[Security]

Hash Algorithm

Bill Kim(김정훈) | ibillkim@gmail.com

목차

Hash Argorithm

- 단방향(One Way) 암호화 기법
- 평문(Plain Text)를 암호화된 텍스트(Cipher Text)를 변환
- 입력값이 달라도 출력값은 고정된 길이 반환
- 해시 함수를 통하여 암호화 텍스트 추출

Hash Function 특성

- 역상저항성

입력값 A에 의해 B가 출력되었다면, 출력된 B값만 주어졌을 때 입력값인 A값을 찾는 것이 계산적으로 불가능함을 의미한다.

- 제2역상저항성

입력값 A와 출력값 B가 모두 주어졌을 때, 똑같은 B를 반환하는 A2 를 찾아내거나 만들어내는 것이 계산적으로 불가능함을 의미한다. 단방향 암호화라고도 하는데 양방향 암호화가 A를 B로 암호화하고 다시 B를 A로 복호화하여 원본 내용을 확인할 수 있다면, 단방향 암호화는 A로 만들어진 B를 가지고 다시 A로 역산할 수 없다.

- 충돌저항성

똑같은 B라는 출력값이 나오는 X가 단일하지 않고 중복이 되는 또 다른 Xn을 발견하는 것이 계산적으로 어려운 성질을 의미한다. 충돌저항성은 제2역상저항성의 외부효과(부수효과, Side effect) 이자 부분집합이다.

Hash Function 효과

- 압축효과

암호화 해시 함수가 반환하는 일정한 길이의 작은 해시값만으로 크기가 거대한 데이터의 무결성을 보장할 수 있는 외부효과를 의미한다. 예를 들어 SHA-256의 경우 100GB의 파일도 단 256bit의 해시값으로 그 내용의 무결성을 보장할 수 있다.

- 눈사태 효과

눈사태 효과 란 입력값의 아주 작은 변화로도 결과값이 전혀 다르게 도출되는 효과를 의미한다.
입력값에 점 하나만 추가되어도 전혀 다른 출력값이 출력된다.
또한 변경되는 부분에 있어 어떠한 규칙성도 찾을 수 없다.

Hash Function 종류

제산법: H(key) = Key % Prime No, 키를 소수(Prime Number)로 나눈 값으로 주소 결정

제곱법: (Key^2)한 값의 중간 부분으로 주소 결정

폴딩법: 키 값을 여러 부분으로 분류하여 분류한 부분을 더하거나 XOR하여 주소를 계산

기수 변환법: 특정 진법으로 표현한 레코드 키 값을 다른 진법으로 간주하고 키 값을 변환하여 홈 주소를 취함

계수 분석법: 주어진 모든 키 값들에서 그 키를 구성하는 자릿수들의 분포를 조사하여 비교적 고른 분포를 보이는 자릿수를 택함

사용예

파일의 Checksum(검사합) 활용 : 파일 변조 체크

암호 저장 : 암호값을 저장하여 비교

데이터 탐색 : 해시테이블을 활용하여 바로 원하는 데이터 추출

해시 충돌

사용자가 설정한 key가 해쉬 함수에 의해 배열의 인덱스로 변환되었을 때, 이 인덱스로 배열에 접근했을 때 어떤 값이 이미 그 자리를 차지하는 경우를 말한다.

해시 함수를 통하여 나온 값이 이미 있는 값일 경우 이를 해시 충돌이라고 한다.

해시 충돌 해결법

1. 체이닝(Chaining)

버켓 내에 연결리스트(Linked List)를 할당하여, 버켓에 데이터를 삽입하다가 해시 충돌이 발생하면 연결리스트로 데이터들을 연결하는 방식

2. 개방 주소법(Open Addressing)

체이닝의 경우 버켓이 꽉 차더라도 연결리스트로 계속 늘려가기에, 데이터의 주소값은 바뀌지 않는다.(Closed Addressing) 하지만 개방 주소법의 경우에는 다르다.

해시 충돌이 일어나면 다른 버켓에 데이터를 삽입하는 방식을 개방 주소법이라고 한다. 개방 주소법은 대표적으로 3가지가 있다.

(1)선형 탐색(Linear Probing) : 해시 충돌 시 다음 버켓 혹은 몇 개를 건너띄어 데이터를 삽입

(2)제곱 탐색(Quadratic Probing) : 해시 충돌 시 제곱만큼 건너뛴 버켓에 데이터를 삽입(1, 4, 9, 16, ...)

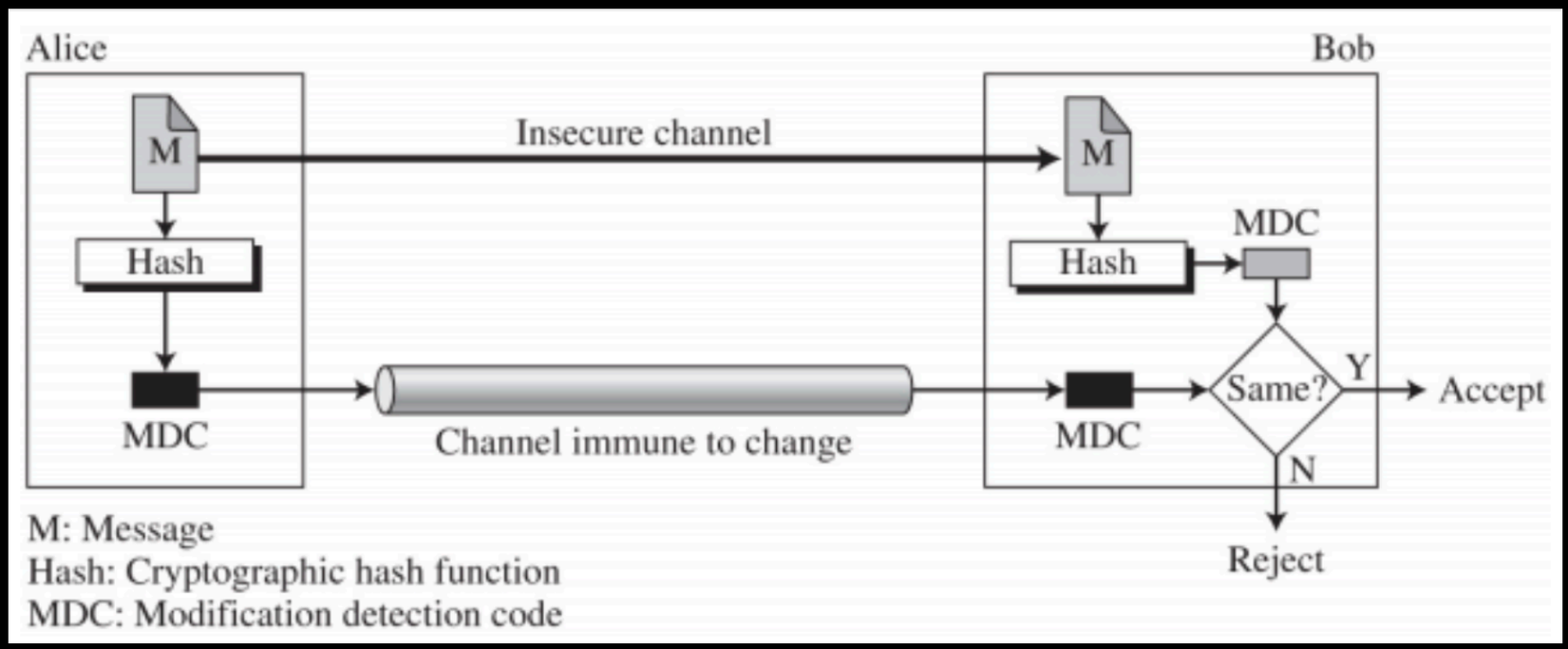
(3)이중 해시(Double Hashing) : 해시 충돌 시 다른 해시 함수를 한번 더 적용한 결과를 이용

MDC

메시지의 무결성(메시지가 변하지 않았다는 것.) 보장하는 메시지 다이제스트

수신한 메시지의 MDC를 계산하고 송신측이 보내준 MDC와 비교하여 동일한지 비교

무결성은 보장하나 메시지 송신자는 알 수 없음



MD5

MD5(Message-Digest algorithm 5)는 128비트의 해시 값을 생성하는 해시 함수
MD5는 암호화가 아닌 단방향 해시 알고리즘이다.

MD5 코드는 32비트로 총 32자의 16진수로 이루어져 있다.
그렇다면 나올 수 있는 총 가짓 수는 16^32(16의 32승)이 된다.

하지만 문제는 이세상의 모든 어구(입력값)는 16^32보다 많다는 것이 있다.

SHA-256

MSHA-256은 SHA(Secure Hash Algorithm) 알고리즘의 한 종류로서 256비트의 해시 값을 생성하는 해시 함수이다.

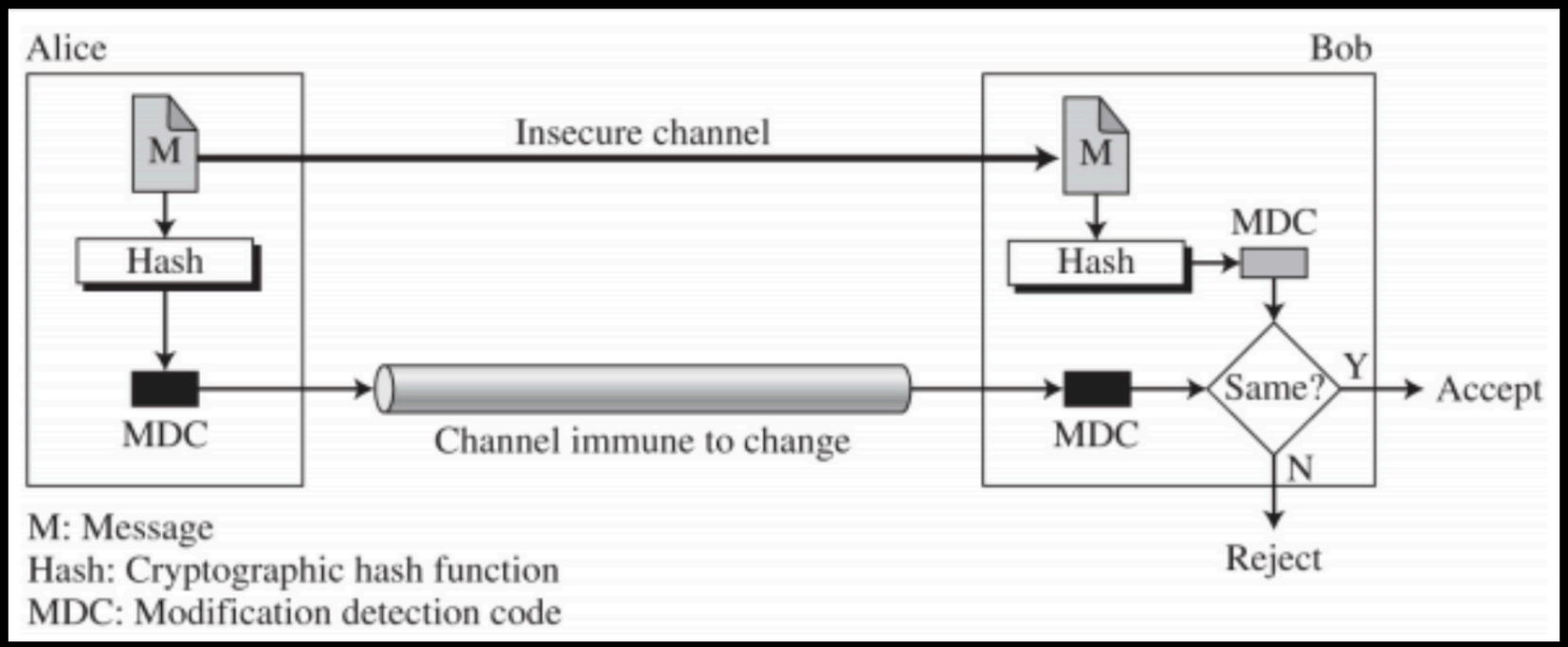
SHA-256은 미국의 국립표준기술연구소(NIST; National Institute of Standards and Technology)에 의해 공표된 표준 해시 알고리즘인 SHA-2 계열 중 하나이며 블록체인에서 가장 많이 채택하여 사용하고 있다.

MDC

메시지의 무결성(메시지가 변하지 않았다는 것.) 보장하는 메시지 다이제스트

수신한 메시지의 MDC를 계산하고 송신측이 보내준 MDC와 비교하여 동일한지 비교

무결성은 보장하나 메시지 송신자는 알 수 없음



MAC

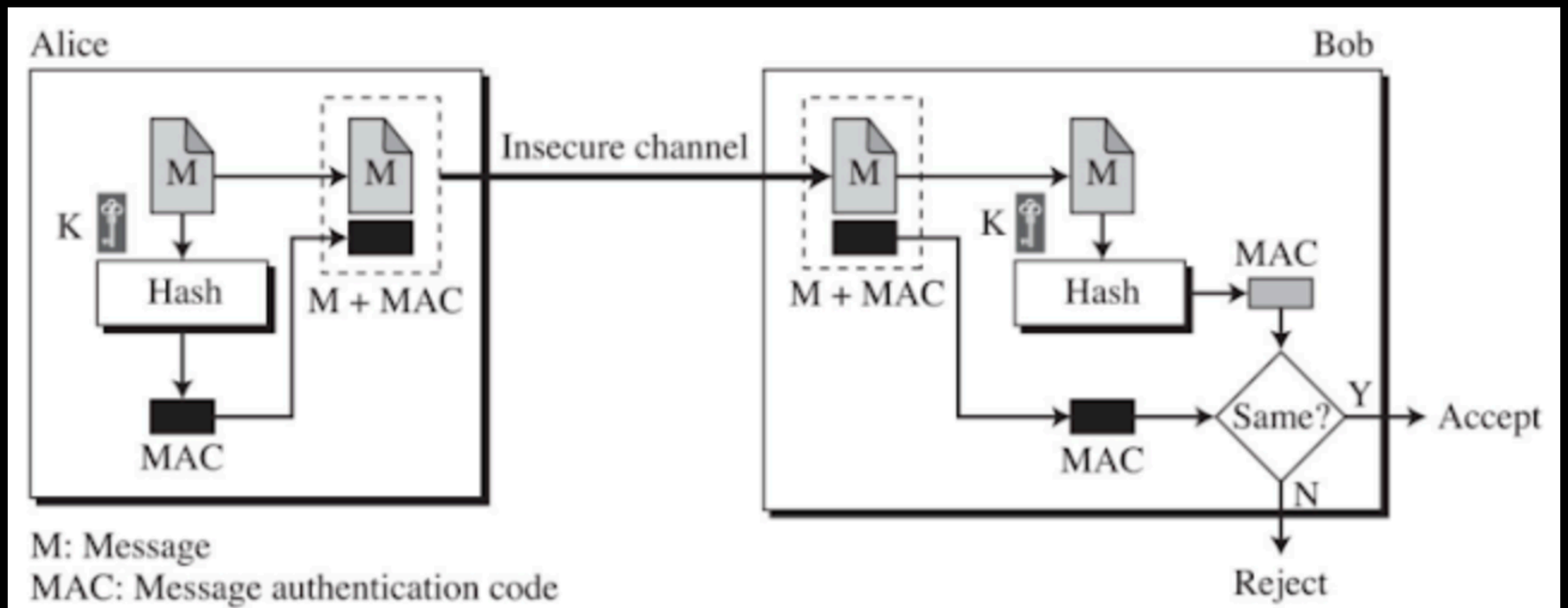
해시함수 + 대칭키로 메시지 무결성을 인증하고 거짓행세(메시지 인증으로 검출)를 검출

무결성을 확인하고 메시지에 대한 인증을 하는 기술

임의 길이의 메시지와 송신자 및 수신자가 공유하는 키. 두 개를 입력으로 하여 고정 비트길이의 출력을 만드는 함수. 이 출력값을 MAC이라고 함

적극적 공격인 데이터 위조 같은 것을 방어하는데 사용
키 K는 오직 송,수신자만 알고 있음

MAC



M: Message
MAC: Message authentication code
K: A shared secret key

축소 MAC(Nested MAC)

MAC의 안전성을 높이기 위한 방법

최종 MAC을 생성하기 위해 두번의 해시함수를 거침

1. H(K||M)을 통해 1차 다이제스트 M1을 생성 함
2. H(K||M1)을 통해 2차(최종) 다이제스트를 생성함

HMAC(Hash MAC)

Nested MAC의 표준을 만들었는데 그것이 HMAC.
좀더 복잡하며, n 비트의 패딩을 붙이고 두번의 해싱 단계를 거침

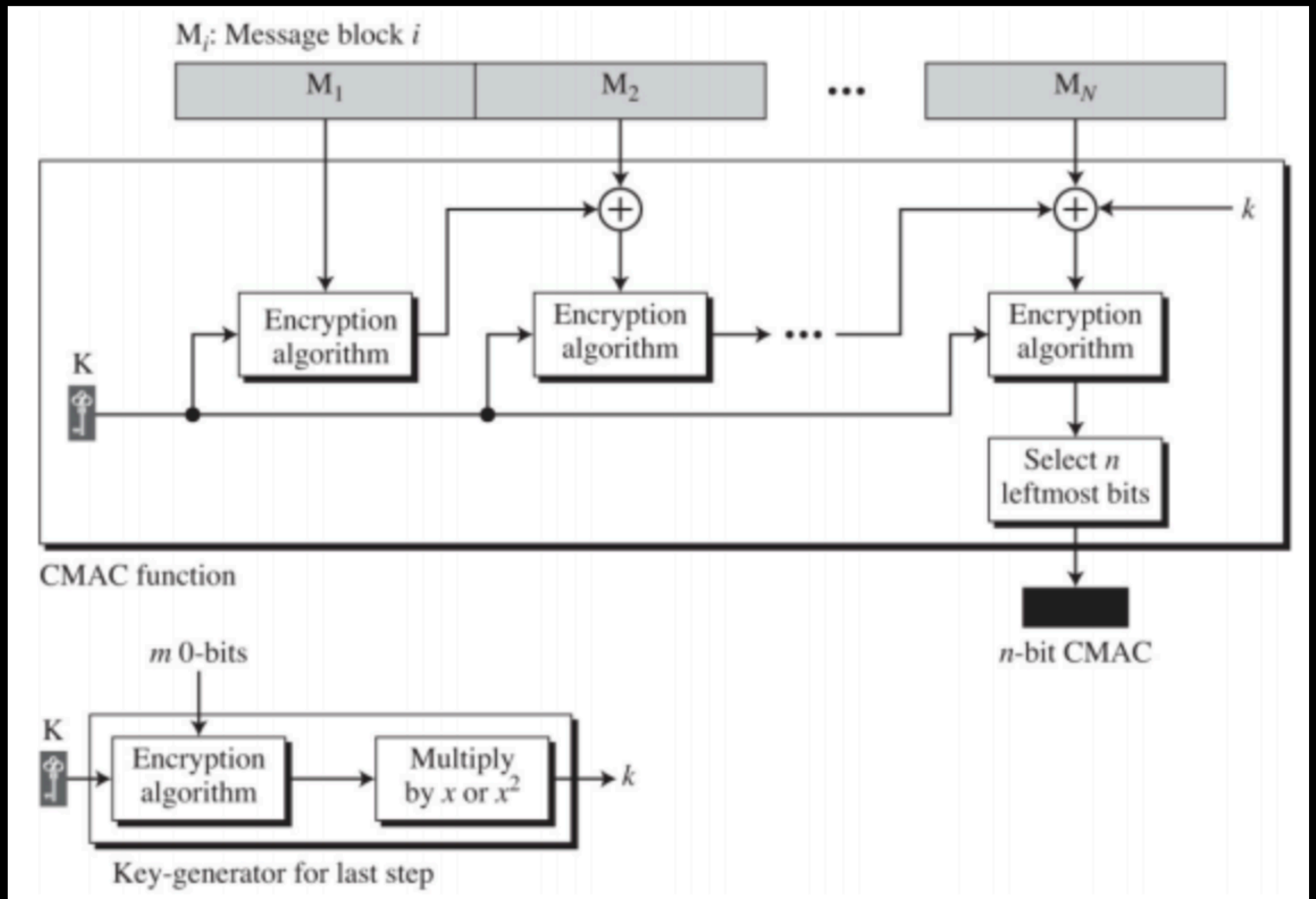
CMAC(Cipher-based Message Authentication Code)

대칭키 암호시스템의 암호 블록체인(CBC)모드와 유사한 표준

메시지를 n개의 블록으로 나누고 키와 블록 1번부터 n번까지 각 블럭과 암호화 기법을 적용한 후 다음번 블록과 XOR를 적용. 이것을 마지막 n번 블록까지 수행.

키 생성: 0으로 이루어진 m비트의 블록을 비밀키 K로 암호화하여 패딩이 적용되지 않았을 경우 x를 곱하고 패딩이 적용되면 x^2을 곱함

CMAC(Cipher-based Message Authentication Code)



<CMAC 동작 방식>

MAC 사용예

IPSec

IP 계층에서 보안 기능을 추가할 때 사용. 통신 내용의 인증, 무결성 제공하기 위해 MAC을 사용

SSL/TLS

웹에서 온라인 쇼핑시 사용되는 통신 프로토콜, 인증, 무결성 확인을 위해 MAC을 사용

MAC 문제점

MAC의 키 배송 문제

MAC에서는 송신자와 수신자가 키를 공유해야 함. 이를 위해 대칭키 암호화 방식에서 처럼 키 배송 문제가 일어남.

MAC 공격 방법 예

재전송 공격(replay 공격): 적극적 공격자 멜로리는 저장해둔 MAC값을 반복해서 송신.

방어방법

① 순서번호

송신 메시지에 순서번호를 붙이면서, MAC 값의 계산에서는 순서번호도 메시지에 포함시킨다.

② 타임스탬프

송신 메시지에 현재 시각을 넣음. 이전 메시지가 오면 MAC값이 바르더라도 오류로 판단. 단, 송수신자 사이에 타임 동기화 필요.

③ 비표(nonce)

메시지 수신전, 수신자가 송신자에게 일회용 랜덤 값(비표) 전송. 송신자가 메시지 안에 비표를 넣어 MAC값을 계산. 매번 비표가 바뀌므로 재전송 공격이 막힘

MAC 문제점

- 제3자에 대한 증명

앨리스로부터 메시지를 받은 밥이 메시지가 정말 앨리스가 보낸 것이라는 것을 제 3자인 검증자 빅터에게 증명할 수 없음.

증명하기 위해선 공유키(앨리스와 밥이 공유하고 있는 키)를 빅터에게 알려줘야만 하고, 공유키는 앨리스와 밥이 가지므로 둘 중 누가 메시지를 작성했는지 검증할 수 없음.

- 부인 방지

밥이 MAC과 함께 메시지를 받고, 메시지가 앨리스로부터 온것이라는 것은 확실히 증명이 가능함.

하지만 앨리스가 전송 자체를 부정할 경우 제 3자에게 이를 증명할 수 없음.

즉, 앨리스가 송신자체를 부인(repudiation)하는 것. 따라서 MAC으로는 부인 방지를 할 방법이 없음

- 해결방법

전자서명: 공개키기반이므로 송신자는 자신의 개인키로 메시지에 서명하고, 수신자는 송신자의 공개키로 서명을 검증하여 부인방지를 함

References

References

[1] Hash - MD5와 SHA256 해시(Hash)와 암호화(Encryption)의 차이 :
https://jongmin92.github.io/2019/12/18/Java/hash/

[2] 암호화 알고리즘 종류 :
https://jusungpark.tistory.com/34

[3] 암호화(Encryption)와 해시(Hash) :
https://baekjungho.github.io/technology-encrypt/

[4] 해시함수에 대한 개념 정리하기 :
https://velog.io/@zuyonze/해시함수에-대한-개념-정리하기

[5] 해시 알고리즘 요약 정리, 테스트 코드 :
https://hsp1116.tistory.com/35

References

[6] 해시 함수, MDC :
https://m.blog.naver.com/PostView.nhn?blogId=sdug12051205&logNo=221575584222&proxyReferer=https:%2F%2Fwww.google.com%2F

[7] 해시함수(Hash Function) :
https://allfriend123.blogspot.com/2017/11/hash-function.html

[8] 정보보안기사 정리 2 - 해시함수, MDC, MAC(메세지 인증 코드) : http://vnfmsehdy.blogspot.com/2016/06/2-mdc-mac.html

[9] 자료구조 - 해시함수 종류와 충돌 처리 방식 :
https://galid1.tistory.com/170

[10] 비밀번호 해시(MD5, SHA1)알고리즘 :
http://blog.daum.net/_blog/BlogTypeView.do?blogid=0qxd8&articleno=387&categoryId=24&regdt=20160506212625

Thank you!