

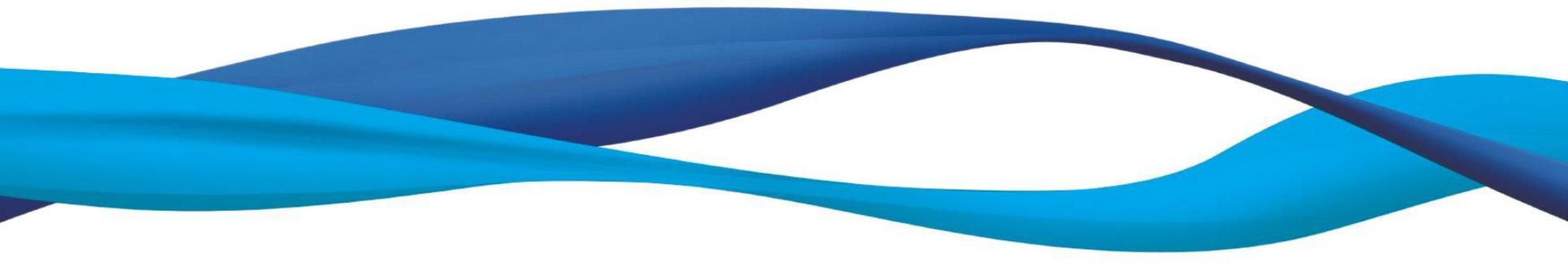
9. 무결성 유지 기술

박 종 혁 교수

UCS Lab

Tel: 970-6702

Email: jhpark1@snut.ac.kr



- **학습 목표**

- 디지털 데이터는 위변조가 쉬우므로 디지털 증거로 사용하기 위해서는 위변조를 입증할 수단이 필요하다. 이런 무결성 유지를 위해 필요한 수단인 전자 서명과 전자 서명의 요소인 해시함수를 학습한다.

- **학습 내용**

- 해시 함수
- 전자 서명
- 시점 확인 서비스
- 디지털 증거의 인증

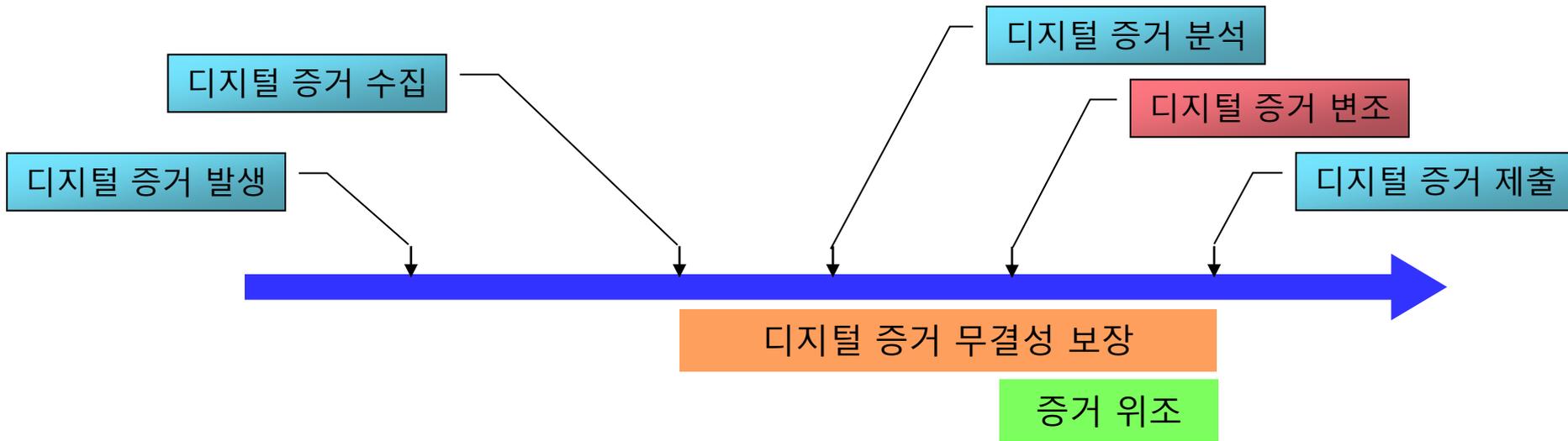
목 차

1. 해쉬 함수
2. 전자 서명
3. 시점 확인 서비스
4. 디지털 증거의 인증

현재 디지털 증거 무결성 확보의 문제점

• 디지털 증거의 신뢰성

- 디지털 증거가 위조되어 법정에 제출되는 경우
- 디지털 증거가 위조되지 않았음에도 불구하고, 용의자 또는 피고소인이 디지털 증거가 위조되었을 가능성을 이유로, 증거 효력을 무력화시키려 하는 경우



해쉬함수(2/10)

응용

전용 해쉬 함수의 응용분야

◆ 데이터
무결성 검사



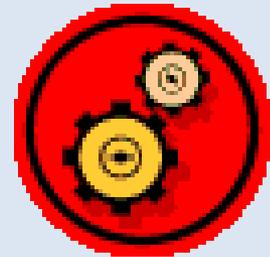
◆ 전자서명



◆ 인증



◆ 난수 생성기



해쉬함수(3/10)

필요성

암호학적 해쉬함수는 왜 필요한가?

무결성

악의적인 사용자에게 의한 데이터 위변조 확인

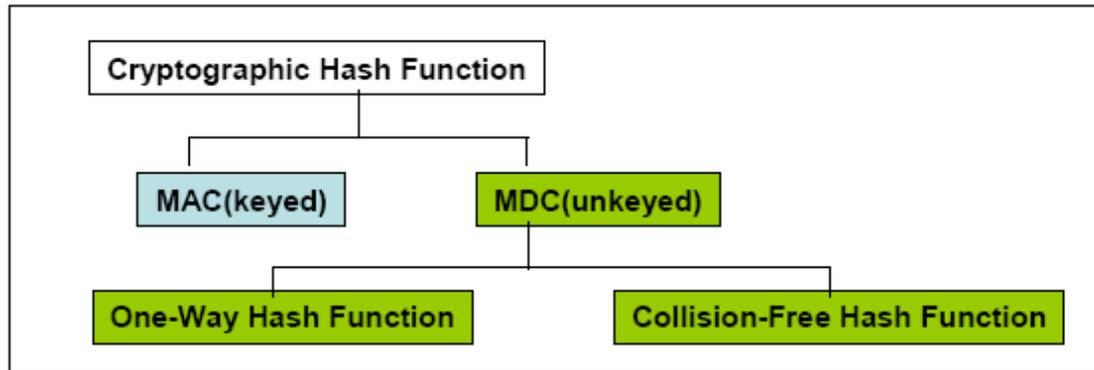
효율성

데이터의 거대화 방지
전자서명시 연산의 효율성을 증대시킴

해쉬함수 (4/10)

분류

키 사용 유무에 따른 해쉬함수의 분류



m \rightarrow $\text{MAC} = h(k, m)$

m \rightarrow $\text{MDC} = h(m)$

해쉬함수 (5/10)

성질

안전성 측면에서 고려되어야 할 중요한 성질

역상 저항성 (Preimage Resistance)

y 가 주어지고, $h(x)=y$ 인 x 를 찾는 것이 어려움

제 2 역상 저항성 (2nd-Preimage Resistance)

$h(x)=y$ 인 x 와 y 가 주어지고, $h(x')=y$ 인 $x' (\neq x)$ 를 찾는 것이 어려움

충돌 저항성 (Collision Resistance)

$h(x)=h(x')$ 인 $x, x' (\neq x)$ 를 찾는 것이 어려움

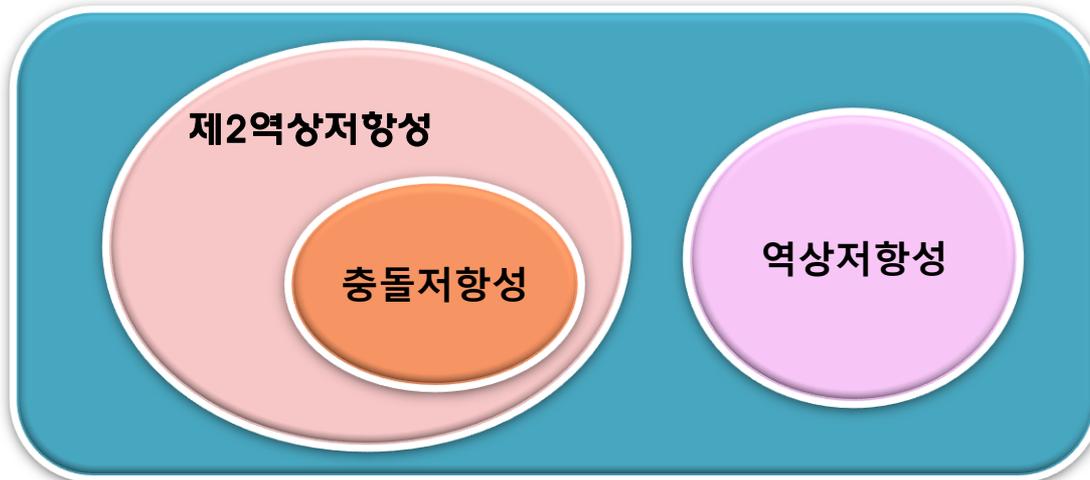
해쉬함수(6/10)

특 성

해쉬함수의 안전성 성질들 간의 관계



- 역상저항성 ! \rightarrow 제 2역상저항성
- 제2역상저항성 ! \rightarrow 역상저항성
- 충돌저항성 \rightarrow 제2역상저항성



해쉬함수(7/10)

생일역설

생일 역설(Birthday paradox)과 해쉬함수의 안전성

생일 역설

가능한 데이터의 개수가 2^n 일 때, $2^{n/2}$ 정도의 데이터가 있으면 일치하는 데이터가 존재할 확률이 $1/2$ 보다 크다.



- 어느 집단에서 생일이 같은 한 쌍이상의 학생이 존재할 확률(p)이 0.5이상이 되기 위한 학생수(r) ?
- $p = 1 - 1(1-1/n)(1-2/n) \dots (1-\{r-1\}/n)$
- When $n=365$ and $r \geq 23$, $p \geq 0.5$

해쉬함수(8/10)

생일역설

생일 역설(Birthday paradox)과 해쉬함수의 안전성

해쉬함수의 안전성

해쉬함수의 출력길이가 n 비트이면, 생일 역설에 의해 그 해쉬함수의 안전성은 $2^{n/2}$ 를 넘을 수 없다.



- MD4, MD5의 경우 : 해쉬함수의 출력 길이가 128비트이므로 해쉬함수의 안전성은 2^{64}
- HAS-160, SHA-1의 경우 : 해쉬함수의 출력 길이가 160 비트이므로 해쉬함수의 안전성은 2^{80}
- SHA-256 의 경우 : 해쉬함수의 출력 길이가 256비트이므로 해쉬함수의 안전성은 2^{128}

해쉬함수(9/10)

MD 구성방법

1989년 CRYPTO에서 Merkle과 Damgard에 의해 제안됨

설계논리

고정된 입출력 크기를 갖는 함수를 이용하여 임의의 길이의 입력값을 다루는 해쉬 함수를 구성하는 방법

압축함수(f)

고정된 입출력 크기를 갖는 함수 : 압축함수(Compression function)



해쉬함수(10/10)

압축함수 설계논리

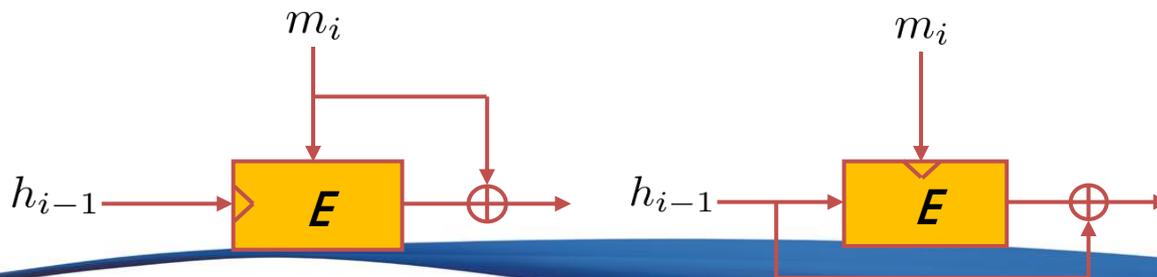
해쉬함수의 기반이 되는 압축함수의 설계논리에 따른 분류

전용 해쉬함수

MD4, MD5, SHA-1, SHA-2, HAS-160 등과 같이 순수하게 해쉬함수로서 이용되기 위하여 설계됨

블록암호(E)기반 해쉬함수

블록암호를 기반으로 역상저항성을 만족하도록 변형하여 설계됨(MDC-2, MDC-4, PGV 모델)



해쉬 함수의 안전성

- 해쉬 함수의 충돌 쌍을 효과적으로 찾을 수 있는 공격법이 개발
 - 표준 해쉬 함수를 만들기 위한 프로젝트가 미국 NIST에서 진행
- MD5에서 충돌 쌍이 발견되어 안전성에 대한 문제가 제기
 - 문제 원인 : 제 2 역상을 발견할 수 있어야 하는데, 충돌 쌍을 발견하는 것과 제 2 역상을 발견하는 것은 근본적인 차이가 있으며, 제 2 역상을 효과적으로 찾는 방법은 알려져 있지 않음
 - 무결성 유지를 위한 MD5 해쉬 함수의 사용 문제 없음
- 향후 SHA-3를 사용하는 것이 바람직함
 - 새롭게 표준 해쉬 함수 SHA-3를 개발하고 있으며, 안전도에 대한 기준이 80비트에서 128비트 변화

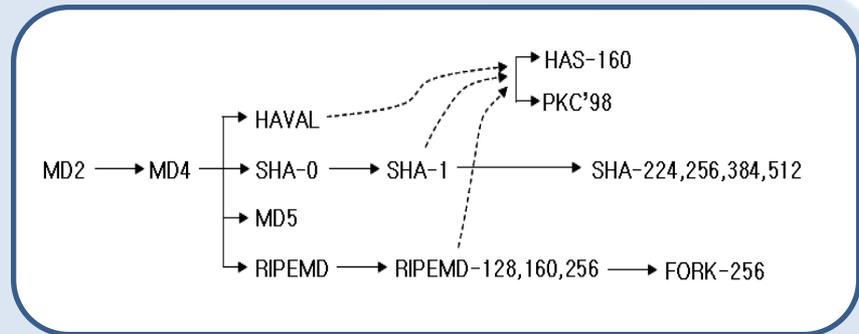
해쉬함수 설계 및 분석동향 (1/4)

MD 계열

전용 해쉬함수 설계 동향



- ◆ 1989 MD2
- ◆ 1990 MD4, RIPEMD
- ◆ 1991 SHA-0
- ◆ 1992 MD5, HAVAL
- ◆ 1993 SHA-1
- ◆ 1996 RIPEMD-128,160, Tiger
- ◆ 2000 HAS-160, Whirlpool, SHA-256,384,512
- ◆ 2002 SHA-224



해시함수 설계 및 분석동향 (2/4)

분석동향 전용 해시 함수의 분석 사례



- ◆ 2004 : MD4, MD5, HAVAL, RIPEMD 분석
- ◆ 2005 : SHA-0, SHA-1 분석



해시함수 응용한 경에 대한 공격

- MD4와 MD5 기반 HMAC/NMAC에 대한 키 복구
- MD4와 MD5 기반 APOP 프로토콜에서의 키 복구 공격

But

- 충돌쌍 공격은 발견되었지만, 실제 응용환경에 적용 가능한 제 2 역상 공격은 발견되지 않았다.

해쉬함수 설계 및 분석동향 (3/4)

표준화

NIST 해쉬함수 표준 공모



- ◆ 다양한 H/W & S/W 환경에서 구현 가능 (8, 32, 64 비트 플랫폼)
- ◆ 256 비트 이상의 메시지 입력
- ◆ 224, 256, 384, 512 비트 해쉬 값 생성



- ◆ 해쉬함수 후보 제출 마감 : 2008년 10월 31일
- ◆ 제 1 차 후보 선정 : 2008년 12월 9일
(기본 요구사항을 만족하는 알고리즘 선정)
- ◆ 제 2 차 후보 선정 : 2009년 7월 24일 (안전성 분석을 통한 알고리즘 선정)
- ◆ 제 3 차 후보 선정 : 2010년 8월 예정
- ◆ 최종 알고리즘 선정 : 2012년 2월 예정

해쉬함수 설계 및 분석동향 (4/4)

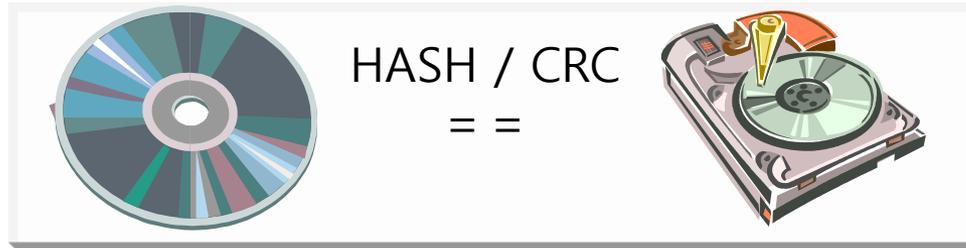
수명

HAS-160 / SHA-1 사용기간

- 
- ◆ SHA-1은 2010년 이후 충돌저항성이 만족되어야 하는 모든 응용분야에서 사용을 금해야 한다.
 - 전자서명 / 디지털 타임 스탬프 등등
 - 2010년 이후부터는 SHA-2를 이용하여야만 한다.
 - SHA-1은 2010년 이후 HMACs / KDFs / RNGs에서만 이용될 수 있음
- 
- ◆ HAS-160은 현재까지 전체 라운드에 대한 충돌쌍 공격이 알려져 있지 않음
 - SHA-1에 비하여 메시지 확산과정이 좋음
 - 하지만 2010년 이후에는 SHA-2와 같이 해쉬값의 길이가 256비트인 국내 표준 해쉬함수의 개발이 요구됨

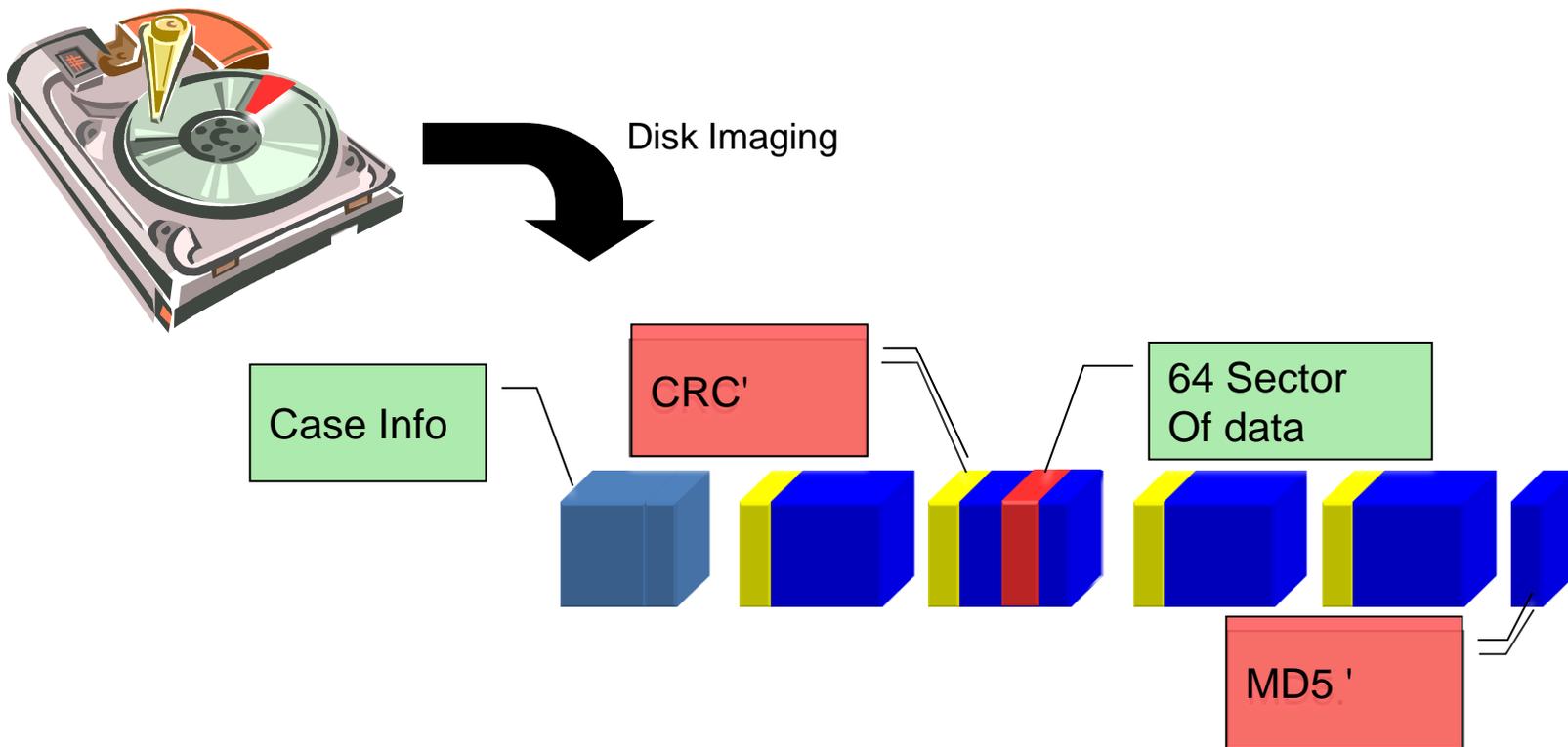
해시함수를 이용한 증거 워변조 방지

- 비트 스트림 복제(Bit Stream Clone) 방식으로 저장매체를 전체 복사하여 하드 디스크 드라이브 이미지를 생성한 후, 해시함수를 적용
- 해시 및 오류 검증 알고리즘을 저장매체와 이미지에 적용
 - 해시 알고리즘 (Hash Algorithm)의 특성
 - 원본 데이터를 1-bit만 바꿔도 해시 함수의 결과 값은 전혀 다른 출력 값을 생성하기 때문에, 증거 무결성에 활용되고 있음
 - 오류 검증 알고리즘의 특성
 - CRC (Cyclic Redundancy Check) : 전송 데이터 내에 에러가 있는지를 확인하기 위한 방법 중의 하나
 - 검증값이 불일치하면 데이터에 오류가 존재함



해시 및 오류 검증 알고리즘을 원본 Disk와 Disk Image에 적용하여 보관한 뒤, 법정 증거 제출 시 무결성을 주장

EnCase의 디지털 증거 무결성 확보 방법

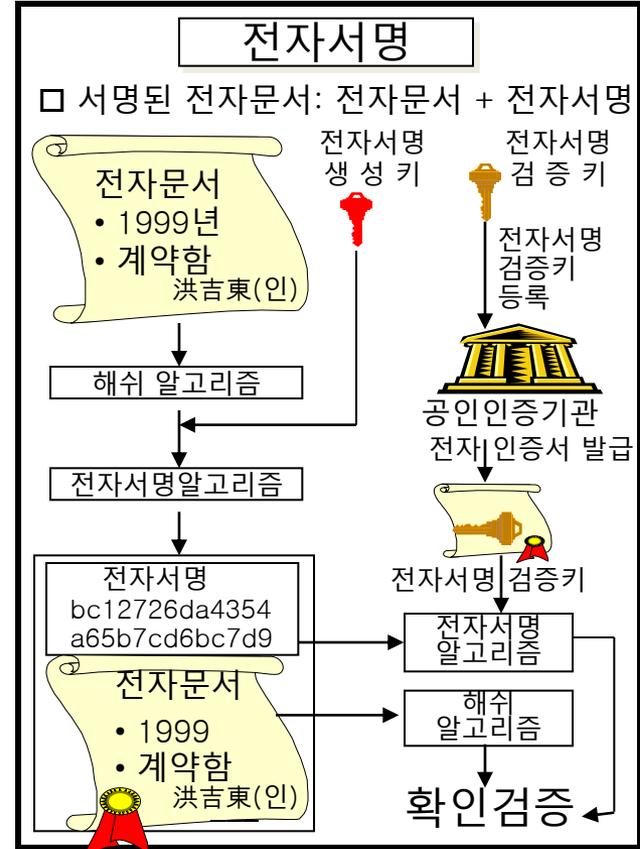
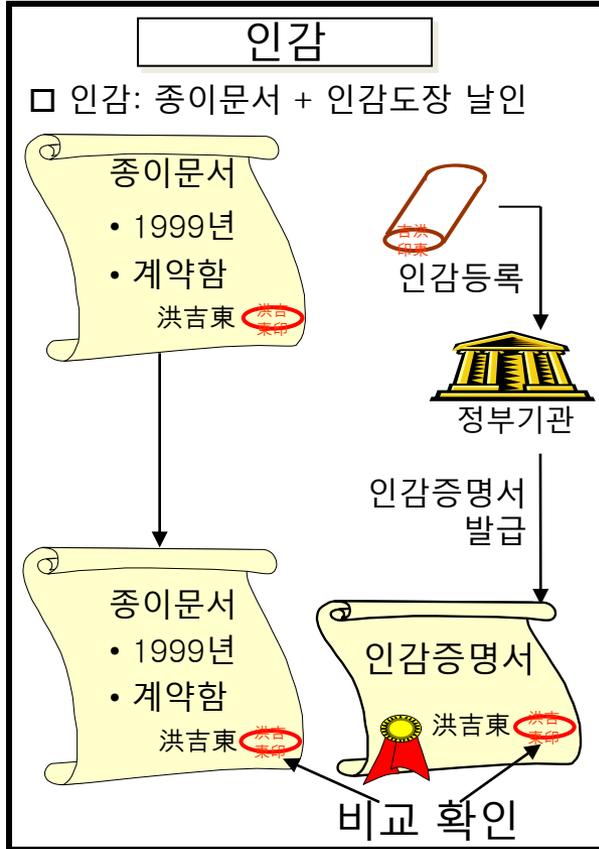


$$V = h(H) = h(I) \quad V' = h(H') = h(I')$$

전자서명

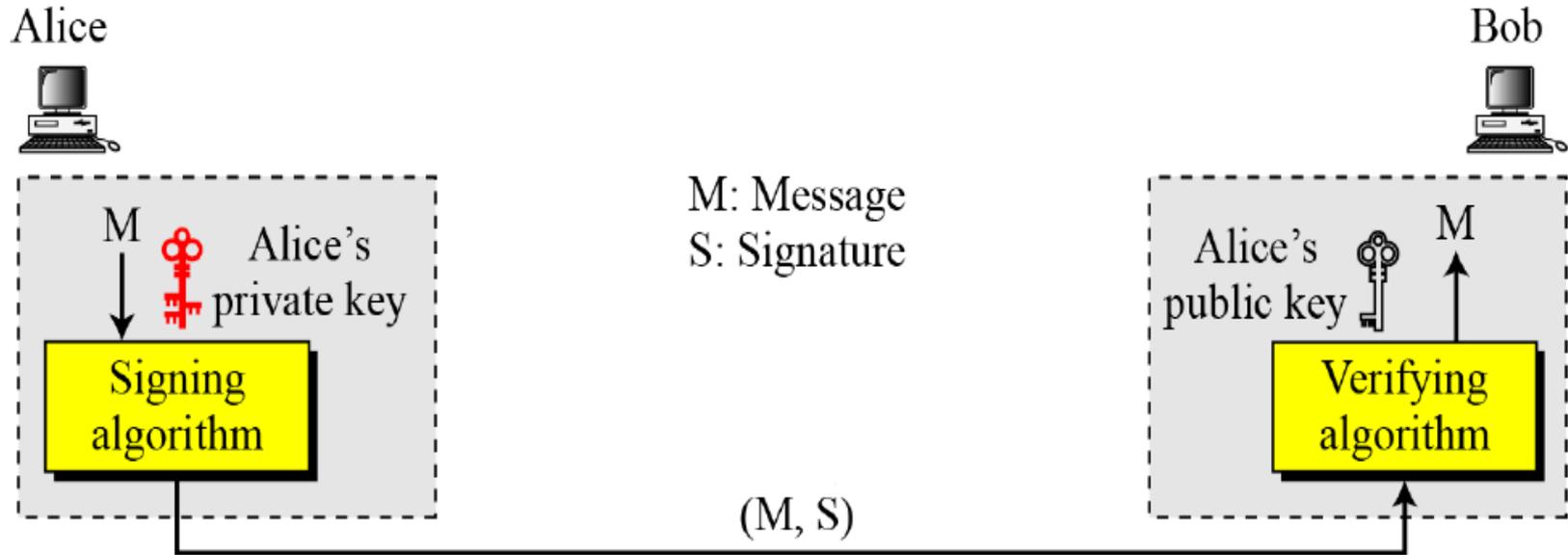
- 현재 사용되고 있는 도장이나 서명을 디지털로 실현(사용자 인증+메시지 인증)
- 특징
 - 무결성(Integrity) : 서명한 데이터의 전송 과정에서 변형되지 않음
 - 위조 불가(Unforgeable) : 서명자만이 서명 생성가능
 - 서명자 인증(Authentic) : 서명자를 확인 가능
 - 재사용 불가(Not Reusable) : 다른 문서의 서명으로 사용불가
 - 변경 불가(Unalterable) : 서명된 문서의 내용 변경 불가
 - 부인 불가(Nonrepudiation) : 서명 사실 부인 불가

전자서명



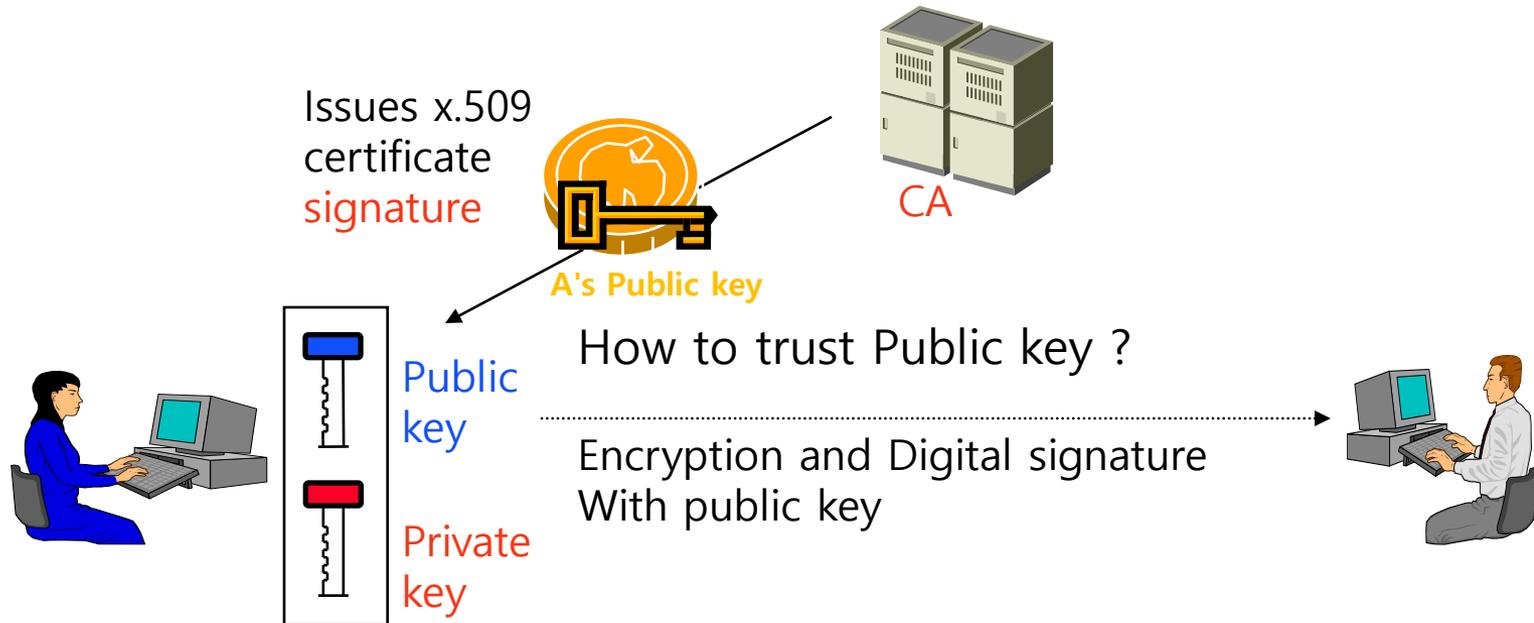
전자 서명

- 전자 서명의 동작 원리



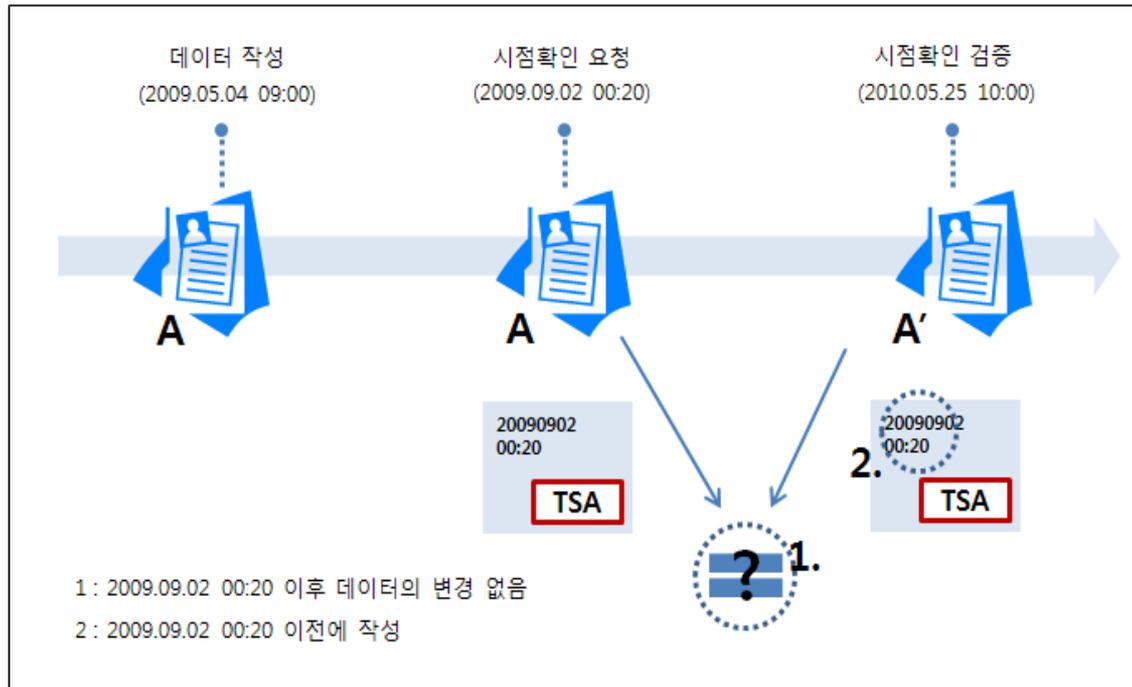
공개키 기반 구조(PKI)

- 공개키 등록, 유지, 폐기 등의 관리체계가 필요



시점 확인 서비스(Time Stamping Service)

- 임의의 디지털 데이터가 특정한 시점에 존재하였으며, 특정 시점 이후에는 데이터의 내용이 변경되지 않았음을 증명해주는 서비스
- 디지털 데이터와 객관적인 시각 정보를 결합한 뒤 제 3자의 전자 서명을 거쳐 시점 확인 토큰(Time Stamping Token)을 생성



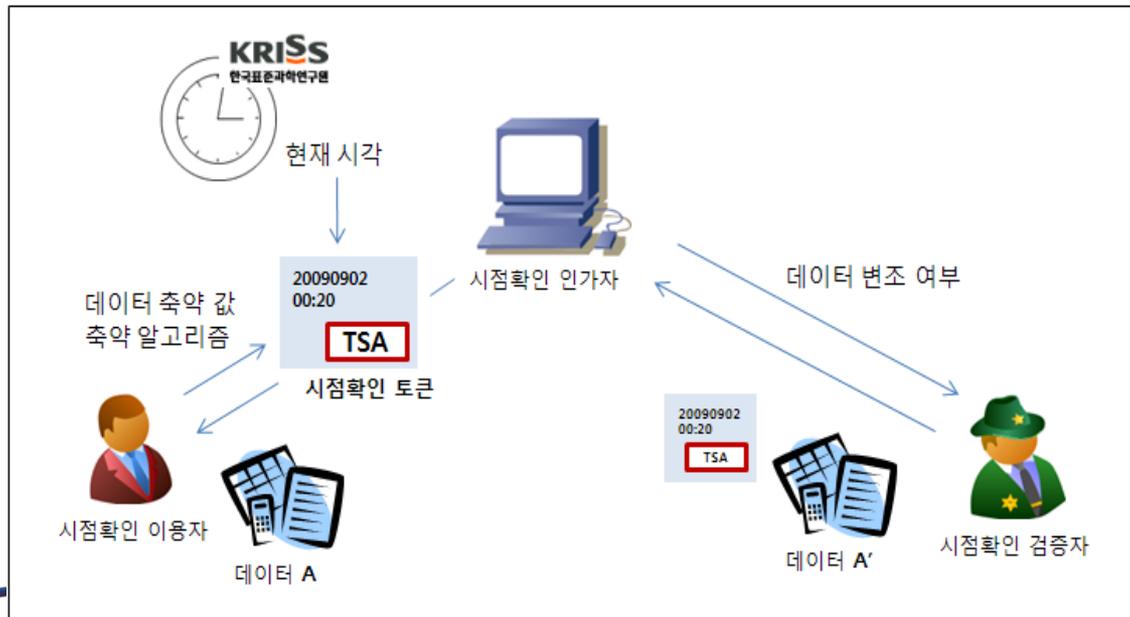
시점 확인 서비스(Time Stamping Service)

- 전자 서명과의 차이

- 전자 서명 : 누가 서명 했느냐에 초점
- 시점 확인 서비스 : 언제 서명했느냐에 초점

- 시점 확인 서비스의 주체

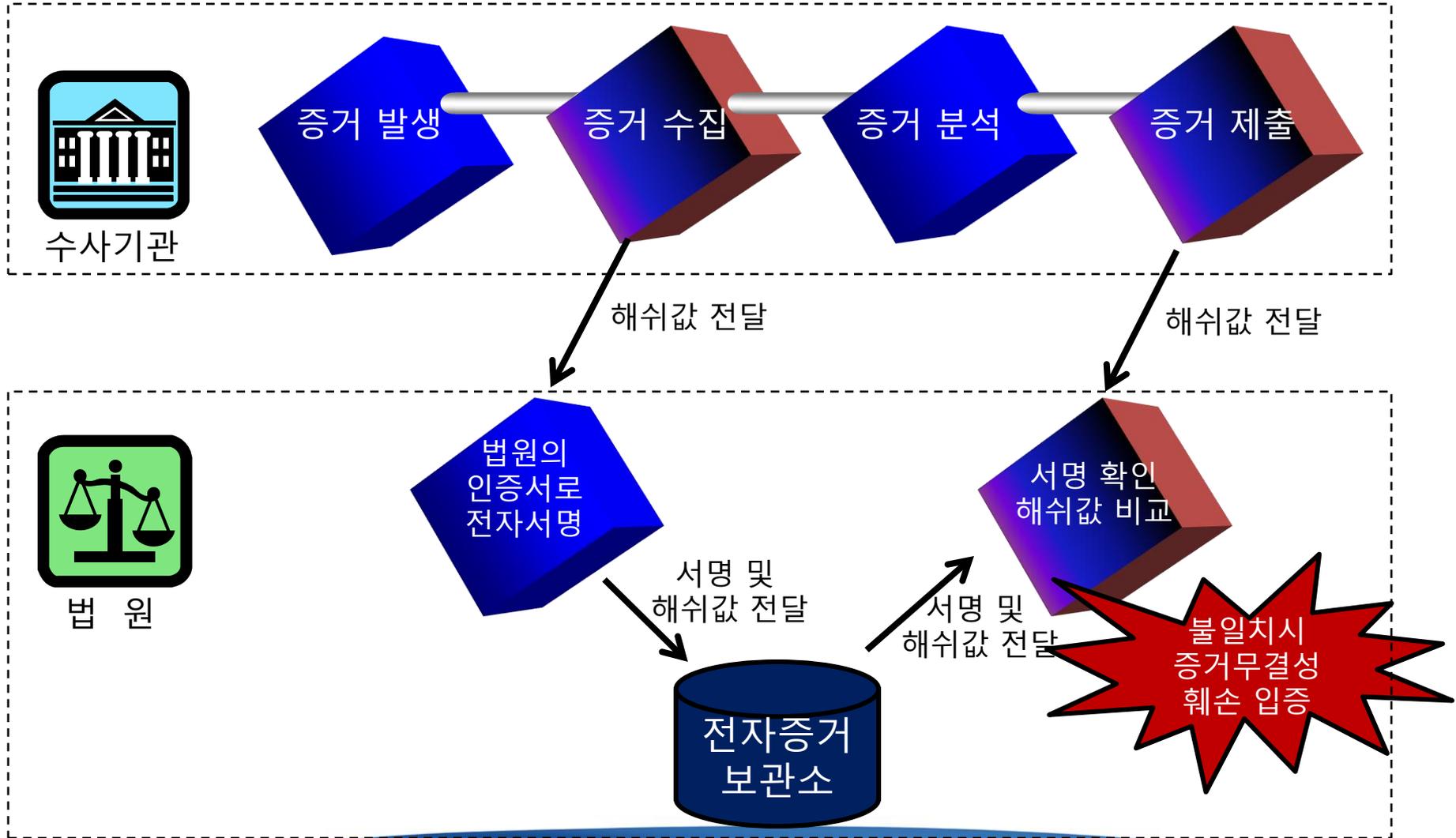
- 시점확인 인가자(Time Stamping Authority, TSA) : 토큰 발행
- 시점확인 이용자 : 시점 확인 요청
- 시점확인 검증자 : 이용자 및 토큰의 유효성 검증



시점 확인 서비스(Time Stamping Service)

- 시점확인 인가자 : 토큰의 무결성 검증을 위해서는 신뢰할수 있는 제 3의 기관(Third Trusted Party, TTP)
 - 한국정보인증, 코스콤 공인인증센터, 금융결제원 전자인증센터
- 시점확인 토큰에 디지털 데이터의 존재성과 무결성에 대한 법적 효력을 명시적으로 부여하고 있지 않음
 - 공인전자서명의 경우 전자서명법 제3조 2항에 의거 서명자 본인 추정력과 내용의 무결성이 인정
 - TTP에 의해 발급된 시점확인 토큰이 디지털 데이터의 존재 시점과 무결성을 입증하기 위해 사용될 수 있음

디지털 증거의 인증



Q & A