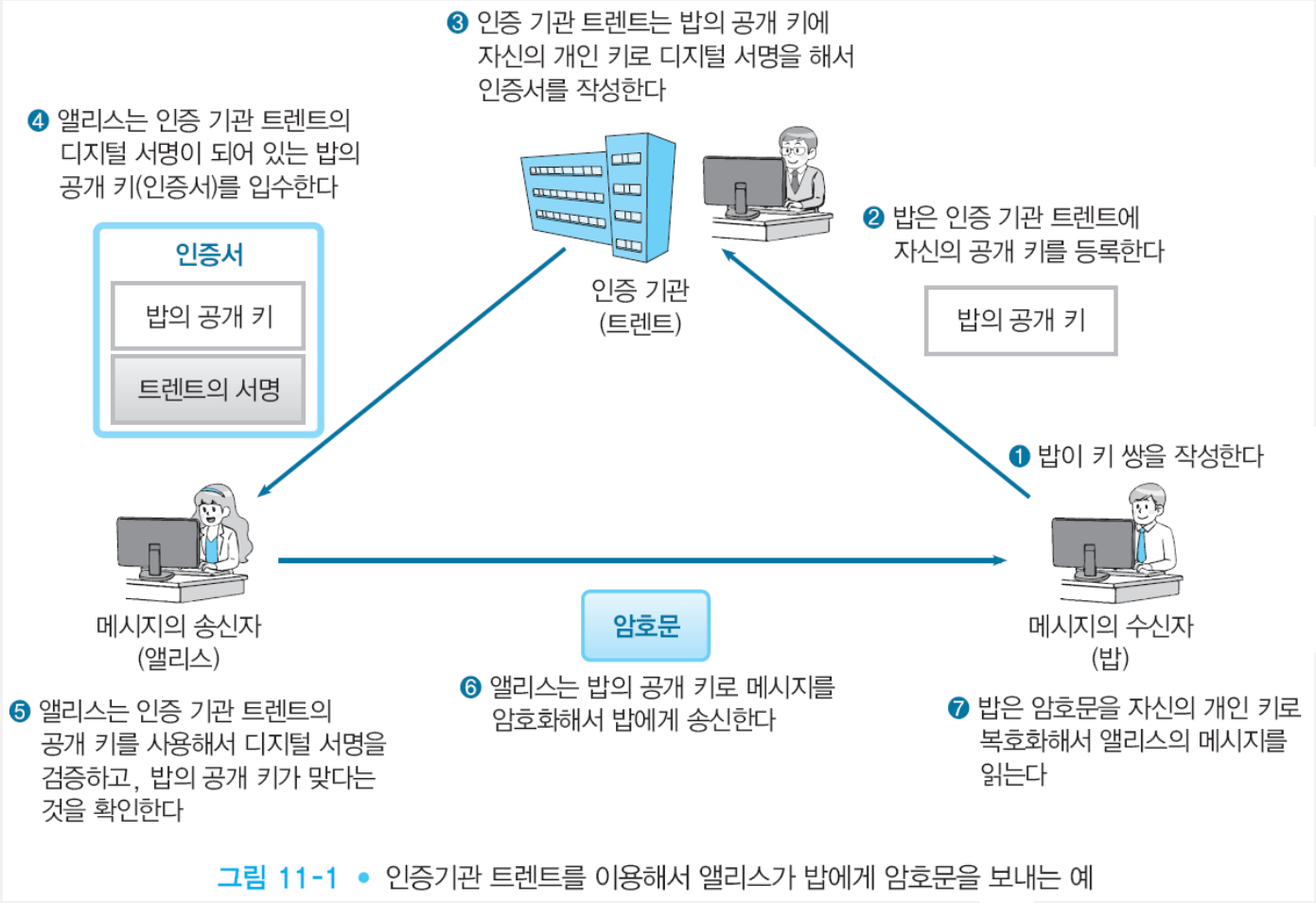

그림 11-1 • 인증기관 트렌트를 이용해서 앨리스가 밥에게 암호문을 보내는 예

칼럼 신분확인과 운용 규정

- 인증기관의 운용규정(CPS :certificate practice statement) 인증기관이 대상자를 인증하는 방법
- 시먼테크사의 인증서 서비스 운용규정
  - 클래스 1: 메일주소로 본인확인
  - 클래스 2: 제 3자 데이터베이스로 본인 확인
  - 클래스 3: 대면인증과 신분 인증서로 본인 확인

Quiz 1 인증 기관은 대단히 바쁘다?

인증기관의 이야기를 읽은 앨리스는 다음과 같이 생각하였다. 인증기관에 대해 조금 이해한 것 같다. 하지만 디지털 서명이 되어 있는 메일을 받을 때마다 인증기관 트렌트에게『디지털 서명의 검증』을 받지 않으면 안되니까 트렌트는 대단히 바쁘겠구나. 이 앨리스의 생각은 잘못된 것이다. 어째서 일까?

Section 02
공인인증서

2.1 공인 인증서 종류

2.2 인증서 표준 규격

2.3 개인 공인 인증서 예

2.4 인증기관 인증서

2.1 공인 인증서 종류

- **범용 공인인증서**
  - 모든 분야에서 이용
  - 인터넷뱅킹, 온라인증권, 전자상거래, 전자정부 민원서비스, 4대 사회보험, 국세청 홈텍스, 전자세금계산서, 전자입찰/조달, 온라인교육, 예비군 등 다양한 분야에서 활용
  - 소정의 수수료

- **용도제한 공인인증서**
  - 은행 및 보험, 신용카드 업무, 정부 민원업무 등 특정분야에서만 이용
  - 해당 기관이 고객에게만 발급
  - 무료

2.2 인증서 표준 규격

- **X.509**
  - 가장 널리 사용
  - ITU(International Telecommunication Union)나 ISO(International Organization for Standardization)에서 규정한 규격
  - 인증서의 생성・교환을 수행할 때 사용
  - 많은 애플리케이션에서 지원

2.3 개인 공인 인증서

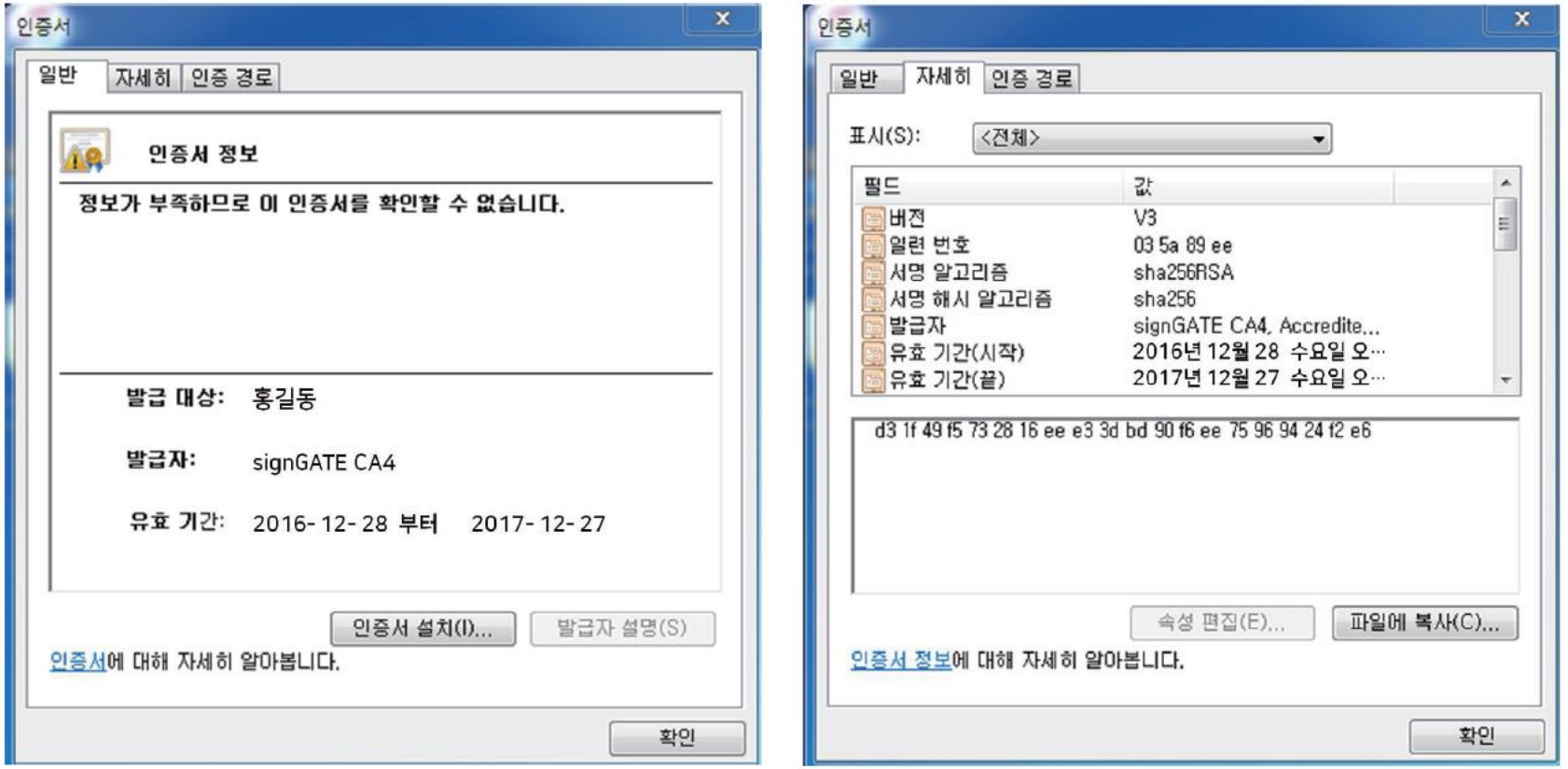

그림 11-2 • 개인 공인 인증서

개인용 인증서 세부 필드

필드	값
버전	V3
일련 번호	03 5a 89 ef
서명 알고리즘	sha256RSA
서명 해시 알고리즘	sha256
발급자	CN = signGATE CA4 OU = AccreditedCA O = KICA C = KR
유효 기간(시작)	2016년 12월 28일 수요일 오후 8:26:15
유효 기간(끝)	2017년 12월 27일 수요일 오후 11:59:59
주체	CN = 홍길동 OU = 중앙우체국 OU = 우체국 OU = 등록기관 OU = licensedCA O = KICA C = KR
공개 키	30 82 01 0a 02 82 01 01 00 c2 2d 87 01 d0 3b 50 d7 a3 ea 72 b4 f3 a5 cf 1e 45 45 7b ac c0 58 6f f1 7b a9 87 18 72 71 c3 b6 d7 8f a8 b9 b8 97 d7 d4 ea ae 1b 00 34 b2 4b c8 b5 5e 45 93 84 54 e7 62 5d d3 2c 7b d2 43 c4 ed a5 7a d5 87 e0 c9 04 a0 ae 98 ae b9 8c 29 62 f8 58 22 46 9b 95 9c 80 d7 fc ab 45 08 91 fc 0c 54 95 74 6f 35 bc 90 47 59 b0 a6 3a 24 64 f3 bc b8 cf 5c 1f b4 3e 16 7c d4 15 a7 01 e0 59 6f ca e3 a5 52 0f 2f 92 db ca 3d a9 9e 3e 96 43 72 f0 26 b3 58 8a 27 74 9b 1c 35 a6 8e 9e eb 96 7e 3c 31 17 59 34 17 90 03 95 5a 5e 35 ef be e7 c9 97 44 1b c8 28 20 2a 98 6a 2f 1f 50 ae c9 e0 c5 2b 50 31 bd 89 6a d6 7e d1 64 13 3e 23 a5 06 eb 64 33 42 1f ed 1f 90 b7 9a 63 c1 3f 0a 8f 04 62 32 b9 76 e0 7f fa e9 1c c5 e2 be c2 01 b9 7f e5 13 26 8d be a9 ba d6 9a 5c 56 89 ef 78 fb f9 3c f1 21 02 03 01 00 01
기관 키 식별자	KeyID=ae 52 fd 0e 0e 01 f8 30 86 37 7e f6 18 c6 49 25 4a 60 09 70 Certificate Issuer: 디렉터리 주소: CN=KISA RootCA 4 OU=Korea Certification Authority Central O=KISA C=KR Certificate SerialNumber=10 0a
주체 키 식별자	67 10 1f 3d 04 47 97 c7 79 22 a2 68 4e a4 77 af 78 04 ad 0d
인증서 정책	[1]Certificate Policy: Policy Identifier=1.2.410.200004.5.2.1.7.1
주체 대체 이름	Other Name: 1.2.410.200004.10.1.1=30 4e 0c 09 ec a0 84 ed 83 9c ec 9d bc 30 41 30 3f 06 0a 2a 83 1a 8c 9a 44 0a 01 01 01 30 31 30 0b 06 09 60 86 48 01 65 03 04 02 01 a0 22 04 20 e9 36 22 bd d2 4a 61 02 d1 e6 84 f2 76 23 d7 cf 20 dc b2 54 f3 a2 41 af 07 d4 61 6f f9 6b 4c 72
CRL 배포 지점	[1]CRL Distribution Point Distribution Point Name: Full Name: URL=ldap://ldap.signgate.com:389/ou=dp6p26866,ou=crldp,ou=AccreditedCA,o=KICA,c=KR
기관 정보 액세스	[1]Authority Info Access Access Method=온라인 인증서 상태 프로토콜 (1.3.6.1.5.5.7.48.1) Alternative Name: URL=http://ocsp.signgate.com:9020/OCSPServer
키 사용	Key Encipherment (20)
지문 알고리즘	sha1
지문	d3 1f 49 f5 73 28 16 ee e3 3d bd 90 f6 ee 75 96 94 24 f2 e6

2.4 인증기관 인증서

표 11-2 • 최상위 인증기관 인터넷진흥원의 인증서

```
Data:
Version: 3 (0x2)
Serial Number: 4 (0x4)
Signature Algorithm: sha1WithRSAEncryption
Issuer: C=KR, O=KISA, OU=Korea Certification Authority Central,
CN=KISA RootCA 1
Validity
Not Before: Aug 24 08:05:46 2005 GMT
Not After : Aug 24 08:05:46 2025 GMT
Subject: C=KR, O=KISA, OU=Korea Certification Authority Central,
CN=KISA RootCA 1
Subject Public Key Info:
Public Key Algorithm: rsaEncryption
RSA Public Key: (2048 bit)
Modulus (2048 bit):
00:bc:04:e4:fa:13:39:f0:34:96:20:6b:6c:68:bb:fa:db:77:ff:27:f7:ac:ec:2f:e7:fd:f0:7f:6d:
6f:8c:2a:cd:25:09:5b:24:f4:a1:68:fc:28:ec:c9:25:e2:ac:ed:de:c8:33:84:f5:b0:a5:09:3a:a7:
b1:47:48:c5:cc:4f:8c:79:9c:f9:06:57:7d:dd:ee:38:f6:cf:14:b2:9c:ea:d3:c0:5d:77:62:f0:47:
0d:b9:1a:40:53:5c:64:70:af:08:5a:c0:f7:cf:75:f9:6c:8d:64:28:1e:20:fe:b7:1b:19:d3:5a:66:
83:72:e2:b0:9b:bd:d3:25:15:0d:32:6f:64:37:94:85:46:c8:72:be:77:d5:6e:1f:28:2f:c7:69:ed:e7:
83:89:33:58:d3:de:a0:bf:40:e8:43:50:ee:dc:4d:6b:bc:a5:ea:a6:c8:61:8e:f5:c3:64:af:06:15:dc:
29:8b:3f:75:8c:bc:71:44:db:fc:ad:b5:17:1d:6d:89:83:cf:c6:33:bd:bf:45:a2:fe:0a:9f:a3:11:
5f:0f:b9:1f:9c:1a:c2:46:cc:9c:28:66:9f:70:26:3c:2e:df:aa:80:fe:8c:c5:04:09:25:
4f:cd:93:47:3c:37:ea:02:67:92:fe:fc:22:24:5c:ac:d2:2c:e0:5c:01:33:8a:c1:19:db
```

2.4 인증기관 인증서

- 인증기관도 인증서를 가지고 있어야함
  - 공인인증서가 인증기관의 개인키로 암호화되어있기 때문에 공인인증서의 내부 정보를 확인하려면 인증기관의 공개키가 필요
  - 인증기관의 공개키를 공개해둠
  - X.509 인증서 양식을 따름

Section 03
공개 키 기반 구조(PKI)

3.1 공개 키 기반 구조(PKI)

3.2 PKI 구성 요소

3.3 인증 기관

3.4 인증기관의 역할

3.5 계층 구조를 갖는 인증서

3.1 공개 키 기반 구조(PKI)

- **공개 키 기반**(public-key infrastructure)
  - 공개 키를 효과적으로 운용하기 위해 정한 많은 규격이나 선택사양의 총칭
    - **PKCS**(Public-Key Cryptography Standards)
      - RSA사가 정하고 있는 규격의 집합
    - **RFC**(Requests for Comments) 중에도 PKI에 관련된 문서
      - 인터넷의 선택사양을 정한다
    - X.509
    - API(Application Programming Interface) 사양서 : PKI 프로그램을 만들 때 사용되는 각 사가 작성한 사양서

3.2 PKI 구성 요소

- 이용자 : PKI를 이용하는 사람
- 인증기관 : 인증서를 발행하는 사람
- 저장소 : 인증서를 보관하고 있는 데이터베이스
- 개체(entity) : 인증서나 키를 주고 받는 주체적인 존재

PKI 구성 요소

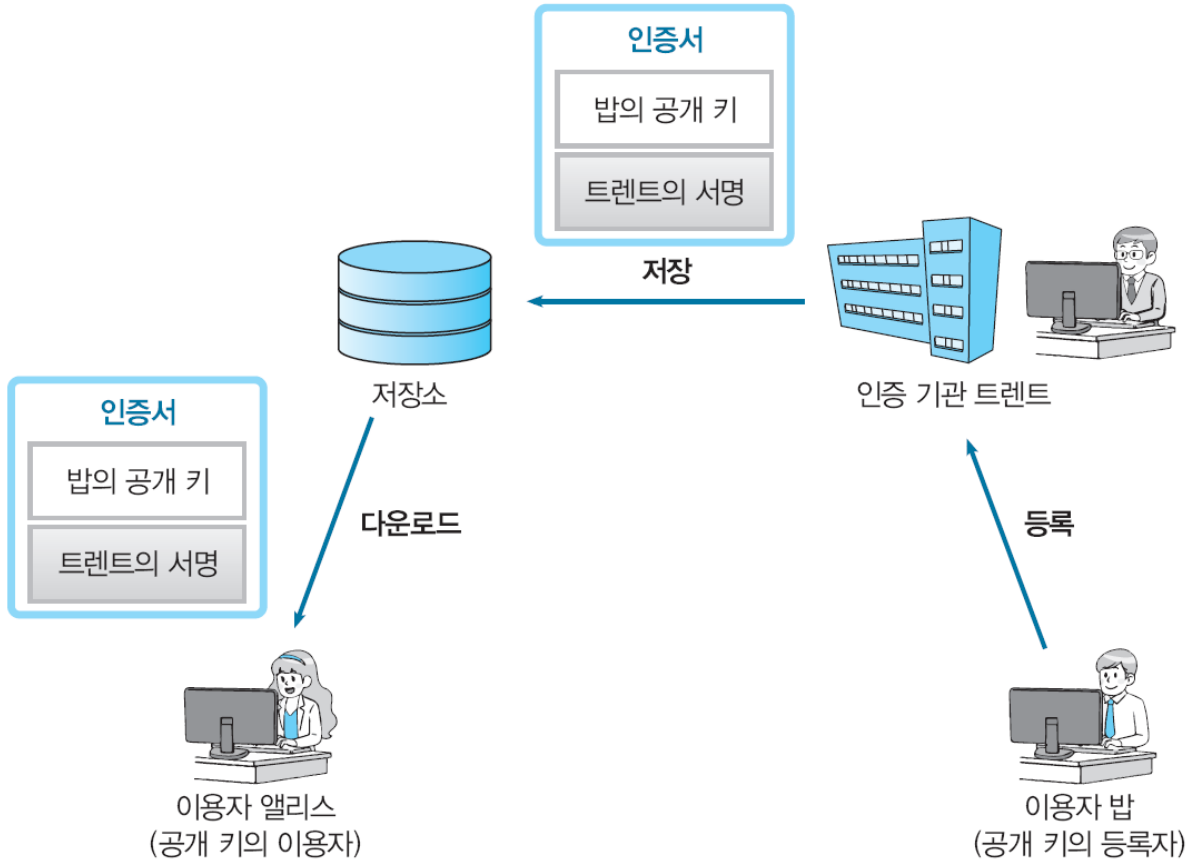

그림 11-3 • PKI 구성 요소

이용자

- PKI를 사용해서 자신의 공개 키를 등록하고 싶어 하는 사람
- 등록되어 있는 공개 키를 사용하고 싶어 하는 사람

공개키를 등록하는 이용자가 하는 일

- 키 쌍을 작성한다(인증기관이 작성하는 경우도 있다)
- 인증기관에 공개 키를 등록한다
- 인증기관으로부터 인증서를 발행 받는다
- 필요할 경우 인증기관에 신청해서 등록한 공개 키를 무효로 한다
- 수신한 암호문을 복호화한다
- 메시지에 디지털 서명을 한다

공개키 사용자가 하는 일

- 메시지를 암호화해서 수신자에게 송신한다
- 디지털 서명을 검증 한다

인증기관

- **인증기관**(certification authority; CA)
  - 인증서의 관리를 행하는 기관
  - 트렌트라고 명명
  - 키 쌍을 작성한다(이용자가 작성하는 경우도 있다)
  - 공개 키 등록 때 등록자의 신원을 인증한다
  - 인증서를 작성해서 발행한다
  - 인증서를 폐지한다

등록기관

- **등록기관**(RA; registration authority)
  - 인증기관의 일 중 「공개 키의 등록과 본인에 대한 인증」을 대행하는 기관

3.3 공인 인증기관

- 미래창조과학부 산하에 민간 최상위 인증기관인 한국인터넷진흥원(KISA)이 있음
- 전자서명법 제 4조의 규정에 의해 지정된 공인인증기관은 5개가 있음(표 10-2 참조)
  - 개인 또는 기업 등의 요청에 따라 공인인증서를 발급
  - 철저한 심사 절차를 통해 발급
  - 법적 효력과 안전성 보장

공인 인증기관

표 11-3 • 공인 인증기관

공인 인증기관	웹페이지	전화번호
한국정보인증(주)	http://www.signgate.com	1577-7337
(주)코스콤	http://www.signkorea.com	1577-7337
금융결제원	http://www.yessign.or.kr	1577-5500
한국전자인증(주)	http://www.crosscert.com	1566-0566
한국무역정보통신	http://www.tradesign.net	1566-2119

한국의 전자서명 인증관리 체계도

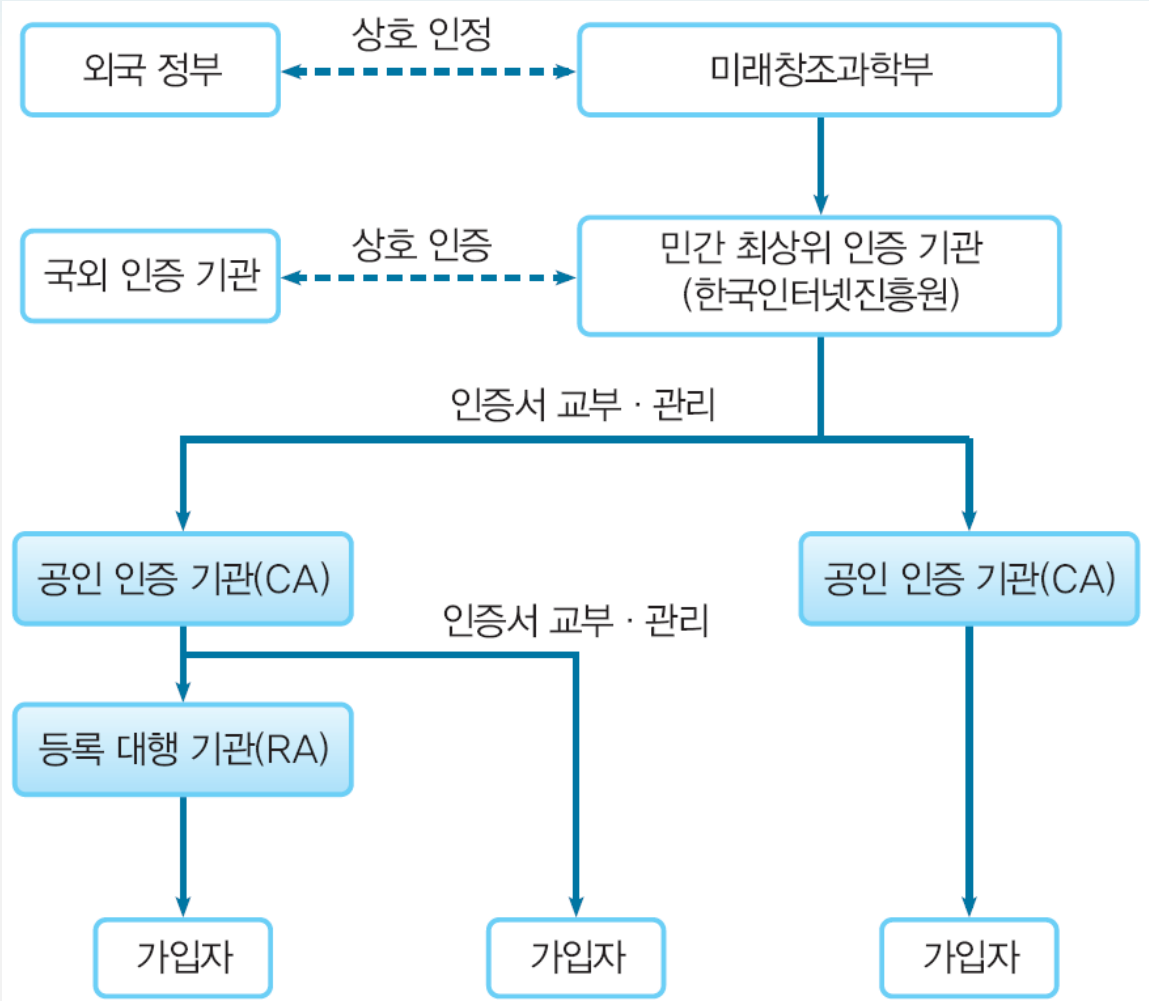

그림 11-4 • 한국의 전자서명 인증관리 체계도

저장소

- **저장소**(repository)
  - 인증서를 보존
  - PKI 이용자가 인증서를 입수할 수 있도록 한 데이터베이스
  - **인증서 디렉토리**

3.4 인증기관의 역할

- 키 쌍의 작성
- 인증서 등록
- 인증서 폐지와 CRL

키 쌍의 작성

- PKI의 이용자가 작성하기
- 인증기관이 작성하기
  - 「개인 키를 이용자에게 보내는」 추가 업무
    - 방법은 RFC 7292 (PKCS #12: Personal Information Exchange Syntax Standard)로 정의

인증서 등록

- 이용자는 인증기관에 인증서 작성을 의뢰
  - 규격은 RFC 2986 (PKCS #10(Certification Request Syntax Standard) 등으로 정의
- 운용 규격(certification practice statement; CPS)에 근거해서 이용자를 인증하고, 인증서를 작성
  - 인증서 형식은 PKCS #6(Extended-Certificate Syntax Standard)나 X.509로 정의

인증서 폐지와 CRL

- 인증서를 **폐지**(revoke)해야 할 경우
  - 이용자가 개인 키를 분실 혹은 도난
- 인증서 폐지 목록(CRL: certificate revocation list)을 작성
- CRL : 폐지된 인증서의 일련 번호의 목록에 대해 인증기관이 디지털 서명을 붙인 것
- 인증기관의 최신 CRL을 조사해서 그 인증서 유효성 확인 필요

3.5 계층 구조를 갖는 인증서

- **인증기관의 공개키 인증서를 생성**
  - 인증기관의 공개키에 대해 다른 인증기관이 디지털 서명을 하는 것으로 인증기관의 공개키를 검증할 수 있다

회사 내의 사내 PKI

서울 본사(서울 본사 인증기관)

↓

충남 지사(충남 지사 인증기관)

↓

논산 지점(논산 지점 인증기관)

- **루트 CA**
  - 최상위 인증기관
- **셀프 서명**(self-signature)
  - 자기 자신의 공개 키에 대해서 자신의 개인 키로 서명하는 디지털 서명

계층구조를 갖는 인증기관

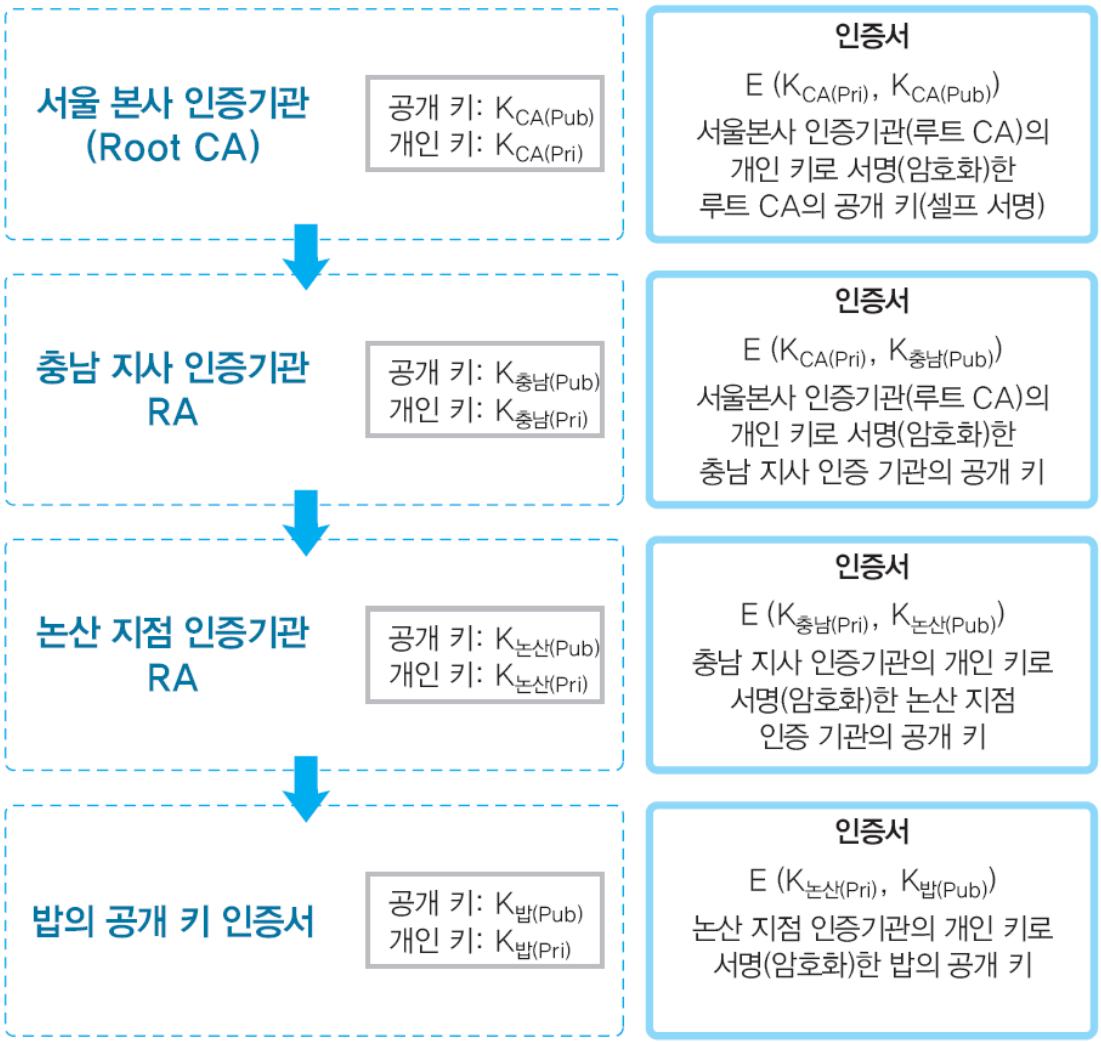

그림 11-5 • 계층구조를 갖는 인증기관

앨리스가 밥의 바른 공개키를 얻는 과정(I)

- 서울 본사 인증기관의 바른 공개키를 앨리스가 가지고 있다고 가정
- 앨리스는 충남지사 인증기관의 공개키인증서를 입수 (서울본사 인증 기관의 디지털 서명이 붙어있음)
- 앨리스는 서울본사 인증기관의 바른 공개키로 디지털 서명을 검증, 검증에 성공하면 충남지사 인증기관의 바른 공개키를 입수한 것임
- 앨리스는 논산 지점 인증기관의 공개키 인증서를 입수 (충남지사 인증기관이 디지털 서명이 붙어 있음)

앨리스가 밥의 바른 공개키를 얻는 과정(II)

- 앨리스는 충남지사 인증기관의 바른 공개키로 디지털 서명을 검증, 검증에 성공하면 논산지점 인증기관의 바른 공개키를 입수한 것임
- 앨리스는 논산지점의 직원 밥의 공개키 인증서를 입수 (논산지점 인증기관의 디지털 서명이 붙어있음)
- 앨리스는 논산 지점 인증기관의 바른 공개키로 디지털 서명을 검증, 만약 검증에 성공하면 앨리스는 논산 지점의 직원 밥의 바른 공개키를 입수한 것임

앨리스가 밥의 바른 공개키를 얻는 과정(III)

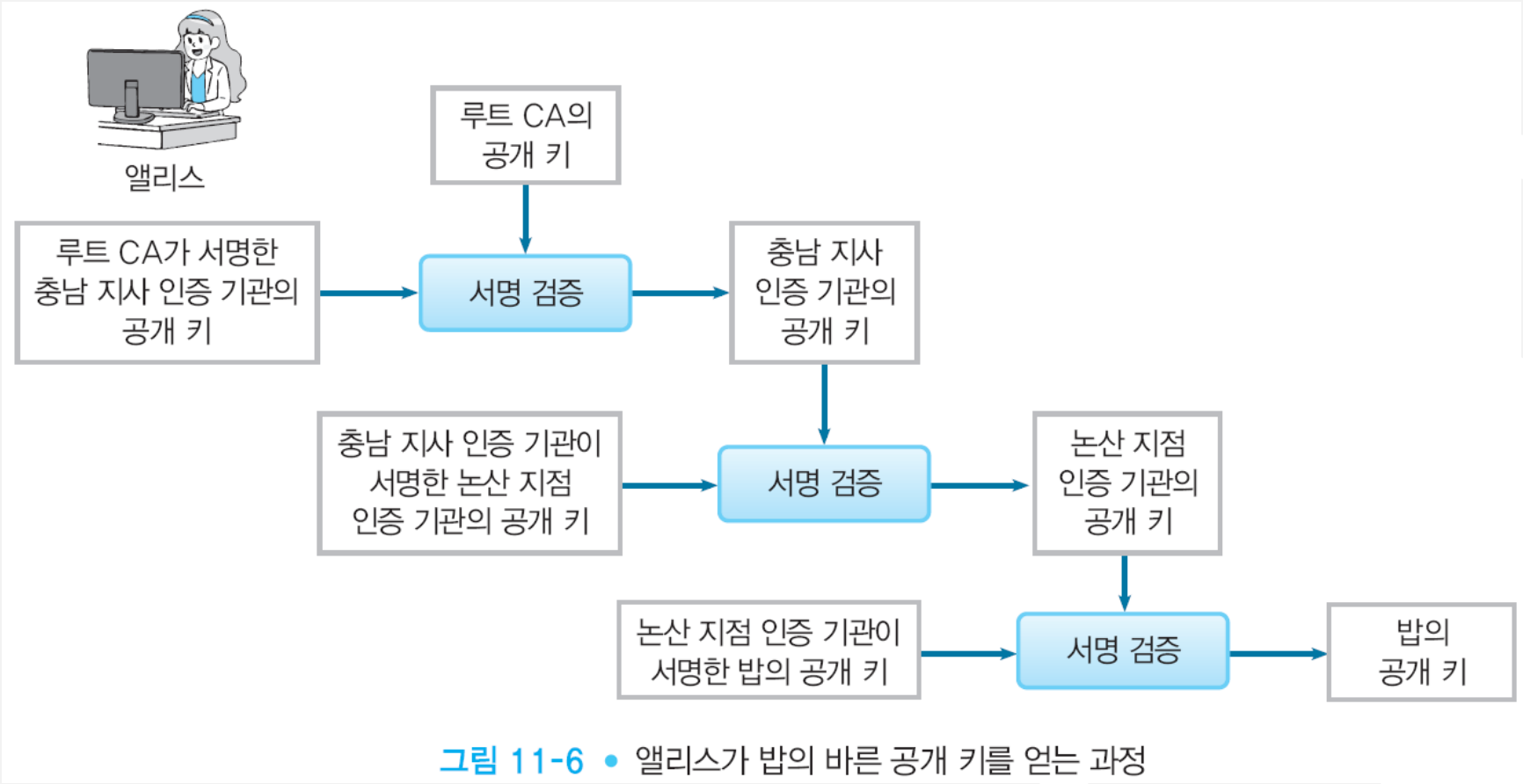

그림 11-6 • 앨리스가 밥의 바른 공개 키를 얻는 과정

다양한 PKI

- 누구나 인증기관이 될 수 있고 실제로 세계에는 무수히 많은 인증기관이 존재
  - 인증기관은 공개키에 디지털 서명을 할 수 있으면 됨
- 사내 이용 방법
  - 인증기관의 계층을 회사의 조직 계층에 적용
  - 부서별로 PKI 운영하고 상호 인증
- 우리나라 PKI
  - **한국인터넷진흥원 전자서명인증관리센터**에서 관리
  - http://rootca.kisa.or.kr
  - 인증기관의 계층이나, 운용 규약, 공개 키의 등록・인증서 발행 등을 규정

Section 04
인증서에 대한 공격

4.1 공개 키 등록 이전 공격

4.2 닮은 사람을 등록하는 공격

4.3 인증 기관의 개인 키를 훔쳐내는 방법

4.4 공격자 자신이 인증 기관이 되는 공격

4.5 CRL의 허점을 찌르는 공격 1

4.6 CRL의 허점을 지르는 공격 2

4.7 Superfish

4.1 공개 키 등록 이전 공격

- 인증기관이 디지털 서명을 수행하기 이전에 적극적 공격자 맬로리가 공개 키를 자신의 것과 살짝 바꿔치기 한다
- 인증기관은「밥의 정보」와「맬로리의 공개 키」의 조합에 대해 디지털 서명을 하게 된다
- 밥이 공개키를 인증기관에 등록할 때 인증 기관의 공개키를 사용해서 밥의 공개키를 암호화 한다
- 인증기관이 밥 본인을 확인할 때 공개키의 핑거프린트도 함께 받아 밥을 확인

ID 베이스 암호

- 평문과 ID만으로 암호화할 수 있는 암호 알고리즘
- ID로 부터 공개키를 직접 만드는 방법
- ID로 메일 주소 사용 : 앨리스가 밥의 메일 주소를 근거로 밥의 공개키를 만들고 그것을 사용해서 암호문을 작성 (ID의 신뢰를 이용하여 인증서를 불필요하게 한 것)
- 인증기관은 불필요하지만 개인키 생성 기관(PKG : Private Key Generator)이 필요
- 개인키 생성 기관은 ID로 개인키를 생성하고 올바른 수신자에게 그 개인키를 안정하게 배송하는 일을 수행

4.2 닮은 사람을 등록하는 공격

- **오인하기 쉬운 사용자 정보**를 사용
  - **Name = Bob**
    을
  - **Name = BOB**
- 이 공개 키는 이름은 BOB으로 되어 있지만, 맬로리의 공개 키
- 맬로리는 밥의 행세를 하며 **Name = BOB**으로 되어 있는 인증서를 앨리스에게 보낸다
- 인증서에 포함된 정보가 그 개인의 정보인지 인증기관이 바르게 확인할 필요가 있다
- 본인 확인에 실패했을 경우에는 인증서를 발행하지 말아야 한다는 규칙이 필요

4.3 인증 기관의 개인 키를 훔쳐내는 방법

- **인증기관의 개인 키를 훔쳐낸다**

- 인증기관의 개인 키가 도난당했다면(누설되었다면), 인증기관은 자신의 키가 누설되었다는 것을 **CRL을 사용해서 이용자에게 통지**

4.4 공격자 자신이 인증 기관이 되는 공격

- **맬로리 자신이 인증기관이 된다**
- 인증기관이 된 맬로리는 자신의 공개 키라도 「이것은 밥의 공개 키이다」라고 주장하는 인증서를 자유롭게 발행
- **인증기관을 신뢰할 수 없으면 인증서가 아무리 바르더라도 그 공개 키를 사용해서는 안 된다**

4.5 CRL의 허점을 찌르는 공격 1(I)

- 공격자 맬로리는 **CRL이 도착할 전에 빠른 공격**을 시도
  - 맬로리는 밥의 컴퓨터에 몰래 들어가 밥의 개인키를 훔친다
  - 맬로리는 밥의 행세를 하며 앨리스에게 보낼 메일을 쓴다
  - 돈을 맬로리의 계좌에 이체하게 하는 메일이다
  - 조금 전 훔친 밥의 개인키를 이용하여 메일에 디지털 서명을 해둔다.
  - 다음날 밥은 자신의 컴퓨터에 침입자가 있었고 개인키가 도난당했다는 것을 감지
  - 밥은 인증기관 트렌트에게 연락하여 자신의 공개키가 무효가 되었다는 것을 알린다.
  - 트렌트는 방의 키를 무효화하는 CRL을 작성하여 배포
  - 앨리스는 밥으로부터 온 메일을 읽고 지정된 계좌에 돈을 이체하려고하며 그전에 앨리스는 디지털 서명을 검증
  - 디지털 검증에 성공한 앨리스는 돈을 이체한다
  - 조금 후 앨리스는 트렌트로부터 CRL을 받고 인증서가 무효였다는 것을 알고 충격을 받는다

4.5 CRL의 허점을 찌르는 공격 1(II)

- 방어방법
  - 공개 키가 무효가 되면 가능한 빨리 인증기관에 전한다(밥)
  - CRL은 신속하게 발행한다(트렌트)
  - CRL은 정확히 갱신한다(앨리스)
  - 공개 키를 이용하기 전에는 공개 키가 무효가 되지 않았나를 재확인한다(앨리스)

4.6 CRL의 허점을 지르는 공격 2

- CLR의 허점을 찌름으로써 부인의 가능성
  - 밥이 앨리스로부터 돈을 뜯어낼 계획수립
  - 밥은 가명을 써서 계좌 X-5897을 개설
  - 앨리스에게 송금 요청을 하고 자신의 서명을 붙인다
  - 트렌트에게 개인키가 도난 당했다고 보고한다
  - 앨리스에게 CRL이 도착하기 전에 앨리스가 해당 금액을 계좌 X-5897로 송금을 했다면 밥은 예금을 인출한다
  - 앨리스가 나중에 CRL을 받고 밥에게 항의한다
  - 밥은 자신의 개인키가 도난 당했다고 주장하고 돈을 착복한다
- 인증서가 무효가 되어있는지 어떤지 신속하게 확인하기 위한 프로토콜 : RFC 2560(X.509 Internet Public Key Infrastructure Online Certificate Status Protocol)

4.7 Superfish

- 2015년에 PC 벤더 Lenovo사의 컴퓨터에서 중대한 사건이 발생
  - 컴퓨터에 프리인스톨 되어 있던 Superfish라는 애드웨어가 보안상의 문제 일으킴
  - 통신을 가로채 개인 정보를 수집하여 사용자의 인터넷 이용 맞춤 광고를 내보내는 소프트웨어
  - 루트 인증서를 인스톨하여 Web사이트와 Web브라우저 사이에 들어간 후, 방문한 Web사이트 인증서를 바꿔치기 하여 Web브라우저에 제시
  - 전형적인 중간자 공격으로 통신 갈취
  - 임의의 웹사이트에 대한 인증서를 동적으로 발생시키기 위하여 소프트웨어 속에 서명을 하기 위한 개인키를 포함
  - 악의있는 소프트웨어가 Superfish를 이용하여 임의의 거짓 사이트를 진짜 사이트로 보이게 하는 인증서를 만드는 것이 가능

Section 05
인증서에 대한 Q&A

5.1 인증서의 필요성

5.2 독자적인 인증 방법을 사용하는 것이 안전한 것이 아닌가?

5.3 인증 기관을 어떻게 신뢰할 것인가?

5.1 인증서의 필요성

- **의문:** 인증서의 필요성을 모르겠다. 인증기관의 인증서를 사용해서 공개 키를 입수하는 것과, 공개 키만을 받는 것과는 같은 것이 아닌가?
- **답**
  - 신뢰할 수 없는 경로(예를 들면 메일)로 공개 키를 입수하는 경우, 중간자(man-in-the-middle)공격이 가능해 진다.
  - 인증기관으로부터 인증서를 입수하면 중간자 공격(man-in-the-middle attack)의 가능성을 줄일 수 있다.

인증기관의 필요성

- 신뢰할 수 있는 공개 키를 입수할 수 있다면 인증기관은 불필요하다.
- 신뢰할 수 있는 인증기관의 공개 키를 가지고 있고, 인증기관의 본인 확인을 신뢰한다면, 그 인증기관이 발행한 인증서에 의해 입수한 공개 키는 신용할 수 있다.

5.2 독자적인 인증 방법을 사용하는 것이 안전한 것이 아닌가?

- **의문:** 인증서 형식이든 PKI든 공개되어 있는 기술을 사용하는 것에 불안을 느낀다. 공개되어 있는 기술을 사용한다는 것은 공격자에게 공격을 위한 정보를 제공하는 것이 된다고 생각한다. 그것보다는 사내에서 독자적으로 개발한 비밀 인증 방법을 사용하는 편이 안전하지 않을까?
- **답**
  - 그렇지 않다.
  - 비밀 인증 방법을 독자 개발하는 것은「감추는 것에 의한 보안」(security by obscurity)라는 전형적인 잘못이다.

5.3 인증 기관을 어떻게 신뢰할 것인가?

- **의문:** 인증기관의 기능은 대강 이해를 했지만, 결국 맴도는 것 같은 느낌이 든다. 공개 키를 신뢰하기 위해서는 인증서를 발행한 인증기관을 신뢰해야 하는데, 그렇다면 인증기관은 어떻게 신뢰하는 것일까?
- **답:**
  - 이 의문은 정당하다.
  - 이 의문은「신뢰」가 어떻게 형성 되는가 하는 본질적인 문제와 관계되어 있기 때문이다.

06 이장의 정리

- 디지털 서명이라는 기술을 신뢰할 수 있고 인증서를 사용하면 신뢰할 수 있는 공개키를 입수할 수 있다.
- 그러나 그 서명을 수행한 인증기관을 신뢰할 수 있는지, 인증서는 CRL에 올라있지 않은지, 자신이 사용하고 있는 소프트웨어에 들어 있는 인증서는 올바른 것인지, 라는 많은 전제가 신뢰의 그림자 뒤에 숨어 있다.
- **아무것도 신뢰할 수 없는 상태에서 신뢰를 만들어 내는 기술은 아직 개발되어 있지 않다.**

Quiz 2 인증서의 기초 지식

- 인증서와 PKI에 관한 다음 문장 중 바른 것에는○, 틀린 것에는 X를 표시하시오.

(1) 인증서는 이용자의 공개 키를 인증기관이 암호화 한 것이다.

(2) 인증서에 포함되어 있는 공개키가 바른지 어떤지를 확인하기 위해서는 인증기관의 공개키가 필요하다.

(3) 전 세계에서 발행되고 있는 인증서는 모두 인증기관의 계층을 따라가면 유일한 루트 CA까지 도달한다.

(4) 자신의 개인키가 누설되었다는 것을 안 이용자는 공개 키를 등록하고 있는 인증기관에 서둘러 연락할 필요가 있다.

(5) 이용자는 인증기관으로부터 정기적으로 CRL을 입수할 필요가 있다.