TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

Mật mã và độ phức tạp thuật toán

5. Mật mã khóa công khai
(Mật mã khóa bất đối xứng)

Nội dung

- Giới thiệu mật mã khóa công khai
- Hệ mật RSA
- So sánh mật mã khóa bí mật - mật mã khóa công khai

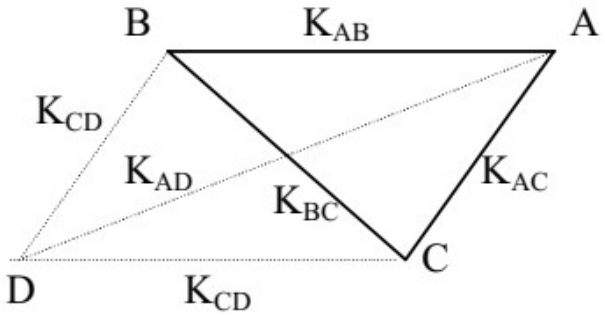
1/ Giới thiệu mật mã khóa công khai

Vấn đề của mật mã khóa bí mật

- Đặc điểm của hệ mã khóa bí mật
  - Bên gửi và bên nhận: cùng sử dụng một khóa
  - Trước khi truyền nhận dữ liệu - cần trao đổi khóa

Vấn đề của mật mã khóa bí mật

- Một số vấn đề của mật mã khóa bí mật
  - Quản lý khóa
    - Mỗi kết nối (2 người trong cộng đồng)- khóa riêng
    - Cộng đồng n bên: n(n-1)/2 khóa
    - Mỗi người: quản lý n-1 khóa bí mật



Vấn đề của mật mã khóa bí mật

- Một số vấn đề của mật mã khóa bí mật
  - Vấn đề trao đổi khóa:
    - Khóa - bí mật chia sẻ chỉ giữa 2 bên gửi nhận
    - Cần thống nhất, trao đổi
    - Kênh trao đổi khóa an toàn không dễ thiết lập
  - Vấn đề chống chối bỏ
    - Bên nhận có thể làm được mọi điều bên gửi có thể và ngược lại
    - Không thể đảm bảo chống chối bỏ

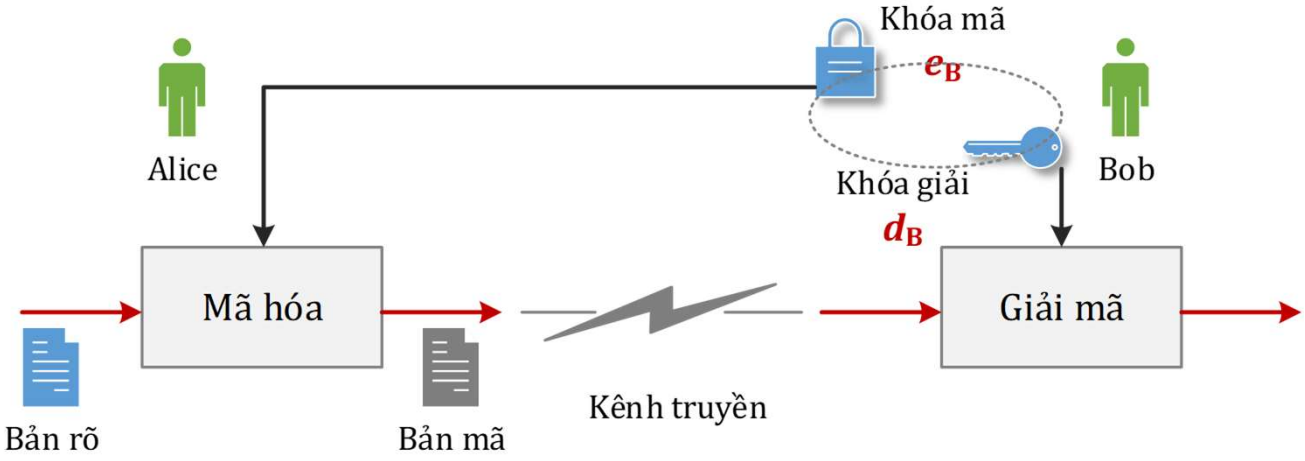
Hệ mật mã khóa công khai

- Mật mã khóa công khai
  - Ý tưởng về hệ thống mã hóa khóa công khai được Martin Hellman, Ralph Merkle và Whitfield Diffie tại Đại học Stanford giới thiệu vào năm 1976.

    Diffie, W.; Hellman, M.E. (November 1976). "New directions in cryptography". IEEE Transactions on Information Theory
  - Sau đó, phương pháp Diffie-Hellman của Martin Hellman và Whitfield Diffie đã được công bố
  - 1977: công bố thuật toán RSA

Hệ mật mã khóa công khai

- Ý tưởng:
  - Mỗi người dùng: sử dụng một cặp khóa (khóa công khai, khóa bí mật)
    - khóa công cộng (public key) được công bố rộng rãi và được sử dụng trong mã hóa thông tin,
    - khóa riêng (private key) chỉ do một người nắm giữ và được sử dụng để giải mã thông tin đã được mã hóa bằng khóa công cộng tương ứng.

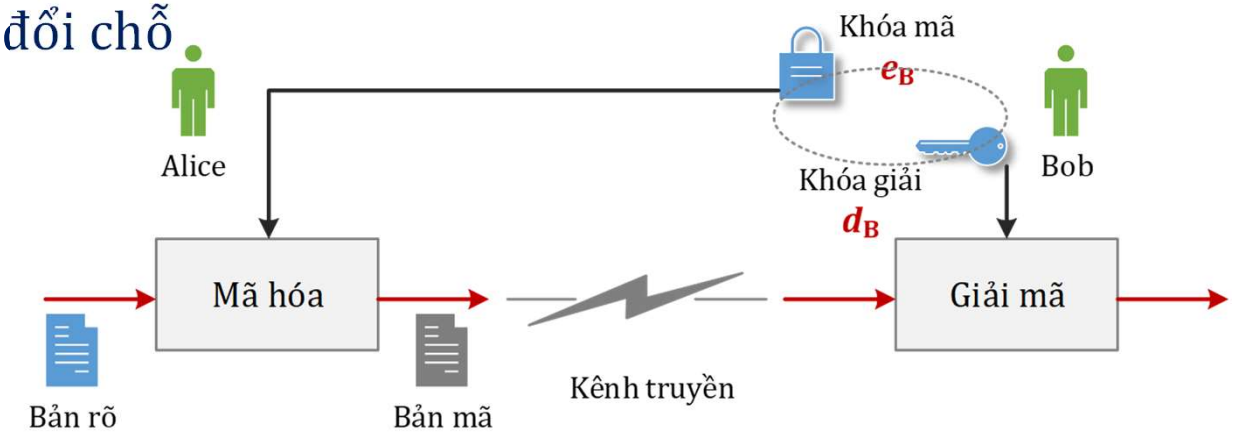


Hệ mật mã khóa công khai

- Ý tưởng:
  - Mỗi người dùng: sử dụng một cặp khóa (khóa công khai, khóa bí mật)
    - Mã hóa: A muốn gửi thông điệp cho B - mã hóa bằng khóa công khai của B

      $y = E(e_B , x)$
    - Giải mã: B giải mã bằng khóa bí mật của mình

      $x = D(d_B , y)$
    - Hàm mã hóa và giải mã có thể đổi chỗ



Hệ mật mã khóa công khai

- Ý tưởng:
  - Các phương pháp mã hóa này khai thác những ánh xạ f mà
    - Biết x, tính y=f(x) dễ dàng
    - Biết y, việc thực hiện ánh xạ ngược f^{-1} tính x là rất khó

  Hàm f có tính chất trên thường gọi là hàm một chiều
  - Ví dụ:
    - Cho các số nguyên tố p_1 , p_2 , … , p_n
    - Tính N= p_1 * p_2 * … * p_n - dễ
    - Ngược lại, biết N, tìm p_1 , p_2 , … , p_n là khó

Hệ mật mã khóa công khai

- Ý tưởng:
  - Hàm cửa sập (trap door):
    - Để xây dựng hệ mã khóa công khai - thường dùng hàm một chiều đặc biệt có tham số/cửa sập:

      nếu biết tham số thì tính f^{-1} dễ dàng
  - Hàm cửa sập => xây dựng mật mã khóa công khai
    - Hàm mã hóa - là hàm cửa sập
    - Khóa (bí mật) - chính là thông tin tham số - bẫy trap door

2/ Hệ mật RSA

- Về hệ mật
  - Năm 1977, R.L.Rivest, A.Shamir và L.Adleman đề xuất hệ thống mã hóa khóa công cộng RSA (hay còn được gọi là “hệ thống MIT”).

    Năm 1977, Ronald Rivest, Adi Shamir và Leonard Adleman đã công bố chi tiết về RSA trên "The Scientific American"
  - Cuối 1977: MIT đăng ký patern: Cryptographic communications system and method, được cấp U.S. Patent 4,405,829 năm 1983

Hệ mật RSA

- Bài toán
  - Bài toán phân tích ra thừa số nguyên tố (lớn)

    chọn p , q - 2 số nguyên tố

    tính N = p.q dễ

    biết N tìm p , q - khó

Hệ mật RSA

- Mô tả hệ mật
  - Các phép tính được thực hiện trên Z_n với $n = p.q$
  - S=<P, C, K, E, D>

$n = pq$ với p và q là hai số nguyên tố lẻ phân biệt. $\phi(n) = (p-1)(q-1)$

$P = C = Z_n$

$K = \{ (n, p, q, a, b) : n = pq, p, q$ là số nguyên tố, $ab \equiv 1 \bmod \phi(n) \}$

Với mỗi $k = (n, p, q, a, b) \in K$, định nghĩa:

$e_k(x) = x^b \bmod n$ và

$d_k(y) = y^a \bmod n$, với $x, y \in Z_n$

Hệ mật RSA

- Áp dụng hệ mật
  - Sinh hai số nguyên tố có giá trị lớn: p và q
  - Tính $n = pq$ và $\phi(n) = (p - 1)(q - 1)$
  - Chọn ngẫu nhiên một số nguyên e $(1 < e < \phi(n))$ thỏa $\gcd(e, \phi(n)) = 1$
  - Tính giá trị $d = e^{-1} \bmod \phi(n)$ (bằng thuật toán Euclide mở rộng)
  - Giá trị n và e được công bố (khóa công khai = (n,e))
  - giá trị p, q, d được giữ bí mật (khóa bí mật = (p,q,d))

Hệ mật RSA

- Ví dụ

Giả sử B chọn p = 101 và q = 113, khi đó n = 11413 và $\phi(n) = 11200$.

Giả sử B chọn b = 3533 , khi đó bằng thuật toán Euclidean mở rộng ta tính được

$$a = b^{-1} = 6597 \bmod 11200.$$

B công khai bộ (n=11413, b=3533)

Bây giờ giả sử A muốn gửi từ hiện 9726 cho B, A sẽ tính

$$9726^{3533} \bmod 11413 = 5761,$$ là từ mã.

Khi B nhận được 5761, anh ta sẽ tính

$$5761^{6597} \bmod 11413 = 9726$$

và thu được từ hiện A muốn gửi.

Cài đặt thuật toán

- Tìm số nguyên số
- Tìm số nghịch đảo
- Lũy thừa số lớn

Giả sử B chọn p = 101 và q = 113, khi đó n = 11413 và $\phi(n) = 11200$.

Giả sử B chọn b = 3533 , khi đó bằng thuật toán Euclidean mở rộng ta tính được

$a = b^{-1} = 6597 \bmod 11200$.

B công khai bộ (n=11413, b=3533)

Bây giờ giả sử A muốn gửi từ hiện 9726 cho B, A sẽ tính

$9726^{3533} \bmod 11413 = 5761$, là từ mã.

Khi B nhận được 5761, anh ta sẽ tính

$5761^{6597} \bmod 11413 = 9726$

và thu được từ hiện A muốn gửi.

Cài đặt thuật toán - lũy thừa số lớn

- Tính lũy thừa số lớn
  - Thuật toán bình phương và nhân
    - Tính giá trị của biểu thức $z = x^b \bmod n$

    Biểu diễn nhị phân $b = b_{l-1}b_{l-2} \dots b_0, b_i \in \{0,1\}, 0 \leq i < l$

    $z = 1$

    $x = x \bmod n$

    for $i = l$-1 **downto** 0

      $z = z^2 \bmod n$

      if $b_i = 1$ **then** $z = z \times x \bmod n$

    end for

Cài đặt thuật toán - lũy thừa số lớn

- Tính lũy thừa số lớn
  - Ví dụ: tính $z = 2^8 \bmod 15$

8=1000

x=2

n=15

Lặp-i	z	Z=z2 mod	Bi	Z=z.x mod	Z
3	1	1	1	2	2
2	2	4	0	-	4
1	4	1	0	-	1
0	1	1	0	-	1

$z = 1$

$x = x \bmod n$

for $i = l\text{-}1$ **downto** 0

$z = z^2 \bmod n$

if $b_i = 1$ **then** $z = z \times x \bmod n$

end for

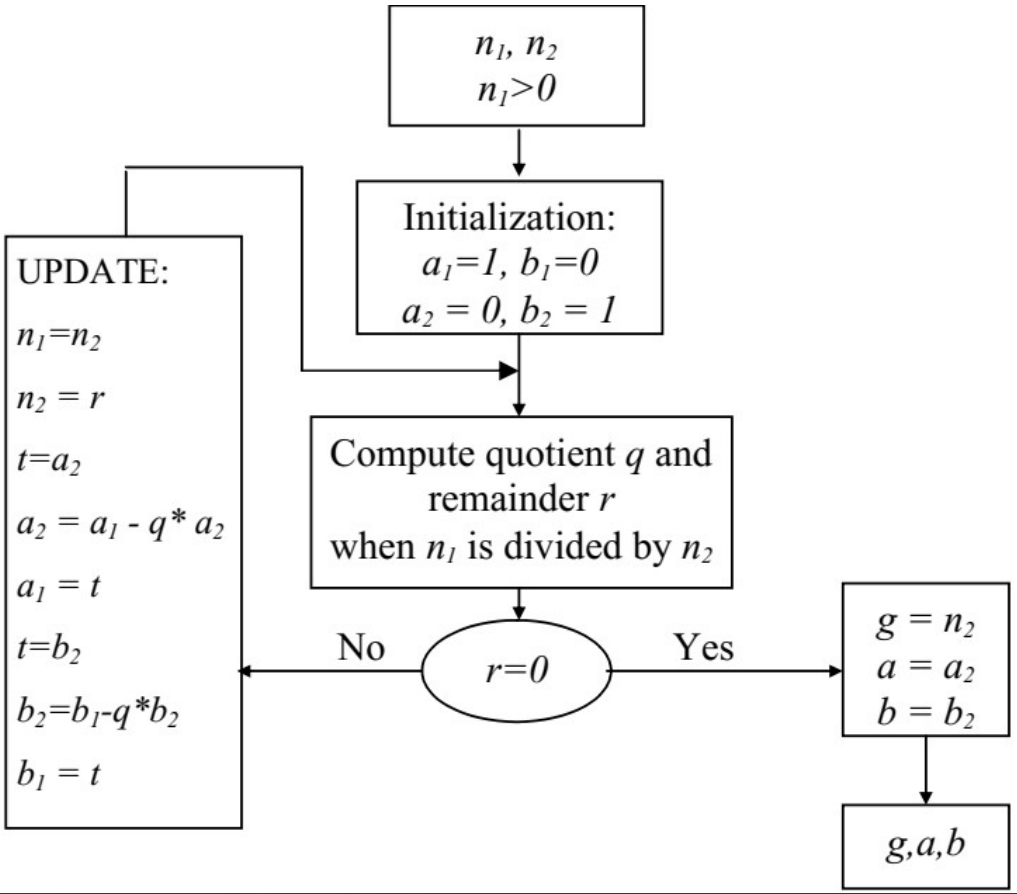
Cài đặt thuật toán - số nghịch đảo

- Thuật toán tính số nghịch đảo $a = b^{-1} \bmod n$
  - Dựa trên thuật toán tìm ƯCLN:

thuật toán Euclide mở rộng

  - Tính nghịch đảo của n1 mod n2:

$a = a2$ - là nghịch đảo



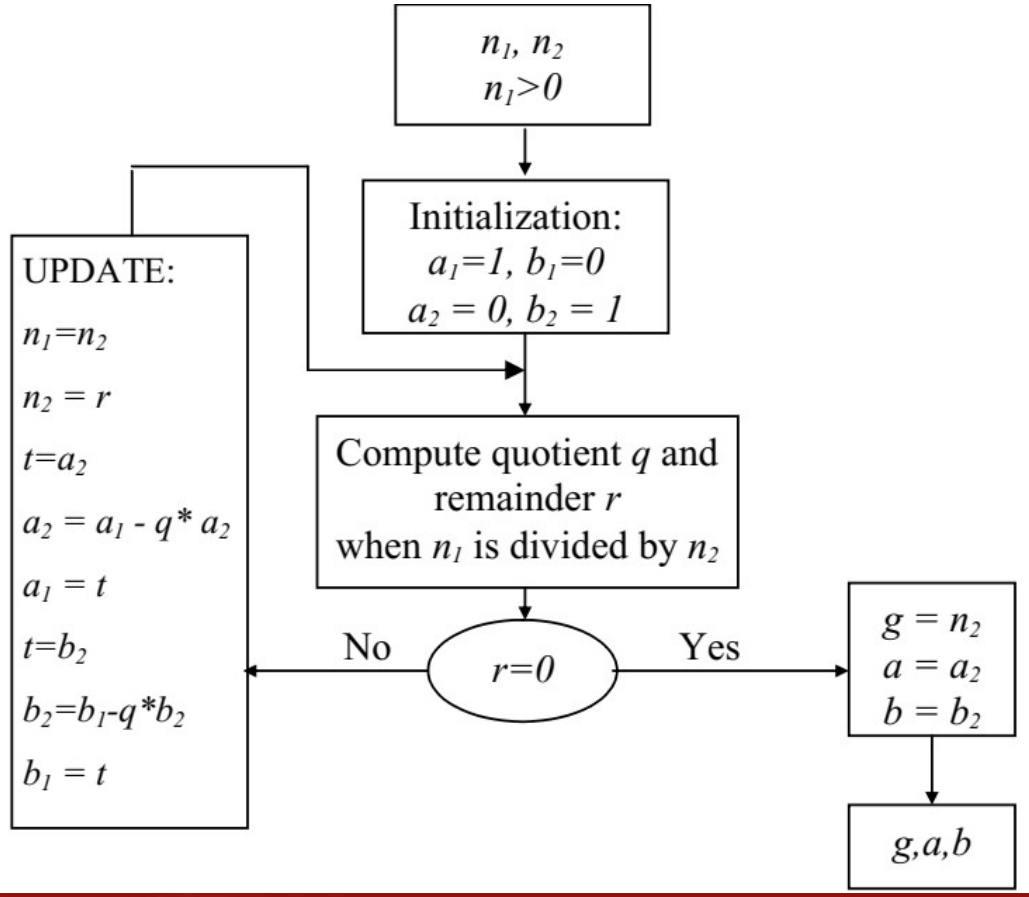
Cài đặt thuật toán - số nghịch đảo

- Thuật toán tính số nghịch đảo $a = b^{-1} \bmod n$
  - ví dụ: Tìm nghịch đảo của 39 mod 11

n1=39, n2=11

n_1	n_2	r	q	a_1	b_1	a_2	b_2
39	11	6	3	1	0	0	1
11	6	5	1	0	1	1	-3
6	5	1	1	1	-3	-1	4
5	1			-1	4	2	-7

$a=a2=2=39^{-1} \bmod 11$



Cài đặt thuật toán - Tìm số nguyên tố

- Tìm số nguyên tố (lớn) p
  - Không tồn tại thuật toán để tạo ra tất cả các số nguyên tố =>
    Thay vì tìm số nguyên tố, đưa về bài toán chọn ngẫu nhiên một số nguyên n và kiểm tra n có phải số nguyên tố
  - Kiểm tra tính nguyên tố:
    - Có thể kiểm tra chính xác - thuật toán tất định:

      chia n cho k chạy từ 2 đến $\sqrt{n}$

      thuật toán không hiệu quả

Cài đặt thuật toán - Tìm số nguyên tố

- Tìm số nguyên tố (lớn) *p*
  - Kiểm tra tính nguyên tố:
    - thuật toán xác suất kiểu MonteCarlo:

    Thuật toán xác suất: thuật toán có sử dụng các số ngẫu nhiên
    - *Thuật toán Monte-Carrlo hướng đúng (yes-biase) là thuật toán xác suất cho bài toán quyết định, trong đó câu trả lời "khẳng định" luôn đúng, còn câu trả lời "phủ định" có thể không đúng. Tương tự ta có thuật toán Monte-Carrlo hướng sai (no-biase)*.
    - Sai số: thuật toán trả lời sai với xác suất không quá *e*

Cài đặt thuật toán - Tìm số nguyên tố

- Tìm số nguyên tố (lớn) p
  - Thuật toán Soloway - Strassen
  - Thuật toán Miller - Rabin
    - Là thuật toán Monte Carlo hướng đúng cho bài toán hợp số !

Cài đặt thuật toán - Tìm số nguyên tố

- Thuật toán Miller - Rabin
  - Mô tả thuật toán:

    $n - 1 = 2^k m$ (m lẻ)

    chọn ngẫu nhiên a: $1 \leq a \leq n - 1$

    $b = a^m \ mod \ n$

    if ($b \equiv 1 \ mod \ n$) { return (n là nguyên tố) }

    for (i=0 to k-1)

    {

    if($b \equiv 1 \ mod \ n$) {return (n là nguyên tố) }

    else { $b = b^2 \ mod \ n$}

    }

    n là hợp số
  - Xác suất lỗi: 0.25

Cài đặt thuật toán - Tìm số nguyên tố

- Thực tế:
  - Mật độ số nguyên tố

    Số lượng số nguyên tố < N: $\frac{N}{lnN}$

    Chọn ngẫu nhiên số p , xác suất để p là nguyên tố: $\frac{1}{lnN}$

    Nếu p cỡ 512 bit thì xác suất ~ 1/177
  - Xác suất lỗi của thuật toán Miller Rabin: 0.25

    Để tin cậy => thử *nhiều* lần:

STT	2^{-m}
1	0.500
2	0.250
5	0.312×10^{-1} .
10	0.977×10^{-3} .
20	0.954×10^{-6} .
50	0.888×10^{-15} .
100	0.789×10^{-30} .

Một số vấn đề khi cài đặt RSA

- Chọn số nguyên tố p, q
  - Giả định độ mật dựa trên độ khó phân tích thừa số nguyên tố
  - p , q phải đủ lớn (chẳng hạn từ 100 chữ số thập phân, 300 bit)
  - p và q tương đương

- Bản rõ không luôn được mã hóa
  - Có những bản rõ kh được mã hóa tốt

VD:

$n = 35 = 5 \times 7,$ $e = 5$ $X = 8$ $Y = X^e \bmod 35 = 8 = X$	$m = 4$ x 6 $(GCD\ (5,24) = 1)$

Tấn công hệ mã RSA

- RSA là hệ mật khóa công khai => nên việc tấn công thường dựa vào khóa công khai (n, b) để xác định được khóa bí mật tương ứng (p, q, a)

  => Tấn công hiển nhiên là phân tích modulus n để xác định p, q

  • Thay vì tìm cách phân tích n, có thể tìm $\phi(n)$

  • Độ khó tìm $\phi(n) = (p-1)\,(q-1)$ còn lớn hơn việc phân tích n ra thừa số

- Là hệ mật khóa công khai

  => thay vì phân tích n, có thể

  • Tìm khóa bí mật d=a

  Biết các cặp (X, Y): Y=ek(X) => để tìm khóa bí mật: giải phương trình $X = Y^a \ mod \ n$

  • Tìm bản rõ X khi nghe được bản mã Y

  Tìm căn thức đồng dư $Y = X^a$

Tấn công hệ mật RSA

Cài đặt không an toàn:

- Sử dụng chung modulus
  - Một nhóm nhiều người dùng các khóa công khai (n,e): chung modulus n
  - Khi đó:
    - nếu thu được 2 bản mã Y1, Y2 - mã từ cùng bản rõ X
    - các thành phần e1, e2 nguyên tố cùng nhau

    thì có cách để giải được bản mã hiệu quả:
    - *do UCLN (e1, e2) =1 => tìm được a, b để: a*e1+b*e2 = 1*

    từ đó: $Y_1^a * Y_2^b = X^{a*e1} * X^{b*e2} = X^{a*e1+b*e2} = X^1 = X mod\ n$
- Khóa công khai (n,e) có thành phần e nhỏ:
- Khóa bí mật (p,q,d) có d nhỏ

Thuật toán phân tích ra thừa số: p-1

- Thuật toán Pollard p-1 (1974) là một trong những thuật toán đơn giản hiệu quả dùng để phân tích ra thừa số nguyên tố các số nguyên lớn.
- Input
  - n (lẻ) - số cần phân tích
  - và B - ngưỡng cho trước
-

Thuật toán phân tích ra thừa số: p-1

Input: n , B

```
1. a = 2
2. for j = 2 to B do
       a = a^j mod n
3. d = gcd(a – 1, n)
4. if 1 < d < n then
       d là thừa số của n (thành công)
   else
       không xác định được thừa số của n
       (thất bại)
```

Thuật toán phân tích ra thừa số p-1

- Ví dụ:
  - Giả sử n = 15770708441, B=180
  - Áp dụng thuật toán $p-1$ với $B = 180$, chúng ta xác định được $a = 1162022142^5$ ở bước 3 của thuật toán
  - và xác định được giá trị $d = 135979$.
  - Trong trường hợp này, việc phân tích ra thừa số nguyên tố thành công do giá trị 135978 chỉ có các thừa số nguyên tố nhỏ khi phân tích ra thừa số nguyên tố:

$$135978 = 2 \times 3 \times 131 \times 173$$

  - Do đó, khi chọn $B \geq 173$ sẽ đảm bảo điều kiện $135978 \mid B!$
- Hệ mật an toàn với p-1 : chọn số nguyên tố $p1$ lớn, mà $p = 2p_1 + 1$ cũng là số nguyên tố

Thuật toán phân tích ra thừa số p-1

- Giải thuật Pollard hiệu quả khi n có thừa số nguyên tố p mà p-1 phân tích thành các thừa số nguyên tố nhỏ.
- Hệ mật an toàn với p-1 : chọn số nguyên tố $p1$ lớn, mà $p = 2p_1 + 1$ cũng là số nguyên tố

3/ So sánh với mật mã khóa bí mật

So sánh DES và RSA

- Nói chung: tốc độ mã hóa, giải mã của mật mã khóa bí mật nhanh hơn nhiều mật mã khóa công khai
  - DES: nhanh hơn RSA 100 (cài đặt phần mềm) đến 10000 lần (cài đặt phần cứng)
- Kích cỡ khóa: để đảm bảo an toàn, nói chung kích cỡ khóa của RSA lớn hơn nhiều
  - giải thuật sàng số: thời gian là $L(n) \approx 10^{9.7+\frac{1}{50}\log_2 n}$
  - đã từng phân tích được số nguyên n 768 bit ~ 232 chữ số, là tích của 2 số ng tố (cỡ bằng nhau)
  - Để đảm bảo an toàn, hiện nay đề nghị khóa tối thiểu 1024 bit

So sánh

- Vấn đề của mật mã khóa bí mật
  - Quản lý khóa
  - Trao đổi khóa
- Vấn đề của mật mã khóa công khai
  - Khóa công khai dễ bị tấn công hơn khóa bí mật
  - Tốc độ, chi phí tính toán
  - Kích thước khóa

So sánh

- Áp dụng kết hợp
  - Trao đổi dữ liệu: mật mã khóa bí mật => cần thống nhất và trao đổi khóa
  - Trao đổi khóa: dùng mật mã khóa công khai
  - Sơ đồ này được áp dụng khá phổ biến