QCVN 5 : 2016/BQP

QUY CHUẨN KỸ THUẬT QUỐC GIA VỀ CHỮ KÝ SỐ SỬ DỤNG TRONG LĨNH VỰC NGÂN HÀNG

National technical regulation on digital signature used in banking

MỤC LỤC

Lời nói đầu

QCVN 5 : 2016/BQP do Cục Quản lý mật mã dân sự và Kiểm định sản phẩm mật mã - Ban Cơ yếu Chính phủ biên soạn, Ban Cơ yếu Chính phủ trình duyệt, Bộ Khoa học và Công nghệ thẩm định và được ban hành theo Thông tư số 161/2016/TT-BQP ngày 21 tháng 10 năm 2016 của Bộ trưởng Bộ Quốc phòng.

QUY CHUẨN KỸ THUẬT QUỐC GIA VỀ CHỮ KÝ SỐ SỬ DỤNG TRONG LĨNH VỰC NGÂN HÀNG

National technical regulation on digital signature used in banking

1. QUY ĐỊNH CHUNG

1.1. Phạm vi điều chỉnh

Quy chuẩn kỹ thuật quốc gia này quy định mức giới hạn của các đặc tính kỹ thuật mật mã của chữ ký số sử dụng trong lĩnh vực ngân hàng.

1.2. Đối tượng áp dụng

Quy chuẩn này áp dụng đối với các doanh nghiệp kinh doanh sản phẩm, dịch vụ mật mã dân sự trong lĩnh vực ngân hàng; các tổ chức tín dụng (trừ quỹ tín dụng nhân dân cơ sở có tài sản dưới 10 tỷ, tổ chức tài chính vi mô) sử dụng sản phẩm, dịch vụ mật mã dân sự.

1.3. Tài liệu viện dẫn

- *Recommendation for Random Number Generation Using Deterministic Random Bit Generators*, NIST SP 800-90A Rev. 1, National Institute of Standards and Technology, June 2015. *(Khuyến cáo cho bộ sinh số ngẫu nhiên sử dụng bộ sinh bit ngẫu nhiên tất định,* NIST SP 800-90A Rev. 1, Viện tiêu chuẩn và công nghệ quốc gia (Mỹ), tháng 6 năm 2015).

- RSA Laboratories. *PKCS#1 v2.1: RSA Cryptography Standard.* June 2002. (Phòng thí nghiệm RSA. *PKCS#1 v2.1: Tiêu chuẩn mật mã RSA.* Tháng 6 năm 2002).

- TCVN 7635:2007 Kỹ thuật mật mã - Chữ ký số.

- NIST, Federal Information Processing Standards Publication 186-4, *Digital Signature Standard (DSS)*, July 2013. (NIST, Tiêu chuẩn xử lý thông tin liên bang, *Chuẩn chữ ký số (DSS),* tháng 7 năm 2013).

- NIST, Federal Information Processing Standards Publication 180-4, *Secure Hash Standard (SHS)*, August 2015. (NIST, Tiêu chuẩn xử lý thông tin liên bang, Chuẩn *hàm băm an toàn (SHS)*, tháng 8 năm 2015).

1.4. Giải thích từ ngữ

Trong Quy chuẩn này, các từ ngữ dưới đây được hiểu như sau:

1.4.1. Thông tin không thuộc phạm vi bí mật nhà nước

Là thông tin không thuộc nội dung tin "tuyệt mật", "tối mật" và "mật" được quy định tại Pháp lệnh Bảo vệ bí mật nhà nước ngày 28 tháng 12 năm 2000.

1.4.2. Mật mã

Là những quy tắc, quy ước riêng dùng để thay đổi hình thức biểu hiện thông tin nhằm bảo đảm bí mật, xác thực, toàn vẹn của nội dung thông tin.

1.4.3. Mật mã dân sự

Là kỹ thuật mật mã và sản phẩm mật mã được sử dụng để bảo mật hoặc xác thực đối với thông tin không thuộc phạm vi bí mật nhà nước.

1.4.4. Sản phẩm mật mã dân sự

Là các tài liệu, trang thiết bị kỹ thuật và nghiệp vụ mật mã để bảo vệ thông tin không thuộc phạm vi bí mật nhà nước.

1.4.5. Kỹ thuật mật mã

Là phương pháp, phương tiện có ứng dụng mật mã để bảo vệ thông tin.

1.4.6. Mã hóa

Phép biến đổi (khả nghịch) dữ liệu bởi thuật toán mật mã để tạo ra bản mã, tức là che giấu nội dung thông tin của dữ liệu.

14.7. Giải mã

Phép toán ngược với phép mã hóa tương ứng.

1.4.8.

Mật mã phi đối xứng

Hệ thống dựa trên kỹ thuật mật mã phi đối xứng, trong đó phép biến đổi công khai được sử dụng để mã hóa, phép biến đổi bí mật được sử dụng để giải mã.

1.4.9. Kỹ thuật mật mã phi đối xứng

Kỹ thuật mật mã phi đối xứng sử dụng hai phép biến đổi liên quan đến nhau, phép biến đổi công khai (được xác định bởi khóa công khai) và phép biến đổi bí mật (được xác định bởi khóa riêng). Cả hai phép biến đổi này có tính chất là cho biết phép biến đổi công khai, về mặt tính toán không thể có khả năng xác định được phép biến đổi bí mật.

1.4.10. Chữ ký số

Một chuỗi số, kết quả của phép biến đổi mật mã trên thông điệp dữ liệu nhằm cung cấp một phương tiện để kiểm tra tính xác thực của nguồn gốc thông điệp dữ liệu, tính toàn vẹn của dữ liệu và tính không thể chối bỏ của người đã ký.

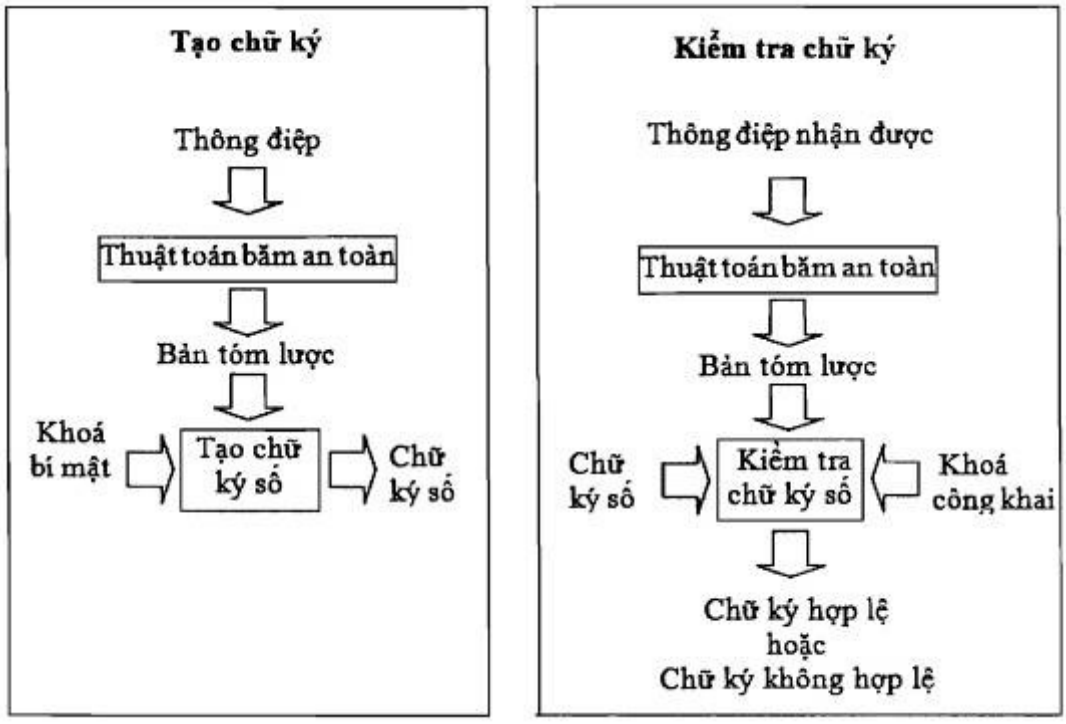
1.5. Các ký hiệu

AES	Tiêu chuẩn mã hóa tiên tiến *(Advanced Encryption Standard)*
Octet	Chuỗi bit có độ dài bằng 8
n	Modulo RSA
d	Số mũ bí mật RSA
e	Số mũ công khai RSA
(n,e)	Khóa công khai RSA của người ký
c	Một biểu diễn của bản mã, là số nguyên thuộc (0,*n*-1)
C	Bản mã được biểu diễn ở dạng chuỗi octet
EM	Chuỗi Octet biểu diễn thông điệp đã được ghi mã

emLen	Độ dài theo Octet của *EM*
k	Độ dài modulo *n* tính theo octet
m	Một biểu diễn của thông điệp (văn bản), là số nguyên thuộc (0,*n-1)*
M	Thông điệp (văn bản), chuỗi octet
I2OSP	Hàm cơ sở chuyển đổi từ dạng số nguyên sang chuỗi octet *(Integer-to-Octet-String Primitive)*
OS2IP	Hàm cơ sở chuyển đổi từ chuỗi octet sang số nguyên *(Octet- String-to-Integer-Primitive)*
LCM	Bội chung nhỏ nhất *(Least Common Multiplier)*
nlen	Độ dài modulo *n* theo bit
security_strength	độ mạnh về an toàn *(security_strength)* là một số nguyên biểu thị lượng tính toán cần thiết để phá hệ mã
N	là ký hiệu độ dài theo bit của số nguyên tố *q*
L	là ký hiệu độ dài theo bit của số nguyên tố *p*
||	Toán tử nối hai chuỗi
PKCS	Tiêu chuẩn mật mã khóa công khai *(Puplic Key Cryptography Standard)* do Phòng thí nghiệm RSA (Mỹ) ban hành.
PSS	Lược đồ ký xác suất *(Probabilistic Signature Scheme)*
RSA	Tên của hệ mã do ba nhà toán học Rivest, Shamir và Adleman RSA sáng tạo ra
RSAVP	Phép toán cơ sở phục vụ cho kiểm tra chữ ký RSA
RSASP	Phép toán ký RSA cơ sở
RSASSA	*Lược đồ ký RSA kèm phụ lục (RSA Signature Scheme with Appendix)*
SHA	Thuật toán băm an toàn *(Secure Hash Algorithm).*
Word	Từ (32 bit)
DSA	Thuật toán chữ ký số
EC	Đường cong Elliptic
ECDSA	Thuật toán chữ ký số dựa trên đường cong Elliptic

2. QUY ĐỊNH KỸ THUẬT

2.1. Chữ ký số



Hình 1 - Mô tả quá trình tạo và kiểm tra chữ ký số

Chữ ký số là một dạng chữ ký điện tử được tạo ra bằng sự biến đổi một thông điệp dữ liệu sử dụng hệ thống mật mã phi đối xứng theo đó người có được thông điệp dữ liệu ban đầu và khóa công khai của người ký có thể xác định được chính xác:

- Việc biến đổi nêu trên được tạo ra bằng đúng khóa bí mật tương ứng với khóa công khai trong cùng một cặp khóa;

- Sự toàn vẹn nội dung của thông điệp dữ liệu kể từ khi thực hiện việc biến đổi nêu trên.

Quá trình sinh chữ ký số trên một thông điệp dữ liệu yêu cầu sử dụng: 1) hàm băm mật mã thực hiện tính toán trên dữ liệu sẽ được ký, 2) sử dụng khóa mật mã và thuật toán ký để tạo chữ ký số trên đầu ra của hàm băm. Quy chuẩn này quy định sử dụng khóa mật mã trong thuật toán ký số, hàm băm mật mã được sử dụng trong quá trình sinh chữ ký số.

2.1.1. Quy định kỹ thuật

2.1.1.1. Quy định chung:

Quá trình chữ ký số	Sử dụng	Ghi chú
Tạo chữ ký số	Độ an toàn ≥ 112 bit: DSA: $\|p\| \geq 2048$ và $\|q\| \geq 224$ RSA: $\|n\| \geq 2048$ EC: $\|n\| \geq 224$	
Xác thực chữ ký số	Độ an toàn ≥ 112 bit: DSA: $\|p\| \geq 2048$ và $\|q\| \geq 224$ RSA: $\|n\| \geq 2048$ EC: $\|n\| \geq 224$	

2.1.1.2. Quy định chi tiết về nguồn ngẫu nhiên:

Các số ngẫu nhiên được sử dụng cho các mục đích khác nhau như để sinh các tham số mật mã, các khóa mật mã, các giá trị ngẫu nhiên dùng một lần và các giá trị thách đố xác thực.

Một số bộ sinh bit ngẫu nhiên tất định DRBG được chấp thuận để sử dụng theo quy định chung bao gồm: HASH_DRBG, HMAC_DRBG và CTR_DRBG.

Các bộ sinh bit ngẫu nhiên RBG tuân theo SP800-90A phiên bản sửa đổi lại năm 2015 để sinh bit ngẫu nhiên cũng được chấp thuận để sử dụng tiếp.

2.1.2. Chữ ký số RSA-PSS

2.1.2.1. Các yêu cầu chung

1. Cặp khóa RSA dùng để ký thì không được dùng cho mục đích khác (chẳng hạn dùng lại để mã thông điệp);

2. Hai số nguyên tố *p*, *q* và số mũ bí mật *d* cần phải được giữ bí mật tránh việc bị truy cập bất hợp pháp, làm lộ hoặc sửa đổi. Modulo *n* và số mũ công khai *e* phải được công bố công khai;

3. Mỗi người sử dụng cần có Modulo *n* riêng;

4. Độ dài của Modulo *n* (*nlen*) không được nhỏ hơn 2048 bit và nên được thay đổi theo thời gian như sau.

Năm (y)	độ an toàn	***nlen* tối thiểu**
y<2020	112	2048
	128	3072

Trong đó, độ mạnh về an toàn (*security_strength*) là một số nguyên biểu thị lượng tính toán cần thiết đề phá hệ mã.

Vì các phương pháp phá hệ mã thường xuyên được hoàn thiện nên cần phải định kỳ 3 đến 5 năm một lần xem xét lại *nlen* tối thiểu (có thể tham khảo chi tiết yêu cầu này trong tài liệu *NIST Special Publication 800-57: Recommendation for Key Management - Part1: General, January 2016*).

- Phiên bản áp dụng: Áp dụng phiên bản 2.1 của tiêu chuẩn RSA Cryptography Standard PKCS #1 v2.1.

- Áp dụng lược đồ RSAES-OAEP để mã hóa và RSASSA-PSS để ký.

2.1.2.2. Quy định chi tiết về các khóa RSA

1. Số mũ công khai *e* cần phải được chọn với các ràng buộc sau:

a) Số mũ công khai *e* cần được chọn trước khi tạo số mũ bí mật *d*;

b) Số mũ công khai *e* cần phải là số nguyên dương lẻ sao cho

$$65{,}537 \le e < 2^{nlen-2security\_strength}$$

Với *nlen* là độ dài của modulo *n* theo bit.

Chú ý rằng *e* có thể là giá trị bất kỳ mà thỏa mãn ràng buộc 1(b); *p* và *q* sẽ được chọn (trong mục 2) sao cho *e* là nguyên tố cùng nhau với cả (*p* - 1) và (*q* - 1).

2. Hai số nguyên tố *p* và q được tạo ngẫu nhiên và giữ bí mật cần phải được chọn với các ràng buộc sau:

a) (*p* - 1) và (*q* - 1) cần phải là nguyên tố cùng nhau với số mũ công khai *e*;

b) Mỗi một trong bốn số (*p* + 1), (*p* - 1) và (*q* + 1), (*q* - 1) cần phải có các nhân tử nguyên tố lớn hơn $2^{security-strength+20}$;

c) Nhân tử nguyên tố bí mật *p, q* cần phải được chọn ngẫu nhiên từ các số nguyên tố thỏa mãn ($\sqrt{2}$) $(2^{(nlen/2)-1}) \le q < p \le (2^{(nlen/2)} - 1)$;

d) $|p - q| > 2^{(nlen/2-100)}$.

3. Số mũ bí mật *d* cần phải được lựa chọn sau khi tạo *p* và *q* với các ràng buộc:

a) Số mũ *d* cần phải lớn hơn $2^{(nlen/2)}$, và

b) $d = e^{-1} mod(LCM((p - 1), (q - 1)))$

(Chi tiết về hàm tạo các tham số RSA có thể tham khảo trong tài liệu *FIPS 186-4: Digital Signature Standard).*

2.1.2.3. Tạo chữ ký số

RSASSA - PSS - SIGN(K, M)

Đầu vào:	*K*	khóa bí mật RSA của người ký
	M	thông điệp sẽ được ký, là một chuỗi octet
Đầu ra:	*S*	chữ ký, chuỗi octet có độ dài *k*, với *k* là độ dài của modulo RSA theo octet
Thông báo lỗi:		"văn bản quá dài", "lỗi định dạng"

Các bước:

1. Mã hóa *EMSA - PSS*: Áp dụng thao tác *EMSA - PSS - ENCODE* vào văn bản *M* để tạo ra thông điệp được định dạng *EM* có độ dài (*modBits*–1)/8 octet sao cho độ dài bit của số nguyên *OS2IP*(EM) nhiều nhất là *modBits*- 1, với *modBits* là độ dài theo bit của số *n* (modun RSA):

$$EM = EMSA - PSS - ENCODE (M, modBits - 1).$$

Chú ý rằng độ dài octet của EM sẽ bằng *k* -1 nếu *modBits* - 1 chia hết cho 8 và bằng *k* nếu *modBits* -1 không chia hết cho 8. Nếu hàm *EMSA - PSS - ENCODE* cho ra thông báo lỗi "văn bản quá dài" thì *RSASSA - PSS - SIGN* cũng cho ra thông báo lỗi "văn bản quá dài" và dừng lại. Nếu *EMSA - PSS - ENCODE* cho ra thông báo "lỗi định dạng" thì *RSASSA - PSS - SIGN* cũng cho ra thông báo "lỗi định dạng" và dừng lại.

2. Chữ ký *RSA*:

a. Chuyển thông điệp đã được định dạng (chuỗi octet) *EM* thành biểu diễn thông điệp ở dạng số nguyên *m*.

$$m = OS2IP (EM).$$

b. Áp dụng phép toán cơ sở *RSASP* với *K* là khóa bí mật RS*A* và *B*iểu diễn thông điệp *m* để tạo ra biểu diễn chữ ký là số nguyên *s*:

$$s = RSASP(K,m).$$

c. Chuyển chữ ký *s* dạng số nguyên thành chữ ký *S* dạng chuỗi octet có độ dài *k*:

$$s = I2OSP(s,k).$$

Đưa ra chữ ký *S*.

2.1.2.4. Xác thực chữ ký số

RSASSA - PSS - VERIFY((n, e), M, s)

Đầu vào: *(n, e)* khóa công khai RSA của người ký

M thông điệp mà chữ ký của nó cần được kiểm tra, là chuỗi octet

S chữ ký được kiểm tra, chuỗi octet có độ dài *k*, với *k* là độ dài theo octet của số *n*, modulo RSA

Đầu ra: "chữ ký hợp lệ" hoặc "chữ ký không hợp lệ"

Các bước:

1. Kiểm tra độ dài: Nếu độ dài của chữ ký *S* không là *k* octet, cho ra thông báo lỗi "chữ ký không hợp lệ" và dừng lại;

2. Kiểm tra chữ ký RSA;

a. Chuyển chữ ký *S* thành biểu diễn chữ ký ở dạng số nguyên *s*;

$$s = OS2IP(S)$$

b. Áp dụng phép toán cơ sở RSAVP với khóa công khai RSA là (*n, e*) và biểu diễn chữ ký *s* để tạo ra *m* là số nguyên biểu diễn thông điệp;

$$m = RSAVP((n,e),s)$$

c. Chuyển biểu diễn thông điệp *m* thành thông điệp đã được định dạng *EM* có độ dài *emLen* = (*modBits* - 1)/8 octet, với *modBits* là độ dài theo bit của số *n* (Modulo RSA):

$$EM = I2OSP\ (m,\ emLen)$$

Chú ý rằng *emLen* sẽ bằng *k* - 1 nếu *modBits* - 1 chia hết cho 8 và bằng *k* nếu *modBits* - 1 không chia hết cho 8. Nếu *I2OSP* cho ra thông báo lỗi "số nguyên quá lớn" thì *RSASSA - PSS - VERIFY* cho ra thông báo lỗi "chữ ký không hợp lệ" và dừng lại.

3. *Kiểm tra EMSA - PSS:* Áp dụng thao tác kiểm tra *EMSA - PSS - VERIFY* vào thông điệp *m* và thông điệp đã được định dạng *EM* để xác định xem chúng có tương ứng với nhau hay không;

$$Result = EMSA - PSS - VERIFY\ (M,EM,modBits - 1).$$

4. Nếu kết quả (Result) là "phù hợp" thì cho ra "chữ ký hợp lệ". Ngược lại sẽ cho ra "chữ ký không hợp lệ".

2.1.3. Chữ ký số ECDSA

2.1.3.1. Quy định chi tiết về các khóa ECDSA:

Kiểm tra tính hợp lệ của các tham số miền *(p,SEED, a, b, G, n, h)* như sau:

Xâu *SEED* dùng để sinh ngẫu nhiên đường cong Elliptic xác định trên trường F_p với *p* là số nguyên tố lẻ.

Trước khi sử dụng một bộ tham số miền, tính hợp lệ của nó phải được kiểm tra theo thuật toán dưới đây:

1. Kiểm tra *p* là một số nguyên tố lẻ.

2. Kiểm tra *a, b,* x_G, y_G là các phần tử của trường F_p.

3. Kiểm tra rằng *a* và *b* được dẫn xuất tương ứng từ *SEED*.

4. Kiểm tra ($4a^3 + 27b^2$) khác 0 và $j(E) \neq 0$; 1728 trong F_p

5. Kiểm tra $y_G^2 = x_G^3 + ax_G + b$ trong F_p.

6. Kiểm tra *n* là nguyên tố và $n > 4\sqrt{p}$.

7. Kiểm tra $nG = 0_E$.

8. Kiểm tra đường cong có thuộc danh sách các đường cong yếu:

a. Thỏa mãn điều kiện MOV, (chú ý rằng một đường cong thỏa mãn điều kiện MOV sẽ không phải là đường cong siêu biến).

b. Kiểm tra đường cong không kì dị, nghĩa là $\#E \neq p$.

Nếu bất kỳ sự kiểm tra nào ở trên thất bại thì tham số miền phải được xem là không hợp lệ.

Điều kiện MOV được hiểu là không có giá trị *k* nguyên dương nào $0 < k < B$ để cho $p^k - 1$ chia hết cho *n*. Trên thực hành hiện nay $|p| = 224$ bit thì người ta xét với $B = 15$ là đủ vì khi đó $|p^k| = 3360 > 2048$.

Các hệ số *a,b* của đường cong được sinh ngẫu nhiên trên F_p từ đầu vào *SEED* và có thể kiểm tra được.

Khóa bí mật *d* phải được sinh ngẫu nhiên trong khoảng [1,*n* - 1].

Đường cong Elliptic xác định trên trường hữu hạn F_p với tối thiểu |*p*| = 224 bit và được xác định cụ thể như sau:

Độ dài bit của n	**Độ dài bit của p**
224 - 255	|p| = 224
256 - 383	|p| = 256
384-511	|p| = 384
≥ 512	|p| = 521

Đại lượng Cofactor được định nghĩa và ký hiệu là $h = \#E(F_p)/n$ tuân theo bảng dưới đây:

Độ dài bit của n	**Giá trị h cực đại cho phép**
224 - 255	2^{14}
256 - 383	2^{16}
384 - 511	2^{24}
≥ 512	2^{32}

2.1.3.2. Tạo chữ ký số

Thuật toán Chữ ký số Đường cong Elliptic (ECDSA) thực hiện việc sinh chữ ký của thông báo *m*, làm việc như sau:

Sinh chữ ký *ECDSA*

Đầu vào: Các tham số miền *(E,P)*, khóa bí mật *d*, thông báo *m*.

Đầu ra: Chữ ký *(r, s).*

1. Lấy $0 < k < q$ một cách ngẫu nhiên
2. $(x_R,y_R) \leftarrow kP$
3. $r \leftarrow x_R \bmod q$
4. Nếu $r = 0$ thì chuyển về bước 1
5. $k \leftarrow k^{-1} \bmod q$
6. $e \leftarrow H(m)$
7. $s \leftarrow k(e + rd) \bmod q$
8. Nếu $s = 0$ thì quay về bước 1
9. Trả về *(r,s)*

2.1.3.3. Xác thực chữ ký số

Để kiểm tra chữ ký số (*r,s*) của thông báo *m*, người kiểm tra tính các bước sau ($Q = dP$ - ký hiệu khóa công khai):

Kiểm tra chữ ký *ECDSA*

Đầu vào: Các tham số miền (*E,P*), khóa công khai *Q*, thông báo *m*, chữ ký *(r,s)*

Đầu ra: Chấp nhận hoặc bác bỏ chữ ký.

1. Kiểm tra rằng $0 < r,s < q$
2. $s' \leftarrow s^{-1} \bmod q$
3. $e \leftarrow H(m)$
4. $h_1 \leftarrow s'e \bmod q$
5. $h_2 \leftarrow s'r \bmod q$
6. $R = (x_R, y_R) \leftarrow h_1P + h_2Q$
7. Nếu $R = 0$ thì bác bỏ chữ ký.
8. Nếu $x_R \bmod q = r$ thì chấp nhận, ngược lại thì bác bỏ.

2.1.4. Chữ ký số DSA

2.1.4.1. Quy định chi tiết về các khóa DSA: