

# 11장 키 관리(Key Management)

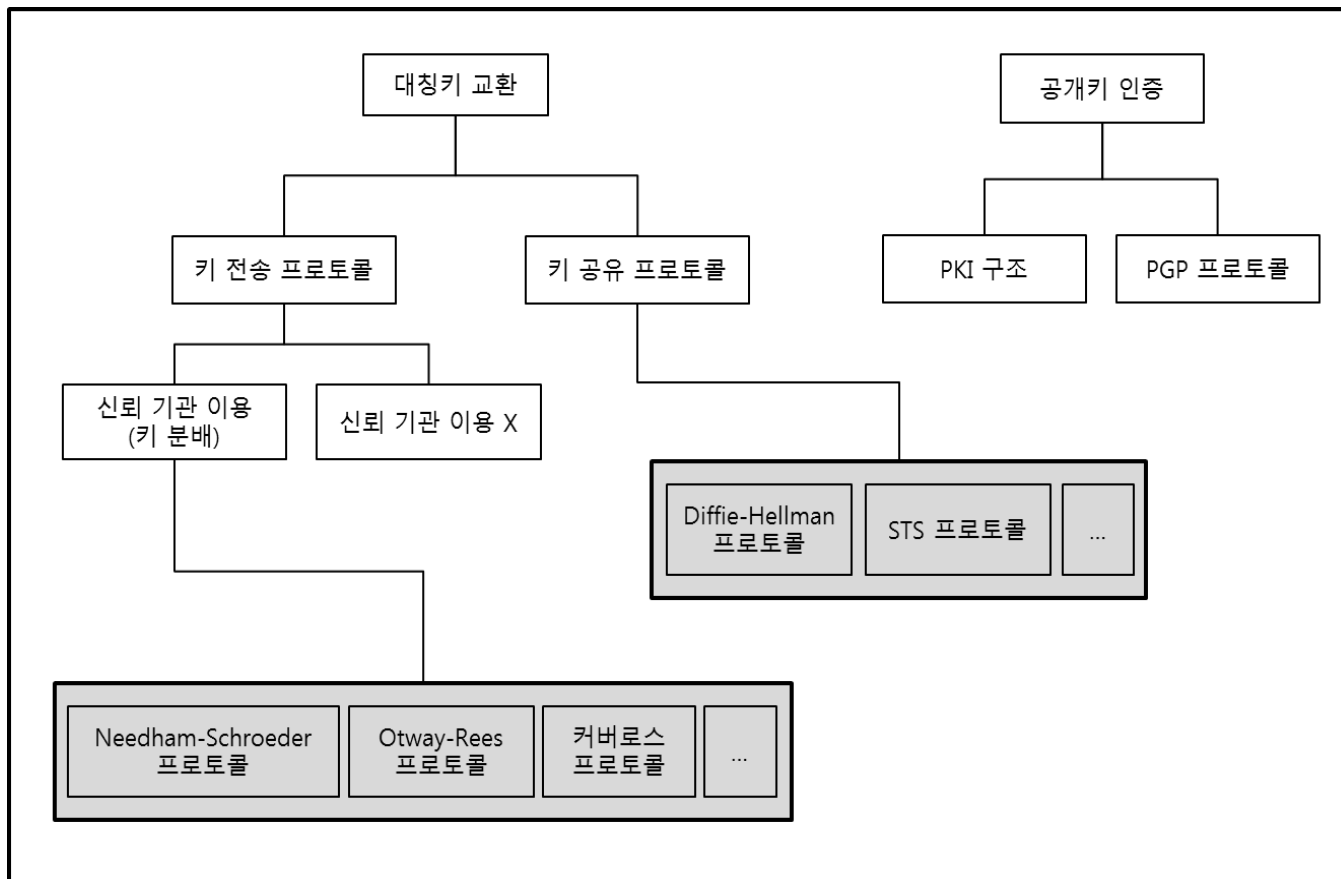
---

정보보호이론

Spring 2015

# 11.1 개요

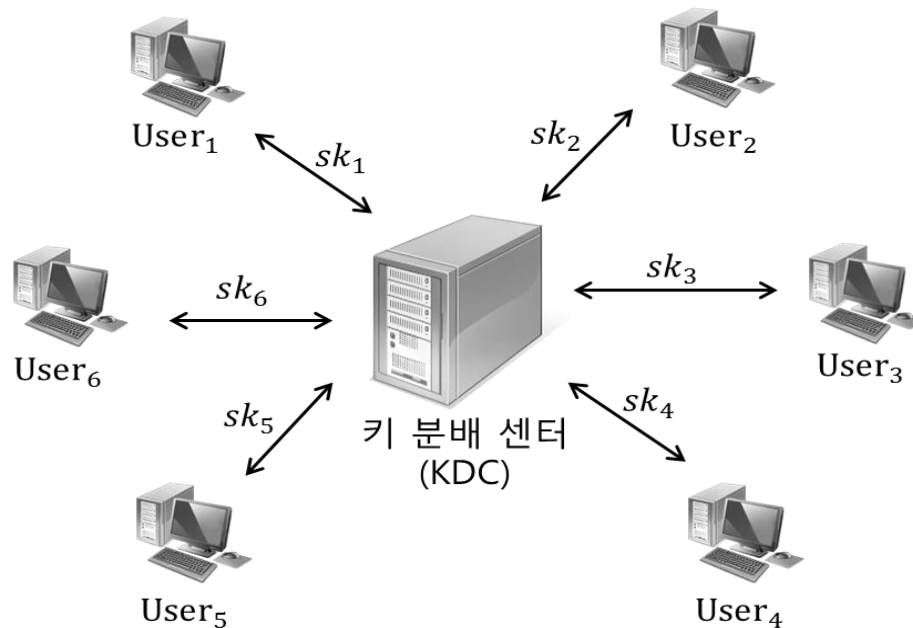
## ■ 키 관리 방법



# 11.2 키 분배(Key Distribution)

## ■ 대칭키를 이용한 키 분배

- ✖  $n$ 명 :  $\frac{n(n-1)}{2}$  키 필요, 사용자는  $(n-1)$  관리
- ✖ 제 3자인 키 분배 센터(KDC)를 이용



# 11.2 키 분배(Key Distribution)

## ■ 대칭키를 이용한 키 분배

✕ 키 분배 센터를 이용한 키(세션 키) 분배 방법



Alice

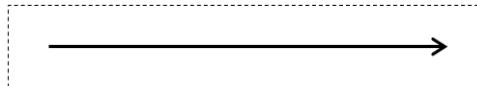


KDC

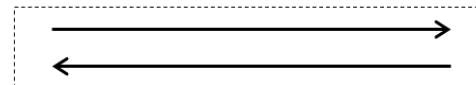


Bob

① Bob과의 세션키 발급 요청



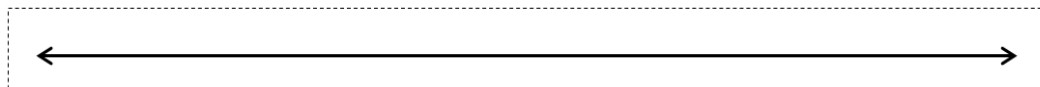
② Alice의 요청 통보 및 Bob의 동의



③ 세션 키 생성 및 전송



④ 비밀 통신



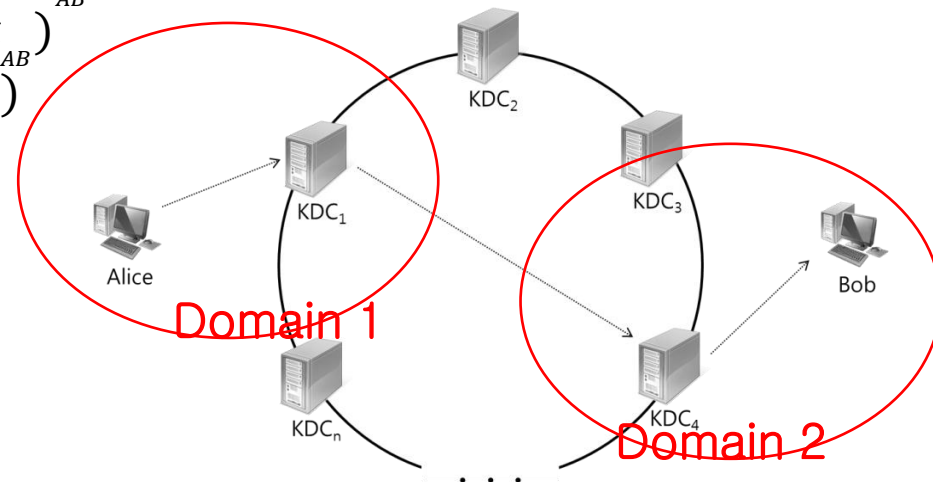
# 11.2 키 분배(Key Distribution)

## ■ 대칭키를 이용한 키 분배

✗ 분산된 키 분배 센터를 이용한 키 분배 방법

▶ 평등 다중(Flat Multiple) 구조의 키 분배 센터

1. Alice  $\rightarrow$  KDC<sub>1</sub>: 세션키 생성 요청
2. KDC<sub>1</sub>  $\rightarrow$  KDC<sub>4</sub>: Alice의 요청 전달
3. KDC<sub>4</sub>  $\rightarrow$  Bob : Alice의 요청 알림
4. Bob  $\rightarrow$  KDC<sub>4</sub> : 동의
5. KDC<sub>4</sub>  $\rightarrow$  KDC<sub>1</sub> : Bob의 동의 알림
6. KDC<sub>1</sub>  $\rightarrow$  KDC<sub>4</sub> : 세션키( $k_{AB}$ ) 전송
7. KDC<sub>1</sub>  $\rightarrow$  Alice :  $E_{sk_A}(k_{AB})$
8. KDC<sub>4</sub>  $\rightarrow$  Bob :  $E_{sk_B}(k_{AB})$

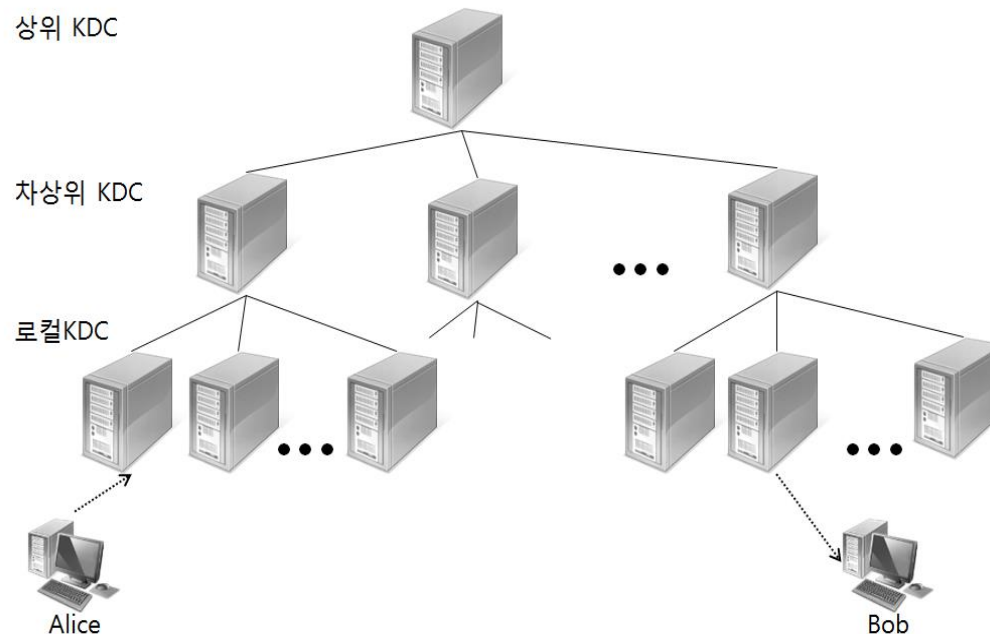


# 11.2 키 분배(Key Distribution)

## ■ 대칭키를 이용한 키 분배

✘ 분산된 키 분배 센터를 이용한 키 분배 방법

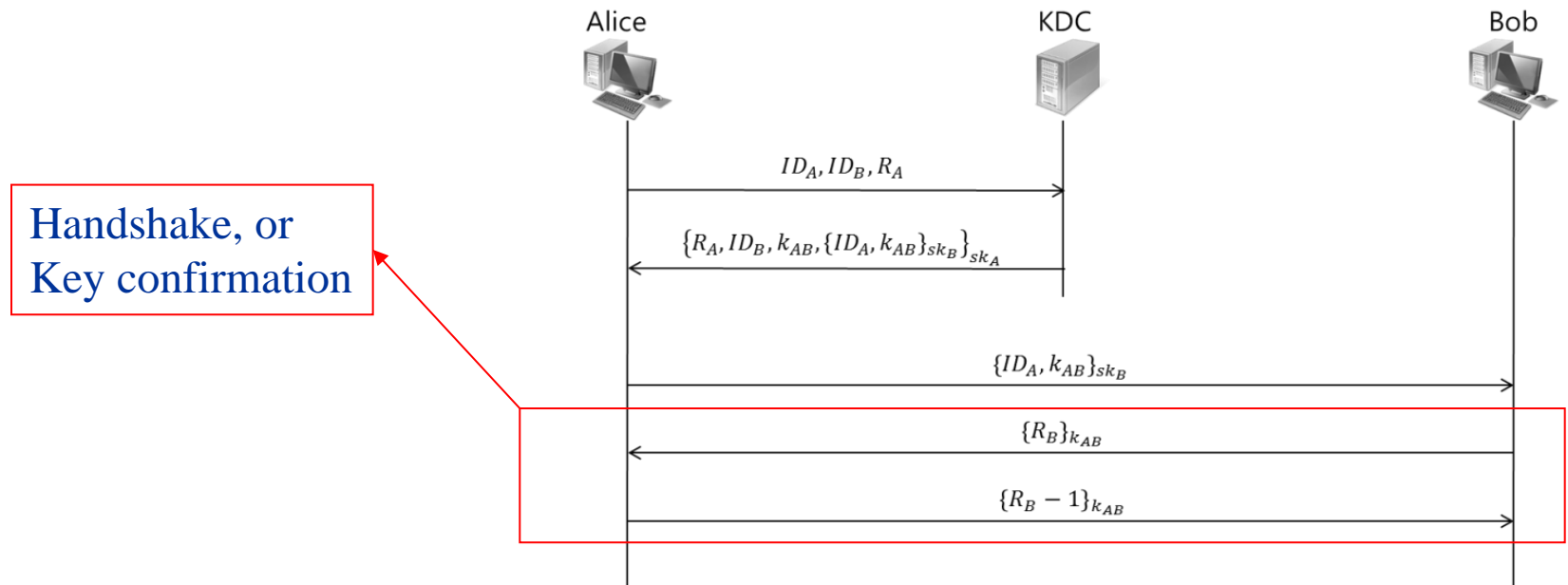
▶ 계층 다중(Hierarchical Multiple) 구조의 키 분배



# 11.2 키 분배(Key Distribution)

## ■ Needham-Schroeder 프로토콜

1. Alice  $\rightarrow$  KDC :  $ID_A, ID_B, R_A$
2. KDC  $\rightarrow$  Alice :  $E_{sk_A}(R_A, ID_B, k_{AB}, E_{sk_B}(ID_A, k_{AB}))$
3. Alice  $\rightarrow$  Bob :  $E_{sk_B}(ID_A, k_{AB})$
4. Bob  $\rightarrow$  Alice :  $E_{k_{AB}}(R_B)$
5. Alice  $\rightarrow$  Bob :  $E_{k_{AB}}(R_B - 1)$



# 11.2 키 분배(Key Distribution)

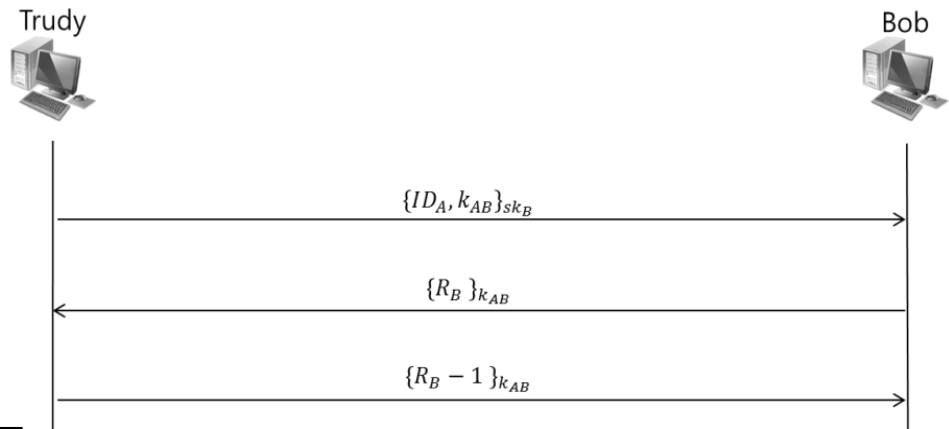
## ■ Needham-Schroeder 프로토콜 재전송 공격(Replay Attack)

✗ Trudy with old session key  $k_{AB}$

3. Trudy  $\rightarrow$  Bob :  $E_{sk_B}(ID_A, k_{AB})$

4. Bob  $\rightarrow$  Alice(Trudy) :  $E_{k_{AB}}(R_B)$

5. Trudy  $\rightarrow$  Bob :  $E_{k_{AB}}(R_B - 1)$



✗ 재전송 공격 방지

▶ 세션키에 새로움 제공

$\rightarrow E_{sk_B}(ID_A, k_{AB}, T)$  in Step 2



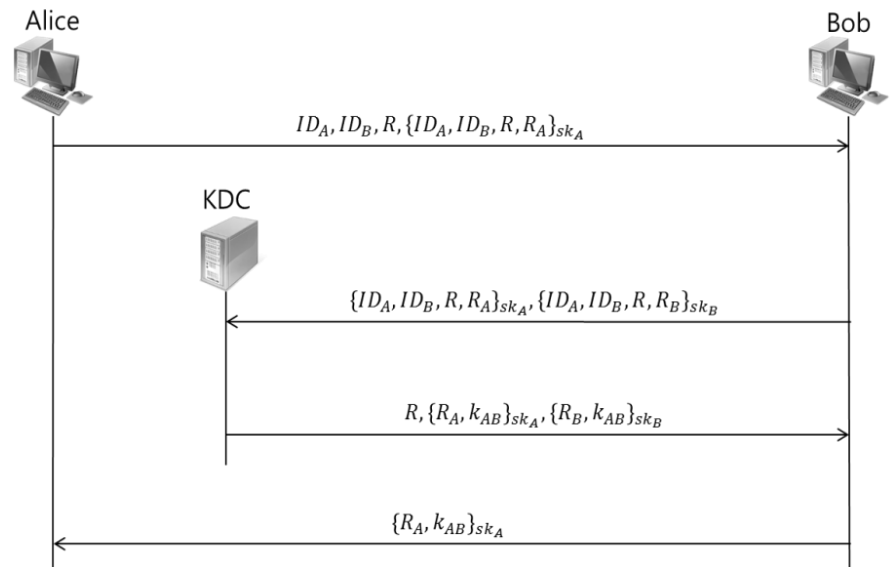
# 11.2 키 분배(Key Distribution)

## ■ Otway-Rees 프로토콜

1. Alice  $\rightarrow$  Bob :  $ID_A, ID_B, R, E_{sk_A}(ID_A, ID_B, R, R_A)$
2. Bob  $\rightarrow$  KDC :  $E_{sk_A}(ID_A, ID_B, \boxed{R}, R_A), E_{sk_B}(ID_A, ID_B, \boxed{R}, R_B)$
3. KDC  $\rightarrow$  Bob :  $(\boxed{R}, E_{sk_A}(R_A, k_{AB}), E_{sk_B}(R_B, k_{AB}))$
4. Bob  $\rightarrow$  Alice :  $E_{sk_A}(R_A, k_{AB})$

- ▶  $R$  : Index number
- ▶  $R_A$  : Alice 확인
- ▶  $R_B$  : Bob 확인

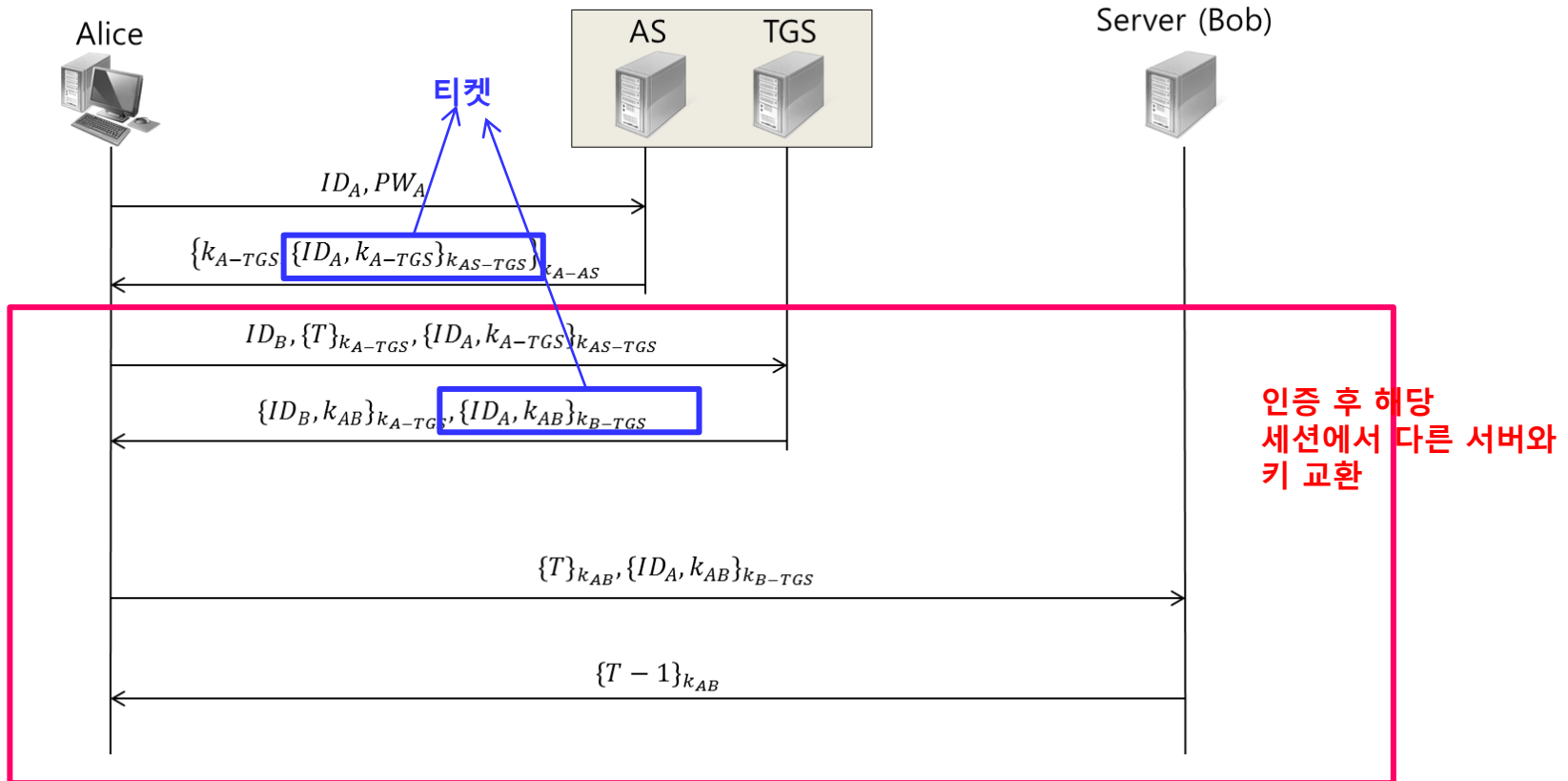
✗ 재전송 공격?



# 11.2 키 분배(Key Distribution)

## ■ 커버로스(Kerberos)

✂ MIT에서 네트워크 내부 사용자 인증

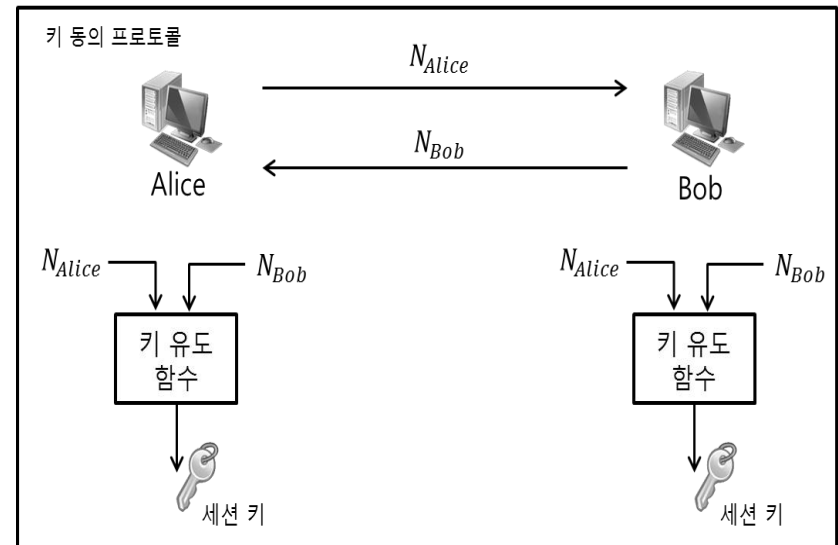


# 11.3 키 교환(Key Exchange or Establishment)

## ■ 키 전송(Key Transport) 프로토콜



## ■ 키 동의(Key Agreement) 프로토콜



# 11.3 키 교환(Key Exchange or Establishment)

---

## ■ 키 교환 프로토콜의 안전성

### ✖ 전방향 안전성(Forward Secrecy)

- ▶ 사용자의 비밀키를 알고 있는 공격자라도 정직한 구성원 간에 성공적으로 확립된 이전의 세션키에 대한 어떠한 정보도 얻을 수 없어야 함

1. Alice  $\rightarrow$  Bob :  $E_{pk_B}(k_n)$
2. Eve :  $\{E_{pk_B}(k_1), E_{pk_B}(k_2), \dots, E_{pk_B}(k_n)\}$  저장 &  $pk_B$  노출  
 $\rightarrow$ 이전 세션의 정보가 노출

### ✖ 기지-키 안전성(Known-Key Secrecy)

- ▶ 여러 세션에서 얻은 세션키들을 이용해도 노출되지 않은 세션 키들의 기밀성에는 영향을 주지 않아야 함

1. 새로운 세션키  $k = h(k', ID_A, ID_B)$
2. Eve : 세션키  $k' \rightarrow k$  계산

# 11.3 키 교환(Key Exchange or Establishment)

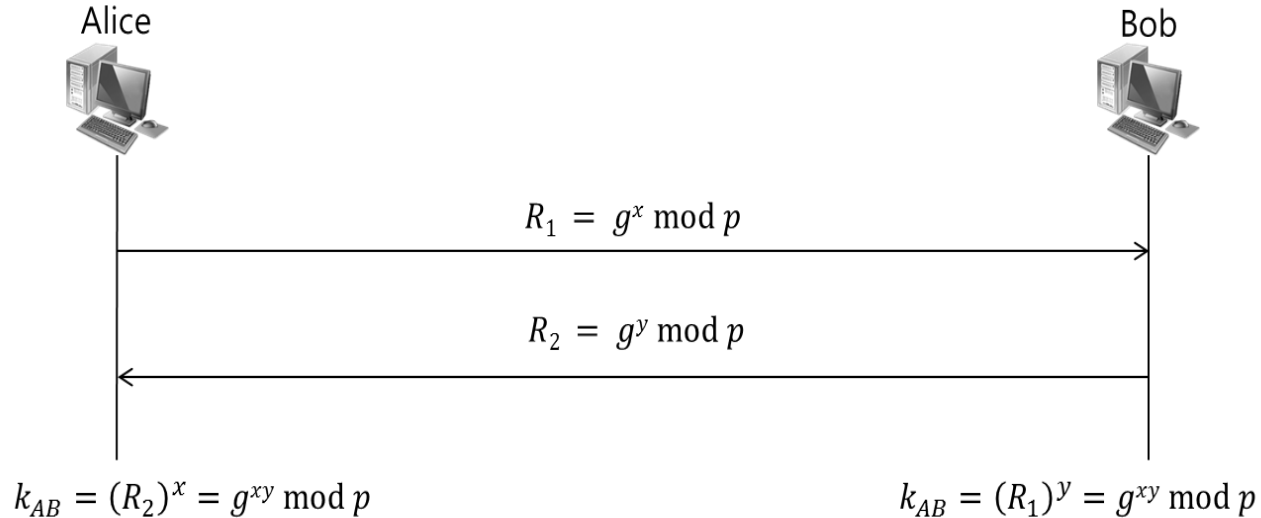
---

## ■ 키 교환 프로토콜의 안전성

- ✗ 세션 상태 노출에 대한 안전성(Security against Session State Reveal)
  - ▶ 공격자가 세션키를 만드는 데 사용되는 난수 값을 가지고서도 세션키를 알 수 없어야 함
  - ▶ 롱텀키(long-term key)인 비밀키 보다는 일회용 비밀 값인 난수들이 더욱 쉽게 노출될 수 있다는 관점
- ✗ 비밀키 사용 위장에 대한 안전성(Security against Key Compromise Impersonation)
  - ▶ Eve가 Alice의 비밀키로 Bob으로 위장함을 방지
- ✗ 파트너 혼돈 공격에 대한 안전성(Security against Unknown Key Share)
  - ▶ Alice와 Bob이 동일한 세션키를 계산했다면 Alice는 현재 Bob과 키 교환을 하고 있다고 인식해야 하며, Bob 또한 Alice와 키 교환을 하고 있다고 인식해야 함

# 11.3 키 교환(Key Exchange or Establishment)

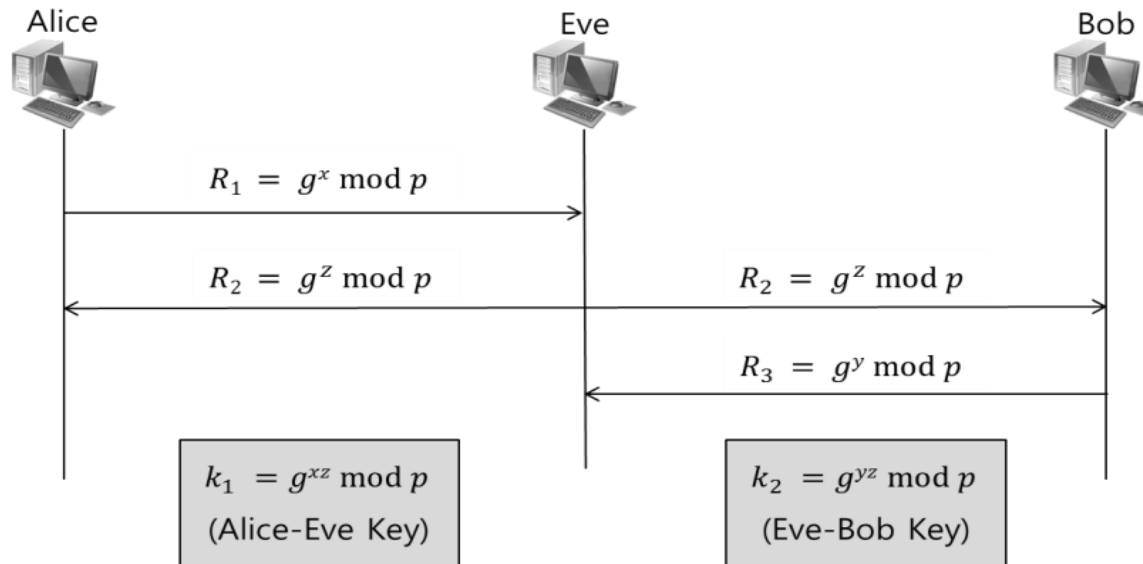
## ■ Diffie-Hellman 동의 프로토콜



# 11.3 키 교환(Key Exchange or Establishment)

## ■ Diffie-Hellman 동의 프로토콜

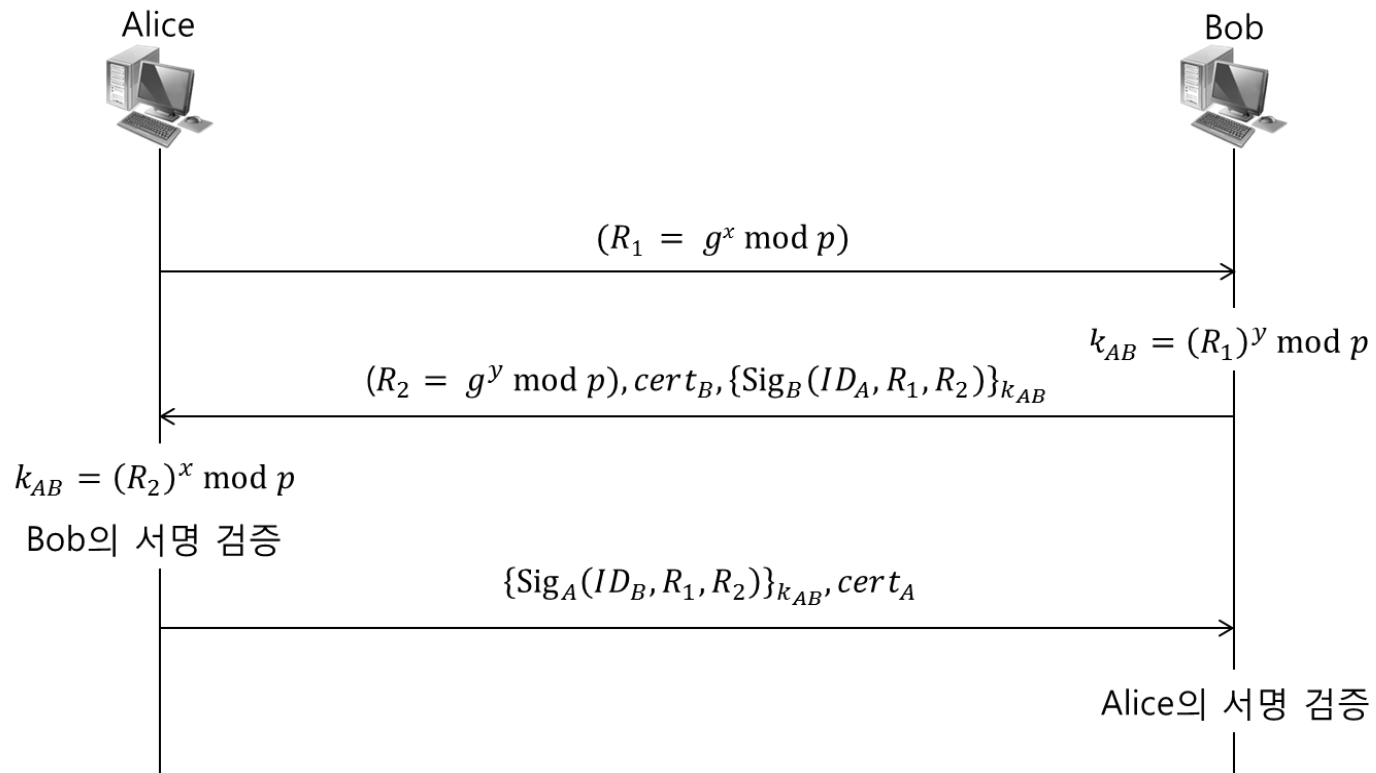
✗ 중간자 공격(Man-in-the-Middle Attack)



✗  $R_1$ 이 Alice의 인증서,  $R_2$ 가 Bob의 인증서인 경우?

# 11.3 키 교환(Key Exchange or Establishment)

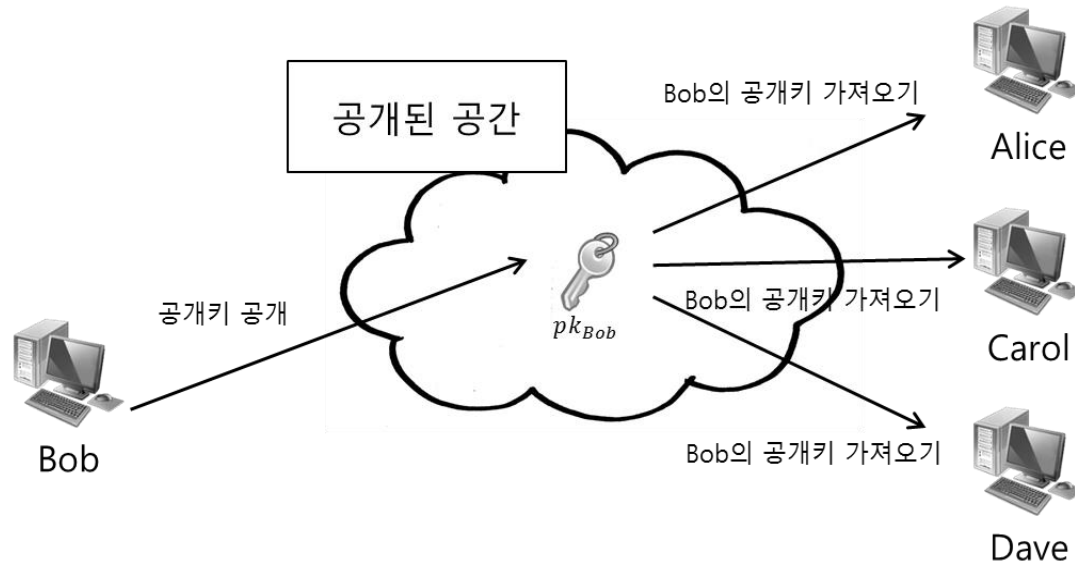
## ■ STS(Station-To-Station) 프로토콜





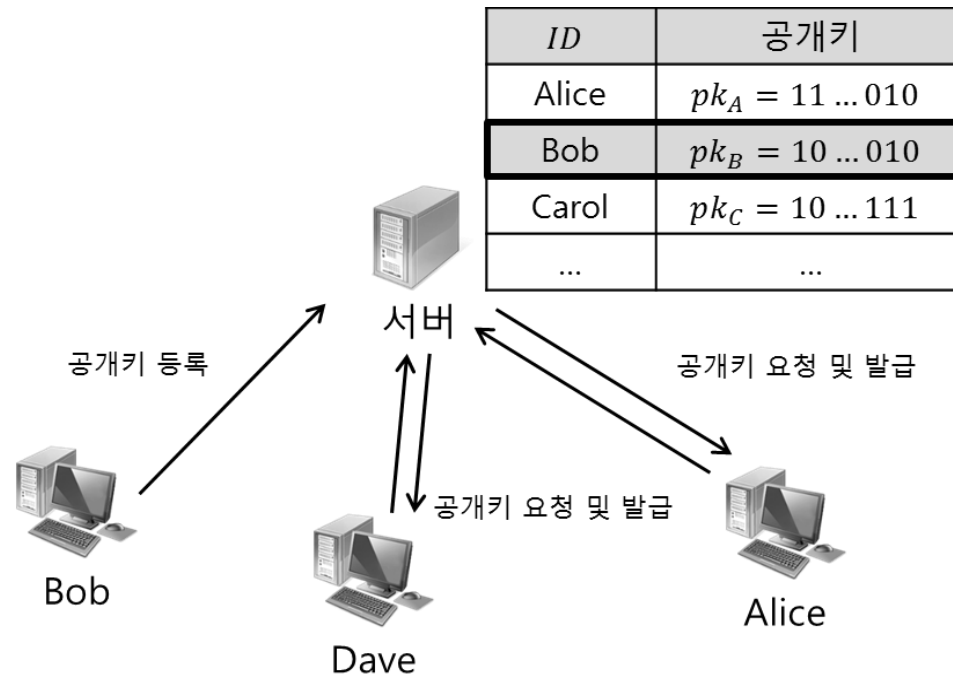
# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

- 공개키 암호시스템을 이용한 키 교환
  - ✗ 공개키 공개 선언 → 신뢰성?



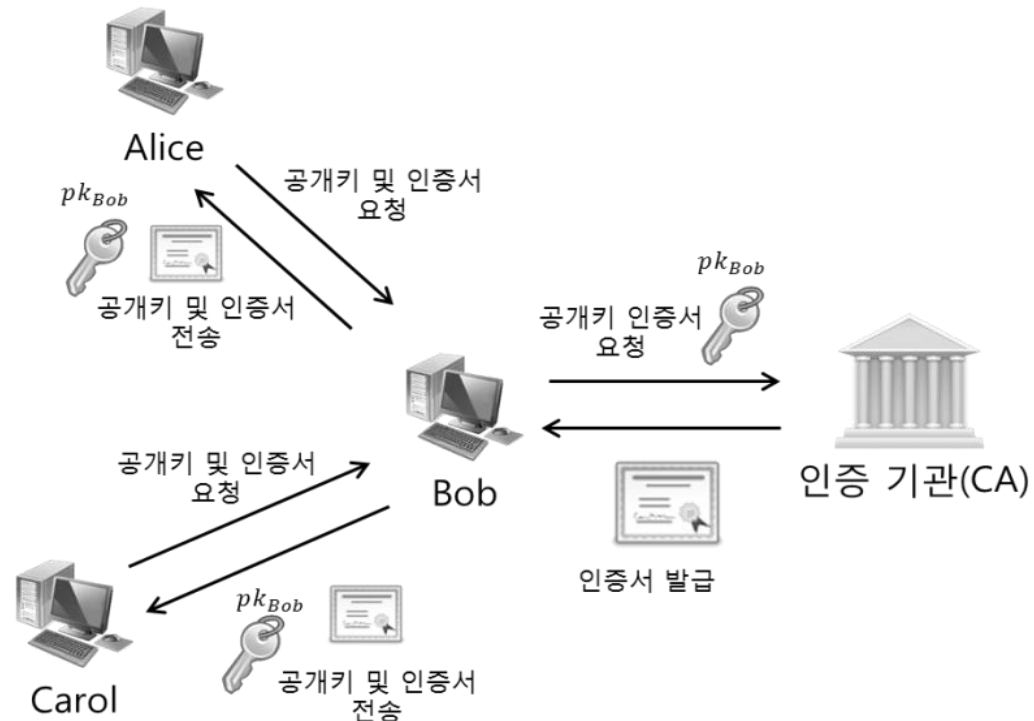
# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

- 공개키 암호시스템을 이용한 키 교환
  - ✗ 신뢰할 수 있는 서버 이용 → 서버에 과부하



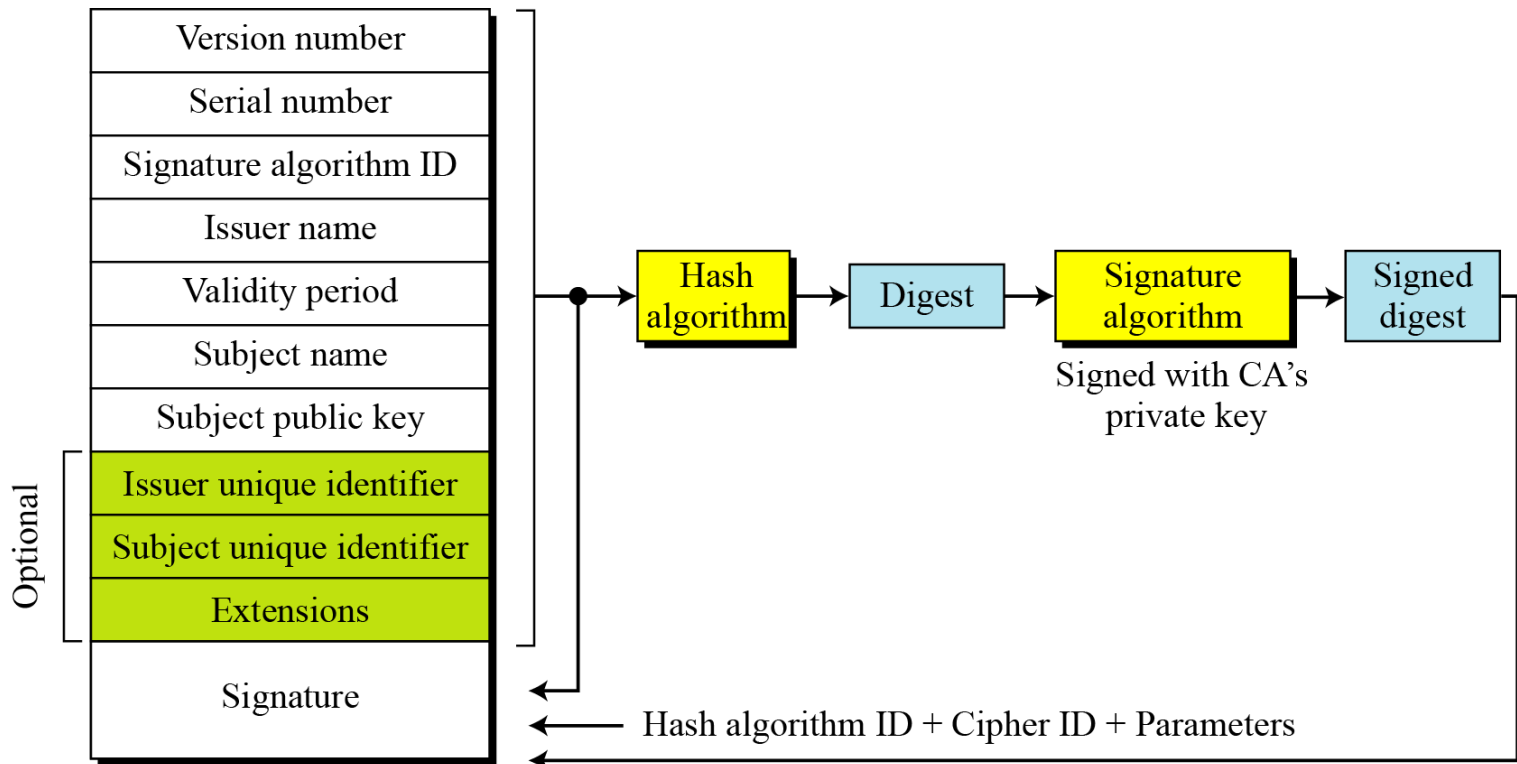
# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

- 공개키 암호시스템을 이용한 키 교환
  - ✗ 인증서를 이용한 공개키 인증



# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

- 공개키 암호시스템을 이용한 키 교환
  - ✗ 인증서 형태(X.509)



# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

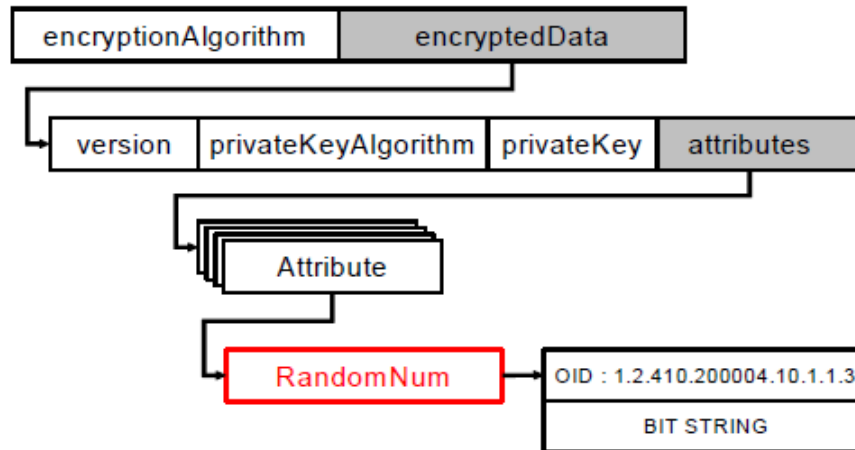
## ■ Private Key 저장방식 (PKCS#5v2.0, PKCS#8)

✗  $\text{Key} = \text{PBKDF}(\text{PW}, \dots)$

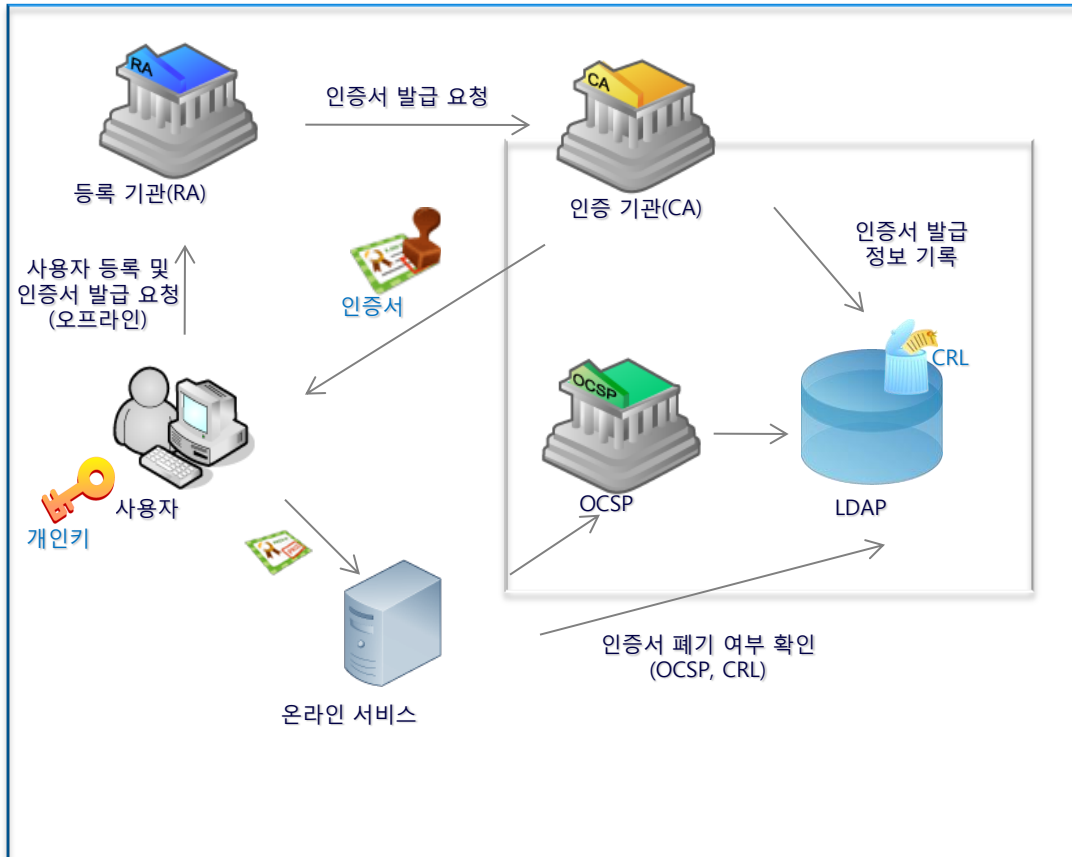
▶ PBKDF : PKCS#5 password based key derivation function

✗  $\text{SEED}_{\text{Key}}(\text{Private\_Key} \mid \text{R})$

PKCS #8 EncryptedPrivateKeyInfo



# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)



CA	Certificate Authority 사용자의 인증서를 발급하는 기관
RA	Registration Authority 사용자와 직접 대면 후 인증 기관에 사용자 정보를 등록해주는 기관
인증서	Certificate CA의 서명이 들어 있는 X.509 표준 규격의 인증서
개인키	Private Key 인증서 내의 공개키와 쌍이되는 개인키(PKCS #1, #8)
온라인 서비스	Online Service PKI 를 통한 사용자 인증을 필요로 하는 서비스 프로바이더(예 : 뱅킹)
OCSP	Online Certificate Status Protocol 실시간 인증서 상태(예 : 폐기여부) 검증 서비스
LDAP	Lightweight Dir. Access Protocol 인증서 저장소(LDAP DB)를 액세스 하는 프로토콜(CRL 서비스 수행)

사용되는 국제 표준 규격	설 명
RFC 2459/3280	X.509 인증서와 CRL(인증서 폐기 목록)에 대한 정의
RFC 2510/2511	CMP 프로토콜에 대한 명세(인증서 발급 과정에 사용 됨)
RFC 2560	실시간 인증서 상태 검증 프로토콜인 OCSP에 대한 명세
RFC 1430/2253	LDAP 프로토콜 명세(LDAP DB 는인증서 저장소로 사용 됨)

# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

## ■ 가입자 등록 및 인증서 발급

IETF RFC 2511 (1999), Internet X.509 Certificate Request Message Format



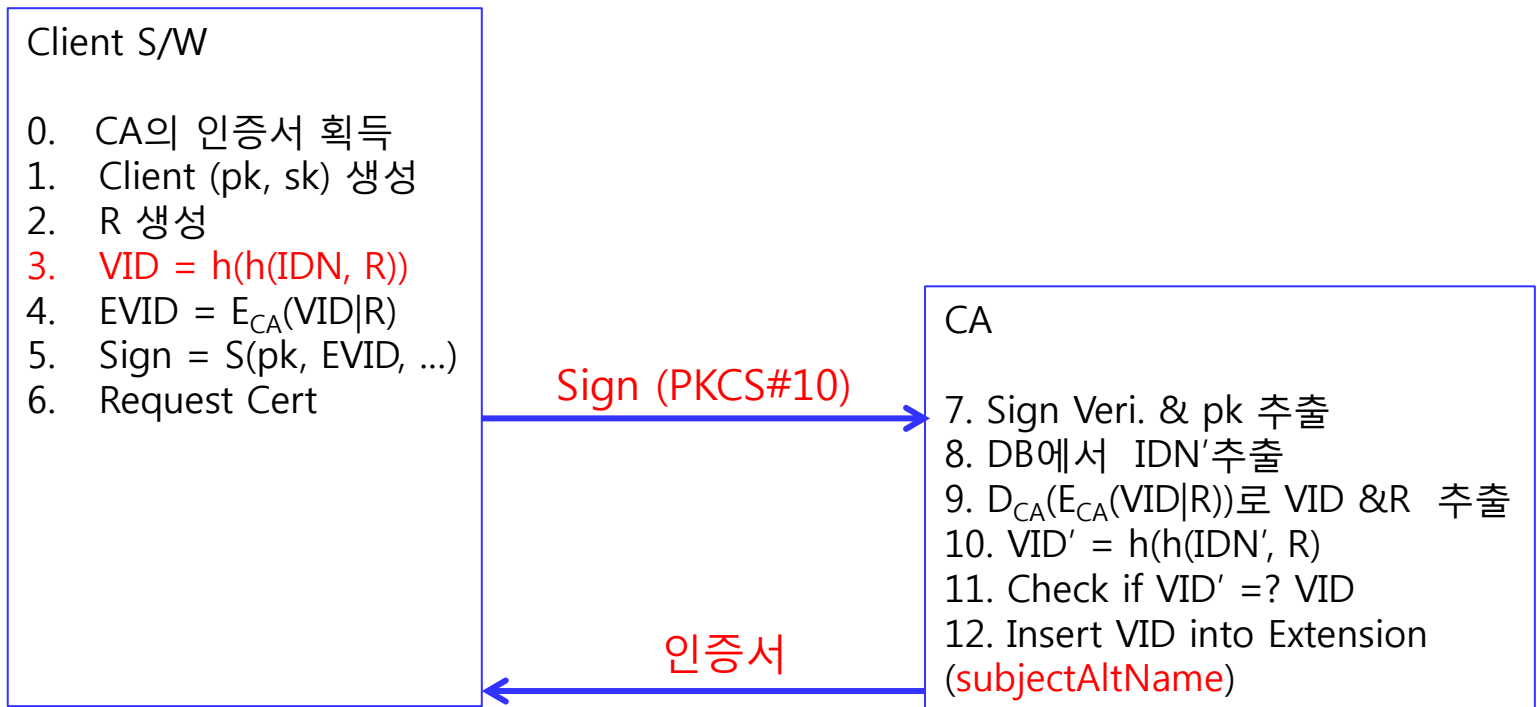
## 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

### ■ 가입자 등록 및 인증서 발급 : VID (Virtual ID)

✗  $VID = h(h(IDN, R))$

▶ IDN : 주민번호 or 사업자등록번호 (“-”는 삭제), R: 160 bit 난수

✗ 인증서 생성시 VID 정보 주입절차

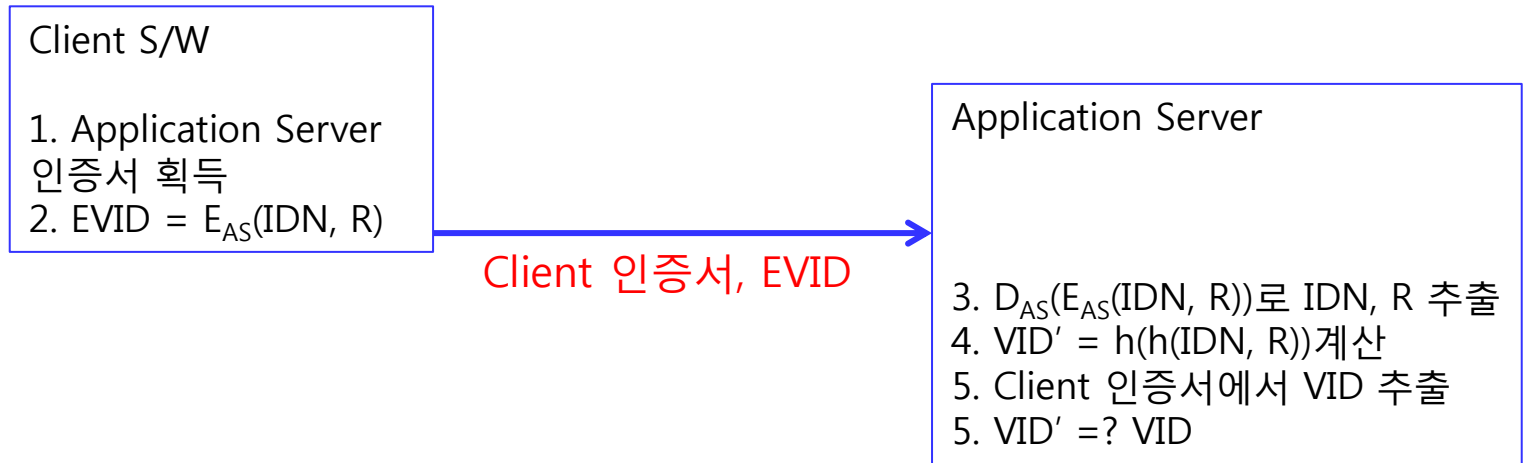




# 인증서를 이용한 신원확인

---

## ■ 인증서 로그인(예)



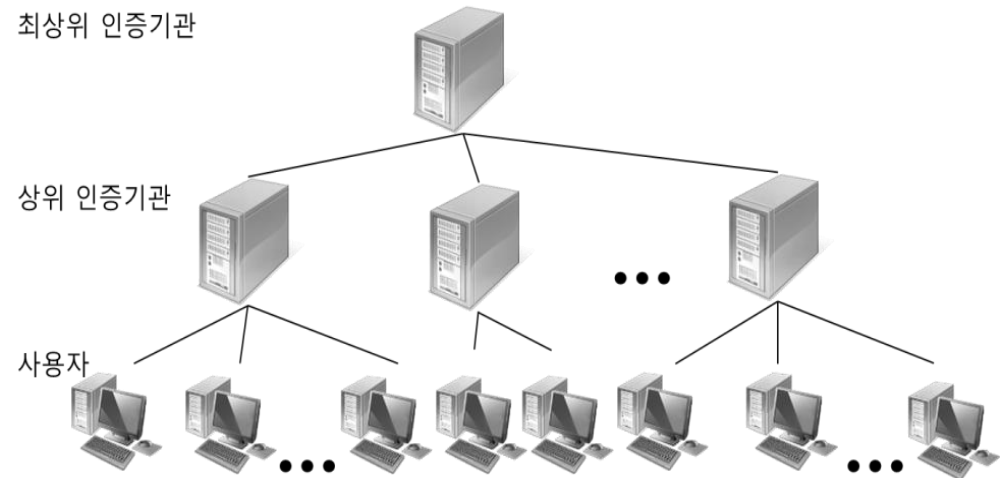
# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

## ■ 공개키 암호시스템을 이용한 키 교환

✕ 인증 기관들 간 신뢰 모델

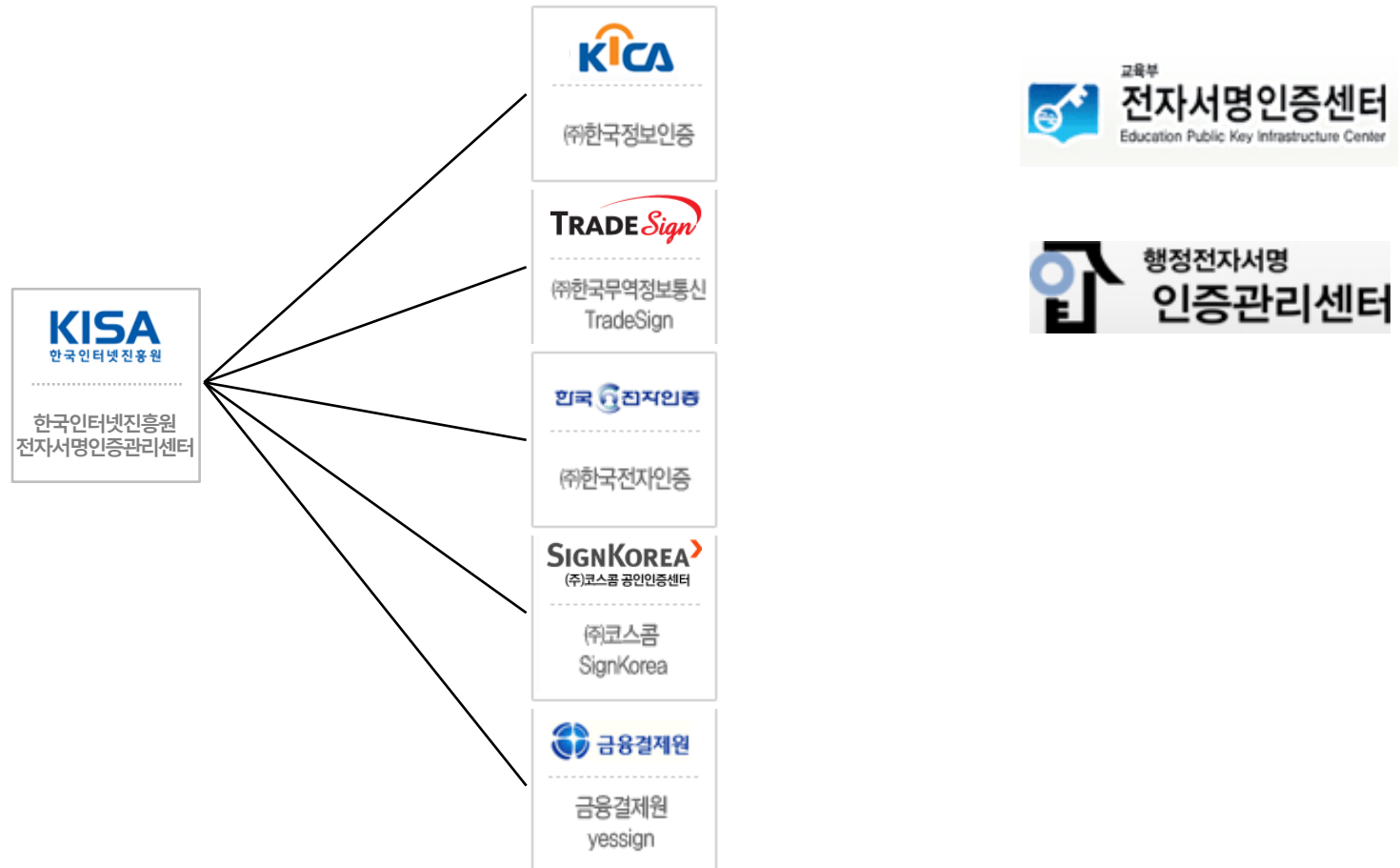
▶ 계층 모델(Hierarchical Model) - 국내모델

- 상위 계층이 바로 아래 계층의 인증서를 발급, 최상위 계층인 루트 인증 기관은 self-signing



# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

- 공개키 암호시스템을 이용한 키 교환
  - ✕ 인증 기관(Certificate Authority, CA)



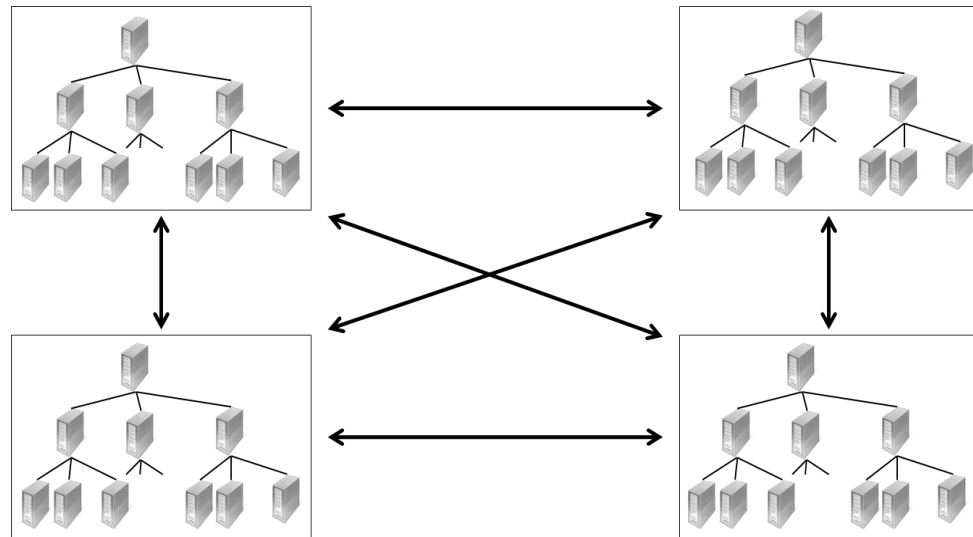
# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

- 공개키 암호시스템을 이용한 키 교환
  - ✕ KISA Root 인증서

최상위인증기관 인증서 - KISA RootCA 1(RSA)			
발급일자	2005-08-24	인증서버전	X.509 v3
만료일자	2025-08-24	인증서일련번호	04
인증서 다운로드		<input type="button" value="PEM 다운로드"/> <input type="button" value="DER 다운로드"/>	
인증서 내용	Root CA Certificate		
	<p>Data:</p> <p>Version: 3 (0x2)</p> <p>Serial Number: 4 (0x4)</p> <p>Signature Algorithm: sha1WithRSAEncryption</p> <p>Issuer: C=KR, O=KISA, OU=Korea Certification Authority Central, CN=KISA RootCA 1</p> <p>Validity</p> <p>Not Before: Aug 24 08:05:46 2005 GMT</p> <p>Not After : Aug 24 08:05:46 2025 GMT</p> <p>Subject: C=KR, O=KISA, OU=Korea Certification Authority Central, CN=KISA RootCA 1</p> <p>Subject Public Key Info:</p> <p>Public Key Algorithm: rsaEncryption</p> <p>RSA Public Key: (2048 bit)</p> <p>Modulus (2048 bit):</p> <div>00:bc:04:e4:fa:13:39:10:34:96:20:6b:6c:68:bb: fa:db:77:ff:27:f7:ac:ec:2f:e7:fd:10:7f:6d:6f: 8c:2a:cd:25:09:5b:24:14: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:77:62:10:47:0d: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:c8:61:9e:f5:c3:64:af:06:15:dc:23:8b: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:c4:93:47:3c:37:ea:02:67: 92:fe:fc:22:24:5c:ac:d2:2c:e0:5c:01:33:8a:c1: ...</div>		

# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

- 공개키 암호시스템을 이용한 키 교환
  - ✕ 인증 기관들 간 신뢰 모델
    - ▶ 메쉬 모델(Mesh Model)
      - 국가간 상호인증



# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

---

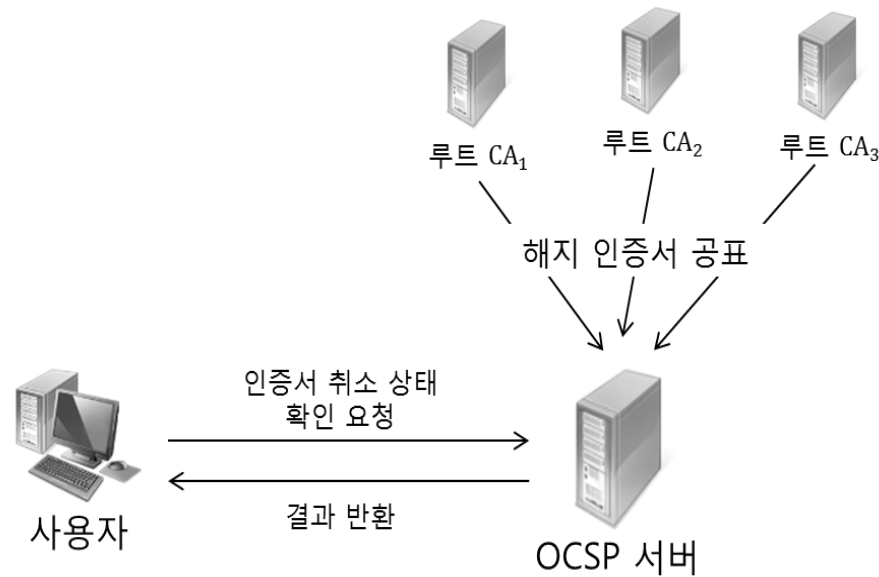
## ■ 인증서 취소 상태 확인

✕ 인증서 해지 목록 (Certificate Revoked List, CRL)

서명 알고리즘 ID
발행자 이름
금번 업데이트 시간
다음 업데이트 일자
첫 번째 폐지 인증서
...
마지막 폐지 인증서
서명

# 11.4 공개키 기반 구조(Public-Key Infrastructure, PKI)

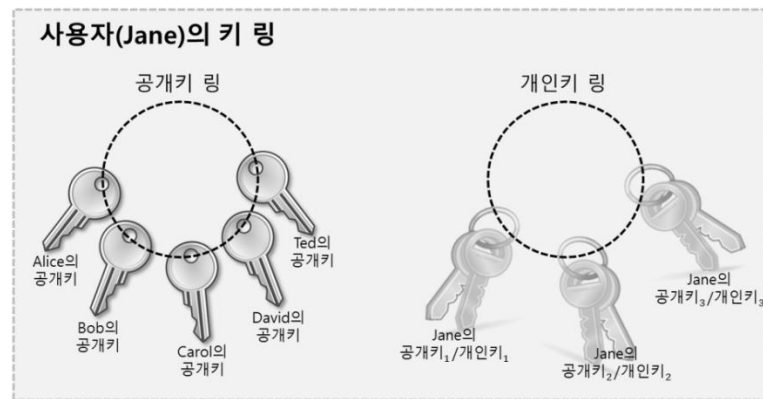
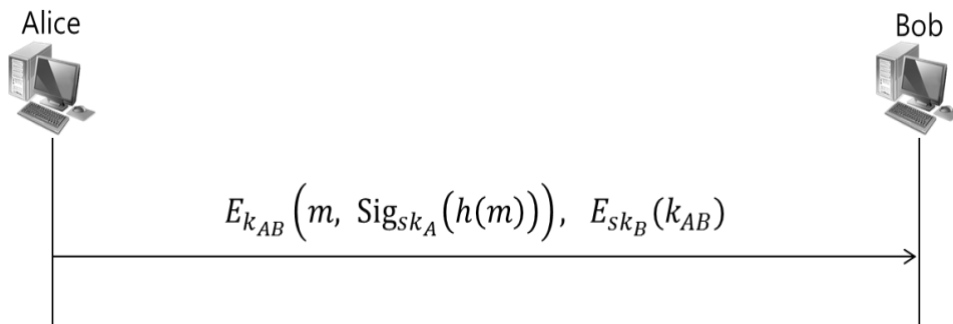
- 실시간 인증서 상태 확인 기술(Online Certificate Status Protocol, OCSP)



# 11.5 PGP(Pretty Good Privacy)

## ■ 전자우편 보안 프로토콜

- ✗ 필 짐머만(Phil Zimmermann)
- ✗ 전자우편의 기밀성, 무결성, 인증 등 제공

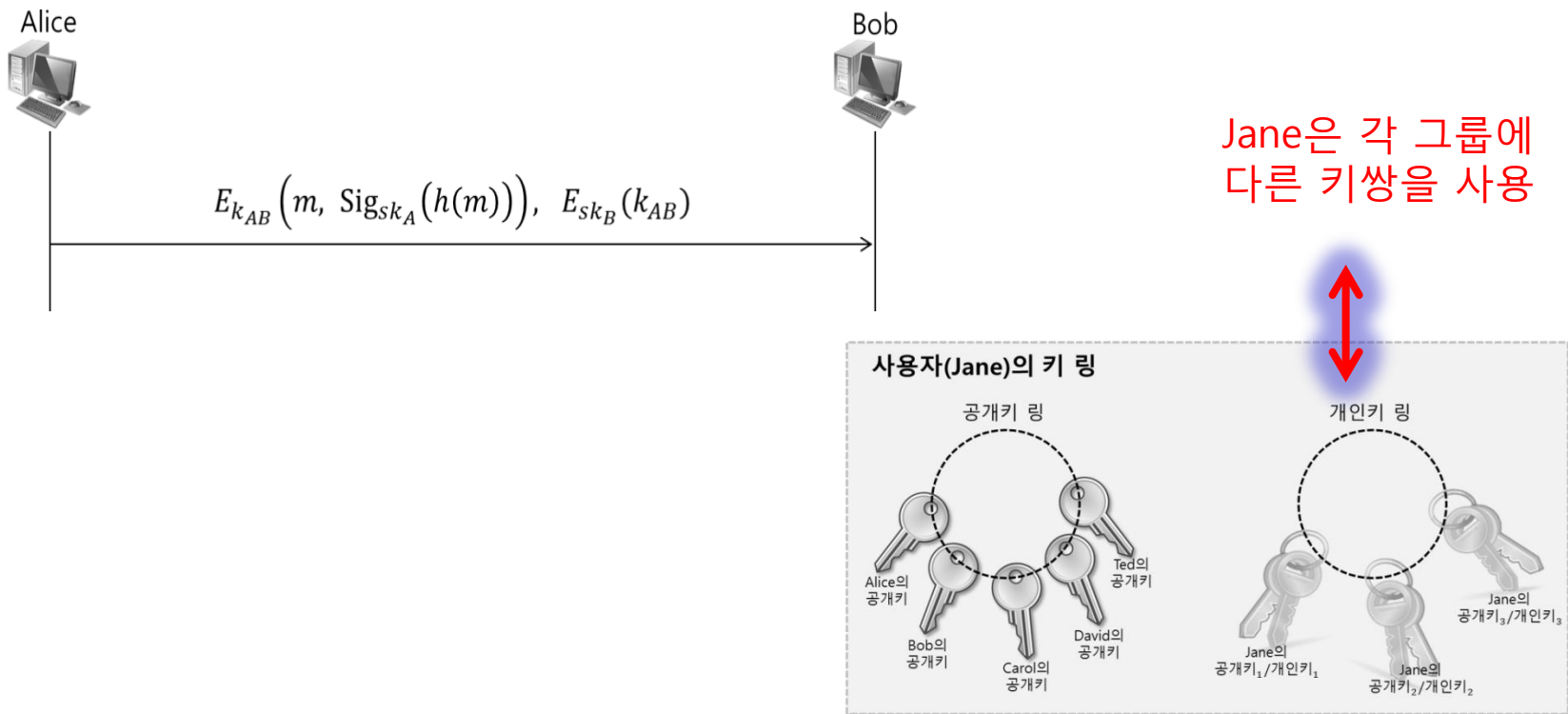




# 11.5 PGP(Pretty Good Privacy)

## ■ PGP 인증서

- ✗ 필 짐머만(Phil Zimmermann)
- ✗ 전자우편의 기밀성, 무결성, 인증 등 제공



# 11.5 PGP(Pretty Good Privacy)

---

## ■ 개인키 링 테이블(Private key ring table)

사용자 ID	키 ID	공개키	암호화된 개인키	타임스탬프
alice@korea.ac.kr	CD11...25	CD11...25...38	32A6....72	120528-15:32



The first 64 bit of the PK



The time of creation

# 11.5 PGP(Pretty Good Privacy)

- 공개 키 링 테이블(Public key ring table) 생성
  - ✗ PGP의 인증서는 공개키 링에 속해있는 사용자들이 서로에 대한 인증서를 발급
  - ✗ Jane의 공개키 테이블

사용자 ID	키 ID	공개키	생성자 신뢰등급	인증서	인증서 신뢰등급	키 적법성	타임 스탬프
Alice@...	CD11..	CD11..	F			F	...

Jane의 Alice에 대한  
신뢰등급

Alice의 소개자

인증서 신뢰등급으로 계산  
소개자의 신뢰등급과 동일

# 11.5 PGP(Pretty Good Privacy)

---

## ■ 공개 키 링 테이블(Public key ring table) 생성

사용자 ID	키 ID	공개키	생성자 신뢰등급	인증서	인증서 신뢰등급	키 적법성	타임 스탬프
Alice@...	CD11..	CD11..	F			F	...
Bob@...	45A2...	45A2...	P			P	...

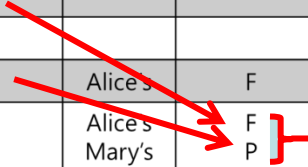
사용자 ID	키 ID	공개키	생성자 신뢰등급	인증서	인증서 신뢰등급	키 적법성	타임 스탬프
Alice@...	CD11..	CD11..	F			F	...
Bob@...	45A2...	45A2...	P			P	...
Mary@...	3B34...	3B34...	P	Alice's	F	F	...



# 11.5 PGP(Pretty Good Privacy)

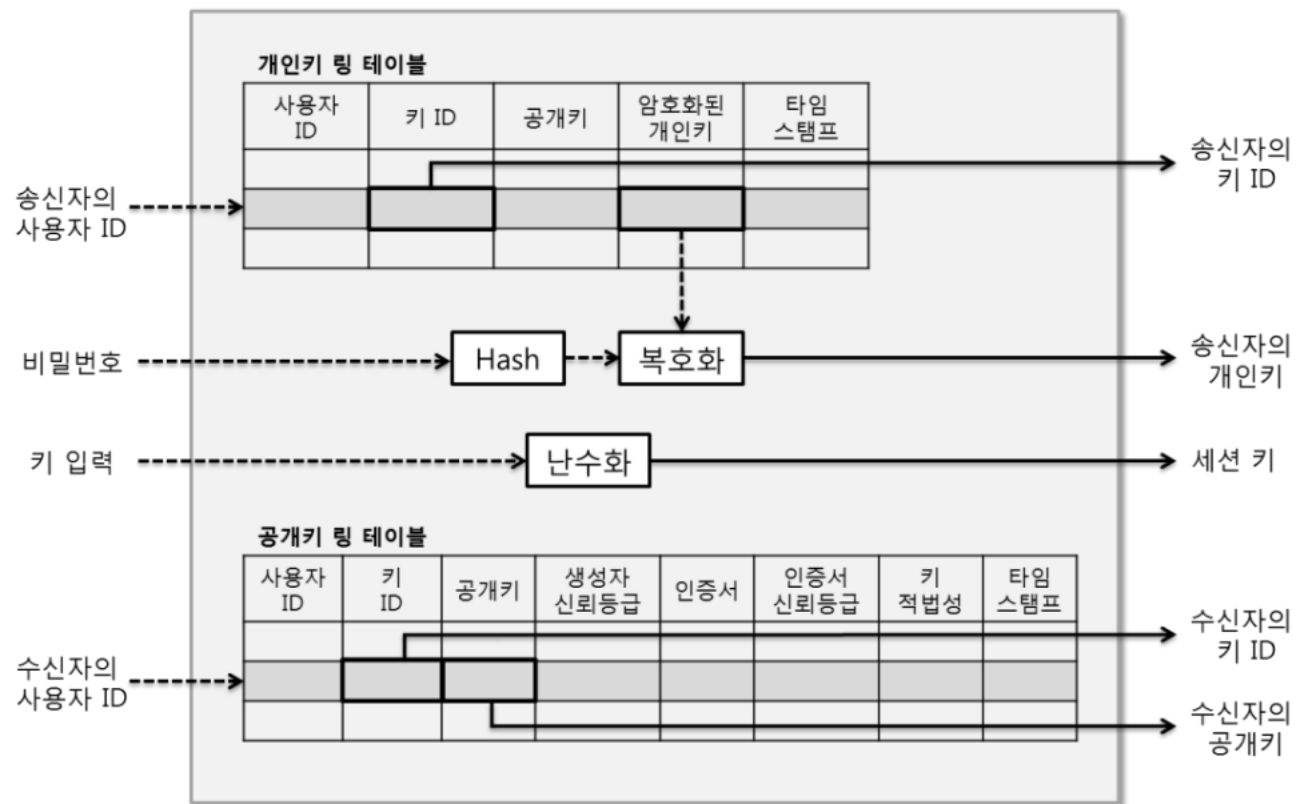
## ■ 공개 키 링 테이블(Public key ring table) 생성

사용자 ID	키 ID	공개키	생성자 신뢰등급	인증서	인증서 신뢰등급	키 적법성	타임 스탬프
Alice@...	CD11..	CD11..	F			F	...
Bob@...	45A2...	45A2...	P			P	...
Mary@...	3B34...	3B34...	P	Alice's	F	F	...
Kate@...	E5A3...	E5A3...	N	Alice's Mary's	F P	F	...



# 11.5 PGP(Pretty Good Privacy)

## ■ 송신자의 키 링 테이블 사용



# 11.5 PGP(Pretty Good Privacy)

## ■ 수신자의 키 링 테이블 사용

