- **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**

  (S, O, A)은 Subject S가 현재 Object O에 Access Mode A로 접근을 행사하고 있음을 의미한다.

  $f_o(O)$는 O의 보안 분류의 수준, $f_s(S)$는 S의 보안 허가의 수준, 그리고 $f_c(S)$는 S의 현재 보안 수준이다.

  SS-property:

  $$(S_i, O_j, \text{read}) \text{ has } f_c(S_i) \geq f_o(O_j)$$

  *-property:

  $$(S_i, O_j, \text{append}) \text{ has } f_c(S_i) \leq f_o(O_j)$$

  $$(S_i, O_j, \text{write}) \text{ has } f_c(S_i) = f_o(O_j)$$

  DS-property

  $$(S_i, O_j, A_x) \text{ implies } A_x \in M[S_i, O_j]$$

- **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**

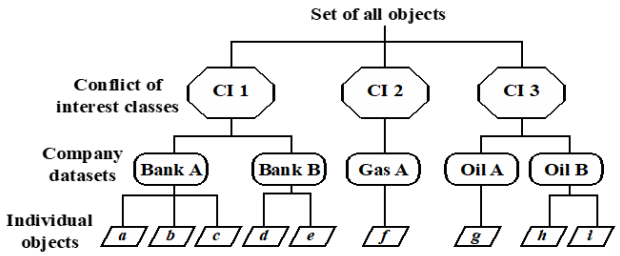
  Biba 모델은 Modify, Observe, Execute 그리고 Invoke 네 가지 Access Mode를 정의한다. 처음 세 가지 Mode는 BLP 모델의 Access Mode와 유사하다. Invoke Mode는 Subject가 다른 한 Subject와 통신하는 것이다.

  Simple Integrity (No-write-up Policy): Subject는 무결성 수준이 자신과 같거나 낮은 Object만 Modify 할 수 있다. ($I(S) \geq I(O)$)

  Integrity Confinement (No-read-down-Policy): Subject는 무결성 수준이 자신과 같거나 높은 Object만 읽을 수 있다. ($I(S) \leq I(O)$)

  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 S는 Object O를 O가 이미 S에 의해 접근된 Object와 동일한 DS에 있거나, S가 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 S는 Object O를 O가 SS Rule에 의해 읽을 수 있고 S가 읽을 수 있는 모든 O가 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**

$$a|b$$

If a and b are integers with $a \neq 0$, we say that a *divides* b if there is an integer c so that $b = ac$. When a divides b we say that a is a *factor* of b and that b is a *multiple* of a.

- **Divisibility Theorems**

For integers a, b, and c it is true that
if $a|b$ and $a|c$, then $a|(b + c)$
if $a|b$, then $a|bc$ for all integers c
if $a|b$ and $b|c$, then $a|c$

- **Primes**

A positive integer p greater than 1 is called prime if the only positive factors of p are 1 and p. 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 n is a composite integer, then n has a prime divisor less than or equal $\sqrt{n}$.

- **The Division Algorithm**

Let a be an integer and d a positive integer. Then there are unique integers q and r, with $0 \leq r < d$, such that $a = dq + r$.

d is called the divisor, a is called the dividend, q is called the quotient, and r is called the remainder.

- **Greatest Common Divisors**

$$\gcd(a, b)$$

Let a and b be integers, not both zero. The largest integer d such that $d|a$ and $d|b$ is called the GCD of a and b.

- **Least Common Multiples**

$$\gcd(a, b)$$

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

- **Relatively Prime**

Two integers a and b are relatively prime if $\gcd(a, b) = 1$.

- **Congruences**

$$a \equiv b \bmod m \rightarrow a \bmod m = b \bmod m$$

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

- **The Euclidean Algorithm**

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

When $a > b, a = qb + r \rightarrow \gcd(a, b) = \gcd(b, r)$
if $c = \gcd(a, b)$, then $c|r$. Thus, $c \leq \gcd(b, r)$.
if $d = \gcd(b.r)$, then $d|a$. Thus, $d \leq \gcd(a, b)$.
$\therefore \gcd(a, b) = \gcd(b, r)$

- **Extended Euclidean Algorithm**

$$ax + by = d = \gcd(a, b)$$

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

Example: Find x which satisfies the following equation:

$$123x \equiv 1 \bmod 158$$

$123x \equiv 1 \bmod 158 \rightarrow 123x - 158y = 1$
$\gcd(158,123) \rightarrow 158 = 123 \cdot 1 + 35$
$\gcd(123,35) \rightarrow 123 = 35 \cdot 3 + 18$
$\gcd(35,18) \rightarrow 35 = 18 \cdot 1 + 17$
$\gcd(18,17) \rightarrow 18 = 17 \cdot 1 + 1 \rightarrow 17 = \frac{18 - 1}{1}$
$35 = 18 \cdot 1 + \frac{18 - 1}{1} \rightarrow 35 = 18 \cdot 2 - 1 \rightarrow 18 = \frac{35 + 1}{2}$
$123 = 35 \cdot 3 + \frac{35 + 1}{2} \rightarrow 123 \cdot 2 = 35 \cdot 7 + 1 \rightarrow 35 = \frac{123 \cdot 2 - 1}{7}$
$158 = 123 \cdot 1 + \frac{123 \cdot 2 - 1}{7} \rightarrow 158 \cdot 7 = 123 \cdot 9 - 1$
$\therefore 123 \cdot 9 - 158 \cdot 7 = 1$

- **Fermat's Theorem**

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

$$a^{p-1} \equiv 1 \bmod p \text{ or } a^p \equiv a \bmod p$$

- **Euler's Theorem**

States that for every a and n that are relatively prime:

$$a^{\phi(n)} \equiv 1 \bmod n \text{ or } a^{\phi(n)+1} = a \bmod n$$

- **Miller-Rabin Test**

Let p be an odd prime and write $p - 1 = 2^k q$ with q odd. Let a be any number not divisible by p. Then, one of the following two conditions is true.

1. a^q is congruent to 1 modulo p.

2. One of a^q, a^{2q}, a^{4q}, ... $a^{2^{k-1}q}$ is congruent to -1 modulo p.

Example: Determine if 100003 is a prime number.

$p - 1 = 100002 = 2^1 \cdot 50001; a = 2 \rightarrow 2^{50001} \bmod 100003 \neq 1$
$j = 0; a^{2^j q} = 2^{2^0 \cdot 50001} \rightarrow 2^{50001} \bmod 100003 = 10002$
$\therefore 100003$ is probably prime

Example: Determine if 100005 is a prime number.

$p - 1 = 100004 = 2^2 \cdot 25001; a = 2 \rightarrow 2^{25001} \bmod 100005 \neq 1$
$j = 0; a^{2^j q} = 2^{2^0 \cdot 25001} = 2^{25001} \rightarrow 2^{25001} \bmod 100005 \neq 100000$
$j = 1; a^{2^j q} = 2^{2^1 \cdot 25001} = 2^{50002} \rightarrow 2^{50002} \bmod 100005 \neq 100000$
$\therefore 100005$ is not prime

- **Chinese Remainder Theorem**

  When the number of equations is more than 2,

$$\begin{cases} x \equiv a_1 \mod m_1 \\ x \equiv a_2 \mod m_2 \\ \quad\vdots \\ x \equiv a_k \mod m_k \end{cases}$$

$$M = m_1 \cdot m_2 \cdot \cdots \cdot m_k; M_i = M/m_i$$

  Then, we have

$$x \equiv \sum_{i=1}^{k} a_i \cdot M_i \cdot (M_i^{-1} \bmod m_i) \bmod M$$

  Example: Solve the following simultaneous congruence equation for x.

$$\begin{cases} x \equiv 3 \mod 5 \\ x \equiv 2 \mod 7 \end{cases}$$

$$M = 5 \cdot 7 = 35$$
$$x \equiv \sum_{i=1}^{k} a_i \cdot M_i \cdot (M_i^{-1} \bmod m_i) \bmod 35$$
$$= \left(3 \cdot 7 \cdot (\mathbf{3} \cdot 7 \bmod 5 = 1) + 2 \cdot 5 \cdot (\mathbf{3} \cdot 5 \bmod 7 = 1)\right) \bmod 35$$
$$= (3 \cdot 7 \cdot 3 + 2 \cdot 5 \cdot 3) \bmod 35 = 93 \bmod 35 = 23$$

-