Euclid's Algorithm.

This is a very fast method to compute gcd of two integers.

Lemma: Let l & m be two integers s.t. $l = qm + r$ $(0 \le r < m)$, q & r are integers, & $l \ge m$ then $\underline{\gcd(l,m)} = \underline{\gcd(m,r)}$

Proof: Let d be a common divisor of l & m.

$\Rightarrow$ $l - qm$ is divisible by d.

$\Rightarrow$ r is divisible by d

$\Rightarrow$ d is a common divisor of r & m.

Now, let d be a common divisor of m & r

$\Rightarrow$ d is a divisor of $qm + r$

$\Rightarrow$ d " " " l

$\Rightarrow$ d is a common divisor of l & m.

$\Rightarrow$ (l,m) & (m,r) have the same set of common divisors

$\Rightarrow$ $\gcd(l,m) = \gcd(m,r)$

Euclid's Algorithm: Let a, b be two integers such that $a \ge b$ then

$\exists$ q_1 & r_1 non-negative integers s.t.

$$a = bq_1 + r_1 \qquad (0 \le r_1 < q_1)$$

$$b = r_1 q_2 + r_2 \qquad (0 \le r_2 < q_2)$$

$$r_1 = r_2 q_3 + r_3 \qquad (0 \le r_3 < q_3)$$

$$\vdots$$

$$r_{n-1} = \boxed{r_n} q_{n+1} + 0$$

$$\gcd(a,b) = \gcd(b, r_1) = \gcd(r_1, r_2) = \cdots = \gcd(r_{n-1}, r_n)$$
$$= r_n \quad [\text{last nonzero remainder}$$

$$\boxed{\gcd(a,b) = r_n}$$

Ex: $a = 273, \quad b = 156$

$273 = 1 \times 156 + 117$
$156 = 1 \times 117 + \boxed{39}$
$117 = 3 \times 39 + 0$

$\gcd(273, 156) = 39$

Note: If $\gcd(a,b) = r$ then $\exists$ integers p & q s.t.

$$pa + qb = r$$

If $\gcd(a,b) = 1 = pa + qb$

$1 \equiv pa \bmod b$ or $\bmod q$

$\Rightarrow a^{-1} \equiv p \bmod b$ or $a^{-1} \equiv p \bmod q$.

If $\gcd(a,b) \neq 1$ then $a^{-1} \bmod b$ will not exists.

$\Rightarrow$ For the existence of $a^{-1} \bmod b$, a & b should be coprime.

$k = 7$

$7^{-1} \bmod 26$

$26 = 3 \times 7 + 5$
$7 = 1 \times 5 + 2$
$5 = 2 \times 2 + \boxed{1}$ $\rightarrow$

$5 = 26 - 3 \times 7$ — ①
$2 = 7 - 1 \times 5$ — ②
$1 = 5 - 2 \times 2$
$= 5 - 2 \times (7 - 1 \times 5)$
$= 5 - 2 \times 7 + 2 \times 5$
$= 3 \times 5 - 2 \times 7 = 3 \times (26 - 3 \times 7) - 2 \times 7$

$= 3 \times 26 - 11 \times 7$

$$1 = 3\times 26 - 11\times 7$$

$\Rightarrow \quad 1 \equiv -11\times 7 \bmod 26$

$\Rightarrow \quad 7^{-1} \equiv -11 \bmod 26 = (26-11) \bmod 26 = 15 \bmod 26$

$\Rightarrow \quad 7^{-1} \bmod 26 = \underline{\underline{15}}$.

Inverse of 7 in $(\mathbb{Z}_{26}^{*}, \times)$ is 15.

Chinese Remainder Theorem

Consider the following system

$$\left.\begin{aligned} x &\equiv a_1 \bmod m_1 \\ x &\equiv a_2 \bmod m_2 \\ &\vdots \\ x &\equiv a_k \bmod m_k \end{aligned}\right\} \quad (1)$$

where $m_1, m_2, \ldots, m_k$ are pairwise coprime.

$$\gcd(m_i, m_j) = 1, \quad i \neq j, \ \forall i, j.$$

Then, (1) has a unique solution modulo $m_1 m_2 \cdots m_k = M$ and the solution is

$$x = \left[\sum_{i=1}^{k} a_i \left(\frac{M}{m_i}\right) \left\{ \left(\frac{M}{m_i}\right)^{-1} \bmod m_i \right\} \right] \bmod M$$

Ex:

$$\left.\begin{aligned} x &\equiv 2 \bmod 3 \\ x &\equiv 3 \bmod 5 \\ x &\equiv 2 \bmod 7 \end{aligned}\right\} \quad (1)$$

$m_1 = 3, \ m_2 = 5, \ m_3 = 7$ are pairwise coprime.

$M = 105$

$$\frac{M}{m_1} = 35, \quad \frac{M}{m_2} = 21, \quad \frac{M}{m_3} = 15$$

$$\left(\frac{M}{m_1}\right)^{-1} \bmod m_1 = 35^{-1} \bmod 3 = 2$$

$$\left(\frac{M}{m_2}\right)^{-1} \bmod m_2 = 21^{-1} \bmod 5 = 1$$

$$\left(\frac{M}{m_3}\right)^{-1} \bmod m_3 = 15^{-1} \bmod 7 = 1$$

$$x \equiv (2 \times 35 \times 2 + 3 \times 21 \times 1 + 2 \times 15 \times 1) \bmod 105 = \underline{23 \bmod 105}$$

Asymmetric Key Cryptography:

Alice

Key: $(K^{(A)}_{public}, K^{(A)}_{private})$

Bob

Key: $(K^{(B)}_{public}, K^{(B)}_{private})$

$$P \rightarrow \boxed{\text{Encryption}} \xleftarrow{} K^{(B)}_{public}$$

$$\boxed{\text{Encryption}} \xrightarrow{C} \boxed{\text{Insecure Channel}} \xrightarrow{C} \boxed{\text{Decryption}} \rightarrow P$$

$$K^{(B)}_{private} \rightarrow \boxed{\text{Decryption}}$$

* If a message is encrypted using $K^{(R)}_{public}$ then it can only be decrypted using $K^{(R)}_{private}$.

* $(K_{public}, K_{private})$ will be generated using a key generation process.

* AKC is based on personal secrecy.

* Plaintext and Ciphertext are integers.

* Encryption & decryption in AKC are mathematical functions.

$$C = f(K_{public}, P) \quad \& \quad P = g(K_{private}, C)$$

* Here, f is a trapdoor one-way function.

* AKC is slower than the SKC.

* AKC is needed for authentication, digital signatures, & in secret key exchanges.

* AKC & SKC complements each other.

One-way function: A function f which satisfies

(1) $f(x)$ is easy to compute i.e. For given x it is easy compute $y=f(x)$

(2) f^{-1} is difficult to find. i.e. for a given y it is very hard to find x s.t. $x=f^{-1}(y)$

Ex: $n = \underbrace{p \times q}$ - p & q are very large primes.

$f(p,q) = p \times q = n$

$\Rightarrow$ finding $f(p,q)$ is very easy for given p & q.

But it is very difficult to find p & q when n is given.

There is no algorithm which can find p & q in polynomial time when n is a very large number.

$\Rightarrow$ $f(p,q)$ is a one-way function.

Trapdoor One-way function: A one-way function f which satisfies

(1) Given a y and a trapdoor (secret) x can be computed easily.

Ex: Let n be a large number.

$$y \equiv \underbrace{x}^{k} \bmod n$$

When x, k & n are given, calculating y is easy.

" y, k & n " " then it is very difficult to compute x.

Suppose k' is such that

$$\underbrace{kk'}\equiv 1 \bmod \phi(n)$$

Then, $y^{k'} \equiv (x^k \bmod n)^{k'}$

$$\equiv (x^k)^{k'} \bmod n \equiv x^{k\,k'} \bmod n$$

$$\equiv x^{1 \bmod \phi(n)} \bmod n$$

$$\equiv x^{m\times\phi(n)+1} \bmod n \quad , \quad m\in\mathbb{Z}$$

$$y^{k'} \equiv x \bmod n \qquad \{\text{ by Euler's theorem}$$

$$\Rightarrow \underline{x} \equiv \underbrace{y^{k'}}_{\text{Easy to calculate}} \bmod n$$

$f(x) = x^k \bmod n$ is a trap door one-way function

RSA Cryptosystem

R - Rivest
S - Shamir
A - Adleman

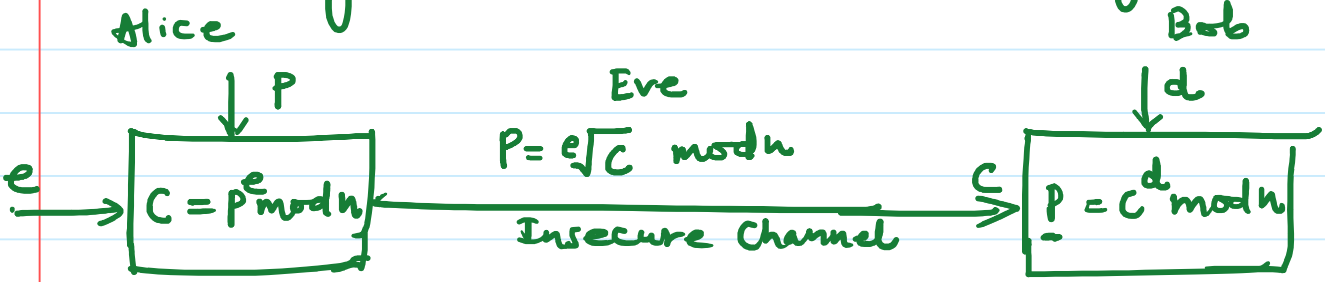
* RSA is the most popular public-key Cryptosystem.

* It uses two number e & d where e is public and d is private.

P - plaintext, C - Ciphertext, n - very large integer.

Encryption: $C = P^e \bmod n$
Decryption: $P = C^d \bmod n$

Calculating p^e & c^d mod n is easy.



Computing $\underbrace{\sqrt[e]{C}}$ is very difficult because the complexity of this problem is exponential.

Key Generation

1. Choose large primes p & q.
2. $n = pq$
3. $\phi(n) = \phi(p) \cdot \phi(q) = (p-1)(q-1)$
4. Choose $e = \{1, 2, \ldots, \phi(n)-1\}$
   s.t. $\gcd(e, \phi(n)) = 1$
5. Compute $K_{private} = d$ by

$$d \cdot e \equiv 1 \bmod \phi(n)$$

$K_{private} = d \equiv e^{-1} \bmod \phi(n)$

$K_{public} = (n, e)$, $K_{private} = d$

<u>Proof of RSA</u>: Let plaintext retrieved by Bob is P_1

$$\Rightarrow P_1 = c^d \bmod n = (p^e \bmod n)^d \bmod n$$
$$= p^{ed} \bmod n$$

$\because$ $ed = 1 \bmod \phi(n) = k\phi(n) + 1$ where $k \in \mathbb{Z}$

$\therefore$ $P_1 = p^{k\phi(n)+1} \bmod n$

$P_1 \equiv P \bmod n$

<u>Ex</u>: $p = 7$, $q = 11$

$n = 77$, $\phi(n) = 6 \times 10 = 60$

choose e s.t. $e \in \{1, 2, \ldots, \phi(n)-1\}$ & $\gcd(e, \phi(n)) = 1$

$e \in \{1, 2, \ldots, 59\}$ & $\gcd(e, \phi(n)) = 1$.

Let $e = 13$

Now, $d \equiv e^{-1} \bmod \phi(n) = 13^{-1} \bmod 60$

$d \equiv \underline{37 \bmod 60}$.

plaintext $P = 5$

Then, encryption : $C = p^e \bmod n = 5^{13} \bmod 77 = 26 \bmod 77$

Decryption : $P = 26^{37} \bmod 77 = 5 \bmod 77$.