MẬT MÃ ỨNG DỤNG TRONG AN TOÀN THÔNG TIN

Bài 06. Chuẩn mật mã RSA

1. Giới thiệu chung
2. Khóa RSA và các phép biến đổi cơ sở
3. Lược đồ mã hóa
4. Lược đồ ký số
5. Tiêu chuẩn tham số

1. Giới thiệu chung
2. Khóa RSA và các phép biến đổi cơ sở
3. Lược đồ mã hóa
4. Lược đồ ký số
5. Tiêu chuẩn tham số

Thuật toán RSA nguyên thủy

$$c = m^e \bmod n$$

$$m = c^d \bmod n$$

- Khóa công khai $K_P = (n, e)$
- Khóa bí mật $K_S = (n, d)$

4

Tiêu chuẩn về RSA

PKCS#1. Ver 1.0-2.2. RSA Cryptography Standard

TCVN 7635:2007. Tiêu chuẩn mật mã – Chữ kí số

5

Tiêu chuẩn về RSA

- Khóa RSA
- Hàm chuyển đổi dữ liệu cơ sở I2OSP, OS2IP
- Phép mã hóa, giải mã cơ sở RSAEP, RSADP
- Phép ký số và kiểm tra chữ ký số cơ sở RSASP, RSAVP
- Lược đồ mã hóa và giải mã
- Lược đồ ký số và kiểm tra chữ ký số
- Lược đồ định dạng (encode) dữ liệu
- Cú pháp ASN.1 để biểu diễn khóa và xác định lược đồ

6

Khóa RSA và các phép biến đổi cơ sở

❑**Khóa RSA**

- Khóa công khai: $K_P = (n, e);\quad GCD(e, \lambda(n)) = 1;$
  $\lambda(n) = LCM(p-1, q-1)$
- Khóa bí mật dạng bộ 2:
  $K_S = (n, d);\quad de \equiv 1 \mod \lambda(n)$
- Khóa bí mật dạng bộ 5:
  $(p, q, dP, dQ, qInv);\quad d > q;\quad q \cdot qInv \equiv 1 \mod p$
  $e \cdot dP \equiv 1 \mod (p-1);\quad e \cdot dQ \equiv 1 \mod (q-1)$

8

Khóa RSA và các phép biến đổi cơ sở

❑**Hàm chuyển đổi dữ liệu cơ sở**

- OS2IP: Octet String To Integer Primitive

x = OS2IP(X)

$$X = X_1 X_2 \ldots X_{xLen};\qquad x_{xLen-i} = X_i$$

$$x = x_{xLen-1} 256^{xLen-1} + x_{xLen-2} 256^{xLen-2} + \cdots + x_0$$

❑**Ví dụ**

$$X = 22AA33FF$$

$$x = 22_h \cdot 256^3 + AA_h \cdot 256^2 + 33_h \cdot 256 + FF_h$$

$$= 34 \cdot 256^3 + 170 \cdot 256^2 + 51 \cdot 256 + 255 = 581.579.775$$

9

Khóa RSA và các phép biến đổi cơ sở

❑**Hàm chuyển đổi dữ liệu cơ sở**

- I2OSP: Integer To Octet String Primitive

X = I2OSP(x, sLen), x≥0

$$X = X_1 X_2 \ldots X_{xLen}$$

❑**Ví dụ**

$$X = \text{I2OSP}(581.579.775,\ 6)$$

$$x = 581.579.775$$

$$= 00_h \cdot 256^5 + 00_h \cdot 256^4 + 22_h \cdot 256^3 + AA_h \cdot 256^2 + 33_h \cdot 256 + FF_h$$

$$X = 000022AA33FF$$

10

Khóa RSA và các phép biến đổi cơ sở

❑**Phép mã hóa, giải mã cơ bản**

- Mã hóa: c = RSAEP(K_P, m)

$c = m^e \bmod n$

- Giải mã: m = RSADP(K_S, c)

$m = c^d \bmod n$

Giải mã dùng khóa bộ 5:

$$m_1 = c^{dP} \bmod p;\qquad m_2 = c^{dQ} \bmod q$$

$$h = (qInv \cdot (m_1 - m_2)) \bmod p$$

$$m = m_2 + h \bullet q$$

11

Khóa RSA và các phép biến đổi cơ sở

❑**Phép ký và kiểm tra chữ ký cơ bản**

- Ký: s = RSASP(K_S, m)

$s = m^d \bmod n$

Ký dùng khóa bộ 5:

$$s_1 = m^{dP} \bmod p;\qquad s_2 = m^{dQ} \bmod q$$

$$h = (qInv \cdot (s_1 - s_2)) \bmod p$$

$$s = s_2 + h \bullet q$$

- Kiểm tra chữ ký: m = RSAVP(K_P, s)

$m == s^e \bmod n$

12

1. Giới thiệu chung
2. Khóa RSA và các phép biến đổi cơ sở
3. **Lược đồ mã hóa**
4. Lược đồ ký số
5. Tiêu chuẩn tham số

Lược đồ mã hóa và giải mã

Lược đồ mã hóa RSAES-OAEP

- Phép mã hóa cơ bản RSAEP
- Phép giải mã cơ bản RSADP
- Lược đồ định dạng dữ liệu EME-OAEP

14

Lược đồ mã hóa và giải mã

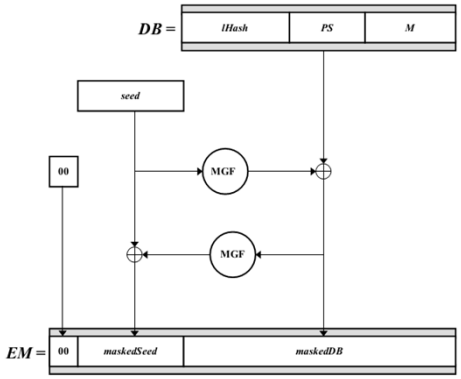
❑Lược đồ định dạng dữ liệu EME-OAEP

EM = EME-OAEP-ENCODE(M, L)

- EME = Encoding Method for Encryption
- OAEP = Optimal Asymmetric Encryption Padding
- M = Message, kích thước "bất kì"
- L = Label, có thể là xâu rỗng
- EM = Encoded Message, kích thước bằng k (octet), có tính ngẫu nhiên dù M cố định.
- Lược đồ sử dụng hàm băm Hash() và hàm sinh mặt nạ MGF()

15

Lược đồ mã hóa và giải mã

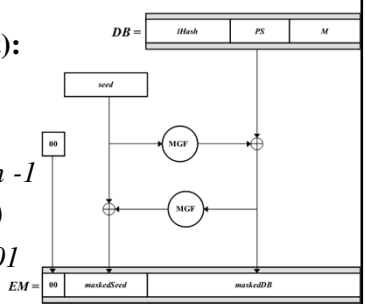


16

Lược đồ mã hóa và giải mã

Kích thước các thành phần (octet):

- *emLen = k*
- *seedLen=hLen*
- *dbLen = k - hLen -1*
- *lHash = Hash(L)*
- *PS = 0x00.00....01*
- *min(psLen)=1*
- *max(mLen)=k-2hlen-2*

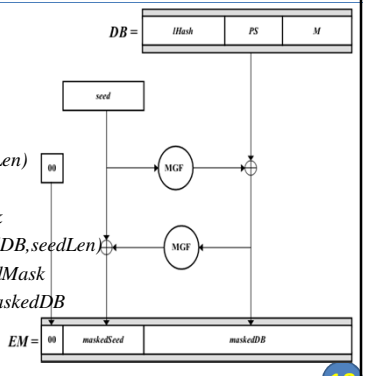


17

Lược đồ mã hóa và giải mã

Các bước biến đổi M

- *Kiểm tra kích thước M*
- *Xác định psLen và PS*
- *Tính lHash*
- *Sinh ngẫu nhiên Seed*
- *dbMask=MGF(Seed, dbLen)*
- *maskedDB=DB⊕dbMask*
- *seedMask=MGF(maskedDB,seedLen)*
- *maskedSeed =Seed⊕seedMask*
- *EM=00||maskedSeed||maskedDB*

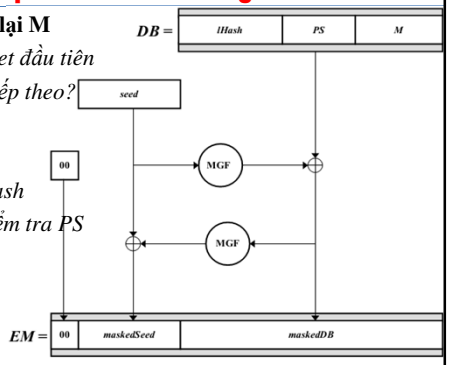


18

Lược đồ mã hóa và giải mã

Các bước tìm lại M

- *Kiểm tra octet đầu tiên*
- *hLen octet tiếp theo?*
- *Seed = ?*
- *DB = ?*
- *Kiểm tra lHash*
- *Xác định, kiểm tra PS*
- *Xác định M*



19

Lược đồ mã hóa và giải mã

Phép mã hóa

RSAES-OAEP-ENCRYPT(Kp,M,L)

1. Kiểm tra kích thước của M
2. EM = EME-OAEP-ENCODE(M, L)
3. m = OS2IP(EM)
4. c = RSAEP(Kp, m)
5. C = I2OSP(c, k)

20

Lược đồ mã hóa và giải mã

Phép giải mã

RSAES-OAEP-DECRYPT(Ks,C, L)

1. c = OS2IP(C)
2. m = RSADP(Ks, c)
3. EM = I2OSP(m,k)
4. M = EME-OAEP-DECODE(EM, L)

21

Lược đồ kí và kiểm tra chữ kí

Lược đồ kí số RSASSA-PSS

- Phép kí số cơ bản RSASP
- Phép kiểm tra chữ kí số cơ bản RSAVP
- Lược đồ định dạng dữ liệu EMSA-PSS

23

Lược đồ kí và kiểm tra chữ kí

❑**Lược đồ định dạng dữ liệu EMSA-PSS**

EM = EMSA-PSS-ENCODE(M, emBits)

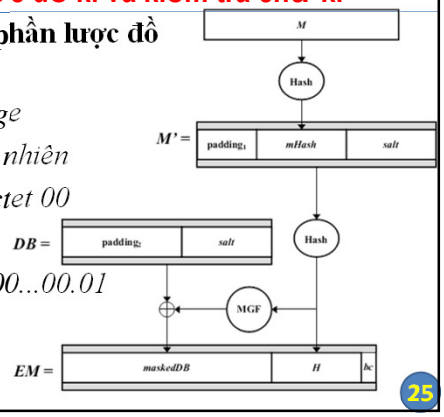
- Sử dụng: sLen, Hash, MGF
- EMSA = Encoding Method for Signature with Apendix
- PSS = Probabilistic Signature Scheme
- M = Message, kích thước bất kì
- emBits: độ dài bit tối đa của OS2IP(EM); tối thiểu là 8hLen + 8sLen + 9
- EM = Encoded Message, kích thước bằng emLen = ⌈emBits/8⌉ (octet)

24

Lược đồ kí và kiểm tra chữ kí

Các thành phần lược đồ EMSA-PSS

- *M: Message*
- *salt: ngẫu nhiên*
- *pad1: 8 octet 00*
- *pad2: 00.00...00.01*
- *Hash()*
- *MGF()*

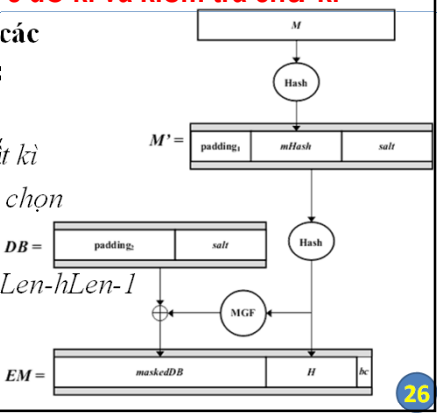


25

Lược đồ kí và kiểm tra chữ kí

Kích thước các thành phần:

- *mLen = bất kì*
- *sLen = tùy chọn*
- *hLen*
- *dbLen=emLen-hLen-1*

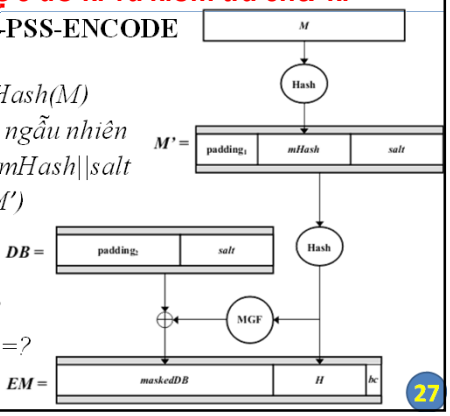


26

Lược đồ kí và kiểm tra chữ kí

EM=EMSA-PSS-ENCODE (M, emBits)

- *mHash = Hash(M)*
- *salt = sinh ngẫu nhiên*
- *M'=pad1||mHash||salt*
- *H=Hash(M')*
- *pad2=?*
- *DB=?*
- *dbMask=?*
- *maskedDB=?*
- *EM=?*

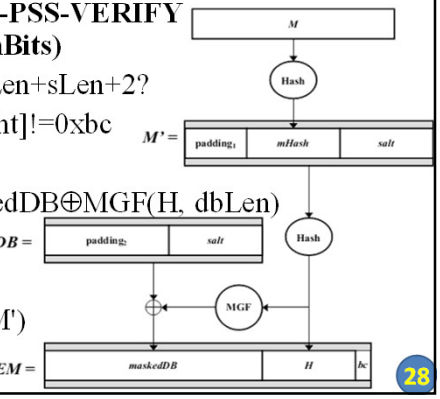


27

Lược đồ kí và kiểm tra chữ kí

EM=EMSA-PSS-VERIFY (M, EM, emBits)

- emLen<hLen+sLen+2?
- emLen[right]!=0xbc
- H=EM[...]
- DB=maskedDB⊕MGF(H, dbLen)
- DB valid?
- M'=
- H'=Hash(M')
- H==H'?



28

Lược đồ kí và kiểm tra chữ kí

Tạo chữ kí

RSASSA-PSS-SIGN(Ks, M)

1. EM = EMSA-PSS-ENCODE(M, modBits-1)
2. m = OS2IP(EM)
3. s = RSASP(Ks, m)
4. S = I2OSP(s, k)

29

Lược đồ kí và kiểm tra chữ kí

Kiểm tra chữ kí

RSASSA-PSS-VERIFY(Kp, M, S)

1. s = OS2IP(S)
2. m = RSAVP(Kp, s)
3. EM = I2OSP(m, emLen);
   emLen=⌈(modBits-1)/8⌉
4. EMSA-PSS-VERIFY(M, EM, modBits-1)

30

1. Giới thiệu chung
2. Khóa RSA và các phép biến đổi cơ sở
3. Lược đồ mã hóa
4. Lược đồ ký số
5. **Tiêu chuẩn tham số**

Yêu cầu đối với khóa RSA

- Theo TCVN 7635:2007
- Cặp khóa RSA dùng để ký thì không được dùng cho mục đích khác (ví dụ, mã hóa)
- Độ dài của mô-đun không được nhỏ hơn 1024 bít và thay đổi theo thời gian

Thời gian sử dụng	Security Strength	nLen
Tới năm 2010	80	1024
Tới năm 2020	112	2048
Sau năm 2020	128	3072

32

Yêu cầu đối với khóa RSA

- p, q ngẫu nhiên và
  $\sqrt{2}\left(2^{nLen/2-1}\right) \le q < p \le \left(2^{nLen/2}-1\right)$
  $p - q > 2^{(nLen/2)-(security\_strength+20)}$
- Từng số trong 4 số: p±1, q±1 phải có nhân tử nguyên tố lớn hơn $2^{secruity\_strength+20}$
- Phải xác định e trước khi xác định d
- e là số lẻ và $65537 \le e < 2^{nLen-2\cdot security\_strength}$
- $d > 2^{nLen/2}$

33