Exercise 2

Consider the following control access in a bank, where users are Alice, Bob, Charlie and John:

User	Permission
Alice	read account of Mr X
Alice	read account of Mr Y
Alice	write in project New Bank
Alice	start application Money
Alice	start application Create New client
Bob	read account of Mr Y
Bob	write in project New Bank
Bob	start application Create New client
Bob	read account of Mr X
Charlie	read account of Mr X
Charlie	read account of Mr Y
Charlie	write in project New Bank
Charlie	start application Create New client
John	read account of Mr Y
John	start application Money
John	start application Create New client

Propose a RBAC model for improving this situation.

Solution :

User are: Alice, Bob, Charlie and John

Admin are: Alice, Bob and Charlie

Superuser are: Alice and John

Role	*Permission*
User	*read account of Mr Y*
User	*start application Creat New client*
Admin	*write in project New Bank*
Admin	*read account of Mr X*
Superuser	*start application Money*

Exercise 3

Let E be an IND-CCA2 secure encryption scheme. We modify this scheme into E' (m) = E(m)||h(m), where h is an hash function. This should help the user to detect some errors in the transmission of the messages. Prove that the new scheme E' is not IND-CPA. It means give an attack against IND-CPA for E'.

Solution :

Consider following adversary :

Notice that E'(mb) = E(mb)||h(mb)

A1(n, pk) : m1, m0, s

A2(n, pk, s, m1, m0, E'(mb)) : if h(m0) = h(mb) then return 0 else 1

Chapter. Access Control & Security Policy

☐ Bell-LaPadula Model

○ 의의 : gave a formal, mathematical model of MLS
to provide higher policy assurance of correctness

- 시스템이 어떤 환경에서 어디까지 할 수 있는지를 명확히 알 수 있다.
- 정보가 아래에서 위로 흐른다.

○ problem

- integrity는 다루지 않고 confidentiality만 다룬다.
(Biba Model → Clark-Wilson Model)
- 접근권한의 변경을 통제하는 policy를 가지고 있지 않다.
(Harrison-Ruzzo-Ullman Model → Chinese Wall Model)
- covert channel을 통한 information flow를 다루지 않는다.

☐ BLP Model in formal

○ Elements of Access Control

- a set of subjects S
- a set of objects O
- set of access operations A = {execute, read, append, write}
- a set of security Levels L, with a partial ordering <=

○ The State Set

- A state : (b, M, f), includes
- Access operations currently in use b
  · List of tuples (s, o, a), $s \in S$, $o \in O$, $a \in A$
- Access permission matrix
  · $M = (M_{s,o})_{s \in S, o \in O}$, where $M_{s,o} \subset A$
- Clearance and classification $f = (f_S, f_C, f_O)$
  · $f_S : S \rightarrow L$ maximal security level of a subject
  · $f_C : S \rightarrow L$ current security level of a subject ($f_C \leq f_S$)
  · $f_O : O \rightarrow L$ classification of an object

○ Simple Security Property (SS-Property) : <Read down>

- A state (b, M, f) satisfies the SS-property if
  · $\forall (s, o, a) \in b$, such that $a \in \{read, write\}$
  · $f_O(o) \leq f_S(s)$
- 즉, subject는 더 낮은 classification의 object들만 observe할 수 있다.

○ *-Property (Star-Property) : <Write up>

- A state (b, M, f) satisfies the *-property if
  - $\forall$(s, o, a)$\in$b, such that a$\in${append, write}
  - $f_C(s) \leq f_O(o)$

and

- if $\exists$(s, o, a)$\in$b where a $\in${append, write},
- then $\forall o', a' \in$ {read, write}, such that (s, o', a')$\in$b
- $fo(o') \leq fo(o)$

- subject는 더 높은 classification의 object들만 변경할 수 있으며, low-level object에 쓰고 있는 동안에는 high-level object를 읽을 수 없다.

☐ Covert Channel

○ 은닉채널 : 설계자가 당초에 의도했던 입출력 통로가 아닌 다른 통로를 통해 정보를 끄집어내는 것 ex) 카메라로 찍기

○ Covert channel은 *-property로 막을 수 없다.

○ 모든 covert channel을 막는 것은 일반적으로 매우 어려운 일이며, covert channel의 bandwidth를 제한하는 노력을 할 수 있다.

○ Military는 covert channel을 통해 keys를 유출하는 트로이목마를 방지하기 위해 암호 component를 hardware로 구현할 필요가 있다.

○ (예) High User(subject)를 전염시킨 트로이목마로부터 Low User(subject)를 전염시킨 트로이목마로 정보가 유출된다.(High User는 정보가 유출된 사실을 알지 못한다.)

☐ Biba Model

○ SS-Property : No read-down

- Subjects can only view content at or above their own integrity level

○ *-Property : No write-up

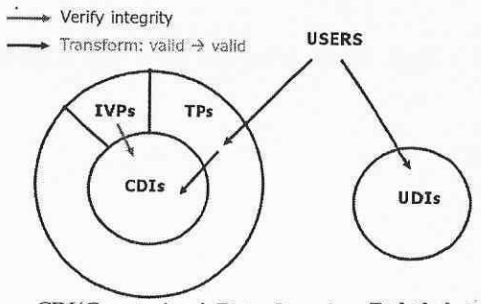
- Subjects can only create content at or below their own integrity level

☐ Clark-Wilson Model

○ 최초의 상업용 보안정책 모델

○ Integrity는 well formed transaction(시스템을 하나의 consistent state에서 다른 consistent state로 움직임)을 통해서 준수

○ User는 데이터 자체보다 프로그램에 대한 접근을 가진다.



- CDI(Constrained Data Items) : 무결성이 극도로 요구되는 데이터(예: 계좌 잔액)
- UDI(Unconstrained Data Items) : 무결성이 그다지 중요하지 않은 데이터 (예: 계좌 소유주에게 주어진 선물)
- IVP(Integrity Verification Procedures) : 상주하면서 CDI의 무결성을 체크 (예: 전날 잔고+입금-출금=현재 잔고)
- TP(Transformation Procedures) : valid state → valid state(예: 입금, 출금, 이체)

☐ Harrison-Ruzo-Ullman Model

○ 접근권한의 변경, subject와 object의 생성 및 삭제에 대한 정책 모델

○ Six primitive operation

- enter r into $M_{S,O}$, delete r from $M_{S,O}$
- create subject s, delete subject s
- create object o, delete object o

○ 접근권한이 변함에 따라 secure state를 계속 유지하는가의 문제는 undecidable problem이다. 그러나 특정 조건하에서는 검증이 가능하다.

☐ Chinese Wall Model

○ How it works:

- User들은 최초에는 wall을 갖지 않는다.
- 일단 한 파일이 access되면 경쟁사 정보에 관한 파일은 접근이 불가능해진다.
- 다른 모델과 달리, access control 규칙이 user의 행동에 따라 변한다.

○ Company Dataset : 동일한 회사와 관련된 object들
Conflict of Interest class : 경쟁관계에 있는 회사들. object의 내용에 관해 알아서는 안 되는 회사들의 집합

○ SS-Property : subject가 conflict of interest에 노출되는 것을 방지한다.

- object가 user에 의해 이미 열려있는 company dataset에 속하거나, 완전히 다른 conflict of interest class에 속할 때 접근이 허락된다.

○ *-Property : un-sanitized information이 company dataset 외부로 유출되는 것을 방지한다.

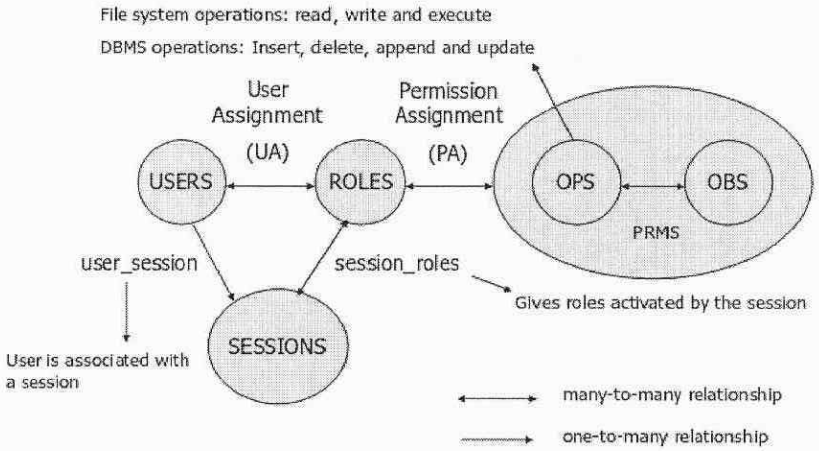
- write access granted if no other object can be read that :
  - belongs to a different company dataset
  - contains un-sanitized information

□ RBAC(Role-Based Access Control) Model : 역할기반 접근제어 모델

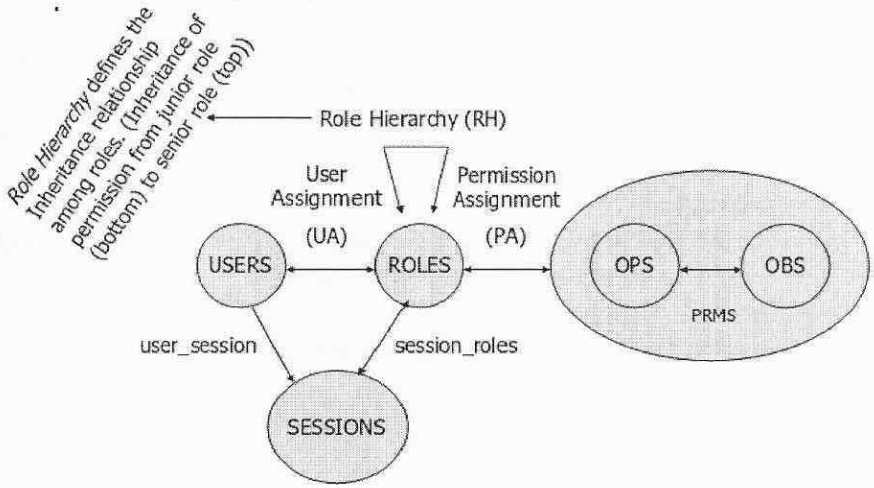
○ 회사 내에서 user가 보유하는 역할에 기반하여 resource에 대한 접근을 허락하는 모델

- User는 자주 바뀌나 역할은 자주 바뀌지 않으므로 사람에게 권한을 부여하는 것보다 에러가 적고 직관적이다.
- MAC, DAC은 사람에게 권한을 부여

○ Core RBAC($RBAC_0$)



○ Hierarchical RBAC($RBAC_1$)



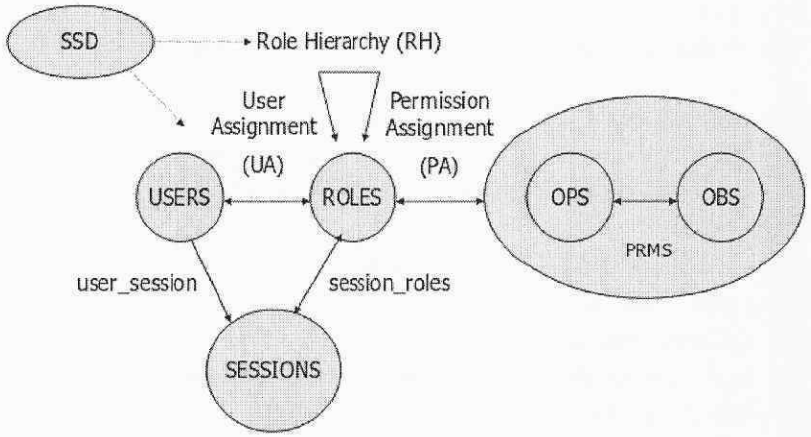
- Role Hierarchy는 역할들 사이의 상속 관계를 정의한다.
(junior role에서 senior role로의 permission의 inheritance)
  - General role hierarchies : 역할들 간에 permission과 user membership의 다중 상속개념을 포함
  - Limited role hierarchies : 역할은 하나 이상의 immediate ascendants를 가질 수 있으나 하나의 immediate descendent만으로 제한

※ role hierarchy가 없다면 역할들 간에 permission 할당이 중복될 수 있고, needs-to-know 원리가 깨질 수 있다.

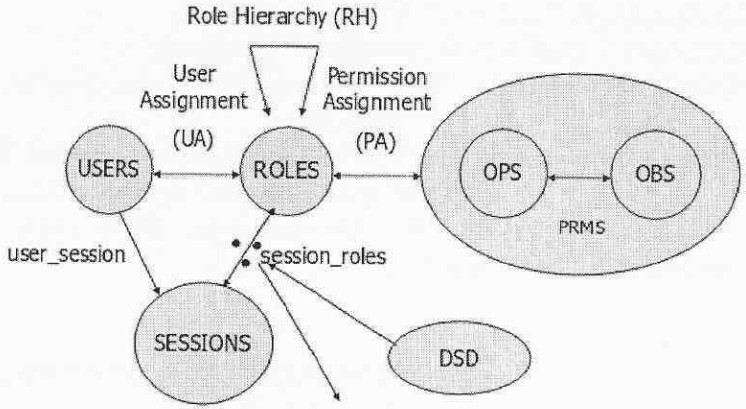
○ Constrained RBAC($RBAC_2$)

- Static Separation of Duty : user는 2개의 충돌되는 역할을 동시에 부여받을 수 없다. (mutually exclusive)
- Dynamic Separation of Duty : user가 2개의 역할을 동시에 수행할 수 없다.

SSD on the roles r_1 and r_2 implies that they should not be assigned to the same user.



· role r1과 r2는 같은 user에게 할당될 수 없다.



DSD on the conflicting roles r_1 and r_i implies that they should not be invoked in the same session by the same user. But the same user may invoke roles r_1 and r_2 in different sessions!

- role r1과 r2는 같은 user에 의해 같은 session에서 invoke될 수 없다. 그러나 같은 user가 다른 session에서 role r1과 r2를 invoke할 수 있다.

☐ Information Flow Security Model

○ access control 보다는 information의 흐름에 기반하고 있다.

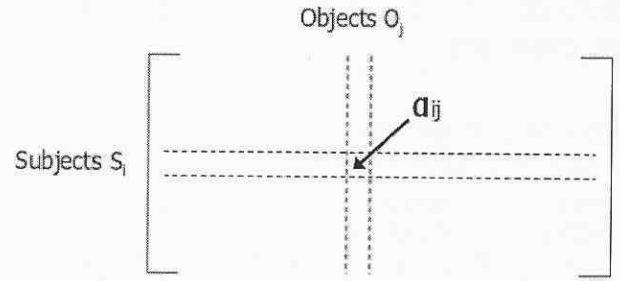
- object의 흐름은 다양한 레벨의 object들이 어디로 흘러갈지를 명세하는 보안정책에 의해 통제된다.
- covert channel을 통한 data 유출을 방지한다.

○ information flow의 측정 방법

- Entropy-based Analysis
  - Entropy : amount of information that can be derived from an observation
  - Change in entropy → flow
  - information flow from x to y : H(x|y) of x, given y
- Lattice-based Model
  - object c1으로부터 object c2로의 information flow는 c1≤c2인 경우에만 허용된다. 이것을 위반하는 모든 information flow는 illegal이다.(covert)
  - illegal한 information flow가 없다면 system은 secure하다.
  - (장점) 모든 종류의 information flow를 cover한다.
    (단점) 그러한 system을 디자인하기가 매우 어렵다. (주어진 system이 information flow 모델 하에서 secure한지를 체크하는 것은 undecidable problem)

Access Control Structures

☐ Access Control Matrix



○ Columns indexed by objects

○ Rows indexed by subjects

○ Matrix entries are access operations

☐ Capabilities & ACL

○ Capabilities

- Access Control Matrix에서 행(rows)을 나타내며, 사용자를 중심으로 접근권한을 리스팅 함으로써 subject가 각 자원에 대해 어떠한 접근권한을 가질 수 있는지 나타냄
- 자원에 접근할 수 있는 권한을 증명하는 데이터 소유의 부분임
- token, ticket 또는 key의 형태로 user와 함께 저장

예) Kerberos : 사용자 인증과 접근제어를 분리. 사용자 인증을 한 번만 하면 그 이후의 서비스는 권한만 있다면 별도 인증 없이 사용 가능

Android : Permission이 manifest file을 통해 application에 의해 요청되고, install 중에 user에 의해 승인되거나 거절된다.

○ Access Control List

- Access control Matrix에서 열(columns)을 나타내며, 자원을 중심으로 접근권한을 리스팅 함으로써 자원별로 subject의 접근권한을 나타냄
- resource와 함께 저장

예) Windows NT : NTFS permissions, Share permissions

■ Example

	R1	R2	R3	R4
S1	*rw*	*rwx*		
S2		*x*	*rwx*	*rwx*
S3	*rwx*	*r*		*r*

Capabilities:

S1: {(R1, *rw*), (R2, *rwx*)}

S2: ...

Access Control lists:

R1: {(S1, *rw*), (S3, *rwx*)}

R2: ...

Reference Monitor

☐ Safety and Liveness

- ○ Safety Properties : Some "bad thing" doesn't happen.
  - Execution Monitoring의 성격과 맞는다.
- ○ Liveness Properties : Some "good thing" eventually happen
  - 히스토리 정보를 계속 축적해야 하므로 Execution Monitoring의 성격과 맞지 않는다.
  - 예) information flow

☐ What EM can & can't do

- ○ EM can do access control (DAC, MAC, MLS, ...)
- ○ EM cannot do information flow
- ○ EM cannot do Liveness/Availability(e.g. DoS)

☐ Reference Monitor : 우회되지 않는 100% 확실한 기능을 커널에 삽입하여 실시간으로 보안정책을 위반하는지를 체크

- ○ Execution monitor that forwards events to security-policy-specific validity checks
- ○ Implementation requires
  - Capturing all policy- relevant events
  - Tamper-proof & Non-bypassable
  - Small enough to be analyzable

Chapter. Cryptography & Security Design

Step 3) Design

☐ Emphases of Modern Cryptography

- ○ The central role of definitions : security의 formal한 정의는 암호디자인에 있어 필수적인 첫 단계
- ○ The importance of formal and precise assumptions : 이러한 가정들은 명백하고 모호함이 없이 정의되어야 한다.
- ○ The possibility of rigorous proofs of security : 암호구조가 안전함을 증명할 수 있다.

I. Symmetric Ciphers

☐ Theoretical Construction of Secret-Key Crypto (대칭키의 secure design method)

- ○ NP : RSA, Discrete Log, Factoring, ...
- ○ One-Way Function (or One-Way Permutation)
- ○ Hard-Core Predicate
- ○ Pseudorandom Generator with +1 Expansion
- ○ Pseudorandom Generator with Arbitrary Expansion
- ○ Pseudorandom Function
- ○ (Strong) Pseudorandom Permutation ↔ ○ Block Ciphers
  (Pseudorandom Function + Feistel Network)
  CPA-Secure Secret-Key Encryption Scheme for fixed-length message
  CPA-Secure Secret-Key Encryption Scheme for arbitrary-length message & Existentially Unforgeable MAC
  CCA-Secure Secret-Key Encryption Schemes

※ Practical Construction of Secret-Key

- NP 문제만 찾으면 설계가 가능하다. 그러나 이론적 단계를 모두 따라하면 시스템이 무거워지고 비효율적이 된다.

- One-Way Functions : 계산하기는 쉬우나 역을 계산하기는 어려운 Function
- Hard-Core Predicate : 절대로 복원이 안 되는 중요한 부분으로 one-way permutation으로부터 pseudorandom generator를 만드는 방법을 제공한다.
- PRG(Pseudorandom Generator) : input으로 uniform random bit string을 받아서 uniform random string과 (polynomial time algorithm에 의해) 구분할 수 없는 좀 더 긴 bit string을 output으로 내는 deterministic function이다.
  → StreamCipher가 해당
- PRF(Pseudorandom Function) : n-bit string과 n-bit string을 mapping하는 모든 function들의 집합으로부터 random하게 function을 선택하는 것
  → BlockCipher가 해당

※ 암호학적으로 안전한 난수발생기가 가져야 할 조건?
- 일반적으로 균등분포를 만족해야 하고 Next-bit Test를 통과해야 한다.

☐ Construction Secure Secret-Key Encryption
① Uniform Distribution : 모든 수가 동등한 확률로 나와야 한다.
② Avalanche Effect : key, message가 1bit 변하면 C는 50%이상 변해야 한다.
③ Differential (y1+y2) : Uniform Distribution 차분해독을 막아야 한다.

II. Asymmetric Ciphers

※ random oracle 모델은 Hash 함수는 안전하다는 전제조건을 추가하면서 수식과 증명을 간단히 하였으며 이에 따라 Provable security 이론의 Practical한 approach가 가능해졌다.

II.1 Provably Secure Public-Key Encryption

☐ Attack Method

Type of attack	Known to cryptanalyst
Ciphertext only	· Encryption algorithm · Ciphertext
Known plaintext	· Encryption algorithm · Ciphertext · Several pairs plaintext-ciphertext (random하게 주어진다.)
Chosen plaintext	· Encryption algorithm · Ciphertext · Several pairs plaintext-ciphertext, where the plaintext was chosen by the attacker
Chosen ciphertext	· Encryption algorithm · Ciphertext · Several pairs plaintext-ciphertext, where the ciphertext was chosen by the attacker · CCA1 : attacker가 ciphertext를 한꺼번에 선택 · CCA2 : attacker가 하나의 ciphertext를 선택하여 plaintext를 얻고 그 결과를 보고 다음 ciphertext를 선택

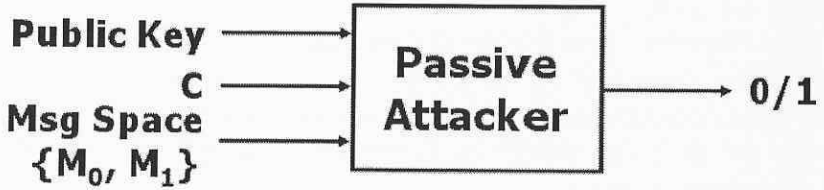
☐ OW-CPA
○ Security Goal : One-wayness
- 평문으로부터 암호문을 계산하기는 쉬우나 그 역은 어렵다.

○ Attacker Model

○ Security Proof : adversary가 암호알고리즘 A의 비밀성을 깰 수 있다면, 우리는 계산하기 어려운(Computationally Difficult) 문제 B를 깰 수 있다.

☐ IND-CPA
○ Security Goal : Polynomial Security
- 2개의 ciphertext를 구분할 수 없다. (Indistinguishability)

○ Attacker Model



→ Encryption Algorithm must be probabilistic

☐ Polynomial Sec vs Semantic Sec

○ Semantic Security : ciphertext로부터 plaintext의 어떠한 partial information도 노출되지 않는다.

○ Semantic Security = Polynomial Security :
Indistinguishable → Ciphertext로 부터 어떠한 정보도 유출되지 않는다.

○ Random Padding → Polynomial Security(IND-CPA)
⇔ Semantic Security → Ciphertext leaks no information

Indistinguishability – *IND*

Encryption scheme: $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$
Adversary: $A = (A_1, A_2)$

For any $k \in \mathbb{N}$ define $\mathrm{Adv}^{\mathrm{ind}}_{A,\Pi}(k) \stackrel{\mathrm{def}}{=}$

$$2 \cdot \Pr\left[(pk, sk) \leftarrow \mathcal{K}(1^k)\ ;\ (x_0, x_1, s) \leftarrow A_1(pk)\ ;\right.$$
$$\left. b \leftarrow \{0,1\}\ ;\ y \leftarrow \mathcal{E}_{pk}(x_b) :\ A_2(x_0, x_1, s, y) = b\right] - 1\ .$$

Π is *IND*-secure iff

A PPTM $\Longrightarrow \mathrm{Adv}^{\mathrm{ind}}_{A,\Pi}(k)$ negligible.

Chosen Ciphertext Security v1 (IND-CCA1)

(Naor–Yung 1990)
a.k.a. lunchtime attack.

Encryption scheme: $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$
Adversary: $A = (A_1, A_2)$

For any $k \in \mathbb{N}$ define $\mathrm{Adv}^{\mathrm{ccs\text{-}1}}_{A,\Pi}(k) \stackrel{\mathrm{def}}{=}$

$$2 \cdot \Pr\left[(pk, sk) \leftarrow \mathcal{K}(1^k)\ ;\ (x_0, x_1, s) \leftarrow A_1^{\mathcal{D}_{sk}}(pk)\ ;\right.$$
$$\left. b \leftarrow \{0,1\}\ ;\ y \leftarrow \mathcal{E}_{pk}(x_b) :\ A_2(x_0, x_1, s, y) = b\right] - 1\ .$$

Π is *CCS-1*-secure iff

A PPTM $\Longrightarrow \mathrm{Adv}^{\mathrm{ccs\text{-}1}}_{A,\Pi}(k)$ negligible.

Chosen Ciphertext Security v2 (IND-CCA2)

(Rackoff–Simon 1991)

Encryption scheme: $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$
Adversary: $A = (A_1, A_2)$

For any $k \in \mathbb{N}$ define $\mathrm{Adv}^{\mathrm{ccs\text{-}2}}_{A,\Pi}(k) \stackrel{\mathrm{def}}{=}$

$$2 \cdot \Pr\left[(pk, sk) \leftarrow \mathcal{K}(1^k)\ ;\ (x_0, x_1, s) \leftarrow A_1^{\mathcal{D}_{sk}}(pk)\ ;\right.$$
$$\left. b \leftarrow \{0,1\}\ ;\ y \leftarrow \mathcal{E}_{pk}(x_b) :\ A_2^{\mathcal{D}_{sk}}(x_0, x_1, s, y) = b\right] - 1\ .$$

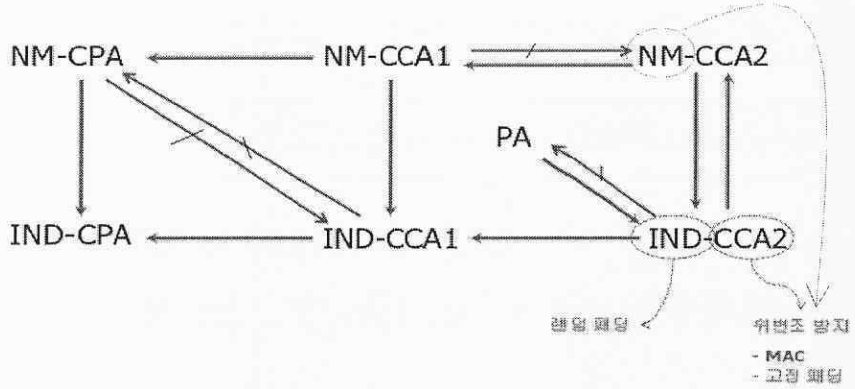
Π is *CCS-2*-secure iff

A PPTM $\Longrightarrow \mathrm{Adv}^{\mathrm{ccs\text{-}2}}_{A,\Pi}(k)$ negligible.

☐ Non-Malleability (NM)

- For any non-trivial relation R, E(M) → E(R(M)) is hard
- adversary가 어떤 암호문을 관련있는 평문으로 복호화되는 다른 암호문으로 변환할 수 있다면 encryption algorithm은 malleable하다.

☐ Relations & Countermeasures



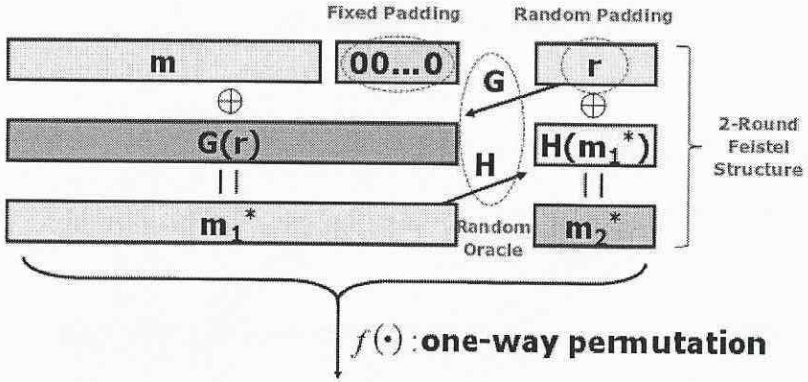
A→B : A를 충족하면 B를 충족한다는 것이 증명
A↛B : A를 충족하는 것이 B를 충족하지는 않는다는 것이 증명
※ 암호화 알고리즘이 안전하려면 ① 랜덤패딩, ② 위변조 방지 기술이 필요
- 랜덤패딩은 polynomial security, semantic security를 충족시킨다.

☐ Generic Composition Results

Composition Method	Security
1) Encrypt-and-MAC $E_{Ke,Km}(M) = E_{Ke}(M)\|\|T_{Km}(M)$	It is possible for the MAC of a message to leak the entire message.
2) MAC-then-Encrypt $E_{Ke,Km}(M) = E_{Ke}(M\|\|T_{Km}(M))$	Insecure [can fail to provide authenticity]
3) Encrypt-then-MAC $E_{Ke,Km}(M) = E_{Ke}(M)\|\|T_{Km}(E_{Ke}(M))$	Secure [always provides privacy and authenticity]

☐ OAEP (Optimal Asymmetric Encryption Padding)

○ RSA-OAEP



$$C = f(OAEP(m,r)) = (m_1^* || m_2^*)^e \bmod N$$

II.2 Provably Secure Secret-Key Encryption

II.3 Provably Secure Hybrid Encryption

II.4 Provably Secure 2-Party Encryption

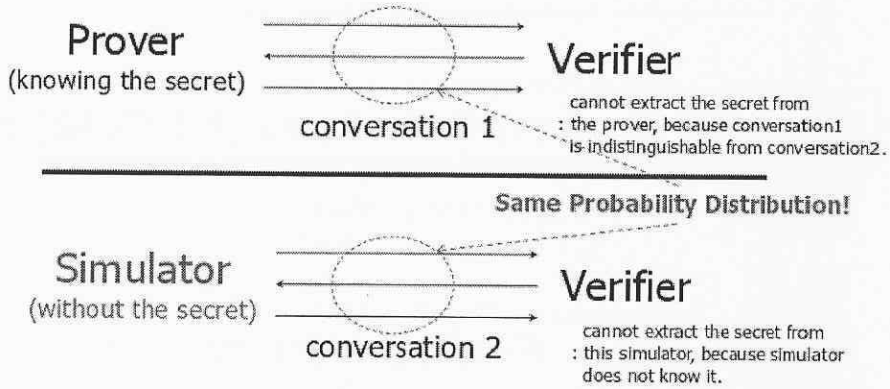
☐ Zero-knowledge proof : prover는 secret을 노출시키지 않으면서 그가 secret을 알고 있다는 사실을 verifier에게 확신시킨다.

☐ Zero-knowledge property

○ 어떠한 정보도 노출되지 않았다는 것을 증명

○ verifier가 프로토콜이 시작하기 전에 알고 있는 지식으로 실제 프로토콜 메시지들과 구별할 수 없는 가짜 프로토콜 메시지들을 생성해낼 수 있는 simulator가 존재한다.
(모든 메시지들이 verifier의 초기 지식으로 부터 simulate되었기 때문에,
verifier는 프로토콜로부터는 어떠한 것도 배우지 않는 것이다)

○ Zero knowledge proofs are simulatable
(conversation distributions are indistinguishable)



II.5 Provably Secure Multiparty Protocols

☐ Secure Multiparty Computation

○ 당사자들이 각자의 private input을 가지고, privacy와 correctness와 같은 보안성질이 유지되면서 그들의 input들에 대한 function 값을 함께 계산해내기를 원할 때 사용
예) 전자투표, 경매

- 당사자들의 일부가 프로토콜을 악의적으로 공격하더라도 보안성질은 유지되어야

III. Data Integrity Techniques

III.1 Provable Security for MAC

III.2 Provably Security for Digital Signatures

☐ Security Goal & Attack Model

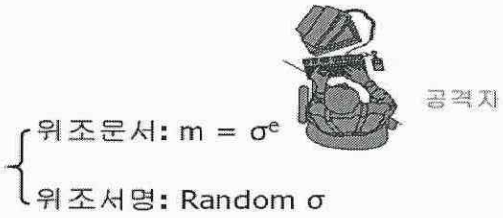
o Security Goal(Target)

- Total Break : Find private key
- Selective Forgery : Signature on selected message
- Existential Forgery : Signature on some message

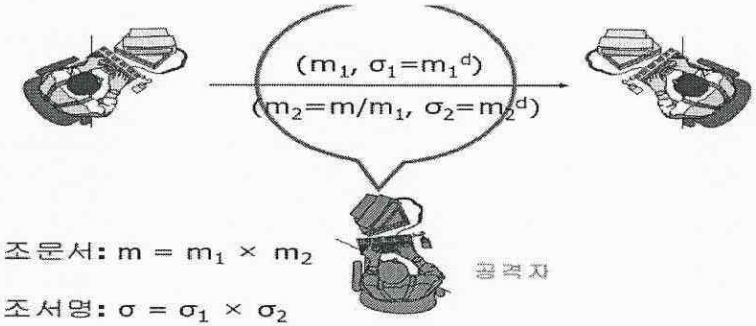
o Attack Model

- Key-Only Attack : 공개키만 가지고 서명 위조
- Known Message Attack : 공개키와 과거 서명값과 이에 대응하는 메시지를 알고 서명 위조
- Chosen Message Attack : 공개키와 임의로 선정한 메시지와 이에 대응하는 서명값을 알고 target message에 대한 서명 위조

☐ Existential Forgery - KOA



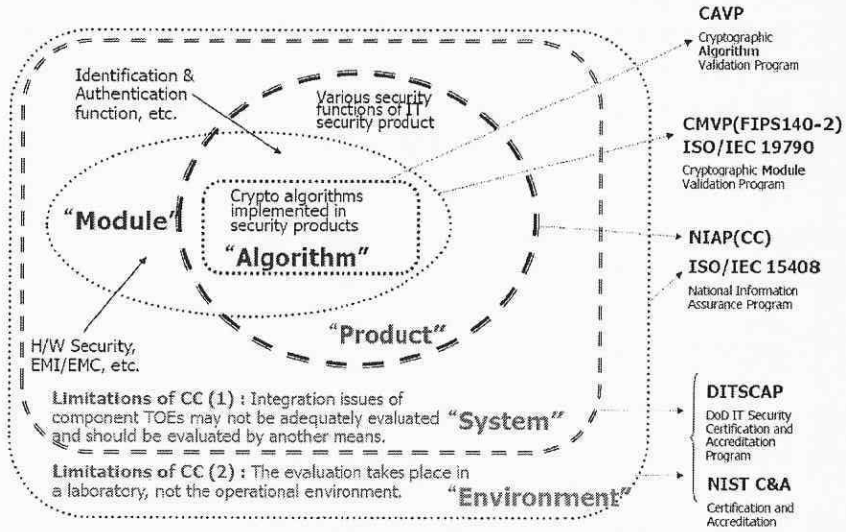
☐ Selective Forgery - CMA



Chapter. Security Evaluation

Step 4) Development & Evaluation

Security Implementation



<우리나라와 미국의 비교>

o 미국의 경우에는 누가 개발했는지 상관없이 일정기준을 충족하면 CAVP 리스트에 등재가 되나 우리나라는 국정원장이 승인한 알고리즘만 사용하므로 CAVP가 없다

o 미국의 CMVP는 CAVP를 통과한 알고리즘에 대해 암호모듈을 평가하나, 우리나라의 K-CMVP는 국정원장이 승인한 알고리즘에 대해 암호모듈을 평가한다.

o 우리나라는 2006.5월 CCRA 인증서 발행국으로 협약에 가입하였으나 CC제도가 국내용과 국제용으로 이원화되어 있으며, 국내용의 경우 평가보증등급에서 요구하는 평가항목 중 보안기능과 취약성시험을 제외한 일부 항목을 표본추출하여 평가하므로 Assurance가 훼손될 수 있다.

o 우리나라에는 C&A가 없다 : C&A는 CC, CMVP 등을 모두 포함하는 개념으로 산재된 모든 보안정책을 통합하여야 하나 현실적으로 불가능하다.

※ CC의 한계

① 단품 Product 위주의 평가로 Product가 통합된 전체 시스템을 평가할 수 없다.

② 평가과정이 운영환경이 아닌 실험실 환경에서 이루어져서 실제 환경과 차이가 발생한다.

○ Information Security 관점 : CAVP, CMVP, CC
- information과 information system을 보호, 단품 위주, 실험실 평가
○ Information Assurance 관점 : DITSCAP→DIACAP, NIACAP, NIST C&A
- business as a whole 보호, 인적보안 · 물리적보안 포함, 전체 lifecycle 평가

☐ IT Security Evaluation
○ IT 제품(H/W, F/W, S/W)의 security function들이 효과적이고 정확히 구현되었는지를 평가하고 인증하는 절차
○ 이를 위해, 개발자가 소프트웨어 엔지니어링 절차를 따랐는지(당초 의도한 대로 제대로 구현되었는지), 현재까지 알려진 모든 vulnerabilities에 대해 secure한 제품을 만들었는지(우회되지 않고 정확히 동작하는지) 점검

※ SSE-CMM : System Security Engineering Capability Maturity Model
(제품의 Quality가 아닌 제대로 된 제품을 개발할 수 있는 회사의 능력과 프로세스를 평가)

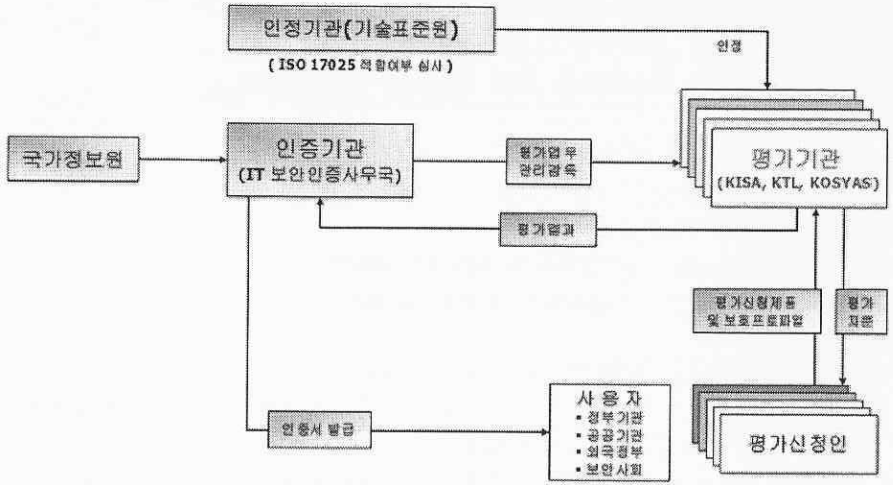
CC (Common Criteria)

☐ Terminology
○ Assurance(보증) : level of the trust that it really does
(해당 기능이 우회되지 않고 정확하게 동작하는 정도(확률))
○ Evaluation(평가) : process of determining the assurance level of a product
○ Validation(인증) : review of an IT security evaluation by an evaluation authority to determine if issuance of a CC Certificate is warranted

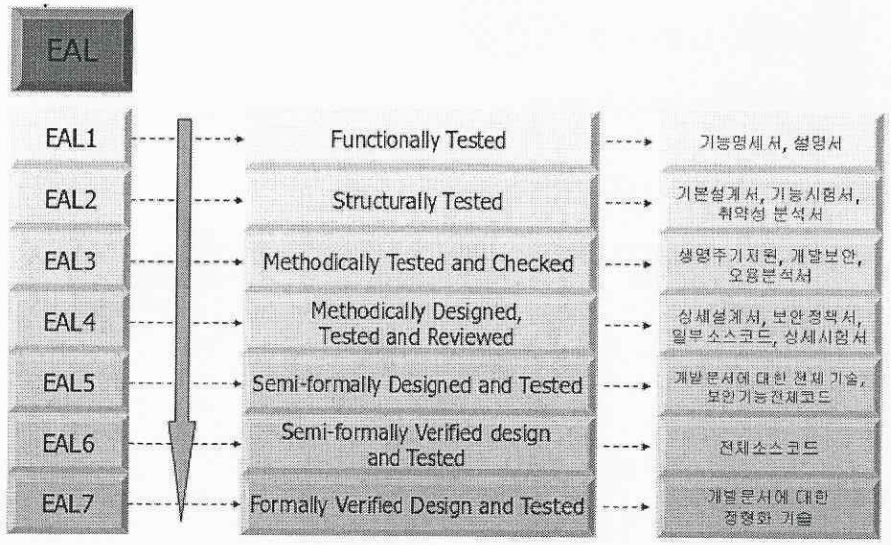
☐ What is the CC?
○ International Standard Meta-language for describing IT Security requirements
- 각종 제품의 평가기준을 만들기 위한 방법과 용어 정의
- 그것을 기준에 따라 평가하기 위한 평가방법론

☐ What is the CCRA?
○ EAL1부터 4까지의 CC평가를 인정하기 위한 CC 인증서 발행국과 인증서 수용국간의 국제적 협정(International agreement)
(EAL 5,6,7 등급은 상호인정에서 제외된다. 이는 미국에서 국가기밀을 다루는 등급으로 NSA가 관여하여 외산 제품에 정책 적용이 불가하기 때문이다.)
- IT product와 PP들의 중복 평가를 제거함으로써 벤더와 이용자의 시간과 자원을 절약
○ CAP(인증서 발행국) : Certificate Authorizing Participants
CCP(인증서 수용국) : Certificate Consuming Participants

☐ CCEVS(평가인증체계)



☐ CC Documents(CC 평가제출물)
○ You should prove that
- If product has been developed by using the S/W development models defined in SE and
- IT product is secure against known vulnerabilities

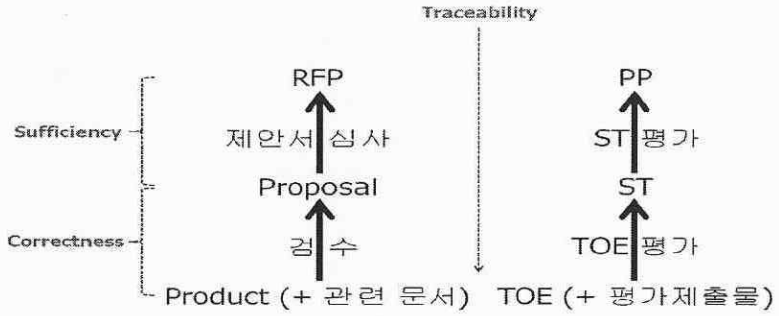


※ EAL1 등급의 평가제출물 : 기능명세서, 설명서 → 개발자가 아니더라도 신청 가능
왜 EAL1 등급이 필요한가? 평가를 받은 제품을 써야 한다고 규정된 경우.
회사가 사라질 경우에 대비

□ CC Structure
- ○ Part 1 : General Information
- ○ Part 2 : Catalogue of Functional Requirements
  - Security Function이 무엇인가?
- ○ Part 3 : Catalogue of Assurance Requirements
  - 얼마나 깊이, 엄밀하게 Security function이 평가되는가?
- ○ Common Criteria Evaluation Methodology (CEM)
  - 평가자가 평가업무를 어떻게 수행하는가?

□ TOE, PP and ST
- ○ TOE(Target Of Evaluation)
  - 평가의 대상이 되는 IT product 또는 시스템과 그와 관련된 administrator와 user의 guidance documentation
- ○ PP(Protection Profile, 보호프로파일)
  - CC에서 정의된 매우 특별한 방법으로 기술된 consumer의 security requirements의 목록
  - 구현 독립적
  - PP represents "I want"
- ○ ST(Security Target, 보안목표명세서)
  - product가 실제로 무엇을 하는지를 식별·명세하는 문서
  - 구현 종속적
  - ST represents "I provide"

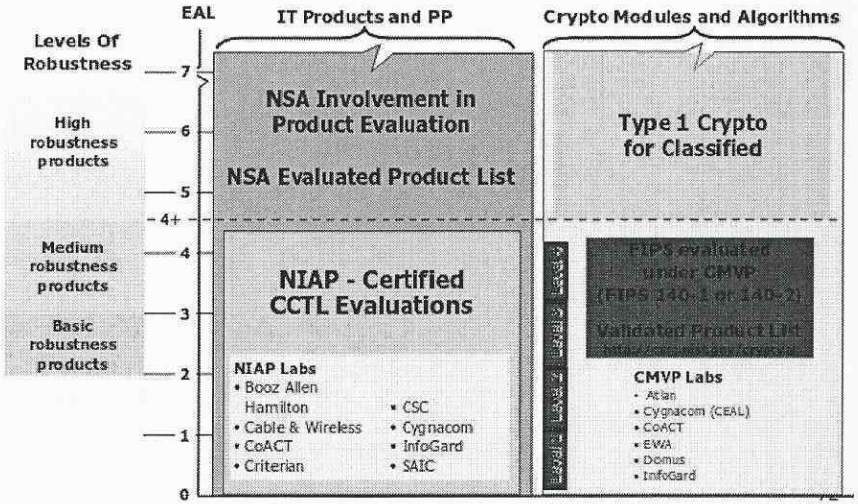
□ CC Evaluation Process



○ 충분성(sufficiency) : ST 평가 단계에서 자산의 위협에 대응하고 위협을 줄이기 위해 충분한 보안기능을 갖추고 있는지를 평가
○ 정확성(correctness) : TOE 평가 단계에서 구현된 보안기능이 우회되지 않고 정확하게 작동하는지를 평가
○ 추적성(traceability) : PP, ST, TOE 작성시 대응관계를 만족하도록 처음부터 끝까지 일관성 있게 작성
○ In TOE evaluation, you should prove that
- TOE has been developed by using the S/W development models defined in SE and TOE is secure against known vulnerabilities

□ EAL(Evaluation Assurance Level) : 해당 보안기능이 우회되지 않고 제대로 동작할 확률을 나타내는 등급(correctness와 관련)

□ EAL - CC vs. CMVP



□ 국제/국내용 CC제도 차이점
- ○ 국제용 평가인증제도
  - CCRA 수준의 제품 평가
  - 국제용 평가 인증서는 CCRA 회원국들 간에 상호 인정됨
- ○ 국내용 평가인증제도
  - 평가보증등급에서 요구하는 평가항목 중 보안기능과 취약성시험을 제외한 일부 항목을 표본추출하여 평가
  - 국내용 평가 인증서는 국내에서만 인정됨

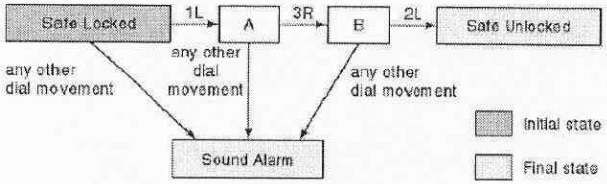
※ 국제용, 국내용 인증서 모두 국내에서는 동일한 효력을 가짐

중간고사 시험범위 中 발췌

☐ Types of Assurance

- o **Policy assurance** is evidence establishing
  security requirements in policy is complete, consistent, technically sound
- o **Design assurance** is evidence establishing
  design sufficient to meet requirements of security policy
- o **Implementation assurance** is evidence establishing
  implementation consistent with security requirements of security policy
- o **Operational assurance(= administrative assurance)** is evidence establishing
  systems sustains the security policy requirements
  during installation, configuration and day-to-day operation

☐ FSM Example



[State Transition Diagram]

Dial movement \ Current state	Safe locked	A	B
1L	A	Sound Alarm	Sound Alarm
1R	Sound Alarm	Sound Alarm	Sound Alarm
2L	Sound Alarm	Sound Alarm	Safe Unlocked
2R	Sound Alarm	Sound Alarm	Sound Alarm
3L	Sound Alarm	Sound Alarm	Sound Alarm
3R	Sound Alarm	B	Sound Alarm

Table of Next States

[Transition Table]

☐ Accountability

- o Audit information must be selectively kept and protected
  so that actions affecting security can be traced to the responsible party(책임 추적성)
  - 이를 위해, Audit information이 유지되고 보호되어야 하며, Access Control이 필요

☐ Nonrepudiation : Digital signature가 제공가능

- o Provide unforgeable evidence that a specific action occurred
  - Nonrepudiation of origin, delivery, receipt

☐ Dependability

- o **Dependability = Reliability (Accidental Failures)** */* 사람의 실수, 자연재해 등 */*
  + Security (Intentional Failures) */* 고의적 해킹 */*

☐ Survivability

- o recovery of the system after massive failure

☐ Design Decision Principles

① Focus of Control : Should protection focus on data, operations or users?

② The Man-Machine Scale : In which layer should security be placed?

③ Complexity vs Assurance : Should security focus on simplicity or security?
  - complexity와 assurance는 역의 관계(trade-off)를 갖는다.
  - assurance level이 높을수록(higher) 시스템은 더 간단(simpler)해져야 한다.

④ Centralized vs Decentralized : Should security control tasks
be given to a central entity or left to individual components?

※ 비교

Centralized	Decentralized
보안정책의 일관성을 유지하기가 용이	보안정책의 일관성을 유지하기 어려움
가용성이 좋지 않음(중단시 모두 중단)	가용성이 좋음

⑤ How to prevent the attacker from accessing the layer below the protection boundary?

☐ MS's STRIDE Threat Model

Spoofing : 공격자가 ID를 위변조하여 접근할 수 있는가?
Tampering : 공격자가 data를 위변조 할 수 있는가?
Repudiation : 공격자가 부정할 경우 그것이 잘못되었음을 증명할 수 있는가?
Information disclosure : 공격자가 사적인 또는 잠재적으로 해로운 데이터에 접근할 수 있는가?
Denial of service : 공격자가 시스템의 가용성을 훼손할 수 있는가?
Elevation of privilege : 공격자가 권한 상승을 할 수 있는가?
(privileged user의 id를 가장 할 수 있는가?)

☐ MS's DREAD Risk Analysis Model

Damage potential : 성공적 exploit의 결과(damage)는 무엇인가?

Reproducibility : exploit이 항상 일어나는가 또는 어떤 특정 환경 하에서만 발생하는가?

Exploitability : 공격자가 vulnerability를 이용하기 위해 skill이 얼마나 좋아야 하는가?

Affected users : 얼마나 많은 user들이 성공적 exploit에 영향을 받는가?

Discoverability : 공격자가 vulnerability의 존재를 얼마나 쉽게 발견할 수 있는가?

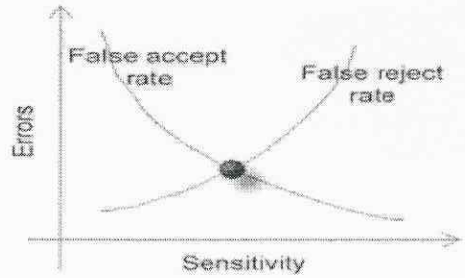
	High(3)	Medium(2)	Low(1)
Damage potential	공격자가 극도로 민감한 데이터에 접근하고 이를 차단·파괴할 수 있다.	공격자가 민감한 데이터에 접근할 수 있으나 그 밖의 행동은 드물다	공격자가 거의 해가 없는 데이터에만 접근할 수 있다.
Reproducibility	항상 발생	Timing 의존적 : time window에서만 동작	드물게 발생
Exploitability	누구나 할 수 있다.	얼마간의 지식과 skill이 필요하다.	전문화된 지식과 skill이 필요하다.
Affected users	거의 모든 유저	얼마간의 유저	거의 없다.
Discoverability	공격자가 vulnerability를 쉽게 발견할 수 있다.	공격자가 vulnerability를 발견할 수 있다.	공격자가 vulnerability를 발견하기 어렵다.

☐ Authentication(인증) v. Identification(개인식별)

- Authentication(=verification)
  - Verifying the user is actually who he says he is (or who she says she is)
  - Identity claim from the user
  - 1 : 1 matching
- Identification
  - you don't know anything about the person and you are trying to identify them
  - No identity claim
  - 1 : N matching

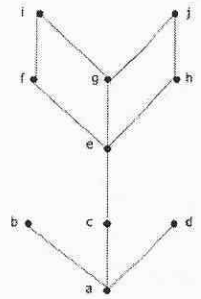
☐ Biometric Authentication Terms

- FRR : False Rejection Rate
  - Type I Error
  - when a biometric system rejects an authorized individual
- FAR : False Acceptance Rate
  - Type II Error
  - when a biometric system accepts an individual who should have been rejected
  - most dangerous and most important to avoid
- EER : Equal Error Rate
  - FAR = FRR인 point를 나타낸다. percentage로 표현된다.
  - 시스템의 accuracy를 결정하기 위해 중요한 상태
  - 더 낮은 값일수록 더 나은 accuracy를 나타냄
  - CER(Crossover Error Rate)라고도 함
  - 2개의 서로 다른 biometric system을 비교하는데 유용함



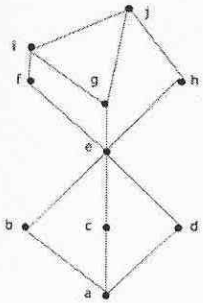
☐ Lattice

- A finite set together with a partial ordering on its elements
  such that for every pair of elements
  there is a least upper bound and a greatest lower bound
- Lattice가 필요한 이유
  - security level이 다른 2개의 object가 주어졌을 때, 두 object 모두를 읽기 위해 subject가 가져야 하는 minimal security level은 무엇인가?
  - security level이 다른 2개의 subject가 주어졌을 때, 두 subject 모두에게 읽힐 수 있기 위해 object가 가져야 하는 maximal security level은 무엇인가?



- (e.g.) Given two objects at different security levels b, c, what is the minimal security level a subject must have to be allowed to read both objects?

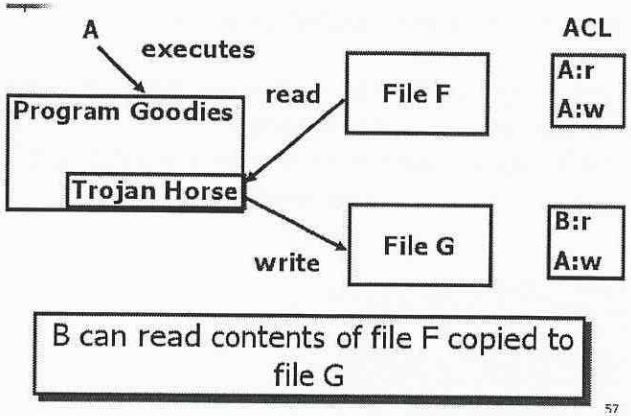
- lattice? No, because the pair {b, c} does not have a least upper bound
- Bounds, glb, lub of {c, e} ?
  - Lower bounds : {a, c}, thus glb is c
  - Upper bounds : {e, f, g, h, i, j}, thus lub is e
- Bounds, glb, lub of {b, i} ?
  - Lower bounds : {a}, thus glb is a
  - Upper bounds : ∅, thus lub does not exist



■ (e.g.) Given two objects at different security levels b, c, what is the minimal security level a subject must have to be allowed to read both objects?

○ lattice? Yes, for every pair of elements, there is a least upper bound and a greatest lower bound

☐ DAC Model Weaknesses



· Alice는 File F에 대해 읽기, 쓰기 권한이 있다.
· Alice는 File G에 대해 쓰기 권한이 있고, Bob은 File G에 대해 읽기 권한만 있다.
· Alice가 실행한 악성코드에 의해 File F의 정보가 File G에 기록되는 경우,
File F에 대한 정보가 File G로 유출되어 File F에 대해 권한 없는 Bob이 File f의 정보를 읽을 수 있다.

☐ Time-memory trade off → $N^{2/3}$로 줄임
· 5비트 문자셋, 6자리 패스워드 : 메모리 공간?
$(2^5)^6 = 2^{30}$ bits
· 얼마나 빨리 찾을 수 있는가? a trial encryption take one millisecond
$(2^{30})^{2/3} \times 10^{-3}$ 초 = $2^{20} \times 10^{-3}$초 = 1048.576초