1. 전자 상거래의 이해

전자 상거래의 시작

- 본격적인 전자 상거래의 시작
  - 1994년에 피자헛이 처음으로 웹 사이트를 통해 주문을 받은 것이 인터넷을 통한 본격적인 전자상거래의 시작
  - 이 시기에 만들어진 넷스케이프 1.0은 SSL(Secure Sockets Layer) 암호화로 안전한 거래를 제공
  - 1995년에는 제프 베저스가 세계 최초의 가상 서점 아마존 설립

1994년의 피자헛 주문 사이트

제프 베저스와 1994년의 아마존 사이트

1. 전자 상거래의 이해

■ 전자 상거래의 보안 요건

▪ 전자 상거래 공격 유형

- **인증(Authentication) 공격**
  - 네트워크로 접근한 사용자가 적절치 않은 인증으로 다른 사용자로 위장하는 것
  - 가짜 은행 사이트를 만들어 은행 이용자의 공인인증서 정보를 획득한 뒤 악용하는 사례

- **송수신 부인(Repudiation) 공격**
  - 부인방지(Non-repudiation): 어떤 행위나 사건이 일어났다는 사실을 증명하여 나중에 그 사실을 부인하지 못하도록 함
  - 네트워크를 통해 수행한 인증 및 거래 내역을 부인하는 것
  - 계좌이체나 신용카드로 지불을 받고도 받지 않았다고 부인하거나 소매점으로부터 상품을 받은 후 받지 않았다고 부인하는 사례

- **기밀성(Confidentiality) 공격**
  - 기밀: 아주 중요한 비밀 – 정보 자체
  - 기밀성: **정당하지 않은 사용자나 시스템에 대해서 정보가 노출되지 않게 하는 특성.** 이는 승인된 사람이나 시스템만이 민감한 데이터를 열람할 수 있도록 보장
  - 네트워크로 전달되는 인증 정보 및 주요 거래 정보가 유출되는 것
  - 전자결제를 할 때 신용카드 번호 정보가 유출되어 악용되는 경우

- **무결성(Integrity)에 대한 공격**
  - 네트워크 도중에 거래 정보 등이 변조되는 것
  - 온라인 계좌이체 등을 이용한 전자결제 시 수신 계좌나 금액 등을 변조하는 사례

1. 전자 상거래의 이해

전자 상거래의 보안 요건

- 전자상거래가 성공하기 위한 보안 요건
  - 신분 확인 수단 제공
    - 원격의 거래 상대를 신뢰할 수 없기 때문에 네트워크에서 상대방이나 자신에 대한 신분 확인 수단이 필요
  - 제삼자의 중재
    - 거래 사실(거래 내역)을 공증할 수 있는 신뢰할 만한 제삼자의 중재 필요
  - 지불 방식의 안전성
    - 전자지불 방식(과정)의 안전성을 보장하는 방법이 확보되어야 함
  - 블록체인을 활용하는 비트코인과 같은 거래 체계가 활성화된다면 전자 상거래의 세 가지 보안 요건 중 '제삼자의 중재'는 앞으로 완전히 사라질 수 있음

비트코인

2. 공개 키 기반 구조

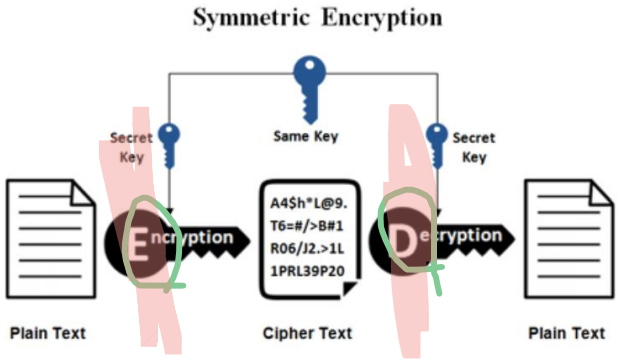
■ 공개 키 기반 구조의 개념

- 공개 키 기반 구조 (PKI)
  - 메시지의 암호화 및 전자 서명을 제공하는 복합적인 보안 시스템 환경
  - 공개 키 기반 구조는 '인터넷에서 신분증을 검증해주는 관청'의 역할
  - 가장 가까운 관청인 주민센터가 있고 그 위에 구청, 시청이 있으며 맨 위에 정부가 있는 것과 마찬가지
  - 공개 키 기반 구조에 속하는 사람은 어디서든지 자신의 인터넷상 신분을 **인증 기관(CA)**에서 **공인인증서(인터넷 신분증)**를 이용하여 증명 가능

주민센터에서의 신분 확인을 위한 신분증 제시

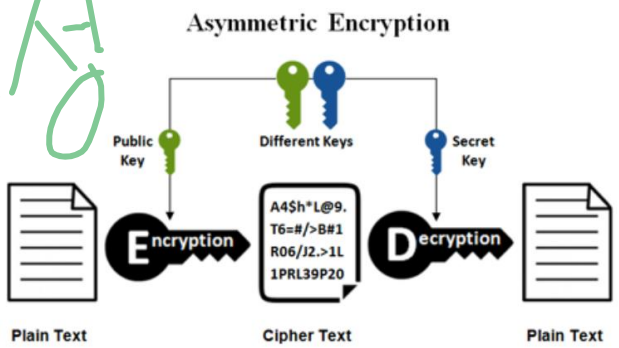
대칭 vs 비대칭(공개키) 암호화 방식

- **대칭(Symmetric)**
  - 암·복호화에 사용하는 키가 동일함
  - 기밀성을 제공하나, 무결성/인증/부인방지를 보장하지 않음
  - 대표적 알고리즘 : **SEED**, DES, 3DES, AES, ARIA, 최근 주목받고 있는 암호인 ChaCha20



use Hash to 무결성

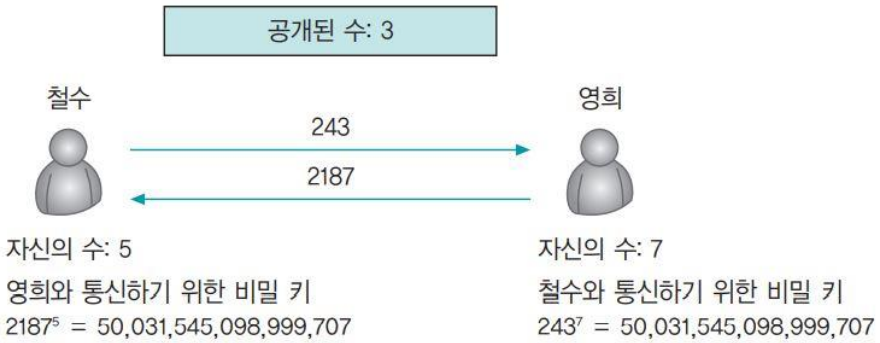
- **비대칭(Asymmetric)**
  - 암・복호화에 사용하는 키(비밀키, 공개키)가 서로 다름
  - 키분배 필요 X, **기밀성/인증/부인방지 기능 제공**
  - 대표적 알고리즘: **Diffie Hellman, RSA, DSA**



비대칭 암호화 방식

비대칭 암호화 방식의 발견

- 키 전달과 관련한 아이디어는 휫필드 디피와 마틴 헬먼이 처음 발표
- 비대칭 암호화 알고리즘 아이디어는 20세기 암호학의 혁명으로 불렸지만 실제로 사용하기엔 약점이 많았음



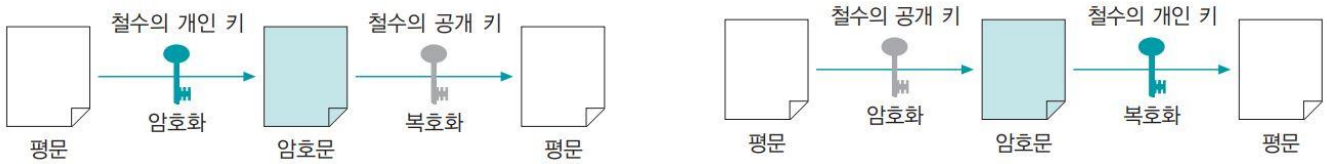
- 철수는 자신이 정한 수 5(개인키)를 사용하여 3(공개된 수, 공개키)의 5제곱인 243(3^5)을 영희에게 전송
- 영희도 자신의 수를 7(개인키)로 정하고 3(공개된 수, 공개키)의 7제곱인 2187(3^7)을 철수에게 전송
- 철수 (2187^5) 와 영희 (243^7) 는 상대에게서 받은 수에 자신의 수를 적용하여 제곱
- 둘은 자신이 정한 5와 7이라는 수를 상대방에게 전달하지 않아도 50,031,545,098,999,707이라는 같은 키를 공유

비대칭 암호화 방식

비대칭 암호화의 구조

- 비대칭 암호화 알고리즘은 RSA(Rivest-Shamir-Adleman) 알고리즘이 나오면서 정립
- 각 개인이 공개 키와 개인 키를 소유하는 구조지만 서로의 개인 키는 얻을 수 없음

- 대칭 암호화 알고리즘과 달리 메시지의 암호화와 복호화가 같은 키로 이루어지지 않음
- 언제나 한 쌍의 개인 키와 공개 키로 암호화와 복호화가 이루어짐

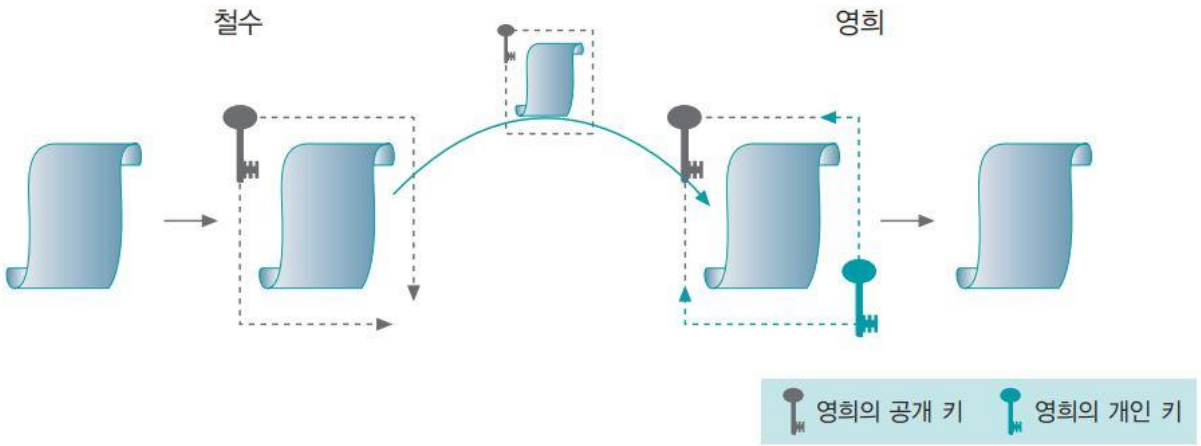


비대칭 암호화 방식

■ 비대칭 암호화의 기능

▪ 기밀성

- 비대칭 암호화 알고리즘은 대칭 암호화 알고리즘보다 더 엄밀한 기밀성을 제공
  - 철수는 전화번호부에서 전화번호를 찾듯이 영희의 공개 키를 구함
  - 편지를 암호화한 후 공개키를 이용하여 전송
  - 영희는 자신이 가진 개인 키로 철수의 편지를 복호화하여 읽을 수 있음
  - 민수가 중간에서 편지를 가로채더라도 영희의 공개 키로 암호화한 편지를 민수의 개인 키로는 복호화 불가



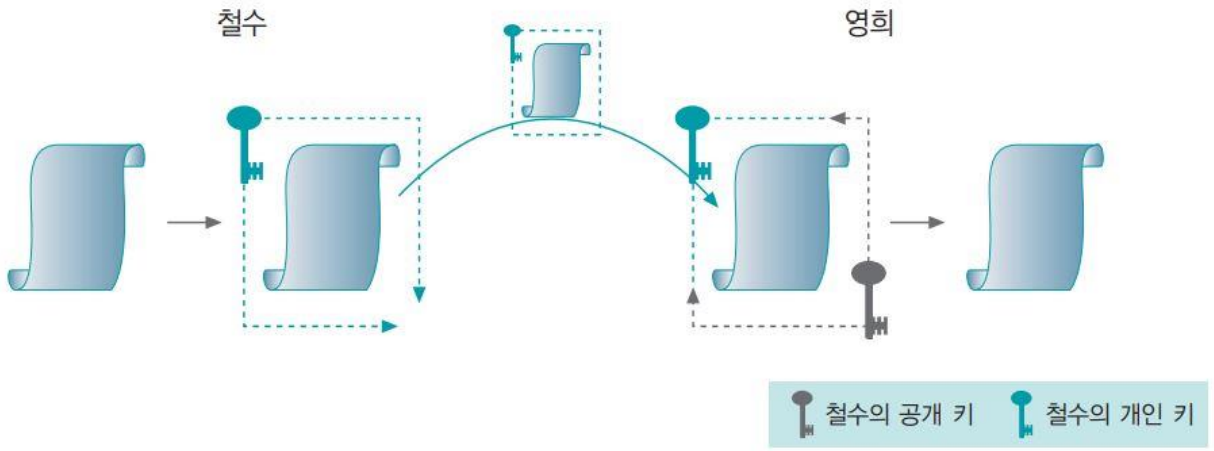
기밀성을 확보하기 위해 공개 키로 암호화하기

비대칭 암호화 방식

■ 비대칭 암호화의 기능

▪ 부인 방지

- 대칭 암호화 알고리즘에는 없는 기능으로 쉽게 말하면 '발뺌 방지'
  - 철수의 개인 키로 암호화된 편지는 철수의 공개 키로만 열 수 있음
  - 철수의 개인 키는 철수만 가지고 있으므로, 영희는 받은 편지가 철수의 공개 키로 풀려야만 그 편지는 철수가 보낸 편지라고 확신할 수 있음
  - 영화나 소설에서 두 사람이 어쩔 수 없이 헤어져야 할 때 훗날을 위한 증표로 장신구를 건네는 것과 유사



부인 방지 기능을 확보하기 위해 개인 키로 암호화하기

비대칭 암호화 방식

비대칭 암호화의 기능

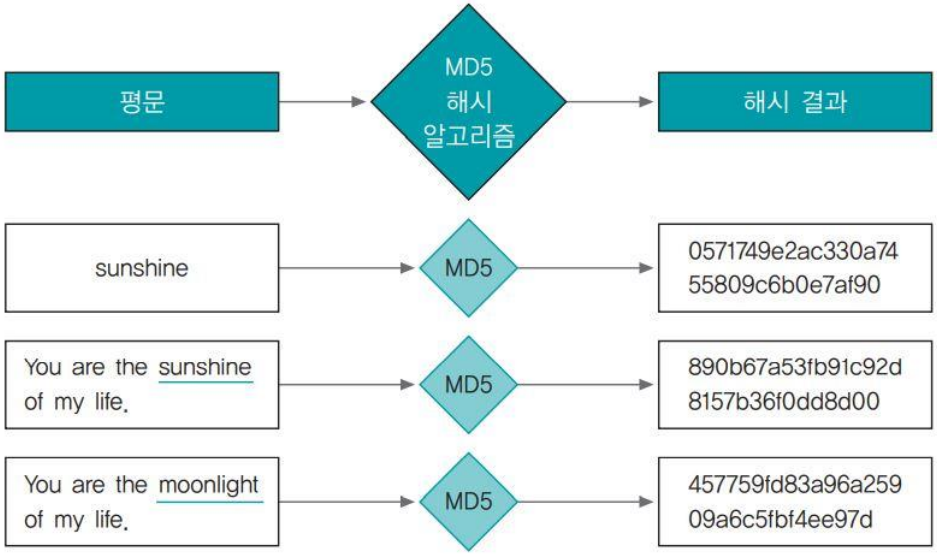
- 암호화 키와 관련된 용어
  - 대칭 키: 암호화할 때 쓰는 키와 복호화할 때 쓰는 키가 같은 것
  - 비밀 키: 암호화할 때와 복호화할 때 사용되는 키가 같으므로 암호문이 효력을 발휘하려면 발신자와 수신자 사이의 키에 대한 정보가 비밀로 유지
  - 비대칭 키: 암호화할 때 쓰는 키와 복호화할 때 쓰는 키가 다른 것 (공개 키와 개인 키를 묶어 비대칭 키)
  - 공개 키와 개인 키: 발신자와 수신자가 각각 한 쌍을 소유

해시(Hash)

해시의 특징

- 하나의 문자열을 더 짧은 길이의 값이나 키로 변환하는 것.
- 정보의 위조·변조를 확인하기 위한 것, 즉 정보의 무결성을 확인하기 위한 것
- 해시를 사용하여 전자서명, 전자봉투, 전자화폐 등 다양한 전자상거래 기능 구현 가능
- 대표적인 해시 알고리즘은 MD5

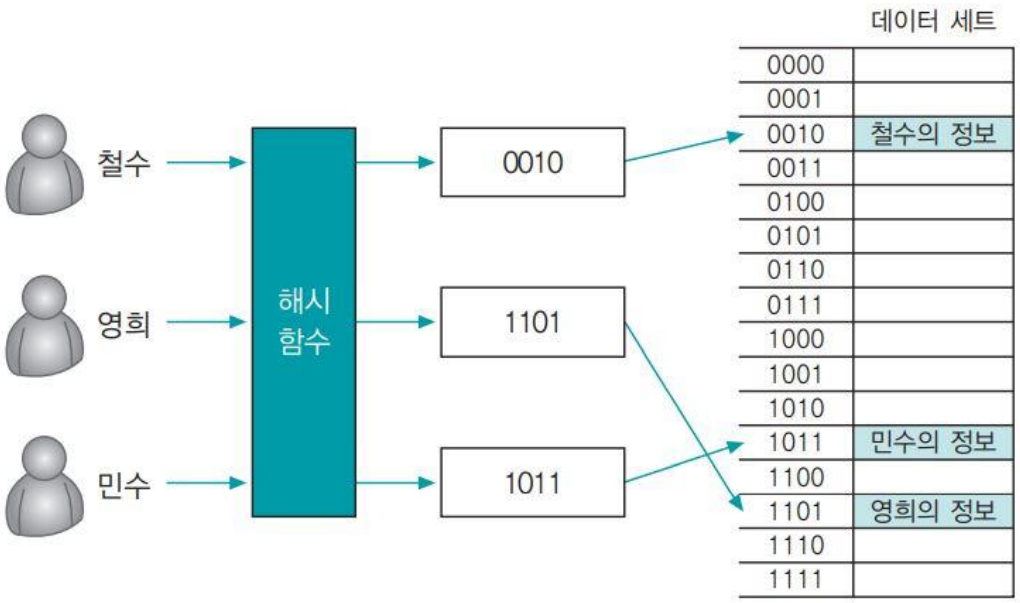
- 세 평문은 길이가 각각 다르지만 해시 결과는 32개 문자로 길이가 모두 같음
- 두 번째와 세 번째 평문은 단어 하나만 다를 뿐인데 해시 결과는 완전히 다름
  - 해시되기 전의 값을 해시 값으로 추측하기가 불가능하다는 특징 때문에 일어난 결과
  - 충돌: 다른 값의 데이터를 입력하더라도 해시 결과 값이 같을 수 있는 상황



해시

해시의 역할

- 원래 해시는 데이터베이스의 탐색을 효과적으로 구현하기 위해 만들어진 것
- 보안에서는 해시가 완전히 똑같은 데이터만 해시 값이 같고 조금만 달라도 해시 값이 전혀 다르다는 점을 이용하여 데이터가 임의로 변경되지 않았다는 데이터 무결성을 확인하기 위한 도구로 사용

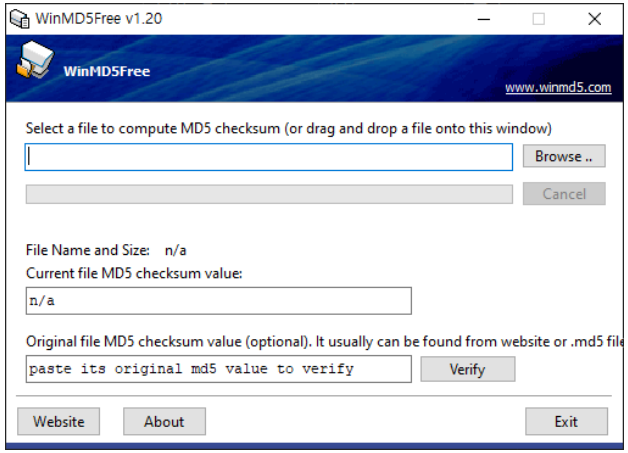


데이터베이스에서 해시 값을 통한 값의 참조

해시

해시 검증

① WinMD5Free 홈페이지 접속 & 파일 다운로드(https://www.winmd5.com/)



Download WinMD5 (only 249KB):

WinMD5 Freeware Download

WinMD5Free.zip MD5: 73f48840b60ab6da68b03acd322445ee

WinMD5Free.exe MD5: 944a1e869969dd8a4b64ca5e6ebc209a

You may simply download it, then unzip and put the exe to any fol

② winMD5.exe 파일을 실행하여 md5sum 검사 수행 - Browse 기능을 통해 다운로드 받은 파일 경로를 입력하면, 해당 파일의 md5 체크값(checksum)이 생성됨

③ 다운로드 한 파일이 전송 중에 변조되었는 가를 확인

- 다운로드 사이트에서 제공된 해시 값을 하단에 입력하고 'Verify'를 눌러주면 해당 파일의 변조 여부를 알 수 있음.

해시

해시의 종류

- MD 알고리즘
  - 로널드 리베스트가 공개 키 기반 구조를 만들기 위해 RSA와 함께 개발한 것으로 MD2, MD4, MD5가 있음
  - 1989년에 개발된 MD2는 8비트 컴퓨터에 최적화
  - 1990년에 개발된 MD4와 1991년에 개발된 MD5는 32비트 컴퓨터에 최적화
  - MD5 알고리즘은 MD4의 확장판으로 MD4보다 속도가 빠르지는 않지만 데이터 보안성이 더 뛰어남
- SHA
  - 160비트의 값을 생성하는 해시 함수로, MD4가 발전한 형태
  - MD5보다 조금 느리지만 좀 더 안전하다고 알려져 있으며 SHA에 입력하는 데이터는 512비트 크기의 블록임
  - SHA 알고리즘은 크게 SHA-1과 SHA-2로 나눌 수 있음

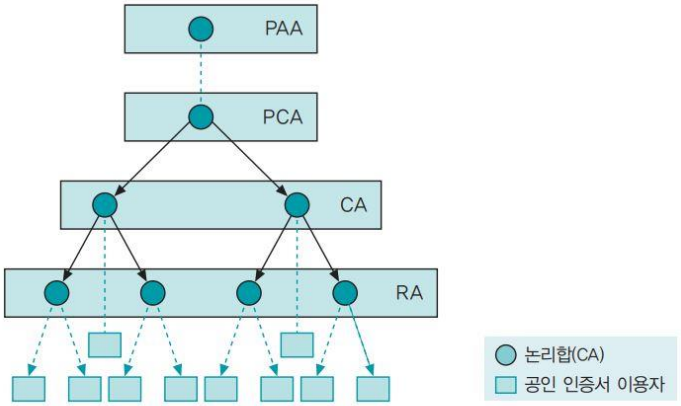
알고리즘	메시지 문자 크기	블록 크기	해시 결과 값 길이	해시 강도
SHA-1	$<2^{64}$	512비트	160비트	0.625
SHA-256	$<2^{64}$	512비트	256비트	1
SHA-384	$<2^{128}$	1024비트	384비트	1.5
SHA-512	$<2^{128}$	1024비트	512비트	2

2. 공개 키 기반 구조

공개 키 기반 구조의 개념

트리형 공개 키 기반 구조

- 공개 키 기반 구조가 되려면 인증 정보를 일원화하여 호환성을 갖춤으로써 개인이 쉽게 접근할 수 있어야 함
- 순수 계층 구조: 트리형으로 구성된 공개 키 기반 구조



트리형 공개 키 기반 구조

- PAA(Policy Approving Authority): 정책 승인 기관으로 공인인증서 정책을 결정하고 하위 기관 정책을 승인. **정부 기관**
- PCA(Policy Certifying Authority): 정책 인증 기관으로 Root CA 인증서를 발급하고 기본 정책을 수립. **국가 인증 기관**
- CA(Certification Authority): PCA의 하위 기관인 인증 기관으로 인증서 발급과 취소 등의 실질적인 업무 담당. **한국인터넷진흥원(KISA), 한국전자인증**
- RA(Registration Authority): 등록 기관으로 인증서를 신청하는 사람이나 기관의 신원을 직접 확인하고, 그 정보가 정확한지 검증해서 CA에게 인증서 발급을 요청. **은행**

2. 공개 키 기반 구조

공인 인증서

공인인증서의 개념

- 공개 키와 공개 키의 소유자를 연결해주는 전자문서.
- 오늘날 사용하는 대부분의 공인인증서는 X.509 인증서(버전 3)를 표준으로 따름
- SPK I인증서, PGP 인증서가 있음

공인인증서의 특성

- 누구나 사용자의 공인 인증서와 공개 키를 획득할 수 있음
- 인증 기관 이외에는 공인 인증서를 수정 및 발급할 수 없음
- 같은 인증 구조 내의 사용자 간에는 상호 인증의 신뢰가 가능

공인인증서의 구성

① 버전: 공인 인증서의 형식을 구분
(우리가 사용하는 대부분의 공인 인증서는 버전 3)

② 일련 번호: 공인 인증서를 발급한 인증 기관 내의 인증서 일련번호

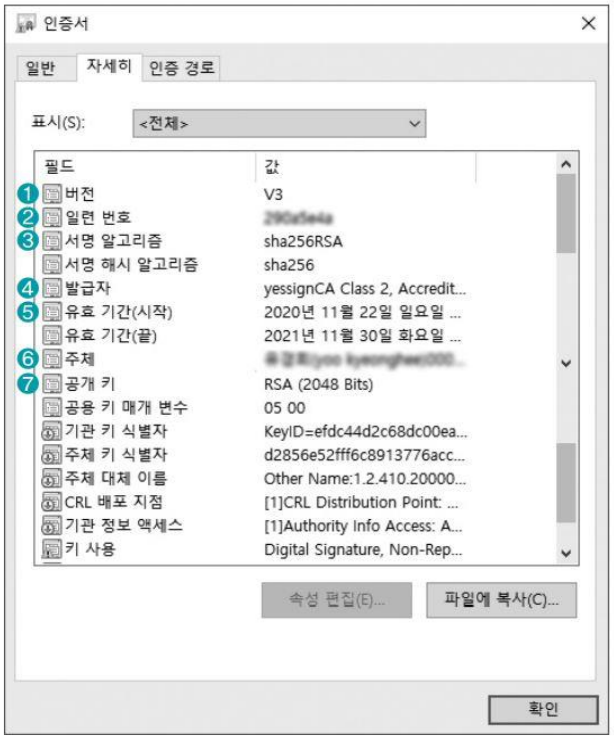
③ 서명 알고리즘: 공인 인증서를 발급할 때 사용한 알고리즘

④ 발급자: 공인 인증서를 발급한 인증 기관의 DN(Distinguished Name; 고유 이름)

⑤ 유효 기간: 공인 인증서를 사용할 수 있는 시작일과 만료일로 초 단위까지 표기

⑥ 주체: 공인 인증서 소유자의 DN

⑦ 공개 키: 공인 인증서의 모든 영역을 해시하여 인증 기관의 개인 키로 서명한 값



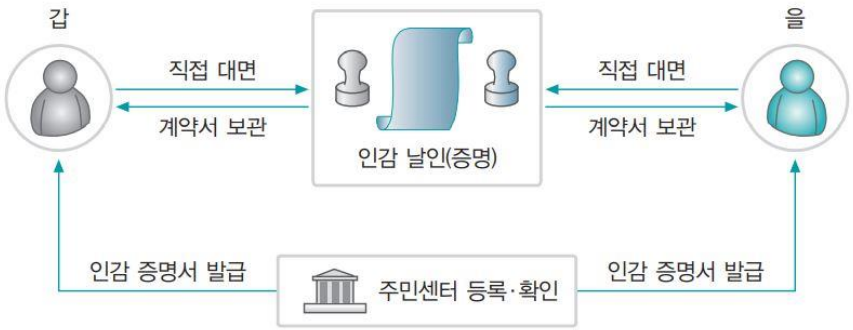
공인 인증서 구성(ch09-09_new)

3. 전자 서명과 전자 봉투

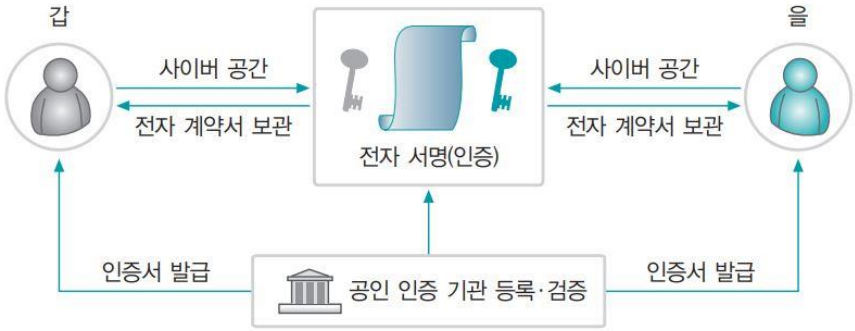
■ 전자서명

▪ 전자 서명의 정의

- 전자서명: 서명자가 해당 전자 문서에 서명하였음을 나타내기 위해 전자 문서에 첨부되거나 논리적으로 결합된 전자적 형태의 정보
- 인감도장처럼 전자서명도 공인된 인증 기관에 등록 및 검증하여 사용 가능



인감도장을 사용한 계약서 날인



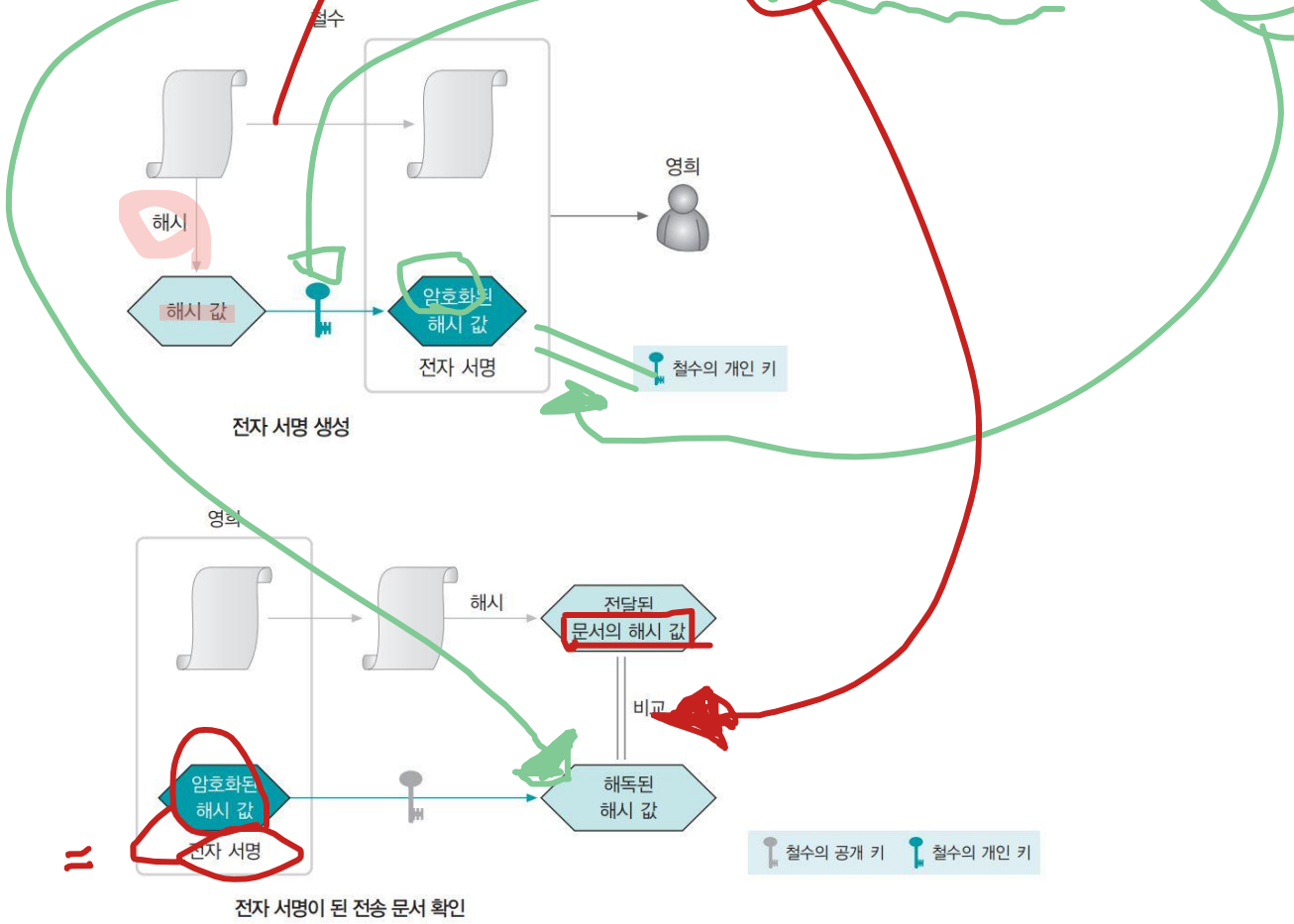
전자 서명을 사용한 인증

3. 전자 서명과 전자 봉투

전자서명

전자서명 구현 원리

- 전자서명은 원본의 해시 값을 구한 뒤 부인 방지 기능을 부여하기 위해 **공개 키 방법(개인 키)**을 사용
- 복호화한 해시 값**(공개 키 사용)**과 편지에서 구한 해시 값이 일치하면 **위조 되지 않았다고 확신(무결성)**할 수 있음



3. 전자 서명과 전자 봉투

전자서명

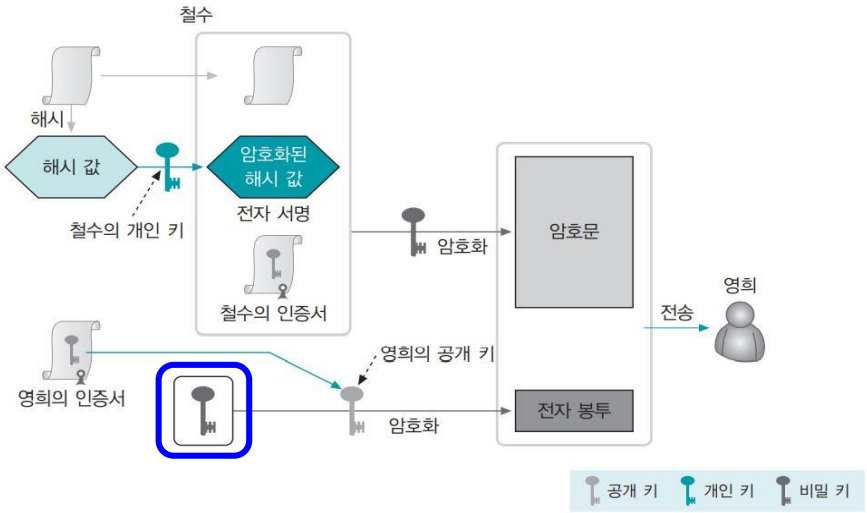
- 전자 서명이 제공하는 기능
  - 변경 불가
    - 서명된 문서는 내용을 변경할 수 없기 때문에 데이터가 변조되지 않았음을 보장하는 **무결성**을 만족
  - 부인 방지
    - 서명자가 서명한 사실을 나중에 **부인할 수 없음**

- 전자 서명의 사용 예
  - 부동산 계약: 집 사고팔거나 전월세 계약할 때도 공인된 전자계약 시스템에서 전자서명을 사용
  - 학교/회사 내부 문서: 출석부, 보고서 승인, 휴가 신청 등 내부 시스템에서 결재하거나 확인할 때 전자서명
  - 은행 업무: 은행 창구에서 종이 대신 태블릿에 전자서명

3. 전자 서명과 전자 봉투

전자 봉투

- 전달하려는 메시지를 암호화하여 한 사람을 통해 보내고 암호화 키는 다른 사람이 가져가도록 암호학적으로 구현
- **대용량 데이터를 빠르게 비밀키로 암호화**하고, 그 데이터를 푸는 열쇠인 철수의 **비밀키(대칭키)를 안전하게 영희의 공개키로 암호화하여 전달**



전자 봉투를 이용한 암호 전송

① 철수는 전자 봉투를 사용하기 위해 먼저 전자 서명을 생성. 원문으로 부터 해싯값을 생성한 후 **개인 키(비대칭 키)**를 사용하여 해싯값을 암호화

② 전자 서명과 원문, 자신의 공개 키가 들어 있는 인증서를 철수의 비밀 키(**DES 알고리즘 등에 사용되는 대칭 키**)로 암호화

- **철수의 인증서: 인증서 안에 공개 키와 함께 철수의 정보(철수의 정보, 인증서 발급기관(인증 기관) 정보, 인증 기관(CA)자신의 개인키로 이 인증서 전체를 암호화한 전자서명 등)가 같이 들어있음**
- 이 서명을 통해 "이 인증서에 담긴 ' 철수의 정보 ' 와 ' 공개 키 ' 는 철수의 것이 맞고, 중간에 위·변조되지 않았음을 보증

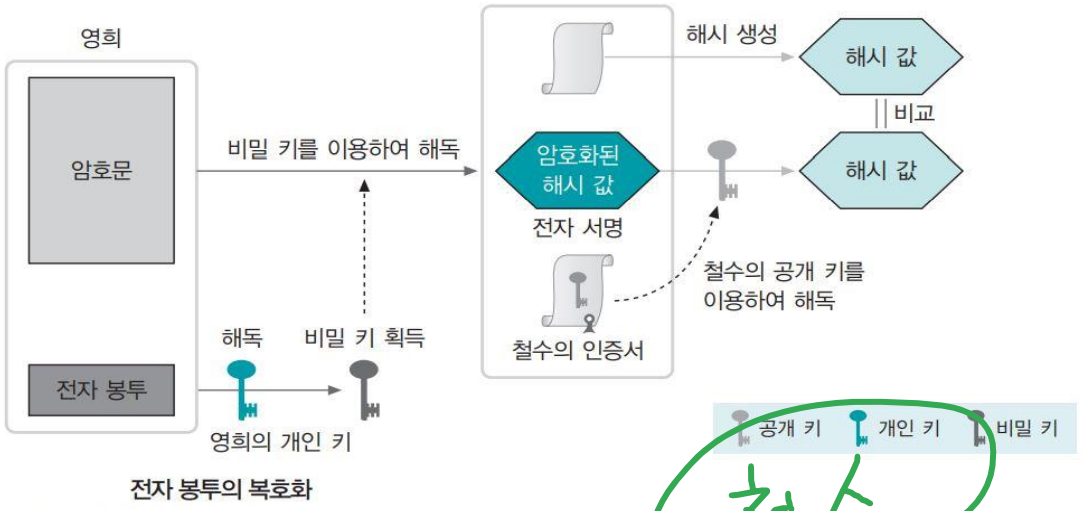
③ 영희의 인증서, 철수의 비밀 키가 영희의 **공개 키(비대칭 키)**로 암호화

④ 최종적으로 철수는 비밀 키로 암호화한 결과와 비밀 키가 암호화된 전자 봉투를 영희에게 전송

3. 전자 서명과 전자 봉투

■ 전자 봉투

- 전자봉투는 기밀성, 무결성, 부인 방지를 모두 지원



전자 봉투의 복호화

- 전달받은 영희는 전자 봉투를 자신의 개인 키로 복호화하여 비밀 키를 획득
- 비밀 키를 이용하여 전자 서명과 편지, 철수의 인증서를 복호화(해독)
- 복호화한 인증서에서 철수의 공개 키를 얻어 전자 서명을 복호화하고 이를 편지의 해시 결과와 비교

3. 전자 서명과 전자 봉투

전자 봉투 사용

- **중요한 데이터나 파일 안전하게 주고받기**
  - 인터넷으로 계약서 파일, 개인 정보가 담긴 서류, 회사 기밀 자료 같은 크기가 크거나 아주 중요한 정보를 보낼 때 전자봉투 방식을 사용하여 암호화
  - 받는 사람의 공개키로 비밀키를 암호화하므로 오직 받는 사람만 자기 개인키로 그 비밀키를 꺼내서 데이터를 복호화
- **암호화된 파일 시스템이나 스토리지**
  - 사용자의 컴퓨터나 클라우드 저장소에 파일을 저장할 때, 파일 자체는 빠른 대칭키로 암호화해두고, 그 대칭키는 사용자의 공개키로 암호화해서 저장
  - 사용자 개인키가 있어야만 암호화된 대칭키를 복호화하고 파일에 접근할 수 있어서 다른 사람이 함부로 사용자의 파일을 열어볼 수 없음.
- **디지털 콘텐츠 보안 (DRM)**
  - 온라인으로 영화나 음악 같은 디지털 콘텐츠를 판매하거나 제공할 때, 콘텐츠 파일 자체는 암호화해두고, 컨텐츠를 볼 수 있는 ' 열쇠 ' 인 비밀키는 구매한 사용자의 공개키로 암호화해서 함께 전송하는 방식
  - 콘텐츠를 구매한 사람만 자신의 개인키로 비밀키를 복호화해서 콘텐츠를 볼 수 있게 함

4. 전자 결제

SET

- 1996년 비자와 마스터카드의 합의로 만들어진 프로토콜
- SET 방식의 **이중 서명**은 신용카드 거래에서 사실상의 표준
- SET은 보안의 주요 요소를 모두 실현
  - 개인정보 보호
  - 기밀성 보장
  - 무결성 보장
  - 부인 방지

SET 구성

① 신용카드 사용자: 신용카드를 소지한 사람. SET에 이용하는 공인 인증서를 소유

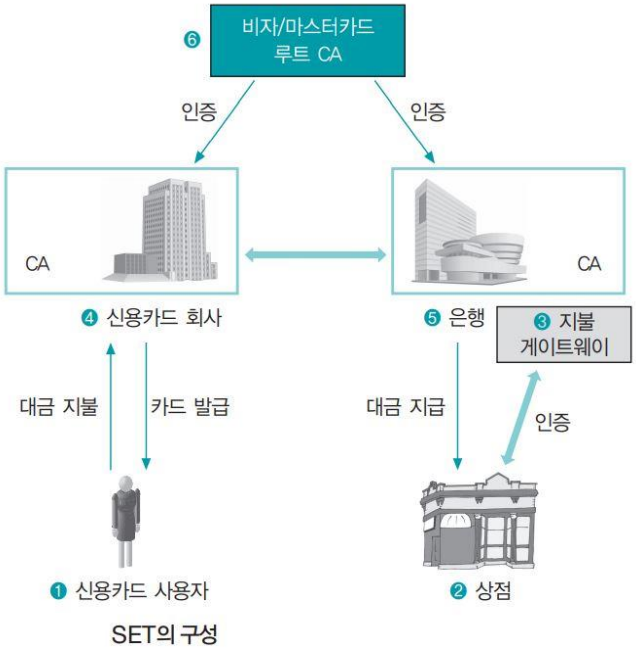
② 상점: 인터넷 쇼핑몰을 운영하며 SET를 이용하여 상품을 판매

③ 지불 게이트웨이(PG): 기존의 신용카드 지불 방식으로 은행과 거래 내역을 주고받음

④ 신용카드 회사: 사용자에게 신용카드를 발급하고, CA(Certification Authority)를 운영하여 사용자에게 공인 인증서를 발급

⑤ 은행: 상점의 계좌가 있는 곳으로 지불 게이트웨이를 운영하고, CA를 운영하여 상점에 공인 인증서를 발급

⑥ 인증 기관: SET에 참여하는 모든 구성원의 정당성을 보장하는 루트 CA



SET의 구성

4. 전자 결제

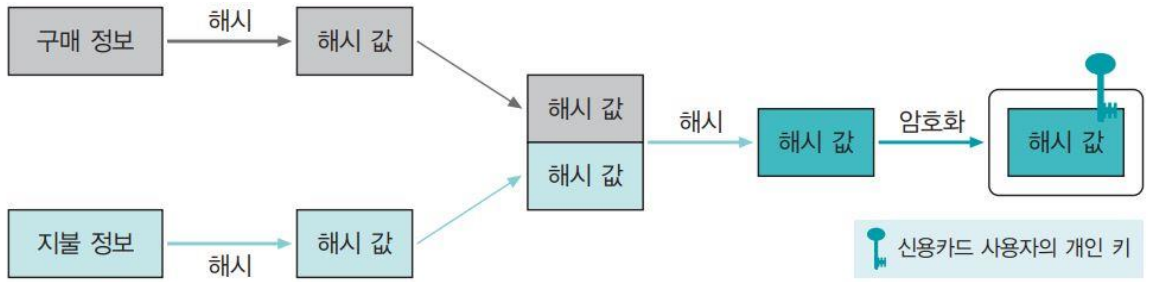
SET

SET 지불 과정

- 신용카드 사용자가 SET를 이용하여 상점에 결제 의뢰
- 주문서를 받은 판매자는 고객의 신용카드 회사에서 신용카드의 유효성 여부 확인
- 신용카드가 정상임을 확인하면 주문 확인 메시지를 고객에게 전송하고 고객은 자신의 신용카드 정보를 판매자에게 전송
- 판매자는 고객에게 받은 정보를 신용카드 결제에 다시 이용, 이때 SET는 **전자 봉투와 이중 서명을 사용**

이중 서명(Dual Signature)

- 신용카드 사용자의 구매 정보(구매 상품/수량, 거래 금액/날짜/시간, 가맹점 정보 등)와 지불 정보(신용카드 번호, 가드 소유자, 카드사 정보, 지불 방식(일시불/할부 등) 등)를 각각 해시한 후 두 값을 합하여 다시 해시함
- 최종 해시 값을 신용카드 사용자의 개인 키로 암호화(서명)하면 이중 서명 값 생성
- 상점에 대금을 지불하는 은행은 신용카드 사용자가 구입한 물건을 모르지만 상점이 요구한 결제 대금이 정확한지 확인 할 수 있게 하기 위한 목적



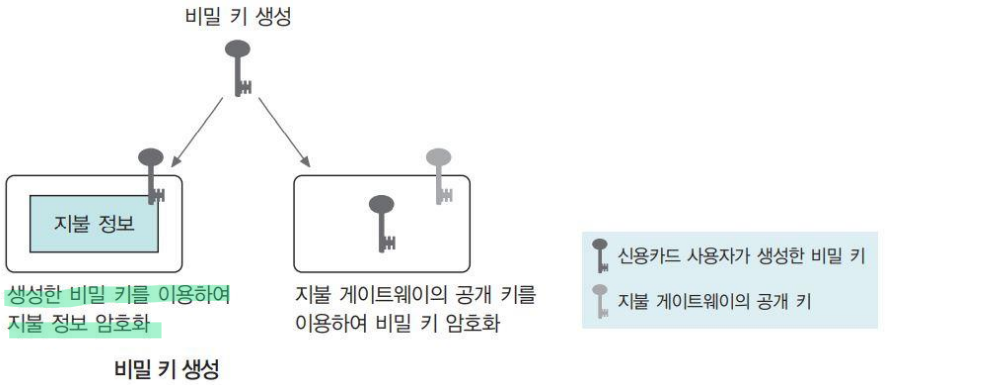
이중 서명의 기본 동작

4. 전자 결제

SET

이중 서명의 원리

- 신용카드 사용자는 하나의 **비밀 키(대칭 키)**를 생성
- 비밀 키를 사용하여 지불 정보를 암호화
- 비밀 키는 은행이 운영하는 지불 게이트웨이의 **공개 키**로 암호화
- 신용카드 사용자는 결제를 위한 데이터를 모두 생성하여 상점에 전송



비밀 키 생성

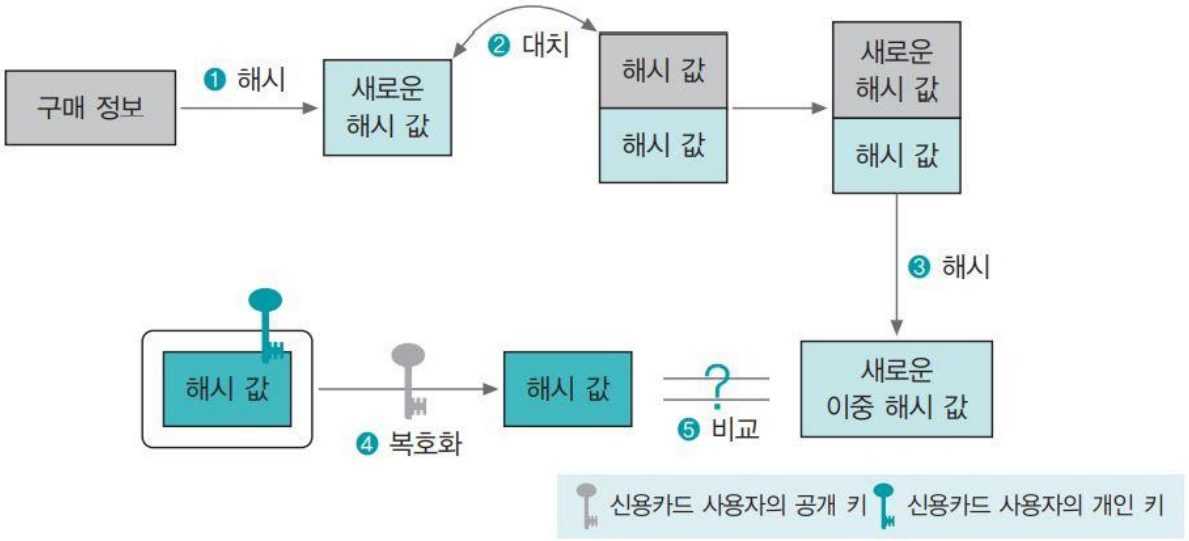
결제를 위해 신용카드 사용자가 상점에 전송하는 데이터

4. 전자 결제

SET

이중 서명의 원리

- 상점은 구매 정보를 확인
  - ① 신용카드 사용자가 구매한 물건에 대한 **구매 정보의 해시**를 구함
  - ② 신용카드 사용자가 보내온 **한 쌍의 해시 값**을 **새로 구한 해시**로 대치
  - ③ **새로운 이 중 해시**를 구함
  - ④ 신용카드 사용자의 **개인 키**로 암호화된 **해시 값**을 복호화하여 이를 새로 구한 **이중 해시 값**과 비교



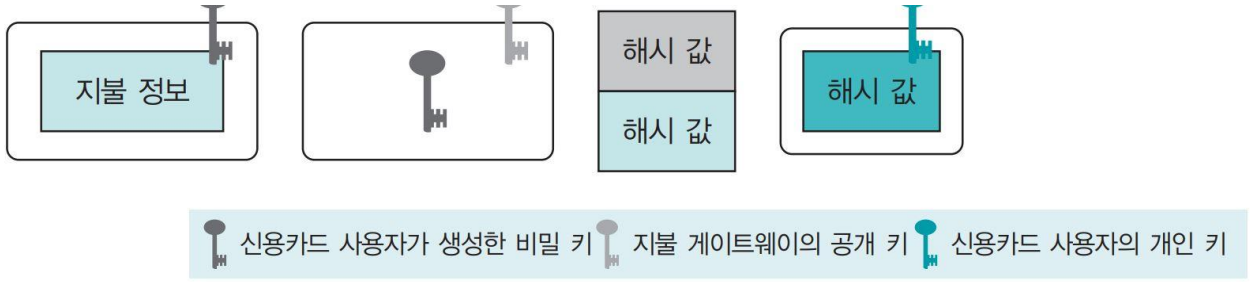
이중 해시 값을 이용한 구매 정보 확인

4. 전자 결제

SET

이중 서명의 원리

- 구매 정보를 확인한 상점은 다시 데이터 세트를 만들어 지불 게이트웨이로 전송
- 상점이 지불 게이트웨이로 보내는 데이터는 구매 정보만 빼면 신용카드 사용자가 처음 상점에 전송한 데이터와 같음



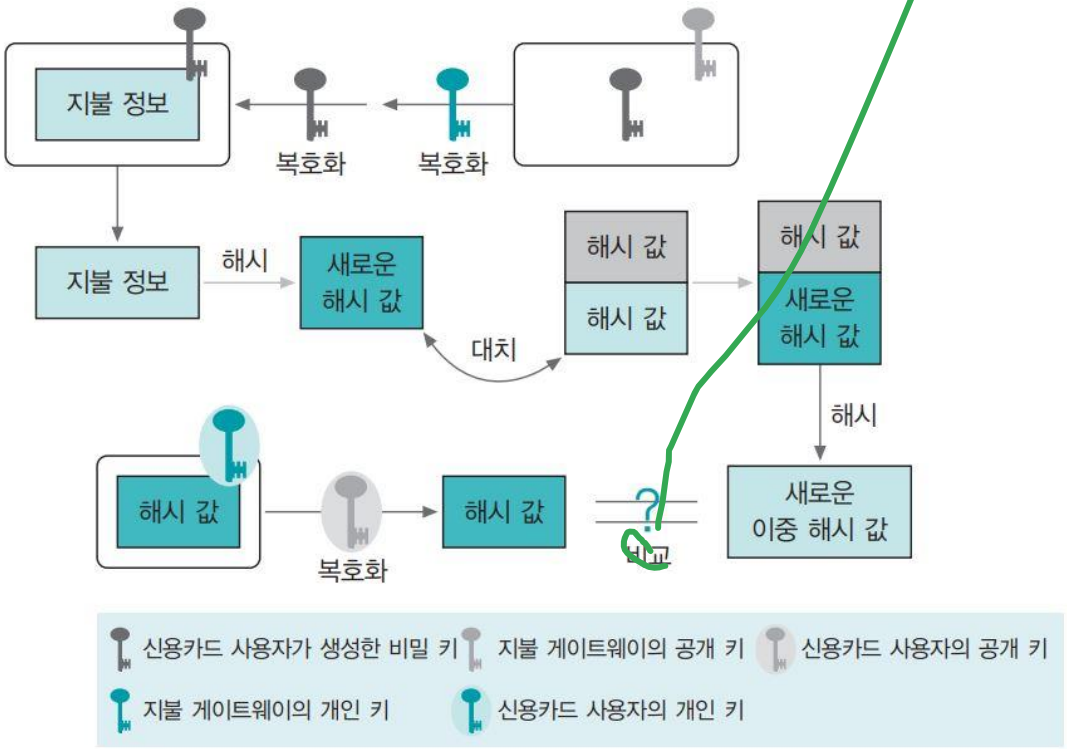
상점이 지불 게이트웨이로 보내는 데이터

4. 전자 결제

SET

이중 서명의 원리

- 데이터를 상점으로부터 받은 지불 게이트웨이는 자신의 개인 키로 비밀 키를 복호화하여 지불 정보를 확인
- 상점이 한 것처럼 지불 정보를 해시한 값으로 한 쌍의 해시 값을 대치하여 이중 해시 값을 비교
- 지불 정보의 변조 여부를 확인한 뒤 상점에 대금을 지불



지불 정보 확인

5. 네트워크 암호화

■ 네트워크 암호화

- 전자상거래에 이용되는 암호화 프로토콜에는 실제 거래에 사용하는 응용 프로그램에 의한 것도 있고 2~4계층에서 동작하는 암호화 프로토콜도 있음

OSI 계층별 암호화 프로토콜

5. 네트워크 암호화

네트워크 암호화

- 2계층의 암호화 프로토콜
  - PPTP
    - 마이크로소프트가 제안한 VPN 프로토콜로 두 대의 컴퓨터가 직렬 인터페이스로 통신할 때 이용
    - 전화선으로 서버에 연결하는 PC에서 자주 사용됨
  - L2TP
    - 시스코가 제안한 L2F와 PPTP가 결합된 프로토콜

- PPTP와 L2TP의 공통점과 차이점
  - PPTP와 L2TP는 모두 PPP 트래픽을 암호화하므로 다양한 상위 로컬 네트워크 프로토콜 사용
  - PPP에서 제공하는 사용자 인증이나 데이터 암호화 및 압축(CCP, ECP) 등의 보안 기능 사용

PPTP와 L2TP 프로토콜의 비교

구분	PPTP	L2TP
네트워크	통신을 위해 양단의 네트워크가 IP를 기반으로 한다.	프레임 릴레이(frame relay), ATM 등에서도 사용할 수 있다.
터널링	두 시스템 사이에 하나의 터널만 지원한다.	여러 개의 터널을 허용하므로 QoS(Quality of Service)에 따라 서로 다른 터널을 이용할 수 있다.
압축 및 인증	해당 기능이 없다.	헤더 압축 및 터널에 대한 인증 기능을 제공한다.

5. 네트워크 암호화

네트워크 암호화

3계층의 암호화 프로토콜

- IPSec
  - 3계층의 암호화 프로토콜. IP를 기반으로 한 네트워크에서만 동작.
  - IP 스푸핑이나 스니핑 공격에 대한 대응 방안이 될 수 있음
  - 주요 기능은 AH를 이용한 인증, ESP를 이용한 기밀성, IKE를 이용한 비밀 키 교환

IPSec 프로토콜의 기능

기능	설명
AH(Authentication Header)	• 데이터가 전송 도중에 변조되었는지 확인할 수 있도록 데이터 무결성을 검사한다. • 데이터를 스니핑한 뒤 해당 데이터를 다시 보내는 재생 공격(replay attack)을 막을 수 있다.
ESP(Encapsulating Security Payload)	• 메시지 암호화를 제공한다. • 암호화 알고리즘에 DESCBC, 3DES, RC5, IDEA, 3IDEA, CAST, blowfish가 있다.
IKE(Internet Key Exchange)	• ISAKMP(Internet Security Association and Key Management Protocol), SKEME, Oakley 알고리즘의 조합으로 두 컴퓨터 간의 보안 연결(Seucrity Association, SA)을 설정한다. • IKE를 이용한 연결에 성공하면 8시간 동안 SA를 유지한다. 8시간이 넘으면 SA를 다시 설정해야 한다.

5. 네트워크 암호화

■ 네트워크 암호화

▪ 4계층의 암호화 프로토콜

- SSL
  - 넷스케이프가 개발한 SSL은 40비트와 128비트 키를 가진 암호화 통신 가능
  - L2TP, IPSec보다 상위 수준에서 암호화 통신 기능을 제공하여 4계층(전송 계층)과 5계층(세션 계층) 사이의 프로토콜
- SSL의 기능
  - 클라이언트 인증: 클라이언트의 인증서를 확인하여 서버 에 접속할 자격이 있는지 확인하는 작업
  - 암호화 세션: 암호화된 통신, 40비트와 128비트의 암호화 세션을 형성
  - 서버 인증: 클라이언트가 자신이 신뢰할 만한 서버에 접속을 시도하고 있는지 확인하는 것

OSI에서 SSL의 동작 위치