#4 네트워크

암호

어떻게 정보를 보호할 것인가? 비밀성, 무결성, 가용성... > 암호 기술 도입!

통신의 내용이 밖으로 새지 않도록, 제 3자는 모르고 같은 편끼리만 알게 한 신호나 부호

- 암호 기술
  - 비밀통신, 인증, 접근제어
- 암호 시스템
  - 키생성, 암호화, 복호화

> 평문(M : Message) : 암호화 대상이 되는 문자열, 암호문을 복호화한 본래의 문자열
>
> 암호문(C : Cyper Text) : 제 3자의 불법적 획득을 방지하기 위해 평문을 암호알고리즘(E : Encyption)을 이용하여 암호화함으로서 생성되는 해독할 수 없는 문장
>
> 암호화 : 비밀성을 보장하기 위해 암호알고리즘에 의하여 평문을 암호문으로 바꾸는 과정
>
> 복호화 : 암호화된 문장을 평문으로 바꾸는 과정, 복호화알고리즘(D : Decryption)에 의하여
>
> 암호알고리즘 = 암호화 + 복호화 + 키생성

평문 > (암호화 알고리즘 + KEY : 암호기술) > 암호문

암호문 > (복호화알고리즘 + KEY : 암호기술) > 평문

- $E_{ke}(M) = C$
- $D_{kd}(C) = M$
- $D_{kd}(E_{ke}(M)) = M$

KEY

- 대칭키 시스템

암/복호화 키가 같음. 비밀키 또는 대칭키 암호라고 불린다.

- 문제
  - 키 전송(다른 수신자와는 다른 키가 필요하게됨. 따라서 필요한 키가 엄청 많아짐 생성/배송),
  - 키 관리 문제(키를 안전하게 보관해야함) 대두 > 키 전송을 위한 안전한 채널이 필요.
- 암복호화 속도가 빠름
- 128bit 이상의 키 필요
- DES, 3DES, SEED, AES 등

- 비대칭키 시스템

암/복호화 키가 다르며 공개키, 개인키 또는 비대칭키 암호라 불린다.

- 공개키/개인키(Public key/Private key)가 한 쌍
- 공개키는 공개, 개인키는 비밀 보관
- 누구나 암호화 가능. 복호화는 해당 사용자만 가능
- 속도가 느림
- RSA, DH, ECC

- 하이브리드 암호

대칭키 암호 + 공개키 암호

대칭키의 빠른 속도와 공개키 암호로 키 분배 문제의 장점만을 취득해서 쓰겠다!

- 암호화
  - 대칭키로 암호화된 평문
  - 공개키로 암호화된 세션키(대칭키)를 보냄
- 복호화
  - 세션 키를 수신자의 개인키를 이용해서 복호화
  - 세션키를 이용해 대칭 암호로 암호화된 암호문을 복호화

해시함수(Hash Function)

임의의 길이를 갖는 메세지 입력/ 고정된 길이의 메세지를 출력하는 함수

단방향/일방향 해시함수는 다시 복호화가 되지 않음 > 역산 불가능

- 속도가 매우 빠름
- 일방향 해시함수는 무결성을 입중한다
  - 타인으로 부터 위조, 변조가 되지 않음을 보장
  - 원본 파일의 해시값을 저장하고 후에 확인하고 싶은 파일의 해시 값을 비교하여 무결성을 확인할 수 있다

> 일방향성(약)
>
> - 해시값 H로부터 h(M) = H가 되는 메세지 M을 찾는 것은 불가능(역산이 되지 않는다)
>
> 일방향성(강)
>
> - 메세지 M과 h(M) = H가 주어졌을 때 h(M')=H를 찾는 것은 불가능(메세지 값이 다르면 해시 값이 다르다)
>
> 충돌회피성
>
> - h(M) = h(M')가 되는 메세지 쌍 M, M'(M!=M')를 찾는 것은 불가능

무결성 확인 방법

A => 메세지, 자신의 해시값(M) => B

B 메세지의 해시값(M')과 받은 해시값(M) 비교

해시함수는 조작과 변경에 대해서는 검증이 가능하다. 하지만 거짓 행세에 대해서는 검증이 불가능해 메세지에 대한 인증(>메세지 인증코드, 전자 서명)이 필요하다!

메세지 인증코드 MAC : Message Authentication Code

1. 무결성을 확인하고 메세지의 인증을 하는 코드
2. 메세지, 비밀키를 입력받아 고정비트 길이의 코드 출력 일방향 해시함수와 다르게 키를 입력받는다.
3. 송/수신 측은 사전에 비밀키 K를 공유하고 있음을 가정

무결성 확인 방법

M, 비밀키로 암호화한 MAC을 보냄

MAC을 비밀키로 푼 M과 받은 M을 비교

- 공격방법
    - 정당한 MAC을 여러번 재 전송하는 공격
        - EX) 계좌이체 시 입금에 대한 MAC
        - 보존해 둔 메세지와 MAC값을 반복 송신
- 공격 방지
    - 순서 번호 : 메세지를 보내면서 시퀀스에 대한 순서변호를 만들어 보내고 받음
    - 타임 스탬프 : 유효 시간을 정해 MAC값이 만들어진 시간 내에 송신을 요청받은 것만 처리
    - 비표 : 송신자가 임의로 찍어낸 번호로 검증

MAC은 제 3자가 인증을 할 시에 한계가 존재. 누가 만든 MAC인지 확인할 수 없음 > 전자서명 필요!