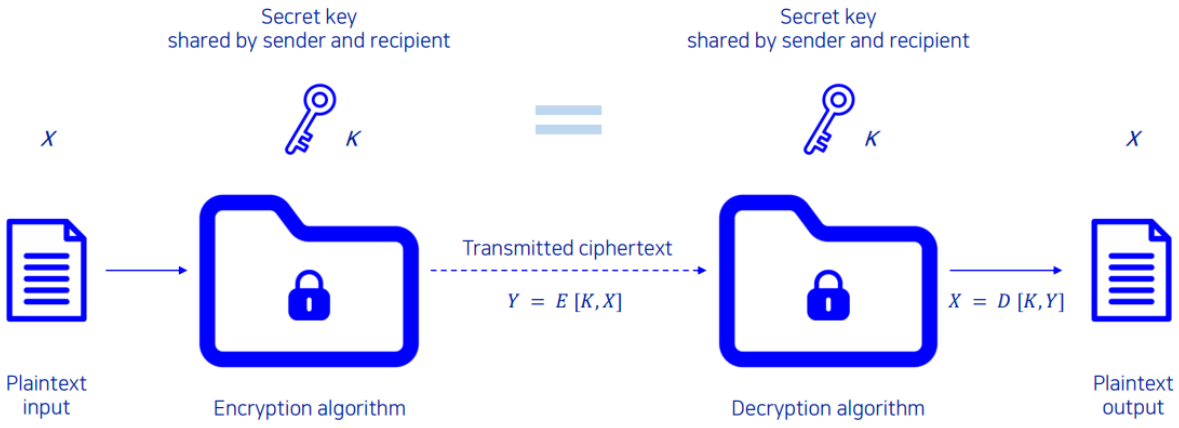
2주차

Cryptograpic Tools

Symmetric encryption(대칭 암호화)

- 데이터의 전송 또는 저장에 대한 기밀성(confidentiality)을 제공하기 위한 보편적인 기술
- 전통적인 암호화 또는 단일 키 암호화(single-key encryption)로도 불림
- 안전한 사용을 위한 두 가지 요구 사항:
  - 강력한 암호화 알고리즘이 필요함
  - 발신자와 수신자가 비밀 키의 사본을 안전한 방식으로 획득하고 키를 안전하게 보관해야 함

Symmetric encryption model(대칭 암호화 모델)



Attacking symmetric encryption(대칭 암호화 공격)

Cryptanalytic attacks(암호해독공격)

- 다음을 의존함
  - 알고리즘의 특성
  - 평문의 일반적인 특성에 대한 몇 가지 지식
  - 몇 가지 샘플 평문-암호문 쌍

- 특정 평문이나 사용되고 있는 키를 추측하려는 알고리즘의 특성을 이용함
    - 성공할 경우 해당 키로 암호화된 모든 과거 및 미래 메시지가 침해됨

Brute-force attacks(무차별 대입 공격)

- 평문으로 이해할 수 있는 암호문이 얻어질 때까지 특정한 암호문에 모든 가능한 키를 시도함
    - 평균적으로 성공을 달성하기 위해 가능한 모든 키의 절반을 시도해야 함

대칭 암호화 알고리즘 중 세 가지 비교

	DES (Data Encryption Standard):	Triple DES	AES (Advanced Encryption Standard)
Plaintext block size (bits)(평문)	64	64	128
Ciphertext block size (bits)(암호문)	64	64	128
Key size (bits)	56	112 or 168	128, 192 or 256

Average time required for exhaustive key search(완전한 키 탐색에 필요한 평균시간)

Key size (bits)	Cipher	# of alternative keys	Time required at 10^9 decryptions /s	Time required at 10^{13} decryptions /s
56	DES	$2^{56} \approx 7.2 \times 10^{16}$	2^{55} ns = 1.125 years	1 hour
128	AES	$2^{128} \approx 3.4 \times 10^{38}$	2^{127} ns = 5.3×10^{21} years	5.3×10^{17} years
168	Triple DES	$2^{168} \approx 3.7 \times 10^{50}$	2^{167} ns = 5.8×10^{33} years	5.8×10^{29} years
192	AES	$2^{192} \approx 6.3 \times 10^{57}$	2^{191} ns = 9.8×10^{40} years	9.8×10^{36} years
256	AES	$2^{256} \approx 1.2 \times 10^{77}$	2^{255} ns = 1.8×10^{60} years	1.8×10^{56} years

Data Encryption Standard (DES) -데이터 암호화 표준

- 최근까지 가장 널리 사용된 암호화 방식
  - FIPS PUB 46
  - 데이터 암호화 알고리즘 (DEA)로 불리기도 함
  - 64비트 평문 블록과 56비트 키를 사용하여 64비트 암호문 블록을 생성
- 강도에 대한 우려:
  - 알고리즘 자체에 대한 우려
    - DES는 현재 존재하는 가장 연구된 암호화 알고리즘.
  - 56비트 키 사용에 대한 우려
    - 상업용 off-the-shelf processors의 속도는 이러한 키 길이가 심각하게 불충분하다는 것을 보여줌.

Triple DES (3DES)

- 기본 DES 알고리즘을 두 번 또는 세 번 반복하여 사용하는 암호화 방식
- 처음으로 금융 애플리케이션에서 사용하기 위해 1985년 ANSI 표준 X9.17에서 표준화됨
- 장점:
  - 168비트 키 길이는 DES의 brute-force 공격에 대한 취약성을 극복함
  - 기본 암호화 알고리즘은 DES와 동일함
- 단점:
  - 소프트웨어에서 알고리즘이 느림
  - 64비트 블록 크기를 사용함

Advanced Encryption Standard (AES)

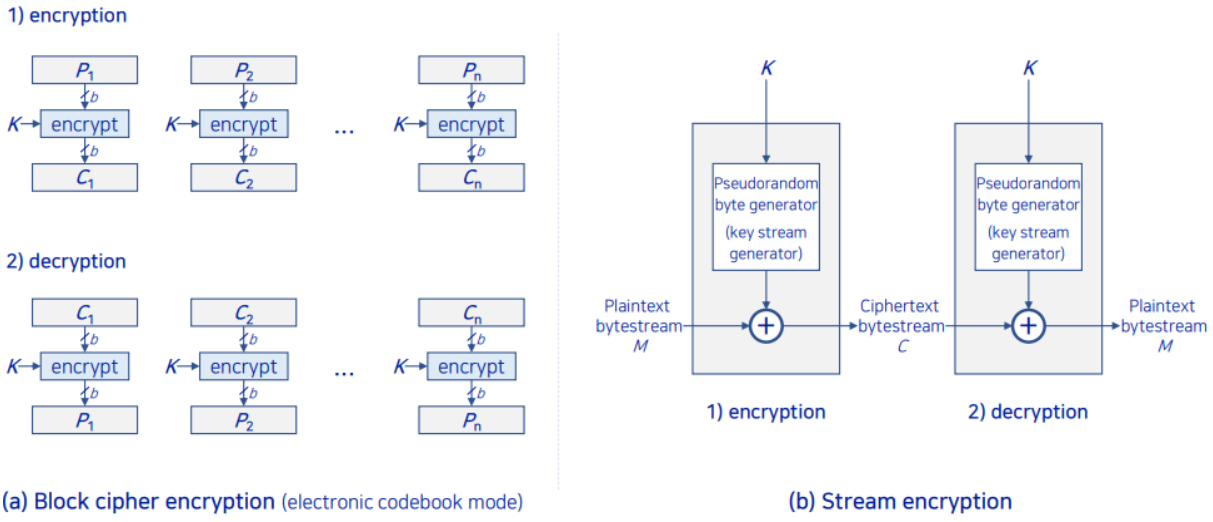
- 3DES의 대체품이 필요했음
  - 3DES는 장기간 사용하기에 합리적이지 않았음
- NIST는 1997년에 새로운 AES를 위한 제안을 요청
  - 보안 강도가 3DES와 동일하거나 더 우수해야 함
  - 효율성이 크게 향상
  - 대칭 블록 암호화

- 128비트 데이터 및 128/192/256 비트 키
- 2001년 11월 Rijndael이 선택됨

Practical security issues(실제 보안 문제)

- 일반적으로 대칭 암호화는 단일 64비트 또는 128비트 블록보다 큰 데이터 단위에 적용
- 전자 코드북 (ECB) 모드는 다중 블록 암호화에 대한 가장 간단한 접근 방식
  - 각 평문 블록은 동일한 키를 사용하여 암호화됨.
  - 암호분석가들은 평문의 규칙성을 이용할 수 있음.
- 운영 모드
  - 대칭 블록 암호화의 보안을 증가시키기 위해 개발된 대안적인 기술
  - ECB의 취약점을 극복

Types of symmetric encryption(대칭 암호화의 유형)



(a) Block cipher encryption (electronic codebook mode)

(b) Stream encryption

Block & stream ciphers(블록 및 스트림 암호화)

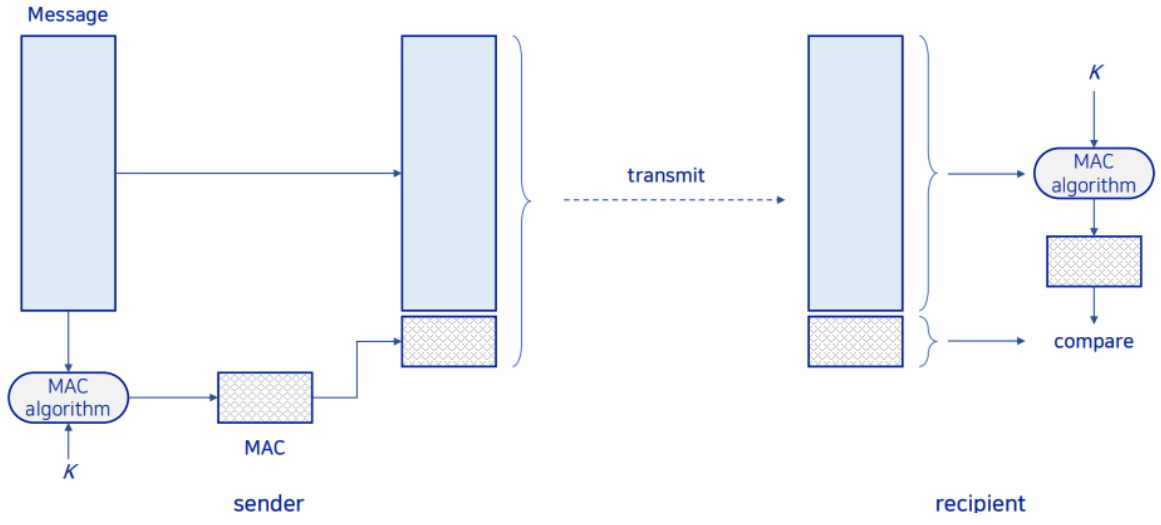
- Block cipher(블록 암호)
  - 입력을 한 번에 하나의 블록 단위로 처리
  - 각 입력 블록에 대해 출력 블록을 생성
  - 키를 재사용할 수 있음
  - 보다 일반적

- Stream cipher(스트림 암호)
  - 입력 요소를 계속해서 처리
  - 한 번에 하나의 요소를 출력
  - 주요 장점은 거의 항상 더 빠르고 코드 사용량이 훨씬 적음
  - 평문을 한 바이트씩 암호화
  - 의사난수 스트림은 입력 키의 지식 없이는 예측할 수 없는 스트림

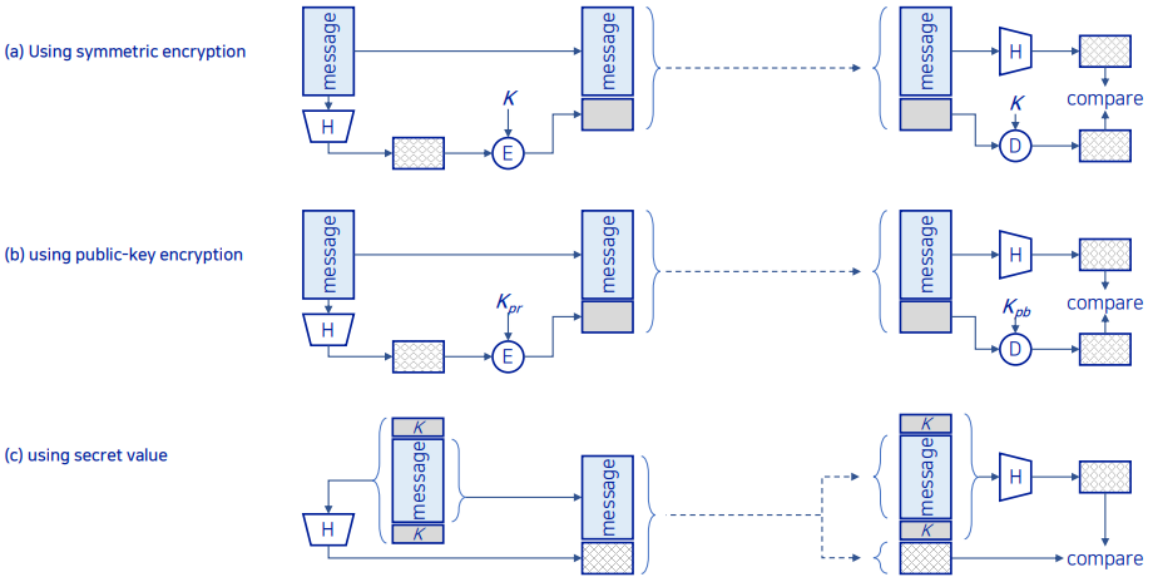
Message authentication(메세지 인증)

- 액티브 공격에 대한 보호
- 수신된 메시지가 실제인지 확인
  - 내용이 변경되지 않았음을 확인
  - 신뢰할 수 있는 출처로부터
  - 적시에 및 올바른 순서로
- 기존 암호화를 사용할 수 있음
  - 송신자와 수신자만 키를 공유함

Message authentication using a message authentication code

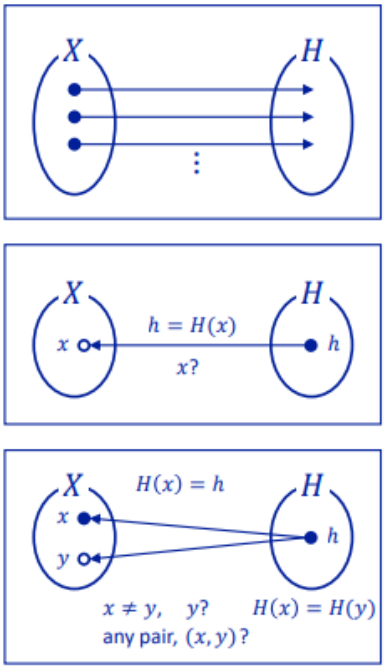


Message authentication using a one-way hash function



Hash function requirements

- 임의의 크기의 데이터 블록에 적용.
- 고정 길이의 출력을 생성.
- $H(x)$는 임의의 x에 대해 상대적으로 쉽게 계산됨.
- 일방향 또는 전 이미지 내성
  - $H(x) = h$인 x를 찾는 것이 계산적으로 불가능함
- $H(y) = H(x)$인 $y \neq x$를 찾는 것이 계산적으로 불가능합니다.
- 충돌 내성 또는 강한 충돌 내성
  - $H(x) = H(y)$인 어떤 쌍 (x, y)를 찾는 것이 계산적으로 불가능합니다.



Security of hash functions

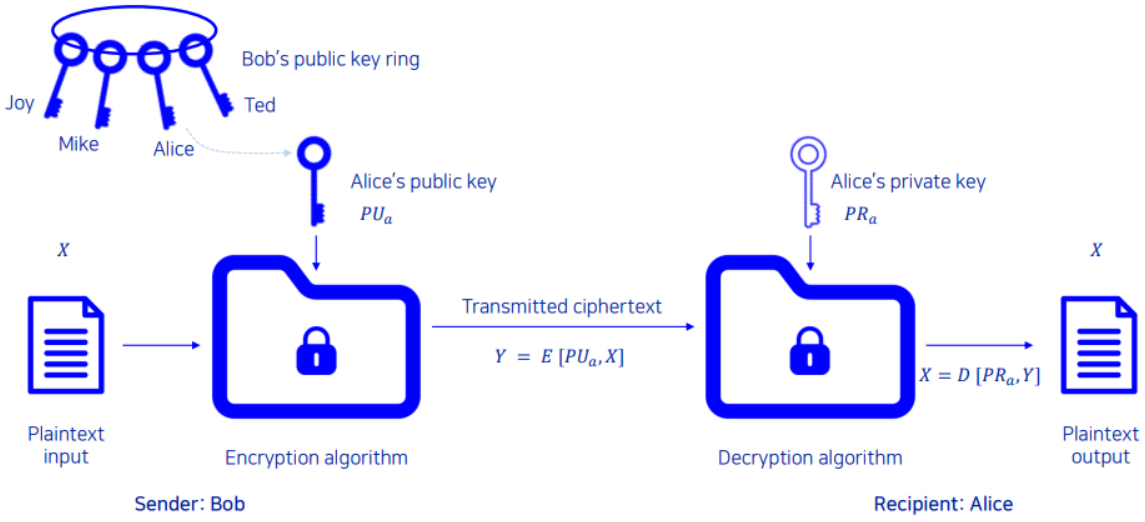
- 해시 함수 공격
  - 알고리즘의 논리적 취약점을 악용하는 암호 분석
  - 무차별 대입 공격: 해시 함수의 강도는 알고리즘이 생성하는 해시 코드의 길이에만 의존
- 안전한 해시 알고리즘 (SHA)
  - 가장 널리 사용되는 해시 알고리즘
  - SHA-1: 160비트의 해시 값 생성 (1993년 FIPS 180에서)
  - SHA-2 (SHA-256, SHA-384, SHA-512): 256, 384 및 512비트의 해시 값 생성 (2002년 FIPS 180-2에서)
  - SHA-3 (2015년)

Public-key encryption structure(공개 키 암호화 구조)

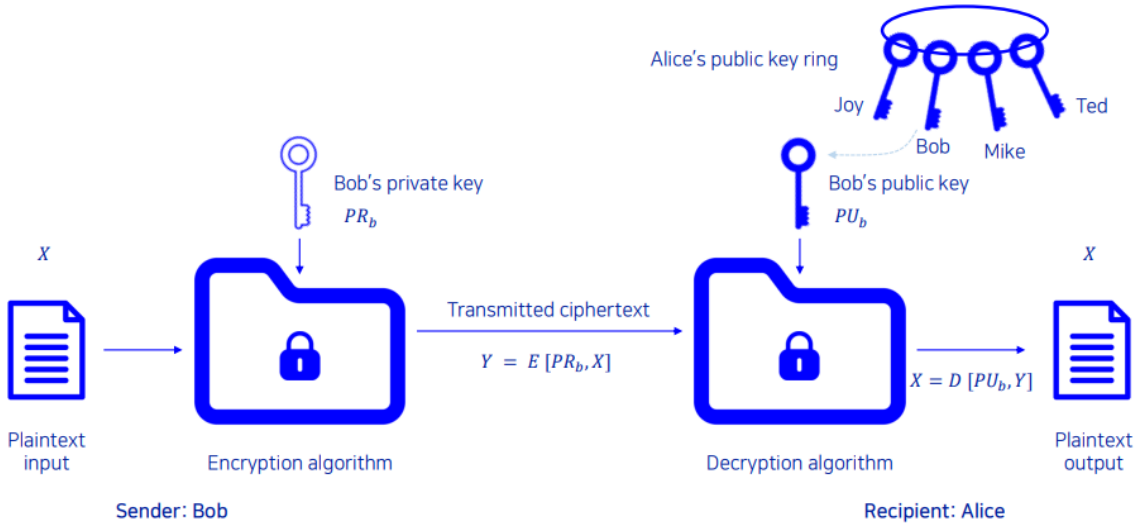
- 1976년 Diffie와 Hellman에 의해 공개적으로 제안됨
- 수학적 함수를 기반으로 함
- 비대칭적임
  - 두 개의 별도 키를 사용함
    - 공개 키 및 개인 키

    - 공개 키는 다른 사람들이 사용할 수 있도록 공개됨
- 배포를 위해 어떤 형태의 프로토콜이 필요함

Encryption with public key(공개 키로 암호화)



Encryption with private key(개인 키로 암호화)



Requirements for public-key cryptosystems(공개 키 암호체계 요구 사항)

- 키 쌍을 생성하는 것이 계산적으로 쉬워야 함
- 수신자가 공개 키를 알고 있는 경우 메시지를 암호화하는 것이 계산적으로 쉬워야 함
    - ciphertext(암호문) $C = E(PUb, M)$

- 수신자가 개인 키를 알고 있는 경우 암호문을 복호화하는 것이 계산적으로 쉬워야 함
  - original message(원본 메세지) $M = D(PRb, C) = D[PRb, E(PUb, M)]$
- 상대방이 공개 키로부터 개인 키를 결정하는 것이 계산적으로 불가능해야 함
- 상대방이 그렇지 않은 경우에도 원본 메시지를 복구하는 것이 계산적으로 불가능해야 함
- 각 역할에 대해 어느 키든 사용될 수 있는 경우 유용함
  - $M = D[PUb, E\ PRb, M\ ] = D[PRb, E(PUb, M)]$

Asymmetric encryption algorithms(비대칭 암호화 알고리즘)

RSA (Rivest, Shamir, Adleman)

- 1977년 개발됨
- 공개 키 암호화에 대한 가장 널리 수용되고 구현된 방법
- 평문과 암호문이 0부터 n-1까지의 정수인 블록 암호

Diffie-Hellman key exchange algorithm

- 두 사용자가 서로에게 안전하게 공유 비밀을 합의할 수 있도록 하며 이후 메시지의 대칭 암호화를 위한 비밀 키로 사용될 수 있음
- 키 교환에만 제한됨

Digital Signature Standard (DSS)

- SHA-1과 함께 디지털 서명 기능만 제공
- 암호화 또는 키 교환에 사용할 수 없음

Elliptic curve cryptography (ECC)-타원 곡선 암호학

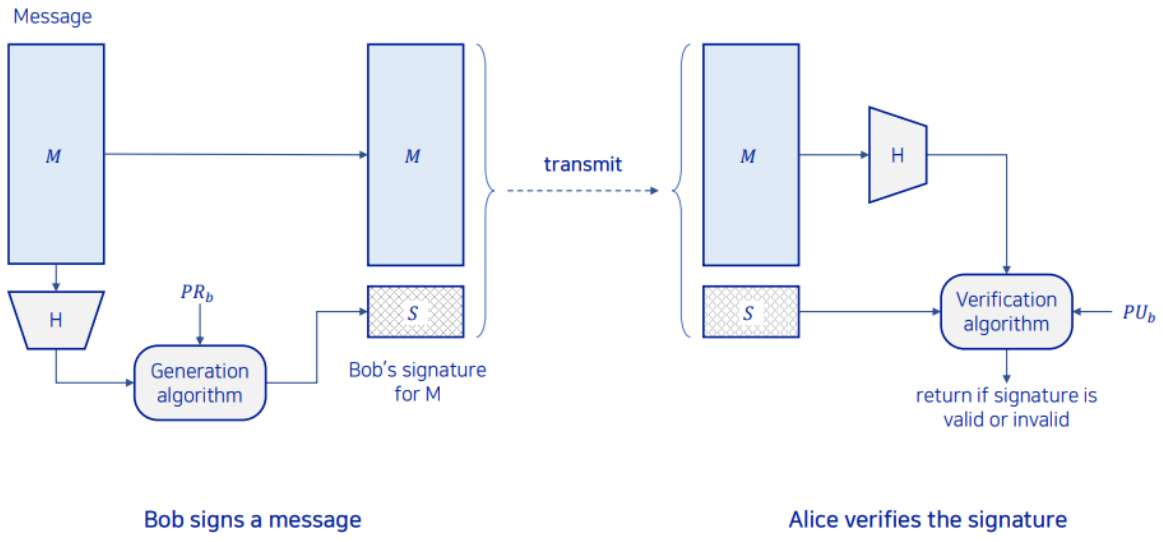
- RSA와 같은 보안 수준을 제공하지만 훨씬 작은 키를 사용함

Digital signatures

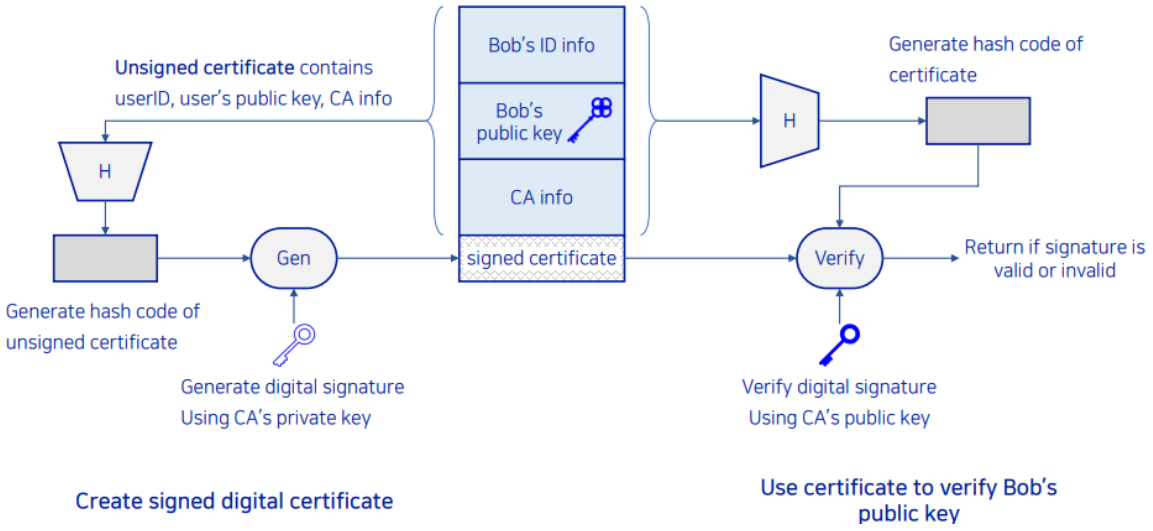
- NIST FIPS PUB 186-4에서 디지털 서명을 다음과 같이 정의함
  - "알맞게 구현될 때 데이터의 암호 변형 결과는 원천 인증, 데이터 무결성 및 발신자 부인 방지를 확인하기 위한 메커니즘을 제공합니다."

- 따라서 디지털 서명은 에이전트가 파일, 메시지 또는 다른 형태의 데이터 블록의 함수로 생성하는 데이터에 따라 달라지는 비트 패턴입니다.
- FIPS 186-4는 세 가지 디지털 서명 알고리즘 중 하나의 사용을 지정함:
  - Digital Signature Algorithm (DSA)
  - RSA Digital Signature Algorithm
  - Elliptic Curve Digital Signature Algorithm (ECDSA)\

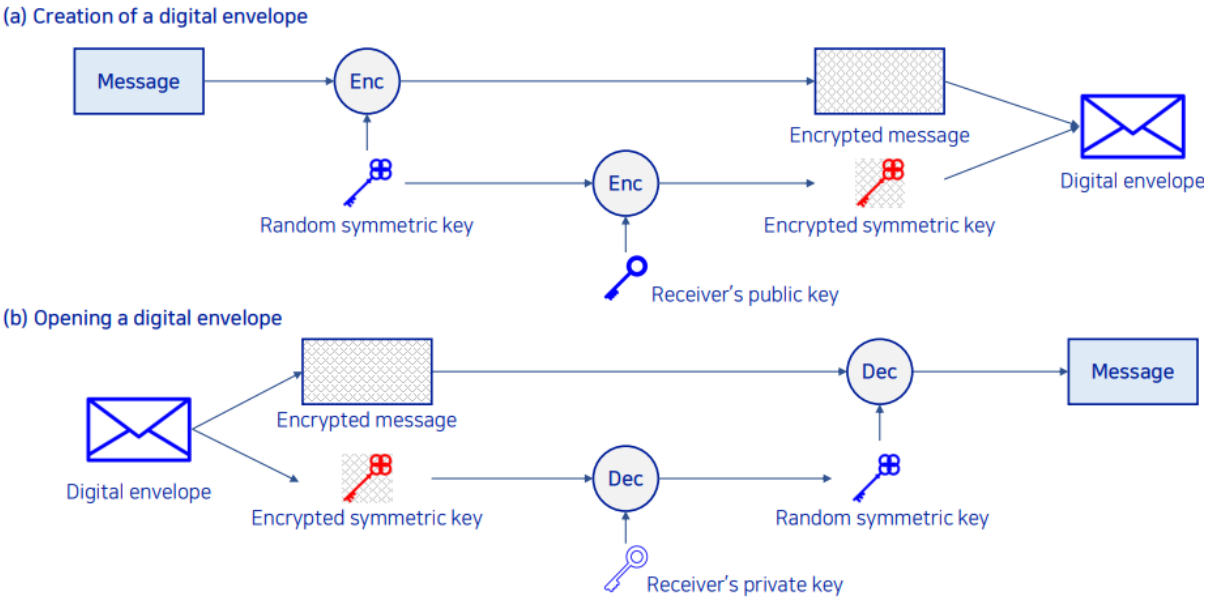
Digital signature process



public-key certificate use



Digital envelops



Random numbers

- 활용사례
  - 공개 키 알고리즘의 키 생성
  - 대칭 스트림 암호의 스트림 키 생성
  - 임시 세션 키 또는 디지털 봉투 생성에 대한 대칭 키 사용
  - 재생 공격을 방지하기 위한 핸드셰이킹
  - 세션 키

Random number requirements

Randomness

- 균일한 분포
  - 각 숫자의 발생 빈도는 대략적으로 동일해야 함.
- 독립성
  - 시퀀스 내의 한 값도 다른 값에서 추론될 수 없어야 함.

Unpredictability

- 각 숫자는 시퀀스 내 다른 숫자와 통계적으로 독립적이어야 함.
- 상대방은 이전 요소를 기반으로 시퀀스의 미래 요소를 예측할 수 없어야 함

Random vs pseudorandom(유사난수)

- 암호학적 응용 프로그램은 일반적으로 난수 생성을 위해 알고리즘 기반 기술을 사용
- 알고리즘은 결정론적이므로 통계적으로 난수가 아닌 숫자 시퀀스를 생성
- 유사난수
  - 통계적 무작위성 테스트를 충족하는 시퀀스를 생성
  - 예측 가능할 가능성이 있음
- 진정한 난수 생성기 (TRNG)
  - 결정론적이지 않은 소스를 사용하여 무작위성을 생성
  - 대부분은 예측할 수 없는 자연 과정을 측정하여 작동
    - ex) 방사선, 가스방전, 누수 전해질
  - 최신 프로세서에서 점점 더 많이 제공됨