TCVN TIÊU CHUẨN QUỐC GIA

TCVN 12214-3:2018
ISO/IEC 14888-3:2016

Xuất bản lần 1

CÔNG NGHỆ THÔNG TIN - CÁC KỸ THUẬT AN TOÀN - CHỮ KÝ SỐ KÈM PHỤ LỤC - PHẦN 3: CÁC CƠ CHẾ DỰA TRÊN LOGARIT RỜI RẠC

Information technology - Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms

HÀ NỘI - 2018

Mục Lục

Lời nói đầu

TCVN 12214-3 : 2018 hoàn toàn tương đương với ISO/IEC 14888-3:2016.

TCVN 12214-3 : 2018 do Cục Quản lý mật mã dân sự và Kiểm định sản phẩm mật mã biên soạn, Ban Cơ yếu Chính phủ đề nghị, Tổng cục Tiêu chuẩn Đo lường Chất lượng thẩm định, Bộ Khoa học và Công nghệ công bố.

Bộ tiêu chuẩn TCVN 12214 (ISO/IEC 14888) *Công nghệ thông tin - Các kỹ thuật an toàn – Chữ ký số kèm phụ lục* gồm các tiêu chuẩn sau:

- TCVN 12214-1 : 2018 (ISO/IEC 14888-1:2008) Phần 1: Tổng quan

- TCVN 12214-2 : 2018 (ISO/IEC 14888-2:2008) Phần 2: Các cơ chế dựa trên phân tích số nguyên

- TCVN 12214-3 : 2018 (ISO/IEC 14888-3:2016) Phần 3: Các cơ chế dựa trên logarit rời rạc

TIÊU CHUẨN QUỐC GIA TCVN 12214-3 : 2018

Công nghệ thông tin - Các kỹ thuật an toàn - Chữ ký số kèm phụ lục Phần 3: Các cơ chế dựa trênlogarit rời rạc

Information technology - Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms

1 Phạm vi áp dụng

Tiêu chuẩn này quy định các cơ chế chữ ký số kèm phụ lục, mà độ an toàn của nó dựa trên bài toán logarit rời rạc.

Tiêu chuẩn này cung cấp:

- Một sự mô tả chung của sơ đồ chữ ký số kèm phụ lục

- Một loạt các cơ chế cung cấp chữ ký số kèm phụ lục.

Với mỗi cơ chế, tiêu chuẩn này mô tả:

- Tiến trình sinh cặp khóa,

- Tiến trình tạo chữ ký,

- Tiến trình kiểm tra chữ ký.

2 Tài liệu viện dẫn

Các tài liệu dưới đây, toàn bộ hoặc một phần, được tham chiếu trong tài liệu này rất cần thiết cho các ứng dụng của nó. Đối với các tài liệu viện dẫn ghi năm công bố, chỉ áp dụng các bản được nêu. Đối với các tài liệu viện dẫn không ghi năm công bố, thì bản tham chiếu cuối cùng được áp dụng.

TCVN 11816-3 (ISO/IEC 10118-3), *Công nghệ thông tin – Các kỹ thuật an toàn – Hàm băm*

TCVN 12214-1 (ISO/IEC 14888-1), *Công nghệ thông tin – Các kỹ thuật an toàn – Chữ ký số kèm phụ lục - Phần 1: Tổng quan*

3 Các thuật ngữ và định nghĩa

Với mục đích của tài liệu này, các thuật ngữ và định nghĩa dưới đây được áp dụng.

3.1

Nhóm giao hoán hữu hạn (finite commutative group)

Tập hợp hữu hạn E với phép toán nhị phân "*" sao cho:

- Với mọi phần tử nhóm $a, b \in E$, $a * b \in E$;
- Với mọi phần tử nhóm $a, b, c \in E$, $(a * b) * c = a * (b * c)$;

- Tồn tại một phần tử nhóm $e \in E$ với $e * a = a$ với mọi phần tử $a \in E$, với e được gọi là phần tử đơn vị của nhóm;
- Với mọi phần tử nhóm $a \in E$, tồn tại một phần tử nhóm $b \in E$ sao cho $b * a = e$;
- Với mọi phần tử nhóm $a, b \in E, a * b = b * a$.

CHÚ THÍCH 1 Trong một vài trường hợp, khi E là tập các điểm trên một đường cong elliptic, phép toán số học trong tập hợp hữu hạn E được mô tả với ký hiệu cộng.

3.2

Nhóm cyclic (cyclic group)

Nhóm giao hoán hữu hạn (3.1) E của n phần tử chứa một phần tử nhóm $a \in E$, được gọi là phần tử sinh có bậc n.

3.3

Nhóm đường cong elliptic (elliptic curve group)

Nhóm cyclic (3.2) được định nghĩa trên các điểm của một đường cong elliptic trên trường hữu hạn.

CHÚ THÍCH 1 Giả sử $F = GF(r)$ là ký hiệu của trường Galois với lực lượng r, với r hoặc là số nguyên tố lẻ p, hoặc $r = 2^m$, với một số m nguyên dương.

Một đường cong elliptic được định nghĩa trên F có thể được xác định bởi công thức đường cong affine, hoặc là $y^2 = x^3 + a_1x + a_2$ (với $r = p, p$ là một số nguyên tố lẻ) hoặc công thức $y^2 + xy = x^3 + a_1x^2 + a_2$ (khi $r = 2^m$, với m là số nguyên dương), với a_1 và a_2 là các phần tử của F. Đường cong elliptic E tương ứng bao gồm tập hợp các điểm affine $F \times F$ cùng với một điểm đặc biệt (không affine) điểm "vô cực".

Một điểm affine P của E có thể được biểu diễn là một cặp thứ tự $(P_x, P_y) \in F \times F$, sao cho sự lựa chọn $x = P_x$ và $y = P_y$ thỏa mãn công thức của đường cong affine đã cho và khi các phép toán số học được thực hiện trên trường F.

Giả sử "+" là ký hiệu của phép toán hai ngôi là phép toán cộng trên "đường cong elliptic", được xác định cho các điểm affine của E theo quy tắc cát tuyến - tiếp tuyến. Khi, tập hợp các điểm affine của E được tăng lên bởi 0_E, một điểm đặc biệt của E "ở vô cực" đóng vai trò như là yếu tố đối với phép "+" (nhưng nó không được biểu diễn như một cặp trong tọa độ), tập hợp E cùng với phép "+" tạo thành một nhóm hữu hạn, giao hoán, nhóm đường cong elliptic, E.

CHÚ THÍCH 2 Lực lượng của nhóm đường cong E, nhiều hơn một so với số lượng cặp thứ tự trong $F \times F$ thỏa mãn công thức đường cong affine với E.

3.4

Bậc (của một phần tử nhóm *a*) (order (of a group element a)

Số nguyên dương n nhỏ nhất sao cho $a^n = e$, với e là phần tử đơn vị của nhóm, a^n được định nghĩa truy hồi sao cho $a^0 = e$ và $a^n = a * a^{m-1} (m > 0)$, và * là phép toán nhóm.

3.5

Cặp (pairing)

Hàm số lấy hai phần tử P và Q từ nhóm đường cong elliptic (3.3) trên trường hữu hạn G_1 như là đầu vào, và sinh ra một phần tử từ nhóm cyclic khác (3.2) trên trường hữu hạn G_2 như là đầu ra, và nó có hai tính chất sau đây (ở đó nó được giả thiết rằng G_1 và G_2 có bậc là q, với q là số nguyên tố, và hai phần tử bất kỳ P, Q, đầu ra của hàm cặp được viết là $<P, Q>$)

- Tính song tuyến: Nếu P, P_1, P_2, Q, Q_1, Q_2 là các phần tử của G_1, và a là một số nguyên thỏa mãn $1 \leq a \leq q - 1$, thì:

$< P_1 + P_2, Q > = < P_1, Q > * < P_2, Q >$,

$< P, Q_1 + Q_2 > = < P, Q_1 > * < P, Q_2 >$,

$< [a]P, Q > = < P, [a]Q > = < P, Q >^a$,

- Tính không suy biến: Nếu P là một phần tử không định danh của G_1, $< P, P > \neq 1$

3.6

Trung tâm sinh khóa tin cậy KGC (trusted key generation centre)

Bên thứ ba tin cậy, một cơ chế chữ kí số dựa trên định danh, sinh ra một khóa riêng cho mỗi người ký.

4 Các ký hiệu và chữ viết tắt

$a \oplus b$	Phép toán XOR được thực hiện trên từng bit của a và b, với a và b là các bit hoặc xâu bit cùng độ dài.
a_1, a_2	Các hệ số của đường cong elliptic
$a \bmod n$	Cho một số nguyên a và một số nguyên dương n, số dư nguyên duy nhất r, $0 \leq r \leq (n-1)$, thỏa mãn $r = a - bn$, với b là một số nguyên nào đó.
(A, B, C)	Các hệ số của công thức chữ ký số, cơ chế được đặc tả trong Điều 6, xác định cách chữ ký được tính toán. CHÚ THÍCH 1 Công thức chữ ký số được đặc tả trong điều 5.2.1. Một tham số được đặc tả trong mối quan hệ giữa khóa tạo chữ ký và khóa kiểm tra
E	Một đường cong elliptic được xác định bởi hai hệ số a_1 và a_2
E	Một nhóm giao hoán hữu hạn ; với các cơ chế dựa trên nhóm nhân, các phần tử của E trong Z_p^*; với các cơ chế dựa trên nhóm cộng của nhóm các điểm trên đường cong elliptic, các phần tử của E là các điểm trên đường cong elliptic E trên $GF(r)$.
$\#E$	Lực lượng của E; với các cơ chế dựa trên nhóm nhân Z_p^*, $\#E$ là $p-1$; với các cơ chế dựa trên nhóm cộng của các điểm đường cong elliptic, $\#E$ nhiều hơn 1 điểm so với số lượng điểm của đường cong elliptic E trên $GF(r)$ [bao gồm 0_E (điểm vô cực)]
F	Một trường hữu hạn
F_p	một trường hữu hạn cấp p
$gcd(N_1, N_2)$	Ước số chung lớn nhất của các số nguyên N_1 và N_2.
G	Một phần tử có bậc q trong E
$GF(r)$	Trường hữu hạn với lực lượng r, với r là lũy thừa của số nguyên tố
G_1	Một nhóm cyclic bậc nguyên tố q; các phần tử của G_1 là các điểm trên đường cong elliptic trên $GF(r)$
G_2	Một nhóm cyclic bậc nguyên tố q; các phần tử của G_2 là các phần tử của một trường hữu hạn $GF(r)$
H_1	Một hàm băm chuyển đổi một xâu dữ liệu vào một phần tử trong G_1 CHÚ THÍCH 2 Xâu dữ liệu đầu vào được biến đổi thành số nguyên, sau đó số nguyên được chuyển thành một điểm của E trên $GF(r)$ bằng cách sử dụng hàm $I2P$, được đặc tả trong Phụ lục C.
h, H_2	Các hàm băm, nghĩa là một trong các cơ chế được đặc tả trong TCVN 11816:2017 (ISO IEC 10118).
ID	Một xâu dữ liệu chứa một định danh của người ký, được dùng trong cơ chế IBS-1 và IBS-2

m	Một bậc nhúng (hoặc bậc mở rộng)
$[n]P$	Phép toán nhân lấy đầu vào là một số nguyên dương n và một điểm P trên đường cong E và đưa ra đầu ra một điểm Q trên đường cong E, với $Q = [n]P = P + P + \cdots + P$, cộng n-1 lần. Phép toán thỏa mãn tính chất $[0]P = 0_E$ (điểm vô cực), và $[-n]P = [n](-P)$
P	Phần tử sinh của G_1 được dùng trong các cơ chế IBS-1 và IBS-2
p	Một số nguyên tố hoặc lũy thừa của một số nguyên tố
q	Một số nguyên tố là ước của $\#E$ và bậc của G_1 và G_2
r	Kích thước của $GF(r)$; trong các cơ chế dựa trên nhóm cộng các điểm của đường cong elliptic, r là lũy thừa của số nguyên tố, p^m, một vài số nguyên tố $p \geq 2$ và số nguyên $m \geq 1$.
T	Nhiệm vụ
T_1	Phần đầu tiên của nhiệm vụ T
T_2	Phần thứ hai của nhiệm vụ T
U	Khóa chủ riêng của KGC, được tạo ra từ việc lựa chọn một số nguyên ngẫu nhiên, được dùng trong các cơ chế IBS-1 và IBS-2
V	Khóa chủ công khai của KGC, một phần tử của G_1, được dùng trong các cơ chế IBS-1 và IBS-2
Z_N^*	Tập hợp các số nguyên i với $0 < i < N$ và $gcd(i, N) = 1$ với các phép toán số học được định nghĩa theo modulo N.
Z_p^*	Tập hợp các số nguyên i với $0 < i < p$ và p là một số nguyên tố, nhóm này là một nhóm nhân.
α	Độ dài của số nguyên tố (hoặc lũy thừa của số nguyên tố) p theo bit.
β	Độ dài của số nguyên tố q theo bit
γ	Độ dài đầu ra của hàm băm h và H_2 theo bit
Π	Tiền chữ ký
Π_X	Tọa độ x của Π trong đó $\Pi = (\Pi_X, \Pi_Y)$ là một điểm của đường cong elliptic
Π_Y	Tọa độ y của Π trong đó $\Pi = (\Pi_X, \Pi_Y)$ là một điểm của đường cong elliptic
Π_a	Phần tử đầu tiên của Π trong đó $\Pi = (\Pi_a, \Pi_b)$ là một phần tử của trường mở rộng bậc 2.
Π_b	Phần tử thứ hai của Π trong đó $\Pi = (\Pi_a, \Pi_b)$ là một phần tử của trường mở rộng bậc 2.
0_E	Điểm ở vô cực trên đường cong elliptic E
<>	Một cặp song tuyến và không suy biến
||	$X\|\|Y$ được dùng với nghĩa của kết quả phép ghép dữ liệu của X và Y theo trật tự được đặc tả

5 Mô hình tổng quát

5.1 Tiến trình tạo tham số

5.1.1 Các cơ chế dựa trên chứng thư

5.1.1.1 Sinh các tham số miền

Đối với các cơ chế chữ ký số dựa trên logarit rời rạc, tập hợp các tham số miền bao gồm các tham số sau:

- E một nhóm giao hoán hữu hạn;
- q, một ước nguyên tố của $\#E$;
- G, một phần tử bậc q trong E.

Trong nhóm E, ký hiệu nhân được sử dụng. Cần lưu ý rằng, với một cơ chế ký cụ thể được lựa cọn có thể cần thêm một số ràng buộc trong việc lựa chọn E, q, G.

5.1.1.2 Tạo chữ ký và kiểm tra chữ ký

Khóa ký X của một chủ thể được sinh ra một cách bí mật ngẫu nhiên hoặc giả ngẫu nhiên sao cho $0 < X < q$. Khóa công khai kiểm tra tương ứng Y là một phần tử của E và được tính như sau:

$$Y = G^{XD}$$

Với D là một tham số được xác định bởi cơ chế được dùng. Giá trị của D là một trong hai giá trị -1 và 1.

CHÚ THÍCH Một cài đặt vẫn được xem là phù hợp nếu loại trừ một vài số nguyên từ việc xem xét các giá trị X. Ví dụ, giá trị 1 có thể bị loại trừ vì giá trị này dẫn đến khóa kiểm tra của người dùng là phần tử sinh G, nó được dễ dàng phát hiện.

5.1.2 Các cơ chế dựa trên định danh

5.1.2.1 Ký hiệu

Hai cơ chế dựa trên định danh được đặc tả trong Điều 7 đều dựa trên việc sử dụng các cặp trên các nhóm đường cong elliptic. Để đặc tả cơ chế dựa trên định danh, ký hiệu nhóm cộng được sử dụng.

5.1.2.2 Sinh các tham số miền

Tập hợp các tham số miền bao gồm các tham số sau:

- E, một nhóm giao hoán hữu hạn;
- $GF(r)$, trường Galois lực lượng r;
- G_1, một nhóm cyclic bậc nguyên tố q
- G_2, một nhóm cyclic bậc nguyên tố q
- P, một phần tử sinh của G_1 ;
- q, là một số nguyên tố, lực lượng của G_1, G_2.
- $<>$, một cặp song tuyến và không suy biến.

5.1.2.3 Sinh khóa chủ

Một khóa chủ riêng của KGC là một khóa ngẫu nhiên bí mật hoặc số giả ngẫu nhiên U sao cho $0 < U < q$. Khóa chủ công khai V là một phần tử của G_1 và được tính như sau : $V = [U]P$.

5.1.2.4 Sinh khóa chữ ký và khóa kiểm tra

Một khóa ký của chủ thể ký là một phần tử của G_1 và được tính bởi KGC như sau $X = [U]Y$.

Với U là khóa riêng của KGC và $Y = H_1(ID)$ là khóa kiểm tra công khai, với ID là một xâu định danh cho KGC và hàm băm H_1.

5.1.3 Lựa chọn tham số

5.1.3.1 Kích thước tham số lựa chọn

Độ dài bit của các tham số cho mức an toàn tiêu biểu được chỉ ra trong Bảng 1. Mức an toàn tối thiểu được đề xuất là 2^{112}.

CHÚ THÍCH 1 Mức an toàn có nghĩa là số lượng bước trong tấn công tốt nhất đã biết lên mật mã nguyên thủy. Nếu 2^{112} bước được yêu cầu trong tấn công tốt nhất lên một hàm băm, thì độ an toàn của hàm băm này là 2^{112}. Để bổ sung các phân tích kích thước tham số, xem tài liệu số [25] và [34].

Không cần thiết phải chọn α, β và γ có độ an toàn như nhau; mức độ an toàn của một lần thực hiện lược đồ chữ ký là nhỏ nhất của mức độ an toàn của các tham số.

Bảng 1 – Kích thước các tham số tuân theo mức an toàn

Mức độ an toàn	2^{80}	2^{112}	2^{128}	2^{192}	2^{256}
α	1024	2048	3072	7680	15360
β	160	224	256	384	512
γ	160	224	256	384	512

Được khuyến nghị mức an toàn là 2^{80} nên sử dụng đối với các ứng dụng kế thừa.

CHÚ THÍCH 2 Không phải mọi cơ chế được đặc tả trong tiêu chuẩn này cung cấp tất cả các mức an toàn được mô tả trong bảng. Với ví dụ là DSA trong 6.1 chỉ cung cấp độ an toàn đến 2^{128}.

5.1.3.2 Lựa chọn một hàm băm

Việc lựa chọn một hàm băm cần dựa vào tiêu chuẩn TCVN 11816-3 (ISO/IEC 10118-3). Nghĩa là h và H_2 sẽ là một trong các cơ chế được đặc tả trong TCVN 11816-3 (ISO/IEC 10118-3), và H_1 biến đổi một xâu bit thu được bởi một trong các cơ chế được đặc tả trong TCVN 11816-3 (ISO/IEC 10118-3) vào một phần tử trong G_1.

Các hàm băm được dùng trong tiêu chuẩn này nên là hàm băm kháng va chạm.

Độ dài an toàn của hàm băm được lựa chọn nên trùng hoặc vượt quá độ dài an toàn của các tham số được dùng trong tạo khóa. Mối quan hệ giữa độ dài an toàn của hàm băm và các tham số tạo khóa được chỉ ra trong 5.1.3.1.

Hơn nữa, việc triển khai kiểm tra chữ ký số phải có cách xác định an toàn mà hàm băm nào được dùng bởi người ký. Nếu không, kẻ tấn công có thể truy vấn người kiểm tra dùng một hàm băm khác yếu hơn, và vì vậy bỏ qua mức an toàn dự định.

5.1.4 Tính đúng đắn của các tham số miền và khóa kiểm tra

Người kiểm tra chữ ký có thể yêu cầu đảm bảo các tham số miền và các khóa công khai kiểm tra là hợp lệ, nếu không sẽ không có đảm bảo việc thỏa mãn các yêu cầu an toàn dự kiến ngay cả khi chữ ký số đã được kiểm tra, và đối phương cũng có thể sinh các chữ ký đã kiểm tra.

Việc đảm bảo tính đúng đắn của các tham số miền có thể được cung cấp theo các cách sau :

- Việc lựa chọn các tham số miền hợp lệ từ một nguồn công khai tin cậy, như là chuẩn ;
- Việc sinh các tham số hợp lệ bởi bên thứ ba tin cậy, như CA hoặc KGC ;
- Tính hợp lệ của lực lượng tham số miền được sinh bởi bên thứ ba tin cậy, như CA hoặc KGC ;
- Đối với người ký, việc tạo các tham số miền hợp lệ bằng cách sử dụng một hệ thống tin cậy ;
- Tính hợp lệ của các tham số miền bởi người dùng (nghĩa là người ký hoặc người kiểm tra).

Việc đảm bảo tính hợp lệ của khóa kiểm tra công khai, có thể được cung cấp bởi một trong các điều sau đây :

- Với người ký, việc tạo ra cặp khóa ký riêng/khóa kiểm tra công khai dùng một hệ thống tin cậy ;
- Với người ký hoặc người kiểm tra, tính hợp lệ của khóa kiểm tra công khai bởi bên thứ ba tin cậy, như CA hoặc KGC ;
- Tính hợp lệ của việc kiểm tra khóa công khai bởi người dùng (hoặc là người ký hoặc là người kiểm tra).

CHÚ THÍCH 1 Tính hợp lệ của các tham số miền và khóa được yêu cầu. Tuy nhiên, cách để đạt được nằm ngoài phạm vi của tiêu chuẩn này.

CHÚ THÍCH 2 Phương pháp xác thực người ký là độc lập trên ứng dụng thực tế, cũng nằm ngoài phạm vi của tiêu chuẩn này.

5.2 Tiến trình ký

5.2.1 Tổng quan

Tất cả các cơ chế chữ ký trong tiêu chuẩn này đều sử dụng một giá trị ngẫu nhiên K, mà được dùng (cùng với thông điệp) để tạo ra bằng chứng R (phần đầu tiên của chữ ký) và một nhiệm vụ (T_1, T_2). Chữ ký cho thông điệp là cặp (R, S) với S (là thành phần thứ hai của chữ ký) được tính bởi các giải pháp của công thức chữ ký.

Trong các cơ chế dựa trên chứng thư số, được đặc tả trong Điều 6, công thức chữ ký là :

$$AK + BX^D + C \equiv 0 (mod\ q),$$

(A, B, C) đã cho là một chuyển vị của (S, T_1, T_2), X là một khóa riêng và D là một tham số phụ thuộc vào cơ chế cụ thể.

Trong các cơ chế dựa trên định danh đặc tả ở Điều 7, công thức chữ ký là :

$$[K]A + [U^D]B + C \equiv 0_E (trong\ G_1)$$

(A, B, C) đã cho là một chuyển vị của (S, T_1, T_2), U là chủ khóa riêng và D là một tham số phụ thuộc vào cơ chế cụ thể.

Việc chuyển vị sẽ được xác định hoặc được thỏa thuận khi cài đặt hệ thống chữ ký số.

Tiến trình tạo chữ ký và định dạng của thông điệp đã ký bao gồm 8 bước (xem Hình 1) :

- Sinh số ngẫu nhiên;
- Tạo ra tiền chữ ký;
- Chuẩn bị thông điệp để ký;
- Tính toán bằng chứng;
- Tính toán nhiệm vụ (không cần thiết tính toán nhiệm vụ trong các cơ chế dựa trên định danh);
- Tính toán phần thứ hai của chữ ký;
- Xây dựng phụ lục;
- Xây dựng thông điệp đã ký.

Trong tiến trình này, chủ thể ký đều sử dụng khóa ký riêng, khóa kiểm tra công khai (lựa chọn) và các tham số miền.

5.2.2 Sinh số ngẫu nhiên

Với mỗi lần ký, chủ thể sinh mới một giá trị ngẫu nhiên bí mật là một số nguyên K với $0 < K < q$. Đầu ra của bước này là K, nó sẽ được giữ bí mật và tiêu hủy an toàn sau khi dùng.

CHÚ THÍCH 1 Số ngẫu nhiên K có thể được xem xét như khóa dùng 1 lần.

CHÚ THÍCH 2 Với lý do hợp lý 5.1.1.2 việc thực thi vẫn được xem xét phù hợp, nếu loại trừ một vài số nguyên từ sự xem xét giá trị K có thể.

5.2.3 Tạo ra tiền chữ ký

Đầu vào của bước này là số ngẫu nhiên K và khóa chữ ký lựa chọn X, với chủ thể ký được tính là tiền chữ ký, Π, nhờ dùng K và tham số công khai làm đầu vào. Trong các cơ chế dựa trên chứng thư số được đặc tả trong Điều 6, nó được tính như sau :

$$\Pi = G^K \text{ thuộc } E.$$

Trong các cơ chế dựa trên định danh được đặc tả trong Điều 7, nó được đặc tả một cách riêng rẽ trong các cơ chế. Đầu ra của bước này là tiền chữ ký Π.

5.2.4 Chuẩn bị thông điệp để ký

Trong các tiến trình chuẩn bị, một trong các thông điệp M_1 và M_2 trở thành thông điệp M, các trường hợp khác là rỗng.

5.2.5 Tính toán các bằng chứng (phần đầu tiên của chữ ký)

Các biến đến bước này là tiền chữ ký Π từ 5.2.3 và M_1 từ 5.2.4. Các giá trị của các biến được lựa chọn như đầu vào của hàm bằng chứng. Đầu ra của hàm bằng chứng là chứng cứ R. Hàm bằng chứng được đặc tả trong các cơ chế.

5.2.6 Tính toán các nhiệm vụ

Các đầu vào hàm nhiệm vụ là phần thứ nhất của chữ ký, với bằng chứng R từ 5.2.5, M_2 từ 5.2.4 và điều kiện khóa kiểm tra Y. Các đầu ra của hàm nhiệm vụ là nhiệm vụ $T = (T_1, T_2)$. Trong các cơ chế dựa trên chứng thư số được đặc tả trong Điều 6, T_1 và T_2 là các số nguyên như sau :

$$0 < |T_1| < q, 0 < |T_2| < q$$

Trong các cơ chế dựa trên định danh được đặc tả trong Điều 7, T_1 và T_2 là các phần tử của G_1. Nó không cần thiết để tính T trong các cơ chế dựa trên định danh.

5.2.7 Tính toán phần thứ hai của chữ ký

Các đầu vào của bước này là số ngẫu nhiên K từ 5.2.1, khóa chữ ký X, nhiệm vụ $T = (T_1, T_2)$ từ 5.2.6 chuyển vị (A, B, C) của (S, T_1, T_2), một biến D trong 5.1.1.2 và tham số miền q như được đặc tả trong 5.1.1.1 và 5.1.2.1.

Trong các cơ chế dựa trên chứng thư số, công thức ký là :

$$AK + BX^D + C \equiv 0 (mod\ q)$$

Và tính công thức chữ ký cho S, các phần thứ hai của chữ ký, với $0 < S < q$.

Các cơ chế dựa trên định danh, các chủ thể ký tính toán công thức chữ ký cho S, phần thứ hai của chữ ký $S \in G_1$. Giải pháp này thỏa mãn công thức chữ ký :

$$[K]A + [U^D]B + C \equiv 0_E (thuộc\ G_1)$$

Cặp (R, S) sẽ được gọi là chữ ký, Σ.

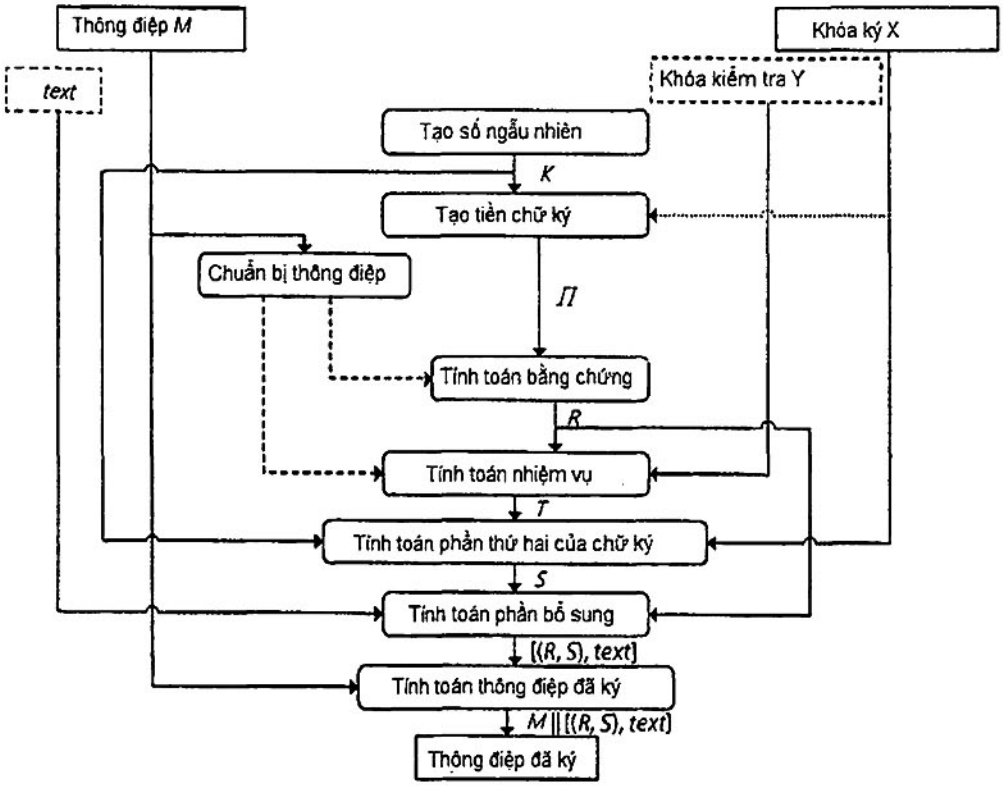
5.2.8. Xây dựng phần phụ lục

Phụ lục được xây dựng từ chữ ký và một trường text tùy chọn, *text*, như là $[(R, S), text]$. Trường text có thể bao gồm một chứng thư số mà mật mã ràng buộc khóa kiểm tra công khai với dữ liệu định danh của chủ thể ký.

Như được chỉ ra trong TCVN 12214-1 (ISO/IEC 14888-1), tùy thuộc vào các ứng dụng, có các phương án khác nhau của việc tạo ra phần phụ lục và để bổ sung cho thông điệp. Yêu cầu chung đối với người kiểm tra có thể có liên quan đến chữ ký số đúng cho thông điệp. Để kiểm tra thành công, cần thiết quá trình tiền kiểm tra, người kiểm tra có thể liên kết khóa kiểm tra đúng với chữ ký.

5.2.9. Xây dựng thông điệp đã ký

Thông điệp đã ký nhận được bằng cách ghép thông điệp M và phần phụ lục, nghĩa là, $M||[(R,S),text]$.



Hình 1 – Tiến trình chữ ký với bằng chứng ngẫu nhiên (một trong số M_1 và M_2 là M, trường hợp còn lại là rỗng)

5.3. Tiến trình kiểm tra

5.3.1 Tổng quan

Quá trình kiểm tra bao gồm 6 bước sau (xem Hình 2) :

- Truy xuất bằng chứng;
- Chuẩn bị thông điệp để kiểm tra;
- Truy xuất các nhiệm vụ (là lựa chọn để tính nhiệm vụ trong các cơ chế dựa trên định danh);
- Tính lại tiền chữ ký;
- Tính lại các bằng chứng;
- Kiểm tra các bằng chứng.

Trong tiến trình này, người kiểm tra dùng khóa kiểm tra của người ký, khóa chủ công khai của KGC (chỉ dành cho các cơ chế dựa trên định danh mô tả ở mục 7) và các tham số miền.

5.3.2 Truy xuất các bằng chứng

Người kiểm tra truy xuất chữ ký số (R,S) từ phần phụ lục, và chia cho bằng chứng R và phần thứ hai của chữ ký S. Cũng vậy, người kiểm tra kiểm tra khoảng và độ dài bit của các phần tử chữ ký, R,S, theo các quy luật được đặc tả bởi mỗi tiến trình ký. Nếu luật định nghĩa trước bị vi phạm, chữ ký sẽ bị từ chối.

5.3.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất M từ thông điệp đã ký và chia thông điệp thành hai phần M_1, M_2.

5.3.4 Truy xuất nhiệm vụ

Bước này giống với 5.2.6. Các đầu vào cho hàm nhiệm vụ bao gồm bằng chứng R từ 5.3.2, M_2 từ Điều 5.3.3, và (được lựa chọn) khóa kiểm tra Y. Nhiệm vụ $T = (T_1, T_2)$ được tính lại như là đầu ra từ hàm nhiệm vụ. Trong các cơ chế dựa trên định danh, không cần thiết phải tính lại T.

5.3.5 Tính lại tiền chữ ký

Các đầu vào cho bước này là tập hợp của các tham số miền, khóa kiểm tra Y, nhiệm vụ $T = (T_1, T_2)$ từ 5.3.4, phần thứ 2 của chữ ký S từ 5.3.2, và lựa chọn R từ 5.3.2. Người kiểm tra chỉ định các hệ số (A, B, C) các giá trị (S, T_1, T_2) theo trật tự được chỉ định bởi hàm chữ ký, và trong các cơ chế dựa trên chứng thư số, tính bởi phần tử Π'.

Trong các cơ chế dựa trên chứng thư số, Π' được tính trong E như sau :

$$\Pi' = Y^m G^n$$

Với $m = -A^{-1} B \bmod q$ và $n = -A^{-1} C \bmod q$.

Trong các cơ chế dựa trên định danh, nó được mô tả cụ thể trong các cơ chế.

5.3.6 Tính lại bằng chứng

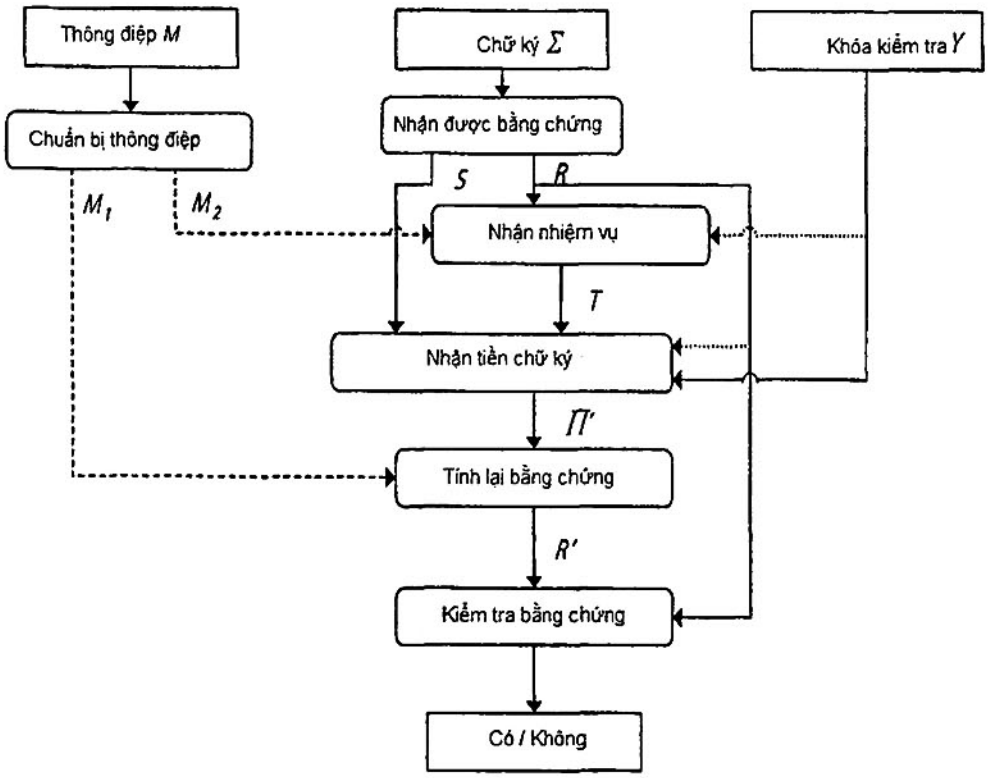
Các tính toán ở bước này như trong 5.2.5. Người kiểm tra thực thi hàm bằng chứng. Các đầu vào Π' từ 5.3.5 và M_1 từ 5.3.3. Các đầu ra được tính lại bằng chứng, R'.

Trong cơ chế IBS-2 được đặc tả trong 7.2, quá trình tính toán lại bằng chứng là tính toán hai hàm kiểm tra thay vì tính R'.

5.3.7 Kiểm tra bằng chứng

Chữ ký số được kiểm tra nếu tính lại bằng chứng R' từ 5.3.6 bằng với R từ 5.3.2.

Trong cơ chế IBS-2 được đặc tả trong 7.2, tiến trình kiểm tra bằng chứng liên quan đến việc kiểm tra xem hai giá trị của hàm kiểm tra được tính trong 5.3.6 có giống hệt nhau không thay vì kiểm tra $R = R'$.



Hình 2 – Tiến trình kiểm tra với bằng chứng ngẫu nhiên

6 Các cơ chế dựa trên chứng thư số

6.1 Tổng quan

Trong số học đường cong elliptic, một điểm của đường cong được biểu diễn như tọa độ affine. Nghĩa là, một điểm Π của đường cong có hai tọa độ: tọa độ x, Π_x, và tọa độ y, Π_y. Các đường cong elliptic cho EC-DSA, EC-KCDSA, EC-GDSA, EC-RDSA, EC-SDSA và EC-FSDSA bị hạn chế với các đường cong không kì dị và không siêu kì dị.

Các hàm băm định danh có thể được dùng để ràng buộc cơ chế chữ ký và hàm băm.

6.2 DSA

6.2.1 Tổng quan

DSA (Thuật toán chữ ký số) là cơ chế chữ ký với $E = Z_p^*$, p là một số nguyên tố, và q là một số nguyên tố là ước của $p - 1$. Tham số D của DSA bằng 1. Thông điệp được chuẩn bị M_1 là rỗng và M_2 là thông điệp được ký, nghĩa là $M_2 = M$. Hàm bằng chứng được xác định theo công thức :

$R = \Pi \bmod q$, *với* $\Pi = G^k \bmod p$ với một vài số nguyên *K* nào đó.

Và hàm nhiệm vụ theo công thức :

$$(T_1, T_2) = (-R, -BS2I(\gamma, H)),$$

Với $H = h(M)$ là mã băm cắt ngắn của thông điệp M, được biến đổi thành một số nguyên theo quy luật biến đổi được đưa trong Phụ lục B.

Các hệ số (A, B, C) của chữ ký số DSA được đặt như sau:

$$(A, B, C) = (S, T_1, T_2)$$

Do vậy, công thức ký trở thành:

$$SK - RX - BS2I(\gamma, H) \equiv 0 (mod\ q)$$

CHÚ THÍCH Cơ chế này được lấy từ viện dẫn [17]. Các ký hiệu được thay đổi một chút so với viện dẫn [17] để phù hợp với các ký hiệu được sử dụng trong một vài nơi của tiêu chuẩn này.

6.2.2 Các tham số

p, số nguyên tố, với $2^{\alpha-1} < p < 2^{\alpha}$.

q, ước nguyên tố của $p - 1$, với $2^{\beta-1} < q < 2^{\beta}$.

G, phần tử của nhóm con bậc q, với $1 < G < p$.

Với bốn lựa chọn cặp (α, β) được cho phép trong DSA, là (1024, 160), (2048, 224), (2048, 256) và (3072, 256). Được khuyến cáo về độ an toàn của cặp (α, β) và γ là như nhau, trừ khi có sự thỏa thuận được thực hiện giữa các bên tham gia để dùng hàm băm mạnh hơn.

Các số nguyên p, q và G có thể được công khai và có thể là chung cho một nhóm người dùng.

Các tham số p, q, G được sinh như đặc tả trong phụ lục D. Nếu phù hợp với tiêu chuẩn của NIST không được yêu cầu thì các tham số p và q có thể được tạo ra nhờ kỹ thuật sinh số nguyên tố trong ISO/IEC 18032.

Một khuyến nghị cho mọi người kiểm tra việc sinh hợp lệ các tham số công khai DSA theo viện dẫn [17].

CHÚ THÍCH Nếu người ký được tự do lựa chọn các tham số miền q để tạo điều kiện cho một va chạm giữa các giá trị băm, một tấn công đối với một trường hợp như vậy của DSA có thể được gắn kết trong các yêu cầu va chạm của hàm băm cơ bản có thể được tìm thấy với độ phức tạp là $2^{74}, 2^{101}, 2^{114}$ (tương ứng với $y = 160, 224, 256$) được đề xuất như là trường hợp an toàn nhất, trong đó độ phức tạp để tìm va chạm sẽ là $2^{80}, 2^{112}, 2^{128}$[39]. Tuy nhiên, tấn công va chạm dễ dàng bị phát hiện. Hơn nữa, tấn công không thể gắn kết khi các tham số miền được sinh ra, như đặc tả trong viện dẫn [17], bao gồm phương pháp được đặc tả trong phụ lục D. Nếu không thể xác minh được việc sử dụng một phương pháp thích hợp để tạo ra các tham số miền thì việc tấn công vẫn có thể được ngăn chặn bằng cách sử dụng các cơ chế mẫu được đặc tả trong các Điều 6.3, 6.4 và 6.7.

Chữ ký số dựa trên SHA-1 được khuyến cáo sử được dùng trong các ứng dụng mang tính kế thừa.

6.2.3 Sinh khóa ký và khóa kiểm tra

Khóa ký của một chủ thể ký là một số nguyên bí mật X được sinh giả ngẫu nhiên hoặc ngẫu nhiên sao cho $0 < X < q$. Tham số D là 1. Tương ứng với khóa kiểm tra công khai Y là $Y = G^X mod\ p$.

Một khóa ký bí mật X của người dùng và khóa kiểm tra công khai Y thường được cố định cho một chu kỳ thời gian. Khóa ký X được giữ bí mật.

6.2.4 Tiến trình trình ký

6.2.4.1 Sinh số ngẫu nhiên

Chủ thể ký tính toán số nguyên K ngẫu nhiên hoặc giả ngẫu nhiên sao cho $0 < K < q$.

6.2.4.2 Tạo tiền chữ ký

Đầu vào của bước này là số ngẫu nhiên K, và chủ thể ký tính :

$$\Pi = G^K mod\ p$$

6.2.4.3 Chuẩn bị thông điệp để ký

Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp được ký, nghĩa là $M_2 = M$.

6.2.4.4 Tính toán bằng chứng

Chủ thể ký tính $R = \Pi mod\ q$, với bằng chứng chỉ đơn giản là một hàm của tiền chữ ký. Vì vậy

$$R = (G^K mod\ p) mod\ q$$

6.2.4.5 Tính toán nhiệm vụ

Chủ thể ký tính toán mã băm; nếu độ dài đầu ra theo bit của hàm băm lựa chọn lớn hơn $\lceil \log_2 q \rceil$, H là tập hợp trái nhất (trọng số cao nhất) $\lceil \log_2 q \rceil$ bit của $h(M_2)$. Ngược lại, H là $h(M_2)$. Sau đó, H được chuyển đổi thành số nguyên theo quy tắc chuyển đổi, *BS2I*, trong phụ lục B. Nhiệm vụ (T_1, T_2) là $(-R, -BS2I(\gamma, H))$.

6.2.4.6 Tính toán thành phần thứ hai của chữ ký

Các chữ ký là (R, S) với R được tính trong 6.2.4.4 và

$$S = \left(K^{-1}(BS2I(\gamma, H) + XR)\right) mod\ q$$

Với H = vị trí trái nhất (trọng số cao nhất) $\min(\beta, \gamma)$ bits của $h(M_2)$.

Giá trị của $h(M_2)$ là một xâu đầu ra γ bit của hàm băm phù hợp trong 6.2.2. Để tính S, chuỗi này sẽ được biến đổi thành số nguyên.

Nó được yêu cầu để kiểm tra $R = 0$ hoặc $S = 0$. Nếu một trong hai giá trị R hoặc $S = 0$, một giá trị mới của K được tạo ra và chữ ký được tính lại (rất hiếm khi xảy ra $R = 0$ hoặc $S = 0$ nếu chữ ký số được sinh đúng).

6.2.4.7 Xây dựng phần phụ lục

Phần phụ lục là phép ghép của (R, S) và một trường lựa chọn text , $text$, $((R, S), text)$.

6.2.4.8 Xây dựng thông điệp đã ký

Một thông điệp đã ký là phép ghép của thông điệp M và phần phụ lục.

$$M||((R, S), text)$$

6.2.5 Tiến trình kiểm tra

6.2.5.1 Tổng quan

Trước khi kiểm tra chữ ký của thông điệp đã ký, người kiểm tra cần phải có các bản sao tin cậy của p, q, G và Y.

6.2.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục. Và kiểm tra xem $0 < R < q$ và $0 < S < q$. Nếu một trong hai điều kiện bị vi phạm, chữ ký sẽ bị từ chối.

6.2.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất $M_2 = M$ từ thông điệp đã ký. M_1 là rỗng.

6.2.5.4 Truy xuất nhiệm vụ

Bước này giống với 6.2.4.5. Các đầu vào cho hàm nhiệm vụ bao gồm bằng chứng R từ 6.2.5.2 và M_2 từ 6.2.5.3. Nhiệm vụ $T = (T_1, T_2)$ được tính lại như là đầu ra của hàm nhiệm vụ 6.2.4.5.

6.2.5.5 Tính lại tiền chữ ký

Các đầu vào cho bước này là các tham số miền, khóa kiểm tra Y, nhiệm vụ $T = (T_1, T_2)$ từ 6.2.5.4 và thành phần thứ 2 của chữ ký số S từ 6.2.5.2. Người kiểm tra có được một giá trị tính lại Π' của tiền chữ ký dùng công thức:

$$\Pi' = Y^{-S^{-1}T1 \bmod q} G^{-S^{-1}T2 \bmod q} \bmod p$$

6.2.5.6 Tính lại bằng chứng

Việc tính toán ở bước này giống như 6.2.4.4. Người kiểm tra thực hiện hàm bằng chứng. Đầu vào là Π' từ 6.2.5.5. Lưu ý rằng M_1 là rỗng. Đầu ra là bằng chứng được tính lại R' sao cho $R' = \Pi' \bmod q$.

6.2.5.7 Kiểm tra bằng chứng

Người kiểm tra so sánh bằng chứng tính lại, R' từ 6.2.5.6 với giá trị của R từ 6.2.5.2. Nếu $R' = R$, thì chữ ký là hợp lệ.

6.3 KCDSA

6.3.1 Tổng quan

KCDSA (Thuật toán chữ ký số dựa trên chứng thư của Hàn Quốc) là cơ chế chữ ký với $E = Z_p^*$, p là số nguyên tố, và q là ước nguyên tố của $p - 1$. Khóa kiểm tra Y là $G^{X^{-1}}$; đó là, tham số D là -1. Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp đã ký, nghĩa là $M_2 = M$. Hàm bằng chứng được xác định theo công thức :

$$R = h(I2BS(\beta, \Pi))$$

Nếu γ là dài hơn β, thì hàm bằng chứng được thay thế bởi công thức :

$$R = I2BS(\beta, BS2I\left(\gamma, h\left(I2BS(\beta, \Pi)\right)\right) \bmod 2^{\beta})$$

Các tham số miền sẽ chỉ định vào hàm băm được dùng. Hàm nhiệm vụ được xác định theo công thức sau :

$$(T_1, T_2) = (V, -1),$$

Với $V = BS2I(\beta, R \oplus H) \bmod q$. Giá trị H là mã băm từ khóa công khai Y và thông điệp M.

Các hệ số (A, B, C) của công thức chữ ký KCDSA như sau :

$$(A, B, C) = (T_2, S, T_1)$$

Vì vậy, công thức chữ ký số trở thành :

$$-K + SX^{-1} + V \equiv 0 (\bmod q)$$

CHÚ THÍCH Cơ chế này được lấy từ mục [36]. Các ký hiệu được thay đổi một chút so với tham chiếu [36] để phù hợp với ký hiệu được dùng trong các phần khác của bộ tiêu chuẩn này.

6.3.2 Các tham số

p một số nguyên tố, với $2^{\alpha-1} < p < 2^{\alpha}$

q một ước nguyên tố của p-1, với $2^{\beta-1} < q < 2^{\beta}$

F một số nguyên sao cho $1 < F < p - 1$ và $F^{(p-1)/q} \bmod p > 1$.

G $F^{(p-1)/q} \bmod p$, một phần tử bậc q trong Z_p^*

l kích thước khối đầu vào (theo bit) của hàm băm được lựa chọn h

Hàm băm của người xác thực hoặc OID với hàm băm được đặc tả.

Ba lựa chọn của bộ ba (α, β, h) được cho phép trong KCDSA, chúng là (2048, 224, SHA-224), (3072, 256, SHA-256) và (2048,224, SHA-256). Giữa chúng, (2048,224,SHA-224) và (3072, 256, SHA-256) được khuyến nghị, còn (2048, 224, SHA-256) có thể được dùng trong trường hợp chỉ có SHA-256 và SHA-224 thì không.

Các số nguyên p, q, G và l có thể được công khai và có thể chung cho một nhóm người dùng.

6.3.3 Sinh khóa ký và khóa kiểm tra

Khóa ký của một chủ thể ký là một số nguyên bí mật X được sinh giả ngẫu nhiên hoặc ngẫu nhiên sao cho $0 < X < q$. Tham số D là -1. Tương ứng với khóa kiểm tra công khai Y là $Y = G^{X^{-1}} \bmod p$.

Một khóa ký bí mật X của người dùng và khóa kiểm tra công khai Y thường được cố định cho một chu kỳ thời gian. Khóa ký X được giữ bí mật.

6.3.4 Tiến trình ký

6.3.4.1 Sinh số ngẫu nhiên

Chủ thể ký tính toán số nguyên K ngẫu nhiên hoặc giả ngẫu nhiên sao cho $0 < K < q$

6.3.4.2 Tạo tiền chữ ký

Đầu vào của bước này là số ngẫu nhiên K, và chủ thể ký tính

$$\Pi = G^K \bmod p$$

6.3.4.3 Chuẩn bị thông điệp để ký

Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp được ký, nghĩa là $M_2 = M$.

6.3.4.4 Tính toán bằng chứng

Chủ thể ký tính bằng chứng $R = h(I2BS(\beta, \Pi))$, trong đó đầu ra của H là mã băm của xâu bit độ dài α được chuyển đổi từ tiền chữ ký Π. Nếu γ dài hơn β, khi đó việc tính toán bằng chứng được thay thế bởi $R = I2BS(\beta, BS2I\left(\gamma, h\left(I2BS(\beta, \Pi)\right)\right) \bmod 2^{\beta})$.

Quy tắc chuyển đổi, *I2BS* và *BS2I* được đưa ra trong phụ lục B.

6.3.4.5 Tính toán nhiệm vụ

Chủ thể ký tính toán nhiệm vụ $(T_1, T_2) = (V, -1)$ với $V = BS2I(\beta, R \oplus H) \bmod q$, với $H = h(Y' || M_2)$ là mã băm của phép ghép $Y' = I2BS(l, Y \bmod 2^l)$ và thông điệp M_2. Giá trị của Y' là một xâu bit có độ dài l. Trong tính toán V, xâu bit $R \oplus H$ sẽ được chuyển đổi thành số nguyên trước khi giảm modulo đối với q.

Nếu γ dài hơn β, thì việc tính toán H được thay thế bởi $H = I2BS(\beta, BS2I(\gamma, h\,(Y' \,||\, M_2)) \bmod 2^{\beta})$.

CHÚ THÍCH Y' là một giá trị cố định cho người dùng, giá trị này có thể được giữ như là tham số của người dùng.

6.3.4.6 Tính toán thành phần thứ hai của chữ ký

Chữ ký là (R, S) với R được tính trong 6.3.4.4 và

$$S = X(K - V) \bmod q$$

6.3.4.7 Xây dựng phần phụ lục

Phần phụ lục là phép ghép của (R, S) và một trường lựa chọn text , $text$, $((R, S), text)$.

6.3.4.8 Xây dựng thông điệp đã ký

Một thông điệp đã ký là phép ghép của thông điệp M và phần phụ lục.

$$M||((R,S),text)$$

6.3.5 Tiến trình kiểm tra

6.3.5.1 Tổng quan

Trước khi kiểm tra chữ ký của thông điệp đã ký, người kiểm tra cần phải có các bản sao tin cậy của p, q và G

Người kiểm tra cũng yêu cầu các dữ liệu cần thiết cho tiến trình kiểm tra: ví dụ, khóa kiểm tra Y (xem TCVN 12214-1 (ISO/IEC 14888-1 :2008), Điều 9 cho các yêu cầu bổ sung).

6.3.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục. Và kiểm tra xem các điều kiện sau có thỏa mãn hay không:

- $0 < S < q$;
- Nếu độ dài của giá trị γ không dài hơn β, độ dài bit của R bằng độ dài bit đầu ra của hàm băm được dùng h;
- Nếu độ dài của giá trị γ là dài hơn β, độ dài bit của R là bằng β.

Nếu vi phạm bất kỳ điều kiện nào ở trên chữ ký số sẽ bị từ chối.

6.3.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất $M_2 = M$ từ thông điệp đã ký. M_1 là rỗng.

6.3.5.4 Truy xuất nhiệm vụ

Bước này giống với 6.3.4.5. Các đầu vào cho hàm nhiệm vụ bao gồm bằng chứng R từ 6.3.5.2 và M_2 từ 6.3.5.3. Nhiệm vụ $T = (T_1, T_2)$ được tính lại như là đầu ra của hàm nhiệm vụ 6.3.4.5.

6.3.5.5 Tính lại tiền chữ ký

Đầu vào của bước này là các tham số miền, khóa kiểm tra Y, nhiệm vụ $T = (T_1, T_2)$ từ 6.3.5.4 và phần thứ hai của chữ ký số S từ 6.3.5.2. Người kiểm tra nhận được giá trị tính toán lại Π' của tiền chữ ký nhờ công thức :

$$\Pi' = Y^{S \bmod q} G^{T1 \bmod q} mod\ p$$

6.3.5.6 Tính lại bằng chứng

Việc tính toán ở bước này giống như 6.3.4.4. Người kiểm tra thực hiện hàm bằng chứng. Đầu vào là Π' từ 6.3.5.5. Lưu ý rằng M_1 là rỗng. Đầu ra là bằng chứng được tính lại R'

6.3.5.7 Kiểm tra bằng chứng

Người kiểm tra so sánh bằng chứng tính lại, R' từ 6.3.5.6 với giá trị của R từ 6.3.5.2. Nếu $R' = R$, thì chữ ký là hợp lệ.

6.4 Thuật toán Pointcheval/Vaudenay

6.4.1 Tổng quan

Phương pháp của Pointcheval/Vaudenay là một biến thể của thuật toán DSA, với $E = Z_p^*$, p là một số nguyên tố, và q là một ước nguyên tố của p – 1. Tham số D bằng 1. Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp được ký, nghĩa là $M_2 = M$. Bằng chứng được xác định bởi công thức :

$$R = \Pi\ mod\ q$$

Và hàm nhiệm vụ được xác định bởi công thức

$$(T_1, T_2) = (-R, -H)$$

Với $H = h(I2BS(\beta, R)||M)$ là mã băm của phép ghép của bằng chứng R và thông điệp M. Lưu ý rằng việc tính T_2 trên yêu cầu việc biến đổi mã băm thành một số nguyên. Hàm biến đổi được đưa ra trong phụ lục B.

Các hệ số (*A*, *B*, *C*) của công thức chữ ký Pointcheval/Vaudenay như sau :

$$(A, B, C) = (S, T_1, T_2)$$

Vì vậy công thức ký là :

$$SK - RX - H \equiv 0(mod\ q)$$

CHÚ THÍCH Cơ chế này dựa trên thuật toán được thiết kế bởi *D*. Pointcheval và *S*. Vaudenay trong viện chiều [31].

6.4.2 Các tham số

p một số nguyên tố

q một ước nguyên tố của $p - 1$

F một số nguyên sao cho $1 < F < p - 1$ và $F^{(p-1)/q} mod\ p > 1$

G $F^{(p-1)/q} mod\ p$

Hàm băm định danh hoặc OID với hàm băm được đặc tả tả

Một khuyến nghị cho mọi người kiểm tra việc sinh các tham số công khai hợp lệ.

6.4.3 Sinh khóa ký và khóa kiểm tra

Khóa ký của một chủ thể ký là một số nguyên bí mật X được sinh giả ngẫu nhiên hoặc ngẫu nhiên sao cho $0 < X < q$. Tham số D là 1. Khóa kiểm tra công khai tương ứng Y là :

$$Y = G^X mod\ p$$

Một khóa ký bí mật X của người dùng và khóa kiểm tra công khai Y thường được cố định cho một chu kỳ thời gian. Khóa ký X được giữ bí mật.

6.4.4 Tiến trình ký

6.4.4.1 Sinh số ngẫu nhiên

Chủ thể ký tính toán số nguyên K ngẫu nhiên hoặc giả ngẫu nhiên sao cho $0 < K < q$

6.4.4.2 Tạo tiền chữ ký

Đầu vào của bước này là số ngẫu nhiên K và chủ thể ký tính:

$$\Pi = G^K mod\ p$$

6.4.4.3 Chuẩn bị thông điệp để ký

Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp được ký, nghĩa là $M_2 = M$.

6.4.4.4 Tính toán bằng chứng

Chủ thể ký tính toán $R = \Pi mod\ q$, với bằng chứng đơn giản là một hàm của tiền chữ ký. Vì vậy,

$R = (G^K mod\ p) mod\ q$.

6.4.4.5 Tính toán nhiệm vụ

Chủ thể ký tính nhiệm vụ $(T_1, T_2) = (-R, -BS2I(\gamma, H))$, với $H = h(I2BS(\beta, R)||M_2)$ là mã băm của việc ghép bằng chứng và thông điệp M_2. Trước khi ghép, bằng chứng sẽ được biến đổi thành xâu bit độ dài $|p|$.

6.4.4.6 Tính chữ ký

Chữ ký là (R, S) với R được tính ở 6.4.4.4 và

$$S = K^{-1}(BS2I(\gamma, H) + XR) mod\ q.$$

6.4.4.7 Xây dựng phần phụ lục

Phần phụ lục là phép ghép của (R, S) và một trường lựa chọn text , $text$, $((R, S), text)$.

6.4.4.8 Xây dựng thông điệp đã ký

Một thông điệp đã ký là phép ghép của thông điệp M và phần phụ lục.

$$M||((R, S), text)$$

6.4.5 Tiến trình kiểm tra

6.4.5.1 Tổng quan

Trước khi kiểm tra chữ ký của thông điệp đã ký, người kiểm tra cần phải có các bản sao tin cậy của p, q, G.

Người kiểm tra cũng yêu cầu các dữ liệu cần thiết cho tiến trình kiểm tra: ví dụ, khóa kiểm tra Y (xem TCVN 12214-1 (ISO/IEC 14888-1:2008), Điều 9 cho các yêu cầu bổ sung).

6.4.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục. Và kiểm tra xem $0 < R < q$ và $0 < S < q$. Nếu một trong các điều kiện bị vi phạm, chữ ký số sẽ bị từ chối.

6.4.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất $M_2 = M$ từ thông điệp đã ký. M_1 là rỗng.

6.4.5.4 Truy xuất nhiệm vụ

Bước này giống với 6.4.4.5. Các đầu vào cho hàm nhiệm vụ bao gồm bằng chứng R từ 6.4.5.2 và M_2 từ 6.4.5.3. Nhiệm vụ $T = (T_1, T_2)$ được tính lại như là đầu ra của hàm nhiệm vụ 6.4.4.5.

6.4.5.5 Tính lại tiền chữ ký

Các đầu vào cho bước này là các tham số miền, khóa kiểm tra Y, nhiệm vụ $T = (T_1, T_2)$ từ 6.4.5.4 và thành phần thứ 2 của chữ ký số S từ 6.4.5.2. Người kiểm tra có được một giá trị tính lại Π' của tiền chữ ký dùng công thức:

$$\Pi' = Y^{-S-1T1\ mod\ q} G^{-S-1T2\ mod\ q}\ mod\ p$$

6.4.5.6 Tính lại bằng chứng

Việc tính toán ở bước này giống như 6.4.4.4. Người kiểm tra thực hiện hàm bằng chứng. Đầu vào là Π' từ 6.4.5.5. Lưu ý rằng M_1 là rỗng. Đầu ra là bằng chứng được tính lại R'

6.4.5.7 Kiểm tra bằng chứng

Người kiểm tra so sánh bằng chứng tính lại, R' từ 6.4.5.6 với giá trị của R từ 6.4.5.2. Nếu $R' = R$, thì chữ ký là hợp lệ

6.5. SDSA

6.5.1 Tổng quan

SDSA (Thuật toán chữ ký số Schnorr) là một cơ chế chữ ký với $E = Z_p^*$, p là một số nguyên tố, và q là một ước nguyên tố của $p - 1$. Tham số $D = 1$. Thông điệp chuẩn bị sao cho M_1 là thông điệp được ký, nghĩa là $M_1 = M$, và M_2 là rỗng. Hàm bằng chứng được xác định bằng cách đặt R bằng một mã băm. Hàm nhiệm vụ được xác định bằng cách đặt $T_1 = -1$ và T_2 là số nguyên âm nhận được bằng cách chuyển đổi R thành một số nguyên theo quy tắc $BS2I$, được đưa ra trong Phụ lục B và sau đó rút gọn modulo q.

Các hệ số (*A, B, C*) của công thức ký SDSA được tính như sau:

$$(A, B, C) = (T_1, T_2, S).$$

Vì vậy, công thức ký trở thành:

$$-K + T_2 X + S \equiv 0 \ (mod\ q).$$

CHÚ THÍCH SDSA là viết tắt của Thuật toán chữ ký số Schnorr. Cơ chế được lấy từ viện dẫn [32] và [33]. Các ký hiệu được thay đổi so với viện dẫn [32] và [33] để phù hợp với ký hiệu được dùng trong tiêu chuẩn này.

6.5.2 Các tham số

p số nguyên tố, với $2^{\alpha-1} < p < 2^{\alpha}$.

q một ước số nguyên tố của $p - 1$, với $2^{\beta-1} < q < 2^{\beta}$.

G một phần tử sinh của nhóm con bậc q, sao cho $1 < G < q$.

Bốn sự lựa chọn cho cặp (α, h) được cho phép trong SDSA, được gọi là (1024, SHA-1), (2048, SHA-224) và (3072, SHA-256). β tương ứng được lựa chọn là α trong Bảng 1.

Các số nguyên p, q, G có thể được công khai và có thể chung cho một nhóm người dùng.

Các tham số p, q, G được sinh như trong đặc tả ở phụ lục D. Các tham số p và q có thể được sinh ra bằng cách sử dụng kỹ thuật sinh số nguyên tố trong ISO/IEC 18032.

Một khuyến nghị đối với toàn bộ người dùng kiểm tra việc sinh hợp lệ của các tham số công khai SDSA theo viện dẫn [17].

Một khuyến nghị đối với chữ ký số dựa trên SHA-1 chỉ nên sử dụng cho các ứng dụng mang tính kế thừa.

6.5.3 Sinh khóa ký và khóa kiểm tra

Khóa ký của một chủ thể ký là một số nguyên bí mật X được sinh giả ngẫu nhiên hoặc ngẫu nhiên sao cho $0 < X < q$. Tham số D là 1. Tương ứng với khóa kiểm tra công khai Y là $Y = G^X mod\ p$.

Một khóa ký bí mật X của người dùng và khóa kiểm tra công khai Y thường được cố định cho một chu kỳ thời gian. Khóa ký X được giữ bí mật

6.5.4 Tiến trình ký

6.5.4.1 Tạo số ngẫu nhiên

Chủ thể ký tính toán số nguyên K ngẫu nhiên hoặc giả ngẫu nhiên sao cho $0 < K < q$

6.5.4.2 Tạo tiền chữ ký

Đầu vào của bước này là số ngẫu nhiên K, và chủ thể ký tính

$$\Pi = G^K mod\ p.$$

6.5.4.3 Chuẩn bị thông điệp để ký

Thông điệp được chuẩn bị sao cho M_1 là thông điệp được ký, nghĩa là $M_1 = M$, và M_2 là rỗng.

6.5.4.4 Tính bằng chứng

Chủ thể ký tính bằng chứng R là mã băm của tiền chữ ký Π và phần đầu tiên của thông điệp M_1

$$R = h(I2BS(\beta, \Pi)||M).$$

6.5.4.5 Tính nhiệm vụ

Giá trị của bằng chứng R được biến đổi thành một số nguyên theo quy tắc biến đổi, BS2I, trong phụ lục B và sau đó rút gọn theo modulo q. Nhiệm vụ $(T1, T2)$ là $(-1, -BS2I(\gamma, R) mod\ q)$.

6.5.4.6 Tính toán thành phần thứ hai của chữ ký

Chữ ký số (R, S) với $S = (K + BS2I(\gamma, R)X) mod\ q$.

Như một sự lựa chọn, một mong muốn để kiểm tra nếu $R = 0$ hoặc $S = 0$. Nếu một trong hai giá trị $R = 0$ hoặc $S = 0$, một giá trị mới của K được sinh ra và chữ ký được tính toán lại (rất hiếm khi xảy ra $R = 0$ hoặc $S = 0$ nếu chữ ký số được sinh đúng).

6.5.4.7 Xây dựng phần phụ lục

Phần phụ lục là phép ghép của (R, S) và một trường lựa chọn text, *text*.

6.5.4.8 Xây dựng thông điệp đã ký

Một thông điệp đã ký là phép ghép của thông điệp M và phần phụ lục.

$$M||((R, S), text)$$

6.5.5 Tiến trình kiểm tra

6.5.5.1 Tổng quan

Chủ thể kiểm tra yêu cầu dữ liệu cần thiết được yêu cầu cho tiến trình kiểm tra.

6.5.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục. Người kiểm tra kiểm tra xem R là một xâu khác 0 trong dải của hàm băm hay không và sao cho $0 < S < q$.

6.5.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất M từ thông điệp đã ký và chia thông điệp thành hai phần M_1 và M_2, sao cho $M_1 = M$ và M_2 là rỗng.

6.5.5.4 Truy xuất nhiệm vụ

Đầu vào của hàm nhiệm vụ bao gồm bằng chứng R từ 6.5.5.2. Nhiệm vụ

$$T = (T_1, T_2) = (-1, -BS2I(\gamma, R) mod\ q).$$

6.5.5.5 Tính lại tiền chữ ký

Các đầu vào cho bước này là các tham số miền, khóa kiểm tra Y, nhiệm vụ $T = (T_1, T_2)$ từ 6.5.5.4 và thành phần thứ 2 của chữ ký số S từ 6.5.5.2. Người kiểm tra có được một giá trị tính lại Π' của tiền chữ ký dùng công thức

$$\Pi' = Y^{(T2\ mod\ q)} G^{(-ST1\ mod\ q)} mod\ p = Y^{(T2\ mod\ q)} G^{(S\ mod\ q)} mod\ p$$

6.5.5.6 Tính lại bằng chứng

Việc tính ở bước này là giống như trong 6.5.4.4 và 6.5.4.5. Người kiểm tra thực hiện hàm bằng chứng. Đầu vào Π' từ 6.5.5.5 và M_1 từ 6.5.5.3. Đầu ra là bằng chứng được tính lại R', là mã băm của việc tính lại tiền chữ ký Π' và thông điệp M.

6.5.5.7 Kiểm tra bằng chứng

Người kiểm tra so sánh với bằng chứng được tính lại R' từ 6.5.5.6 với phiên bản R truy xuất từ 6.5.5.2. Nếu $R' = R$ thì chữ ký được kiểm tra.

6.6 EC-DSA

6.6.1 Tổng quan

EC-DSA (Thuật toán chữ ký số dựa trên đường cong elliptic) là một thuật toán đường cong elliptic liên tục DSA. Các hệ số (A, B, C) của EC-DSA được thiết lập như sau:

$$(A, B, C) = (S, T_1, T_2)$$

Với $(T_1, T_2) = \left(-R, -BS2I(\gamma, H)\right)$ và $H = h(M)$ là mã băm được cắt ngắn của thông điệp M, được biến đổi thành số nguyên theo quy tắc biến đổi được đưa trong phụ lục B. Hàm băm h là một trong các hàm băm SHA-1, SHA-224, SHA-256, SHA-384 và SHA-512 được mô tả trong TCVN 11816-3:2017 (ISO/IEC 10118-3).

Khóa kiểm tra Y là $[X]G$; nghĩa là tham số D bằng 1. Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp được ký, nghĩa là $M_2 = M$. Hàm bằng chứng được xác định theo công thức

$$R = FE2I(r, \Pi_X) mod\ q$$

Quy tắc biến đổi, *FE2I*, được đưa ra trong Phụ lục B.

Vì vậy, công thức chữ ký trở thành

$$SK - RX - BS2I(\gamma, H) \equiv 0 (mod\ q)$$

CHÚ THÍCH Cơ chế này dựa trên thuật toán được mô tả tại viện dẫn [7].

6.6.2 Các tham số

F Một trường hữu hạn

E một nhóm đường cong elliptic trên trường F.

$\#E$ lực lượng của E

q một ước nguyên tố của $\#E$

G một điểm của đường cong elliptic bậc q

Mọi tham số có thể được công khai và có thể dùng chung cho một nhóm người dùng. Độ dài an toàn của hàm băm được lựa chọn nên trùng hoặc vượt quá độ dài an toàn liên quan đến độ dài bit q.

Một khuyến nghị đối với toàn bộ người dùng kiểm tra việc sinh hợp lệ của các tham số công khai EC-DSA theo viện dẫn [7] hoặc [17].

Một khuyến nghị đối với chữ ký số dựa trên SHA-1 chỉ nên sử dụng cho các ứng dụng mang tính kế thừa.

6.6.3 Sinh khóa ký và khóa kiểm tra

Khóa ký của một chủ thể ký là một số nguyên bí mật X được sinh giả ngẫu nhiên hoặc ngẫu nhiên sao cho $0 < X < q$. Tham số D là 1. Tương ứng với khóa kiểm tra công khai Y là

$$Y = [X]G.$$

Một khóa ký bí mật X của người dùng và khóa kiểm tra công khai Y thường được cố định cho một chu kỳ thời gian. Khóa ký X được giữ bí mật

6.6.4 Tiến trình ký

6.6.4.1 Tạo số ngẫu nhiên

Chủ thể ký tính toán một số nguyên K ngẫu nhiên hoặc giả ngẫu nhiên sao cho $0 < K < q$.

6.6.4.2 Tạo tiền chữ ký

Đầu vào của bước này là số ngẫu nhiên K và chủ thể ký tính

$$\Pi = [K]G.$$

6.6.4.3 Chuẩn bị thông điệp để ký

Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp được ký, nghĩa là $M_2 = M$.

6.6.4.4 Tính bằng chứng

Chủ thể ký tính $R = FE2I(r, \Pi_X) mod\ q$.

6.6.4.5 Tính nhiệm vụ

Chủ thể ký tạo ra mã băm; nếu độ dài theo bit đầu ra của hàm băm lớn hơn $\lceil log_2 q \rceil$, H được thiết lập phía trái nhất (trọng số cao nhất) $\lceil log_2 q \rceil$ của $h(M_2)$. Ngược lại, H là $h(M_2)$. Sau đó, H được biến đổi thành số nguyên theo quy tắc biến đổi $BS2I$, trong phụ lục B. Nhiệm vụ (T_1, T_2) là $(-R, -BS2I(\gamma, H))$.

6.6.4.6 Tính thành phần thứ hai của chữ ký

Chữ ký là (R, S) với R được tính trong 6.6.4.4 và

$$S = \left(K^{-1}(XR + BS2I(\gamma, H))\right) mod\ q.$$

Nó được yêu cầu để kiểm tra $R = 0$ hoặc $S = 0$. Nếu một trong hai giá trị R hoặc $S = 0$, một giá trị mới của K được tạo ra và chữ ký được tính lại (rất hiếm khi xảy ra $R = 0$ hoặc $S = 0$ nếu chữ ký số được sinh đúng).

6.6.4.7 Xây dựng phần phụ lục

Phần phụ lục là phép ghép của (R, S) và một trường lựa chọn text , $text$, $((R, S), text)$

6.6.4.8. Xây dựng thông điệp đã ký

Một thông điệp đã ký là việc ghép của M và phần phụ lục.

$$M||((R, S), text)$$

6.6.5 Tiến trình kiểm tra

6.6.5.1 Tổng quan

Chủ thể kiểm tra yêu cầu dữ liệu cần thiết được yêu cầu cho tiến trình kiểm tra.

6.6.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục. Để kiểm tra xem $0 < R < q$ và $0 < S < q$; nếu một trong các điều kiện đó bị vi phạm, chữ ký số sẽ bị từ chối.

6.6.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất M từ thông điệp đã ký và chia thông điệp thành hai phần M_1 và M_2. M_1 là rỗng và $M_2 = M$.

6.6.5.4 Truy xuất nhiệm vụ

Bước này giống với 6.6.4.5. Các đầu của hàm nhiệm vụ bao gồm bằng chứng R từ 6.6.5.2 và M_2 từ 6.6.5.3. Nhiệm vụ $T = (T_1, T_2)$ được tính lại như đầu ra từ hàm nhiệm vụ 6.6.4.5.

6.6.5.5 Tính lại tiền chữ ký

Đầu vào của bước này là hệ thống các tham số, khóa kiểm tra Y, nhiệm vụ $(T = T_1, T_2)$ từ 6.6.5.4 và phần thứ hai của chữ ký S từ 6.6.5.2. Người kiểm tra có được giá trị tính lại của tiền chữ ký Π' bởi công thức sau

$$\Pi' = [-S^{-1}T_1 mod\ q]Y + [-S^{-1}T_2 mod\ q]G$$

6.6.5.6 Tính lại bằng chứng

Việc tính toán ở bước này giống với 6.6.4.4. Người kiểm tra thực hiện hàm bằng chứng. Đầu vào Π' từ 6.6.5.5. Đầu ra là bằng chứng được tính lại R'.

6.6.5.7 Kiểm tra bằng chứng

Người kiểm tra so sánh giá trị bằng chứng được tính lại R' từ 6.6.5.6 với phiên bản R được truy xuất từ 6.6.5.2. Nếu $R' = R$ thì chữ ký số được kiểm tra.

6.7 EC-KCDSA

6.7.1 Tổng quan

EC-KCDSA (Thuật toán chữ ký số dựa trên chứng thư số đường cong elliptic Hàn Quốc) là cơ chế ký với khóa kiểm tra $Y = [X^{-1}]G$; đó là, tham số D bằng -1. Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp đã được ký, nghĩa là $M_2 = M$. Hàm bằng chứng được xác định theo công thức sau:

$$R = h(FE2BS(r, \Pi_X)).$$

Nếu γ dài hơn β, thì hàm bằng chứng được thay thế bởi công thức

$$R = I2BS(\beta, BS2I\left(\gamma, h\left(FE2BS(r, \Pi_X)\right)\right) \bmod 2^{\beta})$$

Các tham số miền sẽ được chỉ thị trong hàm băm được dùng. Các hàm nhiệm vụ được xác định theo công thức :

$$(T_1, T_2) = (V, -1)$$

Với $V = BS2I(\beta, (R \oplus H)) mod\ q$. Giá trị H là mã băm từ khóa công khai Y và thông điệp M.

Các hệ số (A, B, C) của EC-KCDSA được thiết lập như sau :

$$(A, B, C) = (T_2, S, T_1)$$

Vì vậy công thức ký trở thành

$$-K + SX^{-1} + V \equiv 0 (mod\ q).$$

CHÚ THÍCH Cơ chế này được lấy từ viện dẫn [37]. Các ký hiệu được thay đổi một chút so với viễn dẫn [37] để phù hợp với ký hiệu được dùng trong một số nơi khác của tiêu chuẩn này.

6.7.2 Các tham số

l kích thước khối đầu vào (theo bit) của hàm băm đã được lựa chọn h.

F một trường hữu hạn

E một nhóm đường cong elliptic trên trường F

$\#E$ lực lượng của E

q ước nguyên tố của $\#E$

G một điểm trên đường cong elliptic bậc q

Định danh hàm băm hoặc OID với hàm băm được đặc tả.

Tất cả các tham số có thể được công khai và có thể dùng chung cho một nhóm người dùng.

Một khuyến nghị đối với toàn bộ người dùng kiểm tra việc sinh hợp lệ của các tham số công khai EC-KCDSA theo viện dẫn [37].

6.7.3 Sinh khóa ký và khóa kiểm tra

Khóa ký của một chủ thể ký là một số nguyên bí mật X được sinh giả ngẫu nhiên hoặc ngẫu nhiên sao cho $0 < X < q$. Tham số D là -1. Tương ứng với khóa kiểm tra công khai Y là

$$Y = [X^{-1}]G$$

Một khóa ký bí mật X của người dùng và khóa kiểm tra công khai Y thường được cố định cho một chu kỳ thời gian. Khóa ký X được giữ bí mật.

6.7.4 Tiến trình ký

6.7.4.1 Tạo số ngẫu nhiên

Chủ thể ký tính toán một số nguyên K ngẫu nhiên hoặc giả ngẫu nhiên sao cho $0 < K < q$.

6.7.4.2 Tạo tiền chữ ký

Đầu vào cho bước này là số ngẫu nhiên K và chủ thể ký tính

$$\Pi = [K]G$$

6.7.4.3 Chuẩn bị thông điệp để ký

Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp được ký, nghĩa là $M_2 = M$.

6.7.4.4 Tính bằng chứng

Chủ thể ký tính $R = h(FE2BS(r, \Pi_X))$, với đầu ra của h là mã băm của Π_X. Nếu γ dài hơn β, thì việc tính toán bằng chứng được thay bằng $R = I2BS(\beta, BS2I(8\lceil log_{256}(r)\rceil, FE2BS(r, \Pi_X)) \bmod 2^{\beta})$.

Quy tắc biến đổi *FE2BS*, *I2BS* và *BS2I* được đưa ra trong phụ lục B.

6.7.4.5 Tính nhiệm vụ

Chủ thể ký tính nhiệm vụ $(T_1, T_2) = (V, -1)$ với $V = BS2I(\beta, R \oplus H) \bmod q$, với $H = h(Y'||M_2)$ là mã băm của phép ghép Y' và thông điệp M_2. Giá trị của Y' là l bit trái nhất của một chuỗi bit $FE2BS(r, Y_X)||FE2BS(r, Y_Y)$. Nếu l dài hơn độ dài của chuỗi, số 0 được đệm thêm vừa đủ vào cuối chuỗi. Trong việc tính V, xâu bit $R \oplus H$ sẽ được biến đổi thành một số nguyên trước khi rút gọn modulo đối với q.

Nếu γ dài hơn β, thì việc tính H sẽ được thay thế bằng $H = I2BS(\beta, BS2I(\gamma, h(Y'||M_2)) \bmod 2^{\beta})$.

CHÚ THÍCH Y' là một giá trị cố định cho một người dùng, giá trị này có thể được giữ như là một tham số người dùng.

6.7.4.6 Tính toán thành phần thứ hai của chữ ký

Chữ ký (R, S) với R được tính trong 6.7.4.4 và

$$S = X(K - V) mod\ q$$

6.7.4.7 Xây dựng phần phụ lục

Phần phụ lục là phép ghép của (R, S) và một trường lựa chọn text , $text$, $((R, S), text)$.

6.7.4.8 Xây dựng thông điệp đã ký

Một thông điệp đã ký là phép ghép thông điệp M và phần phụ lục.

$$M||((R, S), text)$$

6.7.5 Tiến trình kiểm tra

6.7.5.1 Tổng quan

Trước khi kiểm tra chữ ký của thông điệp đã ký, người kiểm tra cần phải có các bản sao tin cậy của E, q và G

Người kiểm tra cũng yêu cầu các dữ liệu cần thiết cho tiến trình kiểm tra: ví dụ, khóa kiểm tra Y (xem TCVN 12214-1 (ISO/IEC 14888-1), Điều 9 cho các yêu cầu bổ sung).

6.7.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục. Người kiểm tra kiểm tra xem các điều sau có thỏa mãn hay không:

- $0 < S < q$;
- Nếu độ dài của giá trị γ không dài hơn β, độ dài theo bit của $R = h$ (độ dài đầu ra theo bit của hàm băm được sử dụng);
- Nếu độ dài của giá trị γ dài hơn β, độ dài theo bit của $R = \beta$.

Nếu bất kỳ điều nào trong các điều trên không thỏa mãn, chữ ký số sẽ bị từ chối.

6.7.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất $M_2 = M$ từ thông điệp đã ký. M_1 là rỗng.

6.7.5.4 Truy xuất nhiệm vụ

Bước này giống với 6.7.4.5. Các đầu vào cho hàm nhiệm vụ bao gồm bằng chứng R từ 6.7.5.2 và M_2 từ 6.7.5.3. Nhiệm vụ $T = (T_1, T_2)$ được tính lại như là đầu ra của hàm nhiệm vụ 6.7.4.5

6.7.5.5 Tính lại tiền chữ ký

Các đầu vào cho bước này là các tham số miền, khóa kiểm tra Y, nhiệm vụ $T = (T_1, T_2)$ từ 6.7.5.4 và thành phần thứ 2 của chữ ký số S từ 6.7.5.2. Người kiểm tra có được một giá trị tính lại Π' của tiền chữ ký dùng công thức:

$$\Pi' = [S\ mod\ q]Y + [T_1\ mod\ q]G.$$

6.7.5.6 Tính lại bằng chứng

Việc tính toán ở bước này giống như 6.7.4.4. Người kiểm tra thực hiện hàm bằng chứng. Đầu vào là Π' từ 6.7.5.5. Lưu ý rằng M_1 là rỗng. Đầu ra là bằng chứng được tính lại R'

6.7.5.7 Kiểm tra bằng chứng

Người kiểm tra so sánh bằng chứng tính lại, R' từ 6.7.5.6 với giá trị của R từ 6.7.5.2. Nếu $R' = R$, thì chữ ký là hợp lệ

6.8 EC-GDSA

6.8.1 Tổng quan

EC-GDSA (Thuật toán chữ ký số đường cong elliptic Đức) là cơ chế ký với khóa kiểm tra $Y = [X^{-1}]G$; đó là tham số D bằng -1. Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp đã được ký, nghĩa là $M_2 = M$. Hàm bằng chứng được xác định theo công thức sau

$$R = FE2I(r, \Pi_X) mod\ q.$$

Các hệ số (A, B, C) của công thức chữ ký số EC-GDSA được thiết lập như sau:

$$(A, B, C) = (T_1, S, T_2)$$

Với $(T_1, T_2) = (-R, H)$ và H là mã băm của thông điệp M.

Công thức chữ ký trở thành:

$$-RK + SX^{-1} + H \equiv 0\ (mod\ q).$$

CHÚ THÍCH EC-GDSA là viết tắt của thuật toán chữ ký số của đường cong elliptic Đức.[16]

6.8.2 Các tham số

F một trường hữu hạn

E một nhóm đường cong elliptic trên trường F

$\#E$ lực lượng của E

q ước nguyên tố của $\#E$

G một điểm trên đường cong elliptic bậc q

Định danh hàm băm hoặc OID với hàm băm được đặc tả.

Tất cả các tham số có thể được công khai và có thể dùng chung cho một nhóm người dùng.

Một khuyến nghị đối với toàn bộ người dùng kiểm tra việc sinh hợp lệ các tham số công khai.

6.8.3. Sinh khóa ký và khóa kiểm tra

Khóa ký của một chủ thể ký là một số nguyên bí mật X được sinh giả ngẫu nhiên hoặc ngẫu nhiên sao cho $0 < X < q$. Tham số D là -1. Tương ứng với khóa kiểm tra công khai Y là

$$Y = [X^{-1}]G$$

Một khóa ký bí mật X của người dùng và khóa kiểm tra công khai Y thường được cố định cho một chu kỳ thời gian. Khóa ký X được giữ bí mật

6.8.4 Tiến trình ký

6.8.4.1 Tạo số ngẫu nhiên

Chủ thể ký tính một số nguyên ngẫu nhiên hoặc giả ngẫu nhiên K sao cho $0 < K < q$.

6.8.4.2 Tạo tiền chữ ký

Đầu vào của bước này là số ngẫu nhiên K và chủ thể ký tính

$$\Pi = [K]G.$$

6.8.4.3 Chuẩn bị thông điệp để ký

Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp được ký, nghĩa là $M_2 = M$.

6.8.4.4 Tính bằng chứng

Chủ thể ký tính $R = FE2I(r, \Pi_X) mod\ q$ với bằng chứng đơn giản chỉ là một hàm của tiền chữ ký.

6.8.4.5 Tính nhiệm vụ

Chủ thể ký tính nhiệm vụ $(T_1, T_2) = (-R, BS2I(\gamma, H))$ với H là mã băm của thông điệp M_2.

6.8.4.6 Tính toán thành phần thứ hai của chữ ký

Chữ ký (R, S) với R được tính trong 6.8.4.4 và

$$S = X(KR - BS2I(\gamma, H)) mod\ q.$$

6.8.4.7 Xây dựng phần phụ lục

Phần phụ lục là phép ghép của (R, S) và một trường lựa chọn text , $text$, $((R, S), text)$.

6.8.4.8 Xây dựng thông điệp đã ký

Một thông điệp đã ký là phép ghép, M, và phần phụ lục

$$M||((R, S), text).$$

6.8.5 Tiến trình kiểm tra

6.8.5.1 Tổng quan

Chủ thể kiểm tra yêu cầu các dữ liệu cần thiết cho tiến trình kiểm tra.

6.8.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục. Để kiểm tra xem $0 < R < q$ và $0 < S < q$; nếu một trong các điều kiện đó bị vi phạm, chữ ký số sẽ bị từ chối.

6.8.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất M từ thông điệp đã ký và chia thông điệp thành hai phần M_1 và M_2. M_1 là rỗng và $M_2 = M$.

6.8.5.4 Truy xuất nhiệm vụ

Bước này giống với 6.8.4.5. Các đầu vào cho hàm nhiệm vụ bao gồm bằng chứng R từ 6.8.5.2 và M_2 từ 6.8.5.3. Nhiệm vụ $T = (T_1, T_2)$ được tính lại như là đầu ra của hàm nhiệm vụ 6.8.4.5

6.8.5.5 Tính lại tiền chữ ký

Các đầu vào cho bước này là hệ thống các tham số, khóa kiểm tra Y, nhiệm vụ $T = (T_1, T_2)$ từ 6.8.5.4 và thành phần thứ 2 của chữ ký số S từ 6.8.5.2. Người kiểm tra có được một giá trị tính lại Π' của tiền chữ ký dùng công thức:

$$\Pi' = [(-T_1))^{-1} S\ mod\ q]Y + [(-T_1)^{-1} T_2\ mod\ q]G.$$

6.8.5.6 Tính lại bằng chứng

Việc tính toán ở bước này giống như 6.8.4.4. Người kiểm tra thực hiện hàm bằng chứng. Đầu vào là Π' từ 6.8.5.5. Đầu ra là bằng chứng được tính lại R'

6.8.5.7 Kiểm tra bằng chứng

Người kiểm tra so sánh bằng chứng tính lại, R' từ 6.8.5.6 với giá trị của R từ 6.8.5.2. Nếu $R' = R$, thì chữ ký là hợp lệ

6.9 EC-RDSA

6.9.1 Tổng quan

EC-RDSA (Thuật toán chữ ký số đường cong elliptic Nga) là cơ chế ký với khóa kiểm tra $Y = [X]G$; đó là, tham số D bằng 1. Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp đã được ký, nghĩa là $M_2 = M$. Các hệ số (A, B, C) của chữ ký EC-RDSA được thiết lập như sau :

$$(A, B, C) = (T_1, T_2, -S)$$

Với $(T_1, T_2) = (H, R)$ và $H = h(M)$ là mã băm của thông điệp M, được biến đổi thành một số nguyên như mô tả trong 6.9.4.5.

Hàm bằng chứng được xác định theo công thức sau :

$$R = FE2I(r, \Pi_X) mod\ q.$$

Vì vậy công thức ký trở thành

$$HK + RX - S \equiv 0 (mod\ q).$$

CHÚ THÍCH EC-RDSA là viết tắt của thuật toán đường cong elliptic Nga. Cơ chế này được lấy từ viện dẫn [21]. Các ký hiệu được thay đổi so với viễn dẫn [22] để phù hợp với ký hiệu được dùng trong tiêu chuẩn này.

6.9.2 Các tham số

p một số nguyên tố

E một nhóm đường cong elliptic trên trường $GF(p)$

$\#E$ lực lượng của E

q ước nguyên tố của $\#E$

G một điểm trên đường cong elliptic bậc q

Định danh hàm băm hoặc OID với hàm băm được đặc tả.

Tất cả các tham số có thể được công khai và có thể dùng chung cho một nhóm người dùng.

6.9.3 Sinh khóa ký và khóa kiểm tra

Khóa ký của một chủ thể ký là một số nguyên bí mật X được sinh giả ngẫu nhiên hoặc ngẫu nhiên sao cho $0 < X < q$. Tham số D là 1. Tương ứng với khóa kiểm tra công khai Y là

$$Y = [X]G.$$

Một khóa ký bí mật X của người dùng và khóa kiểm tra công khai Y thường được cố định cho một chu kỳ thời gian. Khóa ký X được giữ bí mật.

CHÚ THÍCH Viện dẫn [21] không hoàn toàn đặc tả tiến trình sinh khóa ký bí mật X của một người dùng.

6.9.4 Tiến trình ký

6.9.4.1 Tạo số ngẫu nhiên

Chủ thể ký tính toán một số nguyên K ngẫu nhiên hoặc giả ngẫu nhiên sao cho $0 < K < q$.

6.9.4.2 Tạo tiền chữ ký

Đầu vào của bước này là số ngẫu nhiên K, và chủ thể ký tính

$$\Pi = [K]G.$$

6.9.4.3 Chuẩn bị thông điệp để ký

Thông điệp được chuẩn bị sao cho M_1 là rỗng và M_2 là thông điệp được ký, nghĩa là $M_2 = M$.

6.9.4.4 Tính bằng chứng

Chủ thể ký tính $R = FE2I(r, \Pi_X) mod\ q$.

6.9.4.5 Tính nhiệm vụ

Chủ thể ký tính $H = h(M_2)$. H sau đó được biến đổi thành một số nguyên theo quy tắc biến đổi $BS2I$ trong phụ lục B. Nếu $H = 0\ mod\ q$, thì H được gán bằng 1. Nhiệm vụ (T_1, T_2) là $(BS2I(\gamma, H), R)$, nếu $BS2I(\gamma, H) \neq 0 (mod\ q)$, ngược lại là $(1, R)$.

6.9.4.6 Tính toán thành phần thứ hai của chữ ký

Chữ ký (R, S) với R được tính như trong 6.9.4.4 và

$$S = RX + KH\ mod\ q.$$

Người ký kiểm tra xem $R = 0$ hoặc $S = 0$ hay không. Nếu một trong hai giá trị R hoặc $S = 0$, thì một giá trị mới của K được sinh ra và chữ ký được tính toán lại.

6.9.4.7 Xây dựng phần phụ lục

Phần phụ lục là phép ghép của (R, S) và một trường lựa chọn text , $text$, nghĩa là $((R, S)||text)$.

6.9.4.8 Xây dựng thông điệp đã ký

Một thông điệp đã ký là phép ghép, M, và phần phụ lục

$$M||((R, S)\ ||\ text).$$

6.9.5 Tiến trình kiểm tra

6.9.5.1 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục. Để kiểm tra xem $0 < R < q$ và $0 < S < q$; nếu một trong các điều kiện đó bị vi phạm, chữ ký số sẽ bị từ chối.

6.9.5.2 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất M từ thông điệp đã ký và chia thông điệp thành hai phần M_1 và M_2. M_1 là rỗng và $M_2 = M$.

6.9.5.3 Truy xuất nhiệm vụ

Bước này giống với 6.9.4.5. Các đầu vào cho hàm nhiệm vụ bao gồm bằng chứng R từ 6.9.5.1 và M_2 từ 6.9.5.2. Nhiệm vụ $T = (T_1, T_2)$ được tính lại như là đầu ra của hàm nhiệm vụ 6.9.4.5.

6.9.5.4 Tính lại tiền chữ ký

Các đầu vào cho bước này là hệ thống các tham số, khóa kiểm tra Y, nhiệm vụ $T = (T_1, T_2)$ từ 6.9.5.3 và thành phần thứ 2 của chữ ký số S từ 6.9.5.1. Người kiểm tra có được một giá trị tính lại Π' của tiền chữ ký dùng công thức:

$$\Pi' = [-T_1^{-1} T_2 mod\ q]Y + [T_1^{-1} S\ mod\ q]G.$$

6.9.5.5 Tính lại bằng chứng

Việc tính toán ở bước này giống như 6.9.4.4. Người kiểm tra thực hiện hàm bằng chứng. Đầu vào là Π' từ 6.9.5.4. Đầu ra là bằng chứng được tính lại R'

6.9.5.6 Kiểm tra bằng chứng

Người kiểm tra so sánh bằng chứng tính lại, R' từ 6.9.5.5 với giá trị của R từ 6.9.5.1. Nếu $R' = R$, thì chữ ký là hợp lệ.

6.10 EC-SDSA

6.10.1 Tổng quan

EC-SDSA (Thuật toán chữ ký số đường cong elliptic Schnorr) là cơ chế ký với khóa kiểm tra $Y = [X]G$; đó là, tham số D bằng 1. Thông điệp được chuẩn bị sao cho M_2 là rỗng và M_1 là thông điệp đã được ký, nghĩa là $M_1 = M$. Bằng chứng R là mã băm của thông điệp M và một tiền chữ ký ngẫu nhiên $\Pi = [K]G$, bằng một trong hai phương pháp sau :

Thông thường $R = h(FE2BS(r,\Pi_X) \parallel FE2BS(r,\Pi_Y) \parallel M)$

Hoặc

Được tối ưu $R = h(FE2BS(r,\Pi_X)||M)$.

Phương pháp đầu tiên sinh ra bằng chứng bằng cách băm ghép nối tọa độ của Π theo trục x và trục y, và thông điệp M. Phương pháp thứ hai bỏ qua tọa độ y từ việc tính toán băm và qua đó nâng cao hiệu năng.

Phương pháp thứ hai là một biến thể tối hưu hóa của EC-SDSA (xem viện dẫn [30]).

Các hệ số (A, B, C) của EC-SDSA được thiết lập như sau

$$(A, B, C) = (T_1, T_2, S)$$

Với $(T_1, T_2) = (-1, -BS2I(\gamma, R) mod\ q)$.

Vì vậy công thức ký trở thành

$$-K + T_2 X + S \equiv 0 (mod\ q).$$

CHÚ THÍCH EC-SDSA viết tắt cho thuật toán chữ ký số đường cong elliptic Schnorr. Cơ chế này được lấy từ viện dẫn [33]. Các ký hiệu được thay đổi một chút so với viễn dẫn [33] để phù hợp với ký hiệu được dùng trong tiêu chuẩn này.

6.10.2 Các tham số

F một trường hữu hạn

E một nhóm đường cong elliptic trên trường F

$\#E$ lực lượng của E

q ước nguyên tố của $\#E$

G một điểm trên đường cong elliptic bậc q

Định danh hàm băm hoặc OID với hàm băm được đặc tả.

Tất cả các tham số có thể được công khai và có thể dùng chung cho một nhóm người dùng.

Một khuyến nghị đối với toàn bộ người dùng kiểm tra việc sinh hợp lệ của các tham số công khai theo viện dẫn [7].

6.10.3 Sinh khóa ký và khóa kiểm tra

Khóa ký của một chủ thể ký là một số nguyên bí mật X được sinh giả ngẫu nhiên hoặc ngẫu nhiên sao cho $0 < X < q$. Tham số D là 1. Tương ứng với khóa kiểm tra công khai Y là

$$Y = [X]G.$$

Một khóa ký bí mật X của người dùng và khóa kiểm tra công khai Y thường được cố định cho một chu kỳ thời gian. Khóa ký X được giữ bí mật.

6.10.4 Tiến trình ký

6.10.4.1 Tạo số ngẫu nhiên

Chủ thể ký tính toán một số nguyên K ngẫu nhiên hoặc giả ngẫu nhiên sao cho $0 < K < q$.

6.10.4.2 Tạo tiền chữ ký

Đầu vào bước này là số nguyên nhẫu nhiên K, và chủ thể ký tính $\Pi = [K]G$.

6.10.4.3 Chuẩn bị thông điệp để ký

Thông điệp được chuẩn bị sao cho M_1 là thông điệp được ký, nghĩa là $M_1 = M$, và M_2 là rỗng.

6.10.4.4 Tính bằng chứng

Chủ thể ký tính $R = h\big(FE2BS(r, \Pi_X)||FE2BS(r, \Pi_Y)||M\big)$.

Để tối ưu hóa biến thể của EC-SDSA, chủ thể ký tính $R = h(FE2BS(r, \Pi_X)||M)$.

6.10.4.5 Tính nhiệm vụ

Giá trị của bằng chứng R được biến đổi thành một số nguyên theo quy tắc biến đổi BS2I, trong phụ lục B và rút gọn theo modulo q.

Nhiệm vụ (T_1, T_2) là $(-1, -BS2I(\gamma, R) mod\ q)$.

6.10.4.6 Tính toán phần thứ hai của chữ ký số

Chữ ký (R, S) với $S = (K + BS2I(\gamma, R)X) mod\ q$.

Như một sự lựa chọn, một mong muốn để kiểm tra nếu $R = 0$ hoặc $S = 0$. Nếu một trong hai giá trị $R = 0$ hoặc $S = 0$, một giá trị mới của K được sinh ra và chữ ký được tính toán lại (rất hiếm khi xảy ra $R = 0$ hoặc $S = 0$ nếu chữ ký số được sinh đúng).

6.10.4.7 Xây dựng phần phụ lục

Phần phụ lục là phép ghép của (R, S) và một trường lựa chọn text , $text$.

6.10.4.8 Xây dựng thông điệp đã ký

Một thông điệp đã ký là phép ghép thông điệp M và phần phụ lục.

$$M||((R, S)||text).$$

6.10.5 Tiến trình kiểm tra

6.10.5.1 Tổng quan

Chủ thể kiểm tra yêu cầu các dữ liệu cần thiết được yêu cầu cho tiến trình kiểm tra.

6.10.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục. Người kiểm tra kiểm tra xem R là một xâu khác 0 trong dải của hàm băm hay không và $0 < S < q$. Nếu một trong các điều kiện đó bị vi phạm, chữ ký số sẽ bị từ chối.

6.10.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất M từ thông điệp đã ký và chia thông điệp thành hai phần M_1 và M_2. M_2 là rỗng và $M_1 = M$.

6.10.5.4 Truy xuất nhiệm vụ

Đầu vào của hàm nhiệm vụ bao gồm bằng chứng R từ 6.10.5.2. Nhiệm vụ $T = (T_1, T_2) = (-1, -BS2I(\gamma, R) mod\ q)$.

6.10.5.5 Tính lại tiền chữ ký

Các đầu vào cho bước này là hệ thống các tham số, khóa kiểm tra Y, nhiệm vụ $T = (T_1, T_2)$ từ 6.10.5.4 và thành phần thứ 2 của chữ ký số S từ 6.10.5.2. Người kiểm tra có được một giá trị tính lại Π' của tiền chữ ký dùng công thức:

$$\Pi' = [-ST_1 mod\ q]G + [T_2\ mod\ q]Y = [S\ mod\ q]G + [T_2\ mod\ q]Y.$$

6.10.5.6 Tính lại bằng chứng

Việc tính toán ở bước này giống như 6.10.4.4 và 6.10.4.5. Người kiểm tra thực hiện hàm bằng chứng từ 6.10.4.4. Đầu vào là Π' từ 6.10.5.5. Đầu ra là bằng chứng được tính lại R' là mã băm của tiền chữ ký được tính lại Π' và thông điệp M.

6.10.5.7 Kiểm tra bằng chứng

Người kiểm tra so sánh bằng chứng tính lại, R' từ 6.10.5.6 với giá trị của R từ 6.10.5.2. Nếu $R' = R$, thì chữ ký là hợp lệ.

6.11 EC-FSDSA

6.11.1 Tổng quan

EC-FSDSA (Thuật toán chữ ký số đường cong elliptic Schnorr đầy đủ) là cơ chế chữ ký số với khóa kiểm tra $Y = [X]G$; tham số D bằng 1. Thông điệp được chuẩn bị sao cho M_1 là rỗng và $M_2 = M$ là thông điệp được ký. Bằng chứng R được tính như sau:

$$R = FE2BS(r, \Pi_X) || FE2BS(r, \Pi_Y).$$

Các hệ số (A, B, C) của công thức chữ ký EC-FSDSA được thiết lập như sau:

$$(A, B, C) = (T_1, T_2, S).$$

Với $T = (T_1, T_2) = (-1, -BS2I(\gamma, h(R \parallel M)) mod\ q)$.

Vì vậy, công thức ký sẽ là :

$$-K + T_2 X + S \equiv 0 (mod\ q).$$

CHÚ THÍCH EC-SDSA viết tắt cho thuật toán chữ ký số đường cong elliptic Schnorr đầy đủ. Cơ chế này được lấy từ viện dẫn [33]. Các ký hiệu được thay đổi một chút so với viễn dẫn [33] để phù hợp với ký hiệu được dùng trong tiêu chuẩn này.

6.11.2 Các tham số

F một trường hữu hạn

E một nhóm đường cong elliptic trên trường F

$\#E$ lực lượng của E

q ước nguyên tố của $\#E$

G một điểm trên đường cong elliptic bậc q

Định danh hàm băm hoặc OID với hàm băm được đặc tả.

Tất cả các tham số có thể được công khai và có thể dùng chung cho một nhóm người dùng.

Một khuyến nghị đối với toàn bộ người dùng kiểm tra việc sinh hợp lệ của các tham số công khai theo viện dẫn [7].

6.11.3 Sinh khóa ký và khóa kiểm tra

Khóa ký của một chủ thể ký là một số nguyên bí mật X được sinh giả ngẫu nhiên hoặc ngẫu nhiên sao cho $0 < X < q$. Tham số D là 1. Tương ứng với khóa kiểm tra công khai Y là

$$Y = [X]G.$$

Một khóa ký bí mật X của người dùng và khóa kiểm tra công khai Y thường được cố định cho một chu kỳ thời gian. Khóa ký X được giữ bí mật.

6.11.4 Tiến trình ký

6.11.4.1 Tạo số ngẫu nhiên

Chủ thể ký tính toán một số nguyên ngẫu nhiên hoặc giả ngẫu nhiên K sao cho $0 < K < q$.

6.11.4.2 Tạo tiền chữ ký

Đầu vào bước này là số ngẫu nhiên K, và chủ thể ký tính $\Pi = [K]G$.

6.11.4.3 Chuẩn bị thông điệp để ký

Thông điệp được chuẩn bị sao cho M là thông điệp được ký, nghĩa là $M_2 = M$, và M_1 là rỗng.

6.11.4.4 Tính bằng chứng

Chủ thể ký tính $R = FE2BS(r, \Pi_X)||FE2BS(r, \Pi_Y)$.

6.11.4.5 Tính nhiệm vụ

Chủ thể ký tính mã băm $h(R||M)$. Sau đó, hàm băm được biến đổi thành một số nguyên theo quy tắc biến đổi, BS2I, trong phụ lục B và sau đó được rút gọn theo modulo q. Nhiệm vụ (T_1, T_2) là $(-1, -BS2I(\gamma, h(R||M)) mod\ q)$.

6.11.4.6 Tính toán thành phần thứ hai của chữ ký

Chữ ký (R, S) với $S = (K + BS2I(\gamma, h(R \parallel M))X mod\ q)$.

Như một sự lựa chọn, một mong muốn để kiểm tra nếu $R = 0$ hoặc $S = 0$. Nếu một trong hai giá trị $R = 0$ hoặc $S = 0$, một giá trị mới của K được sinh ra và chữ ký được tính toán lại (rất hiếm khi xảy ra $R = 0$ hoặc $S = 0$ nếu chữ ký số được sinh đúng).

6.11.4.7 Xây dựng phần phụ lục

Phần phụ lục là phép ghép của (R, S) và một trường lựa chọn text , $text$.

6.11.4.8 Xây dựng thông điệp đã ký

Một thông điệp đã ký là phép ghép của thông điệp M và phần phụ lục

$$M||((R, S)||text)$$

6.11.5 Tiến trình kiểm tra

6.11.5.1 Tổng quan

Chủ thể kiểm tra yêu cầu các dữ liệu cần thiết được yêu cầu cho quá trình kiểm tra.

6.11.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục. Người kiểm tra kiểm tra xem R là một xâu khác 0 trong dải của hàm băm hay không và $0 < S < q$. Nếu một trong các điều kiện đó bị vi phạm, chữ ký số sẽ bị từ chối.

6.11.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất M từ thông điệp đã ký và chia thông điệp thành hai phần M_1 và M_2, sao cho $M_2 = M$ và M_1 là rỗng.

6.11.5.4 Truy xuất nhiệm vụ

Đầu vào của hàm nhiệm vụ được tính như trong 6.11.4.5 từ bằng chứng R từ 6.11.4.4 và thông điệp M từ 6.11.4.3. Nhiệm vụ được đưa ra bởi $T = (T_1, T_2) = (-1, -BS2I(\gamma, h(R \parallel M)) mod\ q)$.

6.11.5.5 Tính lại tiền chữ ký

Các đầu vào cho bước này là hệ thống các tham số, khóa kiểm tra Y, nhiệm vụ $T = (T_1, T_2)$ từ 6.11.5.4 và thành phần thứ 2 của chữ ký số S từ 6.11.5.2. Người kiểm tra có được một giá trị tính lại Π' của tiền chữ ký dùng công thức:

$$\Pi' = [-ST_1 mod\ q]G + [T_2 mod\ q]Y = [S\ mod\ q]G + [T_2 mod\ q]Y.$$

6.11.5.6 Tính lại bằng chứng

Việc tính toán ở bước này giống như 6.11.4.4. Người kiểm tra thực hiện hàm bằng chứng. Đầu vào là Π' từ 6.11.5.5. Đầu ra được tính lại là bằng chứng R'.

6.11.5.7 Kiểm tra bằng chứng

Người kiểm tra so sánh bằng chứng tính lại, R' từ 6.11.5.6 với giá trị của R từ 6.11.5.2. Nếu $R' = R$, thì chữ ký là hợp lệ

7 Các cơ chế dựa trên định danh

7.1 Tổng quan

Các dữ liệu được yêu cầu cho tiến trình ký như sau:

- Các tham số miền, $E, GF(r), G_1, G_2, q, P, <>$;

- Khóa chủ công khai V;

- Khóa ký X;

- Thông điệp M;

- Định danh hàm băm cho H_1 và H_2 (tùy chọn);

- Xâu định danh ID;

- Ký tự khác (tùy chọn)

Định danh hàm băm có thể được sử dụng để che giấu cơ chế ký và hàm băm.

Các dữ liệu được yêu cầu cho tiến trình kiểm tra như sau:

- Các tham số miền, $E, GF(r), G_1, G_2, q, P, <>$;

- Khóa chủ công khai V;

- Khóa kiểm tra Y, có thể được dẫn xuất từ một xâu định danh;

- Thông điệp M;

- Chữ ký $\sum$;

- Định danh hàm băm cho H_1 và H_2 (tùy chọn);

CHÚ THÍCH 1 Người ký và người kiểm tra phải thống nhất về các hàm băm cụ thể cho h, H_1 và H_2 được dùng trong cơ chế. Nếu không xác định được hàm băm, thì hàm băm định danh được yêu cầu sử dụng cho cả tiến trình ký và tiến trình kiểm tra (xem TCVN 12214-1 (ISO/IEC 14888-1)).

- Xâu định danh ID;

- Ký tự khác (tùy chọn).

CHÚ THÍCH 2 Các đường cong elliptic điển hình cho IBS-1 và IBS-2 là các đường cong elliptic siêu kỳ dị trên $GF(r)$, trong đó $r = p^m$, p là số nguyên tố ≥ 2 và m là số nguyên ≥ 1.

7.2 IBS-1

7.2.1 Tổng quan

IBS-1 là một lược đồ ký dựa trên định danh trên một nhóm cộng của các điểm trên đường cong elliptic. Ta có:

$$(A, B, C) = (T_1, S, T_2)$$

Trong đó $T_1 = -Y, T_2 = [R]Y. D = -1$. Do đó, công thức ký biến đổi thành:

$[-K]Y + [U^{-1}]S + [R]Y \equiv 0_E \ (trong\ G_1)$.

CHÚ THÍCH Cơ chế này dựa trên thuật toán được thiết kế bởi Hess trong viện dẫn [22].

7.2.2 Các tham số

Cơ chế ký được thực hiện trong trường hợp các thực thể tham gia chia sẻ các tham số được xác định trong Điều 4 như sau: $G_1, G_2, P, q, <>, H_1$ và H_2.

Một khuyến nghị đối với toàn bộ người dùng kiểm tra việc sinh hợp lệ của các tham số công khai.

7.2.3 Sinh khóa chủ và khóa ký/kiểm tra

Cặp khóa chủ của KGC là (U, V), trong đó U là khóa chủ riêng được sinh ra bằng cách lựa chọn một số nguyên ngẫu nhiên sao cho $0 < U < q$ và V là khóa chủ công khai được sinh ra bằng cách tính $V = [U]P$. KGC công khai V và giữ bí mật U.

Cặp khóa ký và kiểm tra của người ký là (X, Y), trong đó Y là khóa kiểm tra công khai được sinh ra từ một xâu định danh ID và hàm băm H_1, tức là $Y = H_1(ID)$ và X là khóa ký riêng được sinh ra bằng cách tính $X = [U]Y$, do KGC thực hiện và gửi cho người ký.

7.2.4 Tiến trình ký

7.2.4.1 Tạo số ngẫu nhiên

Trước tiên người ký lựa chọn ngẫu nhiên hoặc giả ngẫu một số nguyên K nhiên sao cho $0 < K < q$. Người ký giữ bí mật giá trị K.

7.2.4.2 Tạo tiền chữ ký

Người ký lấy đầu vào là K, P và X để tạo ra tiền chữ ký.

$$\Pi =< X, P >^K.$$

CHÚ THÍCH Π là một phần tử trên một trường mở rộng của $GF(p^m)$ và phần mở rộng bậc 4 với $p = 2$, bậc 6 với $p = 3$ và bậc 2 với $p > 3$.

7.2.4.3 Chuẩn bị thông điệp để ký

Người ký chuẩn bị thông điệp ký sao cho M_2 là rỗng và M_1 là thông điệp được ký, tức là $M_1 = M$.

7.2.4.4 Tính bằng chứng

Đặt $\Pi = (\Pi_a, \Pi_b)$. Người ký sử dụng hàm băm H_2 cho $M_1 \parallel FE2BS(r, \Pi_a) \parallel FE2BS(r, \Pi_b)$ (Phép ghép của M_1 và $FE2BS(r, \Pi_a)$ và $FE2BS(r, \Pi_b)$) để thu được bằng chứng.

$$R = BS2I(\gamma, H_2(M_1 \parallel FE2BS(r, \Pi_a) \parallel FE2BS(r, \Pi_b)))mod\ q.$$

Nếu $R = 0$, thì đầu ra là không hợp lệ và dừng.

Đối với các trường có bậc mở rộng cao hơn, sẽ có nhiều thành phần xuất hiện trong giá trị được băm. Ví dụ, đối với bậc mở rộng 3, $\Pi = (\Pi_a, \Pi_b, \Pi_c)$ và đầu vào của H_2 có thể là:

$$M_1 \parallel FE2BS(r, \Pi_a) \parallel FE2BS(r, \Pi_b) \parallel FE2BS(r, \Pi_c).$$

7.2.4.5 Tính nhiệm vụ

Nhiệm vụ $T = (T_1, T_2)$ hoặc $(-Y, [R]Y)$. Tuy nhiên, người ký không cần tính nhiệm vụ.

7.2.4.6 Tính toán phần thứ hai của chữ ký

Người ký tính toán phần thứ hai của chữ ký như sau:

$$S = [K - R]X.$$

Chữ ký là $\sum = (R, S)$.

7.2.4.7 Xây dựng phần phụ lục

Phần phụ lục được xây dựng bởi người ký là phép ghép của (R, S) với một trường text tùy chọn $((R, S), text)$.

7.2.4.8 Xây dựng thông điệp đã ký

Một thông điệp đã ký là phép ghép của thông điệp M và phần phụ lục

$$M \parallel ((R, S) \parallel text)$$

7.2.5 Tiến trình kiểm tra

7.2.5.1 Tổng quan

Chủ thể kiểm tra yêu cầu các dữ liệu cần thiết cho tiến trình kiểm tra.

7.2.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất bằng chứng R và thành phần thứ hai của chữ ký S từ phần phụ lục và kiểm tra xem $S \in G_1$ hay không; Nếu điều kiện nay bị vi phạm, chữ ký số sẽ bị từ chối. Ngược lại, người kiểm tra sẽ thực hiện các bước sau.

7.2.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất M từ thông điệp đã ký và chia thông điệp thành hai phần M_1 và M_2, sao cho $M_1 = M$ và M_2 là rỗng.

7.2.5.4 Truy xuất nhiệm vụ

Nhiệm vụ $T = (T_1, T_2)$ trong đó $T_1 = -Y$ và $T_2 = [R]Y$. Tuy nhiên, người kiểm tra không cần tính nhiệm vụ.

7.2.5.5 Tính lại tiền chữ ký

Người kiểm tra tính lại giá trị tiền chữ ký

$$\Pi' =< S,P >*< Y,V >^{R}.$$

CHÚ THÍCH Cặp $< Y,V >$ có thể tính toán trước.

7.2.5.6 Tính lại bằng chứng

Người kiểm tra tính lại bằng chứng

$$R' = BS2I\left(\gamma, H_2(M_1 \parallel FE2BS(r,\Pi'_a) \parallel FE2BS(r,\Pi'_b))\right) mod\ q$$

Đối với các trường có bậc mở rộng cao hơn, sẽ có nhiều thành phần xuất hiện trong giá trị được băm. Ví dụ, đối với bậc mở rộng 3, $\Pi' = (\Pi'_a, \Pi'_b, \Pi'_c)$ và đầu vào của H_2 có thể là:

$$M_1 \parallel FE2BS(r,\Pi'_a) \parallel FE2BS(r,\Pi'_b) \parallel FE2BS(r,\Pi'_c).$$

7.2.5.7 Kiểm tra bằng chứng

Người kiểm tra so sánh $R' = R$ hay không. Nếu bằng thì chữ ký được kiểm tra, ngược lại chữ ký là không hợp lệ

7.3 IBS-2

7.3.1 Tổng quan

IBS-2 là một lược đồ ký dựa trên định danh trên một nhóm cộng của các điểm trên đường cong elliptic. Ta có:

$$(A,B,C) = (T_1,S,T_2)$$

Trong đó $T_1 = -Y, T_2 = [-H]Y$ và H là một mã băm từ $H_2. D = -1$.

Do đó, công thức chữ ký là:

$$[-K]Y + [U^{-1}]S + [-H]Y \equiv 0_E (trong\ G_1).$$

CHÚ THÍCH Cơ chế này dựa trên thuật toán được thiết kế bởi Hess trong viện dẫn [15].

7.3.2 Các tham số

Các tham số giống như Điều 7.2.2.

7.3.3 Sinh khóa chủ và khóa ký/kiểm tra

Quá trình này giống như Điều 7.2.3.

7.3.4 Tiến trình ký

7.3.4.1 Tạo số ngẫu nhiên

Trước tiên người ký lựa chọn ngẫu nhiên hoặc giả ngẫu một số nguyên K nhiên sao cho $0 < K < q$. Người ký giữ bí mật giá trị K.

7.3.4.2 Tạo tiền chữ ký

Người ký lấy đầu vào là K, P và X để tạo ra tiền chữ ký.

$$\Pi = [K]Y.$$

7.3.4.3 Chuẩn bị thông điệp để ký

Người ký chuẩn bị thông điệp ký sao cho M_1 là rỗng và M_2 là thông điệp được ký, tức là $M_2 = M$.

7.3.4.4 Tính bằng chứng

Chủ thể ký nhận được bằng chứng từ kết quả tiền chữ ký

$$R = \Pi.$$

7.3.4.5 Tính nhiệm vụ

Nhiệm vụ $T = (T_1, T_2)$ trong đó:

$T_1 = -Y$, và

$T_2 = [-H]Y$

Trong đó $H = BS2I(\gamma, H_2(M_2 || FE2BS(r, \Pi_X))) \bmod q$. Tuy nhiên, người ký chỉ cần tính toán giá trị H.

7.3.4.6 Tính toán phần thứ hai của chữ ký

Người ký tính toán phần thứ hai của chữ ký như sau:

$S = [K + H]X$.

Chữ ký là $\sum = (R, S)$.

7.3.4.7 Xây dựng phần phụ lục

Phần phụ lục được xây dựng bởi người ký là phép ghép của (R, S) với một trường text tùy chọn $((R, S), text)$.

7.3.4.8 Xây dựng thông điệp được ký

Một thông điệp đã ký là phép ghép của thông điệp M và phần phụ lục, tức là

$$M \parallel ((R, S) \parallel text).$$

7.3.5 Tiến trình kiểm tra

7.3.5.1 Tổng quan

Chủ thể kiểm tra yêu cầu các dữ liệu cần thiết cho tiến trình kiểm tra.

7.3.5.2 Truy xuất bằng chứng

Người kiểm tra truy xuất tiền chữ ký R và thành phần thứ hai của chữ ký S từ phần phụ lục. Trước tiên, người kiểm tra kiểm tra xem $S \in G_1$ hay không; Nếu điều kiện này bị vi phạm, chữ ký số sẽ bị từ chối. Ngược lại, người kiểm tra sẽ thực hiện các bước sau.

7.3.5.3 Chuẩn bị thông điệp để kiểm tra

Người kiểm tra truy xuất M từ thông điệp đã ký và chia thông điệp thành hai phần M_1 và M_2, sao cho $M_2 = M$ và M_1 là rỗng.

7.3.5.4 Truy xuất nhiệm vụ

Nhiệm vụ $T = (T_1, T_2)$ trong đó:

$T_1 = -Y$, và

$T_2 = [-H]Y$.

Trong đó $H = BS2I(\gamma, H_2(M_2 \parallel FE2BS(r, \Pi_X)))mod\ q$. Tuy nhiên, người kiểm tra chỉ cần tính toán lại giá trị H.

7.3.5.5 Tính lại tiền chữ ký

Người kiểm tra truy xuất giá trị tiền chữ ký là:

$\Pi' = R$.

7.3.5.6 Tính toán lại bằng chứng

Thay bằng việc tính toán lại giá trị bằng chứng R, người kiểm tra tính toán hai cặp $< P, S >$ và $< V, \Pi' + [H]Y >$.

7.3.5.7 Kiểm tra bằng chứng

Người kiểm tra kiểm tra xem $< P, S >=< V, \Pi' + [H]Y >$ hay không. Nếu thỏa mãn điều kiện thì chữ ký được h; ngược lại, chữ ký không hợp lệ.

Phụ lục A
(Quy định)
Định danh đối tượng

Phụ lục A liệt kê các định danh đối tượng gán cho các cơ chế ký số đặc tả trong tiêu chuẩn này và xác định cấu trúc tham số thuật toán.

```
DigitalSignatureWithAppendixDL {
   iso(1) standard(0) digital-signature-with-appendix (14888) part3(3)
      asn1-module(1) discrete-logarithm-based-mechanisms(0) }
DEFINITIONS EXPLICIT TAGS ::= BEGIN
```

- - Xuất ra toàn bộ --

THÊM

```
      HashFunctions
         FROM DedicatedHashFunctions {
            iso(1) standard(0) encryption-algorithms(10118) part3(3)
            asn1-module(1)
                dedicated-hash-functions(0) } ;

   OID ::= OBJECT IDENTIFIER -- alias
```

- - Đồng bộ --

```
id-dswa-dl OID ::= {
   iso(1) standard(0) digital-signature-with-appendix(14888) part3(3)
      algorithm(0) }
```

-- Thực hiện –

```
id-dswa-dl-DSA      OID  ::=   {  iso(1)  member-body(2)  us(840)   ansi-x9-57(10040)
x9cm(4) dsa(1) }
id-dswa-dl-KCDSA    OID  ::=   { id-dswa-dl kcdsa(2) }
id-dswa-dl-PVS      OID  ::=   { id-dswa-dl pvs(3) }
id-dswa-dl-EC-DSA   OID  ::=   {  iso(1)  member-body(2)  us(840)   ansi-x9-62(10045)
signatures(4) ecdsa-with-Recommended(2) }
id-dswa-dl-EC-KCDSA OID  ::=   { id-dswa-dl ec-kcdsa(5) }
id-dswa-dl-EC-GDSA  OID  ::=   { id-dswa-dl ec-gdsa(6) }
id-dswa-dl-IBS-1    OID  ::=   { id-dswa-dl ibs-1(7) }
id-dswa-dl-IBS-2    OID  ::=   { id-dswa-dl ibs-2(8) }
id-dswa-dl-EC-RDSA  OID  ::=   { id-dswa-dl ec-rdsa(9) }
id-dswa-dl-SDSA     OID  ::=   { id-dswa-dl sdsa(10) }
id-dswa-dl-EC-SDSA  OID  ::=   { id-dswa-dl ec-sdsa(11) }
id-dswa-dl-EC-FSDSA OID  ::=   { id-dswa-dl ec-fsdsa(12) }
id-dswa-dl-EC-SDSA-opt OID ::= { id-dswa-dl ec-sdsa-opt(13) }

DigitalSignatureWithAppendix ::= SEQUENCE {
   algorithm  ALGORITHM.&id({DSAlgorithms}),
   parameters ALGORITHM.&Type({DSAlgorithms}{@algorithm})  OPTIONAL
}

DSAlgorithms ALGORITHM ::= {
   dswa-dl-DSA      |
   dswa-dl-KCDSA    |
   dswa-dl-PVS      |
   dswa-dl-EC-DSA   |
   dswa-dl-EC-KCDSA |
   dswa-dl-EC-GDSA  |
   dswa-dl-IBS-1    |
   dswa-dl-IBS-2    |
   dswa-dl EC-RDSA  |
   dswa-dl SDSA     |
```

```
   dswa-dl EC-SDSA  |
   dswa-dl EC-FSDSA |
   dswa-dl EC-SDSA-opt,

   ...   -- Expect additional algorithms --
}

dswa-dl-DSA ALGORITHM ::= {
   OID id-dswa-dl-DSA PARMS NullParms
}

dswa-dl-KCDSA ALGORITHM ::= {
   OID id-dswa-dl-KCDSA PARMS HashFunctions
}

dswa-dl-PVS ALGORITHM ::= {
   OID id-dswa-dl-PVS PARMS HashFunctions
}

dswa-dl-EC-DSA ALGORITHM ::= {
   OID id-dswa-dl-EC-DSA PARMS NullParms
}

dswa-dl-EC-KCDSA ALGORITHM ::= {
   OID id-dswa-dl-EC-KCDSA PARMS HashFunctions
}

dswa-dl-EC-GDSA ALGORITHM ::= {
   OID id-dswa-dl-EC-GDSA PARMS HashFunctions
}

dswa-dl-IBS-1 ALGORITHM ::= {
   OID id-dswa-dl-IBS-1 PARMS HashFunctions
}

dswa-dl-IBS-2 ALGORITHM ::= {
   OID id-dswa-dl-IBS-2 PARMS HashFunctions
}

dswa-dl-EC-RDSA ALGORITHM ::= {
   OID id-dswa-dl-EC-RDSA PARMS HashFunctions
}

dswa-dl-SDSA ALGORITHM ::= {
OID id-dswa-dl-SDSA PARMS HashFunctions
}

dswa-dl-EC-SDSA ALGORITHM ::= {
OID id-dswa-dl-EC-SDSA PARMS HashFunctions
}

dswa-dl-EC-FSDSA ALGORITHM ::= {
OID id-dswa-dl-EC-FSDSA PARMS HashFunctions
}

dswa-dl-EC-SDSA-opt ALGORITHM ::= {
   OID id-dswa-dl-EC-SDSA-opt PARMS HashFunctions
}


NullParms ::= NULL

-- Cryptographic algorithm identification --

ALGORITHM ::= CLASS {
   &id     OBJECT IDENTIFIER  UNIQUE,
   &Type   OPTIONAL
}
 WITH SYNTAX { OID &id [PARMS &Type] }

END -- DigitalSignatureWithAppendixDL --
```

CHÚ THÍCH 1 Các OID thay thế cho KCDSA được biểu diễn trong KCAC.TG.OID như sau :

{iso(1) member-body(2) korea(410) kisa(20004) npki-alg(1) kcdsa1(21)}

- KCDSA

{iso(1) member-body(2) korea(410) kisa(20004) npki-alg(1) kcdsalWithHAS160(22)}

- KCDSA với HAS160, trong đó HAS160 là một thuật toán băm tiêu chuẩn của Hàn Quốc

{iso(1) member-body(2) korea(410) kisa(20004) npki-alg(1) kcdsalWithSHA1(23)}

- KCDSA với SHA1

CHÚ THÍCH 2 OID thay thế cho EC-KCDSA với HAS160 được biểu diễn trong TTAS.KO-12.0015 là

{iso(1) member-body(2) korea(410) kisa(20004) npki-alg(1) ecc(100) signature(4)

eckcdsa-with-HAS160(1)}.

Phụ lục B
(Quy định)
Các hàm biến đổi (I)

B.1 Biến đổi từ một phần tử trường thành một số nguyên: *FE2I(r,x)*

Các giá trị đầu vào

- r – một số nguyên tố hoặc lũy thừa của một số nguyên tố.
- x – là một phần tử trong trường Galois $GF(r)$.

Các giả định

- Khi $r = p$, trong đó p là một số nguyên tố lẻ:

$x \in GF(p)$ được biểu diễn là một số nguyên trong tập $\{0,1, \dots, p-1\}$.

- Khi $r = p^m$, trong đó p là một số nguyên tố lẻ và m là một số nguyên lớn hơn 1:

$x \in GF(p^m)$ được biểu diễn bằng một xâu p phân độ dài m ;

$x = x_{m-1}x_{m-2} \dots x_0$, trong đó $x_i \in \{0,1,..,p-1\}$ với $0 \leq i < m$.

- Khi $r = 2$:

$x \in GF(2)$ được biểu diễn dưới dạng một số nguyên trong tập $\{0,1\}$.

- Khi $r = 2^m$, trong đó m là một số nguyên lớn hơn 1:

$x \in GF(2^m)$ được biểu diễn dưới dạng một xâu nhị phân có độ dài m;

$x = x_{m-1}x_{m-2} \dots x_0$, trong đó $x_i \in \{0,1\}$ với $0 \leq i < m$.

Giá trị đầu ra

- Khi $r = p$, trong đó p là một số nguyên tố lẻ:

$FE2I(r,x) = x \in \{0,1,\dots,p-1\}$.

- Khi $r = p^m$, trong đó p là một số nguyên tố lẻ, m là một số nguyên lớn hơn 1 và x được biểu diễn dưới dạng một xâu p phân $x_{m-1}x_{m-2} \dots x_0$:

$FE2I(r,x) = p^{m-1}x_{m-1} + p^{m-2}x_{m-2} + \dots + x_0 \in \{0,1, \dots, p^m - 1\}$.

- Khi $r = 2$:

$FE2I(r,x) = x \in \{0,1\}$.

- Khi $r = 2^m$, trong đó m là một số nguyên lớn hơn 1 và x được biểu diễn dưới dạng xâu nhị phân $x_{m-1}x_{m-2} \dots x_0$:

$FE2I(r,x) = 2^{m-1}x_{m-1} + 2^{m-2}x_{m-2} + \dots + x_0 \in \{0,1, \dots, 2^m - 1\}$.

B.2 Biến đổi một số nguyên thành một phần tử trường: *I2FE(r,x)*

Các giá trị đầu vào

- r – một số nguyên tố hoặc lũy thừa của một số nguyên tố.
- x – là một số nguyên trong tập $\{0,1, \dots, r-1\}$.

Các giả định

- Khi $r = p$, trong đó p là một số nguyên tố lẻ:

Các phần tử của $GF(p)$ được biểu diễn dưới dạng các số nguyên trong tập $\{0,1, \dots, p-1\}$.

- Khi $r = p^m$, trong đó p là một số nguyên tố lẻ và m là một số nguyên lớn hơn 1:

Các phần tử của $GF(p^m)$ được biểu diễn dưới danh một xâu p phân độ dài m.

- Khi $r = 2$:

Các phần tử của $GF(2)$ được biểu diễn dưới dạng các số nguyên trong tập $\{0,1\}$.

- Khi $r = 2^m$, trong đó m là một số nguyên lớn hơn 1:

Các phần tử trong $GF(2^m)$ được biểu diễn dưới dạng một xâu nhị phân có độ dài m.

Giá trị đầu ra

- Khi $r = p$ là một số nguyên tố lẻ:

$I2FE(r,x) = x \in \{0,1,\ldots,p-1\}$.

- Khi $r = p^m$, trong đó p là một số nguyên tố lẻ, m là một số nguyên lớn hơn 1:

$I2FE(r,x) = x_{m-1}x_{m-2}\ldots x_0$.

Một xâu p phân gồm m thành phần (biểu diễn mở rộng của x theo cơ số p, đệm thêm 0 nếu cần để đạt độ dài mong muốn) được tính toán như sau:

```
a := x;
i := 0;
While (i< m ) do {
   b := ⌊a/p⌋;
   x_i := a - (p)(b);
   a := b;
   i:=i+1}
```

- Khi $r = 2$:

$I2FE(r,x) = x \in \{0,1\}$.

- Khi $r = 2^m$, trong đó m là một số nguyên lớn hơn 1:

$I2FE(r,x) = x_{m-1}x_{m-2}\ldots x_0$,

Một xâu nhị phân gồm m thành phần (biểu diễn mở rộng của x theo cơ số 2, đệm thêm 0 nếu cần để đạt độ dài mong muốn) được tính toán như sau:

```
a := x;
i := 0;
While (i < m ) do {
   b := ⌊a/2⌋;
   x_i := a − (2)(b) ;
   a := b;
   i := i + 1}
```

B.3 Biến đổi một phần tử trường thành một xâu nhị phân: *FE2BS(r,x)*

Các giá trị đầu vào

- r – một số nguyên tố hoặc lũy thừa của một số nguyên tố.
- x – là một phần tử trong trường Galois $GF(r)$.

Các giả định

- Khi $r = p$, trong đó p là một số nguyên tố lẻ:

$x \in GF(p)$ được biểu diễn dưới dạng một số nguyên trong tập $\{0,1,\ldots,p-1\}$.

- Khi $r = p^m$, trong đó p là một số nguyên tố lẻ và m là một số nguyên lớn hơn 1:

$x \in GF(p^m)$ được biểu diễn dưới dạng một xâu p phân độ dài m;

$x = x_{m-1}x_{m-2}\ldots x_0$, trong đó $x_i \in \{0,1,..,p-1\}$ với $0 \le i < m$.

- Khi $r = 2$:

$x \in GF(2)$ được biểu diễn dưới dạng một số nguyên trong tập $\{0,1\}$.

- Khi $r = 2^m$, trong đó m là một số nguyên lớn hơn 1:

$x \in GF(2^m)$ được biểu diễn bằng một xâu nhị phân có độ dài m;

$x = x_{m-1}x_{m-2} \dots x_0$, trong đó $x_i \in \{0,1\}$ với $0 \le i < m$.

Giá trị đầu ra

- $FE2BS(r,x) = I2BS(g, FE2I(r,x))$,

Trong đó $g = 8\lceil log_{256}(r) \rceil$.

B.4 Biến đổi một xâu nhị phân thành một số nguyên: *BS2I (g,x)*

Các giá trị đầu vào

- g – một số nguyên dương, chính là độ dài của xâu đầu vào.
- $x = x_{g-1}x_{g-2} \dots x_0$ – một xâu nhị phân độ dài g.

Giá trị đầu ra

- $BS2I(g,x) = 2^{g-1}x_{g-1} + 2^{g-2}x_{g-2} + \cdots + x_0 \in \{0,1,\dots,2^g - 1\}$.

B.5 Biến đổi từ một số nguyên thành một xâu nhị phân: *I2BS (g,x)*

Các giá trị đầu vào

- g – một số nguyên dương, chính là độ dài của xâu đầu ra.
- x – một số nguyên trong tập $\{0,1,\dots,2^g - 1\}$.

Giá trị đầu ra

- $I2BS(g,x) = x_{g-1}x_{g-2} \dots x_0$.

Một xâu với các thành phần g (biểu diễn mở rộng của x theo cơ số 2, đệm thêm 0 nếu cần để đạt độ dài mong muốn) được tính toán như sau:

a := x;
i := 0;
While (i < g) do {
  $b \coloneqq \lfloor a/2 \rfloor$;
  $x_i \coloneqq a - (2)(b)$;
  a := b;
  i := i + 1}

B.6 Biến đổi giữa một số nguyên và một xâu octet: $I2OS\ (h,x)$ & $OS2I(h,M)$

I2OS(h,x):

Các giá trị đầu vào

- h – một số nguyên dương, chính là độ dài của xâu octect đầu ra.
- x – một số nguyên trong tập $\{0,1,\dots,256^h - 1\}$.

Giá trị đầu ra

- Tính toán một xâu các số nguyên $x_{h-1}x_{h-2} \dots x_0$, trong đó $x_i \in \{0,1,\dots,255\}$ với $0 \le i < h$, biểu diễn mở rộng của x theo cơ số 256, đệm thêm 0 nếu cần thiết để đạt độ dài h. Các giá trị x_i được tính toán như sau:

a := x;
i := 0;

While (i < h) do {

$b := \lfloor a/256 \rfloor$;

$x_i := a - (256)(b)$;

a := b;

i := i + 1}

- $I2OS(h, x) = M_{h-1}M_{h-2} \dots M_0$,

Trong đó octet M_i tương đương xâu nhị phân độ dài 8 bit $I2BS(8, x_i)$.

OS2I(h,M):

Các giá trị đầu vào

- h - một số nguyên dương, chính là độ dài của xâu octet đầu vào.
- $M = M_{h-1}M_{h-2} \dots M_0$ – một xâu octet độ dài h.

Giả định

- Với $0 \leq i < h, M_i$ được biểu diễn bằng một xâu nhị phân 8 bit.

Giá trị đầu ra

- Tính toán xâu các số nguyên $x_{h-1}x_{h-2} \dots x_0$,

Trong đó $x_i = BS2I(8, M_i) \in \{0,1, \dots, 255\}$ với $0 \leq i < h$.

$OS2I(h, x) = 256^{h-1}x_{h-1} + 256^{h-2}x_{h-2} + \cdots + x_0 \in \{0,1, \dots, 256^h - 1\}$.

Phụ Lục C
(Tham khảo)
Các hàm biến đổi (II)

Phụ lục C đặc tả hàm $I2P$ (Biến đổi số nguyên thành điểm), được sử dụng để mô tả hai cơ chế ký số dựa trên định danh.

Hàm này thỏa mãn các tính chất ngẫu nhiên và một chiều, tức là biến đổi một số nguyên thành một điểm sao cho điểm đó có phân bố ngẫu nhiên trong nhóm đã chọn và với những thông tin cho trước, việc khôi phục lại số nguyên từ điểm là không khả thi về mặt tính toán. Hàm này cũng được sử dụng trong IEEE P1363. Tuy nhiên, chưa có chứng minh nào về độ an toàn của hàm này được chính thức công bố. Vì lý do trên nên hàm này được đưa ra với mục đích tham khảo.

Cho trước một tập các tham số miền của đường cong elliptic (r, q, a_1, a_2), hàm $I2P$ có đầu vào là một số nguyên u và đầu ra là một điểm T bậc q trên đường cong E trên $GF(r)$, ký hiệu là $T = I2P(u)$. Trong phần phụ lục này, phép tính cộng và nhân của các phần tử trên trường hữu hạn được quy định trong ISO/IEC 15946-1 và các phép toán của KDF1 được quy định trong TCVN 11367-2 :2016 (ISO/IEC 18033-2).

a) Đặt $v = BS2I(8\lceil log_{256}(r)\rceil, KDF1_{H2}(I2OS(\lceil log_{256}(u)\rceil, u)$, độ dài tính theo byte của giá trị đặc trưng của $r)) \; mod \; r$. Nếu $v = 0$, đầu ra không hợp lệ và dừng.
b) Đặt $\lambda = u \; mod \; 2$.
c) Nếu r là một số nguyên tố ($r = p$) và đường cong E là $Y^2 = X^3 + a_1X + a_2$ được định nghĩa trên $GF(p)$, tính toán điểm T như sau:
    1) Cho x giá trị bằng v.
    2) Tính toán phần tử trường $c = x^3 + a_1x + a_2 \; mod \; p$. Nếu $c = 0$, đầu ra không hợp lệ và dừng.
    3) Tìm căn bậc hai d của c mô-đun p (tức là số nguyên d với $0 < d < p$ sao cho $d^2 = c \; mod \; p$) hoặc khẳng định rằng không tồn tại căn bậc hai.
        i. Khẳng định tồn tại căn bậc hai, tính toán $\delta = c^{(p-1)/2} \; mod \; p$. Nếu $\delta = 1$, tồn tại d, ngược lại, không tồn tại d.
        ii. Nếu $\delta \neq 1$, tính $u = u + 1$ và quay lại bước a).
        iii. Nếu $\delta = 1$, tìm d.
        Phép tìm hai phần tử d trên trường sao cho $d^2 = c \; mod \; p$ được định nghĩa trong tài liệu viện dẫn [4] và [23]. Để có được kết quả duy nhất, nếu ứng dụng của cơ chế này có một yêu cầu cụ thể về giá trị được chọn, thì thực hiện theo yêu cầu đó. Ngược lại, thì khuyến nghị lấy giá trị tuyệt đối nhỏ nhất theo mô-đun p.
        iv. Đặt $y = (I2FE(r, p-1))^{\lambda} \times d$.
        v. Đặt điểm $T = (x, y)$, tính $T = [\#E/q]T$ và đầu ra là T.
d) Nếu r là lũy thừa của một số nguyên tố lẻ ($r = p^m, p > 2, m \geq 2$) và đường cong E là $Y^2 = x^3 + a_1x^2 + a_2$ (khi $p = 3$) và $Y^2 = x^3 + a_1x + a_2$ (khi $p > 3$) được định nghĩa trên $GF(p^m)$, tính toán điểm T như sau:
    1) Đặt $x = I2FE(r, v)$.
    2) Nếu ($p = 3$), đặt $c = x^3 + a_1x^2 + a_2$ trên $GF(p^m)$. Nếu $c = 0$, đầu ra là không hợp lệ và dừng.

3) Nếu ($p > 3$), đặt $c = x^3 + a_1x + a_2$ trên $GF(p^m)$. Nếu $c = 0$, đầu ra là không hợp lệ và dừng.
4) Tìm một căn bậc hai d của c trên $GF(p^m)$ [tức là một phần tử d trên $GF(p^m)$ thỏa mãn $d^2 = c$ trên $GF(p^m)$] hoặc khẳng định rằng không tồn tại căn bậc hai. Nếu kết quả là không tồn tại căn bậc hai, đặt $u = u + 1$ và quay lại bước a).

CHÚ THÍCH 1 Phép tính để xác định sự tồn tại và tìm một căn bậc hai của một phần tử trên trường được đưa ra trong tài liệu viện dẫn [4] và [23]. Để có được kết quả duy nhất, nếu ứng dụng của cơ chế này có một yêu cầu cụ thể về giá trị được chọn, thì thực hiện theo yêu cầu đó. Ngược lại, so sánh bậc cao nhất của các kết quả và lựa chọn d với bậc nhỏ hơn. Nếu bậc cao nhất của các kết quả là giống nhau, thì lựa chọn d với bậc có hệ số giá trị tuyệt đối nhỏ hơn. Nếu cả bậc và hệ số giống nhau, thì so sánh bậc lớn nhất thứ hai và lựa chọn d với bậc có hệ số giá trị tuyệt đối nhỏ hơn. Lặp lại quá trình này đến khi giá trị d duy nhất được lựa chọn.

5) Đặt $y = (I2FE(r, p - 1))^{\lambda} \times d$.
6) Đặt điểm $T = (x, y)$, tính toán $T = [\#E/q]T$ và đầu ra là T.

e) Nếu r là lũy thừa nguyên tố của 2 ($r = 2^m, m \geq 2$) và đường cong E là $Y^2 + XY = X^3 + a_1X^2 + a_2$ được định nghĩa trên $GF(2^m)$, tính toán điểm T như sau:
1) Đặt $x = I2FE(r, v)$.
2) Đặt $c = x + a_1 + a_2x^{(-2)}$ trong $GF(2^m)$. Nếu $c = 0$, đầu ra không hợp lệ và dừng.
3) Tìm một phần tử d trên trường thỏa mãn $d^2 + d \equiv c$ trong $GF(2^m)$ hoặc khẳng định rằng không tồn tại số nguyên như vậy. Nếu kết quả là không tồn tại số nguyên như vậy, đặt $u = u + 1$ và quay về bước a).

CHÚ THÍCH 2 Các phép tính để xác định sự tồn tại và tìm một phần tử d trên trường sao cho $d^2 + d = c$ trên $GF(2^m)$ được chỉ ra trong tài liệu viện dẫn [4] và [23]. Để có được kết quả duy nhất, nếu ứng dụng của cơ chế này có một yêu cầu cụ thể về giá trị được chọn, thì thực hiện theo yêu cầu đó. Ngược lại, so sánh bậc cao nhất của các kết quả và lựa chọn d với bậc cao nhất nhỏ hơn. Nếu bậc cao nhất của hai giá trị giống nhau, thì so sánh bậc lớn nhất thứ hai và lựa chọn d với bậc lớn nhất thứ hai nhỏ hơn. Lặp lại quá trình này đến khi giá trị d duy nhất được lựa chọn.

4) Đặt $y = (d + I2FE(r, \lambda)) \times x$.
5) Đặt điểm $T = (x, y)$, tính toán $T = [\#E/q]T$ và đầu ra là T.

Phụ lục D
(Quy định)
Sinh các tham số miền DSA

D.1 Sinh số nguyên tố p và q

Lược đồ sinh số nguyên tố bắt đầu bằng sử dụng hàm băm thích hợp và một người dùng đã cung cấp *SEED* để xây dựng một số nguyên tố q trong khoảng $2^{\beta-1} < q < 2^{\beta}$. Khi đó, giá trị SEED giống nhau được sử dụng để xây dựng một giá trị X trong khoảng $2^{\alpha-1} < X < 2^{\alpha}$. Sau đó, số nguyên tố p được tạo thành bằng cách làm tròn X thành một số đồng dư với $1 \bmod 2q$ như được mô tả dưới đây. Các hàm biến đổi giữa số nguyên và chuỗi được quy định trong phụ lục B.

Đặt h là một hàm băm thích hợp cho cặp (α, β) và cho $m(= \gamma)$ là độ dài của khối đầu ra tính bằng bit. Cho $\alpha - 1 = n * m + b$, trong đó b và n là các số nguyên và $0 \leq b < m$.

Bước 1: Chọn một chuỗi tùy ý có ít nhất β bit và gọi là SEED. Cho s là độ dài của SEED tính bằng bit.

Bước 2: Tính toán $U = h(SEED) mod 2^{\beta}$.

Bước 3: Tạo q từ U bằng cách thiết lập bit có trọng số cao nhất ($2^{\beta-1}$ bit) và bit có trọng số thấp nhất bằng 1. Ký hiệu phép tính Boolean như sau $q = U\ OR\ 2^{\beta-1}\ OR\ 1$. Lưu ý rằng $2^{\beta-1} < q < 2^{\beta}$.

Bước 4: Sử dụng thuật toán kiểm tra tính nguyên tố mạnh để kiểm tra xem q có là số nguyên tố hay không (phép kiểm tra tính nguyên tố mạnh là một phép kiểm tra trong đó xác suất để một số không phải số nguyên tố vượt qua nhiều nhất là $2^{-\beta/2}$).

Bước 5: Nếu q không phải số nguyên tố, thì quay về bước 1.

Bước 6: Cho $counter = 0$ và $offset = 1$.

Bước 7: Với $k = 0, \ldots, n$ cho

$Vk = h((SEED + offset + k) mod 2^{s})$.

Bước 8: Cho W là một số nguyên $W = V0 + V1 * 2^{m} + \cdots + Vn - 1 * 2^{(n-1)*m} + (Vn \bmod 2^{b}) * 2^{n*m}$ và cho $X = W + 2^{\alpha-1}$. Lưu ý rằng $0 \leq W < 2^{\alpha-1}$, do đó $2^{\alpha-1} \leq X < 2^{\alpha}$.

Bước 9: Cho $c = X \bmod 2q$ và cho $p = X - (c - 1)$. Lưu ý rằng p đồng dư với $1 \bmod 2q$.

Bước 10: Nếu $p < 2^{\alpha-1}$, thì chuyển sang bước 13.

Bước 11: Thực hiện phép kiểm tra tính nguyên tố mạnh lên p.

Bước 12: Nếu p vượt qua phép kiểm tra trong bước 11, chuyển sang bước 15.

Bước 13: Cho $counter = counter + 1$ và $offset = offset + n + 1$.

Bước 14: Nếu $counter \geq 4\alpha$ quay lại bước 1, ngược lại (tức là nếu $counter < 4\alpha$) quay lại bước 7.

Bước 15: Lưu giá trị counter và tùy chọn giá trị SEED để sử dụng trong chứng nhận quá trình sinh p và q là phù hợp.

CHÚ THÍCH Quá trình tạo này được trích dẫn từ tài liệu viện dẫn [17], phụ lục A.

D.2 Tạo ra phần tử sinh G

D.2.1 Quá trình tạo ra G không có kiểm chứng

Phương pháp này được sử dụng để xác định phần tử sinh G khi không có yêu cầu kiểm chứng giá trị G. Giá trị G được xác định từ p và q.

Bước 1: $e = (p-1)/q$.

Bước 2: Đặt F = số nguyên bất kỳ, trong đó $1 < F < p-1$ và F khác với mọi giá trị đã thử trước đó.

Bước 3: $G = F^e \bmod p$.

Bước 4: Nếu $G = 1$, thì quay lại bước 2.

D.2.2 Quá trình tạo ra G có kiểm chứng

Trong phương pháp này, phần tử sinh G được dựa trên các giá trị của $p, q, index$ và $SEED$. *index* là một xâu 8 bit biểu diễn một số nguyên không dấu. *index* được sử dụng để tạo ra các giá trị G khác nhau từ cùng một cặp (p, q). Giá trị SEED là giá trị cuối cùng được lưu trong thuật toán được mô tả trong D.1. Cho h là hàm băm thích hợp của cặp (α, β). Lưu ý rằng phương pháp này hỗ trợ việc tạo ra nhiều giá trị G với các giá trị p và q cụ thể. Sử dụng các giá trị G khác nhau giúp hỗ trợ việc phân tách khóa bằng cách cung cấp nhiều giá trị *index* khác nhau.

Ở đây, 'ggen' là một xâu byte ASCII với giá trị 0x6767656E và count là một bộ đếm 16 bit (tức là xác định một số nguyên không dấu mô-đun 2^{16}).

Bước 1: $e = (p-1)/q$.

Bước 2: $count = 1$.

Bước 3: $U = SEED||\text{"ggen"}||index||count$.

Bước 4: $W = h(U)$.

Bước 5: $G = W^e \bmod p$.

Bước 6: Nếu $G < 2$, thì tăng giá trị *count* thêm 1 và quay lại bước 3.

Phụ lục E
(Tham khảo)
Các cặp Weil và Tate

E.1 Tổng quan

Cặp Weil và cặp Tate đều là các cặp điểm P, Q của hàm $< P, Q >$ trên một đường cong elliptic E. Các cặp điểm này được sử dụng trong hai cơ chế dựa trên định danh được đặc tả trong Điều 7.

Cho G_1 và G_2 là ký hiệu hai nhóm số nguyên tố bậc q, trong đó G_1 với ký hiệu phép cộng để ký hiệu nhóm các điểm trên đường cong elliptic E và G_2 với ký hiệu phép nhân để ký hiệu một nhóm con của nhóm nhân của một trường hữu hạn.

Một cặp là một ánh xạ song tuyến giữa hai nhóm. Hai cặp được nghiên cứu để sử dụng trong mật mã. Đó là cặp Weil [27][35] cùng một phiên bản sửa đổi [11] và một phiên bản sửa đổi của cặp Tate [18][19]. Trong tiêu chuẩn này, <> ký hiệu một ánh xạ song tuyến, tức là $<>: G_1 \times G_1 \rightarrow G_2$, có thể là cặp Weil sửa đổi hoặc cặp Tate sửa đổi.

Cặp Weil và cặp Tate sửa đổi có hai thuộc tính như sau:

- Song tuyến: Nếu P, P_1, P_2, Q, Q_1, Q_2 là các điểm trong nhóm nguyên tố cyclic bậc q và a thỏa mãn $1 \leq a \leq q-1$;

$< P_1 + P_2, Q >=< P_1, Q > * < P_2, Q >$;

$< P, Q_1 + Q_2 >=< P, Q_1 > * < P, Q_2 >$;

$< [a]P, Q >=< P, [a]Q >=< P, Q >^a$;

- Không suy biến: Nếu P là một điểm không định danh trong nhóm cyclic, thì $< P, P > \neq 1$.

E.2 Các hàm f, g và d

Ba hàm được sử dụng để tính các cặp Weil và Tate như sau:

- Cho E là một đường cong elliptic với công thức $y^2 + a_1 * x * y + a_3 * y = x^3 + a_2 * x^2 + a_4 * x + a_6$.

- Cho trước ba điểm hữu hạn $(x_0, y_0), (x_1, y_1), (u, v)$ trên E, xác định hàm $f((x_0, y_0), (x_1, y_1), (u, v))$ như sau:

Nếu $(x_0, y_0) = 0_E$ và $(x_1, y_1) = 0_E$	thì	$f = 1$
còn nếu $(x_0, y_0) = 0_E$	thì	$f = u - x_1$
còn nếu $(x_1, y_1) = 0_E$	thì	$f = u - x_0$
còn nếu $x_0 \neq x_1$	thì	$f = (x_1 - x_0) * v - (y_1 - y_0) * u - x_1 * y_0 + x_0 * y_1$
còn nếu $y_0 \neq y_1$	thì	$f = u - x_0$

Ngược lại

$$f = (a_1 * y_0 - 3 * x_0^2 - 2 * a_2 * x_0 - a_4) * (u - x_0) + (2 * y_0 + a_1 * x_0 + a_3) * (v - y_0)$$
$$= -(v - y_0)^2 - (u - x_0) * (a_1 * (v - y_0) - (u - x_0) * (2 * x_0 + a_2 + u))$$

- Cho trước các điểm A, B, C trên E, cho $g(A, B, C) = f(A, B, C)/f(A + B, -(A + B), C)$.

CHÚ THÍCH Phụ thuộc vào giá trị r được định nghĩa trong mục 4 và đường cong E, hàm f trên có thể được đơn giản hóa. Sau đây là một vài ví dụ được sử dụng rộng rãi, nhưng không bao gồm mọi trường hợp có thể. Nếu r là một số nguyên tố ($r = p$) và đường cong E là $y^2 = x^3 + a_4x + a_6$ được định nghĩa trên $GF(p)$, hoặc nếu r là lũy thừa của một số nguyên tố lẻ ($r = p^m, p > 3, m \geq 2$) và đường cong E là $y^2 = x^3 + a_4x + a_6$ được định nghĩa trên $GF(p^m)$, hàm f được viết như sau $f = (-3 * x_0^2 - a_4) * (u - x_0) + 2 * y_0 * (v - y_0) = -(v - y_0)^2 + (u - x_0)^2 * (2 * x_0 + u)$. Nếu r là lũy thừa bậc nguyên tố của 2 ($r = 2^m, m \geq 2$) và đường cong E là $y^2 + x * y = x^3 + a_2 * x^2 + a_6$ được định nghĩa trên $GF(2^m)$, hàm f được viết như sau $f = (y_0 + x_0^2) * (u + x_0) + x_0 * (v + y_0) = (v + y_0^2) + (u + x_0) * (v + y_0 + (u + x_0) * (u + a_2))$. Nếu r là lũy thừa của 3 ($r = 3^m, m \geq 2$) và đường cong E là $Y^2 = x^3 + a_2x^2 + a_6$ được định nghĩa trên $GF(3^m)$, hàm f được viết như sau $f = 2 * y_0 * (v - y_0) - 2 * a_2 * x_0 * (u - x_0)$.

- Cho trước hai điểm D và C trên E và một số nguyên $l > 2$, hàm Weil $d(D, C, l)$ được tính toán theo thuật toán sau:

a) Đặt $A = D, f = 1$. Cho $l = (n_t, \ldots, n_0)$ là biểu diễn bit của l sao cho $l = \sum_i n_i 2^i$ và $n_t \neq 0$.
b) Với $i = t - 1, t - 2, \ldots, 0$ thực hiện {

$f = f * f * g(A, A, C)$;
$A = A + A$;
Nếu $n_i \neq 0$ thì {
$f = f * g(A, D, C)$;
$A = A + D$;
}
}

c) Đặt $d(D, C, l) = f$ và đầu ra là $d(D, C, l)$.

Bỏ qua tham số l, $d(D, C, l)$ được ký hiệu là $d(D, C)$.

E.3 Cặp Weil

Cho $l > 2$ là số nguyên tố và cho P và Q là các điểm trên E với $[l]P = [l]Q = 0_E$, cặp Weil $< P, Q >$ được tính toán như sau:

- Chọn một số điểm T ngẫu nhiên trên E (sao cho $0_E, Q, T, P + T$ đều khác nhau), thì

- Tính $< P, Q \geq ((d(P, Q - T)/d(P, -T))/(d(Q, P + T)/d(Q, T)))$.

Nếu trong quá trình tính toán cặp điểm, xuất hiện phép chia cho 0 thì phải bắt đầu lại quá trình tính toán với một điểm T mới.

E.4 Cặp Tate

Cho $l > 2$ là số nguyên tố và cho P và Q là các điểm trên E với $[l]P = 0_E$, cặp Tate $< P, Q >$ được tính toán như sau:

- Chọn một số điểm T ngẫu nhiên trên E, thì

- Tính $< P,Q >= (d(P,Q-T)/d(P,-T))$.

Nếu trong quá trình tính toán cặp điểm, xuất hiện phép chia cho 0 thì phải bắt đầu lại quá trình tính toán với một điểm T mới.

CHÚ THÍCH Thông tin chi tiết hơn về việc xây dựng cặp Weil và Tate có trong tài liệu viện dẫn [8], [21] và [30].

E.5 Cặp Tate rút gọn

Cho $l > 2$ là số nguyên tố và cho P và Q là các điểm trên E với $[l]P = O$, cặp $< P,Q >$ được tính toán như sau:

- Chọn một số điểm T ngẫu nhiên trên E.

- Tính $< P,Q >= (d(P,Q-T)/d(P,-T))^{(p^k-1)/l}$;

Trong đó k là kích thước của trường mở rộng (ví dụ: $k = 2$) và p^k là số lượng các phần tử trong trường mở rộng. Nếu trong quá trình tính toán cặp điểm, có phép chia cho 0 thì phải bắt đầu lại quá trình tính toán với một điểm T mới.

CHÚ THÍCH 1 Thông tin chi tiết hơn về việc xây dựng các cặp điểm có trong tài liệu viện dẫn [4] và [23].

CHÚ THÍCH 2 Cặp Tate rút gọn được sử dụng trong các ví dụ số học trong F.11 và F.12.

Phụ lục F
(Tham khảo)
Các ví dụ số

F.1 Tổng quan

Trong phụ lục F, tất cả các kí tự đều được biểu diễn dưới dạng thập lục phân, trừ khi được nêu rõ ràng.

Một khuyến cáo rằng chỉ nên sử dụng chữ ký số dựa trên SHA-1 và RIPEMD-160 cho các ứng dụng kế thừa.

F.2 Cơ chế DSA

F.2.1 Ví dụ 1: 2014 bit, số nguyên tố *P*, SHA-224

F.2.1.1 Tổng quan

Một lời giải thích đầy đủ về việc tạo ra tất cả các giá trị được đưa ra trong Tài liệu tham khảo [17]. Ví dụ này là giá trị mẫu cho DSA với α = 2048 và β= 224. Tất cả việc băm, bao gồm việc sinh các tham số miền, đều thực hiện với SHA-224.

F.2.1.2 Các tham số

α = 2048

β = 224

SEED = 0C088E11 2F88B186 90421876 5614496E C2AF9770 C71D0A56 87F489B6

F = 2

P = B4865EFC 44BFB4CB 7EE034F0 EAE8A72D 25897819 9BF9BA28 8462FD97
19F33272 C010A11B 33BCE4E8 481B6EC7 AB1229D9 FC7BEA43 8055907F
F1E28FAC 33716089 DCED277F 9036440A 887D4B22 CAC5BABD ECD6A1B3
A1731594 20371025 BAAB5F18 D5FDE928 CE4F5EE4 5352785F 20057782
2C20756E 171CBDD8 1CEB932A E0F29109 5CFFD9C2 3A07AC6B C2F5250B
B9F8E2E6 5AF85215 6E8EEBF8 31C098FB 010057BD 425132B8 0A46BB5C
E801E241 05058E58 091383F1 6F124894 FB6DE9CD 3BCC4C6E 64901743
AF8F47C3 5CC2177E B15ED172 B4969174 FE3F645A 9D3BEFC6 811A9074
BF702024 98E5E157 ECDBED3C 1FDF3C4F 00DAB43A CBA49802 79392E18
B515851F

Q = B4D0963C 40D74138 69F42710 BBEF73CB C6C1C4E6 35C6B9F3 CF7A6255

Counter = 24

G = A92434D5 6752B028 CF11954E 0F3B1BED 8804EB74 8DEED793 E2932E80
8F37C34A 15444A06 9A8B17E5 4BF7FB82 7D6FE959 428BA0CC 1F3B2B8E
EA0A25A2 CAF73A0C 68C7DC48 093374A3 CD1F2250 8EF05038 9E8AE58C
E6A8AD50 2510B4CA C42528B7 BCA0993C C959C630 61D7BA3A 885E9C6D
CA6EAE44 E2D3C050 A236645F FBDE4BA6 1ECEB17B 941F85E9 C5234A28
FAD461DE 8B55F033 DB7E0CB4 DA5E115F FFCD416D 5A8BC9CD 9DAA6816
010841CC 9F416A6F E109A40A 823874F0 EDD92F45 738918AC 0CB925E7
AB8E692A 9336DB36 697E6C75 5B0243CA EBB61A38 79EABAF6 AC53F166
2740D6ED 3E3DB9BF A629390A 6A517FB0 B50D02E2 57178145 AF964626
57ABA465

F.2.1.3 Khóa ký và khóa kiểm tra

X = A279D0A3 A4243A2B 16909C9E 0BBFEC32 0589E4DF 1BDDAE72 3BA7353B

Y = 31246FA1 CB8D1430 BDCDEBF0 5BB8C967 D24E6728 BA5C900C 50852741
3AFD496A F12EA9CC D80D8916 62A7B9B3 C2023212 08943D85 5D7EA110
B9512D1B 9E4AABAB 72B99005 25127129 EAB2CC8E 66B6E09C 49341ABF
184B2733 9114E39E FED6B90B 8D7BA182 3E3512D3 EB82F720 76C2815D
A642DE61 D808DCF0 22A76077 1E22AA42 26997E41 EA142BAD BFD00011
F7D27677 08A0313E 42255286 0D184F18 C4890ED3 A6CE8134 E1647DDC
B292B5FD 5C5ED61C 1BF9567A E1E40CC5 F85F5B7D 1A09AAA1 08CFCFE2
469360A9 48F61B4D 1CDCA791 1BB64070 94D9A78B A34ED943 97057791
DFC56691 1B4F7DD9 61A7EBB8 74923C59 2458D43D F171CB81 698AB7EE
2E9B92E6

F.2.1.4 Dữ liệu cho mỗi thông điệp

M= ASCII form of "abc" = 61 62 63

K= 2973C724 7F9BD6DB 3C08CD7A 1DA427DF 6780A7DD F3E09362 E8BA1293

$h(M)$= 23097D22 3405D822 8642A477 BDA255B3 2AADBCE4 BDA0B3F7 E36C9DA7

F.2.1.5 Ký

R = 1DFAAA6F 87DA6148 6529A2F3 4EBC7D89 3D42F405 F8DCBB33 93CC1A00

S= 4A3E6377 D09A4CD6 67BA9F9C E3982EB9 C1AA6E90 70F7C2F7 0EA23173

F.2.1.6 Kiểm tra

R' = 1DFAAA6F 87DA6148 6529A2F3 4EBC7D89 3D42F405 F8DCBB33 93CC1A00

F.2.2 Ví dụ 2: 3072 bit, số nguyên tố *P*, SHA-256

F.2.2.1 Tổng quan

Ví dụ này là giá trị mẫu cho DSA với $\alpha = 3072$ và $\beta = 256$. Tất cả việc băm, bao gồm việc tạo ra tham số miền, đều thực hiện với SHA-256.

F.2.2.2 Các tham số

α = 3072

β = 256

SEED = 193AFCA7 C1E77B3C 1ECC618C 81322E47 B8B8B997 C9C83515 C59CC446
C2D9BD47

F = 2

P = 90066455 B5CFC38F 9CAA4A48 B4281F29 2C260FEE F01FD610 37E56258
A7795A1C 7AD46076 982CE6BB 956936C6 AB4DCFE0 5E678458 6940CA54
4B9B2140 E1EB523F 009D20A7 E7880E4E 5BFA690F 1B9004A2 7811CD99
04AF7042 0EEFD6EA 11EF7DA1 29F58835 FF56B89F AA637BC9 AC2EFAAB
90340222 9F491D8D 3485261C D068699B 6BA58A1D DBBEF6DB 51E8FE34
E8A78E54 2D7BA351 C21EA8D8 F1D29F5D 5D159394 87E27F44 16B0CA63
2C59EFD1 B1EB6651 1A5A0FBF 615B766C 5862D0BD 8A3FE7A0 E0DA0FB2
FE1FCB19 E8F9996A 8EA0FCCD E5381752 38FC8B0E E6F29AF7 F642773E
BE8CD540 2415A014 51A84047 6B2FCEB0 E388D30D 4B376C37 FE401C2A
2C2F941D AD179C54 0C1C8CE0 30D460C4 D983BE9A B0B20F69 144C1AE1
3F9383EA 1C08504F B0BF3215 03EFE434 88310DD8 DC77EC5B 8349B8BF
E97C2C56 0EA878DE 87C11E3D 597F1FEA 742D73EE C7F37BE4 3949EF1A
0D15C3F3 E3FC0A83 35617055 AC91328E C22B50FC 15B941D3 D1624CD8
8BC25F3E 941FDDC6 20068958 1BFEC416 B4B2CB73

Q = CFA0478A 54717B08 CE64805B 76E5B142 49A77A48 38469DF7 F7DC987E
FCCFB11D

Counter = 20

G = 5E5CBA99 2E0A680D 885EB903 AEA78E4A 45A46910 3D448EDE 3B7ACCC5
4D521E37 F84A4BDD 5B06B097 0CC2D2B8 B715F7B8 2846F9A0 C393914C

792E6A92 3E2117AB 805276A9 75AADB52 61D91673 EA9AAFFE ECBFA618
3DFCB5D3 B7332AA1 9275AFA1 F8EC0B60 FB6F66CC 23AE4870 791D5982
AAD1AA94 85FD8F4A 60126FEB 2CF05DB9 A7F0F09B 3397F393 7F2E90B9
E5B9C9B6 EFEF642B C48351C4 6FB171B9 BFA9EF17 A961CE96 C7E7A7CC
3D3D03DF AD1078BA 21DA4251 98F07D24 81622BCE 45969D9C 4D6063D7
2AB7A0F0 8B2F49A7 CC6AF335 E08C4720 E31476B6 7299E231 F8BD90B3
9AC3AE3B E0C6B6CA CEF8289A 2E2873D5 8E51E029 CAFBD55E 6841489A
B66B5B4B 9BA6E2F7 84660896 AFF387D9 2844CCB8 B6947549 6DE19DA2
E58259B0 90489AC8 E62363CD F82CFD8E F2A427AB CD65750B 506F56DD
E3B98856 7A88126B 914D7828 E2B63A6D 7ED0747E C59E0E0A 23CE7D8A
74C1D2C2 A7AFB6A2 9799620F 00E11C33 787F7DED 3B30E1A2 2D09F1FB
DA1ABBBF BF25CAE0 5A13F812 E34563F9 9410E73B

F.2.2.3 Khóa ký và khóa kiểm tra

X = 3ABC1587 297CE7B9 EA1AD665 1CF2BC4D 7F92ED25 CABC8553 F567D1B4
= 0EBB8764

Y = 8B891C86 92D3DE87 5879390F 2698B26F BECCA6B0 75535DCE 6B0C8625
77F9FA0D EF6074E7 A7624121 224A5958 96ABD4CD A56B2CEF B942E025
D2A4282F FAA98A48 CDB47E1A 6FCB5CFB 393EF35A F9DF9131 02BB303C
2B5C36C3 F8FC04ED 7B8B69FE FE0CF3E1 FC05CFA7 13B3435B 2656E913
BA8874AE A9F93600 6AEB448B CD005D18 EC3562A3 3D04CF25 C8D3D698
44343442 FA3DB7DE 618C5E2D A064573E 61E6D558 1BFB694A 23AC87FD
5B52D62E 954E1376 DB8DDB52 4FFC0D46 9DF97879 2EE44173 8E5DB05A
7DC43E94 C11A2E7A 4FBE3830 71FA36D2 A7EC8A93 88FE1C4F 79888A99
D3B61056 97C2556B 79BB4D7E 781CEBB3 D4866AD8 25A5E830 84607228
9FDBC941 FA679CA8 2F5F78B7 461B2404 DB883D21 5F4E0676 CF549395
0AC55916 97BFEA8D 1EE6EC01 6B89BA51 CAFB5F9C 84C989FA 117375E9
4578F28B 3ABC1587 297CE7B9 E0B34CE0 545DA462 66FD77F6 2D8F2CEE
92AB7701 2AFEBC11 008985A8 21CD2D97 8C7E6FE7 499D1AAF 8DE632C2
1BB48CA5 CBF9F310 98FD3FD3 854C49A6 5D920174 4AACE540 354974F9

F.2.2.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "abc" = 61 62 63

K = A6902C1E 6E3943C5 62806158 8A8B007B CCEA91DB F1291548 3F04B24A B0678BEE

$h(M)$ = BA7816BF 8F01CFEA 414140DE 5DAE2223 B00361A3 96177A9C B410FF61 F20015AD

F.2.2.5 Ký

R = 5F184E64 5A38BE8F B4A6871B 6503A9D1 2924C7AB E04B7141 0066C2EC A6E3BE3E

S = 91EB0C7B A3D4B9B6 0B825C3D 9F2CADA8 A2C9D772 3267B033 CBCDCF88 03DB9C18

F.2.2.6 Kiểm tra

R' = 5F184E64 5A38BE8F B4A6871B 6503A9D1 2924C7AB E04B7141 0066C2EC A6E3BE3E

F.3 Cơ chế KCDSA

F.3.1 Ví dụ 1 : Số nguyên tố *P* 2048 bit, số nguyên tố *Q* 224 bit, SHA-224

F.3.1.1 Tổng quan

Ví dụ này sử dụng SHA-224 như hàm băm h. Mã băm chỉ đơn giản là giá trị của SHA-224.

F.3.1.2 Các tham số

l = 200 (i.e. 512 in decimal)

α = 2048

β = 224

P = 8DA8C1B5 C95D11BE 46661DF5 8C9F803E B729B800 DD92751B 3A4F10C6 A5448E9F 3BC0E916 F042E399 B34AF9BE E582CCFC 3FF5000C FF235694 94351CFE A5529EA3 47DCF43F 302F5894 380709EA 2E1C416B 51A5CDFC 7593B18B 7E3788D5 1B9CC9AE 828B4F8F B06E0E90 57F7FA0F 93BB0397 031FE7D5 0A6828DA 0C1160A0 E66D4E5D 2A18AD17 A811E70B 14F4F431 1A028260 3233444F 98763C5A 1E829C76 4CF36ADB 56980BD4 C54BBE29

7E790228 4292D75C A3600FF4 59310B09 291CBEFB C721528A
13403B8B 93B711C3 03A2182B 6E6397E0 83380BF2 886AF3B9
AFCC9F50 55D8B713 6C0EBD08 C5CF0B38 888CD115 72787F6D
F384C97C 91B58C31 DEE5655E CBF3FA53

Q = 864F1884 1EC103CD FD1BE7FE E54650F2 2A3BB997 537F32CC
79A51F53

G = 0E9BE1F8 7A414D16 7A9A5A96 8B079E4A D385A357 3EDB21AA
67A6F61C 0D00C14A 7A225044 B6E9EB03 68C1EB57 B243B5CD
854FD93C 1B2DFB0A 3EA302D2 367E4EC7 2F6E7EE8 EA7F8002
F7704E99 0B954F25 BADA8DA6 2BAEB6F0 6953C0C8 5104AD03
F36618F7 6C62F4EC F3480183 69850A56 17C999DB E68BA17D
5BC72556 74EF4839 22C6A3F9 9D3C3C6F 358896C4 E63C605E
E7DB16FC BD9BE354 E281F7FE 7813D054 27ED1912 B5C7653A
167B9434 9147EEAF 85CC9CE2 E81661F3 21512D5D 2C0580B0
3D1704EE F2317F45 185C8258 387E7EC9 79C04707 EF546241
2784AFE4 1A7B45C8 3B9CBE48 F9127CB4 400BE9E9 6AC5DE17
F2C9DEA3 5E3734E7 9B64673F 85681C4E

F.3.1.3 Khóa ký và khóa kiểm tra

X = 2F1991C1 AF401872 8A5A431B 9B5459DF B16F6D25 6797FE57
0EC6BC65

Y = 04EDE5C6 7EA29297 A8CACB6B DE6F4666 AEA27D1D 3DD1E9E9
582F76A2 F22B8B1B 32230BC5 8F06B768 F8102B49 FA1CAE5E
18921494 7F6239B6 C6CE7C9B C2D230E8 9A40BEE2 C33A8861
FD4F7D35 B788FE95 B2D5885D 8C8FAEA8 1C90BE4C EE2784E3
3577A71D 3B7F085D 71E9A1D4 7815C73F A087ACAA B9FCB565
5AC9570E 6852BE7C 9C0AECEA 8BD9AA75 A44FC314 7F733E90
6ADB0FD7 6D613561 B1DB364B BDC9AFD3 CE8F5F17 E3E71203
4A999350 8059FA52 441FA90D DFE9A0F2 A0B9192F E2220C08
1BD0C0F0 E07CB5F1 EE4FF405 23591F17 8A4FC7CB 5065F6A3
8216E9A0 99C205B2 9B8746D8 65E1AF6D 903E5A13 8004910B
70EB5B84 EED9760E A60578BF 08852898

F.3.1.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "This is a test message for KCDSA usage!" =
54 68 69 73 20 69 73 20 61 20 74 65 73 74 20 6D
65 73 73 61 67 65 20 66 6F 72 20 4B 43 44 53 41
20 75 73 61 67 65 21

K = 49561994 FD2BAD5E 410CA1C1 5C3FD3F1 2E70263F 2820AD5C
566DED80

Y' = 1BD0C0F0 E07CB5F1 EE4FF405 23591F17 8A4FC7CB 5065F6A3
8216E9A0 99C205B2 9B8746D8 65E1AF6D 903E5A13 8004910B
70EB5884 EED9760E A60578BF 08852898

$h(Y'||M)$ = B3F921C1 8DDD06B6 09D02BD5 916F180E 5DFB19C8 9FEC063D
059A3575

V = 5E4E4BEC B42ED14C 1F00A98C D077A8C1 D65E6F5A 50D7ACD1
6AF7EC24

F.3.1.5 Ký

R = EDB76A2D 39F3D7FA 16D08259 4118B0CF 8BA57692 CF3BAAEC
6F6DD951

S = 5260A2DF 2E923DE8 77B130AC 8B5E8B17 63973B88 D5D4627A
DFBACF52

F.3.1.6 Kiểm tra

R' = EDB76A2D 39F3D7FA 16D08259 4118B0CF 8BA57692 CF3BAAEC
6F6DD951

F.3.2 Ví dụ 2 : Số nguyên tố *P* 3072 bit, số nguyên tố *Q* 256 bit, SHA-256

F.3.2.1 Tổng quan

Ví dụ này sử dụng SHA-256 như hàm băm *h*. Mã băm chỉ đơn giản là giá trị của SHA-256.

F.3.2.2 Các tham số

l = 200 (i.e. 512 in decimal)

α = 3072

β = 256

P = CBAEACE3 677E98AD B2E49C00 2B8B0F43 4143B466 515839BF

813B097D 2D1EE681 5008C27A 3415BC22 31609874 5E5844F3
3ECC8887 C16DFB1C FB77DC4C 3F3571CC EEFD4291 8F6C48C3
702AB6EF 0919B7E8 402FC89B 35D09A0E 5040E309 1EE4674B
E891933C 1007E017 EDD40818 7E4114B6 BE5548D7 8DB58B84
8475A422 62D7EB79 5F08D161 1055EFEA 8A6AEB20 EB0F1C22
F002A2E8 195BCBBA 830B8461 3531BDD9 EC71E5A9 7A9DCCC6
5D6117B8 5D0CA66C 3FDAA347 6E97ADCD 05A1F490 2BD04B92
F400C42B A0C9940A 32600443 3B6D3001 28BF930F 484EAA63
02CD7A31 9EE5E561 A12A3625 594020C2 40DBA3BE BD8A4751
5841F198 EBE43218 2639616F 6A7F9BD7 434F0534 8F7F1DB3
115A9FEE BA984A2B 73784334 DE7737EE 3704535F CA2F4904
CB4AD58F 172F2648 E1D62D05 8539AC78 3D032D18 33D2B9AA
D96982C9 692E0DDB B6615508 83ED66F7 AA8BCE8F F0663A0A
D0A226C7 BD0E06DF C72594A3 87C676A3 CA06A300 62BE1D85
F23E3E02 C4D65E06 1B619B04 E83A318E C55ECA06 9EB85603

Q = C2A8CAF4 87180079 66F2EC13 4EABA3CB B07F31A8 F2667ACB
5D9B872F A760A401

G = 17A1C167 AF836CC8 5149BE43 63F1BB4F 0010848F C9B678B4
E026F1F3 87133749 A4B1BBA4 C23252A4 C86F31E2 1E8ACACB
4E33AD89 B7C3D79A 5409268B FBA82B45 814E4352 0C09D631
613FA35D B9CAF18F 791C2729 A4B014BC 79A85A90 CD541037
119ECCDE 0778863F FCB9C259 31FCD33A 6706E5FE 1F495BB8
BCB3D0EE C9B6D5A9 373127A2 121E37D9 8A840330 258DBFCE
E7E06F81 5B69C16C 5D17289C 4CC37E71 9B856298 D4E1574E
4F4F8515 BAF9A850 D11DDA09 55BC30FA 5B16792D 673A3B1F
41512FC3 EB89452D 51509F97 4D878B48 2D2AD2ED 32BE1905
6F574504 2BFF804F B7482796 612B746F E8D70A83 8CC6F496
DD0FFC3D 95C1E0B1 98184D73 523656A0 6431BC52 5C2BC161
9729E8C0 88F6DF91 5645E060 922A4AF3 EDD63047 C7B6077C
667C07D8 8EB00F4C FE59D32E 5F545012 C566516B 7874FB3D

AED51403 31F29528 B30FC8B8 A9371C28 18017B09 53A84FFC

9FBFF84B 64BF0238 AA7E2AF2 ECADC15A 1C06DADC F1F2E7B1

240A5E64 5A6469C9 B002215D 9A91C2A4 ED2FB547 A942D777

F.3.2.3 Khóa ký và khóa kiểm tra

X = 7C28569A 94B46FA7 45C8D306 AD7DC189 96CE046E EBE04383

8391C232 078DB05A

Y = 2574E10E 806F1C42 58F7CF8F A4A6CF2B EB177DBE 60E4EC17

DF21DCDB A72073F6 5565506D A3DF98D5 A6C8EEE6 1B6B5D88

B98C47C2 B2F6FC6F 504FA4FB C7F411E2 3EAA3B18 7A353DAE

D41533A9 558AB932 0A154CAE CC544E43 0008889A 2C899373

EC75A24C FF26247C F297D293 747ECC05 B3483647 A87BCBB8

D4500092 09F5E449 A00A659B 637CE139 CF6487AC A70F9C00

CB670C7F 3B95BFD7 CF236A0A 6F3C93BE 8D9CF591 C9D30686

9415B1AA 97264B90 4167850A 4794C780 BE4527DF FEB67BE6

E66786C5 CCE0378C CB49920D 855558F4 DAC4C42F 92DD229B

483B2257 DB0CE35D C737F980 1A261A02 BDF718C2 FD4D69C5

2E0D9712 B42C4897 BAE7C684 D3D35BC5 726CE899 2696B044

D722AFBA 78EFA858 C4D10F19 72112CE8 FFD39792 49BF14E4

9D8E0D9A CB1B0A9C A90D0551 1803845D 7C670BCF 1B066497

A7743B08 A219E764 EA0A3A2A 617661C1 6A372FE0 58B547A2

8B626ECF 442222E1 8EEF487C C101DBFB 715BC33A B85928EC

F0BD4DEA 30F250A6 A5C86178 83EA0F87 3E7A4651 98C4644B

F.3.2.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "This is a test message for KCDSA usage!" =

54 68 69 73 20 69 73 20 61 20 74 65 73 74 20 6D

65 73 73 61 67 65 20 66 6F 72 20 4B 43 44 53 41

20 75 73 61 67 65 21

K = 83F3008F CEBAE57E C7A64A3A F7EE6EE1 9CC197A6 D5EBA3A5

B3EF79B2 F8F3DD53

Y' = EA0A3A2A 617661C1 6A372FE0 58B547A2 8B626ECF 442222E1

8EEF487C C101DBFB 715BC33A B85928EC F0BD4DEA 30F250A6
A5C86178 83EA0F87 3E7A4651 98C4644B

$h(Y'||M)$ = 4D4F2A9B 83446B62 F571A669 FACB2D30 7ADE18DE 1A3FFB87
649ABA4E 606A0751

V = 1935B399 849AB60F 0AE62FAD 82B281E9 1A098A8F 51E6E7D6
BA581801 F02604A0

F.3.2.5 Ký

R = 547A9902 07DEDD6D FF9789C4 7879ACD9 60D79251 4BD91C51
DEC2A24F 904C03F1

S = 1668797B 26641E72 94AA68D3 8562EAE3 CAA842D0 F446949C
4268AE3D 0392434F

F.3.2.6 Kiểm tra

R' = 547A9902 07DEDD6D FF9789C4 7879ACD9 60D79251 4BD91C51
DEC2A24F 904C03F1

F.3.3 Ví dụ 3: Số nguyên tố *P* 2048 bit, số nguyên tố *Q* 224 bit, SHA-256

F.3.3.1 Tổng quan

Ví dụ này sử dụng SHA-256 như hàm băm *h*. Mã băm chỉ đơn giản là giá trị của SHA-256.

F.3.3.2 Các tham số

l = 200 (i.e. 512 in decimal)

α = 2048

β = 224

P = 8DA8C1B5 C95D11BE 46661DF5 8C9F803E B729B800 DD92751B
3A4F10C6 A5448E9F 3BC0E916 F042E399 B34AF9BE E582CCFC
3FF5000C FF235694 94351CFE A5529EA3 47DCF43F 302F5894
380709EA 2E1C416B 51A5CDFC 7593B10B 7E3788D5 1B9CC9AE
828B4F8F B06E0E90 57F7FA0F 93BB0397 031FE7D5 0A6828DA
0C1160A0 E66D4E5D 2A18AD17 A811E70B 14F4F431 1A028260
3233444F 98763C5A 1E829C76 4CF36ADB 56980BD4 C54BBE29
7E790228 4292D75C A3600FF4 59310B09 291CBEFB C721528A

13403B8B 93B711C3 03A2182B 6E6397E0 83380BF2 886AF3B9
AFCC9F50 55D8B713 6C0EBD08 C5CF0B38 888CD115 72787F6D
F384C97C 91B58C31 DEE5655E CBF3FA53

Q = 864F1884 1EC103CD FD1BE7FE E54650F2 2A3BB997 537F32CC
79A51F53

G = 0E9BE1F8 7A414D16 7A9A5A96 8B079E4A D385A357 3EDB21AA
67A6F61C 0D00C14A 7A225044 B6E9EB03 68C1EB57 B24B45CD
854FD93C 1B2DFB0A 3EA302D2 367E4EC7 2F6E7EE8 EA7F8002
F7704E99 0B954F25 BADA8DA6 2BAEB6F0 6953C0C8 5104AD03
F36618F7 6C62F4EC F3480183 69850A56 17C999DB E68BA17D
5BC72556 74EF4839 22C6A3F9 9D3C3C6F 358896C4 E63C605E
E7DB16FC BD9BE354 E281F7FE 7813D054 27ED1912 B5C7653A
167B9434 9147EEAF 85CC9CE2 E81661F3 21512D5D 2C0580B0
3D1704EE F2317F45 185C8258 387E7EC9 79C04707 EF546241
2784AFE4 1A7B45C8 3B9CBE48 F9127CB4 400BE9E9 6AC5DE17
F2C9DEA3 5E3734E7 9B64673F 85681C4E

F.3.3.3 Khóa ký và khóa kiểm tra

X = 2F1991C1 AF401872 8A5A431B 9B5459DF B16F6D25 6797FE57
0EC6BC65

Y = 04EDE5C6 7EA29297 A8CACB6B DE6F4666 AEA27D10 3DD1E9E9
582F76A2 F22B8B1B 32230BC5 8F06B768 F8102B49 FA1CAE5E
18921494 7F6239B6 C6CE7C9B C2D230E8 9A40BEE2 C33A8861
FD4F7D35 B788FE95 B2D5885D 8C8FAEA8 1C90BE4C EE2784E3
3577A71D 3B7F085D 71E9A1D4 7815C73F A087ACAA B9FCB565
5AC9570E 6852BE7C 9C0AECEA 8BD9AA75 A44FC314 7F733E90
6ADB0FD7 6D613561 B1DB364B BDC9AFD3 CE8F5F17 E3E71203
4A999350 8059FA52 441FA90D DFE9A0F2 A0B9192F E2220C08
1BD0C0F0 E07C85F1 EE4FF405 23591F17 8A4FC7CB 5065F6A3
8216E9A0 99C205B2 9B8746D8 65E1AF6D 903E5A13 80049108
70EB5B84 EED9760E A60570BF 08852898

F.3.3.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "This is a test message for KCDSA usage!" =

54 68 69 73 20 69 73 20 61 20 74 65 73 74 20 6D

65 73 73 61 67 65 20 66 6F 72 20 4B 43 44 53 41

20 75 73 61 67 65 21

K = 49561994 FD2BAD5E 410CA1C1 5C3FD3F1 2E70263F 2820AD5C

566DED80

Y' = 1BD0C0F0 E07CB5F1 EE4FF405 23591F17 8A4FC7CB 5065F6A3

8216E9A0 99C205B2 9B8746D8 65E1AF6D 903E5A13 8004910B

70EB5B84 EED9760E A60578BF 08852898

$I2BS(\beta, BS2I(\gamma, h(Y'||M_2)) \bmod 2^\beta)$ =

894315A9 A68EF862 7D015AAE 4EBB41B3 FED88AAA 9614CC67

09DE2B47

V = 489142E8 F4A1ACE1 4F693AFD 632A6FB7 5A8392AD E61F8A26

3F665A1B

F.3.3.5 Ký

R = C1D25741 522F5483 32686053 2D912E04 A45B1807 700B4641

36B8715C

S = 0AFAEA63 92942822 86B0F9E1 9EC1BC13 BFA29B54 5747F262

0DA3AFC1

F.3.3.6 Kiểm tra

R' = C1D25741 522F5483 32686053 2D912E04 A45B1807 700B4641

36B8715C

F.4 Cơ chế Pointcheval-Vaudenay

F.4.1.1 Tổng quan

Ví dụ này sử dụng SHA-224 như hàm băm *h*. Mã băm chỉ đơn giản là giá trị của SHA-224.

F.4.1.2 Các tham số

L = 200 (i.e. 512 in decimal)

F = 2

P = EBADFD4F 9CC4EFF7 F5B3F5D9 047399E1 00E6D01D 6DE30E40
39B25F76 083743F1 826919FE D5C750F5 BC6AD9C1 C0B9E229
695F306C 1AE9F631 7EC4AC94 74FD88F9 714798E6 2FD2D865
A2EE7BB4 E6C38A8F 2A4C1C30 1FEC7149 1356AD5A 87449AFF
F1AE412B 128ECF26 DB9DBD74 C91CBDCA 4ED4CE4F 43AC2770
E7B38648 6190B6A4 0211E61F F5F767C2 31F9FDEF FE999F47
BA2C2111 821C3EE5 BDD18BC2 E4C89E5A 5C54DC16 0167A8D6
B0235223 44B45E72 AB2A54A7 A3787C59 6A38AA76 3589852C
50D67BFB 469278F0 7E6F5C03 2785FA22 2E635460 65E5E1D5
95BE7E94 F01C274C B3291116 40544989 38824CC9 55D802CE
76C0FD6E 33537111 78C33059 6F25CF05

Q = C3E87841 9B4463CE 1FBA73B6 022498A0 EA51A6B8 050E79BB
A5FEF99D

G = BC376597 C582A659 072A5F1A 2839A1DF 594B0AE2 1262A447
A8A0F6CB 60596837 661C1734 DFC2BB0B 9F756AA3 44AD8ADD
A15193A9 6CA727E3 A9D8F32E B20E9760 178862DF C154A308
2EAD2E4A 2F7C760A 61E86750 D8B3CFDC 3D00A46A E787A3DE
5F657ADA 64CD0BFF A3B2B228 88299DE3 100F58B7 CD593D82
E5B31614 997FF1CA 6E87790A 3C8B551B 82606A1C BC80ACD2
37683EE2 427BA87C 06FFC4EC 17504261 E99DE627 8F9EF77B
A6E49A82 9C9F27E9 87812CD2 8A15EEAA 36391E67 3042E8A2
9C76C8A4 DABFDB3D 48D351F5 8B870F9F 609AE941 FFA9AA2D
886045F9 5EE9B836 9E1C9545 BC18D60C AE7F56AE 8A24B387
395F0CFC 99E36A65 AFB8C269 522352A0

F.4.1.3 Khóa ký và khóa kiểm tra

X = 26534136 393884D5 885F924C 188C83AC 860A2560 17B323B4
118E1B1A

Y = D88E9870 78469EB9 B6296190 64BF87C8 1F5BFFD3 88C03AB8
C6C2048E 5B8D087C 8C1D7A5F E2D3394E F567CA43 8AC89B5A

27F26DF2 CCBFDB4A 4351E2BA 42866D82 72DB1BB2 422157D5

3659BB68 FE50F61C 31573684 E93B19C5 F3F832C5 01B6CD9F

3F0E40FA 9198D358 F2BD39F4 DDC4CC5E BD670CB1 677F4F8F

D0867B95 D9AB1A8E 2A8377A9 8189FEF4 0B241503 2F7EAD42

BBD61A86 C6150431 87614940 435E55DB 55CA0F37 C6F06142

0577EB51 196E0021 431C94DB D746800A F1B4F23E 86F40B57

AB964428 872EE96D A99E3576 0613D279 2FC3E8A8 0CDD1877

117B9ED5 401511ED 43AC10BF 1E1F7450 01AB51CF 842EF938

386947FE BD4CBE9B 1FC3D785 C937D5CE

F.4.1.4 Dữ liệu cho mỗi thông điệp

K = 717548D2 2C7FAF7E BE1BD5DC AD2C2E47 5DA789D5 68C5E081

236214D0

K^{-1} = 787CEDE9 BD149C67 45C83654 E1F9B472 C736D584 ECD4ACE4

4943FD61

M = ASCII form of "abc" = 61 62 63

F.4.1.5 Ký

R = AF246E82 74F24075 343014D5 FC648CE0 09771BD0 48DF1438

EB0D97E7

$R||M$ = AF246E82 74F24075 343014D5 FC648CE0 09771BD0 48DF1438

EB0D97E7 616263

$h(R||M)$ = 5986F5F E1D8BF68 2BD8FE31 25065C4C D9A77CDD 73EE5E85

01D29EF2

S = 370F4FD9 3E761FD9 B97DC37D 4DAD75E2 4EF3A6FD 3D5AC9F1

867A7E33

F.4.1.6 Kiểm tra

Π' = 18A358C2 2635ABE3 85E18D55 AF94B75B 36EC2FAE 8E87EA24

C0BBC507 A8790BA9 0CD0A93D C92E4736 E5A050BD DA2BF62E

AA042C69 07D39346 D909753C 4C91024F DC12D5AA 5A98FE7E

EE3A90AD 64DB83E9 7C91EAC5 CDC41AA0 D383B3BA 094B0A58

C64BF470 0FC81437 BAD1A15D 3FD61B53 25E2991E D841D159

EEF740DE C7984657 D4B0BE24 9DB622E1 0936BE55 2AE4005A

59DFA542 A688F0DE 3C5D67EB C5A37ED4 C5A3D0F1 CAF1B8FC

24E7E658 368FF8C1 E10A3F80 BE42B30C 6255B2AE 15F0251C

CF1CD9F5 41ECCCF7 7E6A6147 B79FE310 09C8172D BD0CD110

52816FA7 04FADDA3 5C73E38F 9C6BD8BA 131A2073 191D9D64

94A3E764 4ED385FF E01D85BD A015FB6C

R' = AF246E82 74F24075 343014D5 FC648CE0 09771BD0 48DF1438

EB0D97E7

F.5 Cơ chế SDSA

F.5.1 Ví dụ 1: số nguyên tố *P* 2048 bit, SHA-224

F.5.1.1 Tổng quan

Đối với ví dụ này, nhóm 'MODP 2048-bit với nhóm con chính 224-bit trong RFC-5114 được sử dụng như nhóm. SHA-224 được sử dụng riêng cho hàm băm, do đó mã băm chỉ đơn giản là giá trị SHA-224, được chuyển đổi theo Phụ lục B cho các mục dữ liệu thích hợp.

F.5.1.2 Các tham số

α = 2048

β = 224

p = AD107E1E 9123A9D0 D660FAA7 9559C51F A20D64E5 683B9FD1

B54B1597 B61D0A75 E6FA141D F95A56DB AF9A3C40 7BA1DF15

EB3D688A 309C180E 1DE6B85A 1274A0A6 6D3F8152 AD6AC212

9037C9ED EFDA4DF8 D91E8FEF 55B7394B 7AD5B7D0 B6C12207

C9F98D11 ED34DBF6 C6BA0B2C 8BBC27BE 6A00E0A0 B9C49708

B3BF8A31 70918836 81286130 BC8985DB 1602E714 415D9330

278273C7 DE31EFDC 7310F712 1FD5A074 15987D9A DC0A486D

CDF93ACC 44328387 315D75E1 98C641A4 80CD86A1 B9E587E8

BE60E69C C928B2B9 C52172E4 13042E9B 23F10B0E 16E79763

C9B53DCF 4BA80A29 E3FB73C1 6B8E75B9 7EF363E2 FFA31F71

CF9DE538 4E71B81C 0AC4DFFE 0C10E64F

G = AC4032EF 4F2D9AE3 9DF30B5C 8FFDAC50 6CDEBE7B 89998CAF

74866A08 CFE4FFE3 A6824A4E 10B9A6F0 DD921F01 A70C4AFA

AB739D77 00C29F52 C57DB17C 620A8652 BE5E9001 A8D66AD7

C1766910 1999024A F4D02727 5AC1348B B8A762D0 521BC98A
E2471504 22EA1ED4 09939D54 DA7460CD B5F6C6B2 50717CBE
F180EB34 118E98D1 19529A45 D6F83456 6E3025E3 16A330EF
BB77A86F 0C1AB15B 051AE3D4 28C8F8AC B70A8137 150B8EEB
10E183ED D19963DD D9E263E4 770589EF 6AA21E7F 5F2FF381
B539CCE3 409D13CD 566AFBB4 8D6C0191 81E1BCFE 94B30269
EDFE72FE 9B6AA4BD 7B5A0F1C 71CFFF4C 19C418E1 F6EC0179
81BC087F 2A7065B3 84B890D3 191F2BFA

q = 801C0D34 C58D93FE 99717710 1F80535A 4738CEBC BF389A99
B36371EB

F.5.1.3 Khóa ký và khóa kiểm tra

X = 602FE736 80BEFCB2 A8B46779 35FF652B 21A3F4DE 46725D07
D7D371A9

Y = A7DBB446 FD8C4B82 61BE026D 94FA9847 74B17110 CABC0944
14AD2013 2EFC8B7D 7C7FF05D D0B902C4 EF736831 61C1F9A3
9D60E7AB E3BD9FE2 B458A96D F4783408 0AA93CAF 09673967
F434548D 44B278E0 4C1FA6D5 E0C41990 CEF37940 66015ED4
748DAD56 429596DD 9259C45C 21B71A5E A4EF099A 06DAC737
8958A107 B11B3E57 384118E0 19897C48 E734F069 E717E23A
DD202405 823A2AE7 A08AFA51 09D2CF6A 0FF546FA 38A8735D
1CE715E0 6AC08EB0 93EB331F EBEC88D6 1DF546E2 DC9E8465
10B63F6A 5BA73FE3 6995BD17 1B9D7D35 9EEB3D7D B801F382
1D582280 DF71A27D 5191E4AB 42BE27A3 1180F537 CABAC0D1
2EBF1698 B1884697 9EDCA15D DC8F860C

F.5.1.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "abc" = 616263

K = 7BFA2DD5 6B31BB27 FFC0D1AE 1ABAA90F A0BB9379 08A542A1
5EFD1E15

Π = 60FCB613 40799851 B5E2DC3A 3865BC21 29100D38 4B1C9A94
6F0C873B 442BEBD8 5904CD09 A4C6A29E 0CD1111E B9E65F82

85F8A578 A5717098 FA2A601F D9183CDD D5FF1586 AB255E1D
4DF4A141 DFE717DC 16DA3B0D 438B1EA5 4976523F 1D73351B
F39B1987 97DA0EC7 E9EE994A 4C0352D8 271D186A 0DEA8AB0
FD5E7862 17016E91 03C5F139 2C1D3C01 B974BADC 88184905
065F8DA8 55656BAF B3B1EDBC 4C14A969 2AEA1A71 D85117F4
08548EF5 A34966B9 0123FC81 72472B44 06D0C2E6 77C3C21D
D0680C63 0DC69BFF BA67D89F F17CA52A 6B0F164F 5452777D
B838BDBD EE60E03B AE773475 42435BD9 09D021DD F97602E0
3FE41463 CC18128B AFA6E661 6F6CA744

R = CC192C12 72872367 48346281 4DF721E5 D9B2A651 0D97F3D3
316AD681

F.5.1.5 Ký

R = CC192C12 72872367 48346281 4DF721E5 D9B2A651 0D97F3D3
316AD681

S = 4C776699 9F9D52E1 52B47F29 335E548F A3B90625 C55FC9A4
FE2D5F1F

F.5.1.6 Kiểm tra

Π' = 60FCB613 40799851 B5E2DC3A 3865BC21 29100D38 4B1C9A94
6F0C873B 442BEBD8 5904CD09 A4C6A29E 0CD1111E B9E65F82
85F8A578 A5717098 FA2A601F D9183CDD D5FF1586 AB255E1D
4DF4A141 DFE717DC 16DA3B0D 438B1EA5 4976523F 1D73351B
F39B1987 97DA0EC7 E9EE994A 4C0352D8 271D186A 0DEA8AB0
FD5E7862 17016E91 03C5F139 2C1D3C01 B974BADC 88184905
065F8DA8 55656BAF B3B1EDBC 4C14A969 2AEA1A71 D85117F4
08548EF5 A34966B9 0123FC81 72472B44 06D0C2E6 77C3C21D
D0680C63 0DC69BFF BA67D89F F17CA52A 6B0F164F 5452777D
B838BDBD EE60E03B AE773475 42435BD9 09D021DD F97602E0
3FE41463 CC18128B AFA6E661 6F6CA744

R' = CC192C12 72872367 48346281 4DF721E5 D9B2A651 0D97F3D3
316AD681

F.5.2 Ví dụ 2: Nguyên tố P 2048 bit, SHA-256

F.5.2.1 Tổng quan

Đối với ví dụ này, nhóm 'MODP 2048-bit với nhóm con chính 256-bit trong RFC-5114 được sử dụng như nhóm. SHA-256 được sử dụng riêng cho hàm băm, do đó mã băm chỉ đơn giản là giá trị SHA-256, được chuyển đổi theo Phụ lục B cho các mục dữ liệu thích hợp.

F.5.2.2 Các tham số

α = 2048

β = 256

p = 87A8E61D B4B6663C FFBBD19C 65195999 8CEEF608 660DD0F2
5D2CEED4 435E3B00 E00DF8F1 D61957D4 FAF7DF45 61B2AA30
16C3D911 34096FAA 3BF4296D 830E9A7C 209E0C64 97517ABD
5A8A9D30 6BCF67ED 91F9E672 5B4758C0 22E0B1EF 4275BF7B
6C5BFC11 D45F9088 B941F54E B1E59BB8 BC39A0BF 12307F5C
4FDB70C5 81B23F76 B63ACAE1 CAA6B790 2D525267 35488A0E
F13C6D9A 51BFA4AB 3AD83477 96524D8E F6A167B5 A41825D9
67E144E5 14056425 1CCACB83 E6B486F6 B3CA3F79 71506026
C0B857F6 89962856 DED4010A BD0BE621 C3A3960A 54E710C3
75F26375 D7014103 A4B54330 C198AF12 6116D227 6E11715F
693877FA D7EF09CA DB094AE9 1E1A1597

G = 3FB32C9B 73134D0B 2E775066 60EDBD48 4CA7B18F 21EF2054
07F4793A 1A0BA125 10DBC150 77BE463F FF4FED4A AC0BB555
BE3A6C1B 0C6B47B1 BC3773BF 7E8C6F62 901228F8 C28CBB18
A55AE313 41000A65 0196F931 C77A57F2 DDF463E5 E9EC144B
777DE62A AAB8A862 8AC376D2 82D6ED38 64E67982 428EBC83
1D14348F 6F2F9193 B5045AF2 767164E1 DFC967C1 FB3F2E55
A4BD1BFF E83B9C80 D052B985 D182EA0A DB2A3B73 13D3FE14
C8484B1E 052588B9 B7D2BBD2 DF016199 ECD06E15 57CD0915
B3353BBB 64E0EC37 7FD02837 0DF92B52 C7891428 CDC67EB6
184B523D 1DB246C3 2F630784 90F00EF8 D647D148 D4795451
5E2327CF EF98C582 664B4C0F 6CC41659

q = 8CF83642 A709A097 B4479976 40129DA2 99B1A47D 1EB3750B
A308B0FE 64F5FBD3

F.5.2.3 Khóa ký và khóa kiểm tra

X = 73018895 20D47AA0 55995BA1 D8FCD701 6EA62E09 18892E07
B7DC23AF 69006B88

Y = 57A17258 D4A3F47C 4545AD51 F3109C5D B41B7878 79FCFE53
8DC1DD5D 35CE42FF 3A9F225E DE650212 6408FCB1 3AEA2231
80B149C4 64E176EB F03BA651 0D8206C9 20F6B1E0 9392E6C8
40A05BDB 9D6875AB 3F4817EC 3A65A665 B788ECBB 447188C7
DF2EB4D3 D9424E57 D964398D BE1C6362 659C6BD8 55C1D3E5
1D64796C A59848OD FDD9580E 55085345 C15E34D6 A33A2F43
E222407A CE058972 D34952AE 2B705C53 2243BE39 4B222329
6161145E F2927CDB C55BBD56 4AAE8DE4 BA4500A7 FA432FE7
8B0F0689 1E408083 7E761057 BC6CB8AC 18FD4320 7582032A
FB63C624 F32E66B0 5FC31C5D FFB25FA9 2D4D00E2 B0D4F721
E88C417D 2E57797B 8F55A2FF C6EE4DDB

F.5.2.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "abc" = 616263

K = 2B73E8FF 3A7C0168 6CA556E0 FABFD74A C8D1FDA4 AD3D503F
23B8EB8A EEC63305

Π = 41979DBA 19606871 A25BBB51 665AD584 9511D125 8C077E93
027D79AC 35EE887C 460C2689 3D7FFC59 0F317B7A 2C01FCB4
3667E373 F8F2D239 0C950BAF 972E6E0B 00A79416 9696CC95
08A895D6 3F8AA7EA 564AA5DE 6104920A 5E9F687F 469BC831
0C02BE30 B8FD4A54 A167E114 01FEE171 EF284811 6146CC69
C0899526 7186C43E 98B5E62E 9CE5C344 646950EF F57B1490
27FE078F CD9C8297 4AFF1DDE 5AF18E4C A3E3787F 66D15D23
292B5F4E 225AAC64 FC904AB3 40A88B76 40F2D436 D2840185
077EACA4 EE75113E 95B26149 F7C6D2CD 554463F9 26E48F09
AD3C99B8 5EA3EC39 E795FEAE C90C8293 FB0D0506 0FE2BF91
5F00E2FB 7C17B2E8 7C462ED0 D49B8E2F

R = CDAC932A 758FCFCE 7E549903 FD891F41 FB5410CB DDD246F3
D6DB0CE6 E0ED696E

F.5.2.5 Ký

R = CDAC932A 758FCFCE 7E549903 FD891F41 FB5410CB DDD246F3
D6DB0CE6 E0ED696E

S = 3505AEA2 E039E18F DDC6580A E89E15DF 0103FB45 C1BB763E
DA4EE6F5 F01783CE

F.5.2.6 Kiểm tra

Π' = 41979DBA 19606871 A25BBB51 665AD584 9511D125 8C077E93
027D79AC 35EE887C 460C2689 3D7FFC59 0F317B7A 2C01FCB4
3667E373 F8F2D239 0C950BAF 972E6E0B 00A79416 9696CC95
08A895D6 3F8AA7EA 564AA5DE 6104920A 5E9F687F 469BC831
0C02BE30 B8FD4A54 A167E114 01FEE171 EF284811 6146CC69
C0899526 7186C43E 98B5E62E 9CE5C344 646950EF F57B1490
27FE078F CD9C8297 4AFF10DE 5AF18E4C A3E3787F 66D15D23
292B5F4E 225AAC64 FC904AB3 40A88B76 40F2D436 D2840185
077EACA4 EE75113E 95B26149 F7C6D2CD 554463F9 26E48F09
AD3C99B8 5EA3EC39 E795FEAE C90C8293 FB0D0506 0FE2BF91
5F00E2FB 7C17B2E8 7C462ED0 D49B8E2F

R' = CDAC932A 758FCFCE 7E549903 FD891F41 FB5410CB DDD246F3
D6DB0CE6 E0ED696E

F.6 Cơ chế EC-DSA

F.6.1 Tổng quan

Với các ví dụ dưới đây, SHA-1 được sử dụng riêng cho hàm băm, do đó mã băm chỉ đơn giản là giá trị của SHA-1, được chuyển đổi theo Phụ lục B cho các mục dữ liệu thích hợp.

Từ quan điểm an toàn, điều quan trọng là phải tránh các đường cong yếu về thuật toán mật mã (ví dụ: Đảm bảo rằng một đường cong cụ thể không dễ bị tấn công vào các trường hợp đặc biệt của đường cong elliptic Logarit rời rạc).

F.6.2 Ví dụ 1: Trường F_2^m, với m = 191, SHA-1

F.6.2.1 Các tham số

Trường F_2^m được mô tả bởi đa thức modulo rút gọn $x^{191} + x^9 + 1$.

Đường con elliptic: $Y^2 + XY = X^3 + aX^2 + b$ trên trường F_2^m.

a = 28665378 67675263 6A68F565 54E12640 276B649E F7526267

b = 2E45EF57 1F00786F 67B0081B 9495A3D9 5462F5DE 0AA185EC

$G = (G_X, G_Y)$

G_X = 36B3DAF8 A23206F9 C4F299D7 B21A9C36 9137F2C8 4AE1AA0D

G_Y = 765BE734 33B3F95E 332932E7 0EA245CA 2418EA0E F98018FB

Q = 40000000 00000000 00000000 04A20E90 C39067C8 93BBB9A5

F.6.2.2 Khóa ký và khóa kiểm tra

X = 340562E1 DDA332F9 D2AEC168 249B5696 EE39D0ED 4D03760F

$Y = (Y_X, Y_Y)$

Y_X = 5DE37E75 6BD55D72 E3768CB3 96FFEB96 2614dEA4 CE28A2E7

Y_Y = 55C0E0E0 2F5FB132 CAF416EF 85B229BB B8E13520 03125BA1

F.6.2.3 Dữ liệu cho mỗi thông điệp

M = ASCII form of "abc" = 61 62 63

K = 3EEACE72 B4919D99 1738D521 879F787C B590AFF8 189D2B69

$\Pi = (\Pi_X, \Pi_Y)$.

Π_X = 438E5A11 FB55E4C6 5471DCD4 9E266142 A3BDF2BF 9D5772D5

Π_Y = 2AD603A0 5BD1D177 649F9167 E6F475B7 E2FF590C 85AF15DA

$h(M)$ = A9993E36 4706816A BA3E2571 7850C26C 9CD0D89D

F.6.2.4 Ký

R = 038E5A11 FB55E4C6 5471DCD4 998452B1 E02D8AF7 099BB930

S = 0C9A08C3 4468C244 B4E5D6B2 1B3C6836 28074160 20328B6E

F.6.2.5 Kiểm tra

$\Pi' = (\Pi'_X, \Pi'_Y)$

Π'_X = 438E5A11 FB55E4C6 5471DCD4 9E266142 A3BDF2BF 9D5772D5

Π'_Y = 2AD603A0 5BD1D177 649F9167 E6F475b7 E2FF590C 85AF15DA

R' = 038E5A11 FB55E4C6 5471DCD4 998452B1 E02D8AF7 099BB930

F.6.3 Ví dụ 2: Trường F_P, số nguyên tố P 192 bit, SHA-1

F.6.3.1 Các tham số

Trường F_P với P là hệ thập lục phân

P = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFFFF FFFFFFFF

Đường cong elliptic: $Y^2 + XY = X^3 + aX^2 + b$ trên trường F_P

a = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFFFF FFFFFFFC

b = 64210519 E59C80E7 0FA7E9AB 72243049 FEB8DEEC C146B9B1

G = (G_X, G_Y)

G_X = 188DA80E B03090F6 7CBF20EB 43A18800 F4FF0AFD 82FF1012

G_Y = 07192B95 FFC8DA78 631011ED 6B24CDD5 73F977A1 1E794811

Q = FFFFFFFF FFFFFFFF FFFFFFFF 99DEF836 146BC9B1 B4D22831

F.6.3.2 Khóa ký và khóa kiểm tra

X = 1A8D598F C15BF0FD 89030B5C B1111AEB 92AE8BAF 5EA475FB

Y_X = (Y_X, Y_Y)

Y_X = 62B12D60 690CDCF3 30BABAB6 E69763B4 71F994DD 702D16A5

Y_Y = 63BF5EC0 8069705F FFF65E5C A5C0D697 16DFCB34 74373902

F.6.3.3 Dữ liệu cho mỗi thông điệp

M = ASCII form of "abc" = 616263

$h(M)$ = A9993E36 4706816A BA3E2571 7850C26C 9CD0D89D

K = FA6DE297 46BBEB7F 8BB1E761 F85F7DFB 2983169D 82FA2F4E

Π = (Π_X, Π_Y).

Π_X = 88505238 0FF147B7 34C330C4 3D39B2C4 A89F29B0 F749FEAD

Π_Y = 9CF9FA1C BEFEFB91 7747A3BB 29C072B9 289C2547 884FD835.

F.6.3.4 Ký

R = 88505238 0FF147B7 34C330C4 3D39B2C4 A89F29B0 F749FEAD

S = E9ECC781 06DEF82B F1070CF1 D4D804C3 CB390046 951DF686

F.6.3.5 Kiểm tra

$$\Pi' = \left(\Pi'_X, \Pi'_Y\right).$$

Π'_X = 88505238 0FF147B7 34C330C4 3D39B2C4 A89F29B0 F749FEAD

Π'_Y = 9CF9FA1C BEFEFB91 7747A3BB 29C072B9 289C2547 884FD835

R' = 88505238 0FF147B7 34C330C4 3D39B2C4 A89F29B0 F749FEAD

F.6.4 Ví dụ 3: Trường F_2^m, với m = 283, SHA-256

F.6.4.1 Các tham số

Trường F_2^m được mô tả bởi đa thức modulo rút gọn $x^{283} + x^{12} + x^7 + x^5 + 1$.

Đường con elliptic: $Y^2 + XY = X^3 + aX^2 + b$ trên trường F_2^m.

a = 1

b = 027B680A C8B8596D A5A4AF8A 19A0303F CA97FD76 45309FA2
A5B1485A F6263E31 3B79A2F5

G = (G_X, G_Y)

G_X = 05F93925 8DB7DD90 E1934F8C 70B0DFEC 2EED25B8 557EAC9C
80E2E198 F8CDBECD 86B12053

G_Y = 03676854 FE24141C B98FE6D4 B20D02B4 516FF702 350EDDB0
826779C8 13F0DF45 BE8112F4

Q = 03FFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFEF90 399660FC
938A9016 5B042A7C EFADB307

F.6.4.2 Khóa ký và khóa kiểm tra

X = 010652D3 7B0A9DB6 4D4033AC 6549CD1D F37E1EED E2612C23
63257C6A FF6C8CB5 DCB63648

Y_X = (Y_X, Y_Y)

Y_X = 0390858E 9327A714 C74AF0C3 ADEDF4E6 C75CAFDC C46507A4
9E415B13 8A094B6F 43E882AC

Y_Y = 00D4A65D 973CD150 A5221BED F872A4BA 207FF442 7DFFFD48
27C5BF16 9E719162 504D0631

F.6.4.3 Dữ liệu cho mỗi thông điệp

M = ASCII form of "Example of ECDSA with B-283"

$h(M)$ = F0BF4AEF 3F694EBD DE0A7944 5C897ADB 2430B918 77C772DA
9B7362CB 03AEA87F

K = 0100EC32 1393E6DD 6C4D47BE 5AE189E5 E3540857 9D086217
8F94CCBB A3C4049A 4D88E297

$\Pi = (\Pi_X, \Pi_Y)$.

Π_X = 077CB284 AC41E72E DA2A93EB 8D6DFF58 620F6C69 D528DFE9
0D909AA5 CABC03A3 4E5D5A76

F.6.4.4 Ký

R = 037CB284 AC41E72E DA2A93EB 8D6DFF58 620F7CD9 9B927EEC
7A060A8F 6FB7D926 5EAFA76F

S = 00A37AC1 0AEBFC22 FC6E6EE2 2E8F235E 3EEB0555 A0F0F9DA
92D9FFA7 34AD7679 56D27F23

F.6.4.5 Kiểm tra

$\Pi' = (\Pi'_X, \Pi'_Y)$

Π'_X = 077CB284 AC41E72E DA2A93EB 8D6DFF58 620F6C69 D528DFE9
0D909AA5 CABC03A3 4E5D5A76

R' = 037CB284 AC41E72E DA2A93EB 8D6DFF58 620F7CD9 9B927EEC
7A060A8F 6FB7D926 5EAFA76F

F.6.5 Ví dụ 4: Trường F_P, số nguyên tố P 256 bit, SHA-256

F.6.5.1 Các tham số

Trường F_P với P là hệ thập lục phân

P = FFFFFFFF 00000001 00000000 00000000 00000000 FFFFFFFF FFFFFFFF
FFFFFFFF

Đường cong elliptic: $Y^2 = X^3 + aX + b$ trên trường F_P

a = FFFFFFFF 00000001 00000000 00000000 00000000 FFFFFFFF
FFFFFFFF FFFFFFFC

B = 5AC635D8 AA3A93E7 B3EBBD55 769886BC 651D06B0 CC53B0F6

3BCE3C3E 27D2604B

G = (G_X, G_Y)

G_X = 6B17D1F2 E12C4247 F8BCE6E5 63A440F2 77037D81 2DEB33A0 F4A13945 D898C296

G_Y = 4FE342E2 FE1A7F9B 8EE7EB4A 7C0F9E16 2BCE3357 6B315ECE CBB64068 37BF51F5

Q = FFFFFFFF 00000000 FFFFFFFF FFFFFFFF BCE6FAAD A7179E84 F3B9CAC2 FC632551

F.6.5.2 Khóa ký và khóa kiểm tra

X = C477F9F6 5C22CCE2 0657FAA5 B2D1D812 2336F851 A508A1ED 04E479C3 4985BF96

Y = (Y_X, Y_Y)

Y_X = B7E08AFD FE94BAD3 F1DC8C73 4798BA1C 62B3A0AD 1E9EA2A3 8201CD08 89BC7A19

Y_Y = 3603F747 959DBF7A 4BB226E4 19287290 63ADC7AE 43529E61 B563BBC6 06CC5E09

F.6.5.3 Dữ liệu cho mỗi thông điệp

M = ASCII form of "Example of ECDSA with P-256"

$h(M)$ = A41A41A1 2A799548 211C410C 65D8133A FDE34D28 BDD542E4 B680CF28 99C8A8C4

K = 7A1A7E52 797FC8CA AA435D2A 4DACE391 58504BF2 04FBE19F 14DBB427 FAEE50AE

Π = (Π_X, Π_Y).

Π_X = 2B42F576 D07F4165 FF65D1F3 B1500F81 E44C316F 1F0B3EF5 7325B69A CA46104F

F.6.5.4 Ký

R = 2B42F576 D07F4165 FF65D1F3 B1500F81 E44C316F 1F0B3EF5 7325B69A CA46104F

S = DC42C212 2D6392CD 3E3A993A 89502A81 98C1886F E69D262C 4B329BDB 6B63FAF1

F.6.5.5 Kiểm tra

$$\Pi' = (\Pi'_X, \Pi'_Y)$$

Π'_X = 2B42F576 D07F4165 FF65D1F3 B1500F81 E44C316F 1F0B3EF5
7325B69A CA46104F

R' = 2B42F576 D07F4165 FF65D1F3 B1500F81 E44C316F 1F0B3EF5
7325B69A CA46104F

F.7 Cơ chế EC-KCDSA

F.7.1 Ví dụ 1: Trường F_P, số nguyên tố P 224 bit, SHA-224

F.7.1.1 Tổng quan

Ví dụ này sử dụng SHA-224 như hàm băm. Mã băm chỉ đơn giản là giá trị của SHA-224.

F.7.1.2 Các tham số

Trường F_P với P là

P = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF 00000000 00000000
00000001

Đường cong elliptic: $Y^2 = X^3 + aX + b$ trên trường F_P

a = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFFFF FFFFFFFF
FFFFFFFE

b = B4050A85 0C04B3AB F5413256 5044B0B7 D7BFD8BA 270B3943
2355FFB4

G = (G_X, G_Y)

G_X = B70E0CBD 6BB4BF7F 321390B9 4A03C1D3 56C21122 343280D6
115C1D21

G_Y = BD376388 B5F723FB 4C22DFE6 CD4375A0 5A074764 44D58199
85007E34

Q = FFFFFFFF FFFFFFFF FFFFFFFF FFFF16A2 E0B8F03E 13DD2945
5C5C2A3D

F.7.1.3 Khóa ký và khóa kiểm tra

X = 562A6F84 E162FFCB 51CD4707 774AE366 81B6CEF2 05FE5D43
912956A2

Y = (Y_X, Y_Y)

Y_X = B574169E 4FCEF1AF 3429D8BB 5481FF7D FA978690 492E1098
B80A5579

Y_Y = 1576819B D9F0B685 19EE844A FE88CCFB 2AD574A5 6472D954
1461AE7E

F.7.1.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "This is a sample message for EC-KCDSA implementation validation." =
54 68 69 73 20 69 73 20 61 20 73 61 6D 70 6C 65 20 6D
65 73 73 61 67 65 20 66 6F 72 20 45 43 2D 4B 43 44 53
41 20 69 6D 70 6C 65 6D 65 6E 74 61 74 69 6F 6E 20 76
61 6C 69 64 61 74 69 6F 6E 2E

K = 76A0AFC1 8646D1B6 20A079FB 223865A7 BCB447F3 C03A35D8
78EA4CDA

Π = $G^k = (\Pi_X, \Pi_Y)$.

Π_X = F807C158 65203FF7 2C69C113 A457DF64 4F627801 DFF99D1B
CC849C2D

Π_Y = ADE18B5B B7118745 017631E5 E54B36C0 332D70B3 CAA8FB10
728B66E0

Y' = B574169E 4FCEF1AF 3429D8BB 5481FF7D FA978690 492E1098
B80A5579 1576819B D9F0B685 19EE844A FE88CCFB 2AD574A5
6472D954 1461AE7E 00000000 00000000

$h(Y'||M)$ = 8C5CB967 71166477 FF84D281 DB766201 2F842138 8AA6FC05
282E2E03

F.7.1.5 Ký

R = 8EA58C91 E0CDCEB5 799B00D2 412D928F DD23122A 1C2BDF43
C2F8DAFA

S = AE8AB53C 7A44A8B2 2F35FDB9 DE265F23 B89F65A6 9A8B7BD4
061911A6

F.7.1.6 Kiểm tra

R' = EEA58C91 E0CDCEB5 799B00D2 412D928F DD23122A 1C2BDF43 C2F8DAFA

F.7.2 Ví dụ 2: Trường F_P, số nguyên tố P 256 bit, SHA-256

F.7.2.1 Tổng quan

Ví dụ này sử dụng SHA-256 như hàm băm. Mã băm chỉ đơn giản là giá trị của SHA-256.

F.7.1.2 Các tham số

Trường F_P với P là

P = FFFFFFFF 00000001 00000000 00000000 00000000 FFFFFFFF FFFFFFFF FFFFFFFF

Đường cong elliptic: $Y^2 = X^3 + aX + b$ trên trường F_P

a = FFFFFFFF 00000001 00000000 00000000 00000000 FFFFFFFF FFFFFFFF FFFFFFFC

b = 5AC635D8 AA3A93E7 B3EBBD55 769886BC 651D06B0 CC53B0F6 3BCE3C3E 27D2604B

G = (G_X, G_Y)

G_X = 6B17D1F2 E12C4247 F8BCE6E5 63A440F2 77037D81 2DEB33A0 F4A13945 D898C296

G_Y = 4FE342E2 FE1A7F9B 8EE7EB4A 7C0F9E16 2BCE3357 6B315ECE CBB64068 37BF51F5

Q = FFFFFFFF 00000000 FFFFFFFF FFFFFFFF BCE6FAAD A7179E84 F3B9CAC2 FC632551

F.7.2.3 Khóa ký và khóa kiểm tra

X = 9051A275 AA4D9843 9EDDED13 FA1C6CBB CCE775D8 CC9433DE E69C5984 8B3594DF

Y = (Y_X, Y_Y)

Y_X = 148EDDD3 734FD5F1 5987579F 516089A8 C9FEF4AB 76B59D7B 8A01CDC5 6C4EDFDF

Y_Y = A4E2E42C B4372A6F 2F3F71A1 49481549 F68D2963 539C853E 46B94696 569E8D61

F.7.2.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "This is a sample message for EC-KCDSA implementation validation." =

54 68 69 73 20 69 73 20 61 20 73 61 6D 70 6C 65 20 6D

65 73 73 61 67 65 20 66 6F 72 20 45 43 2D 4B 43 44 53

41 20 69 6D 70 6C 65 6D 65 6E 74 61 74 69 6F 6E 20 76

61 6C 69 64 61 74 69 6F 6E 2E

K = 71B88F39 8916DA9C 90F555F1 B5732B7D C636B49C 638150BA
C11BF05C FE16596A

Π = $G^k = (\Pi_X, \Pi_Y)$.

Π_X = EC3847B0 CA52038A 823D0230 14546B41 4946EF0A 6EE09228
30948459 5F30E26C

Π_Y = 0640451D 36932442 4ABC681D 65653986 6AD9C494 D26FAC14
69FC2A08 D945F130

Y' = 148EDDD3 734FD5F1 5987579F 516089A8 C9FEF4AB 76B59D7B
8A01CDC5 6C4EDFDF A4E2E42C B4372A6F 2F3F71A1 49481549
F68D2963 539C853E 46B94696 569E8D61

$h(Y'||M)$ = 681C8ED8 9E8B0E1B C369AA10 6F6B9813 E6338F0C 54BE577A
87623492 52F9BEDF

F.7.2.5 Ký

R = 0EDDF680 601266EE 1DA83E55 A6D9445F C781DAEB 14C765E7
E5D0CDBA F1F14A68

S = 9B333457 661C7CF7 41BDDBC0 835553DF BB37EE74 F53DB699
E0A17780 C7B6F1D0

F.7.2.6 Kiểm tra

R' = 0EDDF680 601266EE 1DA83E55 A6D9445F C781DAEB 14C765E7
E5D0CDBA F1F14A68

F.7.3 Ví dụ 3: Trường F_2^m, m = 233, SHA-224

F.7.3.1 Tổng quan

Ví dụ này sử dụng SHA-224 như hàm băm. Mã băm chỉ đơn giản là giá trị của SHA-224.

F.7.3.2 Các tham số

Trường F_2^m được mô tả bởi đa thức modulo rút gọn $x^{233} + x^{74} + 1$.

Đường cong elliptic: $Y^2 + XY = X^3 + aX^2 + b$ trên trường F_2^m.

a = 1

b = 0066 647EDE6C 332C7F8C 0923BB58 213B333B 20E9CE42
81FE115F 7D8F90AD

G = (G_X, G_Y)

G_X = 00FA C9DFCBAC 8313BB21 39F1BB75 5FEF65BC 391F8B36
F8F8EB73 71FD558B

G_Y = 0100 6A08A419 03350678 E58528BE BF8A0BEF F867A7CA
36716F7E 01F81052

Q = 0100 00000000 00000000 00000000 0013E974 E72F8A69
22031D26 03CFE0D7

F.7.3.3 Khóa ký và khóa kiểm tra

X = 00BF 83825505 3DBF499C BE190DE3 5BC14AFC 1EA142F3
5EE69838 5B48D688

Y = (Y_X, Y_Y)

Y_X = 01F4 85A65E59 B336E140 1C8A311F 01C92626 C663E69F
12A627E5 3E8F0675

Y_Y = 01BF 338CE75A DFB07DEB D962E1D8 0C101587 269AC995
1B40422B 12E9DA3E

F.7.3.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "This is a sample message for EC-KCDSA implementation validation." =
54 68 69 73 20 69 73 20 61 20 73 61 6D 70 6C 65 20 6D
65 73 73 61 67 65 20 66 6F 72 20 45 43 2D 4B 43 44 53
41 20 69 6D 70 6C 65 6D 65 6E 74 61 74 69 6F 6E 20 76
61 6C 69 64 61 74 69 6F 6E 2E

K = 00F4 F0881926 8EB1CD8B 4ECB3A53 33746B40 EBF16966
A213B18A 176B2F62

Π = $G^k = (\Pi_X, \Pi_Y)$.

Π_X = 00E4 5041E7AA 060B8B1A 02A7ACAC DB4E95EF F61F33C0
BB8D6EC2 F1C68BA1

Π_Y = 0155 B3A1DA61 F81E04D5 80D07E92 93DF3D4C 7FE34686
BD157374 4D8D3F18

Y' = 01F485A6 5E59B336 E1401C8A 311F01C9 2626C663 E69F12A6
27E53E8F 067501BF 338CE75A DFB07DEB D962E1D8 0C101587

269AC995 1B40422B 12E9DA3E 00000000

$h(Y'||M)$ = E74B3C74 72F2E97E C31861CA 1773472E 58828A98 026277CB

00EF36AC

F.7.3.5 Ký

R = 82EF9427 4AC70A3D AC231E3B AE0F0D31 8FD8E189 EE40A3E0

61EC80BF

S = 00A8 CD7F7573 BAC3C4C4 00F65FDC CCD46F58 EBFC54CE

45571075 FD7704CB

F.7.3.6 Kiểm tra

R' = 82EF9427 4AC70A3D AC231E38 AE0F0D31 8FD8E189 EE40A3E0

61EC80BF

F.7.4 Ví dụ 4: Trường F_2^m, m = 233 (đường cong Koblitz), SHA-224

F.7.4.1 Tổng quan

Ví dụ này sử dụng SHA-224 như hàm băm. Mã băm chỉ đơn giản là giá trị của SHA-224. Ví dụ này sử dụng đường cong Koblitz như đường cong elliptic.

F.7.4.2 Các tham số

Trường F_2^m được mô tả bởi đa thức modulo rút gọn $x^{233} + x^{74} + 1$.

Đường cong elliptic: $Y^2 + XY = X^3 + aX^2 + b$ trên trường F_2^m.

a = 0

b = 1

G = (G_X, G_Y)

G_X = 0172 32BA853A 7E731AF1 29F22FF4 149563A4 19C26BF5

0A4C9D6E EFAD6126

G_Y = 01DB 537DECE8 19B7F70F 555A67C4 27A8CD9B F18AEB9B

56E0C110 56FAE6A3

Q = 80 00000000 00000000 00000000 00069D5B B915BCD4

6EFB1AD5 F173ABDF

F.7.4.3 Khóa ký và khóa kiểm tra

X = 0073 6439374F 72B1C723 AE611CB3 DFBCA0A8 E2C5096B
DB9C2D37 21167B49

Y = (Y_X, Y_Y)

Y_X = 01E9 1DEFBD41 AE655105 E046E03E C13E3860 0E9A2C9
920B8E75 53721605

Y_Y = 0112 9C2706D1 9D134891 C7BAD84A 5600C2AF F86068C4
7497F5BD 498D0B76

F.7.4.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "This is a sample message for EC-KCDSA implementation validation." =
54 68 69 73 20 69 73 20 61 20 73 61 6D 70 6C 65 20 6D
65 73 73 61 67 65 20 66 6F 72 20 45 43 2D 4B 43 44 53
41 20 69 6D 70 6C 65 6D 65 6E 74 61 74 69 6F 6E 20 76
61 6C 69 64 61 74 69 6F 6E 2E

K = 0061 7AA0B7A8 197A2B81 01500BFE 55D5322A 7149E275
F91ADBC7 E30128E4

Π = $G^k = (\Pi_X, \Pi_Y)$.

Π_X = 01BB 9CDB150A 2E5669ED C491320C 3F84E28A 7D6631BC
51127677 A2CF2FEF

Π_Y = 00DA E917793C 12DE86AA 6727C396 A3131B69 33344EDD
B621DD29 BC09B648

Y' = 01E91DEF BD41AE65 5105E046 E03EC13E 38600E9A 2C9A920B
8E755372 16050112 9C2706D1 9D134891 C7BAD84A 5600C2AF
F86068C4 7497F5BD 498D0B76 00000000

$h(Y'||M)$ = FC712972 727661DE B546E86A B6937DB7 D9E61A36 DF5CEA86
044BFF25

F.7.4.5 Ký

R = B164A12F 615CC661 C10B78CB 6E01C9DE 46337C50 C036FAC5
51178752

S = 004A 2109081E B3ADF95C 19FFAE89 5D303B83 147B27C6
EFAE8536 2BFAB89A

F.7.4.6 Kiểm tra

R' = B164A12F 615CC661 C10B78CB 6E01C9DE 46337C50 C036FAC5
51178752

F.7.5 Ví dụ 5: Trường F_2^m, m = 283, SHA-256

F.7.5.1 Tổng quan

Ví dụ này sử dụng SHA-256 như hàm băm. Mã băm chỉ đơn giản là giá trị của SHA-256.

F.7.5.2 Các tham số

Trường F_2^m được mô tả bởi đa thức modulo rút gọn $x^{283} + x^{12} + x^7 + x^5 + 1$.

Đường cong elliptic: $Y^2 + XY = X^3 + aX^2 + b$ trên trường F_2^m.

a = 1

b = 027B680A C8B8596D A5A4AF8A 19A0303F CA97FD76 45309FA2
A581485A F6263E31 3B79A2F5

G = (G_X, G_Y)

G_X = 05F93925 8DB7DD90 E1934F8C 70B0DFEC 2EED25B8 557EAC9C
80E2E198 F8CDBECD 86B12053

G_Y = 03676854 FE24141C B98FE6D4 B20D02B4 516FF702 350EDDB0
826779C8 13F0DF45 BE8112F4

Q = 03FFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFEF90 399660FC
938A9016 5B042A7C EFADB307

F.7.5.3 Khóa ký và khóa kiểm tra

X = 00D64BEC 51F1ADA0 5BBD4F2B 53405B0C E8A1B99C D8DB6309
76A47F76 F08F205E EFC3FBD8

Y = (Y_X, Y_Y)

Y_X = 04313C7E 9C4F80D2 6A287B37 FE7FAA96 BE31F116 2E18BDB4
70CF43D4 DB28DE10 8B007E9F

Y_Y = 0342CCF6 F502F9DF EC208170 24326C26 E867E1FB EC6634CB
17023CA0 222D6112 E0BFA106

F.7.5.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "This is a sample message for EC-KCDSA implementation validation." =

54 68 69 73 20 69 73 20 61 20 73 61 6D 70 6C 65 20 6D

65 73 73 61 67 65 20 66 6F 72 20 45 43 2D 4B 43 44 53

41 20 69 6D 70 6C 65 6D 65 6E 74 61 74 69 6F 6E 20 76

61 6C 69 64 61 74 69 6F 6E 2E

K = 00D18E44 CB7F75F8 01277FA5 CF31A268 8CC2F322 2FA9F26E

E8598126 AFEEE4E3 8DD0E08E

$\Pi = G^k = (\Pi_X, \Pi_Y)$.

Π_X = 01EBE1E7 8BAF9EF6 833189B3 ACC5B3DC B5788292 E0006D90

F8A4E2AE C3027F28 BE47FACA

Π_Y = 0721C5B3 4A038EE1 1DEE2FAE DA84FD46 CBCF37BA 676677BB

AB731AE8 8C52833B AB776F45

Y' = 04313C7E 9C4F80D2 6A287B37 FE7FAA96 BE31F116 2E18BDB4

70CF43D4 DB28DE10 8B007E9F 0342CCF6 F502F9DF EC208170

24326C26 E867E1FB EC6634CB 17023CA0

$h(Y'||M)$ = 148DF2CD 1A4E5437 69F5F0B4 FE07A87A D630C512 A3978248

5B8B8A1A EA50D662

F.7.5.5 Ký

R = 4A23BA73 B29A9010 ACD1E231 3B9A252C E209C7BF 3643926F

A7BF8C87 A8C76D40

S = 03AA4FFF F1F4C3EE BF9C8798 2E717572 71CB7662 BA03463B

8B5F97B0 5C7F7C2C 88A31799

F.7.5.6 Kiểm tra

R' = 4A23BA73 B29A9010 ACD1E231 3B9A252C E209C7BF 3643926F

A7BF8C87 A8C76D40

F.7.6 Ví dụ 6: Trường F_2^m, m = 283 (đường cong Koblitz), SHA-256

F.7.6.1 Tổng quan

Ví dụ này sử dụng SHA-256 như hàm băm. Mã băm chỉ đơn giản là giá trị của SHA-256. Ví dụ này sử dụng đường cong Koblitz như đường cong eliiptic.

F.7.6.2 Các tham số

Trường F_2^m được mô tả bởi đa thức modulo rút gọn $x^{283} + x^{12} + x^7 + x^5 + 1$.

Đường cong elliptic: $Y^2 + XY = X^3 + aX^2 + b$ trên trường F_2^m.

a = 0

b = 1

G = (G_X, G_Y)

G_X = 0503213F 78CA4488 3F1A3B81 62F188E5 53CD265F 23C1567A
16876913 B0C2AC24 58492836

G_Y = 01CCDA38 0F1C9E31 8D90F95D 07E5426F E87E45C0 E8184698
E4596236 4E341161 770D2259

Q = 01FFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFE9AE 2ED07577
265DFF7F 94451E06 1E163C61

F.7.6.3 Khóa ký và khóa kiểm tra

X = 014930E6 6B51F09F EEBBAFFC 9111C5CF 8AE406C9 35AC9618
F0A613B9 6D97F7DB 8F6EBA74

Y = (Y_X, Y_Y)

Y_X = 078A6ACD D5F779F2 5E8AB413 965E217F E6B1E63D 4717EEF5
0DC8C59D F7B1A095 BC3027AE

Y_Y = 07B6D962 5F2D9DDF 516B5037 E1E7B115 26E12AC4 E65AD498
CD85D65A 9E915D58 6976C00F

F.7.6.4 Dữ liệu cho mỗi thông điệp

M = ASCII form of "This is a sample message for EC-KCDSA implementation validation." =
54 68 69 73 20 69 73 20 61 20 73 61 6D 70 6C 65 20 6D
65 73 73 61 67 65 20 66 6F 72 20 45 43 2D 4B 43 44 53
41 20 69 6D 70 6C 65 6D 65 6E 74 61 74 69 6F 6E 20 76
61 6C 69 64 61 74 69 6F 6E 2E

K = 01EA8FB5 72B7B2DA 7149DCD8 78101ECF 3F296400 E13A0D65
C8B6E558 C0237C6D A55268A1

$\Pi = G^k = (\Pi_X, \Pi_Y)$.

Π_X = 0227BDFF E74468EE 3A327AAC 7078252F F545113A 1DD9A2E0
7A0D238B AE601410 34D91C33

Π_Y = 02F519CC E08F4ACC 46AB4323 45DD0F69 408E1346 5E017832
94DE4128 4E24D02B D6916937

Y' = 078A6ACD D5F779F2 5E8AB413 965E217F E681E63D 4717EEF5
0DC8C59D F7B1A095 BC3027AE 07B6D962 5F2D9DDF 516B5037
E1E7B115 26E12AC4 E65AD498 CD85D65A

$h(Y'||M)$ = 23893E3F 87BA26BB B05E0E9B F83A40B0 14EFB95B C87B3AF4
C34902D6 12C8A2B4

F.7.6.5 Ký

R = E214F3CF 8BBB6E92 F779E6C8 A3424BA8 64734002 5EB49EED
C6016746 81B14AFD

S = 0014CC0B B9245B7A 8BC3C6E0 392AAACE DCED8A61 9D9676E9
73D5244D 7F45E01D B425A93E

F.7.6.6 Kiểm tra

R' = E214F3CF 8BBB6E92 F779E6C8 A3424BA8 64734002 5EB49EED
C6016746 81B14AFD

F.8 Cơ chế EC-GDSA

F.8.1 Tổng quan

Với các ví dụ tiếp theo, sử dụng đường cong Brainpool và SHA-2.

F.8.2 Ví dụ 1: Trường F_P, số nguyên tố P 192 bit, SHA-256

F.8.2.1 Các tham số

Trường F_P với P là hệ thập lục phân

P = C302F41D 932A36CD A7A34630 93D18DB7 8FCE476D E1A86297

Đường còn elliptic: $Y^2 = X^3 + aX + b$ trên trường F_P.

a = 6A911740 76B1E0E1 9C39C031 FE8685C1 CAE040E5 C69A28EF

b = 469A28EF 7C28CCA3 DC721D04 4F4496BC CA7EF414 6FBF25C9

G = (G_X, G_Y)

G_X = C0A0647E AAB6A487 53B033C5 6CB0F090 0A2F5C48 53375FD6

G_Y = 14B69086 6ABD5BB8 8B5F4828 C1490002 E6773FA2 FA299B8F

Q = C302F41D 932A36CD A7A3462F 9E9E916B 5BE8F102 9AC4ACC1

F.8.2.2 Khóa ký và khóa kiểm tra

X = 40F95B49 A3B1BF55 311A56DF D3B5061E E1DF6439 84D41E35

Y = (Y_X, Y_Y)

Y_X = 754A8F6C 30D28AE9 A63443C9 7DEC844A 15F797D0 B78FEE03

Y_Y = 63EC81B4 6A9F3833 025037DF E7DCDDE7 AF20C5E7 C6733C35

F.8.2.3 Dữ liệu cho mỗi thông điệp

M = ASCII form of "brainpoolP192r1" =

62 72 61 69 6E 70 6F 6F 6C 50 31 39 32 72 31

K = 5A966260 96288CC4 69F1704E C05F44D1 EC18BD32 CEB02D5B

$\Pi = (\Pi_X, \Pi_Y)$.

Π_X = A00B0AA2 5DB6AB5C 21B86300 D9BC99F5 6E9DD1B7 F1DC4774

Π_Y = 58C0F50E 2E1F6B01 A50E280E 6DB71637 AE9579BC 1565F369

$h(M)$ = 2AE5880D 61FCA83B 2D4C9281 356B9FD2 F7C21359 BA789FBF

D7068AF2 F9A101EC

H = 2AE5880D 61FCA83B 2D4C9281 356B9FD2 F7C21359 BA789FBF

(H được cắt từ mã băm SHA-256 của thông điệp M có độ dài là q bit).

F.8.2.4 Ký

R = A00B0AA2 5DB6AB5C 21B86300 D9BC99F5 6E9DD1B7 F1DC4774

S = 634635EF 813247D7 20245C94 09FB20A2 67C560C8 8EB2B07B

F.8.2.5 Kiểm tra

R' = A00B0AA2 5DB6AB5C 21B86300 D9BC99F5 6E9DD1B7 F1DC4774

F.8.3 Ví dụ 2: Trường F_P, số nguyên tố P 224 bit, SHA-224

F.8.3.1 Các tham số

Trường F_P với P là hệ thập lục phân

P = D7C134AA 26436686 2A183025 75D1D787 B09F0757 97DA89F5
7EC8C0FF

Đường cong eliiptic $Y^2 = X^3 + aX + b$ trên trường F_p.

a = 68A5E62C A9CE6C1C 299803A6 C1530B51 4E182AD8 B0042A59
CAD29F43

b = 2580F63C CFE44138 870713B1 A92369E3 3E2135D2 66DBB372
386C400B

$G = (G_X, G_Y)$

G_X = D9029AD2 C7E5CF43 40823B2A 87DC68C9 E4CE3174 C1E6EFDE
E12C07D

G_Y = 58AA56F7 72C0726F 24C6B89E 4ECDAC24 354B9E99 CAA3F6D3
761402CD

Q = D7C134AA 26436686 2A183025 75D0FB98 D116BC4B 6DDEBCA3
A5A7939F

F.8.3.2 Khóa ký và khóa kiểm tra

X = 7E75BC2C D573B38A ED0977AD 611763DD 57FB29B2 20883344
B81DF037

$Y = (Y_X, Y_Y)$

Y_X = 8B29B268 866CEDCD A528F443 CAE7B07B F82BDC59 1A4BA29A
C5E7BA4E

Y_Y = CD7746C1 4DEEA220 EA1BF164 C203C46E 60AF6699 CE6E1448
076B5807

F.8.3.3 Dữ liệu mỗi thông điệp

M = ASCII form of "brainpoolP224r1" =
62 72 61 69 6E 70 6F 6F 6C 50 32 32 34 72 31

K = 5B604F2C 35ED0401 FCA31E88 0CB55C2A 7456E71A 5CBAA8DF
2FC03CA9

$\Pi = (\Pi_X, \Pi_Y)$.

Π_X = 60FBB2B1 5F055CD1 D482ED6D C5069C8F 624A3405 B67D11B3 B65E0234

Π_Y = 2C4359F0 A5A69F5A 29D1C1F3 86C1DCA6 5A47D160 DA1FBFB2 BA5A2FDB

$h(M)$ = AC2AC36A D5DAF131 951BA30B 330722C7 4BCFFF79 0617D1F0 908E06AF

F.8.3.4 Ký

R = 60FBB2B1 5F055CD1 D482ED6D C5069C8F 624A3405 B67D11B3 B65E0234

S = 5A050F05 AF0B106B A3F14696 E6162CA4 6FBABD2C 144419DB B5BFBDC0

F.8.3.5 Kiểm tra

R' = 60FBB2B1 5F055CD1 D482ED6D C5069C8F 624A3405 B67D11B3 B65E0234

F.8.4 Ví dụ 3: Trường F_P, số nguyên tố P 256 bit, SHA-256

F.8.4.1 Các tham số

Trường F_P với P là hệ thập lục phân

P = A9FB57DB A1EEA9BC 3E660A90 9D838D72 6E3BF623 D5262028 2013481D 1F6E5377

Đường cong eliiptic $Y^2 = X^3 + aX + b$ trên trường F_P.

a = 7D5A0975 FC2C3057 EEF67530 417AFFE7 FB8055C1 26DC5C6C E94A4B44 F330B5D9

b = 26DC5C6C E94A4B44 F330B5D9 BBD77CBF 95841629 5CF7E1CE 6BCCDC18 FF8C07B6

G = (G_X, G_Y)

G_X = 8BD2AEB9 CB7E57CB 2C4B482F FC81B7AF B9DE27E1 E3BD23C2 3A4453BD 9ACE3262

G_Y = 547EF835 C3DAC4FD 97F8461A 14611DC9 C2774513 2DED8E54

5C1D54C7 2F046997

Q = A9FB57DB A1EEA9BC 3E660A90 9D838D71 8C397AA3 B561A6F7
901E0E82 974856A7

F.8.4.2 Khóa ký và khóa kiểm tra

X = 52B929B4 0297437B 98973A2C 437E8F03 A231EB61 E0CD38FD
AD802F00 D55A13A3

$Y = (Y_X, Y_Y)$

Y_X = 90A53E95 D88397AC 76C7C128 297134D9 4BB52866 AD6474C1
3690A5E1 6848AC0D

Y_Y = 0C31F00F 0B3EAC60 A92A19AD 96E9BA31 3A43E100 D4D68FFF
2B8F1D8C D6790714

F.8.4.3 Dữ liệu mỗi thông điệp

M = ASCII form of "brainpoolP256r1" =
62 72 61 69 6E 70 6F 6F 6C 50 32 35 36 72 31

K = E6421272 DDAB9C20 7B119BDD 10C03861 005752EE AB83AC97
513041AD E6286D9

$\Pi = (\Pi_X, \Pi_Y)$.

Π_X = 829349E3 B6E1F3E5 15EB9581 BE0F958D CCAAA6B6 8D83BA77
01DD7A08 67E44EA7

Π_Y = 0E927978 F600F907 68B68C9E 572D33EC 2F8D8FC9 D577D743
8FDEE63D 27CE9763

$h(M)$ = DB7A981C 4E37DDE0 AEA27A34 E3179BD6 DF3072D4 75A7993A
FA93DF1D A7EC9910

F.8.4.4 Ký

R = 829349E3 B6E1F3E5 15EB9581 BE0F958D CCAAA6B6 8D83BA77
01DD7A08 67E44EA7

S = 3DC2F103 296A793E 50DC2266 657470A4 0D2C9EA1 CA797DEA
610042B7 730BBDCE

F.8.4.5 Kiểm tra

R' = 829349E3 B6E1F3E5 15EB9581 BE0F958D CCAAA6B6 8D83BA77
01DD7A08 67E44EA7

F.9 Cơ chế EC-RDSA

F.9.1 Ví dụ 1: Trường F_P, số nguyên tố P 256 bit, SHA-256

F.9.1.1 Tổng quan

Với các ví dụ dưới đây, SHA-256 được sử dụng riêng cho hàm băm, do đó mã băm chỉ đơn giản là giá trị của SHA-256, được chuyển đổi theo Phụ lục B cho các mục dữ liệu thích hợp.

Từ quan điểm an toàn, điều quan trọng là phải tránh các đường cong yếu về thuật toán mật mã (ví dụ: Đảm bảo rằng một đường cong cụ thể không dễ bị tấn công vào các trường hợp đặc biệt của đường cong elliptic Logarit rời rạc).

F.9.1.2 Các tham số

Trường F_P với P là hệ thập lục phân

P = 80000000 00000000 00000000 00000000 00000000 00000000
00000000 00000431

Đường cong eliiptic $Y^2 = X^3 + aX + b$ trên trường F_P.

a = 7

b = 5FBFF498 AA938CE7 39B8E022 FBAFEF40 563F6E6A 3472FC2A
514C0CE9 DAE23B7E

G = (G_X, G_Y)

G_X = 2

G_Y = 08E2A8A0 E65147D4 BD631603 0E16D19C 85C97F0A 9CA26712
2B96ABBC EA7E8FC8

Q = 80000000 00000000 00000000 00000001 50FE8A18 92976154
C59CFC19 3ACCF5B3

F.9.1.3 Khóa ký và khóa kiểm tra

X = 7A929ADE 789BB9BE 10ED359D D39A72C1 1B60961F 49397EEE
1D19CE98 91EC3B28

Y = (Y_X, Y_Y)

Y_X = 7F2B49E2 70DB6D90 D8595BEC 458B50C5 8585BA1D 4E9B788F
6689DBD8 E56FD80B

Y_Y = 26F1B489 D6701DD1 85C8413A 977B3CBB AF64D1C5 93D26627
DFFB101A 87FF77DA

F.9.1.4 Dữ liệu mỗi thông điệp

M = ASCII form of "abc" = 616263

$h(M)$ = BA7816BF 8F01CFEA 414140DE 5DAE2223 B00361A3 96177A9C
B410FF61 F20015AD

K = 77105C98 20BCD312 2823C8CF 6FCC7B95 6DE33814 E95B7FE6
4FED9245 94DCEAB3

$\Pi = (\Pi_X, \Pi_Y)$

Π_X = 41AA28D2 F1AB1482 80CD9ED5 6FEDA419 74053554 A42767B8
3AD043FD 39DC0493

Π_Y = 489C375A 9941A304 9E33B343 61DD2041 72AD98C3 E5916DE2
7695D22A 61FAE46E

F.9.1.5 Ký

R = 41AA28D2 F1AB1482 80CD9ED5 6FEDA419 74053554 A42767B8
3AD043FD 39DC0493

S = 0A7BA472 2DA5693F 229D175F AB6AFB85 7EC2273B 9F88DA58
92CED311 7FCF1E36

F.9.1.6 Kiểm tra

$\Pi' = (\Pi'_X, \Pi'_Y)$

Π'_X = 41AA28D2 F1AB1482 80CD9ED5 6FEDA419 74053554 A42767B8
3AD043FD 39DC0493

Π'_Y = 489C375A 9941A304 9E33B343 61DD2041 72AD98C3 E5916DE2
7695D22A 61FAE46E

R' = 41AA28D2 F1AB1482 80CD9ED5 6FEDA419 74053554 A42767B8
3AD043FD 39DC0493

F.9.2 Ví dụ 2: Trường F_P, số nguyên tố P 512 bit, SHA-512

F.9.2.1 Tổng quan

Với các ví dụ dưới đây, SHA-512 được sử dụng riêng cho hàm băm, do đó mã băm chỉ đơn giản là giá trị của SHA-512, được chuyển đổi theo Phụ lục B cho các mục dữ liệu thích hợp.

Từ quan điểm an toàn, điều quan trọng là phải tránh các đường cong yếu về thuật toán mật mã (ví dụ: Đảm bảo rằng một đường cong cụ thể không dễ bị tấn công vào các trường hợp đặc biệt của đường cong elliptic Logarit rời rạc).

F.9.2.2 Các tham số

Trường F_P với P là hệ thập lục phân

P = 4531ACD1 FE0023C7 550D267B 6B2FEE80 922B14B2 FFB90F04
D4EB7C09 B5D2D15D F1D85274 1AF4704A 0458047E 80E4546D
35B8336F AC224DD8 1664BBF5 28BE6373

Đường cong eliiptic $Y^2 = X^3 + aX + b$ trên trường F_P.

a = 7

b = 1CFF0806 A31116DA 29D8CFA5 4E57EB74 8BC5F377 E49400FD
D788B649 ECA1AC43 61834013 B2AD7322 480A89CA 58E0CF74
BC9E540C 2ADD6897 FAD0A308 4F302ADC

G = (G_X, G_Y)

G_X = 24D19CC6 4572EE30 F396BF6E BBFD7A6C 5213B3B3 D7057CC8
25F91093 A68CD762 FD606112 62CD838D C6B60AA7 EEE804E2
8BC84997 7FAC33B4 B530F1B1 20248A9A

G_Y = 2BB312A4 3BD2CE6E 0D020613 C857ACDD CFBF061E 91E5F2C3
F32447C2 59F39B2C 83AB156D 77F1496B F7EB3351 E1EE4E43
DC1A18B9 1B24640B 6DBB92CB 1ADD371E

Q = 4531ACD1 FE0023C7 550D267B 6B2FEE80 922B14B2 FFB90F04
D4EB7C09 B5D2D15D A82F2D7E CB1DBAC7 19905C5E ECC423F1
D86E25ED BE23C595 D644AAF1 87E6E6DF

F.9.2.3 Khóa ký và khóa kiểm tra

X = 0BA6048A ADAE241B A40936D4 7756D7C9 3091A0E8 51466970
0EE7508E 508B1020 72E8123B 2200A056 3322DAD2 827E2714
A2636B7B FD18AADF C6296782 1FA18DD4

Y = (Y_X, Y_Y)

Y_X = 115DC5BC 96760C7B 48598D8A B9E740D4 C4A85A65 BE33C181
5B5C320C 854621DD 5A515856 D13314AF 69BC5B92 4C8B4DDF
F75C4541 5C1D9DD9 DD33612C D530EFE1

Y_Y = 37C7C90C D40B0F56 21DC3AC1 B751CFA0 E2634FA0 503B3D52
639F5D7F B72AFD61 EA199441 D943FFE7 F0C70A27 59A3CDB8
4C114E1F 9339FDF2 7F35ECA9 3677BEEC

F.9.2.4 Dữ liệu mỗi thông điệp

M = ASCII form of "abc" = 616263

$h(M)$ = DDAF35A1 93617ABA CC417349 AE204131 12E6FA4E 89A97EA2
0A9EEEE6 4B55D39A 2192992A 274FC1A8 36BA3C23 A3FEEBBD
454D4423 643CE80E 2A9AC94F A54CA49F

K = 3b109d0f 05d95496 1a085730 483ecc3a 8a544589 0e76066e
2ec0410c 33c1ee1e 8d86b971 6cb12fd8 f91843c2 c36c82a4
e29fff5e bcef22cd e4062389 76f28e85

$\Pi = (\Pi_X, \Pi_Y)$

Π_X = 13C56557 E300898B F6C91A08 AF0CAF80 1046A2DC 58CF7E84
A15DA3B6 89C0EB29 73F5BE70 27DBDD77 BCE5D337 6AD5793C
21315785 AA6D2536 A20C9158 14F2ADDC

Π_Y = 308AB8AE 5DF67642 7A94C9FC 014CC352 267F3DC4 48003E1C
491768DE 7F660A00 CF4EB1B7 BC67C0CF E7D20256 B84F690C
BB5751A6 A6328140 1287C279 BB96F231

F.9.2.5 Ký

R = 13C56557 E300898B F6C91A08 AF0CAF80 1046A2DC 58CF7E84
A15DA3B6 89C0EB29 73F5BE70 27DBDD77 BCE5D337 6AD5793C
21315785 AA6D2536 A20C9158 14F2ADDC

S = 32C0B15B E367583B B3FABFF1 49AF87D1 18BF18E1 3487E0C6
AB7580B8 62EC104A 41EC9A5F B17B0E0E DBCFFD92 0D6F627E
704A82CC 534127F6 44FDC958 984DDCA0

F.9.2.6 Kiểm tra

$\Pi' = (\Pi'_X, \Pi'_Y)$

Π'_X = 13C56557 E300898B F6C91A08 AF0CAF80 1046A2DC 58CF7E84
A15DA3B6 89C0EB29 73F5BE70 27DBDD77 BCE5D337 6AD5793C
21315785 AA6D2536 A20C9158 14F2ADDC

Π'_Y = 308AB8AE 5DF67642 7A94C9FC 014CC352 267F3DC4 48003E1C
491768DE 7F660A00 CF4EB1B7 BC67C0CF E7D20256 B84F690C
BB5751A6 A6328140 1287C279 BB96F231

R' = 13C56557 E300898B F6C91A08 AF0CAF80 1046A2DC 58CF7E84
A15DA3B6 89C0EB29 73F5BE70 27DBDD77 BCE5D337 6AD5793C
21315785 AA6D2536 A20C9158 14F2ADDC

F.10 Cơ chế EC-SDSA

F.10.1 Ví dụ 1: Trường F_p, số nguyên tố P 256 bit, SHA-256

F.10.1.1 Tổng quan

Với các ví dụ này, đường cong NIST được sử dụng như đường cong elipptic. SHA-256 được sử dụng riêng cho hàm băm, do đó mã băm chỉ đơn giản là giá trị của SHA-256, được chuyển đổi theo Phụ lục B cho các mục dữ liệu thích hợp.

Từ quan điểm an toàn, điều quan trọng là phải tránh các đường cong yếu về thuật toán mật mã (ví dụ: Đảm bảo rằng một đường cong cụ thể không dễ bị tấn công vào các trường hợp đặc biệt của đường cong elliptic Logarit rời rạc).

F.10.1.2 Các tham số

Trường F_p với P là hệ thập lục phân

P = FFFFFFFF 00000001 00000000 00000000 00000000 FFFFFFFF FFFFFFFF FFFFFFFF

Đường cong eliiptic $Y^2 = X^3 + aX + b$ trên trường F_p

a = FFFFFFFF 00000001 00000000 00000000 00000000 FFFFFFFF FFFFFFFF FFFFFFFC

b = 5AC635D8 AA3A93E7 B3EBBD55 769886BC 651D06B0 CC53B0F6 3BCE3C3E 27D2604B

G = (G_X, G_Y)

G_X = 6B17D1F2 E12C4247 F8BCE6E5 63A440F2 77037D81 2DEB33A0 F4A13945 D898C296

G_Y = 4FE342E2 FE1A7F9B 8EE7EB4A 7C0F9E16 2BCE3357 6B315ECE CBB64068 37BF51F5

q = FFFFFFFF 00000000 FFFFFFFF FFFFFFFF BCE6FAAD A7179E84 F3B9CAC2 FC632551

F.10.1.3 Khóa ký và khóa kiểm tra

X = 5202A3D8 ACAF6909 D12C9A77 4CD886F9 FBA61137 FFD3E8E7 6AED363F B47AC492

Y = (Y_X, Y_Y)

Y_X = 09B58B88 323C52D1 080AA525 C89E8E12 C6F40FCB 014640FA 88081ED9 E9352DE7

Y_Y = 5CCBBD18 95385162 38B0B0B2 8ACB5F0B 5E27217C 3A987242 1219DE0A EEBF1080

F.10.1.4 Dữ liệu mỗi thông điệp

M = ASCII form of "abc" = 616263

K = DE7E0E5E 663F2418 3414B7C7 2F24546B 81E9E5F4 10B6BF26
F3CA5FA8 2F5192C8

$\Pi = (\Pi_X, \Pi_Y)$

Π_X = 847CE3CD 474FEC19 722AA9BA 81AFBF34 7EE2D70E D067413F
1F716783 27A758CA

Π_Y = DBFAD4AF 8C1D93AB 9C16467E 96BD11B5 33643AA6 63498D8F
95919C6C A1AD91FC

$FE2BS(r, \Pi_X)||FE2BS(r, \Pi_Y)$ = 847CE3CD 474FEC19 722AA9BA 81AFBF34
7EE2D70E D067413F 1F716783 27A758CA DBFAD4AF 8C1D93AB
9C16467E 96BD11B5 33643AA6 63498D8F 95919C6C A1AD91FC

$R = h(FE2BS(r, \Pi_X)||FE2BS(r, , \Pi_Y)||M)$ =5A79A0AA 9B241E38 1A594B22
0554D096 A5F09FA6 28AD9A33 C3CE4393 ADE1DEF7

Cho tối ưu hóa các biến của EC-SDSA

$R = h(FE2BS(r, \Pi_X)||FE2BS(r, , \Pi_Y)||M)$ = D7FB8135 D8EA45E8 FB3C9059 F146E263
0EF4BD51 C4006A92 EDB4C8B0 849963FB

F.10.1.5 Ký

R = 5A79A0AA 9B241E38 1A594B22 0554D096 A5F09FA6 28AD9A33
C3CE4393 ADE1DEF7

S = 5C0EB78B 67A513C3 E53B2619 F96855E2 91D5141C 7CD0915E
1D04B347 457C9601

Cho tối ưu hóa các biến của EC-SDSA

R = D7FB8135 D8EA45E8 FB3C9059 F146E263 0EF4BD51 C4006A92
EDB4C8B0 849963FB

S = B46D1525 379E02E2 32D97928 265B7254 EA2ED978 13454388
C1A08F62 DCCD70B3

F.10.1.6 Kiểm tra

$$\Pi' = \left(\Pi'_X, \Pi'_Y\right)$$

Π'_X = 847CE3CD 474FEC19 722AA9BA 81AFBF34 7EE2D70E D067413F 1F716783 27A758CA

Π'_Y = D8FAD4AF 8C1D93AB 9C16467E 96BD11B5 33643AA6 63498D8F 95919C6C A1AD91FC

$FE2BS(r, \Pi'_X)||FE2BS(r, \Pi'_Y)$ = 847CE3CD 474FEC19 722AA9BA 81AFBF34 7EE2D70E D067413F 1F716783 27A758CA D8FAD4AF 8C1D93AB 9C16467E 96BD11B5 33643AA6 63498D8F 95919C6C A1AD91FC

$R' = h(FE2BS(r, \Pi'_X)||FE2BS(r, \Pi'_Y)||M)$ = 5A79A0AA 9B241E38 1A594B22 0554D096 A5F09FA6 28AD9A33 C3CE4393 ADE1DEF7

Cho tối ưu hóa các biến của EC-SDSA

$R' = h(FE2BS(r, \Pi'_X) \,||\, M)$ = D7FB8135 D8EA45E8 FB3C9059 F146E263 0EF4BD51 C4006A92 EDB4C8B0 849963FB

F.10.2 Ví dụ 2: Trường F_P, số nguyên tố P 384 bit, SHA-384

F.10.2.1 Tổng quan

Với các ví dụ này, đường cong P384 được sử dụng như đường cong elipptic. SHA-384 được sử dụng riêng cho hàm băm, do đó mã băm chỉ đơn giản là giá trị của SHA-384, được chuyển đổi theo Phụ lục B cho các mục dữ liệu thích hợp.

Từ quan điểm an toàn, điều quan trọng là phải tránh các đường cong yếu về thuật toán mật mã (ví dụ: Đảm bảo rằng một đường cong cụ thể không dễ bị tấn công vào các trường hợp đặc biệt của đường cong elliptic Logarit rời rạc).

F.10.2.2 Các tham số

Trường F_P với P là hệ thập lục phân

P = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFFFF 00000000 00000000 FFFFFFFF

Đường cong eliiptic $Y^2 = X^3 + aX + b$ trên trường F_P

a = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF FFFFFFFE FFFFFFFF 00000000 00000000 FFFFFFFC

b = B3312FA7 E23EE7E4 988E056B E3F82D19 181D9C6E FE814112
0314088F 5013875A C656398D 8A2ED19D 2A85C8ED D3EC2AEF

G = (G_X, G_Y)

G_X = AA87CA22 BE8B0537 8EB1C71E F320AD74 6E1D3B62 8BA79B98
59F741E0 82542A38 5502F25D BF55296C 3A545E38 72760AB7

G_Y = 3617DE4A 96262C6F 5D9E98BF 9292DC29 F8F41DBD 289A147C
E9DA3113 B5F0B8C0 0A60B1CE 1D7E819D 7A431D7C 90EA0E5F

q = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
C7634D81 F4372DDF 581A0DB2 48B0A77A ECEC196A CCC52973

F.10.2.3 Khóa ký và khóa kiểm tra

X = 7E4914FE 4B617089 F9FE80AD 913D5530 EEC4F30B CA1AD66E
7B5AEACF 29D2F567 D9A8F4A4 552A1A1A F3E0B6D0 A49DFCC8

Y = (Y_X, Y_Y)

Y_X = 16C000E5 92FADA35 2CB15605 DA2AF639 67B23F9E 9D36F324
0374D357 BB3B7D04 CB76382C 707B18FE 104B632E E1860C36

Y_Y = 0CBB8CF8 65CFB286 B0009342 CA2799EF 2BE3D040 806BF237
88DEE3F8 2E0C2006 7DCD474F 202B4C34 2C4F36BF 6B84E197

F.10.2.4 Dữ liệu mỗi thông điệp

M = ASCII form of "abc" = 616263

K = 8A29E772 357BBA6F 5C9EA765 D5082B9B C7A74C33 E9D94D49
FB2C9D3B 523A8216 9682ECF1 6F1D0626 9042F3AF 044B4DE8

$\Pi = (\Pi_X, \Pi_Y)$

Π_X = 896BF1AC 124A3DBC FF82705E D7699A45 0C827103 47DEE91C
7D061D4B C512ADE1 2D2660B9 BA0F6615 C9324E0A 5E3C1AE7

Π_Y = B0B11AEF 22E5348D B2E3E106 99215B6F 4F87D11B EC2CBDDB
F971A123 3BAD21A2 18DFE905 C0010D88 E7D64625 2633620E

$FE2BS(r, \Pi_X) || FE2BS(r, \Pi_Y)$ = 896BF1AC 124A3DBC FF82705E D7699A45
0C827103 47DEE91C 7D061D4B C512ADE1 2D2660B9 BA0F6615
C9324E0A 5E3C1AE7 B0B11AEF 22E5348D B2E3E106 99215B6F
4F87D11B EC2CBDDBF971A123 3BAD21A2 18DFE905 C0010D88
E7D64625 2633620E

$R = h(FE2BS(r, \Pi_X) || FE2BS(r, \Pi_Y) || M)$ = F907553B B5C7DE02 9A2A5670
78DFF9B8 03EC6496 0D75BA73 A85590AC C0AC4479 AC52E51D
5691FCB0 69DC5CD2 4E0BCEC7

Cho tối ưu hóa các biến của EC-SDSA

$R = h(FE2BS(r, \Pi_X) \| M)$ = 27D2F5B9 62A3ACF6 390A4718 EA540DA7
9612A60E AA15BEBB 00B9E166 5783F7C7 91CCAC42 2CEE815A
9C5DA367 8AC8D1F0

F.10.2.5 Ký

R = F907553B B5C7DE02 9A2A5670 78DFF9B8 03EC6496 0D75BA73
A85590AC C0AC4479 AC52E51D 5691FCB0 69DC5CD2 4E0BCEC7

S = 0B9D66D5 DE70FAA8 B35634A3 7B33C2C4 60B8DC0B D4C8745B
B84DC15C A8570B07 9258F977 DA8B4061 F3DA6EBD 7C429A89

Cho tối ưu hóa các biến của EC-SDSA

R = 27D2F5B9 62A3ACF6 390A4718 EA540DA7 9612A60E AA15BEBB
00B9E166 5783F7C7 91CCAC42 2CEE815A 9C5DA367 8AC8D1F0

S = 22CC89CE B9E6BE84 15CC14B3 99BC66E6 F3A21E5B A38E09A6
DE8DE670 A145C0E4 74D5CC88 BE8878F0 123CC662 25A1BA12

F.10.2.6 Kiểm tra

$\Pi' = (\Pi'_X, \Pi'_Y)$

Π'_X = 896BF1AC 124A3DBC FF82705E D7699A45 0C827103 47DEE91C
7D061D4B C512ADE1 2D2660B9 BA0F6615 C9324E0A 5E3C1AE7

Π'_Y = B0B11AEF 22E5348D B2E3E106 99215B6F 4F87D11B EC2CBDDB
F971A123 3BAD21A2 18DFE905 C0010D88 E7D64625 2633620E

$FE2BS(r, \Pi'_X)||FE2BS(r, \Pi'_Y)$ = 896BF1AC 124A3DBC FF82705E D7699A45
0C827103 47DEE91C 7D061D4B C512ADE1 2D2660B9 BA0F6615
C9324E0A 5E3C1AE7 B0B11AEF 22E5348D B2E3E106 99215B6F
4F87D11B EC2CBDDB F971A123 3BAD21A2 18DFE905 C0010D88
E7D64625 2633620E

R' = $h(FE2BS(r, \Pi'_X)||FE2BS(r, \Pi'_Y)||M)$ = F907553B B5C7DE02 9A2A5670
78DFF9B8 03EC6496 0D75BA73 A85590AC C0AC4479 AC52E51D
5691FCB0 69DC5CD2 4E0BCEC7

Cho tối ưu hóa các biến của EC-SDSA

R' = $h(FE2BS(r, \Pi'_X) \| M)$ = 27D2F5B9 62A3ACF6 390A4718 EA540DA7
9612A60E AA15BEBB 00B9E166 5783F7C7 91CCAC42 2CEE815A
9C5DA367 8AC8D1F0

F.11 Cơ chế EC-FSDSA

F.11.1 Ví dụ 1: Trường F_P, số nguyên tố P 256 bit, SHA-256

F.11.1.1 Tổng quan

Với các ví dụ này, đường cong NIST P256 được sử dụng như đường cong elipptic. SHA-256 được sử dụng riêng cho hàm băm, do đó mã băm chỉ đơn giản là giá trị của SHA-256, được chuyển đổi theo Phụ lục B cho các mục dữ liệu thích hợp.

Từ quan điểm an toàn, điều quan trọng là phải tránh các đường cong yếu về thuật toán mật mã (ví dụ: Đảm bảo rằng một đường cong cụ thể không dễ bị tấn công vào các trường hợp đặc biệt của đường cong elliptic Logarit rời rạc).

F.11.1.2 Các tham số

Trường F_P với P là hệ thập lục phân

P = FFFFFFFF 00000001 00000000 00000000 00000000 FFFFFFFF
FFFFFFFF FFFFFFFF

Đường cong eliiptic $Y^2 = X^3 + aX + b$ trên trường F_P

a = FFFFFFFF 00000001 00000000 00000000 00000000 FFFFFFFF FFFFFFFF FFFFFFFC

b = 5AC635D8 AA3A93E7 B3EBBD55 769886BC 651D06B0 CC53B0F6 3BCE3C3E 27D2604B

$G = (G_X, G_Y)$

G_X = 6B17D1F2 E12C4247 F8BCE6E5 63A440F2 77037D81 2DEB33A0 F4A13945 D898C296

G_Y = 4FE342E2 FE1A7F9B 8EE7EB4A 7C0F9E16 2BCE3357 6B315ECE CBB64068 37BF51F5

q = FFFFFFFF 00000000 FFFFFFFF FFFFFFFF BCE6FAAD A7179E84 F3B9CAC2 FC632551

F.11.1.3 Khóa ký và Khóa kiểm tra

X = ACCA7F0D D3AC535F 489B340F 6BD7F503 61B0EE09 5AE6A228 9A6AB329 238123E5

$Y = (Y_X, Y_Y)$

Y_X = B54E3003 72FFB75C 0A5E495C 59A036BE ABB54134 00F0ADF3 C2B5B160 BA959578

Y_Y = 58F8505E 1673AA64 51F84C37 BF338519 108AA89E 33AEA991 5168D6F1 E3B67E13

F.11.1.4 Dữ liệu mỗi thông điệp

M = ASCII form of "abc" = 616263

K = 894DEAB4 4D88450F E8DAC663 F0E58650 31E875BA 224C0601 3C53D0E3 0109C207

$\Pi = (\Pi_X, \Pi_Y)$

Π_X = AF312FBD 7792125C 5CDFBA69 E6D36990 0ACE9A70 BA653FFF BD9140E0 0079FAE8

Π_Y = B7CEC570 16A0B97A A069D54E 0DA95E45 FB50B677 1FB69F53 F3F00FC8 B00E1FEC

R = AF312FBD 7792125C 5CDFBA69 E6D36990 0ACE9A70 BA653FFF
BD9140E0 0079FAE8 B7CEC570 16A0B97A A069D54E 0DA95E45
FB50B677 1FB69F53 FEF00FC8 B00E1FEC

$H(R||M)$ = 44E762CA 7B26973F CDFA1301 0658E68E 88EA9D06
FB76782D AFC0EB78 1F9213C2

F.11.1.5 Ký

R = AF312FBD 7792125C 5CDFBA69 E6D36990 0ACE9A70 BA653FFF
BD9140E0 0079FAE8 B7CEC570 16A0B97A A069D54E 0DA95E45
FB50B677 1FB69F53 FEF00FC8 B00E1FEC

S = 25847040 2304BC2D B44F3B2A 20C08FF2 A64F566B AA2EB7BF
37E1619B 6AE09844

F.11.1.6 Kiểm tra

$\Pi' = (\Pi'_X, \Pi'_Y)$

Π'_X = AF312FBD 7792125C 5CDFBA69 E6D36990 0ACE9A70 BA653FFF
BD9140E0 0079FAE8

Π'_Y = B7CEC570 16A0B97A A069D54E 0DA95E45 FB50B677 1FB69F53
FEF00FC8 B00E1FEC

R' = AF312FBD 7792125C 5CDFBA69 E6D36990 0ACE9A70 BA653FFF
BD9140E0 0079FAE8 B7CEC570 16A0B97A A069D54E 0DA95E45
FB50B677 1FB69F53 FEF00FC8 B00E1FEC

F.11.2 Ví dụ 2: Trường F_P, số nguyên tố P 384 bit, SHA-384

F.11.2.1 Tổng quan

Với các ví dụ này, đường cong NIST P384 được sử dụng như đường cong elipptic. SHA-384 được sử dụng riêng cho hàm băm, do đó mã băm chỉ đơn giản là giá trị của SHA-384, được chuyển đổi theo Phụ lục B cho các mục dữ liệu thích hợp.

Từ quan điểm an toàn, điều quan trọng là phải tránh các đường cong yếu về thuật toán mật mã (ví dụ: Đảm bảo rằng một đường cong cụ thể không dễ bị tấn công vào các trường hợp đặc biệt của đường cong elliptic Logarit rời rạc).

F.11.2.2 Các tham số

Trường F_P với P là hệ thập lục phân

P = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF FFFFFFFE FFFFFFFF 00000000 00000000 FFFFFFFF

Đường cong eliiptic $Y^2 = X^3 + aX + b$ trên trường F_P

a = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF FFFFFFFE FFFFFFFF 00000000 00000000 FFFFFFFC

b = B3312FA7 E23EE7E4 988E056B E3F82D19 181D9C6E FE814112
0314088F 5013875A C656398D 8A2ED19D 2A85C8ED D3EC2AEF

G = (G_X, G_Y)

G_X = AA87CA22 BE8B0537 8EB1C71E F320AD74 6E1D3B62 8BA79B98
59F741E0 82542A38 5502F25D BF55296C 3A545E38 72760AB7

G_Y = 3617DE4A 96262C6F 5D9E98BF 9292DC29 F8F41DBD 289A147C
E9DA3113 B5F0B8C0 0A60B1CE 1D7E819D 7A431D7C 90EA0E5F

q = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
C7634D81 F4372DDF 581A0DB2 48B0A77A ECEC196A CCC52973

F.11.2.3 Khóa ký và khóa kiểm tra

X = 95A4D257 A7298C66 10A37558 785036DE 6F9FB997 735C076B
8C8A18B2 AAAC3142 507A2560 3D7C95F9 E5F0307E C5A56D7E

Y = (Y_X, Y_Y)

Y_X = 75E4FB35 F5FBBE88 5AC414BA 51C2F9F7 57D071F8 54A839A5
84031158 4A0559A7 2CDC563E 29D752AB 4D511882 7489F434

Y_Y = 2A693A17 2EC9F108 BE348528 8695EF4E 6B20971F 69859E9D
13044881 FCD76224 C6369A09 B96DF415 4592ADB3 083A2005

F.11.2.4 Dữ liệu mỗi thông điệp

M = ASCII form of "abc" = 616263

K = 51C5B8B2 E59CF78F 54E77CDB 0B2E2669 B66B3284 8B7B5378
01483DE2 39422745 6F4930C8 53FBFF6C 58FA6E1C C5D97466

Π = (Π_X, Π_Y)

Π_X = 29B785AB 7FAC1F80 F64CE2F2 D88ABA8F E6103B25 65FEB5AE
82FB6CF5 8F8CE1E7 3F7A8D68 3BEABD24 B7EA78F6 013C9F70

Π_Y = DE94FAD1 7281F4A6 0AA25423 E9D87122 96540219 5239B839
FCD44CDF 545BF74E 1300C9F5 7FFC830D EFD97B66 D57E9D07

R = 29B785AB 7FAC1F80 F64CE2F2 D88ABA8F E6103B25 65FEB5AE
82FB6CF5 8F8CE1E7 3F7A8D68 3BEABD24 B7EA78F6 013C9F70
DE94FAD1 7281F4A6 0AA25423 E9D87122 96540219 5239B839
FCD44CDF 545BF74E 1300C9F5 7FFC830D EFD97B66 D57E9D07

$h(R||M)$ = 970BC355 820ADDDD 8F39CBD9 E386BE70 143B965F
B4E720AF 1E31658F 87410729 758DEEDD B706221C 332308A6
1D9D0F40

F.11.2.5 Ký

R = 29B785AB 7FAC1F80 F64CE2F2 D88ABA8F E6103B25 65FEB5AE
82FB6CF5 8F8CE1E7 3F7A8D68 3BEABD24 B7EA78F6 013C9F70
DE94FAD1 7281F4A6 0AA25423 E9D87122 96540219 5239B839
FCD44CDF 545BF74E 1300C9F5 7FFC830D EFD97B66 D57E9D07

S = D1858062 C5504E21 78523926 423FDD83 99A8BA2B 85BF4585
3F8E04BF 20441516 E71A78B0 9C7A7EE6 20B7F537 E6C1DEEE

F.11.2.6 Kiểm tra

$\Pi' = (\Pi'_X, \Pi'_Y)$

Π'_X = 29B785AB 7FAC1F80 F64CE2F2 D88ABA8F E6103B25 65FEB5AE
82FB6CF5 8F8CE1E7 3F7A8D68 3BEABD24 B7EA78F6 013C9F70

Π'_Y = DE94FAD1 7281F4A6 0AA25423 E9D87122 96540219 5239B839
FCD44CDF 545BF74E 1300C9F5 7FFC830D EFD97B66 D57E9D07

R' = 29B785AB 7FAC1F80 F64CE2F2 D88ABA8F E6103B25 65FEB5AE
82FB6CF5 8F8CE1E7 3F7A8D68 3BEABD24 B7EA78F6 013C9F70
DE94FAD1 7281F4A6 0AA25423 E9D87122 96540219 5239B839
FCD44CDF 545BF74E 1300C9F5 7FFC830D EFD97B66 D57E9D07

F.12 Cơ chế IBS-1

F.12.1 Ví dụ 1: Trường F_P, số nguyên tố P 512 bit, SHA-1

F.12.1.1 Tổng quan

Với ví dụ này, các phần tử trong trường F_p^2 được biểu diễn như $a + \sigma b$ với a và b là các phần tử trong trường F_p và σ là phần tử trong trường F_p^2 thỏa mãn $\sigma^2 + 1 = 0 \, mod \, p$. Ánh xạ xoắn được dùng để biến đổi một điểm trong một nhóm xoắn với các phần tử của nhóm xoắn khác để tạo thành các cặp không

suy biến hoạt động như sau : $\phi P: (P_X, P_Y) \rightarrow (\beta P_X, P_Y)$ với $\beta \neq 1$ và $\beta \in F_p^2$ thỏa mãn $\beta^3 - 1 = 0 \, mod \, p$. <P,Q> được thực hiện như việc giảm cặp Tate ở hai điểm đầu vào P và ϕ (Q).

H_2 được thực hiện như SHA-1. H_1 được thực hiện như sau: đưa ra thông điệp M, $H_1(M) = I2P(RS2I(512, MGF1(M)))$ với $MGF1(M)$ là 512 bit trái nhất của $SHA-1(M \| 0) \| SHA-1(M \| 1) \| SHA-1(M \| 2) \| SHA-1(M \| 3)$. Ở đây các hằng số 0, 1, 2, và 3 là giá trị 32 bit theo thứ tự để nối dữ liệu.

F.12.1.2 Các tham số

Trường F_P với P là hệ thập lục phân

p = B35FA5FD E47FA1AB BB1E57E9 3BA1FF96 38B89B99 5C49BE81 A38E3194
A0983816 4EE51FB9 1D285832 F9A05D63 9C8D9680 10C93A35 27E561F2
FD6A45CC 70ABA1FB

Đường cong eliiptic $Y^2 = X^3 + 1$ trên trường F_P

q = 80000000 000FFFFF FFFFFFFF FFFFFFFF FFFFFFFF

$\beta = (\beta_a, \beta_b)$

β_a = 59AFD2FE F23FD0D5 DD8F2BF4 9DD0FFCB 1C5C4DCC AE24DF40 D1C718CA
504C1C0B 27728FDC 8E942C19 7CD02EB1 CE46CB40 08649D1A 93F2B0F9
7EB522E6 3855D0FD

β_b = 20C3F863 7D0F88AE 295531C6 82537070 5EFA0F99 39D1CDE4 F58A25F4
3722E229 5410CA05 B2360EA8 625C8203 626C5254 3E63404A B83448B0
3FDA177A 7A2DDB21

$P = (P_X, P_Y)$

P_X = 7EE6F118 9329ADB4 1E8CD405 2295F1A7 6096631D F065DD38 85FFF26B
8ED52022 7BFC3F3D 07FE9CFE 093424BA 9DBD4C0D 73FFF367 3CE2C922
F0B73C50 2992093B

P_Y = 67C855C9 E6B617BE B24B792A E9C3E21E 95F37006 25B91058 3BC0B293
C36A762C FEBA4266 038989CC 59797235 C6116D99 8A97B805 FF82C664
53720E5C 1DE95E3E

U = 50BD36F5 967F41F4 93B1F667 4AD462F6 2B7D69D8

V = $[U]P = (V_X, V_Y)$

V_X = 04863127 7E2F50F6 CCCD235D EC856E8C FB22F17A 232D6229 7B7ABA44
77639ADE F2CC4863 5C19ED50 9E7F3F2C C5EF1F5B 5225E45D 30C269A3
97978783 D7E0DD86

V_Y = 75651DE4 8D8AEC70 10B4084A 05EE2EFB 0D72CF41 4F5B6113 539198F5
DC2317AA 3B6DEE3E E800394F 470020FB E6EE9811 CDDC5991 4BD0CB38
2E203F01 0816E8B7

$\phi(V)$ = $(\beta V_X, V_Y)$

βV_{Xa} = B11C8D6A 2567F930 54B7C63A 455F484F BB2722DC 4AB30D6C E5D0D472
64E66AA6 D57EFB87 6F1B618A AA60BDCD 399606D2 67B64806 8F842D21
319E820A 84BB3338

βV_{Xb} = AA701CC7 9EF755C5 5E33269C 4F54BFB2 819E0F5A 6F6C8084 557428CA
71F0A225 393CB501 66D2456C A6F1D22A C983D4FF CBF03583 813D2893
7087739B 7E279447

V_Y = 75651DE4 8D8AEC70 10B4084A 05EE2EFB 0D72CF41 4F5B6113 539198F5
DC2317AA 3B6DEE3E E800394F 470020FB E6EE9811 CDDC5991 4BD0CB38
2E203F01 0816E8B7

F.12.1.3 Khóa ký và khóa kiểm tra

ID = "alice@network.com"

$MGF1(ID)$ = 85499895 C3B4F412 56E84BDC E3D76405 33612345 D6BF0725 FB7D0391
D53A680B 2DD96646 A32B244E DA614B5D 8E989340 473B7885 AF243CE0
0FBD2661 43B85636

$I2P(BS2I(512, MGF1(ID)))$ = (Y_X, Y_Y)

Y_X = 8BB6EFE1 AF00D035 D0B286F1 94BC3128 A639949A B9D06405 D95F837A
E131C3C6 B37F2905 4D58FB21 9FA190B7 A65BE61F 4A6945B5 B4A1A36C
6166F047 2F63D303

Y_Y = 3D605915 84CF4C40 2E81145C 0047D9BE 2733A3C2 66F7A8F8 6CA2F50C
635424C3 8AFF64FA F9B5CD04 4AD0236F BC7BEB72 0D1909DA 5CC3C3FE
3022CDB7 7946F37D

X = $[U]Y = (X_X, X_Y)$

X_X = B26A00EA 12877F1D 049DC4F2 4030CDB1 632AA9FA D48E209D 8CC1A116
A159CBDD 540F417D FC584FF9 4C959E64 3BFA2131 7EB54356 EBB8650B
8803CB32 F607C87B

X_Y = 3527A3A6 E22290F9 0CC5ECBE F6CFE3DA F0F7EF16 327606C8 A2456C9F
6C7A2818 84832B75 41D8EAAE DB1B0FDA 78335B84 C56FEF1F E88423C3
1393760B AE4520D1

F.12.1.4 Dữ liệu mỗi thông điệp

M = ASCII form of "abc" = 61 62 63

K = 3846BAA7 C64DA228 E356EE28 5E2DE770 6D2C4F83

$\Pi = (\Pi_a, \Pi_b)$

Π_a = 0035A456 EB8285BA CC8912FD 84BA4A9A 68CEA16A D69570FC 5E10C90D
44D84269 2F3097A6 A715AE83 C48E0413 EFEACBA1 A6BAF06D 621B5835
E441C245 4C9601ED

Π_b = 8302C6D9 5413BAAE E5484202 8189CA1F BDB85BA2 D1544428 05158303
ECEE99B4 6A24CCD2 D8580A1F 00A9B2A2 568125C3 DE7925F6 0E1FC61D
F78F9F1D E7A6C76D

F.12.1.5 Ký

$R = BS2I(\gamma, H_2(M_1 \,||\, FE2BS(r, \Pi_a) \,||\, FE2BS(r, \Pi_b))) \bmod q =$
01D7ED22 EC366550 D4246317 1662B5CA 1742C1E8

$S = (S_X, S_Y)$

S_X = 8E911C5B 986E0547 E5F31C57 1FDC384D F8968374 23B34BF6 EE8865C1
2BC1D56B 9E6D4B8A 15DAD877 685CB024 1FA07012 EC0E9183 B18955D5
570DAEBB BF0CD364

S_Y = 00CBFC08 18782909 2F6F2496 D1F2299E 5159B89F 6535E3E4 D708D77D
8FD7F262 E4E05331 81BD4933 A0CFA2FC 710ECC59 99078706 BA0B4F3A
468B61DA 3045A0E8

F.12.1.6 Kiểm tra

R' = 01D7ED22 EC366550 D4246317 1662B5CA 1742C1E8

F.12.2 Ví dụ 2: Trường F_P, số nguyên tố P 512 bit, SHA-1

F.12.2.1 Tổng quan

Với ví dụ này, các phần tử trong trường F_p^2 được biểu diễn như $a + \sigma b$ với a và b là các phần tử thuộc trường F_p và σ là phần tử trong trường F_p^2 thỏa mãn $\sigma^2 + 1 = 0 \, mod \, p$. Ánh xạ xoắn được dùng để biến đổi một điểm trong một nhóm xoắn với các phần tử của nhóm xoắn khác để tạo thành các cặp không suy biến hoạt động như sau : $\phi P: (P_X, P_Y) \rightarrow (-P_X, \sigma P_Y)$. <P,Q> được thực hiện như việc giảm cặp Tate ở hai điểm đầu vào P và ϕ (Q).

H_2 được thực hiện như SHA-1. H_1 được thực hiện như sau: đưa ra thông điệp M, $H_1(M) = I2P(BS2I(512, MGF1(M)))$ với $MGF1(M)$ là kết quả lấy từ 512 bit đầu tiên của $SHA-1(M \parallel 0) \parallel SHA-1(M \parallel 1) \parallel SHA-1(M \parallel 2) \parallel SHA-1(M \parallel 3)$ và số 0 ra đầu tiên.

F.12.2.2 Các tham số

Trường F_P với P là hệ thập lục phân

p = 80000000 00000000 00000000 00000000 00020001 40000000 00000000
00000000 00000000 00010000 80000002 00000000 00000000 00000000
00000000 00080003

Đường cong eliiptic $Y^2 = X^3 + X$ trên trường F_P

$\#E$ = 80000000 00000000 00000000 00000000 00020001 40000000 00000000
00000000 00000000 00010000 80000002 00000000 00000000 00000000
00000000 00080004

q = 80000000 00000000 00000000 00000000 00020001

$P = (P_X, P_Y)$

P_X = 0DB4E0F7 22DD090D A2B6D8FE ADAF21D9 546AB265 1515AF9B A87108F3
4E1AE0E3 EB132C10 81452CC1 E52BB2A7 4287A0CB D8FF8DD9 3A225641
5321F0E4 C8892A50

P_Y 762C096C 49F1AB04 7D7F37DE 537A4E7C 2991C400 22E0C9A9 B3F58B1B
9dF4F28A 4A4330E2 170E14D2 F55A0719 8B667D0B 01E5A482 3F07E921
8516481E 641970AC

U = 5C1A6406 3FCC6EA9 DAF85736 ECE7C438 688F6FA6

$V = [U]P = (V_X, V_Y)$

V_X = 6B8F666B CF6B4672 D4634753 1F734E71 41BCD5FD 125F3EF3 714EDC28
F6426900 75FFB5F7 9E745CC0 FB03F940 3BDCEFE8 ACBE6286 D5D9955C
2A0E5ED7 657748C6

V_Y = 69584E47 F3070FED 9800D6CD E0F314B4 03955126 1C5BFEF6 F3595F94
5958F7D9 34DCBD3D 63125410 CCD363F8 02DF1C7E 4A3D7AC7 24CF3865

0FB16EC1 7BB30A85

$\phi(V)$ = $(-V_X, V_Y)$

$-V_X$ = 14709994 3094B98D 2B9CB8AC E08CB18E BE452A04 2DA0C10C 8EB123D7
09BD96FF 8A004A08 618CA33F 84FC06C1 C4231017 53419D79 2A266AA3
D5F1A128 9A90B73D

F.12.2.3 Khóa ký và khóa kiểm tra

ID = "alice@network.com"

$MGF1(ID)$ = 05499895 C3B4F412 56E84BDC E3D76405 33612345 D6BF0725 FB7D0391
D53A680B 2DD96646 A32B244E DA614B5D 8E989340 473B7885 AF243CE0
0FBD2661 43B85636

$I2P(BS2I(512, MGF1(ID)))$ = (Y_X, Y_Y)

Y_X = 28C1852E 0D0D6AF3 A55F7846 01D2B1B6 3CA8E9B1 597BB19A 2A7D2CCE
2EED68D9 356BEE5D 81B14B2D 095C6B6C 30461DCE A183BC61 8A115FCE
D4DAC9AE 6EB92570

Y_Y = 1C417DF1 D1BB732F 6849AA30 F954042A 142901C2 6A0ED564 3310D3A5
0A6EAC35 3855DDC0 485311E1 A12735E1 95D76E34 063FF0F5 9DA251F3
E261AC66 D7BFA23B

X = $[U]Y = (X_X, X_Y)$

X_X = 494661E1 3FC73652 E6F9A500 CC9009AC D0B2085D 93709163 0E186185
7E61F889 E9B5CAC9 87D23C51 95E3C3DE B4055DA6 BE911267 48CB481F
B115534A EBD6EF0B

X_Y = 06A03100 C34D09DD 9155DE07 61BED764 744ACA6C 6DC9169C FEC9E245
2B058DFD 9D426B73 9E444976 7DB80539 C7B6C7A2 65B1B7EA 3079A704
3949B35D 4AD88F24

F.12.2.4 Dữ liệu mỗi thông điệp

M = ASCII form of "abc" = 61 62 63

K = 0EC982DA 1CAB8A86 692448F4 EB841BD6 CACB63E5

Π = (Π_a, Π_b)

Π_a = 647BA1C8 F922204C 5D0F4AD3 DD0E093B FF068454 D67B803C 82C0F082
876BC786 748E005C EBF2C2DA 67E2CA31 0CE29DED 0EFAF684 3C1660D0

549989A5 A2F46DF8

Π_b = 68DC5B24 5DB19947 E3A8A4C5 0AE4C90F 4C2B5AD5 F55DDFAB 61D46A5E
7BFA66B3 32431664 E1268632 159F22B6 5F745302 5FFD9E93 1314D23A
69D6E89D 759CF6BF

F.12.2.5 Ký

R = $BS2I(\gamma, H_2(M_1 \,||\, FE2BS(r, \Pi_a) \,||\, FE2BS(r, \Pi_b)))$ mod q =
06CD1062 D7B840D5 8B6973C5 7B075BF9 7877680F

S = (S_X, S_Y)

S_X = 6B59A8AB DB1752A5 46EC4F7E 8CB5FC6C 129BEEF3 726036CF 9CA55AE6
04A8EC16 05E987D9 C7234D7E 2D08A57A 42D79861 F350B298 51C3EF24
1331CEC7 B92FA579

S_Y = 4DD608CC 0A09C606 DFEFC33B 190F2CA5 E29E8811 49A54439 E437D8F0
91535531 30B03EE2 21322120 B6AE5964 9D0A11E7 492B0C32 A041029A
ED9E354E 03DBD353

F.12.2.6 Kiểm tra

Π' = (Π'_a, Π'_b)

Π'_a = 647BA1C8 F922204C 5D0F4AD3 DD0E093B FF068454 D67B803C 82C0F082
876BC786 748E005C EBF2C2DA 67E2CA31 0CE29DED 0EFAF684 3C1660D0
549989A5 A2F46DF8

Π'_b = 68DC5B24 5DB19947 E3ABA4C5 0AE4C90F 4C2B5AD5 F55DDFAB 61D46A5E
7BFA66B3 32431664 E1268632 159F22B6 5F745302 5FFD9E93 1314D23A
69D6E89D 759CF6BF

R' = 06CD1062 D7B840D5 8B6973C5 7B075BF9 7877680F

F.13 Cơ chế IBS-2

F.13.1 Ví dụ 1: Trường F_P, số nguyên tố P 512 bit, SHA-1

F.13.1.1 Tổng quan

Với ví dụ này, các phần tử trong trường F_p^2 được biểu diễn như $a + \sigma b$ với a và b là các phần tử trong trường F_p và σ là phần tử trong trường F_p^2 thỏa mãn $\sigma^2 + 1 = 0 \, mod \, p$. Ánh xạ xoắn được dùng để biến đổi một điểm trong một nhóm xoắn với các phần tử của nhóm xoắn khác để tạo thành các cặp không suy biến hoạt động như sau: $\phi P: (P_X, P_Y) \rightarrow (\beta P_X, P_Y)$ với $\beta \neq 1$ và $\beta \in F_p^2$ thỏa mãn $\beta^3 - 1 = 0 \, mod \, p$. <P,Q> được thực hiện như việc giảm cặp Tate ở hai điểm đầu vào P và ϕ (Q).

H_2 được thực hiện như SHA-1. H_1 được thực hiện như sau: đưa ra thông điệp M, $H_1(M) = I2P(BS2I(512, MGF1(M)))$ với $MGF1(M)$ là 512 bit trái nhất của $SHA-1(M \| 0) \| SHA-1(M \| 1) \| SHA-1(M \| 2) \| SHA-1(M \| 3)$.

F.13.1.2 Các tham số

Trường F_p với P là hệ thập lục phân

p = B35FA5FD E47FA1AB BB1E57E9 3BA1FF96 38B89B99 5C49BE81 A38E3194
A0983816 4EE51FB9 1D285832 F9A05D63 9C8D9680 10C93A35 27E561F2
FD6A45CC 70ABA1FB

Đường cong eliiptic $Y^2 = X^3 + 1$ trên trường F_p

q = 80000000 000FFFFF FFFFFFFF FFFFFFFF FFFFFFFF

β = (β_a, β_b)

β_a = 59AFD2FE F23FD0D5 DD8F2BF4 9DD0FFCB 1C5C4DCC AE24DF40 D1C718CA
504C1C0B 27728FDC 8E942C19 7CD02EB1 CE46CB40 08649D1A 93F2B0F9
7EB522E6 3855D0FD

β_b = 20C3F863 7D0F88AE 295531C6 82537070 5EFA0F99 39D1CDE4 F58A25F4
3722E229 5410CA05 B2360EA8 625C8203 626C5254 3E63404A B83448B0
3FDA177A 7A2DDB21

P = (P_X, P_Y)

P_X = 7EE6F118 9329ADB4 1E8CD405 2295F1A7 6096631D F065DD38 85FFF26B
8ED52022 7BFC3F3D 07F29CFE 093424BA 9DBD4C0D 73FFF367 3CE2C922
F0B73C50 2992093B

P_Y = 67C855C9 E6B617BE B24B792A E9C3E21E 95F37006 25B91058 3BC0B293
C36A762C FEBA4266 038989CC 59797235 C6116D99 8A97B805 FF82C664
53720E5C 1DE95E3E

U = 50BD36F5 967F41F4 93B1F667 4AD462F6 2B7D69D8

V = $[U]P = (V_X, V_Y)$

V_X = 04863127 7E2F50F6 CCCD235D EC856E8C FB22F17A 232D6229 7B7ABA44
77639ADE F2CC4863 5C19ED50 9E7F3F2C C5EF1F5B 5225E45D 30C269A3
97978783 D7E0DD86

V_Y = 75651DE4 8D8AEC70 10B4004A 05EE2EFB 0D72CF41 4F5B6113 539198F5
DC2317AA 3B6DEE3E E800394F 470020FB E6EE9811 CDDC5991 4BD0CB38
2E203F01 0816E8B7

F.13.1.3 Khóa ký và khóa kiểm tra

ID = "alice@network.com"

$MGF1(ID)$ = 85499895 C3B4F412 56E84BDC E3D76405 33612345 D6BF0725 FB7D0391
D53A680B 2DD96646 A32B244E DA614B5D 8E989340 473B7885 AF243CE0
0FBD2661 43B85636

I2P(BS2I(512,

Y_X = 8BB6EFE1 AF00D035 D0B286F1 94BC3128 A639949A B9D06405 D95F837A
E131C3C6 B37F2905 4D58FB21 9FA190B7 A65BE61F 4A6945B5 B4A1A36C
6166F047 2F63D303

Y_Y = 3D605915 84CF4C40 2E81145C 0047D9BE 2733A3C2 66F7A8F8 6CA2F50C
635424C3 8AFF64FA F9B5CD04 4AD0236F BC7BEB72 0D1909DA 5CC3C3FE
3022CDB7 7946F37D

X = $[U]Y = (X_X, X_Y)$

X_X = B26A00EA 12877F1D 049DC4F2 4030CDB1 632AA9FA D48E209D 8CC1A116
A159CBDD 540F417D FC584FF9 4C959E64 3BFA2131 7EB54356 EBB8650B
8803CB32 F607C87B

X_Y = 3527A3A6 E22290F9 0CC5ECBE F6CFE3DA F0F7EF16 327606C8 A2456C9F
6C7A2818 84832B75 41D8EAAE DB1B0FDA 78335B84 C56FEF1F E8B423C3
1393760B AE4520D1

F.13.1.4 Dữ liệu mỗi thông điệp

M = ASCII form of "abc" = 61 62 63

K = 1FF4E7C8 5257349D E625E51B 7374BF96 A732E142

$\Pi = (\Pi_X, \Pi_Y)$

Π_X = 509EF0AE 5E9161BF 37657765 213A5F0F 5BACAAF5 0D8487F1 71820AC4
79972203 06A8F304 9231377E 51402041 AC76953C A78AD3B5 AC37E9B3
34257AA7 66106EC5

Π_Y = 97A5D5ED 33F32DC1 D93FAF34 82AF186A CD947D5C DB3F900D AB720999
12789CD9 114AC35A C3887189 F74C36BF 0D3E8F9C 8F7AF48F 7FB2EB54
00DB8B69 1EC16528

H = $H_2(M||FE2BS(r, \Pi_X))$ = 223C3AA6 91EA9C1F 21065A74 C3FD2C0D 507ED8A6

F.13.1.5 Ký

$R = \Pi = (\Pi_X, \Pi_Y)$

Π_X = 509EF0AE 5E9161BF 37657765 213A5F0F 5BACAAF5 0D8487F1 71820AC4
79972203 06A8F304 9231377E 51402041 AC76953C A78AD3B5 AC37E9B3
34257AA7 66106EC5

Π_Y = 97A5D5ED 33F32DC1 D93FAF34 82AF186A CD947D5C DB3F900D AB720999
12789CD9 114AC35A C3887189 F74C36BF 0D3E8F9C 8F7AF48F 7FB2EB54
00DB8B69 1EC16528

$S = (S_X, S_Y)$

S_X = 0FD05023 373BA8E8 26715427 1A474C0D B57E88CC 70EA10AA 445436ED
8DC59EF4 86958DC6 64240FC6 32742CED 9FC96B36 925C0D8E ECD0FD19
CAA7054C FCD34157

S_Y = 1B909401 7DFE3D51 C330C2F5 8CD5F77A D84D97C2 EE7B4A6C 3451596C
99C21809 D422B5D5 4309CA08 3DE45DA7 9F823282 64C15B87 1741795C
EDFAA568 D996A7E0

F.13.1.6 Kiểm tra

$H = H_2(M||FE2BS(r, \Pi_X)) =$ 316A0C05 67D24048 1C653139 F7BEB3EE 4CA81252

$T^1 = <P, S> = (T^1_a, T^1_b)$

T^1_a = 89E32524 8BFFAB75 8C90FF00 46CFE55B 9EAE6AB3 1D1A859E A5295A51
13E4C043 857D2535 ABD88412 AFD36F56 DDE8F61C 28DE8235 635B70DB
B29A34B5 C1151EA5

T^1_b = B32755EC D8547A25 ADED38D0 FB19A23E 6F8042D9 3F871D0F D6599872
EAF31160 0397AC76 4410285F 002AB197 73779C26 C0AF7080 64AE2322
2CA73C2B 3FFA2CC3

$T = <V, \bar{\Pi} + [H]Y> = (T^2_a, T^2_b)$

T^2_a = 89E32524 8BFFAB75 8C90FF00 46CFE55B 9EAE6AB3 1D1A859E A5295A51
13E4C043 857D2535 ABD88412 AFD36F56 DDE8F61C 28DE8235 635B70DB
B29A34B5 C1151EA5

T^2_b = B32755EC D8547A25 ADED38D0 FB19A23E 6F8042D9 3F871D0F D6599872
EAF31160 0397AC76 4410285F 002AB197 73779C26 C0AF7080 64AE2322
2CA73C2B 3FFA2CC3

Phụ lục G
(Tham khảo)

So sánh các lược đồ chữ ký

G.1 Ký hiệu và từ viết tắt cho so sánh các lược đồ chữ ký

Với mục đích của phụ lục này, các ký hiệu và từ viết tắt sau được áp dụng:

exp	Phép lũy thừa modulo (mod p)
exp-G_2	Phép lũy thừa modulo trong G_2
GEXP	Phép nhân vô hướng
GEXP-G_1	Phép nhân vô hướng trong G_1
inv	Phép nghịch đảo (mod p)
pre-c	Phép toán có thể được tính trước
< >	Phép toán cặp

G.2 So sánh các lược đồ chữ ký

Bảng G.1 - So sánh các cơ chế dựa trên chứng thư số (dựa trên Z_p^*)

		DSA	KCDSA	Pointcheval/Vaudenay	SDSA
Độ dài khóa	Khóa ký	β bit	β bit	β bit	β bit
	Khóa kiểm tra	α bit	α bit	α bit	α bit
Độ dài chữ ký		2β bit	$\gamma + \beta$ bit	2β bit	$\gamma + \beta$ bit
Tính toán	Sinh khóa	1 exp	1 exp, 1 inv	1 exp	1 exp
	Sinh chữ ký	1 exp, 1 inv	1 exp	1 exp, 1 inv	1 exp
	Kiểm tra chữ ký	2 exp, 1 inv	2 exp	2 exp, 1 inv	2 exp

Bảng G.2 - So sánh các cơ chế dựa trên chứng thư số (sử dụng đường cong elliptic)

		EC-DSA	EC-KCDSA	EC-GDSA	EC-RDSA	EC-SDSA	EC-FSDSA
Độ dài khóa	Khóa ký	β bit	β bit	β bit	β bit	β bit	β bit
	Khóa kiểm tra[a]	2β bit	2β bit	2β bit	2β bit	2β bit	2β bit
Độ dài chữ ký		2β bit	$\gamma + \beta$ bit	2β bit	2β bit	$\gamma + \beta$ bit	3β bit
Tính toán	Sinh khóa	1 GEXP	1 GEXP, 1 inv	1 GEXP, 1 inv	1 GEXP	1 GEXP	1 GEXP
	Sinh chữ ký	1 GEXP, 1 inv	1 GEXP	1 GEXP	1 GEXP	1 GEXP	1 GEXP
	Kiểm tra chữ ký	2 GEXP, 1 inv	2 GEXP	2 GEXP, 1 inv	2 GEXP, 1 inv	2 GEXP	2 GEXP

[a] Kỹ thuật nén điểm không được sử dụng.

CHÚ THÍCH 1 Độ dài của chữ ký EC-FSDSA là $2\beta'+\beta$ với β' là β được làm tròn là bội của 8. Đây là định nghĩa giả tạo của FE2BS, mà các xâu đầu ra luôn luôn là các số nguyên là bội của 8 bit dài.

CHÚ THÍCH 2 GEXP là viết tắt của phần tử nhóm exp, nó được gọi là phép nhân vô hướng.

Bảng G.3 - So sánh các cơ chế dựa trên chứng thư số

		IBS-1	IBS-2
Độ dài khóa	Khóa chủ bí mật	β bit	β bit
	Khóa chủ công khai	$\lceil \alpha/m \rceil$ bit	$\lceil \alpha/m \rceil$ bit
	Khóa ký	$\lceil \alpha/m \rceil$ bit	$\lceil \alpha/m \rceil$ bit
	Khóa kiểm tra	$\lceil \alpha/m \rceil$ bit	$\lceil \alpha/m \rceil$ bit
Độ dài chữ ký		$\beta + \lceil \alpha/m \rceil$ bit	$2^*\lceil \alpha/m \rceil$
Tính toán	Sinh khóa	1 GEXP - G_1	1 GEXP - G_1
	Sinh chữ ký	1< > (pre-c) 1 exp - G_2 1 GEXP - G_1	2 GEXP - G_1
	Kiểm tra chữ ký	1< > 1 < > (pre-c) 1 exp - G_2	1 GEXP - G_1 2< >

Phụ lục H
(Tham khảo)

Các yêu cầu đặc điểm cho việc lựa chọn một cơ chế

Bảng dưới đây cung cấp các ưu điểm có thể có của các cơ chế chữ ký khác nhau

Bảng H.1 - Yêu cầu đặc điểm cho các cơ chế

Cơ chế chữ ký	Yêu cầu đặc điểm
DSA	U.S.FIPS, được áp dụng rộng rãi
KCDSA	Tiêu chuẩn Hàn quốc với độ an toàn chứng minh được [12],[41] không nghịch đảo (mod Q) trong việc sinh và xác thựcchữ ký
Pointcheval-Vaudenay	Biến DSA với độ an toàn chứng minh được
SDSA	Phổ biến rộng rãi, độ an toàn chứng minh được [30], [32], [33] hiệu quả, không cần nghịch đảo
EC-DSA	U.S.FIPS và chuẩn ANSI với yêu cầu an toàn là chứng minh được, [13], [14] được áp dụng rộng rãi, hiệu quả lưu trữ, tiết kiệm băng thông, hiệu quả tính toán
EC-KDSA	Chuẩn Hàn quốc với yêu cầu an toàn là chứng minh được, [12], [41], không nghịch đảo (mod Q) trong việc sinh và xác thực chữ ký, hiệu quả lưu trữ, tiết kiệm băng thông, hiệu quả tính toán
EC-GDSA	Chuẩn Đức, không nghịch đảo (mod Q) trong việc sinh chữ ký, hiệu quả lưu trữ, tiết kiệm băng thông, hiệu quả tính toán
EC-RDSA	Tiêu chuẩn Nga GOST R 34.10-2012 và Tiêu chuẩn liên Chính phủ GOST 34.310-2004 Với yêu cầu an toàn chứng minh được (không ngẫu nhiên Oracles), hiệu quả lưu trữ, tiết kiệm băng thông, và Hiệu quả tính toán
EC-SDSA	Phiên bản EC SDSA, tính chất tương tự như SDSA
EC-FSDSA	Biến thể Schnorr, không nghịch đảo (mod q) có thể hữu ích với một số thuộc tính kiểm tra

Thư mục tài liệu tham khảo

[1] TCVN 7817-3 Công nghệ thông tin – Các kỹ thuật an toàn – Quản lý khóa — Phần 3: Các cơ chế sử dụng kỹ thuật phi đối xứng.

[2] TCVN 11367-3, Công nghệ thông tin – Các kỹ thuật an toàn – Thuật toán mật mã – Phần 2: Mật mã phi đối xứng

[3] ISO/IEC 15945:2002, Information technology — Security techniques — Specification of TTP services to support the application of digital signatures

[4] ISO/IEC 15946-1:2008, Information technology — Security techniques — Cryptographic techniques based on elliptic curves — Part 1: General

[5] ISO/IEC 18032:2005, Information technology — Security techniques — Prime number generation

[6] ISO/IEC 9594-8:2001 I ITU-T Rec. X.509 (2002 E), Information technology — Open Systems Interconnection — The Directory: Public-key and attribute certificate frameworks

[7] American National Standards Institute, Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA), ANSI X9.62-2005, 2005

[81 BARRET() P., KIM H., LYNN B., SCOTT M. Efficient algorithms for pairing-based cryptosystems, Proceedings of CRYPTO 2002, LNCS 2442, pages 354-369, Springer-Verlag, 2002

[9] BELLARE M., GARAY J., RABIN T. Fast Batch Verification for Modular Exponentiation and Digital Signatures, Advances in Cryptology — Eurocrypt 98 Proceedings, LCNS, Vol. 1403, pp. 236-250, Springer-Verlag, 1998

[10] BOHLI J.M., ROHRICH S., STEINWANDT R. Key substitution attacks revisited: taking into account malicious signers. Int. J. Inf. Secur. 2006, 5 pp. 30-36

[11] BONEIT D., & FRANKLIN M. Identity based encryption from the Weil pairing, Proceedings of CRYPTO 2001, LNCS 2139, pp. 213-229, Springer-Verlag, 2001

[12] BRICKELL E., POINTCHEVAL D., VAUDENAY S., YUNG M. Design Validations for Discrete Logarithm Based Signature Schemes, Proceedings of PKC 2000, LNCS 1751, pp. 276-292, Springer-Verlag, 2000

[13] BROWN D.R.L. Generic Groups, Collision Resistance, and ECDSA. Des. Codes Cryptogr., 35 (1) pp. 119-152

[14] BROWN D.R.L. In: On the Provable Security of ECDSA, "Advances in Elliptic Curves ryptography.

(BLAKE I., SEROUSSI G., SMART N. eds.). Cambridge University Press, Chapter II. 2005

[15] CHA J.C., & CHEON J.H. An identity-based signature from gap Diffie-Hellman groups, Proceedings of PKC 2002, LNCS 2567, pp. 18-30, Springer-Verlag, 2002

[16] ERWIN H., & PASCALE S. Digital Signature Scheme EC-GDSA, German Federal Office for Information Security, December 2005

[17] FIPS PUB 186-4, Digital Signature Standard (DSS). U.S. National Institute of Standards and Technology, Gaithersburg, Maryland, 2013

[18] FREY G., MULLER M., ROCK H. The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems. IEEE Trans. Inf. Theory. 1999, 45 (5) pp. 1717-1719

[19] GALBRAITH S. Supersingular curves in cryptography, Proceedings of Asiacrypt 2001, LNCS 2248. Springer-Verlag, 2001, pp. 495-513 130 ISO/IEC 2016 - All rights reservedISO/IEC 14888-3:2016(E)

[20] GALBRAITH S., HARRISON K., SOLDERA D. Implementing the Tate-pairing, Proceedings of ANTS-V, LNCS 2369, pp. 324-337, Springer-Verlag, 2002

[21] GOST R 34.10-2012, State Standard of the Russian Federation, "Information technology Cryptographic data security Signature and verification processes of [electronic] digital signature." State Committee of the Russian Federation on Standards and Metrology, 2012. (In Russian)

[22] HESS F. Efficient identity based signature schemes based on pairings, Proceedings of SAC 2002, 2002

[23] IEEE P1363, Standard Specifications for public key cryptography

[24] KOBLITZ N. Elliptic Curve Cryptosystems. Math. Comput. 1987, 48 pp. 203-209

[25] LENSTRA A.K., & VERHEUL E.R. Selecting cryptographic key sizes, in Journal of Cryptology, Vol. 14-4, pp. 255-293, 2001

[26] MENEZES A. Elliptic Curve Public Key Cryptosystems. Kluwer Academic Publishers, 1993

[27] MENEZES A.J., OKAMOTO T., VANSTONE S. Reducing elliptic curve logarithms to logarithms in a finite field. IEEE Trans. Inf. Theory. 1993, 39 pp. 1639-1646

[28] MILLER V.S. Use of Elliptic Curves in Cryptology, Proceedings of Crypto 1985, H.C. Williams, Ed., LNCS 218, pp. 417-426, Berlin, Springer-Verlag, 1986

[29] MILLER V. The Weil pairing, and its efficient calculation, Journal of Cryptology, Vol. 17-4, pp. 235-261, 2004

[30] NEVEN G., SMART N., WARINSCHI B. Hash function requirements for Schnorr signatures. Journal of Mathematical Cryptology. 2009, 3 pp. 69-87

[31] POINTCHEVAL D., & VAUDENAY S. On provable security for digital signature algorithms, Technical Report LIENS-96-17, LIENS, 1996

[32] SCHNORR C.P. Efficient identification and signature for smart cards. In Advances in Cryptology Crypto'89, LNCS 435, pp.239-252, Springer-Verlag, 1990

[33] SCHNORR C.P. Efficient signature generation for smart cards. J. Cryptol. 1991, 4 pp. 161-174

[34] SILVERMAN R. A cost-based security analysis of symmetric and asymmetric key lengths, RSA Labs Bulletin, Vol. 13, April 2000 (revised November 2001)

[35] SILVERMAN J.H. Advanced topics in the arithmetic of elliptic curves, GTM 151. SpringerVerlag, 1994

[36] TTAK.KO-12.0001/R3, Digital Signature Mechanism with Appendix - Part 2: Korean Certificatebased Digital Signature Algorithm KCDSA, 2014. (In Korean)

[37] TTAK.KO-12.0015/R2, Digital Signature Mechanism with Appendix - Part 3: Korean Certificatebased Digitial Signature Algorithm using Elliptic Curves EC-KCDSA, 2014. (In Korean)

[38] VARNOVSKII N.P. Provable security of digital signatures in the tamper-proof device model. Discrete Math. Appl. 2008, 18 (4) pp. 427-437

[39] VAUDENAY S. Hidden Collisions on DSS, Proceedings of CRYPTO 1996, LNCS 1109, pp. 83-88, Springer-Verlag, 1996

[40] YEN S., & LAIH C. Improved Digital Signature Suitable for Batch Verification. IEEE Trans. Comput. 1995, 44 (7) pp. 957-959

[41] YUM D.H., & LEE P.J. Security Proof for KCDSA under the Random Oracle Model. Proceedings of CISC. 1999, 1999 pp. 173-180