- **인증 방법: 사용자가 알고 있는 것**

  Password / Pass Phrase / PIN / etc.

  Pros 원거리 신원 확인이 가능하고, 설치 비용이 적다.

  Cons 망각하거나 다른 사람이 추측할 수 있다.

- **인증 방법: 사용자가 소유하고 있는 것**

  열쇠 / 신분증 / 도장 / Token / Smartcard / etc.

  Pros 기억할 수 없이 긴 암호를 저장가능.

  Cons 분실, 도난, 복제, 위임

- **인증 방법: 사용자만의 고유한 특징**

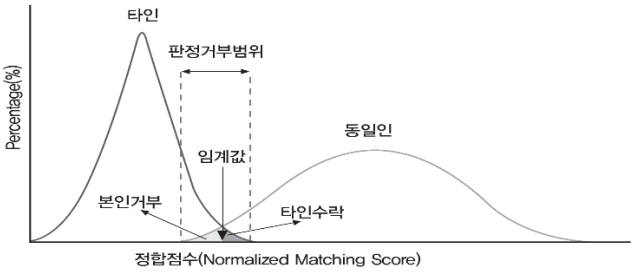
  지문 / 얼굴 / 손 / 망막 / DNA / 목소리 / 수기 서명 / 필체

  Cons 높은 오류율, 큰 비용, 유출 시 사용불가능

- **생체학적(Biometric) 식별 기법**

- **생체공학기반 식별과 인증**

식별	인증
- 1:多 비교 - 입력정보와 등록된 모든 정보 비교 - 유사도가 높은 순으로 정렬된 결과값 - 사용의 편리함 - 구현과정이 복잡함 - 범죄수사 등에 이용	- 1:1 비교 - 입력정보와 이에 해당되는 등록정보의 비교 - 일치/불일치의 결과값 - 사용의 불편함 - 구현이 비교적 쉬움



- TYPE I 에러 (False Negative, 본인 거부).
  거부 오류 비율 (False Rejection Rate, FRR)을 가짐.

- TYPE II 에러 (False Positive, 타인 수락)
  허용 오류 비율 (False Acceptance Rate, FAR)을 가짐.

- 교차 에러 비율 (Cross Error Rate, CER)
  잘못된 거부 비율과 잘못된 허용 비율이 같은 지점을 표시.

생체공학 시스템에서 정확성 외 고려해야 될 요소

- **등록시간 (Enrollment Time)** 평가될 생체공학정 특징들의 샘플을 제공함으로써 시스템 초기 등록에 필요한 시간.

- **처리율 (Process Rate)** 개인들의 데이터가 일단 등록되고 나면 시스템에 의해 처리되고 식별되거나 인증되는 속도.

- **수용 가능성 (Acceptability)** 시스템을 사용할 때 프라이버시, 침해, 정신적 육체적인 안락함에 대한 고려.

- **생체인식 방법의 효율성과 수용성**

효율성	높다 ← 효율성 → 낮다							
생체 인식	손바닥	손기하학	홍채	망막 패턴	지문	성문	얼굴	서명 동작	키보드 동작

수율성	높다 ← 수용성 → 낮다							
생체 인식	홍채	키보드 동작	서명 동작	성문	얼굴	지문	손바닥	손기하학	망막

- **기존 Password의 문제점**

  원격접속시 전송되고 있는 패스워드는 암호화 되어 있지 않아 공격자에 의한 불법적인 도청 등과 같은 위험에 노출되어 있음.

  일반적으로 사용자들이 기억하기 쉬운 유형의 문자열로 구성되므로 공격자에 의한 추측이 가능.

  기억하기 힘든 패스워드를 만들어서 사용할 경우 메모지에 적은 후 컴퓨터 주변에 붙여 둘 때가 있는데, 이는 보안상 취약함.

  시스템 서버의 Password 파일은 모든 사용자들의 Password를 기록하고 있으므로 공격자의 주된 공격의 대상이 될 수 있음.

- **Password 점검기 (Checker)** 전체적인 네트워크 환경이 사용자들의 패스워드를 찾아내기 위하여 행해지는 침입자들의 사전 공격이나 다른 공격들에 대비하도록 보안을 강화 시켜 줌.

- **Password 생성기 (Generator)** 무작위지만 사용자들이 암기하는데 지장 없는 Password를 만들어 준다.

- **Password 에이징 (Aging)** 많은 시스템들은 정기적으로 패스워드를 갱신하도록 유도하며, 사용자들이 마지막으로 사용한 5~10개의 Password 목록을 저장하여 사용자들이 이미 사용한 Password를 다시 사용하지 않도록 하고있다.

- **접속 시도 횟수 제한 (Limited Login Attempts)**

  가능한 접속 시도 횟수를 채우게 되면 그 사용자의 계정은 일정기간 또는 무기한 접속이 제한. 접속 제한을 해제하기 위해서는 관리자가 수동으로 조작해야 한다.

  사전공격과 사용자의 이름과 Password를 발견할 때까지 지속적으로 입력하는 조작 공격에 효과적인 대처방법.

- **인식적 (Cognitive) Password**

  사용자는 자신의 경험에 대한 몇 가지 질문에 답을 제시함으로써 등록을 마침. 다른 메커니즘들보다 오래 사용할 수 있기 때문에 매일 인증과정을 통과할 필요가 없는 사용자들에게 적합하다. (Help desk 서비스에 적합).

- **일회용 (One-Time) Password (OTP)**

  한 번만 사용하고 버리는 Password를 사용. 사용자에게 고정 Password를 부여하는 대신, 어떤 함수를 부여. 따라서 사용자는 함수 계산을 위한 장비 혹은 소프트웨어가 있어야함.

- **GSM (Global System for Mobile Communication)**

  디지털 전송과 cellular 방식에 기반을 두는데, GSM 가입자에게는 SIM(Subscriber Identity Module)이라는 스마트카드가 발급됨. 카드가 이동전화 단말기를 확인하고, 가입자에 대한 신분확인 통화내용의 암호화, 통화내역에 대한 요금징수가 수행된다.

- **권한부여 / 접근제어 / 접근통제**
- **권한부여 기술의 접근 기준 (Access Criteria)**
  - **역할 (Role)** 특정 작업을 수행하는 어떤 유형의 사용자들에게 권한을 부여하는 효과적 방식. 업무 배정이나 기능에 근거.
  - **그룹 (Group)** 몇몇 사용자들이 정보와 자원에 대한 동일한 유형의 접근을 요구할 때 효과적 방식.
  - **물리적, 논리적 위치 (Physical or Logical Location)**
    사용자가 반드시 다른 컴퓨터 앞에서 신원 증명을 입력해야 한다. 허가되지 않은 사용자가 원격으로 서버를 재구성하거나 설정하는 것을 방지할 수 있다.
  - **시간** 하루 중 특정시간 또는 주중 특정 날짜에만 접근 허용.
  - **업무 유형 (Transaction Type)** 특정 유형의 기능이 실행되는 동안 접근할 수 있는 데이터와 그 데이터에 수행될 수 있는 명령을 통제.
- **접근 권한의 원칙**
  - **필요지식 (Need-To-Know)** 사용자들이 업무를 수행하는 데 있어서 꼭 필요한 기본 권한을 설정하고, 설정된 권한에 대하여 허가를 부여 받은 후 시스템에 접근하도록 하는 원칙.
  - **최소권한 (Least-Privilege)** 최소한의 권한을 준다는 것은 업무를 수행하는 데 지장이 없으면서 안정적인 보안을 유지할 수 있도록 하는 접근 권한부여의 원칙.
  - **의무분리 (Separation of Duty)** 자신이나 데이터에 대한 인가받지 않은 접근이나 행위를 수행할 수 없도록 보장하는 정책, 절차 및 기관의 구조.
- **DB 관리 시스템 (DBMS, Database Management System)**

  사용자와 데이터베이스 사이에 위치하여 모든 사용자나 App의 요구에 따라 DB를 조작 및 제어하며 공유할 수 있도록 관리해 주는 기능을 제공하는 SW 시스템.
- **DBMS의 기능**

  데이터의 무결성(Integrity) 보장 / 허가되지 않은 사용자의 불법적인 접근에 대한 차단기능 / 트랜잭션의 병행 수행을 제어 / 장애 발생시 DB를 일관적인 상태로 회복할 수 있는 제어기능.
- **DB 보안** 인가되지 않은 사용자의 접근을 방지하여 의도적이고 악의적인 데이터의 변경, 파괴 등으로부터 데이터를 보호하는 것.
- **DB 보안 요구사항**
  - **부적절한 접근 금지** 인가된 사용자에게만 접근이 허락되어야 하고 접근 요청은 DBMS에 의하여 검사되어야 함.
  - **데이터의 무결성 유지** 인가되지 않은 사용자의 데이터 변경이나 파괴 또는 오류나 바이러스로부터 데이터를 보호하여야 함.
  - **추론 방지** 데이터의 통계적인 값으로부터 개별적이 ㄴ데이터 항목의 정보를 추론하지 못하도록 하여야 함.
  - **감사기능** DB에 대한 모든 접근의 감사기록이 생성되어야 하고 후속적인 분석이 가능해야 함.
  - **제한** 시스템 프로그램 간의 부적절한 정보전송을 방지해야 함.
  - **사용자 인증** DBMS는 엄격한 사용자 인증을 필요로 함.
- **접근제어** 주체가 객체에 접근을 요구했을 때 이 요구를 수락할지, 거절할지를 결정하는 행위.
- **강제적 접근제어 (MAC, Mandatory Access Control)**

  다단계 보안 (MLS, Multi-Level Security) 모델이라 부르기도 함. 주체와 객체에 적절한 보안등급(레이블)을 부여. 접근제어 시 등급을 비교하여 접근의 허용여부를 판단. 주로 군사환경과 같은 엄격한 보안이 요구되는 분야에 적합.

  보안 레이블(Security Label)은 개체(Entity)들이 보안상 얼마나 중요한가를 나내는 속성으로 계층적인 등급을 이룸.
  - **벨-라파둘라(BLP) 모델의 규칙** 중요한 정보를 저장하고 처리하기 위해 미 국방성의 다단계 보안 정책을 정형화 하도록 개발.
    - Role 1. No Read Up

      하위 보안 레벨에 있는 주체가 상위 보안 레벨에 있는 객체로부터 정보를 읽는 상태는 허용되지 않음.
    - Role 2. No Write Down

      높은 레벨의 주체가 낮은 레벨의 보안등급에 있는 객체에 정보를 쓰는 상태는 허용되지 않음.
- **규칙 기반 접근제어 (RBAC, Rule-Based Access Control)**

  기존에 네트워크 관리자에 의해 설정된 접근제어목록(Access Control List)에 의해 결정되는 접근제어 방식. E.g. 방화벽(Firewall)
- **임의적(자율적) 접근제어 (DAC, Discretionary Access Control)**

  객체에 대한 소유권(Ownership)에 기초해서, 소유권을 가진 주체가 객체에 대한 권한의 전부 또는 일부를 다른 주체에게 부여(Grant)하는 것.

  즉, 내가 제3자로부터 부여 받은 권한을 다른 사용자에게 부여 가능. 부여자가 권한을 회수하면 피부여자의 권한도 자동으로 회수되는 것을 원칙.
- **역할기반 접근제어 (RBAC, Role-Based Access Control)**

  기관(Organization)에서 개인이 수행하고 있는 역할(Role)에 의해 결정되는 접근제어 방식.
- **관리적 통제 (Administrative Controls)**

  보안 정책을 구성, 지원 절차, 기준 및 지침의 개발을 위임. 보안 목표달성을 위한 통제방법, 통제 시험이 실행되어야 하는지를 지정.
  - **정책 및 절차 (Policy and Procedure)**

    상위 수준의 계획. 보안이 실행되고 지속되는 방법과 관련된 각 내부 인원과 불복종에 따른 영향을 제시. 절차, 가이드라인 및 기준은 보안 정책을 지원하는 세부사항 제공.
  - **직원 통제 (Personnel Controls)**

    내부자들이 보안 메커니즘을 준수하기 위하여 내부 인력의 효율적인 통제 및 관리. 직무분리, 직무교대, 감독구조, 보안 의식훈련, 모든 보안 통제와 메커니즘을 정기적으로 시험.

- **물리적 통제 (Physical Controls)**
  - **네트워크 분리** 물리적 및 논리적 방법 실행.
  - **경계선 보안** 직원들이 어떤 보안 구획에 들어가기 전 사진이 포함된 신분증 제시 후 인증.
  - **컴퓨터 통제** 컴퓨터의 내부 부품을 훔쳐가지 못하도록 하는 장치
  - **작업 영역 분리** 특정한 개인들만 특정한 시설 구획에 접근 할 수 있도록 지정.
  - **데이터 백업** 비상시나 네트워크 및 시스템 장애의 경우, 정보접근을 보장하는 물리적 통제.
- **논리적(기술적) 통제 (Technical Controls)**

  주체 및 객체로의 접근을 제한하는데 사용되는 HW와 SW에 기반. OS, 부가적 보안 패키지, App, 네트워크 HW 장치, 프로토콜 및 접근 통제 매트릭스의 핵심요소.

  접근할 수 있는 주체의 수를 제한 / 인증 받지 않은 주체에게 개방하는 것을 방지 / 기밀성(Confidentiality) / 무결성(Integrity) / 가용성(Availability) / 감사(Auditing) 도구 / 통제구역(Control Zone)
- **감사 (Auditing) 도구** 네트워크 내부 장치상 또는 특정 컴퓨터상 작업들을 추적하는 기술도구.
- **통제구역 (Control Zone)** 중요 정보가 전파를 통하여 나가는 것을 방지하는 전자파 보안 등.
- **Steganography**

  다른 사람이 인식하지 못하도록 통신문을 감춘다는 뜻. Cryptography가 메시지의 내용을 읽을 수 없게 하는 수단인 반면, Steganography는 존재 자체를 숨긴다.
- **Caesar Cipher** 문자의 순서를 n번 이동했을 때 대응되는 다른 문제로 평문을 치환하여 암호문으로 변환.
- **Vigenere Cipher** Caesar Cipher에서 키(Key)를 정하여 한 문자를 여러 문자로 치환될 수 있도록 하였음.
- 암호 알고리즘과 키, 그리고 이들을 사용 가능하도록 하는 암호 프로토콜을 합하여 **암호 시스템 (Cryptosystem)**이라고 함.

  공격자가 암호 시스템을 깨기 위해 필요한 소요시간과 노력은 키의 길이, 안정성, 그리고 알고리즘의 견고성에 따라 달라짐.
- **대칭키/비밀키 (Symmetric Key/Private Key) 방식**

  암호화와 복호화에 동일한 키를 사용하는 방식.

  Pros 상대적으로 암호화 속도가 빠름

  Cons 기밀성은 보장하지만 인증이나 부인방지 기능은 제공하지 못함.
- **블록 암호 알고리즘 (Block Cipher)** 평문을 N bit씩 나눈 블록 단위로 암호화를 수행. 알고리즘의 구조에 따라 Feistel 구조, SPN 구조 등으로 분류

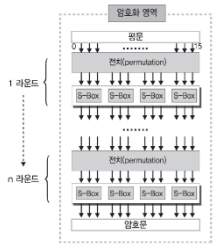
  Pros 평문에 혼돈성을 주어 해독을 어렵게 하고, 완성된 암호문에 내용의 추가 또는 변경이 어렵다.

  Cons 암호화 속도가 상대적으로 느리고, 암호화 시 에러의 파급 효과가 크다.
- **Block Cipher: Feistel Structure (파이스텔 구조)**

  암호화 과정과 복호화 과정이 동일한 방식. 복호화를 위해 암호화의 역과정을 고려할 필요가 없어 구현이 비교적 쉬움.

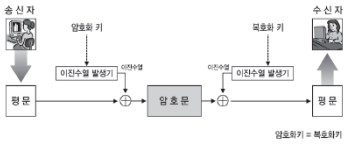
  평문 전체를 블록 단위로 배열하고 블록을 동일한 크기로 분할한 뒤, 블록의 좌측과 우측을 라운드마다 위치를 교환하면서 함수 F와 XOR 연산을 통해 암호화 과정을 실행
- **Block Cipher: Substitution Permutation Network Structure (대입, 치환 구조)**

  Shannon의 혼동(Confusion)과 확산(Diffusion) 이론을 기반. 암호화 과정과 복호화 과정이 달라 암호화의 역과정을 고려해야 하므로 구현이 복잡함.



- **대칭키 방식의 스트림 암호 알고리즘 (Stream Cipher)**

  대칭키 암호화 알고리즘의 한 방식으로 블록 암호보다 HW 구현에서 수행속도가 빠르며, HW 복잡도가 낮다.



  전송오류가 매우 높거나, 메시지 버퍼링이 제한되어 있거나, 문자들이 수신즉시 처리되어야 하는 상황에서 유용하다.

  Pros 암호화 속도가 상대적으로 빠르고 에러의 파급효과가 적다.

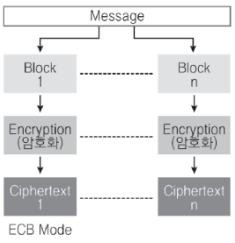
  Cons 평문의 특성(문자 출현 빈도) 등이 암호문에도 그대로 반영됨. 악의적 공격자에 의해 쉽게 내용의 첨가 또는 변경이 가능하다.
- **DES (Data Encryption Standard)**

  64bit 키를 사용해서 64bit 평문을 64bit 암호문으로 만드는 블록 암호시스템. 64bit의 키(외부 키) 중 56bit는 실제의 키(내부 키)가 되고 나머지 8bit는 검사용 비트로 사용된다.

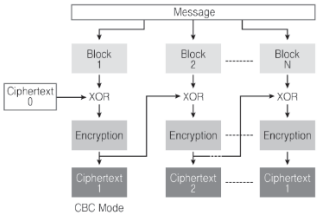
  64bit 단위 블록으로 구성된 평문 메시지를 16라운드의 반복적인 암호화 과정을 실행. 각 라운드마다 전치(Transposition) 및 대입(Substitution)의 과정을 거친 평문과 56bit의 내부 키에서 나온 48bit의 키를 이용하여 암호문을 만든다.

  복호화 과정과 암호화 과정은 동일하지만, 키를 역순으로 사용한다.
  - **ECB (Electronic Code Block) 방식**

    대규모 데이터의 암호화 및 복호화에 적절한 방식으로 어떠한 피드백도 사용하지 않음. 동일한 크기의 평문의 각 블록은 동일한 크기의 암호문 블록으로 변환됨. 난수 발생과 같은 특수한 경우에만 사용된다.

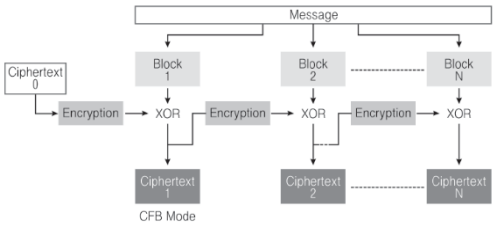


  - **CBC (Cipher Block Chaining) 방식**



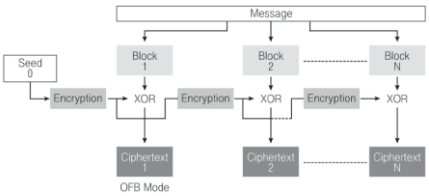
    입력된 초기값이 이전의 암호문과 배타적 논리합의 값이 되며, 복호화 알고리즘은 암호화 알고리즘의 반대이다. 암호학적 특성이 우수하고 인증에 적합하다.

- **CFB (Cipher Feed Back) 방식**



인증에 적합하고 CBC 방식과 유사하다. 특별한 응용인 경우, n-bit보다 작은 r-bit 단위로 암호화할 필요가 있을 때 사용 가능.

- **OFB (Output Feed Back) 방식**



CFB 방식과 유사하게 r-bit 단위로 암호활 수 있다. 단, 초기에 임의로 생성된 값(seed)을 사용하며, 출력값이 암호의 입력값으로 사용된다.

- **Rivest, Shamir, Adleman Algorithm (RSA)**

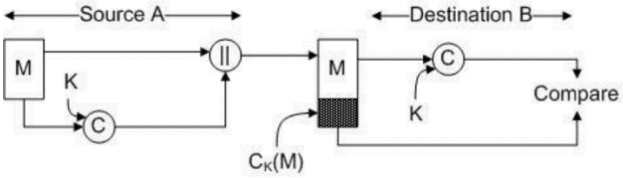
1. Choose two large prime numbers p, q.
2. Compute $n = pq$, $z = (p-1)(q-1)$
3. Choose e with $e < n$ that has no common factors with z. (e, z are "relatively prime")
4. Choose d such that $ed - 1$ is exactly divisible by z. In other words, $ed \bmod z = 1$.
5. Public key is $K_B^+(n, e)$, Private key is $K_B^-(n, d)$.

To encrypt message $m(< n)$, compute $c = m^e \bmod n$
To decrypt message c, compute $m = c^d \bmod n$

$$m = (m^e \bmod n)^d \bmod n$$

- **키-암호화 키** 비밀 암호 시스템의 비밀키나 공개키 암호 시스템의 개인키가 사용되는 키. 세션키보다 사용시간이 비교적 길다. 모든 사용자들은 사전에 기밀성과 무결성이 보장되는 채널을 통해 제공받음. 매 세션마다 새롭게 소요되는 세션키의 설정에 사용됨.
- **마스터키** 다른 유형의 키를 보호할 수 있는 키. 보안 관리자에 의해 생성되고 수작업으로 분배된다. 초기화되는 절차상의 보호를 받음.
- **키 관리 프로토콜** 세션키를 공유하기 위해서 행해지는 일련의 과정. 키 암호 시스템인 경우 상호간에 안전하게 메시지를 전송하기 위해 키 관리 프로토콜이 적용된다.
- **키 복구** 암호기술을 사용할 때 암호문을 해독할 수 있는 키 또는 키와 관련된 정보를 제3자에게 위탁하고, 일정한 조건하에 위탁된 키나 키와 관련된 정보 또는 평문을 권한이 있는 사람에게 전송하는것.
- **키 위탁** 키를 2등분하여 독립된 위탁 기관들이 보관하고, 2등분된 키가 합쳐져야만 암호문을 해독할 수 있도록 관리하는 것.

- **Message Authentication Code (MAC)**



MD5 or SHA-1 hash functions are widely used.

- **Length-Extension Attack on MAC**

$h(a)$, $\text{len}(m)$ 을 알고 있을 때 어떤 m'에 대해서라도

$$h(m \parallel \text{pad}(m) \parallel m')$$

을 계산할 수 있다.

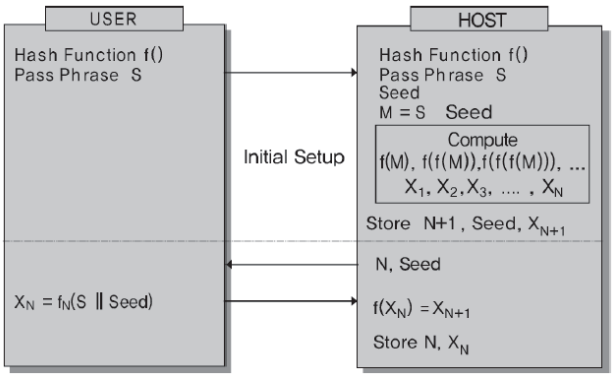
- **HMAC**

$$\text{HMAC}(K, M) = \text{H}\left((K^+ \oplus \text{opad}) \parallel \text{H}\left((K^+ \oplus \text{ipad} \parallel M)\right)\right)$$

M message input to HMAC
K^+ Kpadded with zeros on the left
b number of bits in a block ($b = 512$ for SHA-1)
ipad 00110110 repeated $b/8$ times
opad 01011100 repeated $b/8$ times

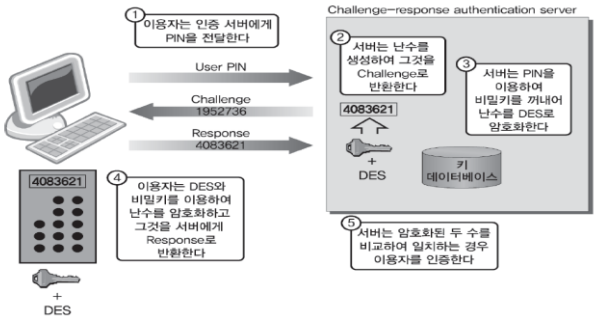
HMAC can resolve length-extension attack.

- **S/Key 일회용 패스워드 기법**

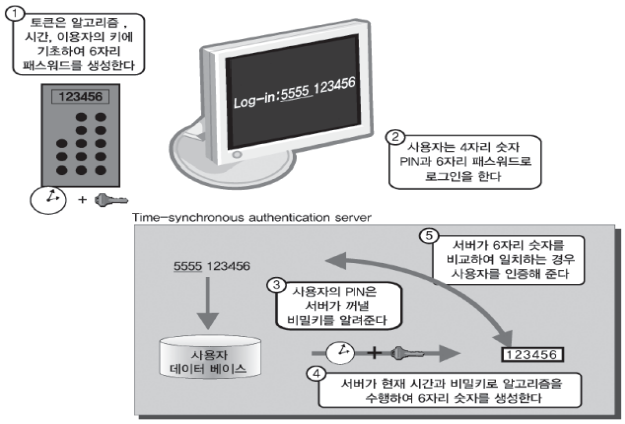


통신할 때마다 새로운 세션 패스워드를 생성. 도청, 재전송 공격 등으로부터 안전하게 사용자를 인증하는 기법.

- **Challenge-Response 기법**



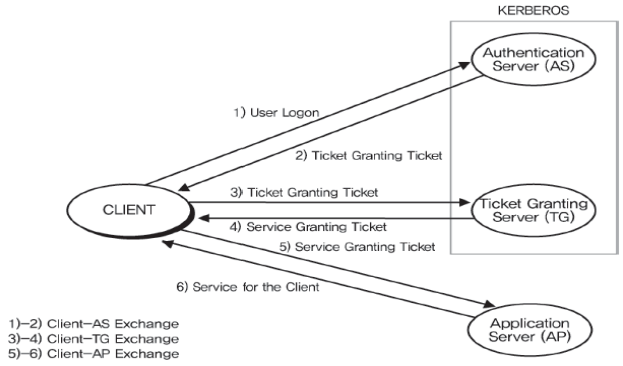
- **시간 동기화 (Time Synchronous) 기법**



일회용 패스워드의 특별한 형태. 시스템과 사용자의 인증 장치의 알고리즘에 의해 시간에 따라 변경. 인증 장치는 스마트카드라고 불리우며 현재의 패스워드를 판독.

사용자 시계와 서버 호스트 시계가 동기화되지 않을 경우 인증이 실패하는 단점을 갖고 있음.

- **Kerberos 인증 기술**



네트워크 기반 인증 시스템으로서 대칭키 암호 방식을 사용하여 분산 환경에서 개체 인증 서비스를 제공. 중앙집중식으로 운영되며 사용자 인증을 위해 티켓을 배포하는 티켓 발급 서버의 역할을 수행.

- 클라이언트 워크스테이션
- 인증 서버 (AS, Authentication Server): 사용자를 인증하여 TG로부터 Service-Granting Ticket을 받을 수 있도록 Ticket-Granting Ticket을 제공.
- 티켓 발행 서버 (TGS, Ticket-Granting Server): App Server로부터 클라이언트가 서비스를 받을 수 있는 Ticket을 제공.
- 응용 서버 (AP, Application Server): 클라이언트를 통해 사용자에게 원하는 서비스를 제공.

- **Summary of Kerberos V4 Message Exchanges**

(1) $\mathrm{C} \rightarrow \mathrm{AS}$

$$\mathrm{ID_C} \parallel \mathrm{ID_{TGS}} \parallel \mathrm{TS_1}$$

(2) $\mathrm{AS} \rightarrow \mathrm{C}$

$$\mathrm{E}_{K_\mathrm{C}}\{K_{\mathrm{C,TGS}} \parallel \mathrm{ID_{TGS}} \parallel \mathrm{TS_2} \parallel \mathrm{Lifetime_2} \parallel \mathrm{Ticket_{TGS}}\}$$

$$\mathrm{Ticket_{TGS}} = \mathrm{E}_{K_\mathrm{TGS}}\{K_\mathrm{TGS} \parallel \mathrm{ID_C} \parallel \mathrm{AD_C} \parallel \mathrm{ID_{TGS}} \parallel \mathrm{TS_2} \parallel \mathrm{Lifetime_2}\}$$

(3) $\mathrm{C} \rightarrow \mathrm{TGS}$

$$\mathrm{ID_C} \parallel \mathrm{Ticket_{TGS}} \parallel \mathrm{Authenticator_C}$$

$$\mathrm{Authenticator_C} = \mathrm{E}_{K_{\mathrm{C,TGS}}}\{\mathrm{ID_C} \parallel \mathrm{AD_C} \parallel \mathrm{TS_3}\}$$

(4) $\mathrm{TGS} \rightarrow \mathrm{C}$

$$\mathrm{E}_{K_{\mathrm{C,TGS}}}\{K_{C,V} \parallel \mathrm{ID_V} \parallel \mathrm{TS_4} \parallel \mathrm{Ticket_V}\}$$

$$\mathrm{Ticket_V} = \mathrm{E}_{K_\mathrm{V}}\{K_{\mathrm{C,V}} \parallel \mathrm{ID_C} \parallel \mathrm{AD_C} \parallel \mathrm{ID_V} \parallel \mathrm{TS_4} \parallel \mathrm{Lifetime_4}\}$$

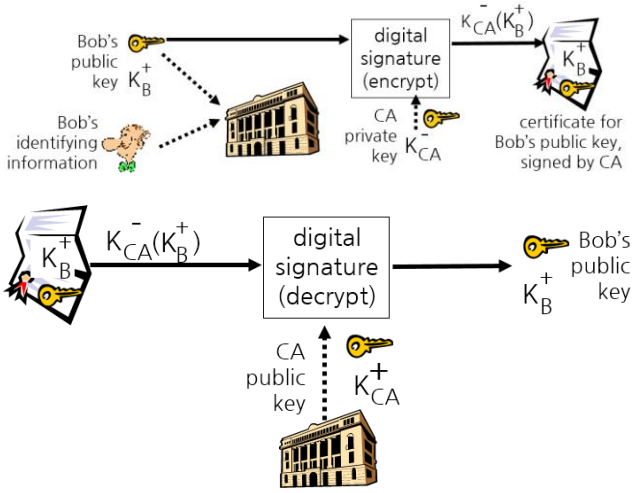
(5) $\mathrm{C} \rightarrow \mathrm{V}$

$$\mathrm{Ticket_V} \parallel \mathrm{Authenticator_C}$$

(6) $\mathrm{V} \rightarrow \mathrm{C}$

$$\mathrm{E}_{K_{\mathrm{C,V}}}\{\mathrm{TS_5} + 1\}$$

- **Certification Authorities (CA)**



Binds public key to particular entity, E. E provides proof of identity to CA. CA creates certificate binding E to its public key. Certificate containing E's public key digitally signed by CA.

1. Bob create public key and private key.
2. Bob register public key to CA.
3. CA create certificate including Bob's public key and signature using CA's private key.
4. Alice request certificate that included Bob's public key.
5. Alice decrypt certificate using CA's public key.
6. Alice check the key is Bob's.

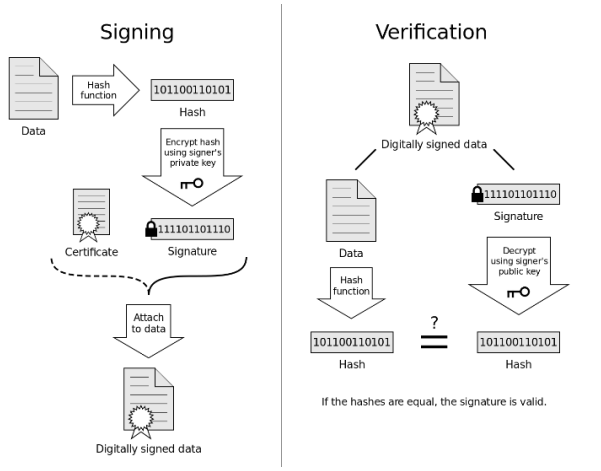
- **단일사용승인 (Single Sign On)**

  하나의 아이디로 여러 사이트를 이용할 수 있는 시스템.

  Pros 사용자의 편의성 증가 / 기업의 관리자 입장에서 회원에 대한 통합관리가 가능하여 마케팅을 극대화 시킬 수 있음 / 단순하고 고정된 패스워드보다는 보안에 강력한 패스워드를 사용 할 수 있음 / 패스워드 변경이나 삭제 등의 관리가 쉬움 / 시스템 자원에 접근하는 시간을 줄일 수 있음.

  Cons 특정 사용자가 일단 초기 로그온을 통해 시스템에 접근하는데 성공하고 나면 그 사용자는 어떤 제한도 받지 않고 통신망을 자유롭게 활보하게 됨.

- **Digital Signatures**



  Cryptographic technique analogous to hand-written signatures. The reason why encrypt after hash is to reduce the transmitted size.