

제 10 장

디지털 서명



박종혁 교수

Tel: 970-6702

Email: jhpark1@seoultech.ac.kr

- 1절 디지털 서명
- 2절 디지털 서명 방법
- 3절 디지털 서명에 대한 의문
- 4절 디지털 서명 활용 예
- 5절 RSA에 의한 디지털 서명
- 6절 다른 디지털 서명
- 7절 디지털 서명에 대한 공격
- 8절 기타 기술과의 비교
- 9절 디지털 서명으로 해결할 수 없는 문제
- 10절 2019 신기술 전자서명 서비스 소개

제1절 디지털 서명

1.1 엘리스 차용서

1.2 메시지 인증코드에서 디지털 서명으로

1.3 서명 작성과 서명 검증

1.4 공개 키 암호와 디지털 서명

1.1 앨리스의 차용서

- 차용서를 이메일로 보내면 어떨까?
 - 메일을 누군가가 변경했을 수 있다
 - 처음부터 앨리스인 것처럼 거짓 행세를 한 누군가가 보낸 것인지도 모른다
 - 나중에 앨리스가 「그런 차용서 난 몰라」라고 부인할 수도 있다

1.2 메시지 인증코드에서 디지털 서명으로

- 메시지 인증 코드의 한계

- 메시지 인증 코드를 사용하면 메시지의 변경과 거짓 행세를 검출할 수 있다
- 메시지 인증 코드는 부인 방지에는 도움이 되지 않는다

1.2 메시지 인증코드에서 디지털 서명으로

- 디지털 서명을 이용한 해결

디지털 서명(digital signature)

- 앨리스가 사용하는 키는 앨리스만이 알고 있는 개인적인 것
- 앨리스는 메시지 송신 시에 그 개인적인 키를 써서 「서명」을 작성
- 수신자 받은 앨리스의 키와는 다른 키를 써서 「서명」을 검증

1.3 서명 작성과 서명 검증

- 메시지의 서명을 작성하는 행위
 - 디지털 서명에서는 「서명용 키」와 「검증용 키」가 나누어져 있어서 검증용 키로 서명을 작성할 수는 없다.
- 메시지의 서명을 검증하는 행위
 - 「서명용 키」는 서명을 하는 사람만이 가지고 있지만, 「검증용 키」는 서명을 검증하는 사람이라면 누구라도 가질 수 있다.

공개키 암호와 디지털 서명

- 공개 키 암호
 - 「암호 키」와 「복호 키」가 나누어져 있어 암호 키로 복호화를 행할 수는 없다
 - 「복호 키」는 복호화를 행하는 사람만이 가지고 있지만, 「암호 키」는 암호화를 행하는 사람이라면 누구나 가질 수 있다
- 디지털 서명은 공개 키 암호를 「역으로 사용」 함으로써 실현

공개키 암호와 디지털 서명 키 사용방법

	개인 키	공개 키
공개 키 암호	수신자가 복호화에 사용	송신자들이 암호화에 사용
디지털 서명	서명자가 서명 작성에 사용	검증자들이 서명 검증에 사용
키는 누가 갖는가?	개인이 갖는다.	필요한 사람은 아무나 가지고 있어도 된다.

1.4 공개 키 암호와 디지털 서명

- 메시지를 개인 키로 암호화하는 것이 서명 작성에 해당
- 암호문을 공개 키로 복호화하는 것이 서명 검증에 해당

공개 키에 의한 암호화

공개 키로 암호화한 암호문은 대응하는
개인 키로만 바르게 복호화할 수 있다.

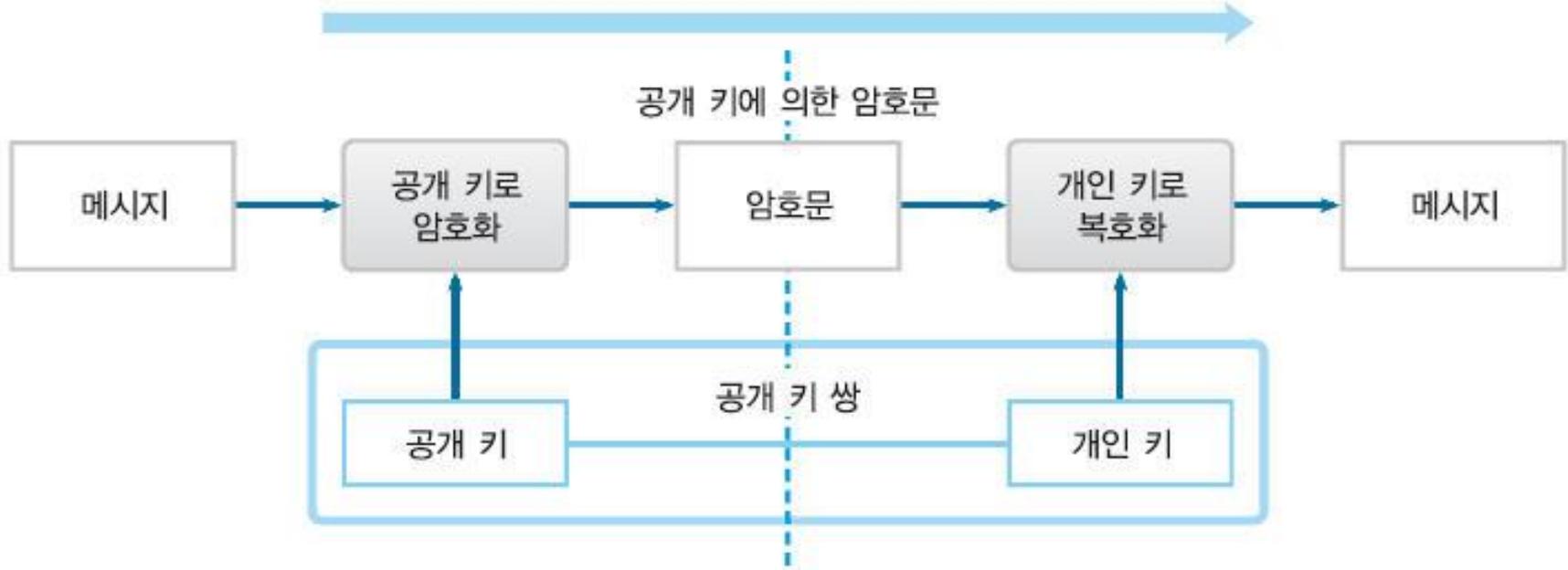


그림 10-1 · 공개 키에 의한 암호화(공개 키 암호)

개인 키에 의한 암호화

개인 키로 암호화한 암호문은 대응하는
공개 키로만 바르게 복호화할 수 있다.

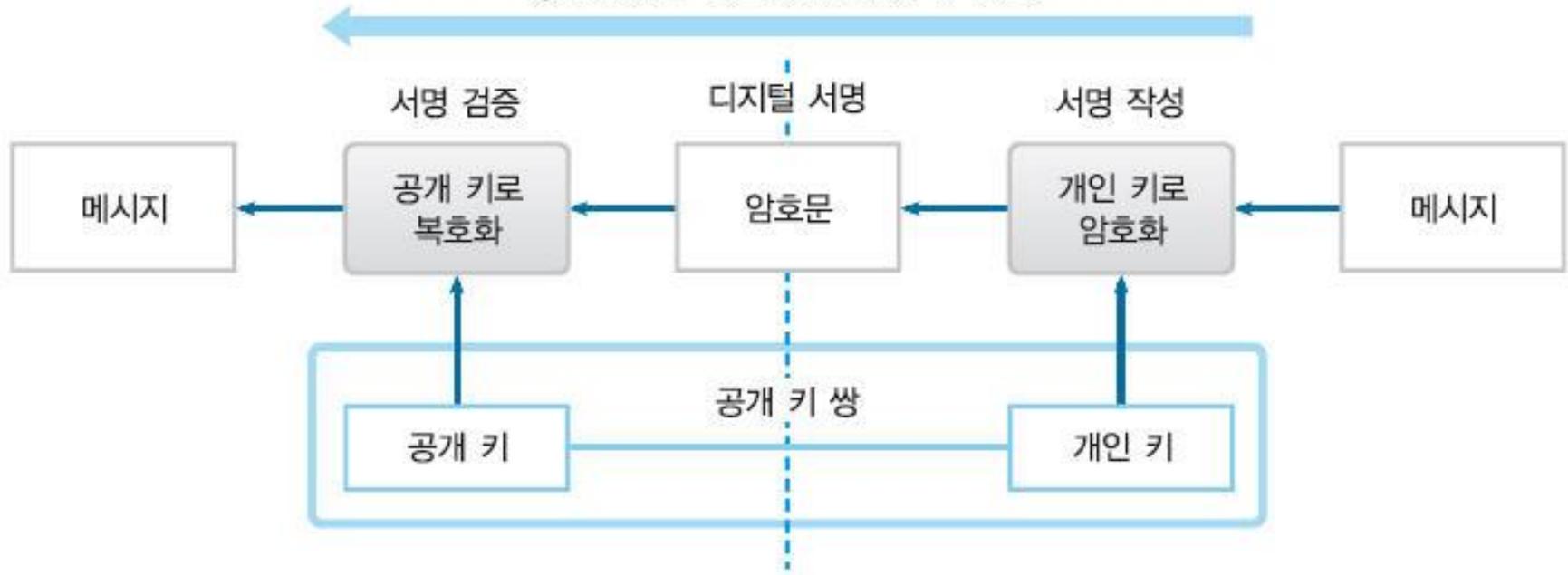
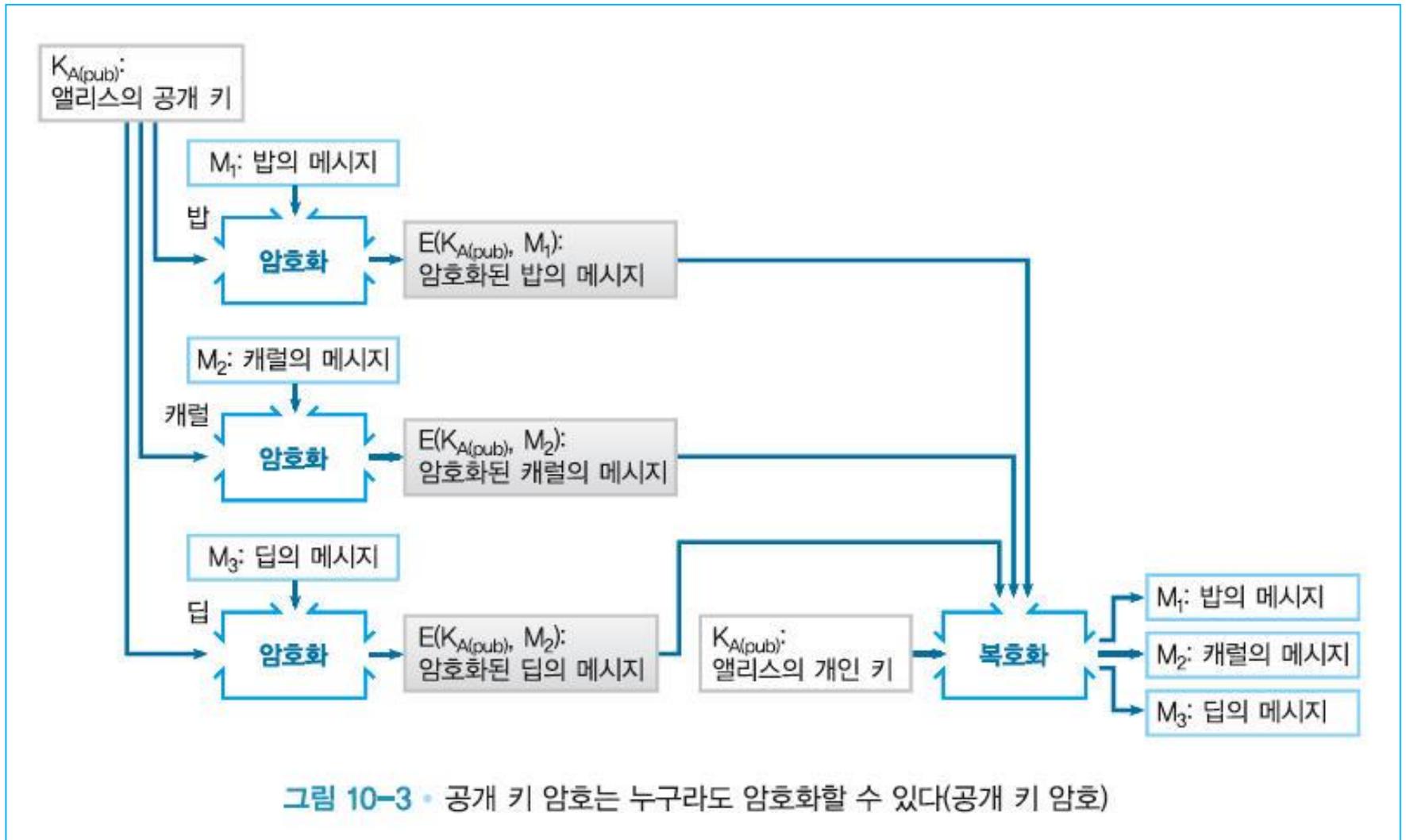


그림 10-2 · 개인 키에 의한 암호화(디지털 서명)

공개키 암호는 누구라도 암호화



디지털 서명은 누구라도 서명검증

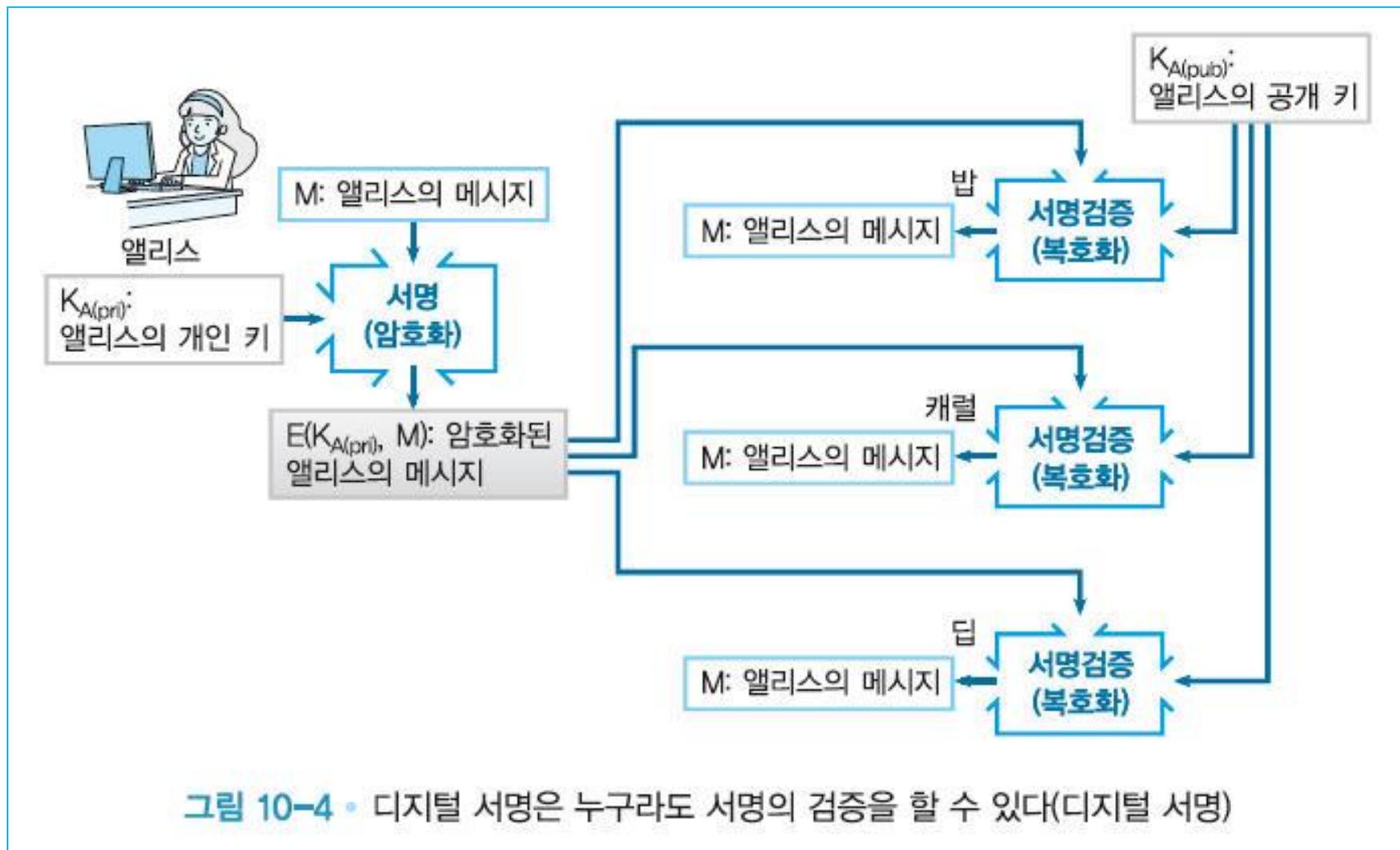


그림 10-4 • 디지털 서명은 누구라도 서명의 검증을 할 수 있다(디지털 서명)

제2절 디지털 서명 방법

2.1 디지털 서명 특징

2.2 디지털 서명 요구 조건

2.3 메시지에 직접 서명하는 방법

2.4 메시지의 해시 값에 서명하는 방법

2.1 디지털 서명 특징

- 위조불가(Unforgeable)
 - 서명자만이 서명문을 생성 가능
- 서명자 인증(Authentic)
 - 서명문의 서명자를 확인 가능
- 재사용 불가(Not Reusable)
 - 서명문의 서명은 다른 문서의 서명으로 사용 불가능
- 변경 불가(Unalterable)
 - 서명된 문서의 내용 변경 불가능
- 부인 불가(Nonrepudiation)
 - 서명자는 후에 서명한 사실을 부인 불가능

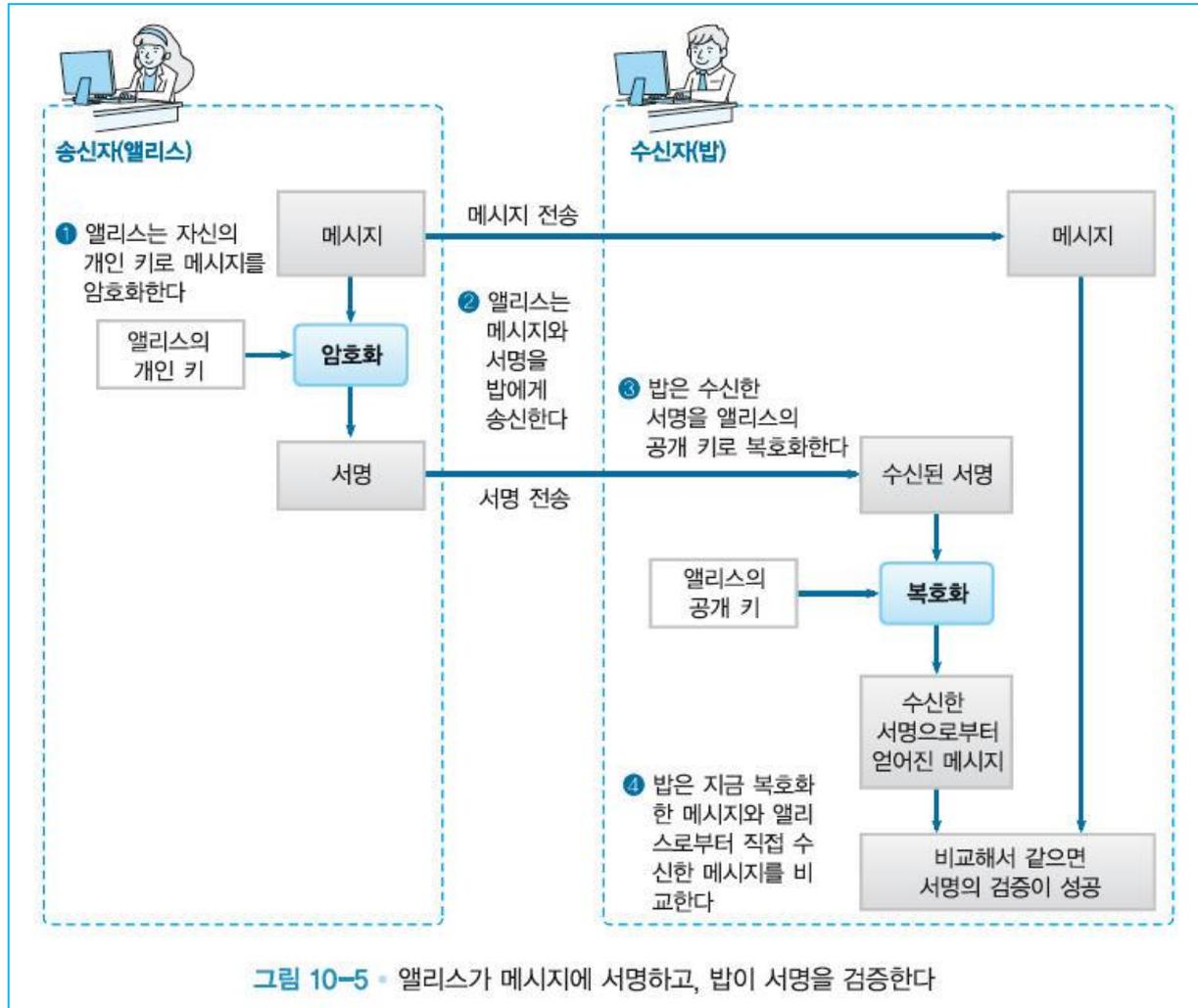
2.2 디지털 서명 요구 조건

- 서명은 메시지에 의존하는 비트 형태이어야 한다.
- 위조와 부인 방지 위해, 송신자의 유일한 정보 비트를 이용해야 함
- 서명문을 만들기가 쉬워야 한다.
- 서명문을 인식, 확인 하기가 쉬워야 한다.
- 서명문을 위조하는 것이 계산적으로 실행 불가능
- 기억장소에 서명문의 복사본을 유지하는 것이 실용적이어야 한다.

2.3 메시지에 직접 서명하는 방법

1. 앨리스는 자신의 개인 키로 메시지를 암호화한다.
2. 앨리스는 메시지와 서명을 밥에게 송신한다.
3. 밥은 수신한 서명을 앨리스의 공개 키로 복호화한다.
4. 밥은 이제 서명을 복호화해서 얻어진 메시지와 앨리스로부터 직접 수신한 메시지를 비교한다.

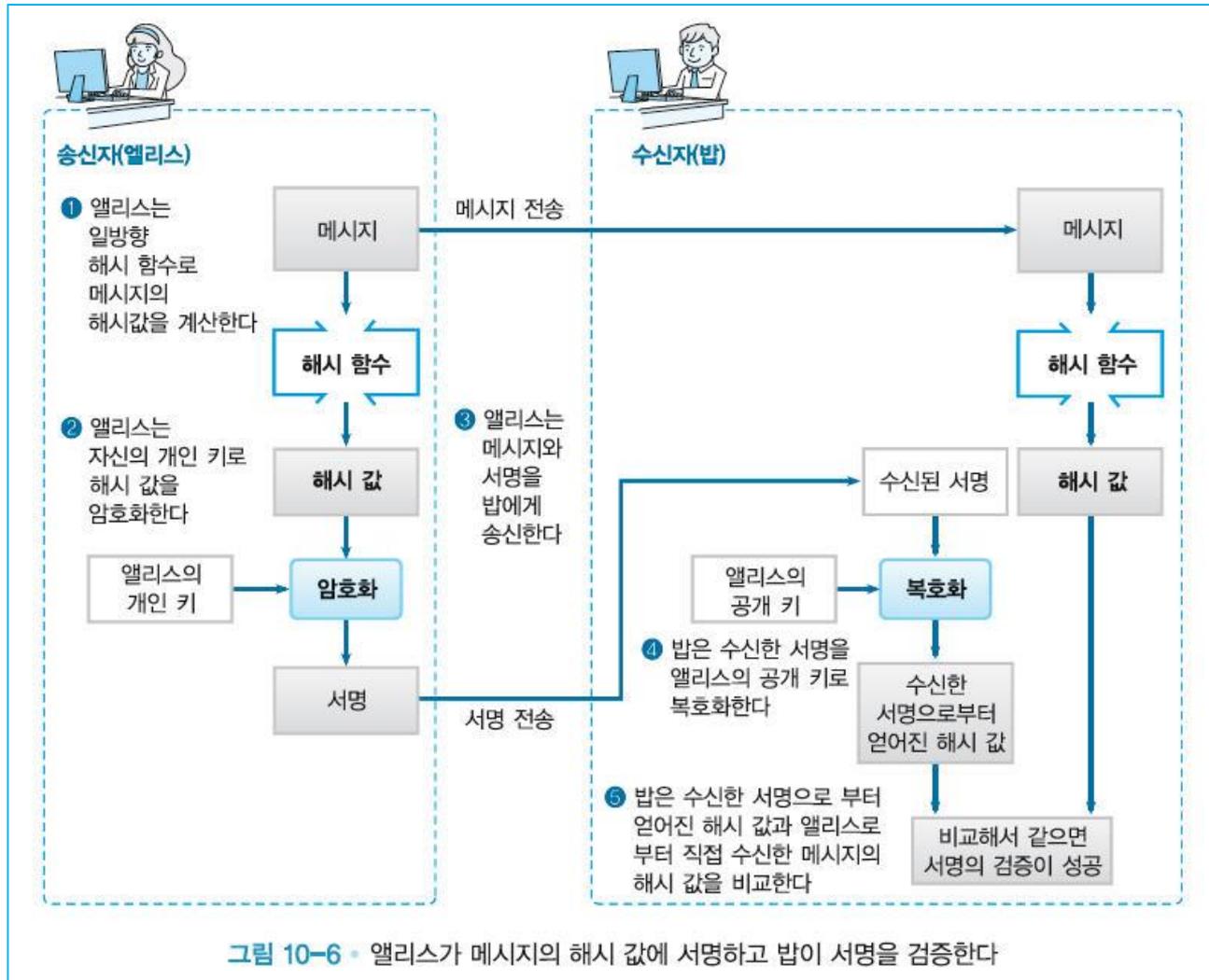
앨리스가 메시지에 서명하고 밥이 서명 검증



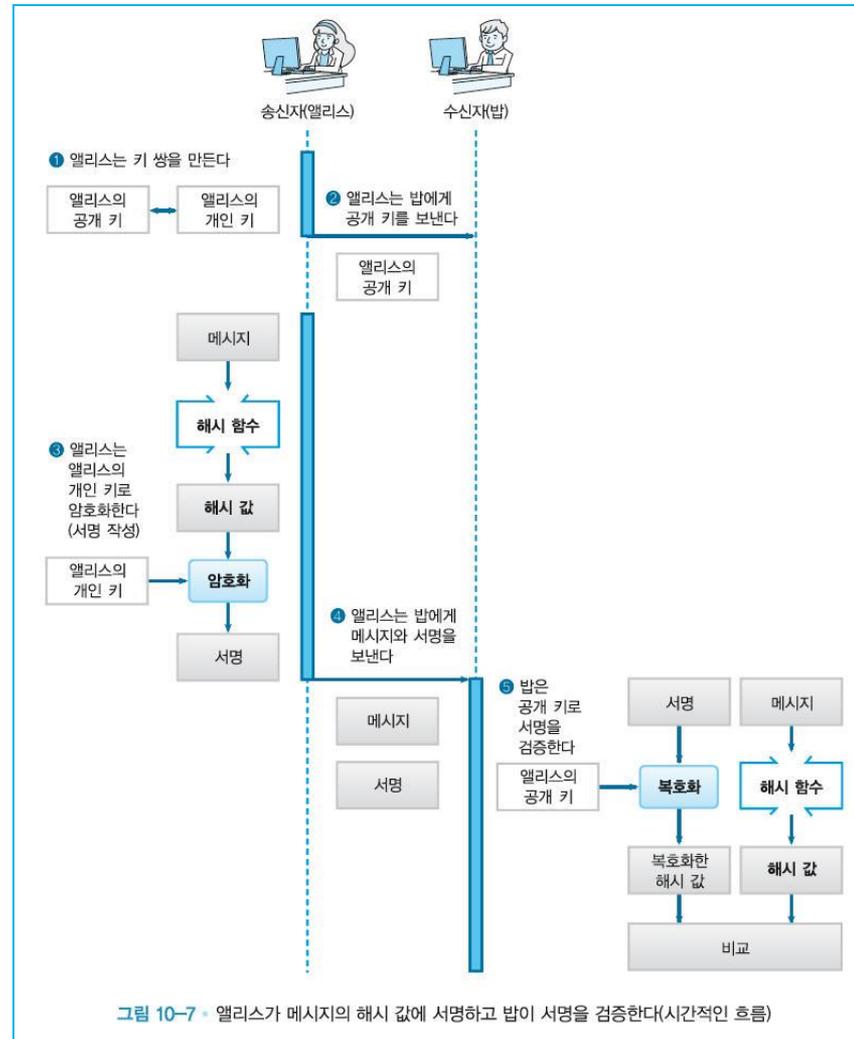
2.4 메시지의 해시 값에 서명하는 방법

1. 앨리스는 일방향 해시 함수로 메시지의 해시 값을 계산한다.
2. 앨리스는 자신의 개인 키로 해시 값을 암호화한다.
3. 앨리스는 메시지와 서명을 밥에게 송신한다.
4. 밥은 수신한 서명을 앨리스의 공개 키로 복호화한다.
5. 밥은 수신한 서명으로부터 얻어진 해시 값과 앨리스로부터 직접 수신한 메시지의 해시 값을 비교한다.

앨리스가 메시지의 해시 값에 서명하고 밥이 서명 검증



앨리스가 메시지의 해시 값에 서명하고 밥이 서명 검증(시간 흐름)



제3절 디지털 서명에 대한 의문

3.1 암호문이 왜 서명으로서 사용 가능한 것인가?

3.2 기밀성을 유지할 수 없는 것은 아닐까?

3.3 복사된 서명이 만들어지는 것은 아닐까?

3.4 서명 변경이 가능한 것은 아닐까?

3.5 서명만 재이용할 수 있는 것은 아닐까?

3.6 서명을 삭제하더라도 계약파기를 할 수 없는 것은 아닌가?

3.7 어떻게 해서 부인 방지가 되는 것인가?

3.8 디지털 서명은 정말로 종이 서명 대용이 되는 것일까?

3.1 암호문이 왜 서명으로서 사용 가능한 것인가?

- 개인 키로 암호화한다는 것은, 행하고 있는 처리의 내용을 설명한 것이지, 여기에서는 기밀성을 실현하기 위해 암호화하고 있는 것은 아니다.
- 인증자(authenticator)
 - 키를 가지고 있는 사람만이 만들 수 있는 정보

3.2 기밀성을 유지할 수 없는 것은 아닐까?

- 맞다. 디지털 서명은 기밀성을 지키기 위한 것은 아니다.
- 만약 기밀성이 필요하다면 메시지를 그대로 보내는 것이 아니고, 암호화를 별도로 행해서 보내야 된다.

3.3 복사된 서명이 만들어지는 것은 아닐까?

- 통상의 파일 복사처럼 서명도 복사본을 간단히 만들 수 있다.
- 하지만, 서명 복사를 만들 수 있다고 해서 서명이 무의미해지는 것은 아니다.
 - 왜냐 하면 복사한 데이터가 표현하고 있는 것은 「특정의 서명자가 특정의 메시지에 대해서 서명했다」고 하는 것뿐이기 때문이다.
 - 복사해도 서명자는 바뀌지 않고 메시지의 내용도 바뀌지 않는다.

복사된 서명

- 특정 서명자와 특정 메시지가 결부되어 있다는 사실이 중요
- 아무리 복사를 해도 「그 메시지에 누가 서명했는가」 하는 사실에는 조금도 변화가 없다.
- 복사는 할 수 있다. 그러나 그것에 의해 서명이 무의미해지는 것은 아니다.

3.4 서명 변경이 가능한 것은 아닐까?

- 확실히 서명한 후에 메시지와 서명의 내용을 수정할 수는 있다. 그러나 수정해 버리면 서명의 검증에 실패하기 때문에, 검증하는 사람은 수정이 행해졌다는 것을 검출할 수 있다.

또 다른 의문

- 서명 대상의 메시지와 서명 양쪽을 수정해서 서명의 검증에 성공할 수 있도록 앞뒤를 잘 맞출 수 있는 않을까?
- 아니다. 그것은 사실상 불가능하다.

3.5 서명만 재이용할 수 있는 것은 아닐까?

- 확실히 서명 부분만을 잘라내서 다른 메시지에 첨부하는 것은 가능하다.
- 그러나 서명의 검증에는 실패한다.

3.6 서명을 삭제하더라도 계약파기를 할 수 없는 것은 아닌가?

- 분명히 디지털 서명이 붙은 차용서는 삭제해도 파기할 수 없다.
- 디지털 서명이 붙은 차용서를 파기하는 경우에는 「영수증」에 상당하는 문서를 새로 만들고, 그것에 대해 상대방에게 디지털 서명을 부탁해야 한다.

3.7 어떻게 해서 부인 방지가 되는 것인가?

- 디지털 서명의 경우 서명을 작성할 수 있는 키(개인 키)는 송신자만 가지고 있다. 그러므로 서명을 작성할 수 있는 것은 송신자뿐이다.
- 그렇기 때문에 송신자는 「그 서명을 작성한 것은 내가 아니다」라고 주장할 수가 없다

3.8 디지털 서명은 정말로 종이 서명 대용이 되는 것일까?

- 한국에서는 1999년 전자서명법이 제정, 시행
- 이 법률들은 전자적으로 실현된 「서명」을 날인이나 손으로 쓴 서명과 같이 취급하기 위한 법적인 근거
- 그러나 실제로는 디지털 서명에 관한 분쟁이 발생하여 디지털 서명의 유효성을 둘러싸고 재판이 일어날 가능성은 충분히 생각할 수 있다.

제4절 디지털 서명 활용 예

4.1 보안 공지

4.2 소프트웨어 다운로드

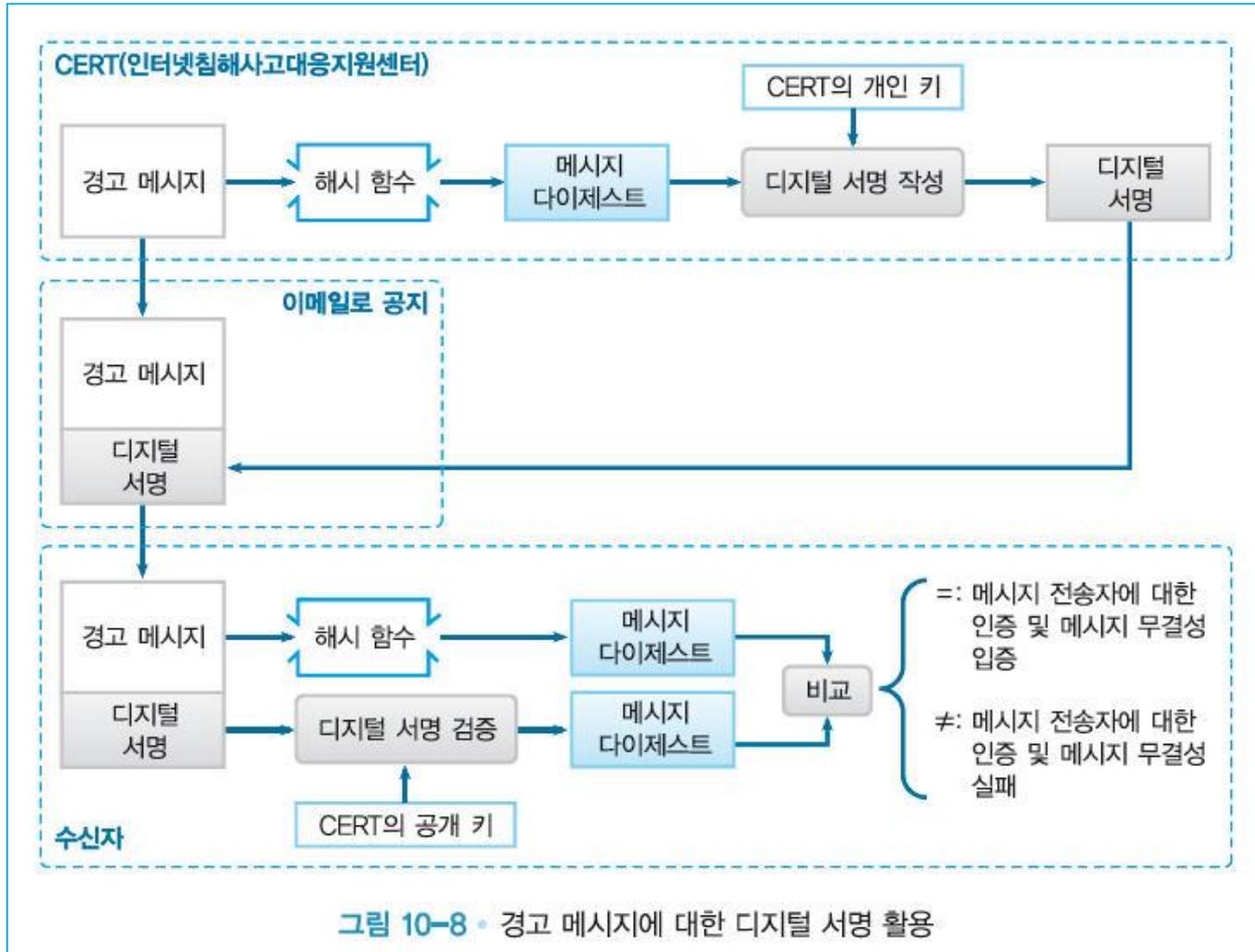
4.3 공개 키 인증서

4.4 SSL/TLS

4.1 보안 공지

- 클리어서명(clearsign)
 - 메시지를 암호화하지 않고 서명만 한 것

경고 메시지에 대한 디지털 서명 활용



4.2 소프트웨어 다운로드

- 소프트웨어의 작성자가 소프트웨어에 디지털 서명을 작성하고, 다운로드한 후에 서명을 검증하면 적극적 공격자 맬로리에 의한 내용 조작을 검출하는 것이 가능

4.3 공개 키 인증서

- 디지털 서명을 검증하려면 바른 공개 키가 필요
- 자신이 입수한 공개 키가 바른 공개 키인지 어떤지를 검증하기 위해서 공개 키를 메시지로 간주하고 그것에 디지털 서명을 한 것
- 공개 키 인증서
 - 공개 키에 디지털 서명을 붙인 것

4.4 SSL/TLS

- SSL/TLS에서는 서버가 올바른 것이라는 것을 인증하기 위해서 서버 인증서를 이용
- 이것은 서버의 공개키에 디지털 서명을 한 것

제5절 RSA에 의한 디지털 서명

5.1 RSA에 의한 서명 작성

5.2 RSA에 의한 서명 검증

5.3 자세한 RSA 서명

5.1 RSA에 의한 서명 작성

- 서명 = (메시지)^D mod N
- D와 N은 서명자의 개인 키

5.2 RSA에 의한 서명 검증

- 서명으로부터 얻어진 메시지 = (서명)^E mod N
- E와 N은 서명자의 공개 키

RSA 서명 작성과 검증

키	공개 키	(E, N)
쌍	개인 키	(D, N)
서명의 작성		$\text{서명} = (\text{메시지})^D \bmod N$ <p>(메시지를 D 제공해서 N으로 나눈 나머지)</p>
서명의 검증		$\text{서명으로부터 얻어진 메시지} = (\text{서명})^E \bmod N$ <p>(서명을 E제공해서 N으로 나눈 나머지)</p> <p>「서명으로부터 얻어진 메시지」와 「메시지를 비교한다</p>

5.3 자세한 RSA 서명

- 공개 키: $E = 5, N = 323$
- 개인 키: $D = 29, N = 323$
- N 이 323이므로 메시지는 0 ~ 322 범위의 정수에서 고른다.
- 여기서는 123이라는 메시지에 서명을 해 보자.

서명 작성

- 메시지^D mod N = $123^{29} \bmod 323 = 157$
- 서명은 157
- 수신자에게 전달할 것
 - (메시지, 서명) = (123, 157)

서명 검증

- (메시지, 서명) = (123, 157) 수신
- 공개 키(E, N) = (5, 323)을 사용해서 서명으로부터 얻어진 메시지를 계산
- $\text{서명}^E \bmod N = 157^5 \bmod 323 = 123$
- 이 메시지 123은 분명히 송신자가 보낸 메시지 123과 일치
- 서명 검증에 성공

제6절 다른 디지털 서명

6.1 ElGamal 방식

6.2 DSA

6.3 Rabin 방식

6.1 ElGamal 방식

- ElGamal 방식은 Taher ElGamal에 의한 공개 키 알고리즘으로 mod N 으로 이산대수를 구하는 것이 곤란하다는 것을 이용
- ElGamal 방식은 공개 키 암호와 디지털 서명에 이용
- 암호 소프트웨어 GnuPG에서도 알고리즘의 하나로 사용

6.2 DSA

- DSA(Digital Signature Algorithm)은 디지털 서명 알고리즘의 일종으로 NIST(National Institute of Standards and Technology)가 1991년에 제정한 디지털 서명 규격(DSS)
- DSA는 Schnorr의 알고리즘과 ElGamal 방식의 변종으로 디지털 서명에만 이용

6.3 Rabin 방식

- Rabin 방식
M. O. Rabin에 의한 공개 키 알고리즘으로 mod N 으로 평방근을 구하는 것이
곤란하다는 것을 이용
- Rabin 방식
공개 키 암호와 디지털 서명에 이용

6.4 기타 디지털 서명

- RSA : Rivest & Shamir & Adleman 제안, 1978
- Schnorr : 1989년 Schnorr가 개인식별방식과 함께 제안
- GQ : 1998년 Guillou 와 Quisquater가 제안
- SIGN : 1991년 후지오까, 오까모도, 미야구찌가 제안
- Fiat Shamir : Fiat, Shamir, 1987
- KCDSA : 1999, TTA표준
- ECDSA : 2000, DSA의 변형, 미국 NIST 전자서명 표준
- EC-KCDSA : 2001, KC-DISA의 변형, TTA표준

제7절 디지털 서명에 대한 공격

7.1 중간자 공격

7.2 일방향 해시 함수에 대한 공격

7.3 디지털 서명을 사용한 공개 키 암호 공격

7.4 잠재적 위조

7.5 기타 공격

7.1 중간자 공격

- 중간자 공격은 디지털 서명에도 위협이 되는 공격
- 중간자 공격을 막으려면 입수한 공개 키가 정확한 상대의 것인지 아닌지를 확인하는 것이 필요
- 핑거프린트(fingerprint)
- 공개 키를 취급하는 소프트웨어는 공개 키의 해시 값을 표시하는 수단을 준비
- 이 해시 값을 핑거프린트라고 한다

7.2 일방향 해시 함수에 대한 공격

- 디지털 서명에서 사용하는 일방향 해시 함수는 충돌내성을 가져야만 한다
- 만약 충돌내성이 없으면 디지털 서명을 한 해시 값과 같은 해시 값을 갖는, 다른 메시지를 만들 수 있다

7.3 디지털 서명을 사용한 공개 키 암호 공격

- 공격자의 교묘한 속임수

밥 씨, 안녕하세요.

저는 암호기술을 연구하고 있는 맬로리라고 합니다.

저는 지금 디지털 서명에 관한 실험을 하고 있는데,

첨부된 데이터에 당신의 서명을 붙여서 회신해 주시면 감사하겠습니다.

첨부된 데이터는 랜덤한 데이터(사실은 도청한 암호문)이기 때문에 문제는 발생하지 않습니다.

협조해 주셔서 감사합니다.

- 맬로리 -

밥의 어리석은 행동

- 밥은 맬로리한테 온 메일을 읽고 첨부 데이터를 보니 확실히 랜덤한 데이터인 것 같다고 여긴다(하지만 실은 이것은 앨리스가 밥의 공개 키로 암호화한 암호문이다).
- 밥은 순수한 마음으로 첨부 데이터에 서명을 한다.

맬로리의 목적 달성

$$\begin{aligned} \text{서명} &= (\text{첨부 데이터})^D \bmod N && \text{(RSA의 서명 작성)} \\ &= (\text{암호문})^D \bmod N && \text{(첨부 데이터는 실은 암호문} \\ & && \text{이기 때문에)} \\ &= \text{메시지} && \text{(복호화 처리가 행하여졌기} \\ & && \text{때문에)} \end{aligned}$$

- 의미를 모르는 메시지에는 절대로 디지털 서명을 하지 않는다

7.4 잠재적 위조

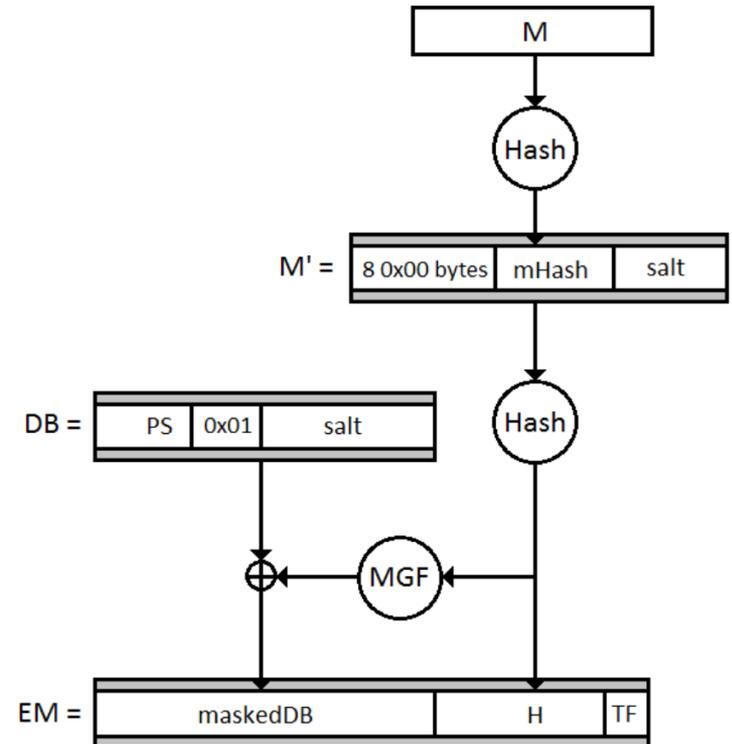
- 개인 키가 없는 공격자가 의미가 있는 메시지를 만들고 그에 대한 바른 디지털 서명을 만들 수 있다면 그 디지털 서명 알고리즘은 안전하지 않음
- 의미가 없는 메시지(예, 랜덤한 비트 열)라고 하더라도, 만약 올바른 디지털 서명을 만들 수 있다면(즉, 검증을 통과할 수 있는 디지털 서명이 된다면), 그것은 디지털 서명 알고리즘에 대한 위협

RSA에서의 잠재적 위조

- 메시지를 RSA로 복호화하는 디지털 서명 알고리즘에서는 잠재적 위조가 가능
 - 랜덤한 비트 열 S 를 RSA의 공개 키로 암호화한 것을 M 이라고 하면, S 가 M 의 바른 디지털 서명이 되어 버리기 때문(이것은 앞 절에서 설명한 것의 역이다).
 - 공개 키는 공격자도 구할 수 있으므로 디지털 서명의 잠재적 위조가 가능

RSA 잠재적 위조 대처 법

- RSA-PSS(Probabilistic Signature Scheme)
 - RSA를 개량한 방법
 - 메시지에 대해서가 아니라 메시지의 해시 값에 대해서 서명하는 방법
 - 해시 값의 계산 시에는 메시지에 솔트를 더해 더욱 안전성을 향상



7.5 기타 공격

- 공개 키 암호에 대한 공격의 대부분은 디지털 서명에 대한 공격으로서도 사용가능.
- 개인 키에 대한 전사공격
- RSA의 N 을 소인수분해하는 공격

제8절 기타 기술과의 비교

8.1 메시지 인증 코드와 디지털 서명

8.2 하이브리드 암호 시스템과 해시 값에 대한 디지털 서명

8.1 메시지 인증 코드와 디지털 서명

- 대칭 암호와 공개 키 암호의 비교 및 메시지 인증 코드와 디지털 서명의 비교

	대칭 암호	공개 키 암호
송신자	공유 키로 암호화	공개 키로 암호화
수신자	공유 키로 복호화	개인 키로 복호화
키 배송 문제	일어난다	일어나지만, 공개 키의 인증이 별도로 필요
기밀성	○	○

	메시지 인증 코드	디지털 서명
송신자	공유 키로 MAC 값을 계산	개인 키로 서명을 작성
수신자	공유 키로 MAC 값을 계산	공개 키로 서명을 검증
키 배송 문제	일어난다	일어나지만, 공개 키의 인증이 별도로 필요
무결성	○	○
인증	○(통신 상대에 대해서만)	○(제삼자에 대해서도)
부인 방지	×	○

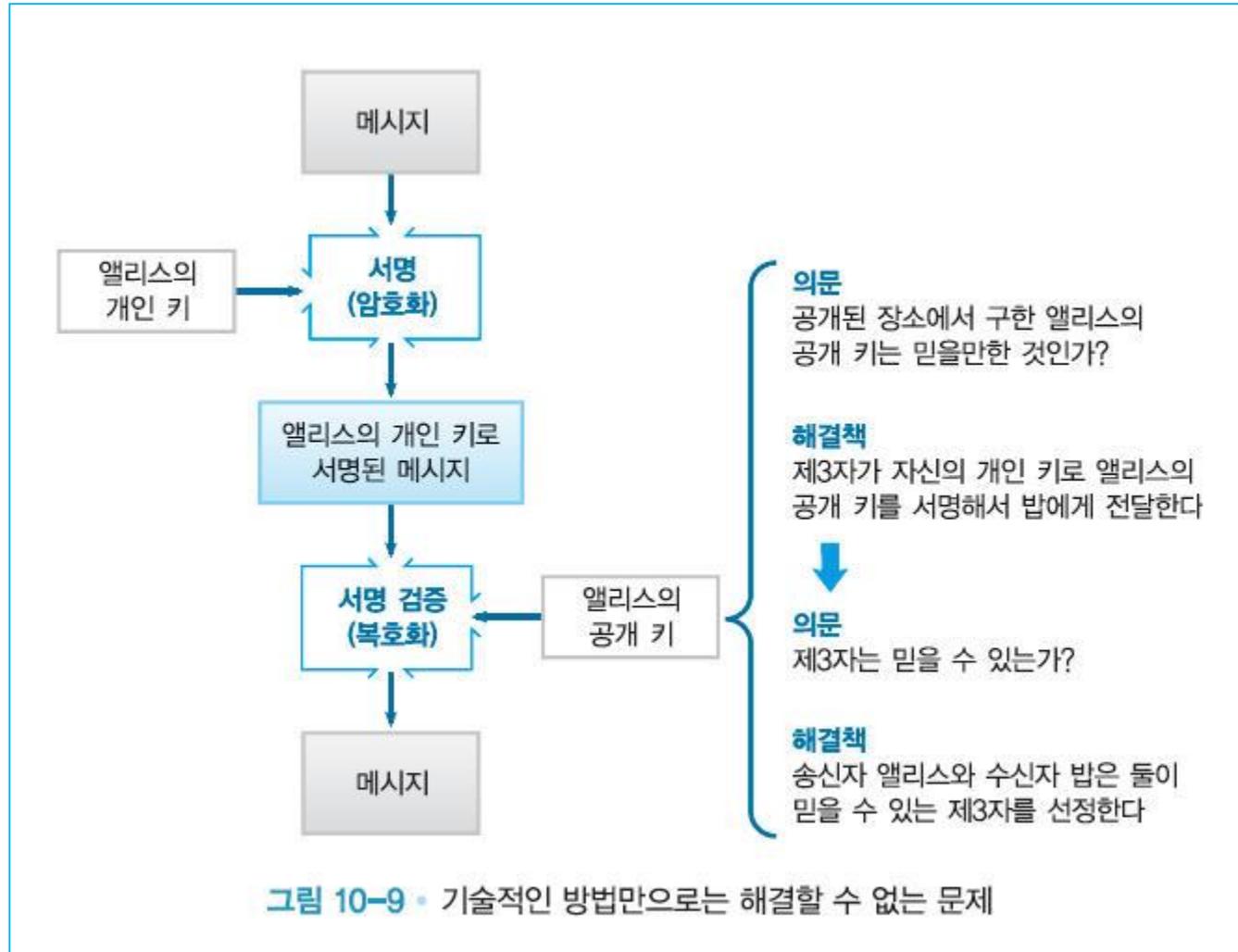
8.2 하이브리드 암호 시스템과 해시 값에 대한 디지털 서명

- 대칭 암호의 키는 기밀성의 엷센스이고,
- 일방향 해시 함수의 해시 값은 무결성의 엷센스

제9절 디지털 서명으로 해결할 수 없는 문제

- 조작되어 있지 않은 공개 키를 「거짓 행세」 하고 있지 않은 송신자로부터 받을 필요가 있다
- 인증서
 - 바른 공개 키를 입수하기 위해 고안된 것이 공개 키 기반 구조 (Public Key Infrastructure)
 - 공개 키 암호 및 디지털 서명의 기술을 사회적인 기반(기반구조)으로 만들어 가는 것

기술적인 방법만으로는 해결할 수 없는 문제



10절. 2019 신기술 전자서명 서비스 소개

구분	인증사업자 (서비스명)	서비스 개요
적용 중인 서비스	금융결제원 (브라우저인증서)	<ul style="list-style-type: none"> ○ 별도 프로그램 설치 없이 웹 표준(HTML5)을 지원하는 클라우드 연동 브라우저 인증 서비스 ○ 금융(10개 은행), 정부 및 공공기관(19개 사이트) 적용
	카카오페이 (카카오페이인증)	<ul style="list-style-type: none"> ○ 모바일 메신저 카카오톡 기반의 인증 서비스로 PKI 전자서명, 블록체인 및 생체인식 FIDO 기술 적용 ○ 국민연금공단, KB증권, BC카드, DB손해보험 등 적용
	한국전자인증 (클라우드사인)	<ul style="list-style-type: none"> ○ 클라우드기반의 생체인증 간편 전자서명서비스 ○ 국세청 홈텍스, 서울시e바로, KB손해보험 등 적용
	에티소프트 (VestPin)	<ul style="list-style-type: none"> ○ 순수 HTML5 기술 적용 간편인증/간편서명 솔루션 ○ 카드사(S,K,B 등) 및 보험사(K,M,H 등)에 적용
	아톤 (ATON mPKI)	<ul style="list-style-type: none"> ○ SW 보안매체(SE) 적용 간편 전자서명 솔루션 ○ IBK기업은행, 케이뱅크 및 PASS인증서 등에 적용
	한국정보인증 (싸인오케이)	<ul style="list-style-type: none"> ○ WebTrust/AATL 인증기반의 전용인증서 및 블록체인 기술을 적용한 온라인 전자계약서비스 ○ 연봉계약/대출약정서/렌탈/개인정보활용동의서 등에 적용

적용 예정인 서비스	위즈베라 (PINsign)	<ul style="list-style-type: none"> ○ PIN 기반 간편 전자서명(사설) 서비스 ○ KB모바일인증서(기적용) 및 의료분야(EMR) 적용 예정
	SKT (모바일 전자증명)	<ul style="list-style-type: none"> ○ 블록체인 및 전자서명 기술 활용 모바일 전자증명서비스 ○ 금융, 통신, 대학 등 전자증명 용도로 적용 예정
	시큐브 (시큐사인)	<ul style="list-style-type: none"> ○ 생체수기서명 인증 기술을 적용한 전자서명서비스 ○ 전자계약, 지불결제, 본인인증(신원확인) 분야에 적용 예정
	라온시큐어 (OmniOne)	<ul style="list-style-type: none"> ○ 블록체인 기술을 활용한 전자서명 인증서비스 ○ 병무청 민원포탈시스템 적용 예정
	코스콤 (OpenPass)	<ul style="list-style-type: none"> ○ 간편비밀번호, 생체, 패턴 등 편리한 인증서비스 ○ 공안사설인증, FIDO, OTP 등 다양한 통합인증서비스 제공
	아이콘루프 (my-ID)	<ul style="list-style-type: none"> ○ 블록체인 기술을 활용한 전자서명 인증서비스 ○ 증권사(9개), 은행(2개) 등 25개 사이트 적용예정

브라우저 및 클라우드 인증서비스 (금융결제원)

• 서비스개요

- (브라우저인증서) 별도 프로그램 설치 없이 웹 표준(HTML5)을 지원하는 브라우저에 공인인증서를 발급·저장하고 이용할 수 있는 서비스

• 개념도



《 브라우저 및 클라우드 인증서비스 주요 개선내용 》

구분		기존 인증서	브라우저인증서 (클라우드 연동)
이용환경	프로그램 설치	필요	불필요
	이용환경 전환 (PC↔모바일)	단말기 간의 인증서 복사	클라우드에 보관된 인증서 다운로드
유효기간		1년	3년
보안성		-	신뢰하는 이용기관*에서만 사용 가능

* 금융결제원에서 이용기관 등록관리

카카오페이 인증 서비스 (카카오페이)

• 서비스개요

- 전국민이 사용하는 모바일 메신저 카카오톡 기반의 인증 서비스
- PKI 전자서명 기술 및 블록체인 기술 활용해 보안성 강화
- 생체인식 국제 표준인 FIDO 기술 적용

• 개념도

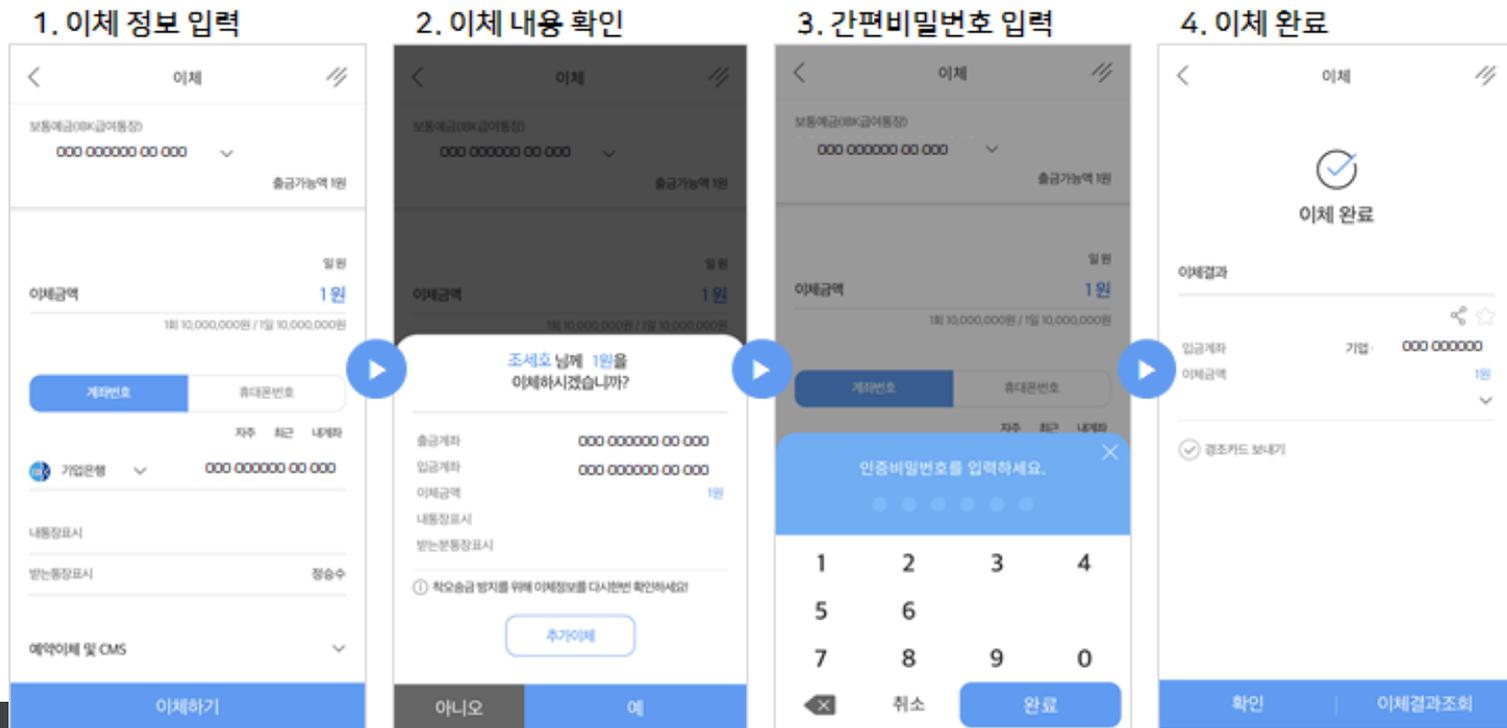


IBK기업은행 i-ONE뱅크에 적용된 간편 전자서명 솔루션 (아톤)

• 서비스개요

- IBK기업은행의 모바일 뱅킹 어플리케이션 i-ONE뱅크에 적용된 간편 전자서명 솔루션
- 기존의 공인인증서 및 추가 인증수단 등을 대체하여 6자리 비밀번호로 간편하게 전자서명이 가능한 서비스 구현

• 개념도

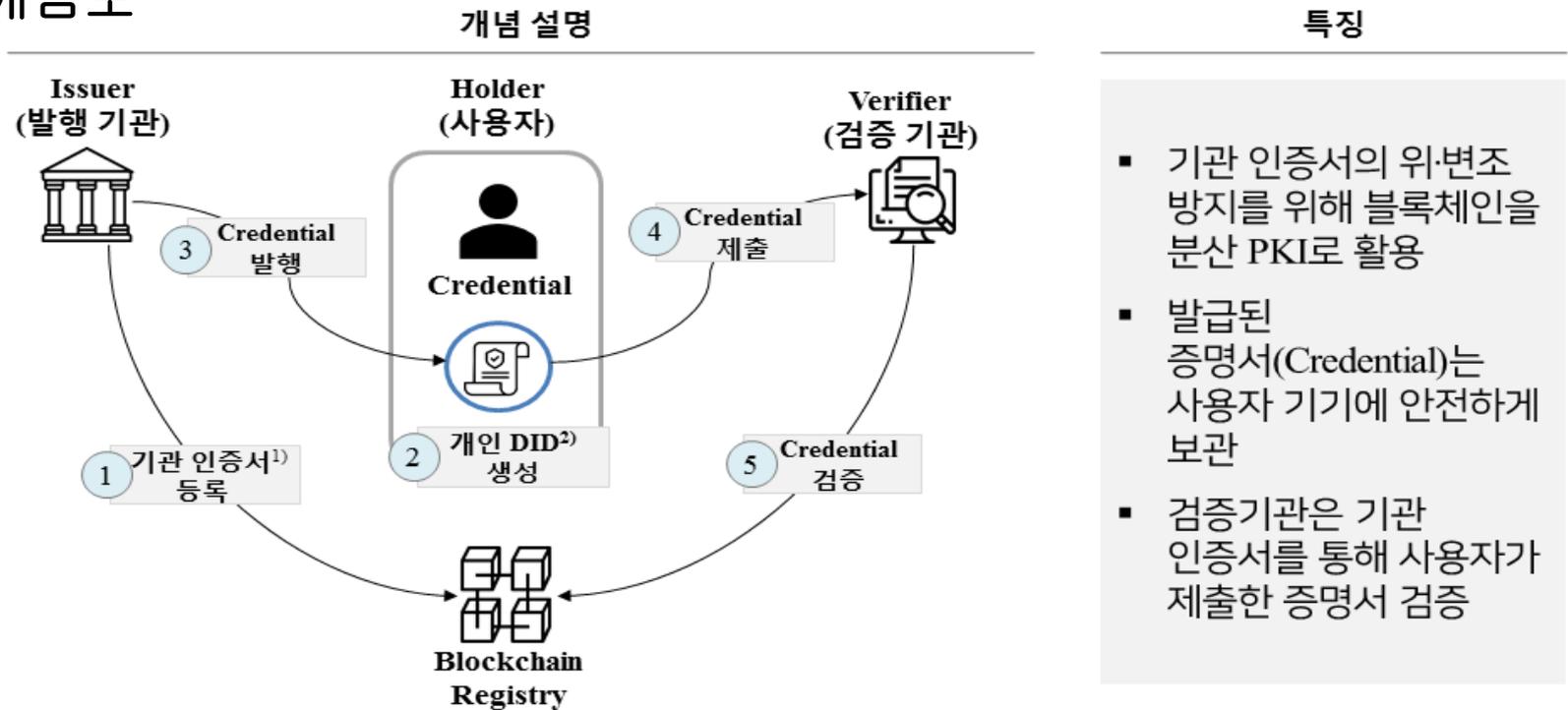


블록체인 기반 자기주권형 모바일 전자증명 서비스 (SKT)

• 서비스개요

- 블록체인 기반 DID(Decentralized Identity) 및 전자서명 기술을 통해 개인이 모바일 앱을 이용해 기관으로부터 검증 가능한 증명서를(Verifiable Credential) 직접 발급받고 제출할 수 있는 자기주권형(Self-Sovereign) 지갑 서비스

• 개념도



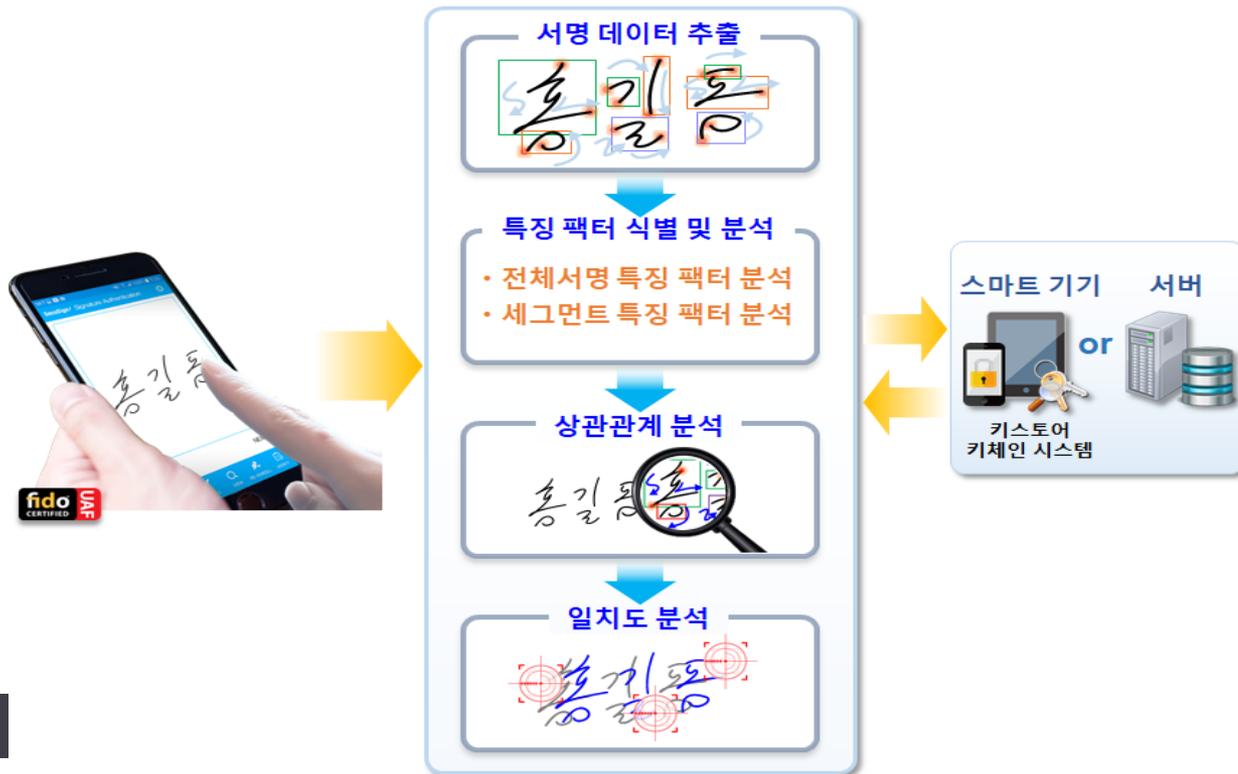
1) 기관 인증서는 해당 기관의 DID 값, 공개키, 서비스 접근 주소 등으로 구성 2) 개인DID는 연동 기관별로 생성

생체수기서명 인증 기반의 전자서명 (시큐브)

• 서비스개요

- 시큐브의 독자적인 생체수기서명 인증 기술인 ‘시큐사인’을 적용한 전자서명 서비스
 - ▶ 시큐사인(SecuSign): 서명과 연관된 개인의 고유한 행위특징 정보를 인식 및 분석하여 서명자의 신원을 확인하는 행위특징 기반 생체인증
- 본인인증(신원 확인), 전자문서(전자계약), 온라인뱅킹, 간편결제, 핀테크 등 다양한 서비스 분야에 적용 가능

• 개념도



참고문헌

- “알기 쉬운 정보보호 개론”, 히로시 유키 저(이재광 외 2 공역, 인피니티 북스, 2017 - 주교재

참고자료

- “네트워크 보안 에센셜 3판”, 윌리엄 스톨링스 저(전태일 등역), 교보문고
- "최신 전자서명 기술·서비스 선보인다", 신기술 전자서명 우수사례 설명회, 과기정통부, 2019. 8

Thank You!

Q & A