**BG Bài 5.2. Sơ đồ chữ ký ElGamal và chuẩn chữ ký điện tử.**

**2.1. Sơ đồ chữ ký ElGamal.**

Sơ đồ chữ ký ElGamal được đề xuất năm 1985, gần như đồng thời với sơ đồ hệ mật mã ElGamal, cũng dựa trên độ khó của bài toán lôgarit rời rạc.

Sơ đồ được thiết kế đặc biệt cho mục đích ký trên các văn bản điện tử, được mô tả như một hệ

*S*  **=** (, , , , ),

trong đó = =, với *p* là một số nguyên tố sao cho bài toán tính lôgarit rời rạc trong là rất khó. Tập hợp gồm các cặp khoá *K* =(*K’*,*K''*), với *K’* = *a* là một số thuộc , *K''* =(*p*, *α*, *β* ), *α* là một phần tử nguyên thuỷ của , và *β* = *α* *a* mod*p*. *K’* là khoá bí mật dùng để ký, và *K''* là khoá công khai dùng để kiểm thử chữ ký. Các thuật toán ký và kiểm thử chữ ký được xác định như sau:

Với mỗi thông báo *x*, để tạo chữ ký trên *x* ta chọn thêm môt số ngẫu nhiên *k* ∈, rồi tính

(*x*,*k* ) =  với

mod*p*,

mod(*p* -1).

Thuật toán kiểm thử được định nghĩa bởi:

(*x*,) = đúng ⇔ *β γ.γ δ* ≡ *αh(x)* (mod*p*).

Dễ thấy rằng sơ đồ chữ ký được định nghĩa như trên là hợp thức.

Thực vậy, nếu

(*x*,*k* ) = 

thì ta có :

*β γ.γ δ* ≡ *αaγ.αkδ* mod*p*

≡ *α x* mod*p*,

vì *kδ* +*aγ* ≡ *x* mod(*p* -1). Do đó, (*x*,) = đúng.

Thí dụ: Giả sử *p* = 467, *α* = 2, *a* = 127. Khi đó *β* = 2127mod467=132. Cho *x* =100; ta chọn ngẫu nhiên *k* =213 (∈**) và được *k* -1mod466 =431. Chữ ký trên văn bản *x* =100 với số ngẫu nhiên *k* =213 là (*γ, δ*), trong đó *γ* =2213mod467 = 29 và *δ* = (100 - 127.29).431mod466 =51.

Để kiểm thử ta tính :

*β γ.γ δ* = 13229.2951 ≡ 189 (mod467),

*α x* = 2100 ≡ 189 (mod467),

hai giá trị đó đồng dư với nhau theo mod467, chữ ký(*γ, δ*)=(29,51) được xác nhận là đúng.

Bài tập

Bước 1. Công bố cho đối tác khóa công khai của hệ mật ElGamal.

Bước 2. Dùng khóa công khai của đối tác B để mã hóa bản tin x.

Bước 3. Dùng khóa bí mật của hệ mật ElGamal của mình để ký trên bản tin x và lấy h(x) = x.

Bước 4. Kiểm thử chữ ký của A xem có đúng không.

Bước 5. Kết luận

Bài tập

Anh Vũ Đức Nguyên p1 =127, tìm α1 , chọn a=61, Tính β1=. Công bố khóa công khai.

Anh Nguyễn Lê Hoàng p2 =149, tìm α2, chọn a2=97, Tính β2= . Công bố khóa công khai.

Bước 1. Anh Nguyên gửi bản tin x=101, và ký trên x.

Số ngẫu nhiên k1=22 để mã hóa và số k2=83 để ký.

Bản mã:

Chữ ký :

Bước 2. Anh Hoàng nhận được bản mã và chữ ký, tiến hành giải mã và xác thực chữ ký.

Bài kiểm tra

Với p đã cho, hãy :

1. Xác định phần tử nguyên thủy α của Zp\*;
2. Xây dựng hệ mật ElGamal với a = 1010.
3. Hai sv ghép thành 1 cặp để nhận và gửi bản mã, ký trên bản tin x=số hóa 3 chữ cái đầu tên của sinh viên.

Với kmã hóa=853, kký = 467.

Mỗi sinh viên vừa mã hóa, gửi tin, ký trên bản tin và nhận tin, giải mã, xác thực chữ ký.

**5.2.2. Tính an toàn của sơ đồ chữ ký ElGamal.**

Sơ đồ chữ ký ElGamal được xem là an toàn, nếu việc ký trên một văn bản là không thể giả mạo được, nói cách khác, không thể có một người nào ngoài chủ thể hợp pháp có thể giả mạo chữ ký của chủ thể hợp pháp đó trên một văn bản bất kỳ. Vì vậy, việc giữ bí mật khoá  dùng để tạo chữ ký là có ý nghĩa quyết định đối với việc bảo đảm tính an toàn của chữ ký. Có thể để lộ khoá bí mật  trong những trường hợp nào, và có thể không để lộ  mà vẫn giả mạo chữ ký được không? Ta sẽ xét sau đây một vài trường hợp đơn giản :

1) *Khả năng để lộ khoá* : Cũng như đối với sơ đồ hệ mật mã ElGamal, khoá bí mật *a* có thể bị phát hiện trong trường hợp để lộ số ngẫu nhiên *k* ở một lần ký nào đó, hoặc sử dụng cùng một số ngẫu nhiên *k* ở hai lần ký khác nhau.

Nếu số ngẫu nhiên *k* được sử dụng khi ký trên văn bản *x* bị lộ, thì khoá bí mật  được tính theo công thức sau đây:

*a* = (*x* - *kδ* ). *γ*-1 mod(*p* –1).

Bây giờ ta xét trường hợp dùng cùng một số ngẫu nhiên *k* cho hai lần ký khác nhau, chẳng hạn cho *x*1 và *x*2. Khi đó ta có chữ ký trên *x*1 là (*γ* ,*δ* 1), trên *x*2 là (*γ* ,*δ* 2), với thành phần thứ nhất bằng nhau (và bằng *γ* =*α k* mod*p*), và các chữ ký đó thoả mãn

(mod*p*),

(mod*p*).

Từ đó ta có

(mod*p*),

điều đó tương đương với

*x* 1 - *x*2 ≡ *k* (*δ*1 - *δ* 2) (mod(*p* -1)).

Đặt *d* = gcd(*δ*1 - *δ* 2, *p* -1). Cả ba số *δ*1 - *δ* 2, *p* -1 và *x* 1 - *x*2 đều chia hết cho *d*, ta đặt



Khi đó đồng dư thức ở trên trở thành

(mod).

Vì gcd=1, nên có thể tính mod, và sau đó giá trị *k* theo mod:

*k* = mod, tức là

*k* = mod(*p* -1)

với *i* là một giá trị nào đó, 0≤ *i* ≤ *d* –1. Thử lần lượt điều kiện

mod*p*

với các giá trị đó của *i* , ta sẽ tìm được *k* ;sau đó từ *k* tính được *a* cần tìm.

2) *Khả năng giả mạo chữ ký trên một văn bản cho trước* :

Giả sử chủ thể A chọn sơ đồ chữ ký ElGamal với cặp khoá *K* =(*K’*,*K''*), trrong đó  là khoá bí mật. Một người ngoài O không biết khoá bí mật  mà muốn giả mạo chữ ký của A trên một văn bản *x* thì phải có khả năng tạo ra được chữ ký (*γ*,*δ* ) mà không cần biết *a*. Có hai cách : hoặc chọn trước *γ* rồi tìm *δ* tương ứng, hoặc ngược lại, chọn trước *δ* rồi tìm *γ* tương ứng.

Nếu chọn trước *γ* rồi tìm *δ* , thì *δ* phải là

mod(*p* -1)

= ( mod(*p* -1)

= mod(*p* -1);

đó là một bài toán tính lôgarit rời rạc, mà ta biết rằng rất khó.

Nếu chọn trước *δ* rồi tìm *γ* thì phải giải phương trình

 mod*p*

với ẩn số *γ* . Ta chưa biết có cách giải hữu hiệu nào không, nhưng chắc là không dễ hơn bài toán tính lôgarit rời rạc.

Như vậy, ta có thể tin rằng khả năng giả mạo chữ ký trên một văn bản cho trước khi không biết khoá bí mật  là rất ít, do đó không có ảnh hưởng đáng kể đến tính an toàn của sơ đồ chữ ký.

3)*Giả mạo chữ ký cùng với văn bản được ký* :

Có một khả năng giả mạo khác là giả mạo cả văn bản gửi đi *x* cùng với chữ ký (*γ*,*δ* ) trên *x*. Khả năng đó xẩy ra khi kẻ giả mạo chọn được *x* và (*γ*,*δ* ) thoả mãn điều kiện kiểm thử, cụ thể khi chọn được *x*,*γ*,*δ* có dạng sau đây :

mod*p*,

 mod(*p* -1),

mod(*p* -1),

trong đó *i, j* là các số nguyên sao cho 0≤ *i, j* ≤ *p* –2, gcd(*j*, *p* –1) = 1, và *j* –1 được tính theo mod(*p* –1). Thực vậy, khi đó ta có

mod*p*

 mod*p*

 mod*p* ,

tức điều kiện kiểm thử được thoả mãn, (*γ*,*δ* ) có thể được xác nhận hợp thức là chữ ký trên *x*.

Có thể có một cách giả mạo khác nữa, nếu kẻ giả mạo sử dụng chữ ký đúng (*γ*,*δ* ) trên một văn bản *x* có từ trước để tạo ra một chữ ký mới cho một văn bản “mới” như sau:

 mod*p*,

 mod(*p* -1),

mod(*p* -1).

Có thể thử lại rằng điều kiện kiểm thử đúng đối với “chữ ký” và “văn bản” , tức là

mod*p*.

Cả hai cách giả mạo nói trên đều cho chữ ký thoả mãn điều kiện kiểm thử đối với văn bản tương ứng, tuy nhiên văn bản đó không phải là văn bản được chọn theo ý muốn của người giả mạo, cho nên khả năng sử dụng các cách giả mạo đó trong thực tế cũng không có giá trị , do đó không thể gây nguy hại đáng kể cho tính an toàn của sơ đồ chữ ký nói chung.

**2.3. Chuẩn chữ ký số (Digital Signature Standard).**

Chuẩn chữ ký số (DSS) được đề xuất từ năm 1991 và được chấp nhận vào cuối năm 1994 để sử dụng trong một số lĩnh vực giao dịch điện tử tại Hoa kỳ. DSS dựa vào sơ đồ chữ ký ElGamal, với một vài sửa đổi. Để bảo đảm an toàn , số nguyên tố *p* cần phải đủ lớn, biểu diễn nhị phân của *p* phải có từ 512 bit trở lên (cụ thể từ 512 đến 1024 bit, số bit là một bội của 64). Tuy nhiên, độ dài chữ ký theo sơ đồ ElGamal là gấp đôi số bit của *p*, mà trong nhiều ứng dụng người ta lại mong muốn có chữ ký độ dài ngắn, nên giải pháp sửa đổi được đề xuất là: trong khi vẫn dùng *p* lớn với độ dài biểu diễn 512 bit trở lên, thì sẽ hạn chế độ dài của *γ* và *δ*  trong chữ ký (*γ*,*δ* ) vào khoảng 160 bit (như vậy cả chữ ký sẽ có độ dài khoảng 320 bit); điều này được thực hiện bằng cách dùng một nhóm con cyclic  của  thay cho chính bản thân , do đó mọi tính toán vẫn được thực hiện như trong  nhưng các dữ liệu và thành phần chữ ký lại thuộc  . Ta được *sơ đồ chuẩn chữ ký số DSS* như mô tả sau đây:

Chọn *p* là một số nguyên tố lớn có độ dài biểu diễn ≥ 512 bit sao cho bài toán tính logarit rời rạc trong *Zp* là khó, *q* là một ước số nguyên tố của *p* -1, có độ dài biểu diễn cỡ 160 bit. Gọi *α* ∈ là một căn bậc *q* của 1 theo mod*p*.

Đặt =, = ×. Chọn  và tính mod*p*. Xác định khoá *K* =(*K’*,*K''*), trong đó khoá bí mật *K’* = *a*, và khoá công khai *K''* = (*p,q,α,β*). Thuật toán ký và thuật toán kiểm thử được định nghĩa như sau: Với *x* ∈ =, ta chọn thêm một số ngẫu nhiên *k* (0≤*k* ≤ *q* -1), và định nghĩa chữ ký

(*x*,*k* ) =  trong đó mod*p*) mod*q*,

mod*q*.

Thuật toán kiểm thử được định nghĩa bởi:

(*x*,) = đúng ⇔ mod*p*)mod*q* = *γ* ,

trong đó mod*q* và mod*q*.

Chú ý rằng ta phải có *δ* ≠ 0 mod*q* để có thể tính được *δ* -1mod*q* dùng trong thuật toán kiểm thử, vì vậy nếu chọn *k*  mà được *δ* ≡ 0 mod*q* thì phải chọn lại số *k* khác để có được *δ* ≠ 0 mod*q*.