해시 함수

유튜브 주소 : https://youtu.be/wtOWMSOIagI

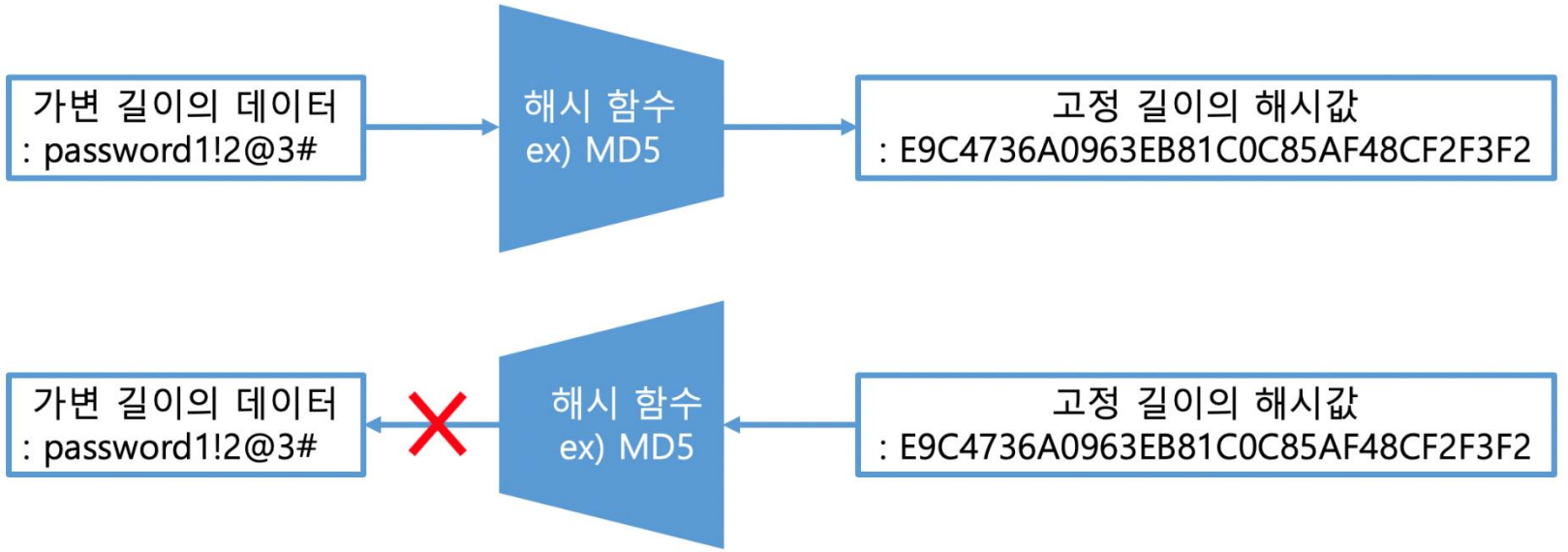
해시 함수

SHA-1

SHA-1 구현 코드 분석

해시 함수

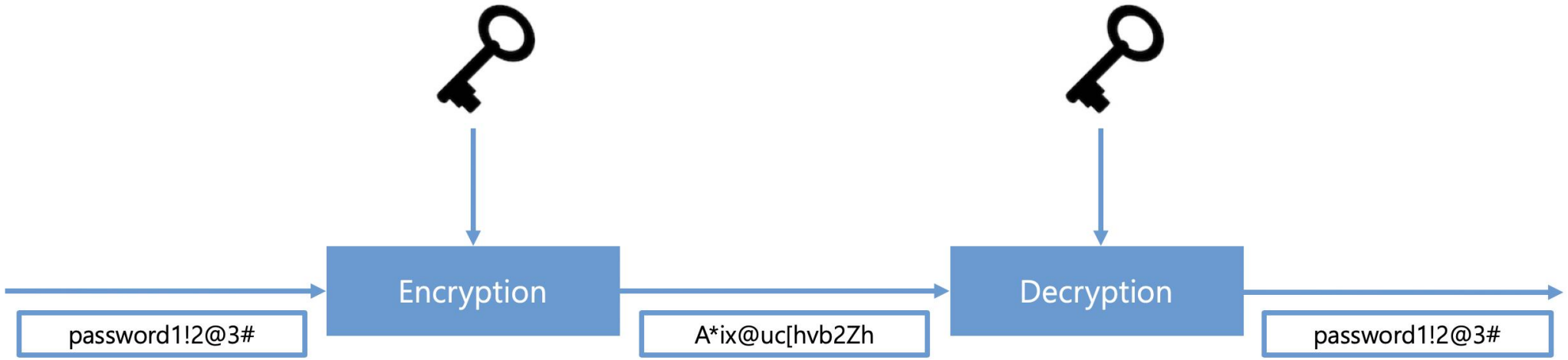
- 임의의 메시지(키)를 고정된 길이의 출력 값(해시 값)으로 바꾸는 함수
  - 키 : 매핑 전 원래의 데이터 값
  - 해시 값 : 매핑 후의 데이터 값
  - 해싱 : 매핑하는 과정
- 해시 함수의 목표
  - 복호화 키 없이 암호화된 데이터로부터 기존의 데이터로 복호화 시킬 수 없게 만드는 것



해시 함수

해시함수와 암호화의 차이점

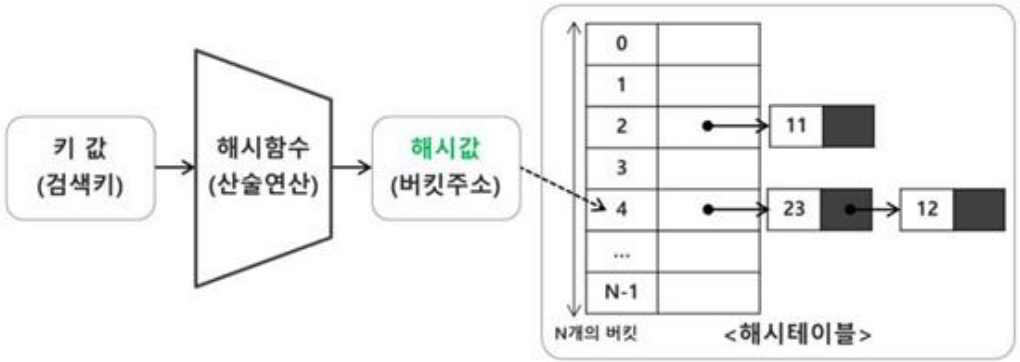
- 암호화 – 통신 중점
  - 양방향 암호화, 복호화 가능
- 해시 함수 – 데이터 보안 중점
  - 단방향 암호화, 복호화 불가능



해시 함수

해시 함수의 목적

- Fast Table Lookup(해시 테이블)
  - 자료구조, 데이터베이스 영역에서의 사용
  - 해시값을 주소로 하는 해시테이블 사용
  - 빠른 CRUD를 위함



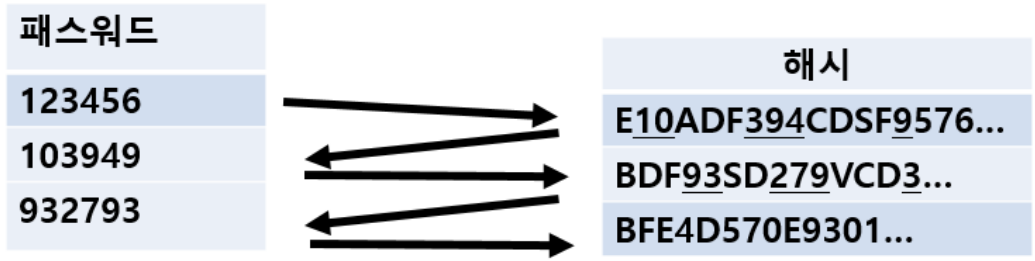
- Message digests(메시지 축약)
  - 임의 크기 메시지를 특정한 고정 크기 블록으로 만드는 과정
  - 해시 값 비교를 통해 데이터 블록간의 빠른 비교 가능

- Encryption
  - 접근 불가능한 데이터를 생성하기 위함

해시 함수

해시 함수의 문제점

- Recognizability(인식 가능성)
  - 메시지가 같으면 같은 해시 값이 도출됨
  - 다른 사용자와 패스워드가 같을 경우 유출 가능
  - 레인보우 테이블 공격에 취약함



- Speed
  - 해시 함수는 처리 속도가 빠름
  - 브루트 포스 공격에 취약

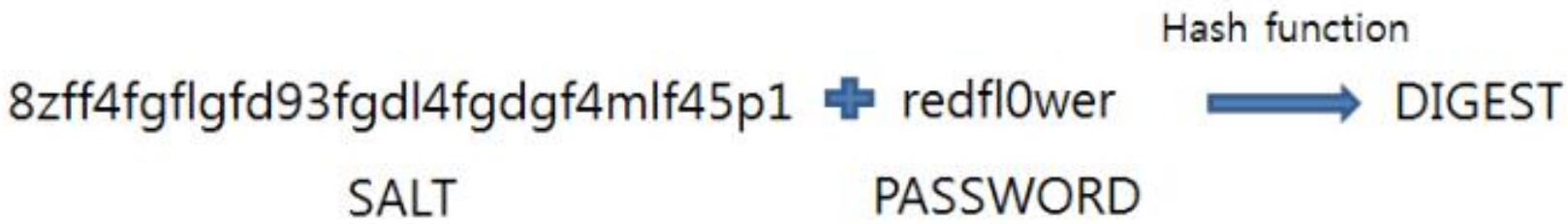
패스워드	해시
123456	BFE4D570E9301...

[Rainbow Table]

해시 함수

해시 함수의 문제점 해결

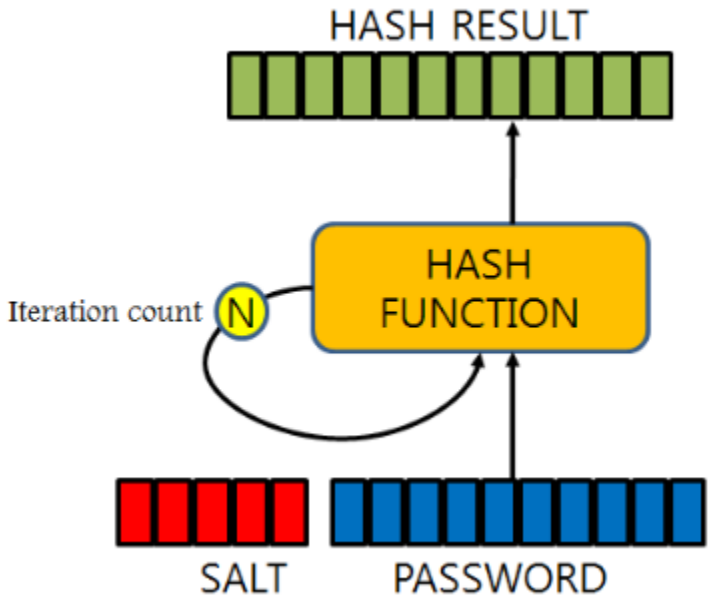
- Salting
  - Salt를 이용해 레인보우 테이블 공격을 회피
  - 패스워드에 salt를 추가하여 Digest 생성
  - 사용자마다 다른 salt를 사용하면 패스워드가 같더라도 Digest는 다르게 생성
  - Salt는 최소 128비트가 넘어야 안전



해시 함수

해시 함수의 문제점 해결

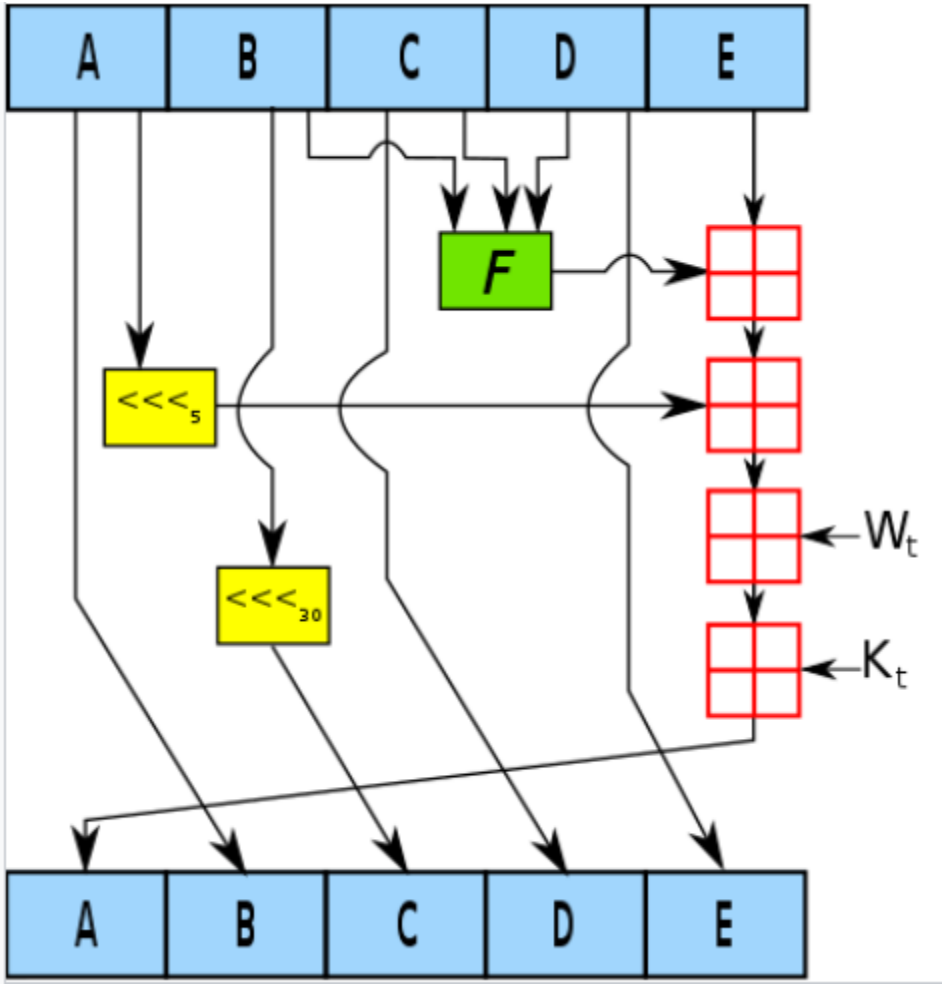
- Key Stretching
  - 해시를 여러 번 반복하여 시간을 늘림
  - 브루트 포스 공격에 대비하기 위함
  - 다이제스트를 반복하여 생성하는 방식
  - 짧고 예측하기 쉬운 패스워드를 추측하기 어렵게 만듦



SHA-1

SHA(Secure Hash Algorithm)

- SHA 함수들 중 하나, 1995년 발표
- 최대 2^64 bit의 메시지 입력
- 160 bit의 해시 값 출력
- 512 bit 입력 블록 단위로 패딩 처리
- 서명문 생성을 위한 알고리즘



Q & A