Network Security Essentials

- 3장 공개키 암호와 메시지 인증(1) -

전 상 기(sanggi@pel.smuc.ac.kr)

상명대학교 프로토콜공학연구실

목차

- 메시지 인증 방법
- 안전 해시함수
- 메시지 인증 코드

메시지 인증 방법

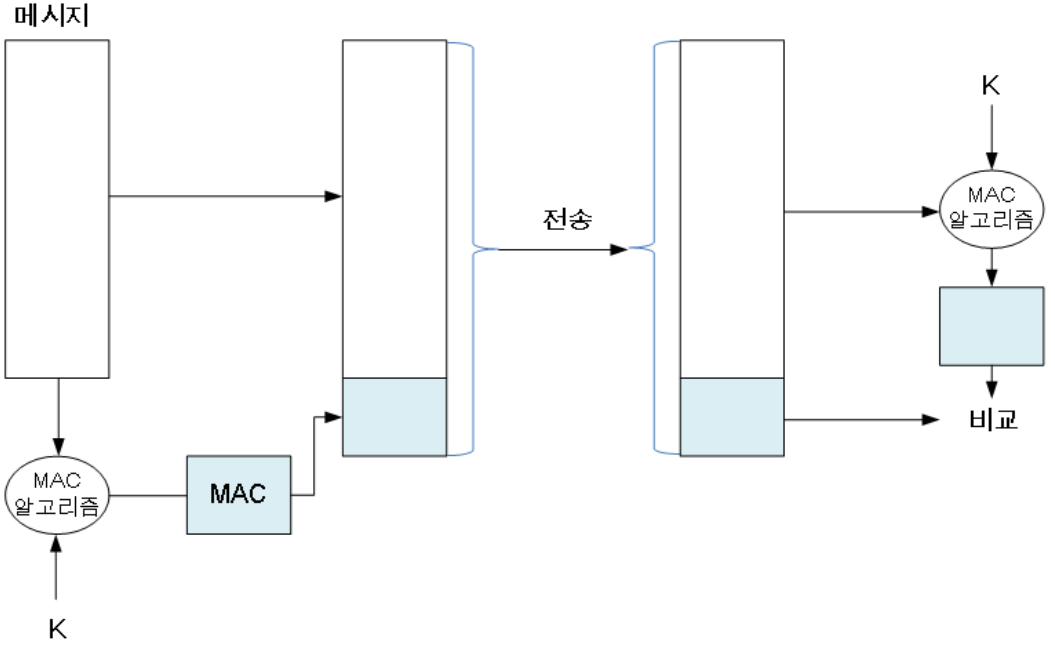
- 메시지 인증(Message authentication)
  - 정의
    - 통신 양측으로 하여금 받은 메시지가 진짜임을 확인하도록 해주는 절차
  - 인증 방법
    - 메시지를 암호화한 인증
      - 키로 인해 진짜 송신자만이 수신자에게 보내는 메시지를 성공적으로 암호화 할 수 있음
      - 타임스탬프(Timestamp)를 통해 고의적인 시간 지연을 확인
    - 메시지 암호화 없이 메시지 인증
      - 인증 꼬리표(Authentication tag)를 생성하고 메시지에 붙여서 전송
      - 메시지를 암호화하지 않아 메시지에 대한 기밀성은 보장되지 않음

메시지 인증 방법

- 메시지 인증(Message authentication)
  - 인증 방법
    - 메시지 암호화 없이 메시지 인증
      - 사용 적합한 세 가지 경우
        - 브로드캐스트하는 경우
        - 부하가 과도하게 걸려 메시지 복호화할 시간이 없을 때
        - 컴퓨터 프로그램을 평문인 채로 인증할 때
      - 메시지 인증 코드(MAC, Message Authentication Code)
        - 메시지에 붙여지는 데이터 블록을 생성하기 위해 비밀키를 이용하는 방법
        - 특징
          - 송수신자는 공통 비밀키를 가지고 있음
          - 메시지가 변경되지 않음을 확신할 수 있음
          - 송신자로부터 송신되었음을 확신할 수 있음
          - 정확한 순서를 확신할 수 있음

메시지 인증 방법

- 메시지 인증(Message authentication)
  - 인증 방법
    - 메시지 암호화 없이 메시지 인증
      - 메시지 인증 코드(MAC, Message Authentication Code)
        - 메시지 인증 코드 과정



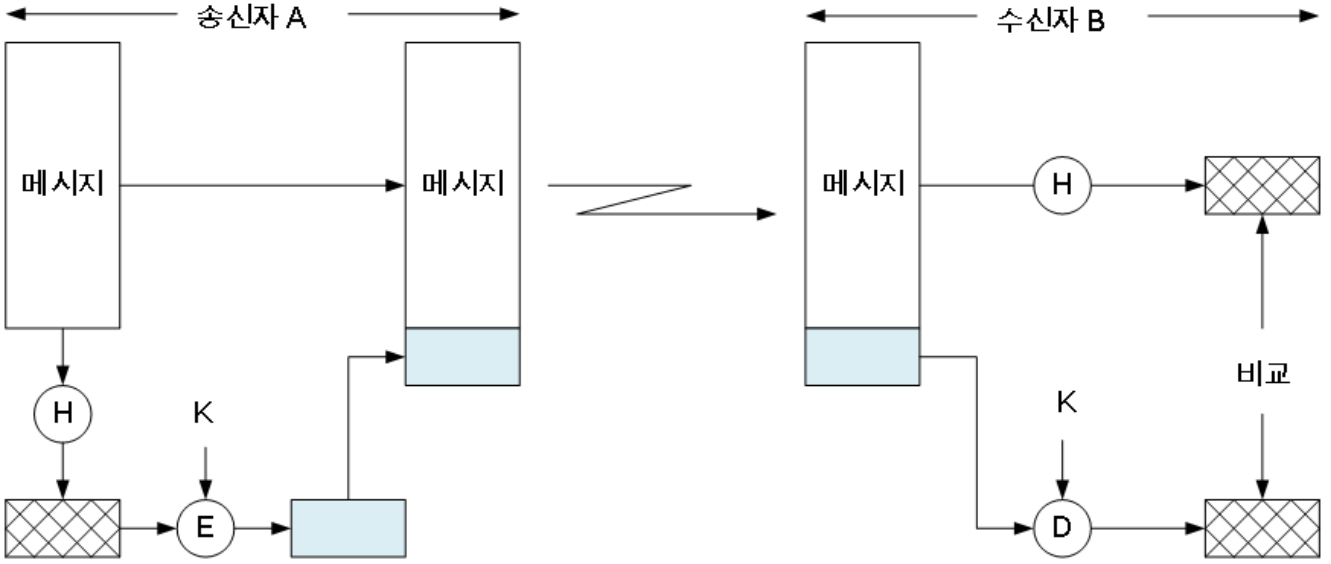
        - NIST 명세 FIPS PUB 113은 DES 알고리즘 사용을 권장

메시지 인증 방법

- 메시지 인증(Message authentication)
  - 인증 방법
    - 메시지 암호화 없이 메시지 인증
      - 일방향 해시함수(One-way hash function)
        - 임의 크기의 메시지 M을 입력으로 메시지 다이제스트(Message digest) H(M)을 출력하는 함수
        - MAC과는 다르게 비밀키를 입력으로 사용하지 않음

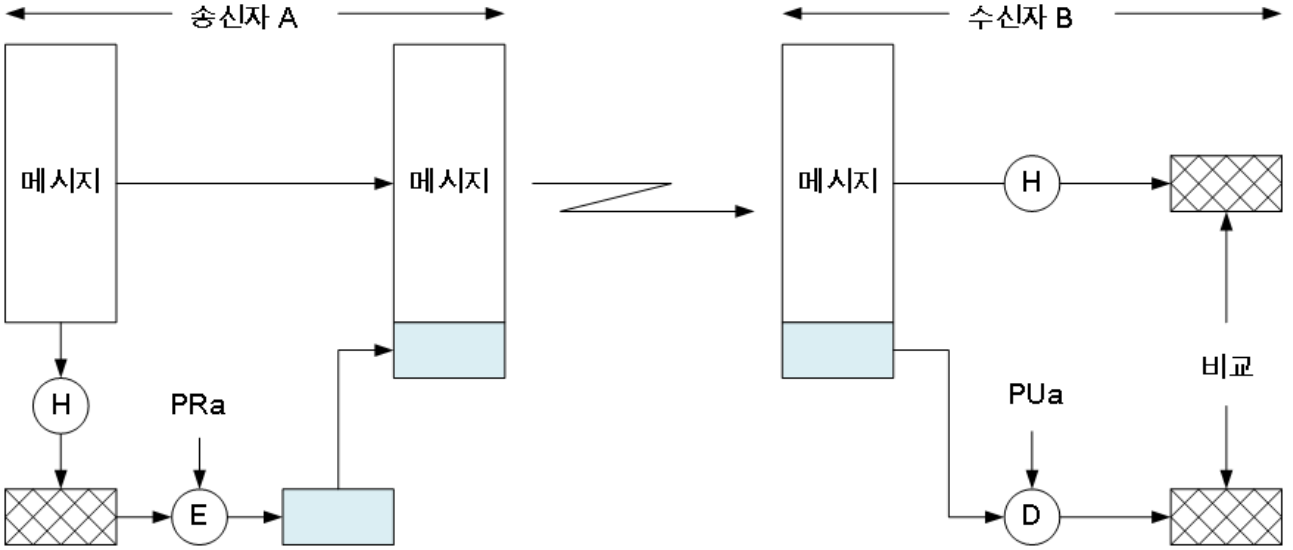
메시지 인증 방법

- 메시지 인증(Message authentication)
  - 인증 방법
    - 메시지 암호화 없이 메시지 인증
      - 일방향 해시함수(One-way hash function)
        - 관용 암호 사용하는 일방향 해시함수
          - 송수신자만이 암호화 키를 소유



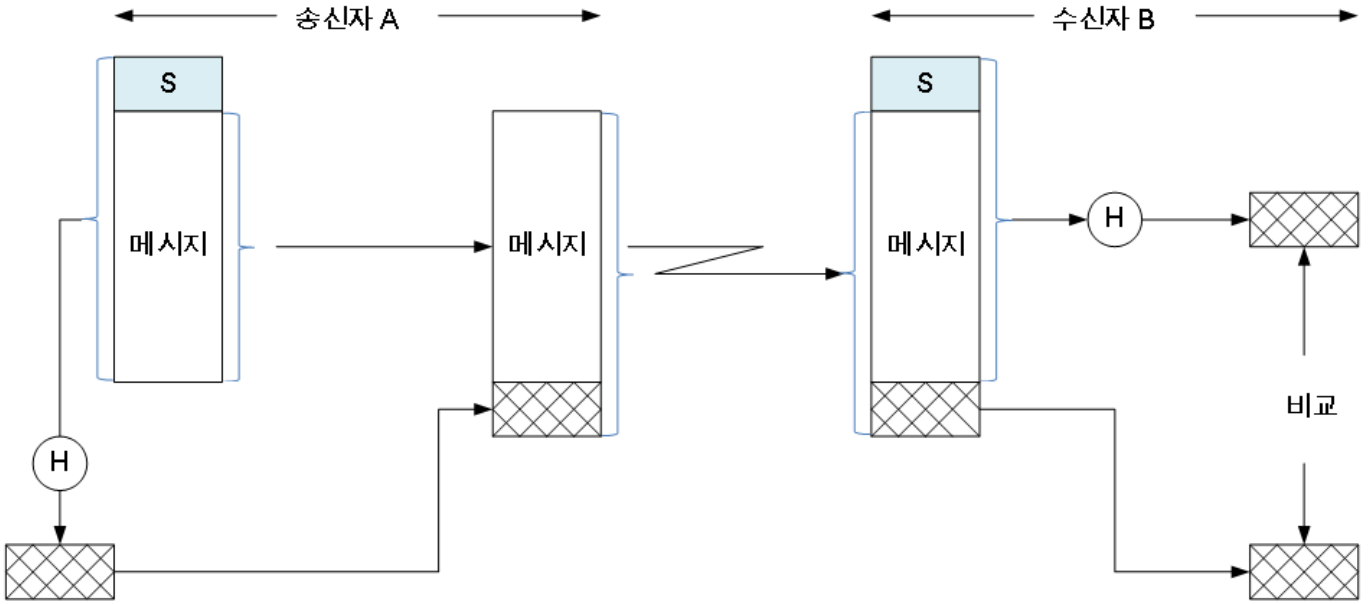
메시지 인증 방법

- 메시지 인증(Message authentication)
  - 인증 방법
    - 메시지 암호화 없이 메시지 인증
      - 일방향 해시함수(One-way hash function)
        - 공개키 암호 사용하는 일방향 해시 함수
          - 송수신자는 공개키를 사용
          - 디지털 서명을 제공
          - 키를 분배할 필요가 없음



메시지 인증 방법

- 메시지 인증(Message authentication)
  - 인증 방법
    - 메시지 암호화 없이 메시지 인증
      - 일방향 해시함수(One-way hash function)
        - 비밀값을 사용하는 일방향 해시함수
          - 암호를 사용하지 않음
          - 송수신자는 공통 비밀값을 가지고 있음



메시지 인증 방법

- 메시지 인증(Message authentication)
  - 인증 방법
    - 메시지 암호화 없이 메시지 인증
      - 일방향 해시함수(One-way hash function)
        - 암호화를 하지 않는 기술을 개발하는 이유
          - 암호 소프트웨어는 속도가 느림
          - 암호 장비의 값이듬
          - 암호 장비는 대용량 데이터처리에 적합
          - 암호 알고리즘은 수출에 제약이 있음

안전 해시함수

- 안전 해시함수(Secure hash function)
  - 인증뿐만 아니라 디지털 서명(Digital signature)에서도 매우 중요한 함수
- 해시함수(Hash function) 요건
  - 어떠한 크기의 데이터 블록에도 적용될 수 있어야 함
  - 일정한 길이의 출력을 생성
  - 계산이 쉽고 하드웨어, 소프트웨어적으로 구현 가능해야 함
  - 일방향 성질(One-way property)을 가짐
    - 주어진 출력값 h에 대해서 H(x) = h가 성립되는 x를 찾는 것이 계산적으로 불가능해야 함

안전 해시함수

- 해시함수(Hash function) 요건
  - 약한 충돌 저항성(Weak collision resistance)을 가짐
    - 주어진 블록 x에 대해서 H(x) = H(y)를 만족하는 y(≠x) 를 찾는 것이 계산적으로 불가능해야 함
  - 강한 충돌 저항성(Strong collision resistance)을 가짐
    - H(x) = H(y)를 만족하는 쌍 (x, y)를 찾는 것이 계산적으로 불가능해야 함
    - 생일공격(Birthday attack)을 막아줌
      - 해시 충돌을 찾아내는 암호해독 공격
      - n명이 있을 때 생일이 같을 확률
        - $P(n) = 1 - (\frac{364}{365})^{C(n,2)}$

n	P(n)
10	11.7%
20	41.1%
23	50.7%
30	70.6%
50	97.0%

안전 해시함수

- 해시함수 보안
  - 전수공격에 대한 해시함수의 강도는 해시코드의 길이에 달려있음
  - 길이가 n-비트인 해시코드에 대한 공격 난이도

프리이미지 저항성	2^n
2차 프리이미지 저항성	2^n
충돌 저항성	$2^{n/2}$

  - Van Oorschot과 Wiener에 의해 24시간 만에 MD5(Message Digest algorithm 5)에서 충돌을 찾아냄

안전 해시함수

- 단순 해시함수(Simple hash function)
  - 입력은 연속된 n-비트 블록으로 간주
    - 세로 덧붙임 검사(Longitudinal redundancy check)
      - 각 비트의 자리별로 패리티를 계산하는 방법
      - 임의의 데이터에 대한 데이터 무결성 검사에 아주 효과적
      - $C_i = b_{i1} \oplus b_{i2} \oplus \bullet\bullet\bullet \oplus b_{im}$

	비트 1	비트 2	● ● ●	비트 n
블록 1	b_{11}	b_{21}		b_{n1}
블록 2	b_{12}	b_{22}		b_{n2}
● ● ●	● ● ●	● ● ●	● ● ●	● ● ●
블록 m	b_{1m}	b_{2m}		b_{nm}
해시 코드	C_1	C_2	● ● ●	C_n

안전 해시함수

- 안전 해시 알고리즘(SHA, Secure Hash Algorithm)
  - NIST가 개발했고 1993년에 FIPS PUB 180으로 출판
    - MD4(Message Digest algorithm 4)에 기초해서 만들어짐
    - 수정판
      - 1995년 FIPS PUB 180-1으로 SHA-1이 나옴
      - 2002년 FIPS 180-2으로 SHA-2(SHA-256, SHA-384, SHA-512)이 나옴
      - 2008년 FIP PUB 180-3(224-비트 버전)이 나옴
        - 하부구조는 모두 동일

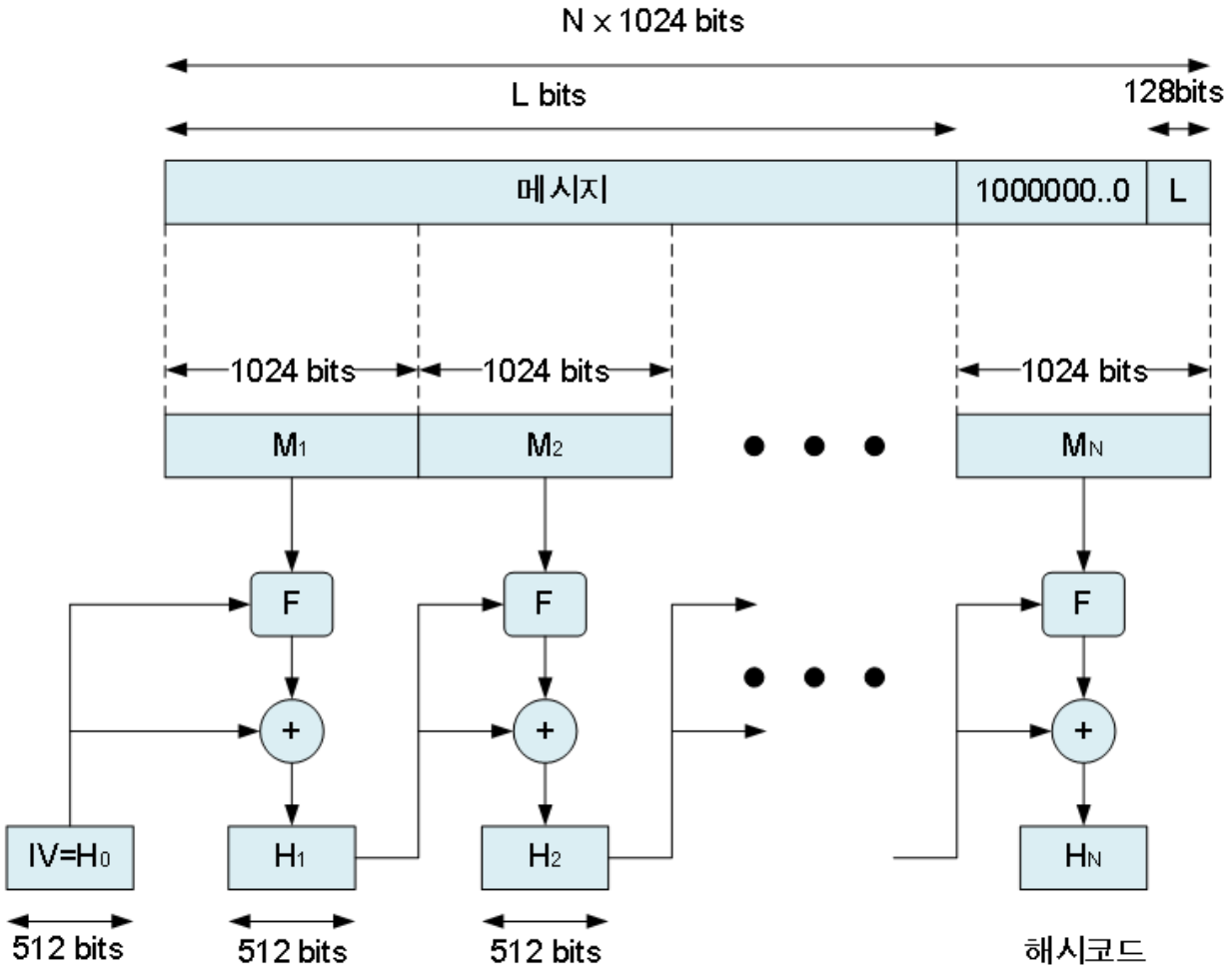
안전 해시함수

- 안전 해시 알고리즘(SHA, Secure Hash Algorithm)
  - SHA 매개변수 비교 표
    - 모든 길이의 단위는 비트

	SHA-1	SHA-224	SHA-256	SHA-384	SHA-512
메시지 다이제스트 길이	160	224	256	384	512
최대 메시지 길이	2^{64}-1	2^{64}-1	2^{64}-1	2^{128}-1	2^{128}-1
블록 길이	512	512	512	1024	1024
단어 길이	32	32	32	64	64
라운드 수	80	64	64	80	80
보안	80	112	128	192	256

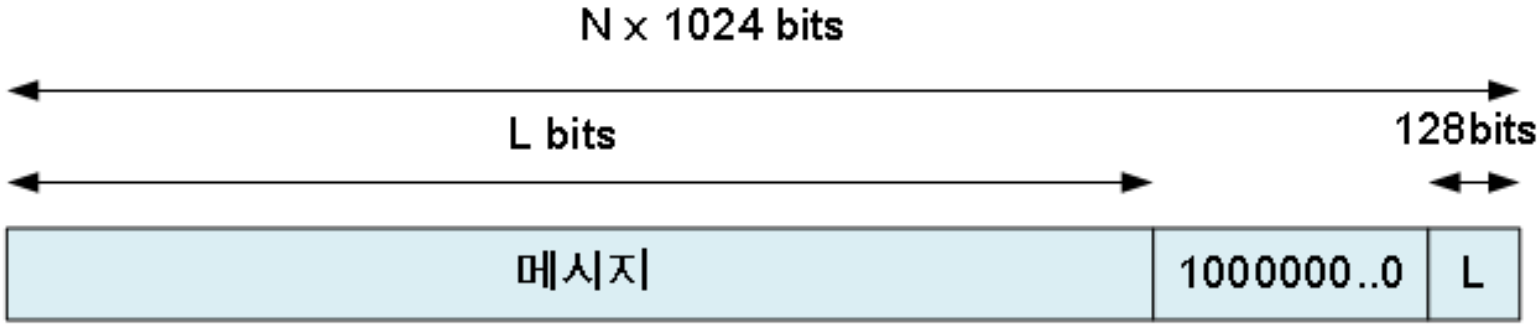
안전 해시함수

- 안전 해시 알고리즘(SHA, Secure Hash Algorithm)
  - SHA-512 구조



안전 해시함수

- 안전 해시 알고리즘(SHA, Secure Hash Algorithm)
  - SHA-512 구조
    1. 패딩 비트 붙이기(Appending padding bits)
       - 메시지 패딩을 추가하여 총 길이를 869(mod 1024)가 되도록 만듦
       - 이미 원하는 길이여도 항상 추가
       - 패딩시 첫 비트는 1, 나머지는 0으로 채움
    2. 길이 붙이기(Append length)
       - 부호 없는 128비트 정수로 패딩전 메시지 길이를 포함



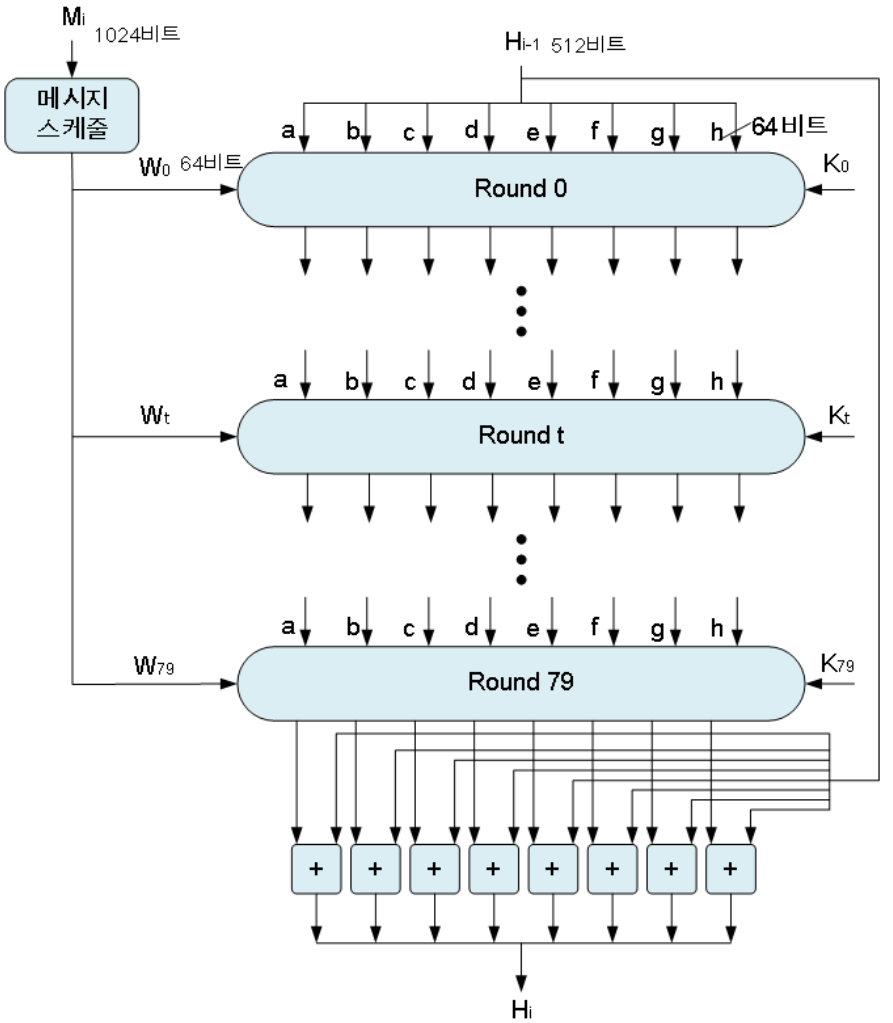
안전 해시함수

- 안전 해시 알고리즘(SHA, Secure Hash Algorithm)
  - SHA-512 구조
    3. MD 버퍼 초기화(Initialize MD buffer)
       - 버퍼를 8개의 64-비트 레지스터로 나타냄
         - 레지스터의 초기값은 16 진법 수로 초기화

a = 6A09E667E3BCC908	e = 510E527FADE682D1
b = BB67AE8584CAA73B	f = 9B05688CEB3E6C1F
c = 3C6EF372FE94F82B	g = 1F83D9ABFB41BD6B
d = A54FF53A5F1D36F1	h = 5BE0CDI9137E2179

안전 해시함수

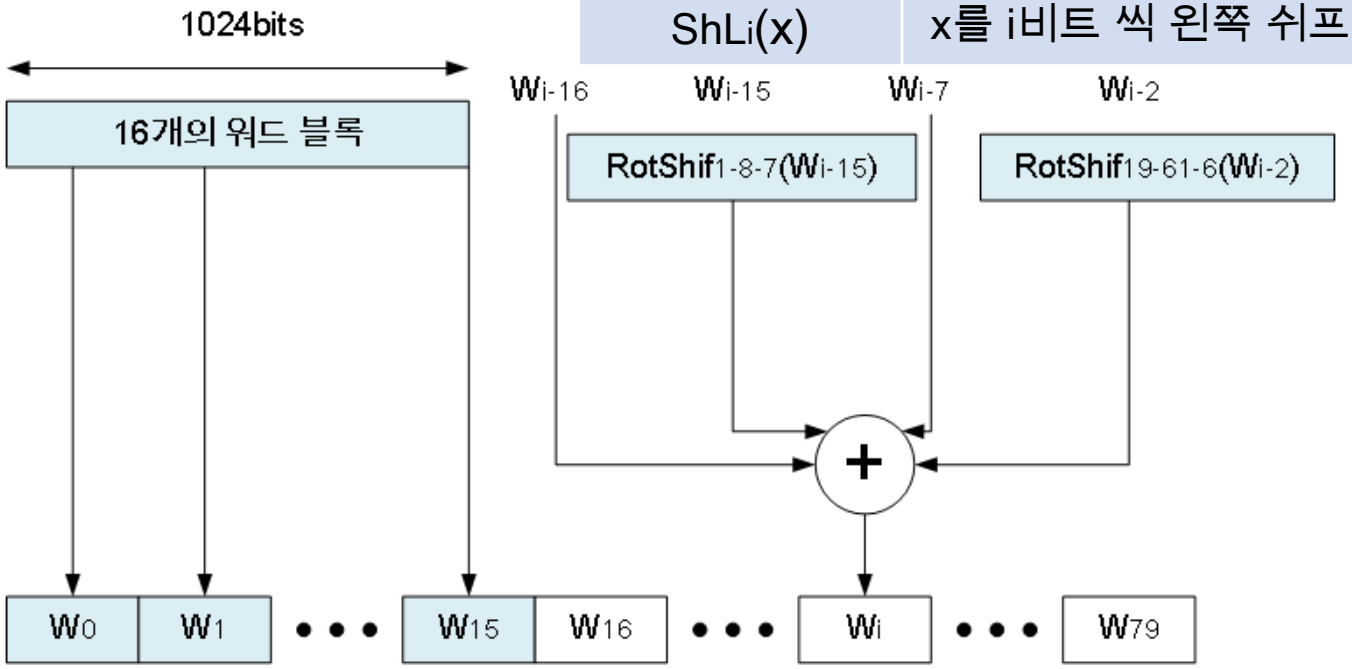
- 안전 해시 알고리즘(SHA, Secure Hash Algorithm)
  - SHA-512 구조
    4. 1024-비트 블록 메시지 처리
       - 라운드 함수에 입력으로 512비트, 스케줄된 메시지 워드 64비트, 덧셈 상수 K 사용
       - 각 블록 처리는 80라운드로 이루어짐



안전 해시함수

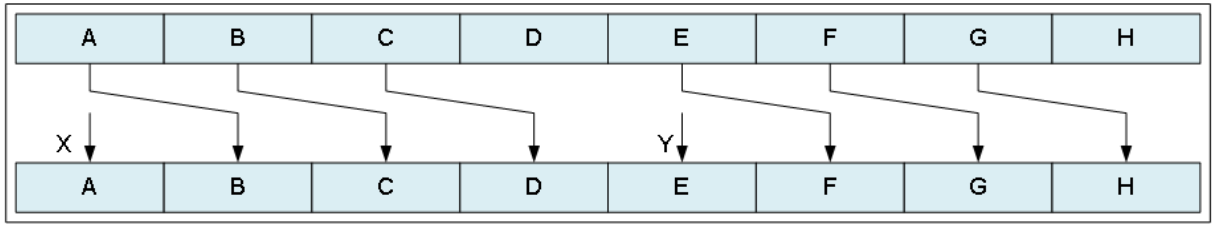
- 안전 해시 알고리즘(SHA, Secure Hash Algorithm)
  - SHA-512 구조
    4. 1024-비트 블록 메시지 처리
       - 메시지 스케줄

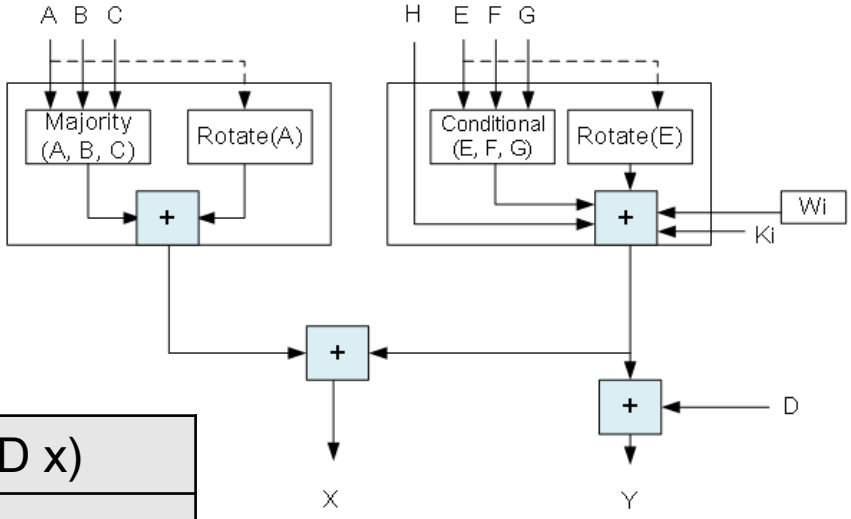
용 어	정 의
$RotShift_{l-m-n}(x)$	$RotR_l(x) \oplus RotR_m(x) \oplus ShL_n(x)$
$RotR_i(x)$	x를 i만큼 오른쪽 쉬프트
$ShL_i(x)$	x를 i비트 씩 왼쪽 쉬프트 하고 0을 왼쪽으로 채움



안전 해시함수

- 안전 해시 알고리즘(SHA, Secure Hash Algorithm)
  - SHA-512 구조
    5. 1024-비트 블록 메시지 처리
       - 라운드 함수 구조
         - 80개의 상수인 K_0~K_{79}가 사용





Majority(x, y, z)	(x AND y) ⊕ (y AND z) ⊕ (z AND x)
Conditional(x, y, z)	(x AND y) ⊕ ($\bar{X}$ AND z)
Rotate(x)	$RotR_{28}$ (x) ⊕ $RotR_{34}$ (x) ⊕ $RotR_{39}$ (x)
+	2^{64} 합 연산 모듈

메시지 인증 코드

- HMAC(Hashed Message Authentication Code) 개요
  - RFC 2104로 출판되었고 IP Security 용 MAC으로 필수적 사용이 규정
    - 전송 계층 보안(TLS, Transport Layer Security)과 안전한 전자 결제(SET, Secure Electronic Transaction) 같은 다른 인터넷 프로토콜에서도 사용됨
  - HMAC 설계 목표
    - 수정하지 않고 쓸 수 있는 해시함수를 만듦
    - 더 좋은 해시함수가 있으면 기존의 해시함수를 쉽게 교체
    - 심각하게 기능저하를 유발하지 않고 해시함수의 원래 성능을 유지
    - 키를 보다 쉽게 다루고자 함
    - 인증 메커니즘의 강도에 대해 암호해독을 확실히 파악할 수 있도록 함

메시지 인증 코드

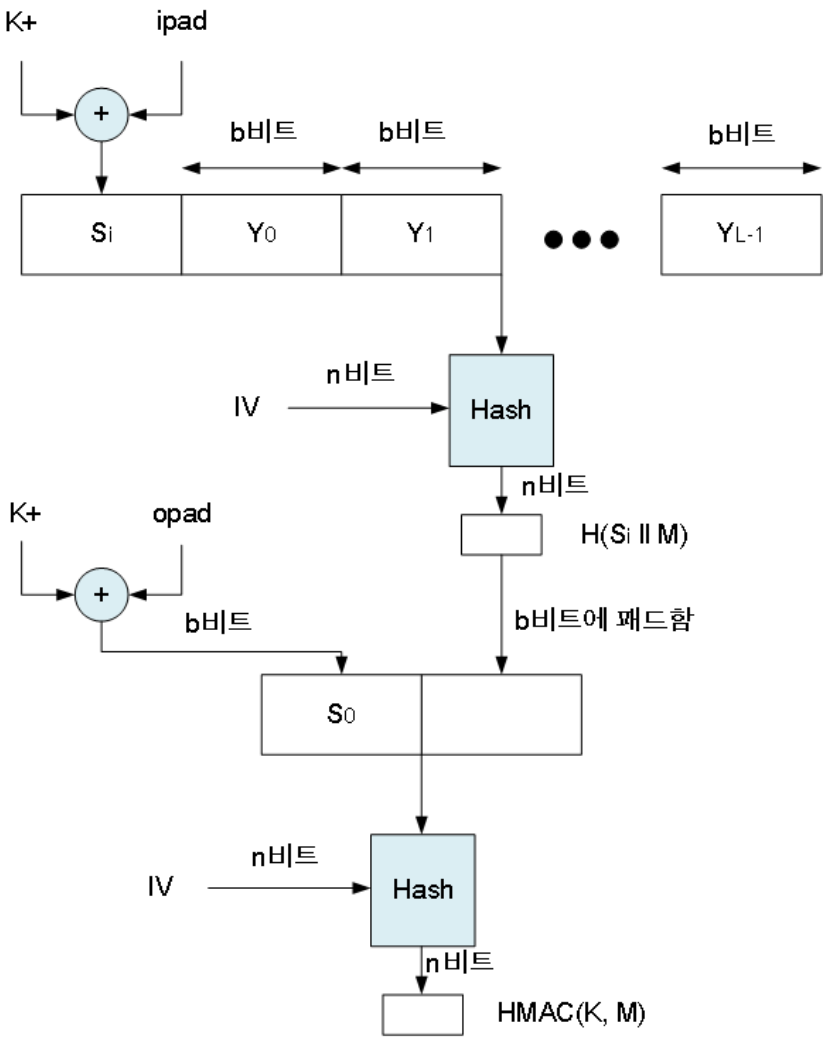
- HMAC(Hashed MAC) 용어

용어	정의
H	내장된 해시함수
M	HMAC의 입력 메시지
Yi	M의 i번째 블록
L	M의 블록 수
b	블록의 비트 수
n	내장된 해시함수에 의해 생성된 해시코드의 길이
K	비밀키, 키의 길이가 b보다 길면 n-비트 키를 생성하는 해시함수에 입력으로 사용
K^+	K의 왼쪽에 0을 붙여서 길이가 b비트가 되도록 한 것
ipad	00110110(16진수 36)을 b/8번 반복한 2진 수열
opad	01011100(16진수 5C)을 b/8번 반복한 2진 수열

메시지 인증 코드

- HMAC(Hashed MAC) 구조

1. b-비트의 K^+를 만들기 위해 K의 왼쪽에 0을 패딩
2. K^+와 ipad를 XOR 하여 b비트 Si 블록을 만듦
3. Si에 메시지 M을 붙이고 IV와 해시 함수에 입력
4. K^+와 opad를 XOR 하여 b비트 S_0을 만듦
5. 3단계의 출력값을 S_0에 붙이고 IV와 해시 함수에 입력
6. 결과값 출력



메시지 인증 코드

- 블록 암호기반 MAC
  - 암호기반 메시지 인증 코드(CMAC, Cipher-based Message Authentication Code)
    - 운용모드는 AES와 3DES를 사용
      - AES인 경우
        - 블록 길이 b = 128비트, 키 길이 K = 128, 192, 또는 256비트
      - 3DES인 경우
        - 블록 길이 b = 64비트, 키 길이 K = 112 또는 168비트

메시지 인증 코드

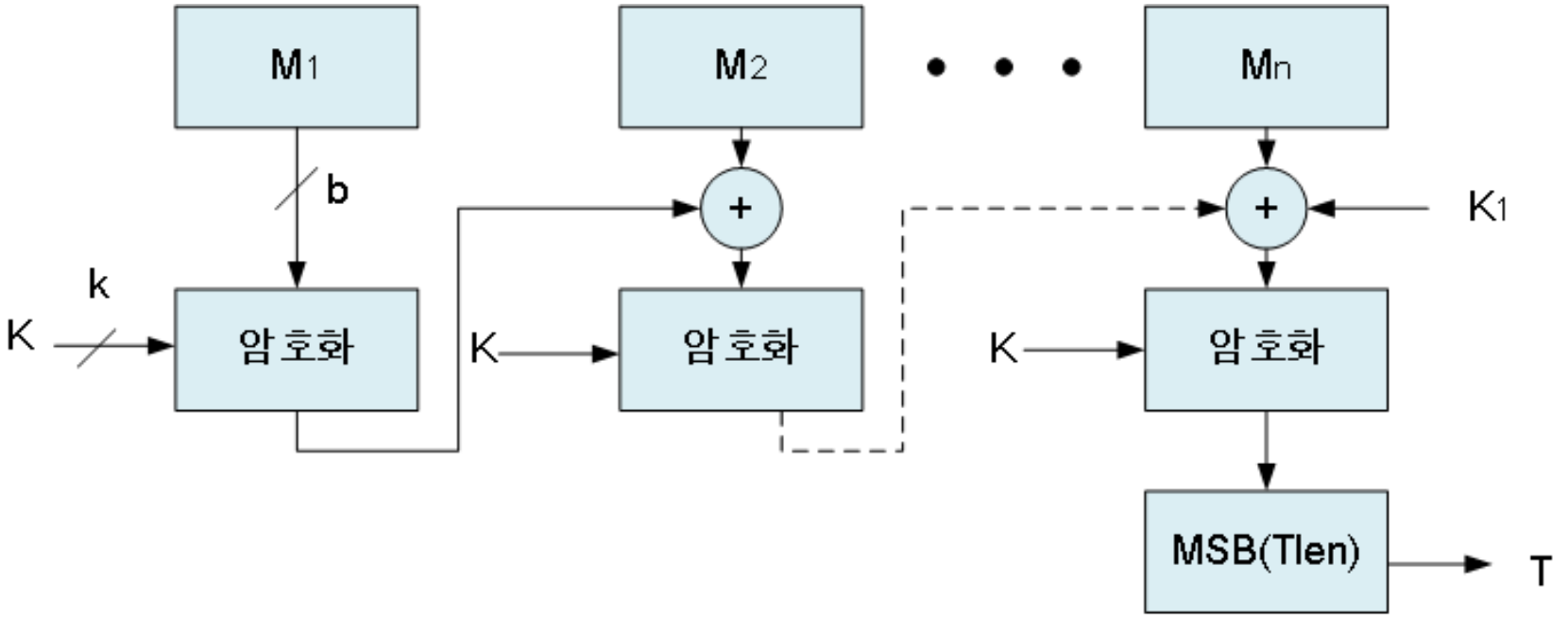
- 블록 암호기반 MAC
  - 암호기반 메시지 인증 코드(CMAC, Cipher-based Message Authentication Code)
    - 계산식과 용어

계산식
$C_1 = E(K, M_1)$
$C_2 = E(K, [M_2 \oplus C_1])$
$C_3 = E(K, [M_3 \oplus C_2])$
•
•
•
$C_n = E(K, [M_n \oplus C_{n-1} \oplus K_1])$
$T = MSB_{Tlen}(C_n)$

용어	정의
T	메시지 인증 코드, “태그(Tag)”
Tlen	T의 비트 길이
MSBs(X)	비트열 X의 왼쪽부터 S개 비트
K_1	$E(0, K) <<1$
K_2	$E(0, K_1) <<1$

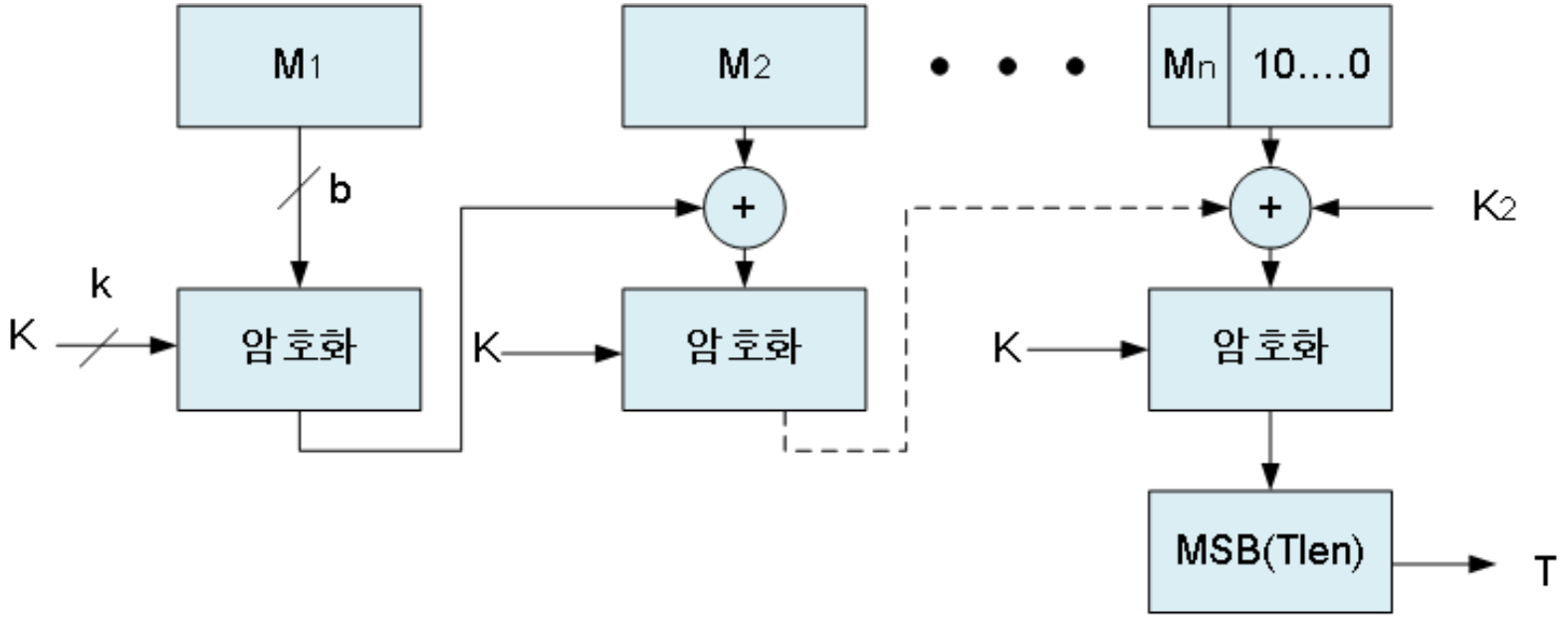
메시지 인증 코드

- 블록 암호기반 MAC
  - 암호기반 메시지 인증 코드(CMAC, Cipher-based Message Authentication Code)
    - 메시지 길이가 블록 길이의 정수배일 때
      - k비트 키 K와 b비트 서브키 K1을 이용



메시지 인증 코드

- 블록 암호기반 MAC
  - 암호기반 메시지 인증 코드(CMAC, Cipher-based Message Authentication Code)
    - 메시지 길이가 블록 길이의 정수배가 아닐 때
      - 마지막 블록의 오른쪽에 패딩을 함
      - k비트 키 K와 b비트 서브키 K2를 이용

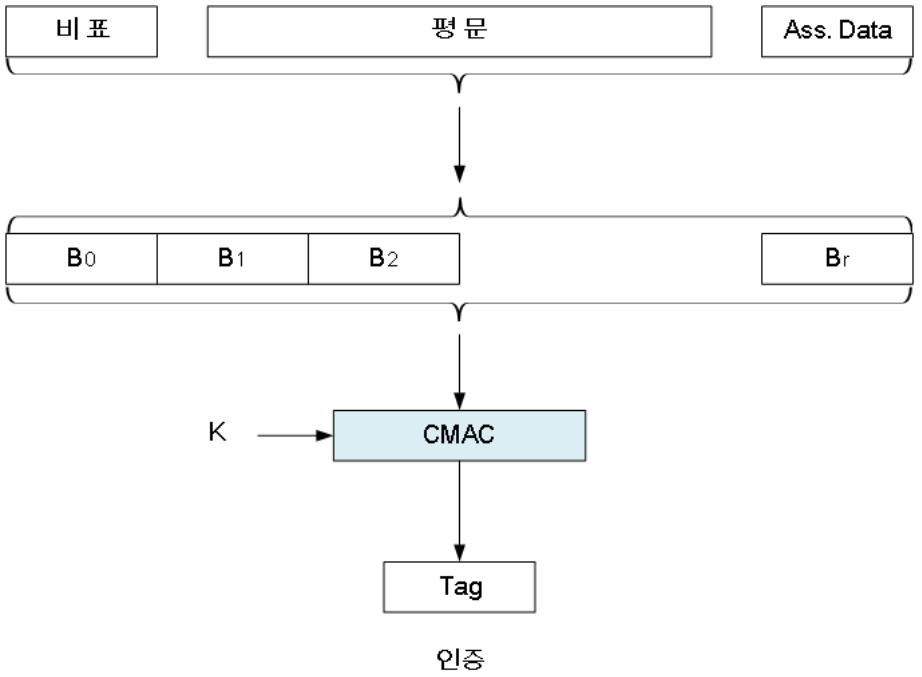


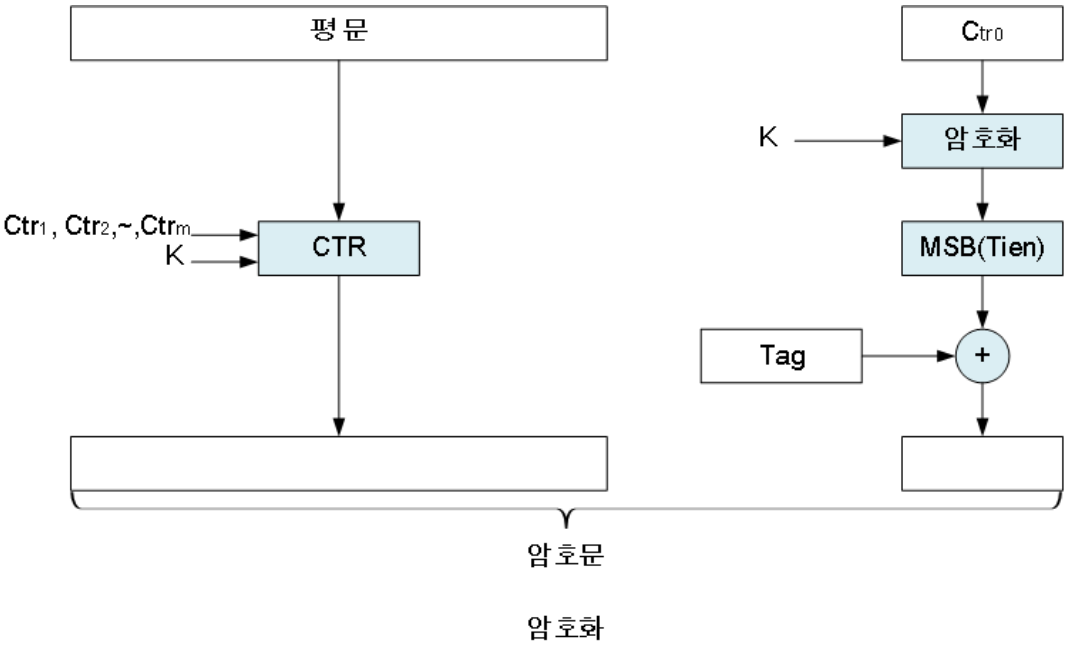
메시지 인증 코드

- 블록 암호기반 MAC
  - 암호 블록 체인 카운터 MAC(CCM, Counter with Cipher block chaining-Message authentication code)
    - 인증된 암호화(Authentication encryption)모드라고도 함
      - 통신상 기밀성과 인증(무결성)을 동시에 보호하는 암호 시스템을 설명할 때 사용하는 용어
    - AES 암호, CTR 운용 모드, CMAC를 사용
    - 암호화와 인증에 동일한 키 K를 사용
    - CCM 암호화 과정에 입력되는 3가지 요소
      - 인증하고 암호화할 데이터
      - 인증을 하는 유관 데이터
      - 비표

메시지 인증 코드

- 블록 암호기반 MAC
  - 암호 블록 체인 카운터 MAC(CCM, Counter with Cipher block chaining-Message authentication code)
    - 구조





감사합니다!