* **Bell-LaPadula (BLP) Model**

Subjects와 Objects는 보안수준이 매겨진다. Subject는 주어진 수준의 보안 허가를 가지고 있고, Object는 주어진 수준의 보안 분류를 가진다. 보안 수준은 Subject가 Object에 접근할 수 있는 방식을 제어한다.

BLP 모델은 Read, Append, Write 그리고 Execute 네 가지 Access Mode를 정의한다.

Read: Subject는 Object를 읽는 것 만이 허용된다.  
Append: Subject는 Object에 쓰는 것 만이 허용된다.  
Write: Subject는 Object에 읽기 또는 쓰기가 허용된다.  
Execute: Subject는 Object를 실행을 위한 호출만 할 수 있다.

여러 범주 또는 데이터 수준이 정의되면 이 요구 사항을 Multilevel Security (MLS)라고 한다.

No Read Up: Subject는 보안 수준이 낮거나 같은 Object만 Read 할 수 있다. (SS-property)

No Write Down: Subject는 보안 수준이 같거나 높은 Object만 Write할 수 있다. 왜냐하면 높은 수준인 Object의 정가 낮은 수준인 Object에 쓰여서는 안되기 때문이다. (\*-property)

DS property: Subject는 특정 사용 권한을 Subject가 자체 재량에 따라 권한을 전달할 수 있음을 나타낸다.

* **BLP Formal Description**

은 Subject 가 현재 Object 에 Access Mode 로 접근을 행사하고 있음을 의미한다.

는 의 보안 분류의 수준, 는 의 보안 허가의 수준, 그리고 는 의 현재 보안 수준이다.

SS-property:

\*-property:

DS-property

* **Limitations to the BLP Model**

정보의 무결성을 배제하고 기밀성만을 고려함으로써 임의의 Subject에 의한 정보의 불법 변경 가능성에 대한 해결책을 제시하지 못한다. 왜냐하면 read-down과 writ-up에 대한 언급은 없으므로 낮은 수준의 Subject가 높은 수준의 Object의 읽기 행위는 허용되지 않지만 쓰기 행위를 할 수도 있기 때문이다.

* **Biba Integrity Model**

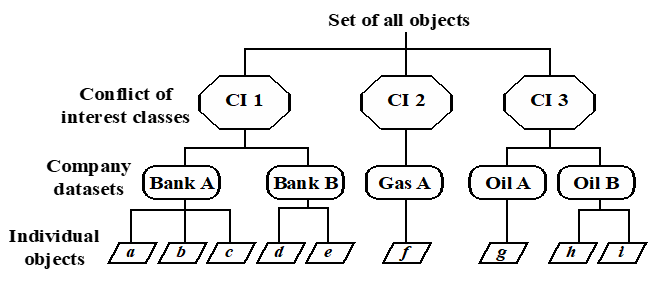
Biba 모델은 Modify, Observe, Execute 그리고 Invoke 네 가지 Access Mode를 정의한다. 처음 세 가지 Mode는 BLP 모델의 Access Mode와 유사하다. Invoke Mode는 Subject가 다른 한 Subject와 통신하는 것이다.

Simple Integrity (No-write-up Policy): Subject는 무결성 수준이 자신과 같거나 낮은 Object만 Modify 할 수 있다. ()

Integrity Confinement (No-read-down-Policy): Subject는 무결성 수준이 자신과 같거나 높은 Object만 읽을 수 있다. ()

Invocation Property: Subject는 무결성 수준이 자신보다 낮은 Subject와만 통신할 수 있다.

* **Chinese Wall Model**



CWM의 항목은 Subject와 Information으로 구분된다. Information은 Object, Dataset 그리고 Conflict of Interest의 3가지 단계로 구분된다. CWM은 Subject나 Object에 안전 수준을 정하지 않는다. 즉, Multilevel Secure Model이 아니다.

대신 이전에 접근한 대상의 기록에 따라 접근 제어가 결정됩니다. CWM의 기본 정책은 Subject가 이미 소유하고있는 다른 정보와 충돌하지않는 정보에만 접근이 허용된다는 것이다.

Simple Security Rule: Subject 는 Object 를 가 이미 에 의해 접근된 Object와 동일한 DS에 있거나, 가 아직 어떤 정보에도 접근하지 않은 CI에 속할 때 Read할 수 있다. 따라서 같은 CI에 속하지만 다른 DS에 있는 Object는 Read할 수 없다.

\*-property Rule: Subject 는 Object 를 가 SS Rule에 의해 읽을 수 있고 가 읽을 수 있는 모든 가 동일한 DS에 있을 때 Write할 수 있다.

* **Buffer Overflow**

할당된 용량보다 많은 입력을 버퍼 또는 데이터 보관 영역에 배치하면서 다른 정보를 덮어는 현상이다. 공격자는 이러한 상황을 악용하여 시스템을 손상시키거나 특수하게 조작된 코드를 삽입하여 시스템을 제어할 수 있다.

* **Division**

If and are integers with , we say that *divides* if there is an integer so that . When divides we say that is a *factor* of and that is a *multiple* of .

* **Divisibility Theorems**

For integers , , and it is true that  
if and , then   
if , then for all integers   
if and , then

* **Primes**

A positive integer greater than 1 is called prime if the only positive factors of are 1 and . A positive integer that is greater than 1 and is not prime is called *composite*.

Every positive integer can be written uniquely as the product of primes, where the prime factors are written in order of increasing size.

If is a composite integer, then has a prime divisor less than or equal .

* **The Division Algorithm**

Let be an integer and a positive integer. Then there are unique integers and , with , such that .

is called the divisor, is called the dividend, is called the quotient, and is called the remainder.

* **Greatest Common Divisors**

Let and be integers, not both zero. The largest integer such that and is called the GCD of and .

* **Least Common Multiples**

The least common multiple of the positive integers and is the smallest positive integer that is divisible by both and .

* **Relatively Prime**

Two integers and are relatively prime if .

* **Congruences**

Let and be integers and be a positive integer. We say that is congruent to modulo if divides .

* **The Euclidean Algorithm**

The Euclidean Algorithm finds the greatest common divisor of two integers and .

* **Extended Euclidean Algorithm**

For given integers and , the extended Euclidean algorithm not only calculate the GCD , but also two additional integers and that satisfy the above equation.

Example: Find which satisfies the following equation:

* **Fermat’s Theorem**

if is prime and is a positive integer not divisible by then

* **Euler’s Theorem**

States that for every and that are relatively prime:

* **Miller-Rabin Test**

Let be an odd prime and write with odd. Let be any number not divisible by . Then, one of the following two conditions is true.

1. is congruent to 1 modulo .
2. One of , , , … is congruent to -1 modulo .

Example: Determine if 100003 is a prime number.

Example: Determine if 100005 is a prime number.

* **Chinese Remainder Theorem**

When the number of equations is more than 2,

Then, we have

Example: Solve the following simultaneous congruence equation for .