RSA 暗号の復号の高速化

RSA 暗号は，暗号化は極端に高速であるが復号が遅い。これを逆にして，暗号化は遅くなるが復号を極端に高速とする方法を考えました。ポイントは，中国人剰余定理を活用することにより，復号のときの冪乗の指数を 2 つに分けて，指数を極々小さくしても安全を保てるようにしたことです。ただし，暗号化のときの冪乗の指数は大きくなるため，暗号化は遅くなります。また，この方法により，RSA 署名の署名生成と検証の計算コストも逆転します。この方法は，SSL/TLS 通信におけるサーバー側の負荷を軽減し，IC カードの認証でも，IC カード側の負荷を軽減すると思われます。さらに，通常の RSA 暗号と, この RSA 暗号を組み合わせれば, CPU パワーの弱い側の負荷をほとんど無くすことができます。また，この方法は，ElGamal 暗号にも適用できます。

鍵作成

素数 p，q を秘密鍵として $N=pq$ を計算する。

極小さな自然数 $d_p, d_q\ (d_p \neq d_q)$ を秘密鍵として，中国人剰余定理により，

$$d \equiv d_p \pmod{p-1} \text{ かつ } d \equiv d_q \pmod{q-1}$$

を満たす d を求める。（$p-1, q-1$ は共に偶数であるから，d_p, d_q は奇数にとる。補足を参照。）

$p-1, q-1$ の最小公倍数を l として

$e \cdot d \equiv 1 \pmod{l}$　　（d と l は互いに素である必要がある）

となる e を求める。

N，e を公開鍵とする。

暗号処理

メッセージを M として，$b = M^e \bmod N$ を計算し，b を受信者に送信する。

復号処理

$b_p = b \bmod p$ と $b_q = b \bmod q$ を求める。

$m_p = b_p^{d_p} \bmod p$ と $m_q = b_q^{d_q} \bmod q$ を計算する。

中国人剰余定理により，$m \equiv m_p \pmod{p}$ かつ $m \equiv m_q \pmod{q}$ を満たす m を求める。

このとき，$m=M$ が成り立つ。

以下に，十進 BASIC のコードと出力結果を示す。確かに復号される。

```
!RSA 暗号
LET p=8423 !安全素数
LET q=7823
```

```
LET n=p*q !公開鍵
LET d1=3 !秘密の乱数、奇数とする
LET d2=5

!公開鍵 e を作成
LET d=Chinese(d1,d2,p-1,q-1) !公開鍵の逆元を作成（復号には用いない）
PRINT "d=";d
LET L=(p-1)*(q-1)/2
CALL ExGCD(d,L,s,t,c)
LET e=MOD(s,L) !公開鍵
PRINT "e=";e

!暗号化
LET M=31415926
LET b=ruijyou(M,e,n)

!復号
LET b1=MOD(b,p)
LET b2=MOD(b,q)
LET k1=ruijyou(b1,d1,p)
LET k2=ruijyou(b2,d2,q)
LET k=Chinese(k1,k2,p,q)
PRINT "M=";M;k;ruijyou(b,d,n) !一致を確認

END

EXTERNAL FUNCTION ruijyou(a,r,N) !累乗の計算
LET s=1
FOR i=1 TO r
    LET s=MOD(s*a,N)
NEXT I
LET ruijyou=s
END FUNCTION

EXTERNAL FUNCTION Chinese(a1,a2,p,q) !中国人剰余定理
CALL ExGCD(p,q,s,t,c)
LET x=a1*t*q+a2*s*p
LET Chinese=MOD(x/c,p*q/c)
End function
```

```
EXTERNAL SUB ExGCD(a,b,s,t,c) !拡張ユークリッドの互除法（再帰版）
IF b=0 THEN
    LET s=1
    LET t=0
    LET c=a
ELSE
    LET q=INT(a/b)
    LET r=MOD(a,b)
    CALL ExGCD(b,r,s1,t1,c)
    LET s=t1
    LET t=s1-t1*q
END IF
END SUB
```

d= 11967665

e= 12377533

M= 31415926　31415926　31415926

デジタル署名

(1) 鍵ペア

N, eを送信者の公開鍵とする。 p, q, d_p, d_qが送信者の秘密鍵である。

(2) 署名

送信者は，文書 M に対して，文書 M のハッシュ値 h＝h(M)を求める。

（安全のため，文書 M にはパディングをする。）

$h_p = h \bmod p$ と $h_q = h \bmod q$ を求め，さらに，$\sigma_p = h_p^{d_p} \bmod p$ と $\sigma_q = h_q^{d_q} \bmod q$ を計算する。

中国人剰余定理により，$\sigma \equiv \sigma_p \pmod{p}$ かつ $\sigma \equiv \sigma_q \pmod{q}$ を満たすσを求める。

送信者は，受信者に文書 M とその署名σを送る。

(3) 検証

受信者は，M からハッシュ値 h=h(M)を求める。

次に，σ, eから$h' = \sigma^e \bmod N$を求める。

h'がhと一致すれば，署名は正当なものである。

安全性の根拠

d_p, d_q を極小さく取ったとしよう。このとき，公開鍵eの$(p-1)(q-1)$を法とする逆元dは，N と

同じ程度の bit 数となる。また，もしも d_p, d_q の値が攻撃者に知られても，攻撃者は $N=pq$ の因数分解を知らないため，中国人剰余定理が使えない。よって，安全であると考える。

課題は，指数 d_p, d_q を安全性を保ちながらどれだけ小さく取れるかである。共に 64bit 程度の数に取ると合わせて 128bit あるから，共通鍵暗号ほどの bit 数となり，十分に安全であると思われる。さらに，例えば，$d_p = 3, d_q = 5$ のように，固定した小さな奇数にすることも可能と思われる。

処理速度

通常の RSA 暗号は暗号化と署名検証が極端に速く，復号と署名生成が遅い。一方，この暗号は，中国人剰余定理を用いて，指数 d_p, d_q を $d_p = 3, d_q = 5$ のように固定した小さな奇数にとると，復号処理において極めて高速である。また，デジタル署名における署名生成でも，極めて高速である。

暗号化と署名検証は，累乗の計算の指数が N 程度の大きさとなるため遅くなる。

補足

定理　中国人剰余定理の拡張

自然数 m, n に対して，m, n の最大公約数を g，最小公倍数を l とおくとき，連立合同式

$$\begin{cases} x \equiv a \pmod{m} \\ x \equiv b \pmod{n} \end{cases} \quad \cdots①$$

が整数解を持つための必要十分条件は，$a \equiv b \pmod{g}$ である。また，この条件が成り立つとき，整数解 x は l を法として一意的に定まる。

証明

①が整数解を持つとすると，

$$x = a + sm = b + tn$$

となる整数 s, t が存在する。これより

$$sm - tn = b - a \quad \cdots②$$

よって，$b-a$ は g の倍数であるから，$a \equiv b \pmod{g}$

逆に，$a \equiv b \pmod{g}$ のとき②は整数解 s, t をもつから，$x = a + sm = b + tn$ とおけば①の解になる。

また，x, x' が共に①の整数解であるとすると，

$x \equiv x' \pmod{m}$ かつ $x \equiv x' \pmod{n}$

であるから　　$m \mid x - x'$ かつ $n \mid x - x'$

したがって，$x - x'$ は m, n の公倍数である。よって $l \mid x - x'$ であるから $x \equiv x' \pmod{l}$

すなわち，x は l を法として一意的に定まる。

証明終り

次に，実際に①の解を求める。まず，拡張ユークリッドの互除法により

$$sm + tn = g \cdots③$$

となる整数 s, t を求める。このとき

$$(b-a)sm/g + (b-a)tn/g = b - a$$

よって,

$$x = a + \{(b-a)/g\}sm = b + \{(a-b)/g\}tn$$

は①の解である。③を用いて変形すると

$$x = a(1 - sm/g) + bsm/g = atn/g + bsm/g$$

これを l で割った余りを求めればよい。

$d \bmod p-1 = d_p$, $d \bmod q-1 = d_q$ となる最小の自然数 d を求めるには, $p-1, q-1$ の最大公約数 g ,

最小公倍数を l として，この定理を用いる。 s, t を

$$s(p-1) + t(q-1) = g$$

を満たす整数として

$$d_p + (d_q - d_p)s(p-1)/g \quad \text{もしくは} \quad d_p t(q-1)/g + d_q s(p-1)/g$$

を l で割った余りが d である。

また， $l = (p-1)(q-1)/g$ は偶数であるから， d と d^{-1} が存在するためには， $d_q - d_p$ が偶数かつ d が

奇数である必要がある。よって， d_p, d_q が共に奇数である必要がある。

補足 2

公開鍵 e は次のように求めることもできる。

まず， d_p, d_q $(d_p \neq d_q)$ に対して

$$e_p \cdot d_p \equiv 1 \pmod{p-1}, \quad e_q \cdot d_q \equiv 1 \pmod{q-1}$$

となる e_p と e_q を求める。

ここで，$p-1$ と $q-1$ は共に偶数であるから，d_p と d_q は共に奇数でなければならない。このとき，e_p と

e_q は共に奇数となるから， $e_q - e_p$ は偶数である。

補足 1 により， $e_q - e_p$ が $p-1, q-1$ の最大公約数 g の倍数であるとき,

$$e \equiv e_p \pmod{p-1} \text{ かつ } e \equiv e_q \pmod{q-1}$$

を満たす e が求まる。 e は $te_p(q-1)/g + se_q(p-1)/g$ を $(p-1)(q-1)/g$ で割った余りである。