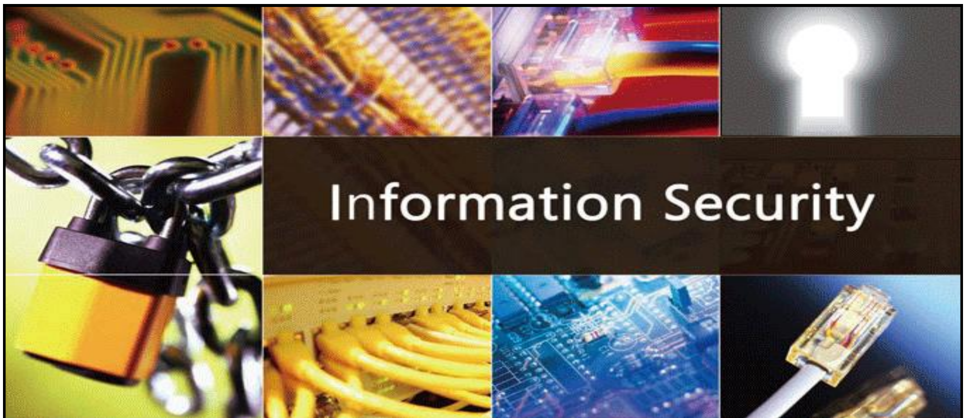


Asymmetric encryption and Key management

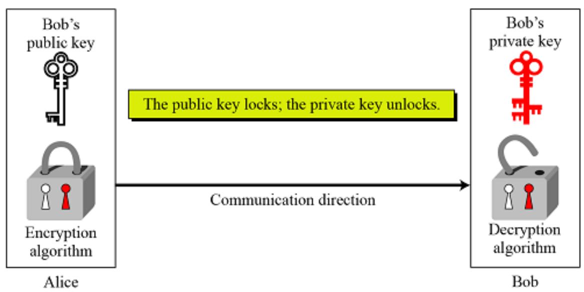
Lecturer: Nguyễn Thị Thanh Vân – FIT - HCMUTE

Objective

- Asymmetric encryption
- Modular arithmetic
- RSA
- Key Management
  - Symmetric-key distribution
  - KERBEROS
  - Symmetric-key agreement: Diffie-Hellman
  - Public-key distribution: CA, X.509
-

Asymmetric encryption

- *Asymmetric encryption* is a form of cryptosystem in which *encryption* and *decryption* are performed using the different keys
  - a public key
  - a private key.
- It is also known as *public-key encryption*



Public – key Cryptography

- Cryptography with **public – key/2 keys/asymmetric** uses **TWO** keys that have one owner**:**
  - **Public - key**,
    - <u>everyone can know </u>and
    - use to **encrypt the message** or
    - to **check the signature** of key's owner.
  - **Private – key**:
    - only <u>owner knows </u>and
    - use to **decrypt the message** or
    - to **create the signature**
- **Public key:**
  - can be calculated from **private key** and other information of cryptography (**P problem**)
  - needs to <u>be distributed safely for everyone</u>, who needs securely send message to key's owner
  - Problem of public key distribution is important – that is **key distribution problem**

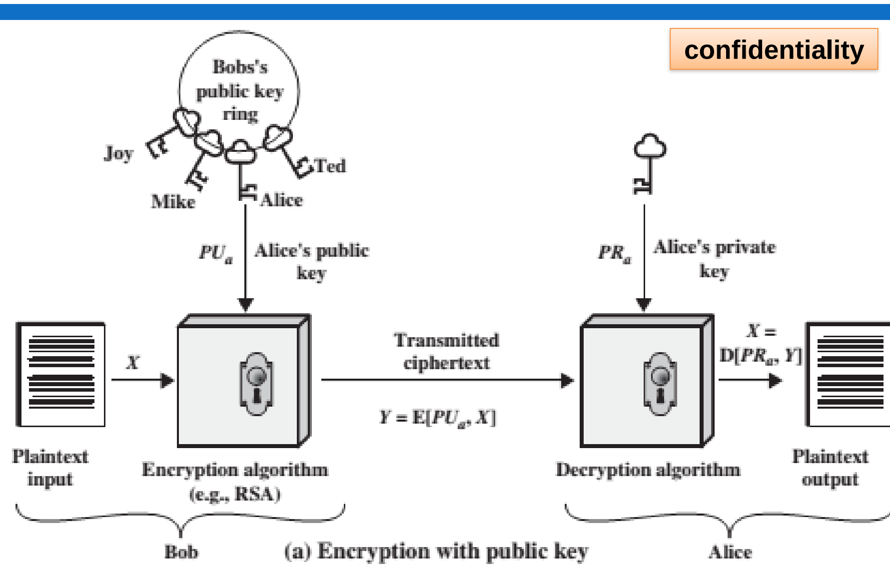
Public – key Cryptography

- In **asymmetric cryptography,** role of sender and recipient are **not** same:
  - *Person who encrypt message either check the signature*
  - *that can not be decrypted or create the signature.*
- Mathematical basis: *One-way functions*
  - ***y = f(x) is the one – way function if y = f(x) is easy to calculate but x = $f^{-1}(y)$ is difficult to find***
  - *x = $f^{-1}(y)$ might be easy to calculate if given additional information (key)*
- *Ex1:* When n is large,
  - n = p × q is a one-way function. Given p and q , it is always easy to calculate n; given n, it is very difficult to compute p and q. This is the factorization problem.
  - y = x^k mod n is a trapdoor one-way function. Given x, k, and n, it is easy to calculate y. Given y, k, and n, it is very difficult to calculate x. This is the discrete logarithm problem. However, if we know the trapdoor, k′ such that k × k ′ = 1 mod f(n), we can use x = $y^{k'}$ mod n to find x.

5

Encryption with public key

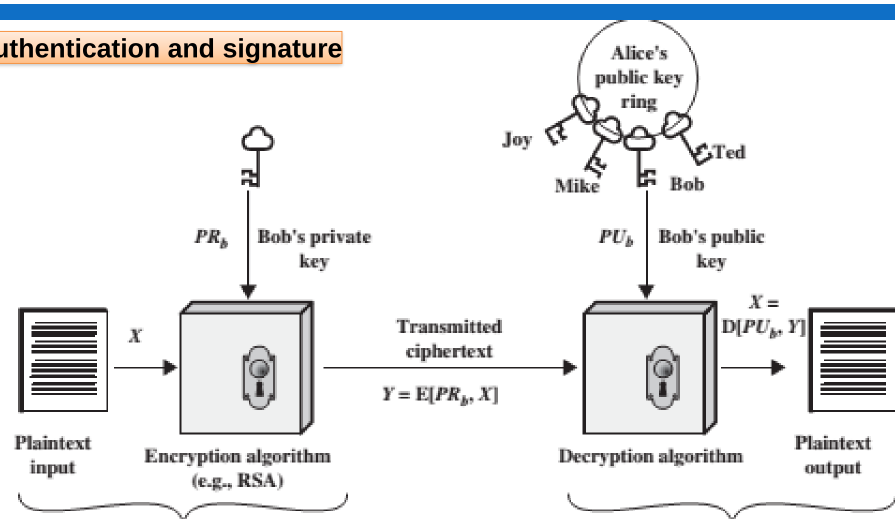
confidentiality



(a) Encryption with public key

Encryption with private key

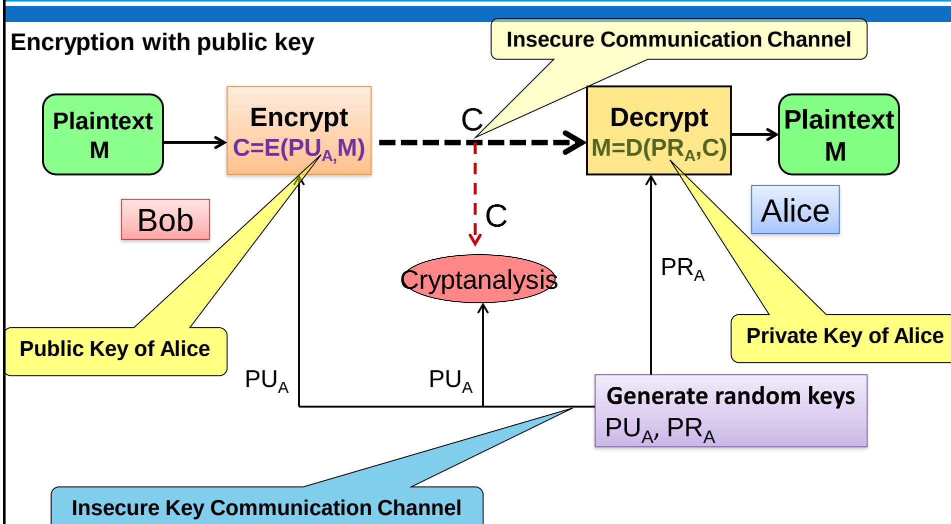
authentication and signature



(b) Encryption with private key

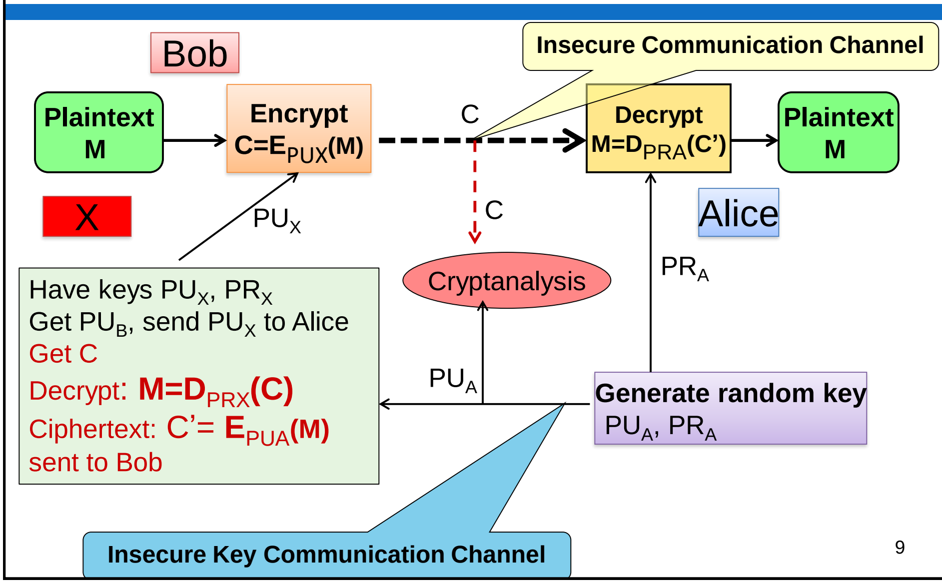
Public – key Cryptography: Model

Encryption with public key



8

Public – key Cryptography: Model



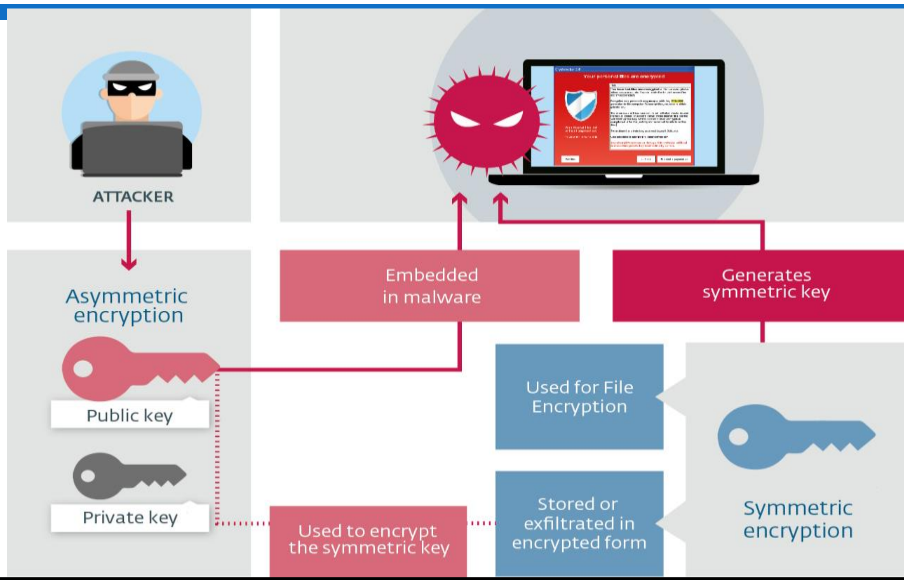
9

Public key cryptography Security

- Security based on **the difference** between the hardness of encryption/decryption problem (**easy**) and cryptanalysis problem (**hard**)
- Cryptanalysis using **key exhaustive key search** is always done theoretically. But in fact, the number of used keys is too large for it (>512 bit)
- To resist some other **advanced cryptanalysis methods**, need to use **the very large keys** (>>512 bit)
- Therefore implementation of public key cryptography is much slower than the secret key cryptography

10

Ex, Symmetric /Asymmetric encryption



Mathematics of Cryptography

Lecturer: Nguyễn Thị Thanh Vân – FIT - HCMUTE

Mathematics of Cryptography

- Primes
- Euler's Phi-Function
- Euclidean Algorithm in GCD
- Modulo,
  - Congruence modulo
  - Properties (addition, subtraction, multiplication, exponentiation)
  - Fermat's Little Theorem
  - Euler's Theorem
  - Modular inverse (additive, multiplicative)
- Extended-Euclid

Primes

- Asymmetric-key cryptography uses primes extensively
- A positive integer is a prime if and only if it is exactly divisible by two integers, 1 and itself..
- Ex: List the primes smaller than 10.
- Infinite Number of Primes

$$\boxed{[n / (\ln n)] \quad < \quad \pi(n) \quad < \quad [n/(\ln n - 1.08366)]}$$

- Checking for Primeness
  - Given a number n, how can we determine if n is a prime? $\sqrt{n}$
  - Ex: Is 97 a prime?
    - The floor of $\sqrt{97}$ = 9. The primes less than 9 are 2, 3, 5, and 7. We need to see if 97 is divisible by any of these numbers. It is not, so 97 is a prime.

Primes

```
Divisibility_Test (n)                    // n is the number to test for primality
{
  r ← 2
  while (r < √n )
  {
   if (r | n) return "a composite"
   r ← r + 1
  }
  return "a prime"
}
```


Euler's Phi-Function

- Euler's Phi-function, $\phi(n)$, (called the Euler's totient) plays a very important role in cryptography
- The function finds the number of integers that are both smaller than n and relatively prime to n.
- To find the value of $\phi(n)$
  1. $\phi(1) = 0.$
  2. $\phi(p) = p - 1$ if p is a prime.
  3. $\phi(m \times n) = \phi(m) \times \phi(n)$ if m and n are relatively prime.
  4. $\phi(p^e) = p^e - p^{e-1}$ if p is a prime.
- Ex: What is the value of $\phi(13)$, $\phi(10)$?
  - Because 13 is a prime, $\phi(13) = (13 - 1) = 12$.
  - We can use the third rule: $\phi(10) = \phi(2) \times \phi(5) = 1 \times 4 = 4$, because 2 and 5 are primes.

Euler's Phi-Function

- We can combine the above four rules to find the value of $\phi(n)$. For example, if n can be factored as

$$n = p_1^{e_1} \times p_2^{e_2} \times \ldots \times p_k^{e_k}$$

then we combine the third and the fourth rule to find

$$\boxed{\phi(n) = (p_1^{e_1} - p_1^{e_1-1}) \times (p_2^{e_2} - p_2^{e_2-1}) \times \cdots \times (p_k^{ek} - p_k^{ek-1})}$$

- Note:

The difficulty of finding $\phi(n)$ depends on the difficulty of finding the factorization of n.

- Ex,
  - What is the value of $\phi(240)$,
  - $\phi(49)=36$?

Euclidean Algorithm in GCD

- Greatest Common Divisor GCD (a,b) of a and b is the largest number that divides evenly into both a and b
- GCD(a,b): GCD(a,b) = GCD(b, a mod b)
- Euclidean Algorithm to compute GCD(a,b) is:

```
EUCLID(a,b)
1. A = a; B = b
2. if B = 0 return A=gcd(a,b)
3. R = A mod B
4. A = B
5. B = R
6. goto 2
```

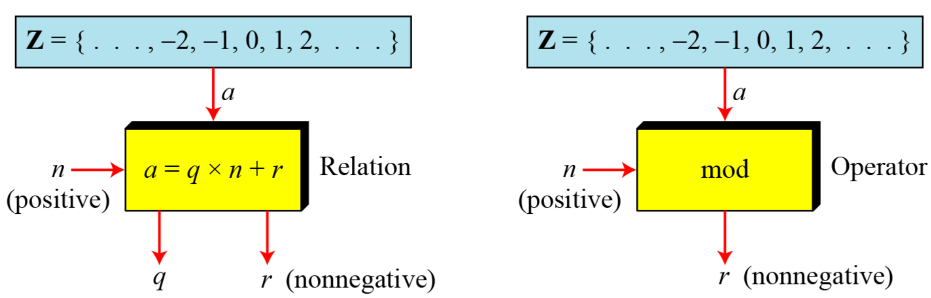
- Example GCD(24,63)

a	b	r
24	63	24
63	24	15
24	15	9
15	9	6
9	6	3
6	**3**	0

Modulo

- Modulo,
  - Modulo properties
  - Congruence modulo
  - Fermat's Little Theorem
  - Euler's Theorem
  - Modular inverse (additive, multiplicative)

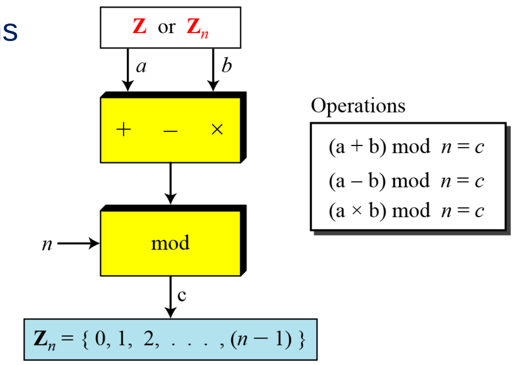
Division algorithm for integers



- Ex: $a = -7$ and $n = 3$: $q = -3$ and $r = 2$, coz: $-7 = (-3)(3) + 2$.
  - Note: n x q < a then n x q + r = a, (r>=0, r<n)
- Find the result of the following operations:
  a. 27 mod 5
  b. 36 mod 12
  c. −18 mod 14
  d. −7 mod 10

Modulo Properties
(addition, subtraction, multiplication, exponentiation)

- Operations



- Ex:

$$(14 + 7) \bmod 15 \rightarrow (21) \bmod 15 = 6$$
$$(7 - 11) \bmod 13 \rightarrow (-4) \bmod 13 = 9$$
$$(7 \times 11) \bmod 20 \rightarrow (77) \bmod 20 = 17$$

Modular, congruence

- In mathematics, **modular arithmetic** is a system of arithmetic for integers, where numbers "wrap around" upon reaching a certain value—the **modulus** (plural **moduli**).
- The modern approach to modular arithmetic was developed by Carl Friedrich Gauss (Germany)
- Ex: Instead of 13 = 1. (in clock ring)
  - write 13 ≡ 1 (mod 12) and read it "13 is congruent to 1 modulo 12" or, to abbreviate, "13 is 1 modulo 12".
- Two numbers are congruent modulo a given number if they give the same remainder when divided by that number.
- In general, a ≡ b (mod n) if a-b is a multiple of n. Equivalently, a ≡ b (mod n) if a and b have the same remainder when divided by n (remainder modulo n)
- Examples:
  - 12 ≡ 0 (mod 12); 17 ≡ 5 (mod 12); 37 ≡ 1 (mod 12); -1 ≡ 11 (mod 12)

Associated with addition, multiplicative

- Additive /Sub

$$\begin{aligned} \textit{If} \qquad a &\equiv b \pmod{m} \qquad \textit{and} \\ c &\equiv d \pmod{m}, \qquad \textit{then} \\ a + c &\equiv b + d \pmod{m}. \end{aligned}$$

- Ex: 9+18 (mod 4) ≡ ? (mod 4).
  - 1 ≡ 9 (mod 4), 2 ≡ 18 (mod 4) ➔ ≡ 9+18 (mod 4) ≡ 1+2 (mod 4).

- Multiplicative

$$\begin{aligned} \textit{If} \qquad a &\equiv b \pmod{m} \qquad \textit{and} \\ c &\equiv d \pmod{m}, \qquad \textit{then} \\ a \times c &\equiv b \times d \pmod{m}. \end{aligned}$$

Ex,

- Examples of Additive:
  - 7 + 8 ≡ 3 (mod 12); 25 + 14 ≡ 1 + 2 ≡ 3 (mod 12);
  - 7 + 2 ≡ ? (mod 12); -1 + 14 ≡ ? (mod 12)
  - 39 + 13 ≡ ? (mod 12)
- Ex, 19 + 23 + 15 ≡ ? (mod 12)
  - First replace each number by its remainder mod 12: 7 + 11 + 3. then do the sum: 21
  - and replace the sum by its remainder modulo 12: 9
    => 19 + 23 + 15 ≡ 9 (mod 12)
- If today is Sunday, what day will it be in 1000 days?
  We need to find the remainder of 1000 when divided by 7
  - 1000: = 700+280+20
- Examples of multiplicative:
  - 7 x 4 ≡ 8 (mod 10); 19 x 28 ≡ 2 (mod 10): 9x8 = 72 ≡ 2 (mod 10)

Modular Exponentiation

- Exponentiation: $y = a^x$ $\rightarrow$ ***Logarithm:*** $x = \log_a y$
- In cryptography, a common modular operation is exponentiation. uses exponentiation. RSA for both encryption and decryption with very large exponents $\boxed{y \equiv a^x \bmod n}$
- Fast exponentiation is possible using the square-and-multiply method

$$y = a^{\boxed{x_{n_b-1} \times 2^{n_b-1} + x_{n_b-2} \times 2^{n_b-2} + \cdots + x_1 \times 2^1 + x_0 \times 2^0}}$$ in which x_i is 0 or 1

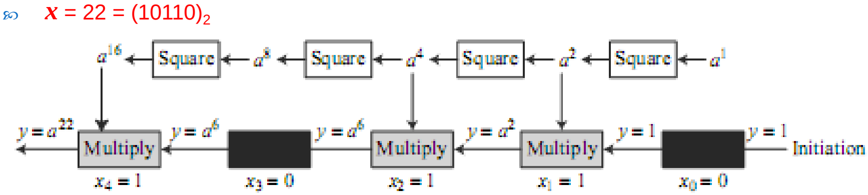
$$\downarrow$$

$$y = \boxed{\boxed{a^{2^{n_b-1}}} \text{ or } \boxed{1}} \times \boxed{\boxed{a^{2^{n_b-2}}} \text{ or } \boxed{1}} \times \cdots \times \boxed{\boxed{a^2} \text{ or } \boxed{1}} \times \boxed{\boxed{a} \text{ or } \boxed{1}}$$

- Ex: $\boxed{y = a^9 = a^{1001_2} = a^8 \times 1 \times 1 \times a}$

Ex: Exponentiation

- Ex, Calculation of 17^{22} mod 21. result is y = 4. $\boxed{y \equiv a^x \bmod n}$
- x = 22 = $(10110)_2$



- => $y=a^{16}.a^4.a^2$

i	x_i	*Multiplication (Initialization: y = 1)*		*Squaring (Initialization: a = 17)*	
0	0		→	$a = 17^2 \bmod 21 = 16$	a^2
1	1	$y = 1 \times 16 \bmod 21 = 16$	→	$a = 16^2 \bmod 21 = 4$	a^4
2	1	$y = 16 \times 4 \bmod 21 = 1$	→	$a = 4^2 \bmod 21 = 16$	a^8
3	0		→	$a = 16^2 \bmod 21 = 4$	a^{16}
4	1	$y = 1 \times 4 \bmod 21 = 4$	→		

Ex

- Compute without a calculator.

15x29 mod 13

2x29 mod 13

2x3 mod 13

−11x3 mod 13

What conclusion?

- Compute without using a calculator:

$x=3^{10}$ mod 13

$x=7^{100}$ mod 13

7^x = 11 mod 13

Ex: x= $18^{489391312}$ (mod 19)

Fermat's Little Theorem

- Fermat's little theorem plays a very important role in number theory and cryptography. 2 versions:

If p is a prime and a is an integer such that $gcd(a,p)=1$	If p is a prime and a is an integer
$a^{p-1} \equiv 1 \bmod p$	$a^p \equiv a \bmod p$

- Ex: Find the result of 6^{10} mod 11.
  - 6^{10} mod 11 = 1. This is the first version of Fermat's little theorem where p = 11.
- Ex: Find the result of 3^{12} mod 11.

$$3^{12} \bmod 11 = (3^{11} \times 3) \bmod 11 = (3^{11} \bmod 11)\,(3 \bmod 11) = (3 \times 3) \bmod 11 = 9$$

Fermat's Little Theorem

- Multiplicative Inverses: If $\boldsymbol{p}$ is a prime and a is an integer such that p does not divide a $(gcd(a,p)=1)$

$$a \times a^{-1} \bmod p = a \times a^{p-2} \bmod p = a^{p-1} \bmod p = 1 \bmod p$$

$$\boxed{a^{-1} \bmod p = a^{p-2} \bmod p}$$

- The answers to multiplicative inverses modulo a prime can be found without using the extended Euclidean algorithm:

a. $8^{-1} \bmod 17 = 8^{17-2} \bmod 17 = 8^{15} \bmod 17 = 15 \bmod 17$

b. $5^{-1} \bmod 23 = 5^{23-2} \bmod 23 = 5^{21} \bmod 23 = 14 \bmod 23$

c. $60^{-1} \bmod 101 = 60^{101-2} \bmod 101 = 60^{99} \bmod 101 = 32 \bmod 101$

d. $22^{-1} \bmod 211 = 22^{211-2} \bmod 211 = 22^{209} \bmod 211 = 48 \bmod 211$

Euler's Theorem

- Euler's theorem can be thought of as a generalization of Fermat's little theorem.
  - The modulus in the Fermat theorem is a prime,
  - The modulus in Euler's theorem is an integer
- Two versions of this theorem.

If a and n are co-prime. $gcd(a,n)=1$

$$\boxed{a^{\phi(n)} \equiv 1 \pmod{n}}$$

If $n = p \times q$, $a < n$, and k an integer

$$\boxed{a^{k \times \phi(n) + 1} \equiv a \pmod{n}}$$

- The second version is used in the RSA cryptosystem

Euler's Theorem

- Ex1: Find the result of 6^{24} mod 35.
  - We have 6^{24} mod 35 = $6^{\varphi(35)}$ mod 35 = 1.
- Ex2: Find the result of 2062 mod 77.
  - If we let k = 1 on the second version:

20^{62} mod 77 = (20 mod 77) ($20^{\varphi(77)+1}$ mod 77) mod 77 = (20)(20) mod 77 = 15.

- Multiplicative Inverses: If n and a are coprime:

$$\boxed{a^{-1} \bmod n = a^{\phi(n)-1} \bmod n}$$

$$a \times a^{-1} \bmod n = a \times a^{\phi(n)-1} \bmod n = a^{\phi(n)} \bmod n = 1 \bmod n$$

a. $8^{-1} \bmod 77 = 8^{\phi(77)-1} \bmod 77 = 8^{59} \bmod 77 = 29 \bmod 77$

b. $7^{-1} \bmod 15 = 7^{\phi(15)-1} \bmod 15 = 7^{7} \bmod 15 = 13 \bmod 15$

c. $60^{-1} \bmod 187 = 60^{\phi(187)-1} \bmod 187 = 60^{159} \bmod 187 = 53 \bmod 187$

d. $71^{-1} \bmod 100 = 71^{\phi(100)-1} \bmod 100 = 71^{39} \bmod 100 = 31 \bmod 100$

Inverses modulo

- Additive Inverse: a +b ≡ 0 (mod n); (a +b ≡ c (mod n))
- Multiplicative Inverse: a. b ≡ 1 (mod n); (a. b ≡ c (mod n))

=> *b* is the multiplicative/ Additive inverse of *a* modulo *n*

- Ex:
  - Add: (mod 10): (0, 0), (1, 9), (2, 8), (3, 7), (4, 6), and (5, 5).
  - Mul: (mod 10): (1, 1), (3, 7) and (9, 9).
- Find additive inverse (modulo=0)
  - 7 + ? ≡ 0 (mod 16) => the additive inverse of 7 mod 16 is 9 (16-7)
- Find multiplicative inverse (modulo=1)
  - 2 x ? ≡ 1 (mod 7) => the multiplicative inverse of 2 is: ?

Find multiplicative inverse: (modulo<>1)

- 2 x ? ≡ 3 (mod 7)

How to find Multiplicative Inverses

- If $ax \equiv 1 \pmod{b}$ then x multiplicative inverse of a modulo b
- Means: $a^{-1} \equiv x \pmod{b}$
- We have: $ax - 1 \vdots b$

$\Leftrightarrow ax - 1 = by$ *by* = 0 (mod *b*) for all integers *y*

$\Leftrightarrow ax - by = 1$

- In fact, the only time a has a multiplicative inverse mod b is when a and x are relatively prime
- Need solve: $ax + by = 1$

Extended-Euclid

- Extended-Euclid dùng để giải phương trình diophantine: ax+by=c
- Theo định lí Bézout (Bézout's indentify): Cho hai số nguyên a, b khi đó luôn tồn tại hai số x, y sao cho:

  ax + by = gcd(a, b)
- Người ta cũng chứng minh được phương trình trên có nghiệm khi và chỉ khi gcd(a, b) = c.
  - => phương trình diophante có thể có vô số nghiệm, và từ mỗi một nghiệm ta có thể sinh ra những nghiệm khác.
- Một trong những ứng dụng quan trọng nhất của thuật toán Extended-Euclid đó chính là dùng để tìm nghịch đảo modulo:

$$a^{-1} \equiv x \pmod{b}$$

- Nhận xét: nếu gcd(a, b) = 1, giải phương trình:

$$ax + by = 1$$

Extended Euclidean Algorithm

- Extended-Euclid(a,b) does it all for us:
  - Returns integers x, y, d such that $d = \gcd(a, b)$ and $ax + by = d$
  - First, return d=1 as the gcd to confirm: a and b are relatively prime
  - If so, it finds the multiplicative inverse x of a mod b
- See

$b = 0$ thì $d = a$, $x = 1$ và $y = 0$.

$b \neq 0$, ta có: $ƯCLN(a, b) = ƯCLN(b, a\%b) \Leftrightarrow ax + by = bx_1 + (a\%b)y_1$

$$\Leftrightarrow ax + by = bx_1 + \left(a - \left[\frac{a}{b}\right].b\right)y_1 \Leftrightarrow ax + by = ay_1 + \left(x_1 - \left[\frac{a}{b}\right].y_1\right)b$$

Suy ra: $x = y_1, y = x_1 - \left[\frac{a}{b}\right].y_1$.

```
function extended-Euclid (a, b)
    if b = 0: return (1, 0, a)
    (x', y', d) = extended-Euclid(b, a mod b)
    return (y', x' - floor(a/b)y', d)
```

35

The calculations

Calculate	Which satisfies	Calculate	Which satisfies
$r_{-1} = a$		$x_{-1} = 1; y_{-1} = 0$	$a = ax_{-1} + by_{-1}$
$r_0 = b$		$x_0 = 0; y_0 = 1$	$b = ax_0 + by_0$
$r_1 = a \bmod b$ $q_1 = \lfloor a/b \rfloor$	$a = q_1 b + r_1$	$x_1 = x_{-1} - q_1 x_0 = 1$ $y_1 = y_{-1} - q_1 y_0 = -q_1$	$r_1 = ax_1 + by_1$
$r_2 = b \bmod r_1$ $q_2 = \lfloor b/r_1 \rfloor$	$b = q_2 r_1 + r_2$	$x_2 = x_0 - q_2 x_1$ $y_2 = y_0 - q_2 y_1$	$r_2 = ax_2 + by_2$
$r_3 = r_1 \bmod r_2$ $q_3 = \lfloor r_1/r_2 \rfloor$	$r_1 = q_3 r_2 + r_3$	$x_3 = x_1 - q_3 x_2$ $y_3 = y_1 - q_3 y_2$	$r_3 = ax_3 + by_3$
• • •	• • •	• • •	• • •
$r_n = r_{n-2} \bmod r_{n-1}$ $q_n = \lfloor r_{n-2}/r_{n-3} \rfloor$	$r_{n-2} = q_n r_{n-1} + r_n$	$x_n = x_{n-2} - q_n x_{n-1}$ $y_n = y_{n-2} - q_n y_{n-1}$	$r_n = ax_n + by_n$
$r_{n+1} = r_{n-1} \bmod r_n = 0$ $q_{n+1} = \lfloor r_{n-1}/r_{n-2} \rfloor$	$r_{n-1} = q_{n+1} r_n + 0$		$d = \gcd(a, b) = r_n$ $x = x_n; y = y_n$

Ex

- $a=1759, b=550, gcd(1759, 550)= 1795.x + 550.y$
- $x, y, d =?$

$a = q_1 b + r_1, r_1 = ax_1 + by_1$
$b = q_2 r_1 + r_2 , r_2 = ax_2 + by_2$
$r_1 = q_3 r_2 + r_3, r_3 = ax_3 + by_3$
.....
$r_{n-2} = q_n r_{n-1} + r_n , r_n = ax_n + by_n$
$r_{n-1} = q_n+1 r_n + 0$

$$x_i = x_{i-2} - q_i x_{i-1}$$

$$y_i = y_{i-2} - q_i y_{i-1}$$

$\underline{d=1, x=-111, y=355}$

i	*r*	*q*	x_i	y_i
	1759		1	0
	550		0	1
1	109	3	1	-3
2	5	5	-5	16
3	4	21	106	-339
4	1	1	-111	355
5	0	4		

Find Multiplicative Inverses, Ex

- Ex: What is multiplicative inverse of 20 Mod 79
  - Are they relatively prime?
  - Euclid or extended-Euclid are the algorithms we use to find out (with the extension not needed). The extension only kicks in after the gcd has been found anyway.
  - returns integers *x*, *y*, *d* such that *d* = gcd(*a*, *b*) and *ax* + *by* = *d*

    20.x + 79.y = gcd(20,79) = 1
  - Remember to put the largest number first and if you have to switch at the beginning, then remember to switch *x* and *y* at the end

Multiplicative Inverse of 20 Mod 79

```
function extended-Euclid (a, b)
if b = 0: return (1, 0, a)
(x', y', d) = extended-Euclid(b, a mod b)
return (y', x' − floor(a/b)y', d)
```

(returns integers x, y, d such that d = gcd(a, b) and $ax + by = d$)

$$x_i = x_{i-2} - q_i x_{i-1}$$
$$y_i = y_{i-2} - q_i y_{i-1}$$

i	r	q	x_i	y_i

39

Multiplicative Inverse of 20 Mod 79

i	r	q	x_i	y_i
-1	20		1	0
0	79		0	1
1	20	0	1	0
2	19	3	-3	0
3	1	1	4	-1
	0			

$ax + by$ = 1 = 79.x + 20.y
B1: 79mod20=19
B2: 20mod19=1
B3: 19mod1=0 =>stop
From B2: 1=20-19
From B1: 19=79-3*20, have:
1=20-79+3*20
1=4*20-79 => x=4

$ax + by$ = 1 = 79(-1) + 20(4)
$ax + by$ = 1 = 20(4) + 79(-1)
Since we initially switched 20 and 79
Thus $x = a^{-1} = 4$

Complexity?

40

Multiplicative Inverse of 12 mod 15?

function extended-Euclid (a, b)
if b = 0: return (1, 0, a)
(x', y', d) = extended-Euclid(b, a mod b)
return (y', x' – floor(a/b)y', d)

i	r	q	x_i	y_i

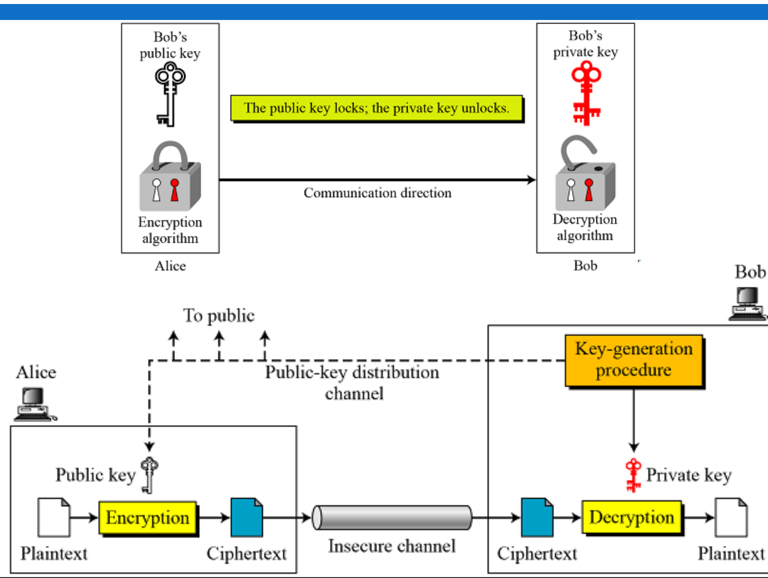
Asymmetric Cryptography

Lecturer: Nguyễn Thị Thanh Vân – FIT - HCMUTE

Asymmetric Cryptography

- Asymmetric-key cryptosystem
- Introduction to RSA
- Procedure of RSA
- Attacks on RSA
- RSA Security

Asymmetric-key cryptosystem



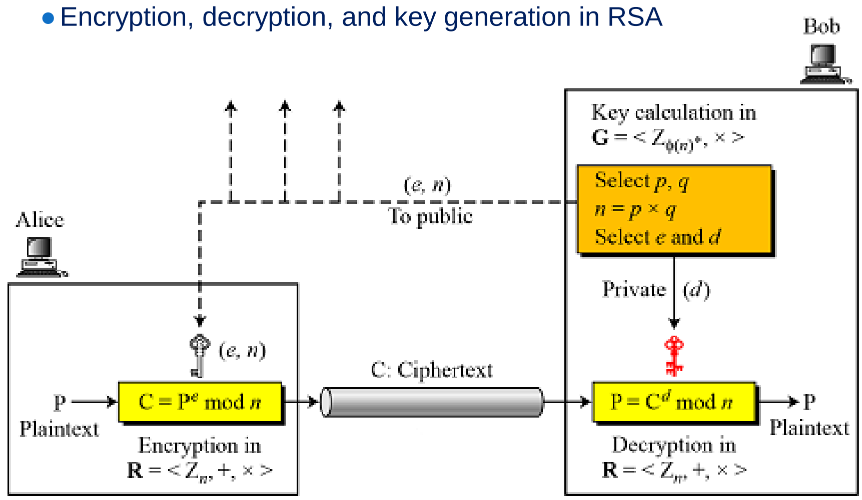
RSA (Rivest, Shamir, Adleman)

- RSA is a well – known and widely popular public key cryptography.
- Firstly published by the authors in 1977 (MIT)
- **Its based on exponentiation on Galos' Field of the integers of modulo prime number**
  - Exponentiation has complexity $O((\log n)^3)$ (**easy**)
- RSA security is based on hardness of the factor analysis and the discrete logarithm problem:
  - Analysis problem has complexity $O(e^{\log n \log \log n})$ (**difficult**)
  - Similarly, discrete logarithm is very hard
- RSA has been copyrighted in North America and in some other countries.

46

Procedure of RSA

- Encryption, decryption, and key generation in RSA



RSA Algorithm

RSA_Key_Generation

{

Select two large primes p and q such that $p \neq q$.

$n \leftarrow p \times q$

$\phi(n) \leftarrow (p-1) \times (q-1)$

Select e such that $1 < e < \phi(n)$ and e is coprime to $\phi(n)$

$d \leftarrow e^{-1} \bmod \phi(n)$ // d is inverse of e modulo $\phi(n)$

Public_key $\leftarrow$ (e, n) // To be announced publicly

Private_key $\leftarrow d$ // To be kept secret

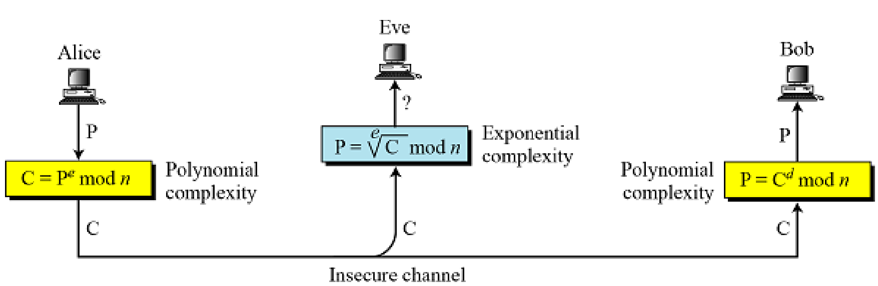
return Public_key and Private_key

}

- **Encryption**: $C = M^e$ **mod** n, M < n using public key
- **Decryption**: $M = C^d$ mod n using private key
- **Signature**: $S = M^d$ mod n, M < n using private key
- **Verification:** $M = S^e$ mod n using public key

48

Complexity of operations in RSA



RSA uses modular exponentiation for encryption/decryption; To attack it, Eve needs to calculate $\sqrt[e]{C} \bmod n$.

24/09/2021 49

RSA Example

1. Select primes: $p = 17$ & $q = 11$
2. Calculate $n = pq = 17 \times 11 = 187$
3. Calculate $ø(n)=(p–1)(q-1)=16x10=160$
4. Select e: $\gcd(e,160) = 1$; choose $e = 7$
5. Determine d: $de = 1 \mod 160$ and $d < 160$

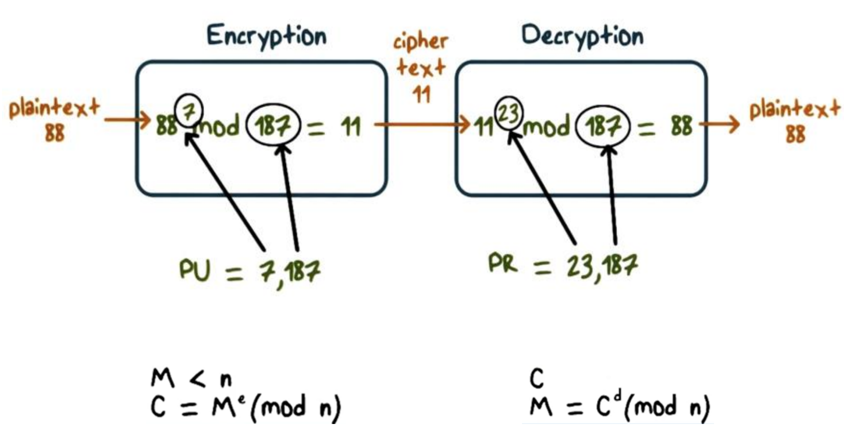
Value is d = 23 since 23x7 = 161 = 1x160 + 1

1. Publish public key PU = {7,187}
2. Keep secret private key PR = {23,187}

- If he chooses e to be 13, then d is 37.

24/09/2021 50

RSA Example



Exercise

1. Let the two primes $p = 41$ and $q = 17$ be given as set-up parameters for RSA.
   - Which of the parameters $e1 = 32$, $e2 = 49$ is a valid RSA exponent? Justify your choice.
   - Compute the corresponding private key $Kpr = (p,q,d)$. Use the extended Euclidean algorithm for the inversion and explain every calculation step.
2. Encrypt and decrypt by means of the RSA algorithm with the following system parameters:
   - $p = 3$, $q = 11$, $d = 7$, $x = 5$
   - $p=5$, $q=11$, $e=3$, $x=9$

expl

- $p = 41$ and $q = 17$
- N = 41.17
- Phi(N) = 40.16 = 640
- Gcd(phi(N),e)=1 => e=49
- d = 49^{-1} (mod 640) => d.49=1 (mod 640)
- => Extend-Euclid: d.49 + y.640 =1

640 = 49.13 + 3

49 = 3.16 + 1

=> 1 = 49 – 3.16

1 = 49 – (640 – 49.13).16 = 49 – 640.16 + 49.13.16

1 = 49.(1 + 13.16) – 16.640

= 49.209

Pr(209,41.17); Pu(49,41.17)

RSA Quiz

Fill in the text boxes:

Given p = 3 and q = 11

1. Compute n: n= ☐

2. Compute φ(n):
φ(n) = ☐

3. Assume e = 7
Compute the value of d:
d = ☐

4. What is the public key

(e,n) = (☐ , ☐)

5. What is the private key

(d,n) = (☐ , ☐)

RSA Encryption Quiz

Given:

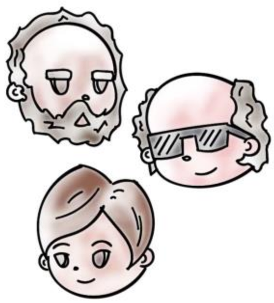
- Public key is (e, n) => (7, 33)
- Private key is (d, n) => (3, 33)
- Message m = 2

What is the encryption of m: ☐

What formula is used to decrypt m? ☐

(Use ** for denoting an exponent)

RSA Characteristics



- Variable key length
- Variable plaintext block size
  - Plaintext treated as an integer, and must be “smaller” than the key
  - Ciphertext block size is the same as the key length

RSA Security

- Four possible approaches to attacking the RSA algorithm are:

1. Brute force: This involves trying all possible private keys.

2. Mathematical attacks: There are several approaches, all equivalent in effort to factoring the product of two primes.

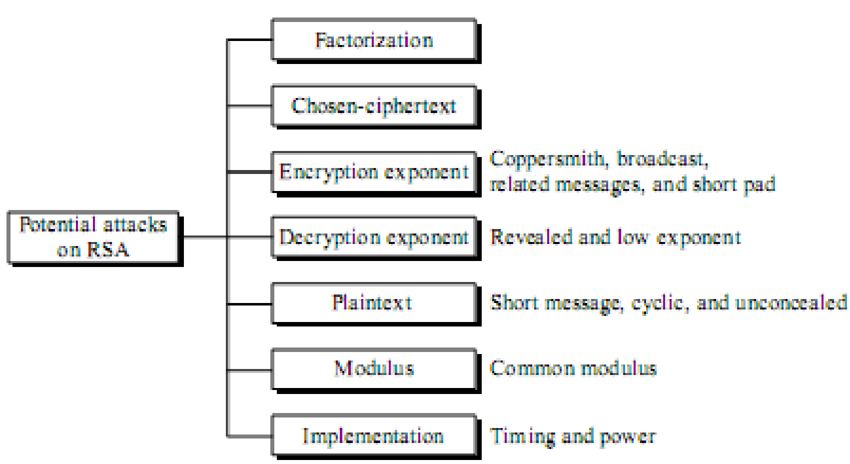
3. Timing attacks: These depend on the running time of the decryption algorithm.

4. Chosen ciphertext attacks: This type of attack exploits properties of the RSA algorithm.

Why RSA is Secure?

- Factoring an integer with at least 512-bit is very hard!
- **But if you can factor big number n then given public key <e,n>, you can find d, and hence the private key by:**
  - Knowing factors p, q, such that, $n = p \times q$
  - Then compute $\phi(n) = (p-1)(q-1)$
  - Then find d such that $e \times d = 1 \bmod \phi(n)$

Attacks on RSA



Key Management

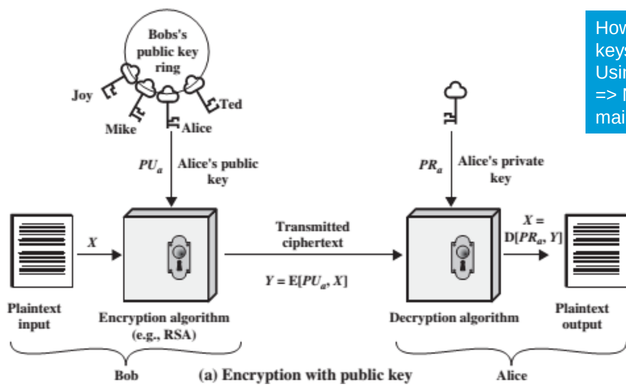
Lecturer: Nguyễn Thị Thanh Vân – FIT - HCMUTE

Key Management

- Symmetric-key distribution
  - Key-distribution center: KDC
  - Session key
- KERBEROS
  - Servers
  - Operation
- Symmetric-key agreement
  - Diffie-Hellman key agreement
- Public-key distribution
  - Public announcement
  - CA
  - X.509

Symmetric-key distribution

- Symmetric-key cryptography
  - is more efficient than asymmetric-key cryptography for enciphering large messages.
  - needs a shared secret key between two parties.

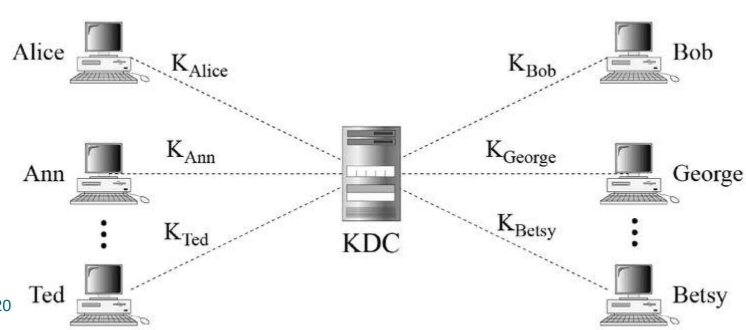


How can she exchange one million keys with one million people? Using the Internet => not secure => Need an efficient way to maintain and distribute secret keys

Key-Distribution Center: KDC

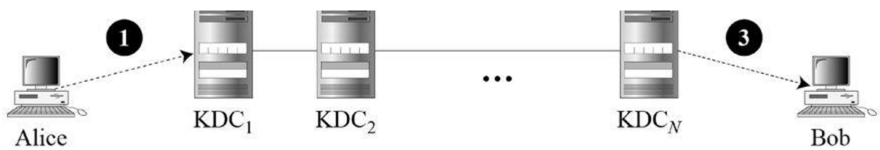
Key-Distribution Center: KDC

- KDC: using a trusted third party,
  - reduce the number of keys
  - prevent MITM from impersonating either of both
- Each person establishes shared secret key with the KDC
  - A secret key is established between the KDC and each member.



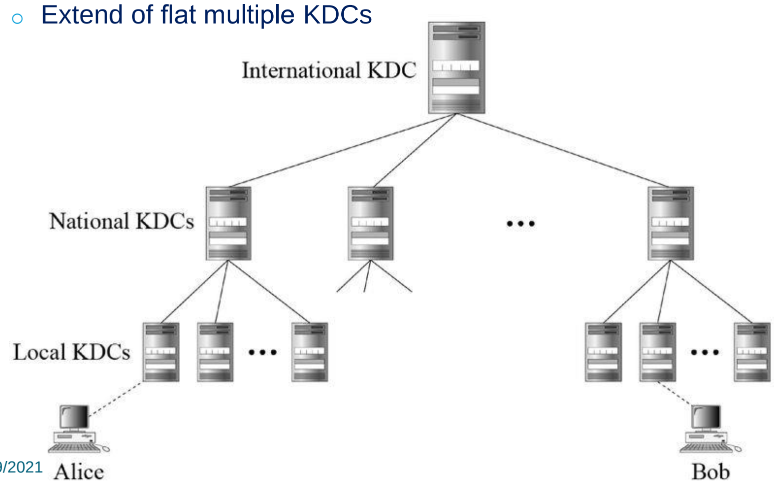
Key-Distribution Center: KDC

- Flat multiple KDCs:
  - Avoid the system becomes unmanageable and a bottleneck when the number of people using a KDC increases
  - divide the world into domains.
  - Each domain can have one or more KDCs (for redundancy in case of failure).



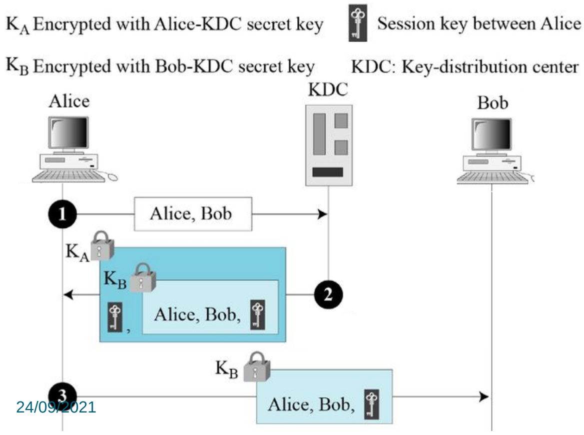
Key-Distribution Center: KDC

- Hierarchical Multiple KDCs:
  - Extend of flat multiple KDCs



Session Keys

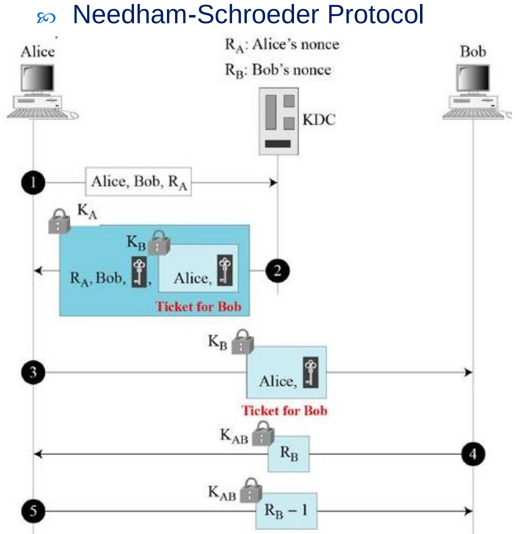
- A KDC creates a secret key for each member.
  - can be used only between the member and the KDC, not between 2 members.
- A Simple Protocol Using a KDC



1. A sends a message to KDC to obtain a symmetric session key K_{AB}
2. KDC creates a ticket, enc using K_B , then sends to A.
A decrypts it and extracts K_{AB}
A is authenticated to KDC
3. A sends ticket to B.
B opens ticket: is authenticated to KDC & A.
A is also authenticated to B

Session Keys

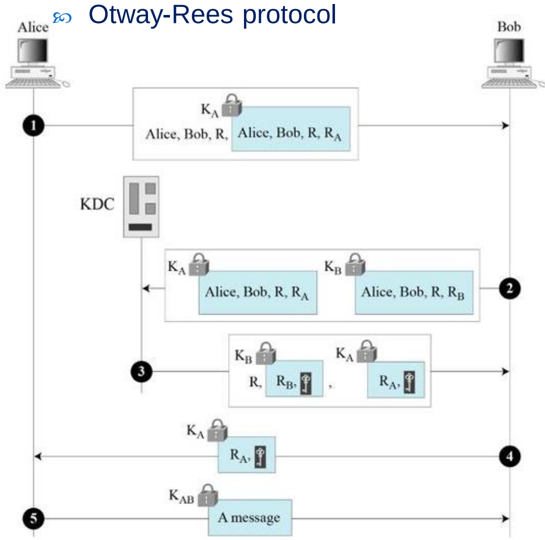
- Needham-Schroeder Protocol



1. Alice sends a message to the KDC that includes R_A, her identity, and Bob's identity.
2. The KDC sends an encrypted message to Alice that includes R_A, Bob's identity, the session key, and an encrypted ticket for Bob. The whole message is encrypted with Alice's key.
3. Alice sends Bob's ticket to him.
4. Bob sends R_B to Alice, encrypted with the session key.
5. Alice responds R_B to Bob.
Note that the response carries R_B – 1 instead of R_B.

Session Keys

- Otway-Rees protocol



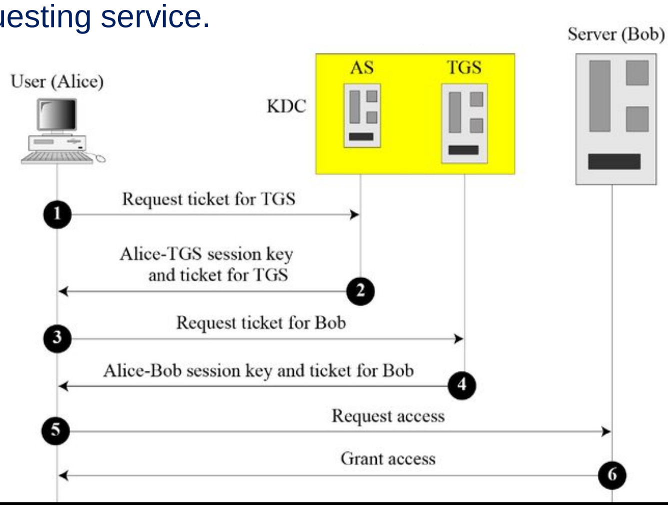
1. Alice sends a message to Bob (a common nonce, R, the identities of Alice & Bob), and a ticket for KDC (R_A, a copy of the R, and the identities of Alice & Bob).
2. Bob creates the same type of ticket, but with R_B. Both tickets are sent to the KDC.
3. The KDC creates a message that contains R, a ticket for Alice and a ticket for Bob; the message is sent to Bob. The tickets contain the corre sponding nonce, R_A or R_B, and the session key, K_{AB}.
4. Bob sends Alice her ticket.
5. Alice sends a short message encrypted with her session key K_{AB} to show that she has the session key.

Kerberos

- Kerberos is an authentication protocol, and at the same time a KDC, that has become very popular.
  - EX Windows 2000, use Kerberos
- Three servers are involved in the Kerberos protocol:
  - an authentication server (AS),
  - A ticket-granting server (TGS),
  - a real (data) server that provides services to others.

Kerberos servers

- Ex: Bob is the real server and Alice is the user requesting service.



Operation

- 1. Alice sends her request to the AS in plain text using her registered identity.
- 2. The AS sends a message encrypted with Alice's $K_{A\text{-}AS}$.
  - contains two items: a session key, $K_{A\text{-}TGS}$, and a ticket for the TGS that is encrypted with the TGS symmetric key, $K_{AS\text{-}TGS}$.
  - when the message arrives, she types her symmetric password.
  - The process now uses $K_{A\text{-}AS}$ to decrypt the message sent. $K_{A\text{-}TGS}$ and the ticket are extracted.
- 3. Alice now sends three items to the TGS.
  - the ticket received from the AS.
  - the name of the real server (Bob),
  - a timestamp that is encrypted by $K_{A\text{-}TGS}$ (prevents a replay by Eve.)
- 4. Now, the TGS sends two tickets, each containing the session key between Alice and Bob, $K_{A\text{-}B}$.
  - The ticket for Alice is encrypted with $K_{A\text{-}TGS}$;
  - the ticket for Bob is encrypted with Bob's key, $K_{TGS\text{-}B}$.

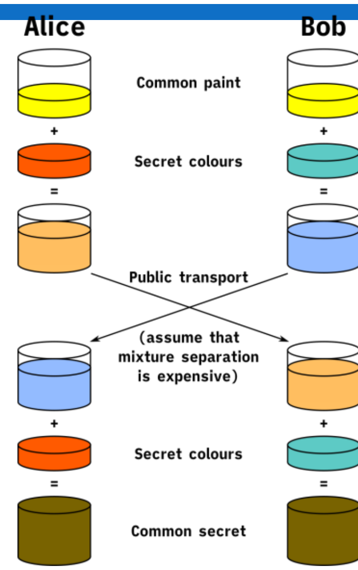
Symmetric-key agreement

- The session-key creation is referred to as the symmetric-key agreement.
- Two common methods,
  - Diffie Hellman and
  - Station-to-station,

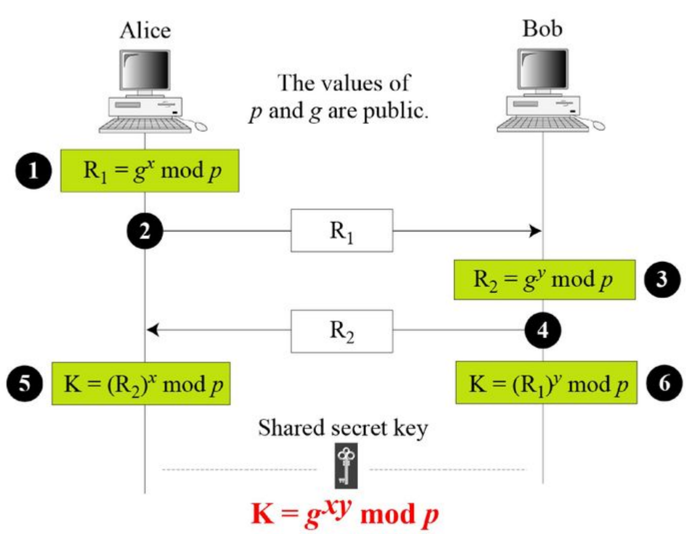
Diffie and Hellman Key Exchange

- **First published** public-key algorithm
- By Diffie and Hellman in 1976 along with the **exposition of public key concepts**
- Used in a number of commercial products
- **Practical method to exchange a secret key** securely that can then be used for subsequent encryption of messages
- Security **relies on difficulty of computing discrete logarithms**

Basic idea



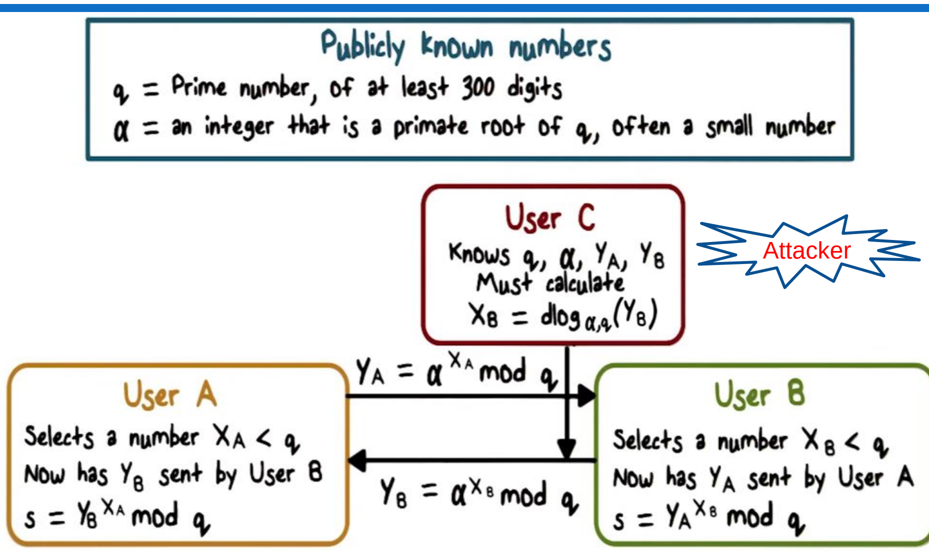
Diffie-Hellman method



Expl

Alice				Bob		
Bí mật	Công khai	Tính	Gửi	Tính	Công khai	Bí mật
a	p, g		p,g →			b
a	p, g, A	g^a mod p = A	A →		p, g	b
a	p, g, A		← B	g^b mod p = B	p, g, A, B	b
a, **s**	p, g, A, B	B^a mod p = s		A^b mod p = s	p, g, A, B	b, **s**

Diffie and Hellman Key Exchange



Diffie-Hellman Example

Have
- Prime number $q = 353$
- Prime number $\alpha = 3$

A and B each computer their public keys
- A computes $Y_A = 3^{97} \bmod 353 = 40$
- B computes $Y_B = 3^{233} \bmod 353 = 248$

Then exchange and computer secret key:
- For A: $K = (Y_B)^{XA} \bmod 353 = 248^{97} \bmod 353 = 160$
- For B: $K = (Y_A)^{XB} \bmod 353 = 40^{233} \bmod 353 = 160$

Attacker must solve:
- $3^{\alpha} \bmod 353 = 40$ which is hard
- Desired answer is 97, then computer key as B does

Diffie-Hellman Quiz

Alice and Bob agree to use
prime q = 29 and primitive root α = 5

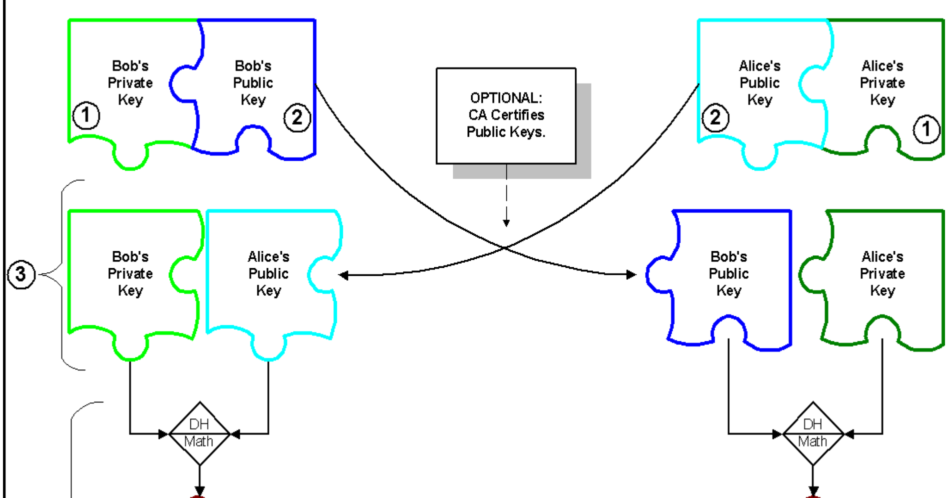
Alice chooses secret a = 6
Bob chooses secret b= 5

What number does Alice send Bob?

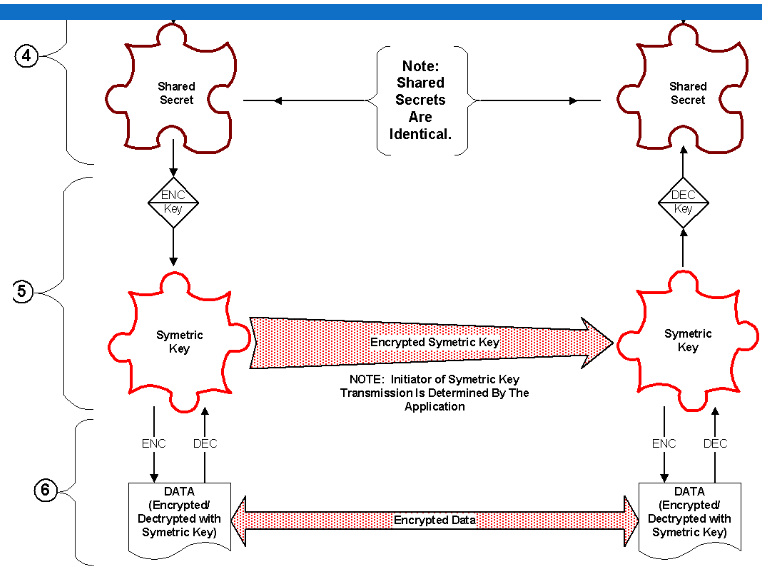
What number does Bob send Alice?

Implementation

Diffie-Helman Key Exchange



Implementation



Applications

- Diffie-Hellman is currently used in many protocols, namely:
  - Secure Sockets Layer (SSL)/Transport Layer Security (TLS)
  - Secure Shell (SSH)
  - Internet Protocol Security (IPSec)
  - Public Key Infrastructure (PKI)

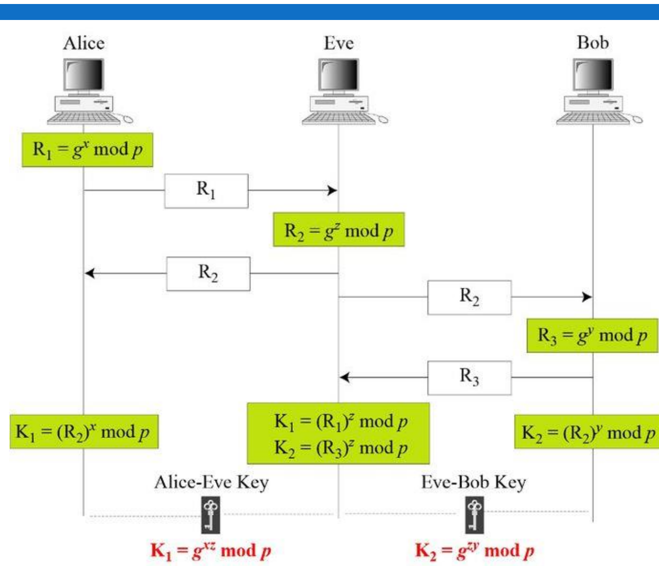
Diffie-Hellman Limitations

- Expensive exponential operation
  - DoS possible
- The scheme itself **cannot be used to encrypt anything –** it is for secret key establishment
- **No authentication**, so you cannot sign anything

Security of Diffie-Hellman

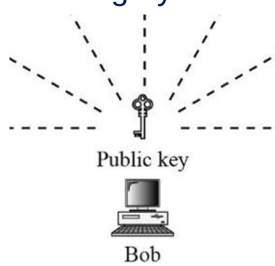
- The discrete logarithm attack
  - Attack can intercept R_1 and R_2.
  - If she can find x from $R_1 = g^x \mod p$ and y from $R2 = g^y \mod p$, then she can calculate the symmetric key $K = g^{xy} \mod p$.
  - => Need make Diffie-Hellman safe:
    - The prime p must be very large (more than 300 decimal digits).
    - The prime p must be chosen such that p − 1 has at least one large prime factor (more than 60 decimal digits).
    - The generator must be chosen from the group <Zp*, × >.
    - Bob and Alice must destroy x and y after they have calculated the symmetric key (use only once)
- the man-in-the-middle attack.
  - Also called Bucket Brigade Attack

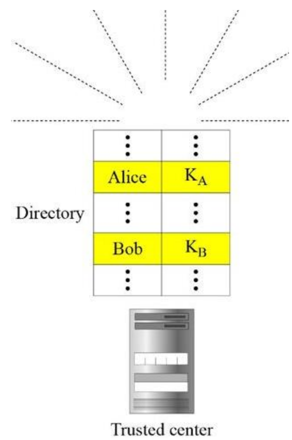
Man-in-the-middle attack



Public-key distribution

- In public-key cryptography, public keys are available to the public. => This approach, however, is not secure; it is subject to forgery.





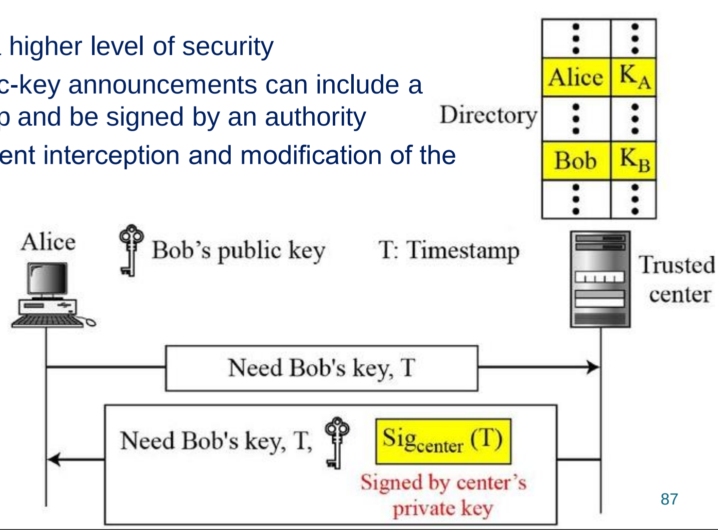
- Trusted Center:
  - A more secure approach is to have a trusted center retain a directory of public keys.

24/09/2021

Controlled trusted center

- Goal:
  - achieve a higher level of security
  - The public-key announcements can include a timestamp and be signed by an authority

  => to prevent interception and modification of the response.

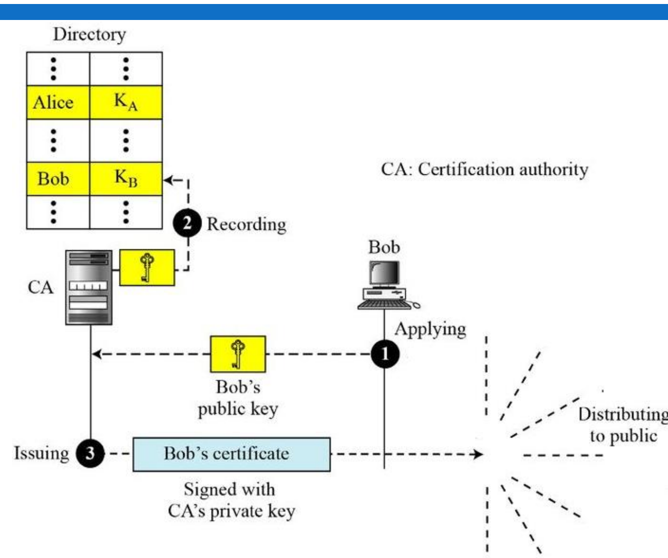


24/09/2021 87

Certification Authority

- The CA has a well-known public key itself that cannot be forged.
- Process of the CA:
  - checks Bob's ID (using a picture ID along with other proof).
  - asks for Bob's public key and writes it on the certificate.
  - signs the certificate with its private key => to prevent the certificate itself from being forged
- => Now Bob can upload the signed certificate. Anyone who wants Bob's public key downloads the signed certificate and uses the center's public key to extract Bob's public key.

Certification Authority

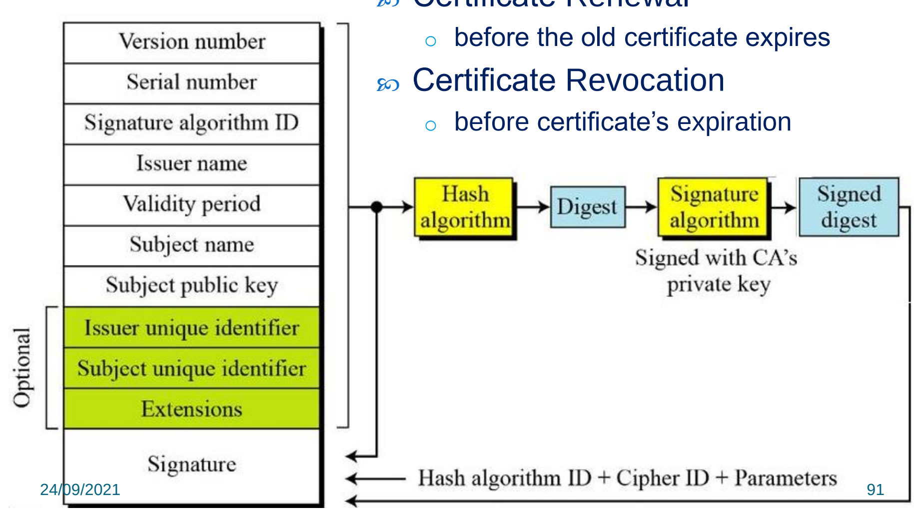


X.509

- Ihe ITU has designed X.509, a recommendation that has been accepted by the Internet with some changes.
- X.509 is a way to describe the certificate in a structured way. It uses a well-known protocol called ASN.1 (Abstract Syntax Notation 1) that defines fields familiar to C programmers.

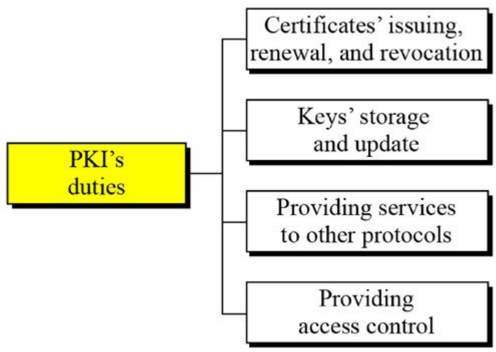
X.509 certificate format

- Certificate Renewal
  - before the old certificate expires
- Certificate Revocation
  - before certificate's expiration



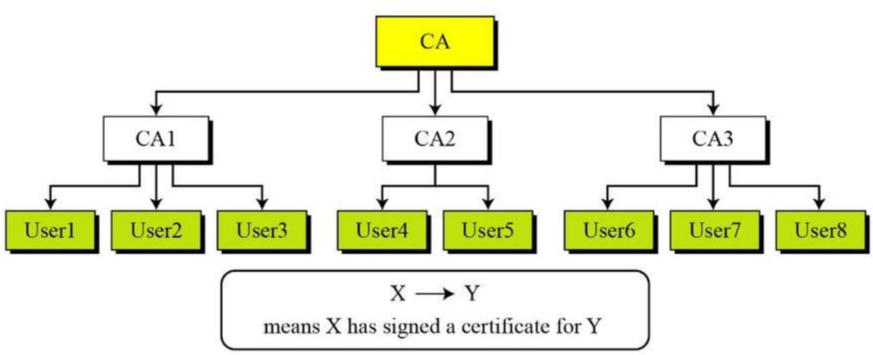
Public-Key Infrastructures (PKI)

- PKI is a model for creating, distributing, and revoking certificates based on the X.509.
  - The Internet Engineering Task Force has created the PKI X.509 (PKIX).
- Duties



PKI hierarchical model

- The trust model defines rules that specify how a user can verify a certificate received from a CA.
- Hierarchical Model: a tree-type structure with a root CA.
  - The root CA has a self-signed, self-issued certificate;
  - it needs to be trusted by other CAs and users for the system to work.



Summary

- Asymmetric encryption
- Modular arithmetic
- RSA
- Key Management
  - Symmetric-key distribution
  - KERBEROS
  - Symmetric-key agreement: Diffie-Hellman
  - Public-key distribution: CA, X.509