2010-1학기 현대암호학

제 II부 인증

박 종 혁

Tel: 970-6702
Email: jhpark1@snut.ac.kr

제 7장 일방향 해시 함수

목차

- 일방향 해시 함수
- 일방향 해시 함수의 응용 예
- 일방향 해시 함수의 예
- 일방향 해시 함수 SHA-1 (512)
- 일방향 해시 함수로 해결할 수 없는 문제

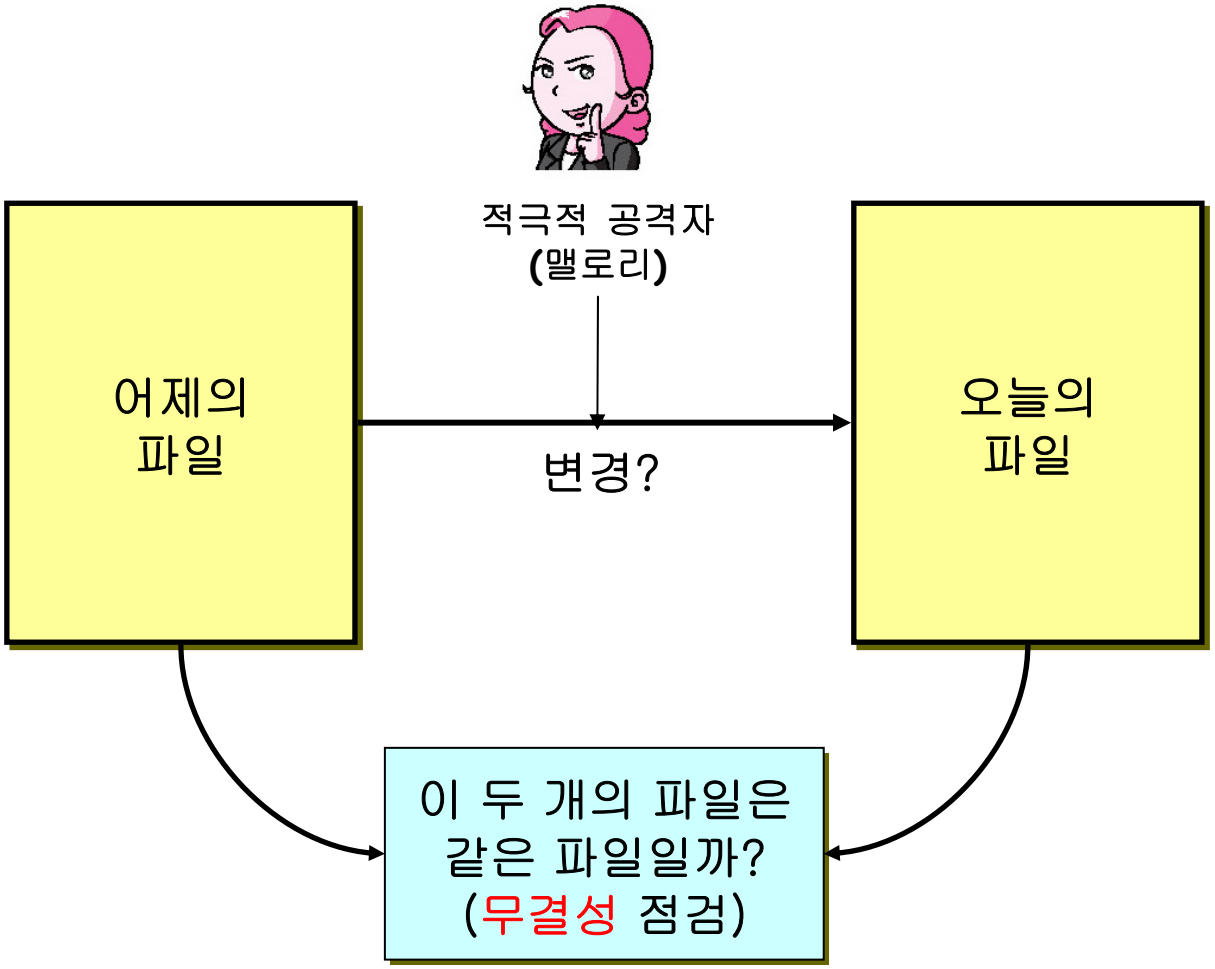
7.0 주요 내용

- 범죄 수사에서는 지문을 이용하는 일이 있다.
- 특정인의 지문과 현장에 남겨진 지문을 비교해서 그 사람이 사건에 관련되어 있는지 조사한다.
- 컴퓨터로 처리하는 메시지에 대해서도 「지문」이 있었으면 할 때가 있다.
- 2개의 메시지가 동일한지 아닌지를 조사할 때 메시지를 직접 비교하는 것이 아니라, 메시지의 지문을 비교하여 판정한다.

7.1 일방향 해시 함수

- 먼저 앨리스가 등장하는 스토리를 통해서 일방향 해시 함수가 필요해지는 장면을 소개하겠다.
- 그런 다음 일방향 해시 함수가 갖추어야 할 성질을 설명하겠다.

7.1.1 파일의 진위



파일 전체를 안전한 장소에 보존해 두고, 나중에 비교하는 방법

적극적 공격자
맬로리

넌센스 ?
- 안전한 장소에 보관 가능?
- 대용량 파일 ?

어제의 파일

변경?

오늘의 파일

안전한 장소에 보존해 둔다

어제 파일의 복사

안전한 장소

비교

무결성 확인

파일을 비교하는 대신에 해시 값을 비교하는 방법

적극적 공격자
맬로리

어제의
파일

변경?

오늘의
파일

일방향 해시함수

일방향 해시함수

안전한 장소에
보존해 둔다

해시값

비교

안전한 장소

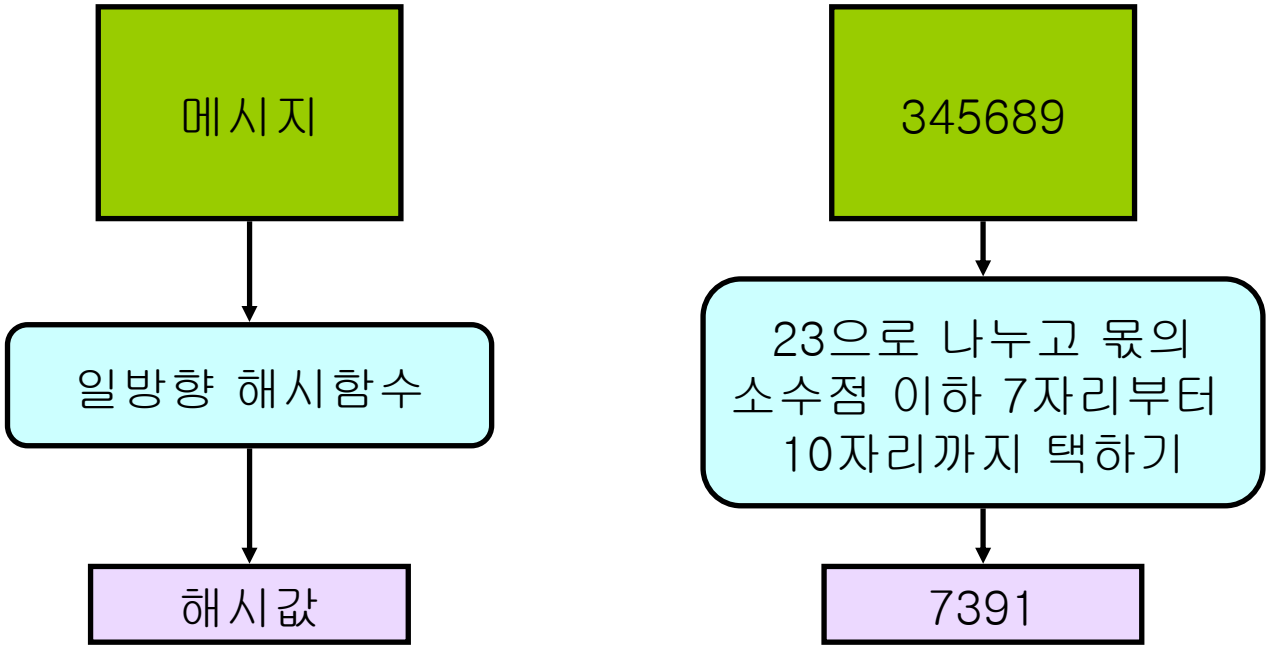
무결성
확인

7.1.2 일방향 해시 함수란

일방향 해시 함수

- 일방향 해시 함수(one-way hash function)
  - 입력과 출력이 각각 1개씩
    - 입력: 메시지(message)
    - 출력: 해시 값(hash value)이라 함

일방향 해시 함수는 메시지를 기초로 해서 해시 값을 계산한다



해시함수의 입력

- 해시함수에 입력되는 메시지는 인간이 읽을 수 있는 문서일 필요는 없다.
  - 화상 파일이라도, 음성 파일이라도 상관없다.
- 일방향 해시 함수는 메시지가 실제로 무엇을 나타내고 있는지를 알 필요는 없다.
- 일방향 해시 함수는
  - 어떤 메시지든지 단지 비트 열로서 취급
  - 그 비트 열을 기초로 해시 값을 계산

해시함수의 출력

- 해시 값의 길이는 메시지의 길이와는 관계가 없다.

- 메시지가 1비트라도, 1메가바이트라도, 100기가바이트라도 일방향 해시 함수는 고정된 길이의 해시 값을 출력으로 배출한다.
- 예를 들면
  - SHA-1이라는 일방향 해시 함수 → 해시 값은 항상 160비트(20바이트)

해시 값은 항상 고정 길이

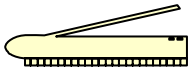
사용자 패스워드

8바이트

일방향 해시함수 (SHA-1)

43 B0 4C 54 3B
67 A2 23 3F 7D
36 2B 7A 2B 49
3C D3 AF 27 4A

해시값 20바이트



스캐너로부터의 영상 데이터

512 킬로바이트

일방향 해시함수 (SHA-1)

73 BF 4C 34 3B
67 A2 45 23 76
3F 76 D2 37 F6
44 47 8F 93 D2

해시값 20바이트

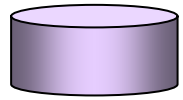
플로피 디스크의 모든 파일

1.4메가바이트

일방향 해시함수 (SHA-1)

54 3B 4C 34 3B
62 3C D3 AF A2
45 67 A2 23 3F
7D 43 B0 4C 19

해시값 20바이트



하드 디스크의 모든 파일

80기가바이트

일방향 해시함수 (SHA-1)

32 2B 23 70 7A
2B 4F 43 B0 4C
54 3B 49 28 67
A2 23 8F 7D 36

해시값 20바이트

7.1.3 일방향 해시 함수의 성질

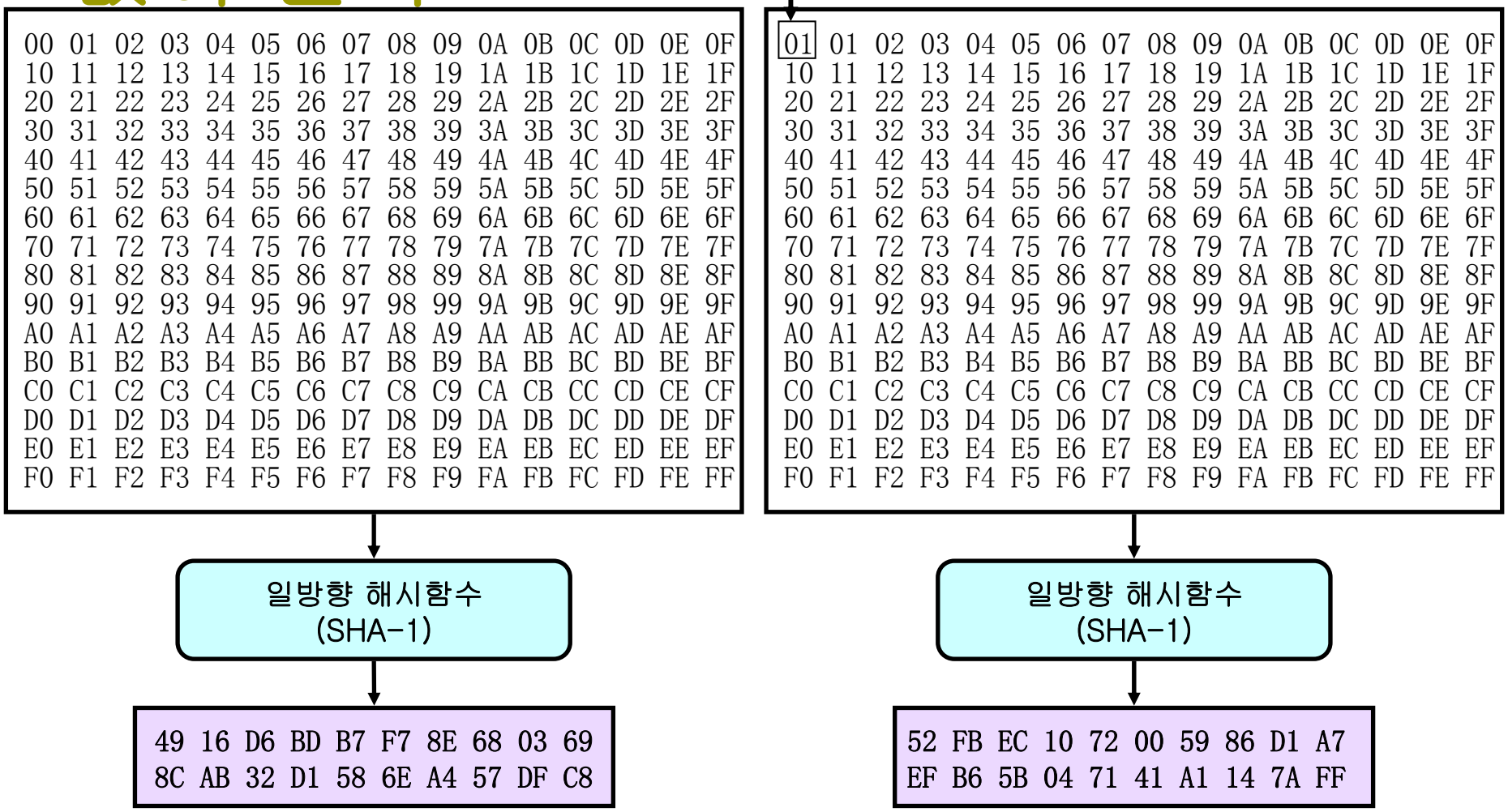
- 임의의 길이 메시지로부터 고정 길이의 해시 값을 계산한다
- 해시 값을 고속으로 계산할 수 있다
- 메시지가 다르면 해시 값도 다르다
- 일방향성을 갖는다

충돌(collision)

- 2개의 다른 메시지가 같은 해시 값을 갖는 것을 **충돌(collision)**이라고 한다.
- 일방향 해시 함수를 무결성 확인에 사용하기 위해서는 충돌이 발견되어서는 안 된다.

메시지가 1비트만 달라도 다른 해시 값이 된다

메시지의 00을 01로 바꿨다(1비트의 변경)



메시지가 1비트만 달라져도
전혀 다른 해시 값이 생성된다

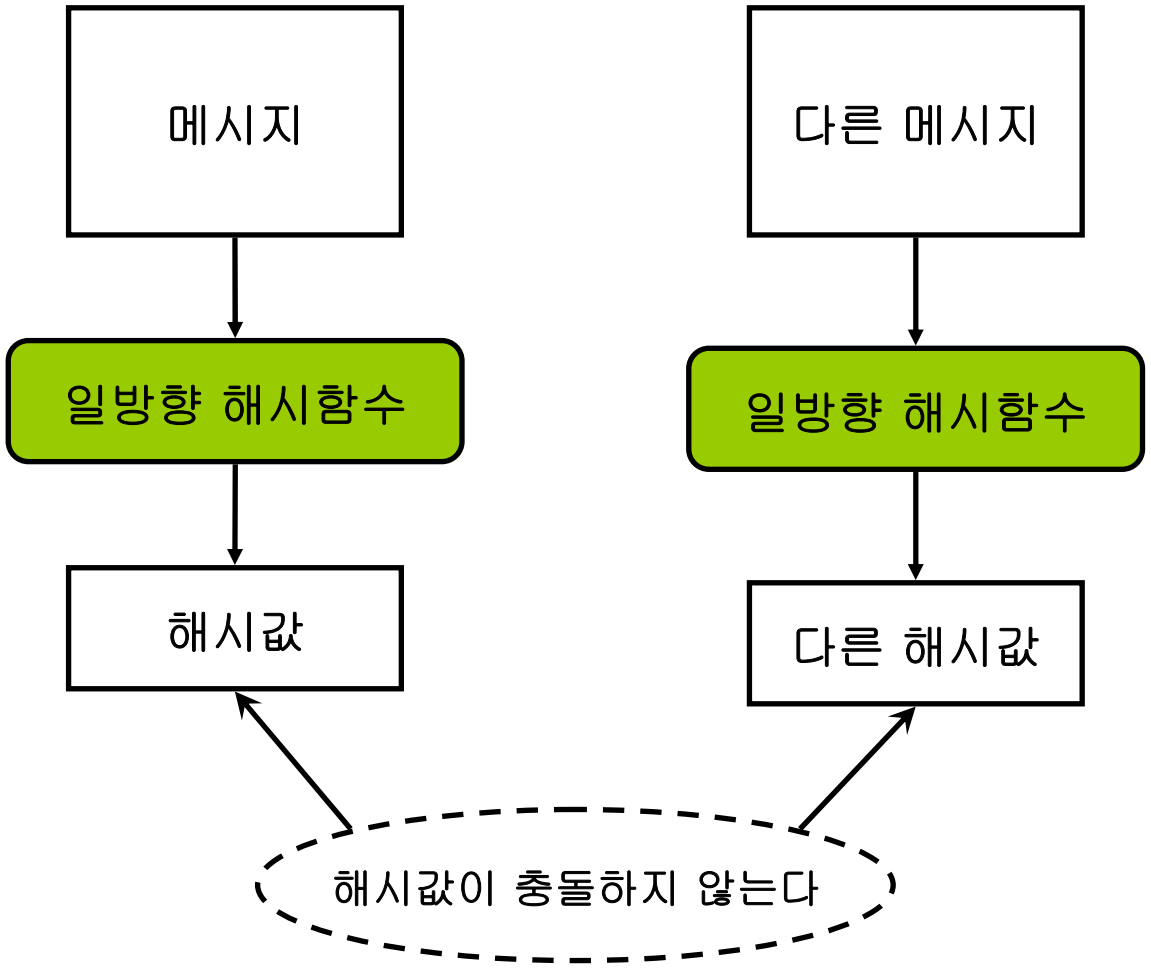
앞에서 예로 들었던 일방향 함수

- 위의 예제에서 우리가 입력으로 사용한 값은 A = 345689 이었고 출력으로 얻은 출력 값은 7391 이었다.
- 이제 동일한 일방향 함수에 새로운 입력으로 B = 232.8395049993 을 넣어보자.
- 그러면 몫은 10.1234567391 가 된다. 따라서 출력되는 값은 73910이 된다.
- 이것을 보면 두 개의 서로 다른 입력 A와 B에 대해서 동일한 출력 값을 배출한다는 것을 알 수 있다.

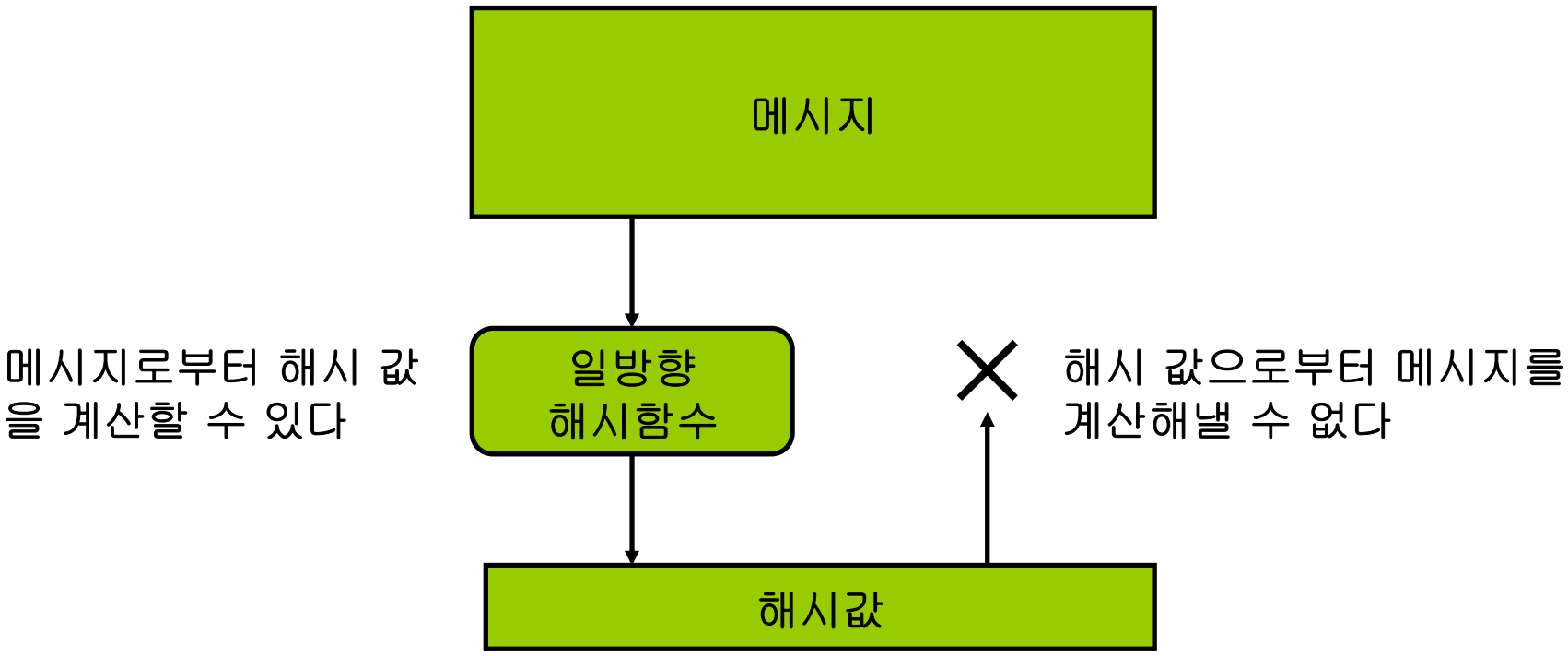
충돌 내성(collision resistance)

- 충돌을 발견하는 것이 어려운 성질을 가리켜 충돌 내성(collision resistance)라고 부른다.
- 암호 기술에서 사용되는 일방향 해시 함수는 충돌 내성을 가질 필요가 있다

일방향 해시 함수의 충돌 내성



일방향 해시 함수의 일방향성



7.1.4 해시 함수관련 용어

- 일방향 해시 함수는
  - 메시지 다이제스트 함수(message digest function)
  - 메시지 요약 함수
  - 암호적 해시 함수

등으로 불린다.

- 일방향 해시 함수의 입력이 되는 메시지는
  - 프리・이미지(pre-image)라고 불리는 일도 있다.

7.1.4 해시 함수관련 용어

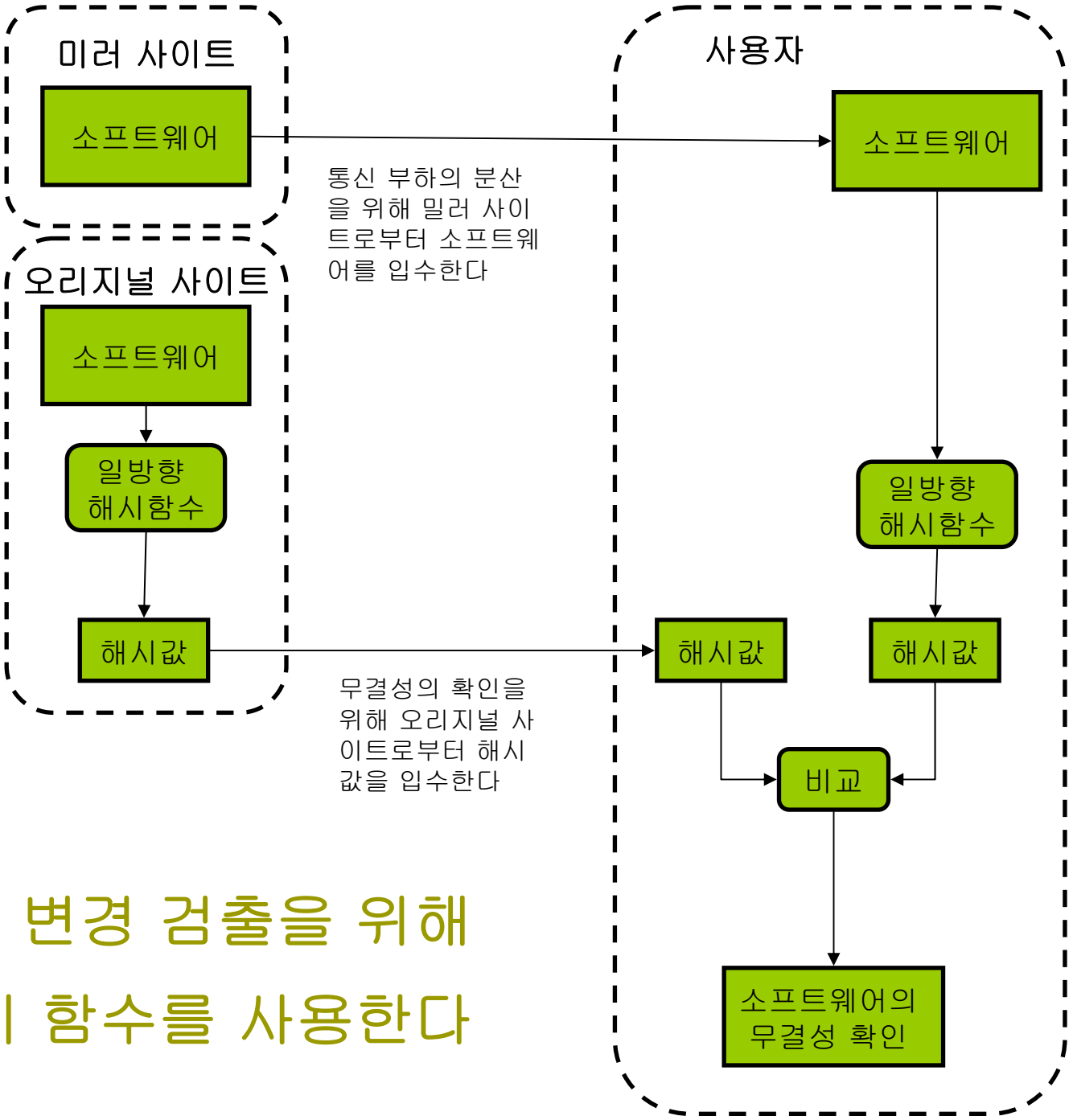
- 일방향 해시 함수의 출력이 되는 해시 값
  - 메시지 다이제스트(message digest)
  - 핑거프린트(fingerprint)

- 무결성
  - 완전성
  - 보전성

7.2 일방향 해시 함수의 응용 예

7.2.1 소프트웨어의 변경 검출

- 어떤 사람이 특정 웹 사이트에서 소프트웨어를 다운로드 받는다고 했을 때
  - 자신이 다운받아 입수한 소프트웨어가 원래 웹 사이트 주인이 올려놓은 소프트웨어와 동일한 것인지
  - 어떤 공격자나 악의를 가진 사람에 의해서 수정된 내용의 소프트웨어인지 확인하기 위해 일방향 해시 함수가 사용



소프트웨어 변경 검출을 위해
일방향 해시 함수를 사용한다

7.2.2 패스워드를 기초로 한 암호화

- 일방향 해시 함수는 패스워드를 기초로 한 암호화(password based encryption; PBE)에서 사용되는 일이 있다.
- PBE에서는 패스워드와 솔트(의사난수 생성기로 생성한 랜덤 값,salt)를 섞은 결과의 해시 값을 구해 그것을 암호화 키로 사용한다.
- 이 방법으로 패스워드에 대한 사전 공격을 막을 수 있다.

7.2.3 메시지 인증 코드

- 일방향 해시 함수를 사용해서 메시지 인증 코드를 구성할 수 있다.
- 메시지 인증 코드란 「송신자와 수신자만이 공유하고 있는 키」와 「메시지」를 혼합해서 그 해시값을 계산한 값이다.
- 메시지 인증 코드를 사용하여 통신 중의 오류나 수정 그리고 「거짓 행세」를 검출하는 것이 가능해 진다.

7.2.4 디지털 서명

- 디지털 서명에 일방향 해시 함수가 사용된다.
- 디지털 서명의 처리에는 시간이 많이 걸리기 때문에
  - 디지털 서명을 메시지 전체에 직접 행하는 일은 별로 없다.
- 일방향 해시 함수를 사용해서 메시지의 해시 값을 일단 구하고,
  - 그 해시 값에 대해 디지털 서명을 행한다.

7.2.5 의사난수 생성기

- 일방향 해시 함수를 사용해서 의사난수 생성기를 구성할 수 있다.
- 암호 기술에 이용하는 난수에서는 「과거의 난수 예로부터 미래의 난수 예를 예측하는 것은 사실상 불가능」이라는 성질이 필요해진다.
- 그 예측 불가능성을 보증하기 위해 일방향 해시 함수의 일방향성을 이용한다.

7.2.6 원타임 패스워드

- 일방향 해시 함수를 사용해서 **원타임 패스워드** (one-time password)를 구성할 수 있다.
- 원타임 패스워드는 정당한 클라이언트인지 아닌지를 서버가 인증할 때에 사용된다.
- 이 방식에서는, 일방향 해시 함수를 써서 통신 경로 상에 흐르는 패스워드가 1회 한정(one-time)이 되도록 고안되어 있다.
- 이 때문에 패스워드가 도청되어도 악용될 위험성이 없다.

7.3 일방향 해시 함수의 예

표 7-1 일방향 해시 함수

알고리즘	출력 해시 값 크기	블록 크기	충돌성
HAVAL	256/224/192/160/128	1024	있음
MD2	128	128	있음
MD4	128	512	있음
MD5	128	512	있음
PANAMA	256	256	있음
RIPEMD	128	512	있음
RIPEMD-128/256	128/256	512	없음
RIPEMD-160/320	160/320	512	없음
SHA-0	160	512	있음
SHA-1	160	512	있음
SHA-256/224	256/224	512	없음
SHA-512/384	512/384	1024	없음
Tiger(2)-192/160/128	192/160/128	512	없음
VEST-4/8 (hash mode)	160/256	8	없음
VEST-16/32 (hash mode)	320/512	8	없음
WHIRLPOOL	512	512	없음

7.3.1 MD4와 MD5

- MD: 메시지 다이제스트(Message Digest)의 약자

- MD4
  - Rivest가 1990년에 만든 일방향 해시 함수로 128비트의 해시 값을 갖는다
  - 그러나 Dobbertin에 의해 MD4의 해시 값의 충돌을 발견하는 방법이 고안되어 현재는 안전하다고 할 수 없다.

7.3.1 MD4와 MD5

- MD5
  - MD4를 만든 Rivest가 1991년에 만든 일방향 해시 함수로 128비트의 해시 값을 갖는다
  - 여기서 입력은 512-비트 블록들로 처리된다.
  - 전수공격과 암호해독에 대한 우려가 심각해진 최근 몇 년을 제외하면 MD5 는 가장 널리 사용되던 안전한 해시 함수이었다. (2005년 깨졌으나, 사용은 되고 있음)

7.3.2 SHA-1, SHA-256, SHA-384, SHA-512

- SHA(Secure Hash Algorithm)은 NIST(National Institute of Standards and Technology)에서 만들어진, 160비트의 해시 값을 갖는 일방향 해시 함수이다.
- 1993년에 미국의 연방정보처리표준규격(FIPS PUB 180)으로서 발표된 것을 SHA라 부른다
- 1995년에 발표된 개정판 FIPS PUB 180-1을 SHA-1이라 부른다.
- SHA-1의 메시지의 길이에는 상한이 있지만, 264비트 미만이라는 대단히 큰 값이므로 사실상 현실적인 적용에는 문제가 없다.

SHA-256, SHA-384, SHA-512

- 2005년에 NIST에서는 SHA-1에 대한 승인을 취소한다고 선언했고 2010년까지 새로운 SHA 버전들로 대체한다고 했다.
- 하지만 이미 NIST에서는 2002년에 NIST는 새 표준인 FIPS 180-2 를 내놓았는데 이때 해시 값이 각각 256, 384와 512 비트인 3 개의 새로운 SHA 버전들을 정의했다.
- 이들은 각각 SHA-256, SHA-384와 SHA-512 이다.

SHA 매개변수 비교 (단위: 비트)

표 7-2 SHA 매개변수 비교 (단위: 비트)

	SHA-1	SHA-256	SHA-384	SHA-512
메시지 다이제스트 길이	160	256	384	512
메지시 길이	$< 2^{64}$	$< 2^{64}$	$< 2^{128}$	$< 2^{128}$
블록 길이	512	512	1024	1024
단어 길이	32	32	64	64
단계 수	80	64	80	80
메시지 다이제스트 길이에 대해 생일공격 시 한 번의 충돌이 나타나는데 필요한 횟수	80	128	192	256

7.3.3 RIPEMD-160

- RIPEMD-160 은 1996 년 에 Hans Dobbertin, Antoon Bosselaers, Bart Preneel에 의해 만들어진, 160비트의 해시 값을 갖는 일방향 해시 함수이다.
- RIPEMD-160은 European Union RIPE 프로젝트로 만들어진 RIPEMD라는 일방향 해시 함수의 개정판이 된다.

7.4 일방향 해시 함수 SHA-1

- 대표적인 일방향 해시

7.4.1 전체의 흐름

- SHA-1
  - 2^{64}비트 미만의 메시지를 기초로 해서 160비트의 해시 값을 계산하는 일방향 해시 함수

160비트의 값을 계산하는 순서

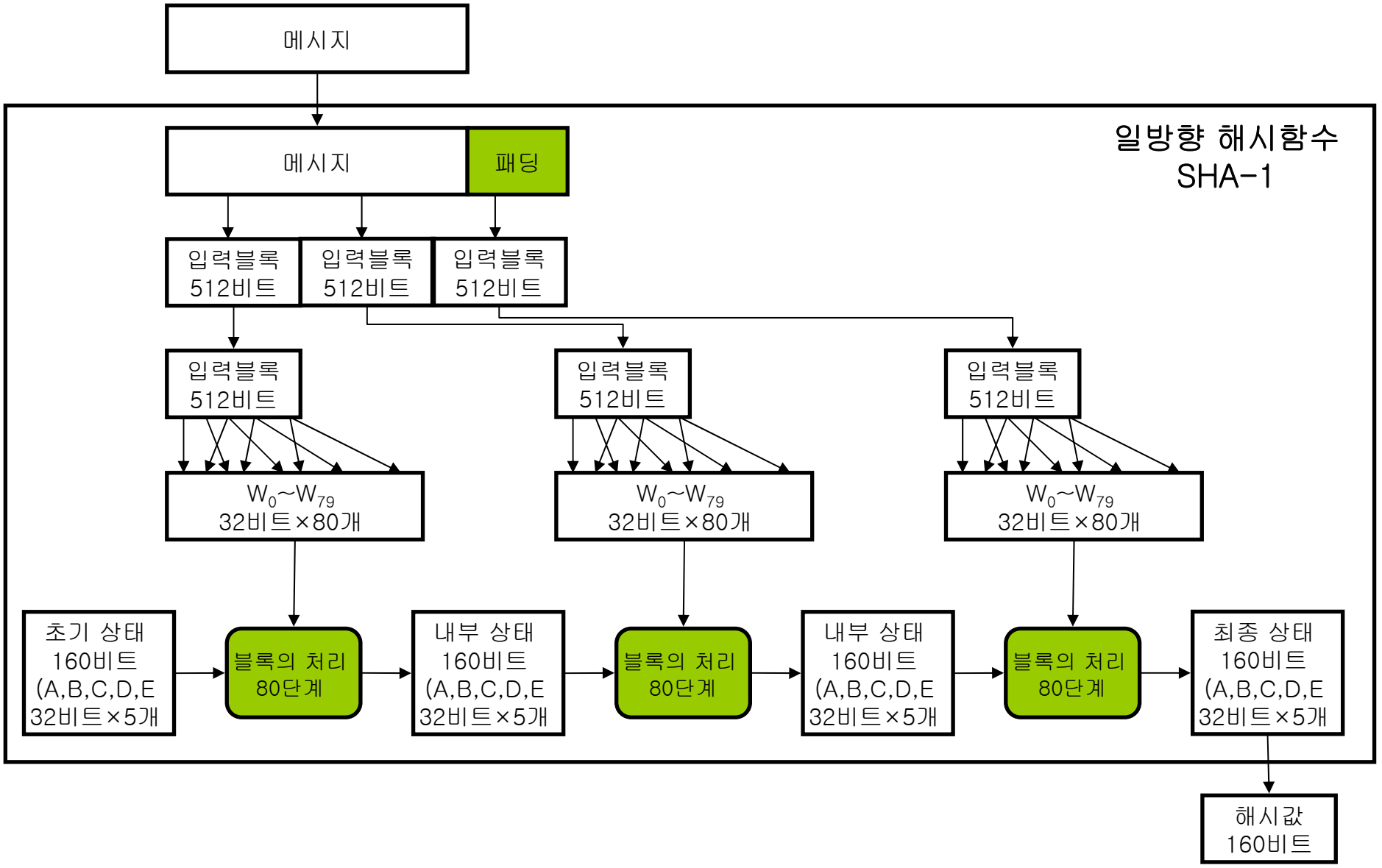
(1) 패딩

(2) W_0~W_{79} 계산

(3) 블록 처리

(4) 단계 1 처리

일방향 해시 함수 SHA-1의 개요



SHA-1 : 패딩

- 0비트 이상 2^{64}비트 미만의 길이를 갖는다.
- SHA-1에서는 이후의 처리를 하기 쉽게 하기 위해 맨 처음에 **패딩**(padding)을 행한다.
- 패딩이라는 것은 「메워 넣기」라는 의미이다.
- 여기서는 메시지 다음에 여분의 데이터를 부가하여 메시지의 길이가 512비트의 정수배가 되도록 하는 것을 가리킨다.
- 이 512비트의 집합을 **입력 블록**이라 부른다.

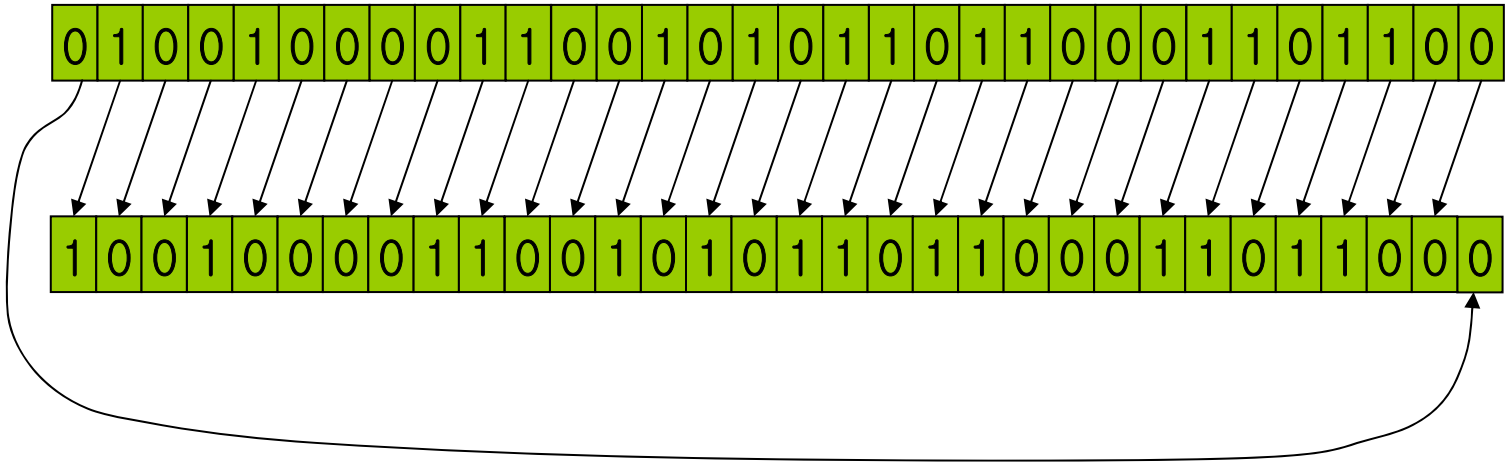
SHA-1 : W_0~W_{79}의 계산

- 패딩이 끝난 다음에는 입력 블록 단위의 처리가 된다.
- 입력 블록 512비트마다 32비트× 80개의 값 (W_0~W_{79})을 계산한다. 이 80개의 값은 「(4) SHA-1 : 단계 1 처리」에서 사용한다.
- 우선 입력 블록 512비트를 32비트× 16개로 분할하여 W_0~W_{15}처럼 이름을 붙여 간다.
- 그리고 W_{16}부터 W_{79}는 아래와 같이 계산한다.

W_{16}계산과 일반 W_t 의 계산

- $W_{16} = (W_0 \oplus W_2 \oplus W_8 \oplus W_{13})$을 1비트 회전
- $W_t = (W_{t-16} \oplus W_{t-14} \oplus W_{t-8} \oplus W_{t-3})$을 1비트 회전

01001000 01100101 01101100 01101100를 1비트 회전한 모양



SHA-1 :블록의 처리

- 입력 블록에 대해 80 단계씩의 처리를 행한다(그림 7-13 참조).
- 입력 블록의 정보를 기초로 내부 상태(160비트)를 변화시킨다.
- 이것을 모든 블록에 대해 행한다.
- 내부 상태 160비트는 A, B, C, D, E라는 이름이 붙은 32비트× 5개의 **버퍼**로 표현되어 있다.

입력 블록 512비트로
부터 80개의 32비트로

부터(W_0~W_{79})을 생성

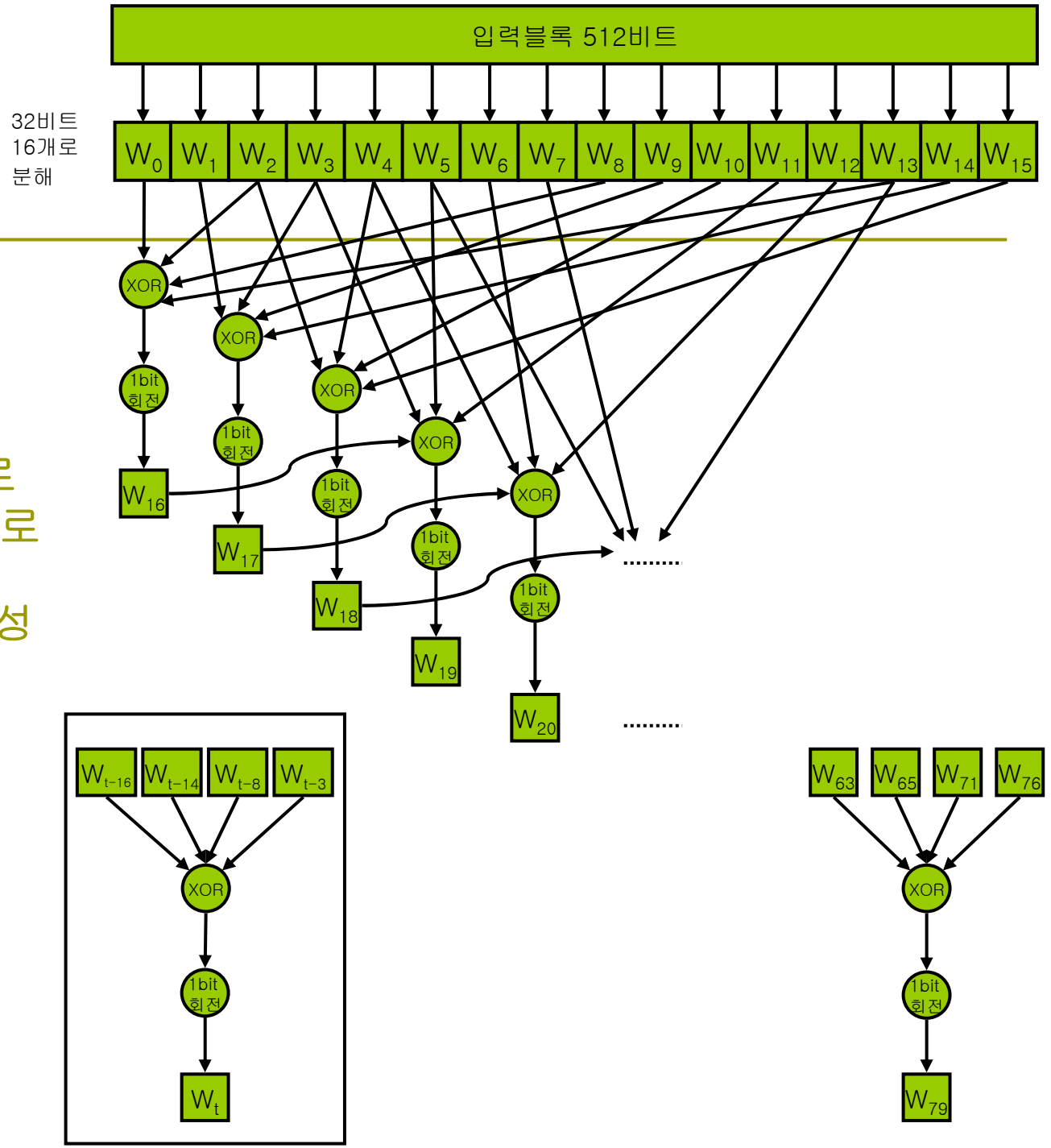
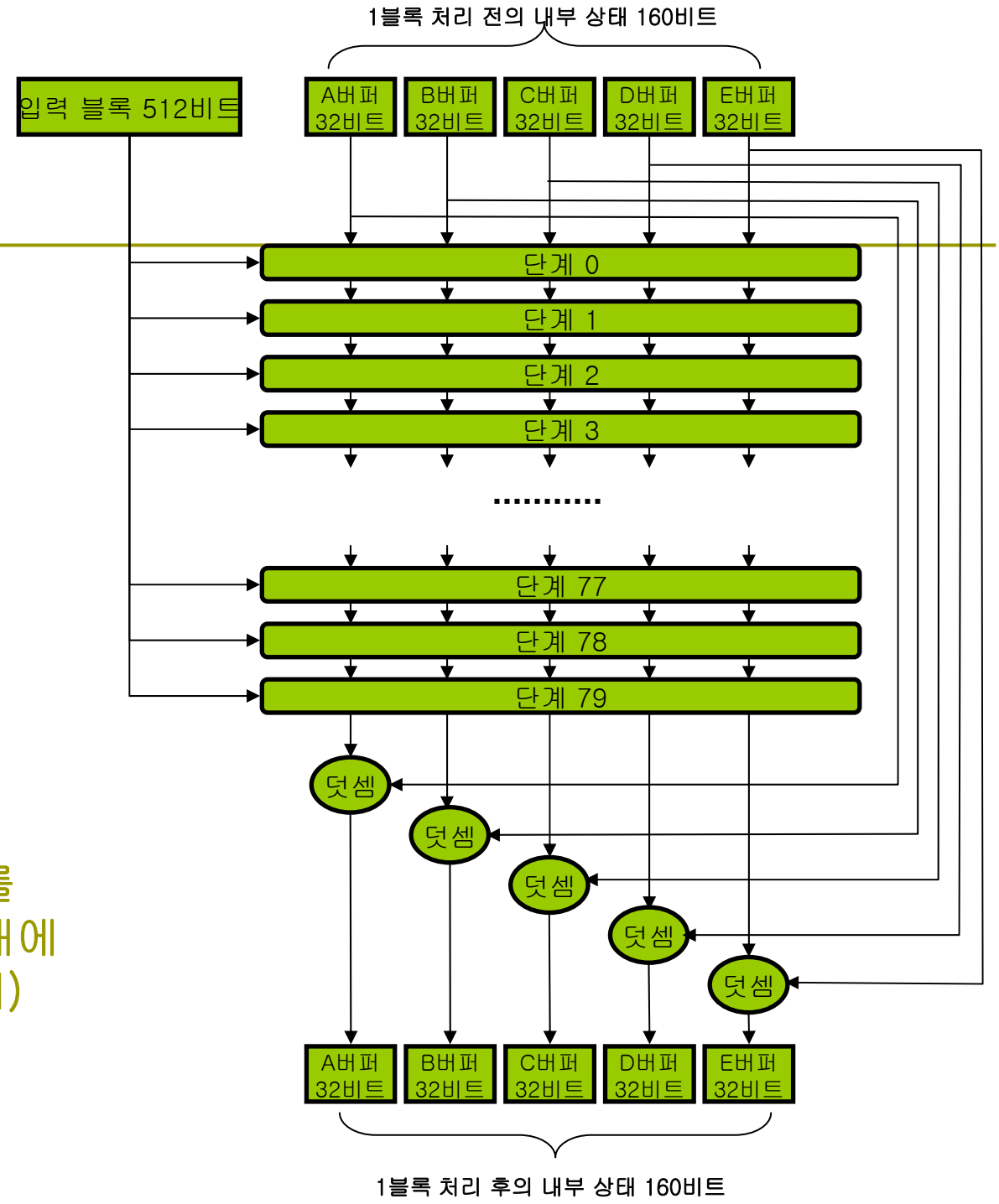


그림 7-12



입력 블록 512비트를
160비트의 내부 상태에
섞어 넣는다(80 단계)

그림 7-13

SHA-1 : 1 단계 처리

- 「(3) SHA-1 : 블록 처리」에 등장한 1 단계의 내용은 그림 7-14에 나타난 처리가 된다.
- W_0~W_{79}를 기초로, 내부 상태(즉 A, B, C, D, E의 값)를 변화시켜 가는 복잡한 처리이다.
- A~E의 버퍼의 초기값은 그림 7-14 에 나타나 있다.

A버퍼의 초기값 67 45 23 01 B버퍼의 초기값 EF CD AB 89 C버퍼의 초기값 98 BA DC FE D버퍼의 초기값 10 32 54 76 E버퍼의 초기값 C3 D2 E1 F0

1 단계 처리

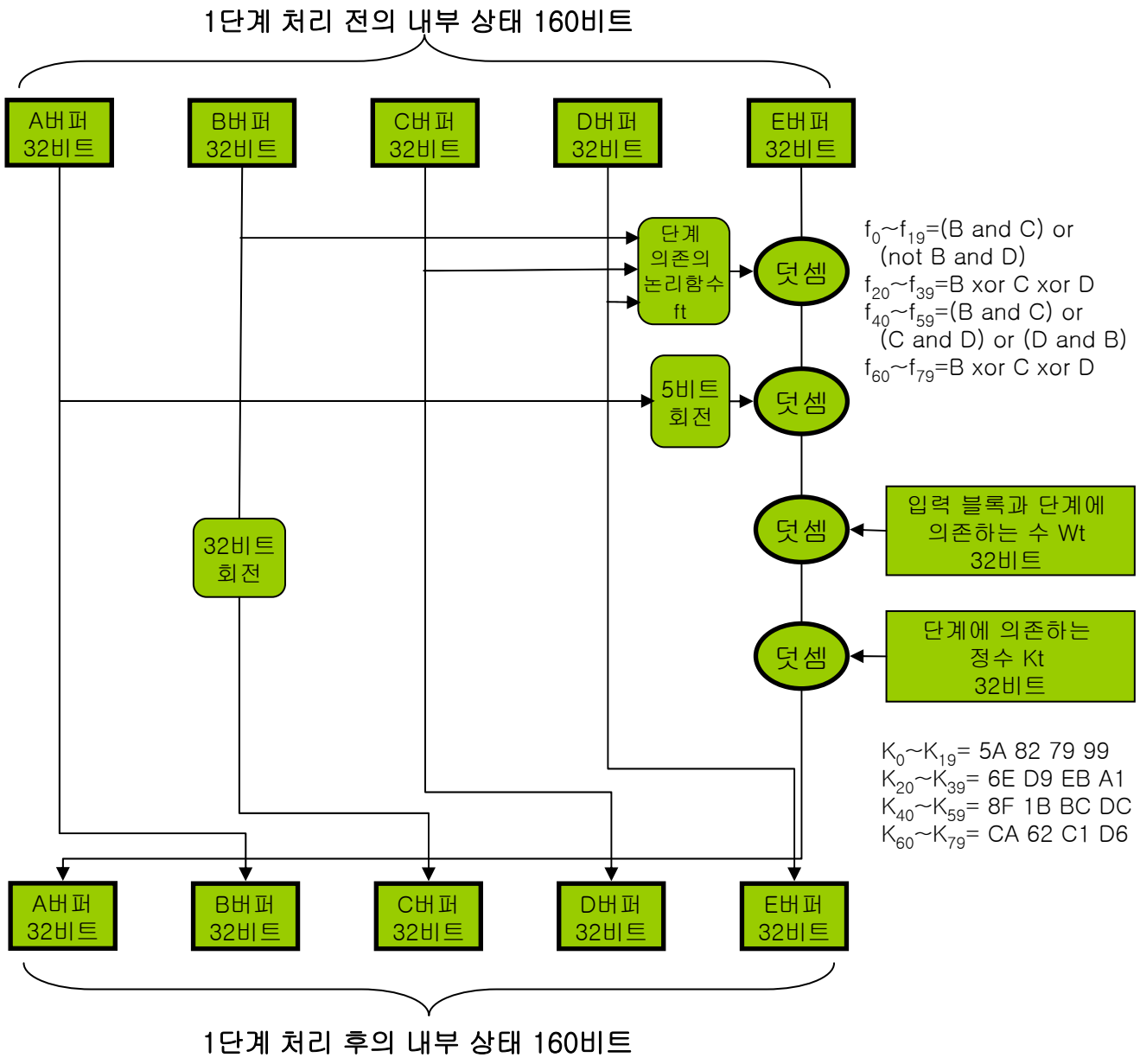


그림 7-14

7.5 일방향 해시 함수 SHA-512

- SHA의 다른 버전들도 구조는 거의 비슷하다
- SHA-512 알고리즘에서는 최대 길이가 2128 비트 이하인 메시지를 입력으로 사용하고 길이가 512 비트인 메시지 다이제스트를 출력한다.
- 입력 데이터는 길이가 1024 비트인 블록으로 나누어서 처리된다.

7.7 일방향 해시 함수로 해결할 수 없는 문제

- 일방향 해시 함수는 「수정 또는 변경」을 검출할 수 있지만, 「거짓 행세」를 검출하는 것은 못 한다.
- 이를 위해서는 무결성 외에 **인증**이라는 절차가 필요해진다.
- 인증을 수행하기 위한 기술이 **메시지 인증 코드**와 **디지털 서명**이다.