マルチメディア通信と分散処理　106－34
コンピュータセキュリティ　16－34
（２００２．２．１５）

モバイル端末に適した検証可暗号法の提案

小栗　伸幸
（株）NTT ドコモ
〒239-8536　神奈川県横須賀市光の丘 3-5
e-mail: oguri@mml.yrp.nttdocomo.co.jp

あらまし　検証可暗号は,公平な文書交換など,多くの暗号プロトコルで利用できる方法である．この検証可暗号に対して，これまで，素因数分解問題や離散対数問題に基づく検証可暗号が提案されてきた．本稿では，ラティス問題に基づき，計算能力の小さいモバイル端末に適した公開鍵暗号 NTRU による検証可暗号を提案する．更には，NTRU による閾値暗号を提案することによって，本稿で提案する検証可暗号の鍵寄託への応用を提案する．
キーワード　検証可暗号，閾値暗号，NTRU，モバイル端末

Verifiable Encryption Scheme suitable for mobile devices

Nobuyuki Oguri
NTT DoCoMo, Inc.
3-5, Hikarinooka, Yokosuka, Kanagawa, 239-8536 Japan
e-mail: oguri@mml.yrp.nttdocomo.co.jp

Abstract　Verifiable encryption scheme is used in many cryptographic protocols such as fair exchange. So far, verifiable encryption scheme based on factorization problem or discrete log problem has been proposed. In this paper, we propose a new verifiable encryption scheme based on lattice problem by using NTRU that is suitable for mobile devices with only small computation power. And then, we propose key escrow scheme using NTRU which verifiable encryption scheme and threshold cryptosystem are applied to.
Keywords　verifiable encryption scheme, threshold cryptosystem, NTRU, mobile device

1　はじめに

検証可暗号とは，文書を暗号化した人が他者にその文書に関する情報を与えないで，文書に関する知識を証明し，暗号化した文書を保証する方法である．

検証可暗号は，[1] の公開検証可秘密分散法において初めて提案された．その後，[2,3] 等で，Elgamal, Okamoto-Uchiyama, RSA などによる検証可暗号が提案され，[2]の公平な署名の交換など，公開検証可秘密分散法以外にも様々な暗号プロトコルでの利用法が提案されている．

一方，[4]において，計算能力の小さいモバイル端末に適した，計算量の少ない公開鍵暗号として，NTRU が提案された．

本稿では、この NTRU を利用して，検証可暗号を実現する方法に関して考察する．

2　NTRU

2.1 準備

多項式環 $R=\mathbf{Z}[x]/(x^N-1)$ において，多項式 $a(x)=a_0+a_1x+\cdots+a_{N-1}x^{N-1}\in R$ をベクトル $a=(a_0,a_1,\cdots,a_{N-1})\in\mathbf{Z}^N$ とみなし，そのセンターノルムを

$$\|a\|_\perp=\sqrt{\sum_{i=0}^{N-1}\left(a_i-\frac{1}{N}\sum_{i=0}^{N-1}a_i\right)^2}=\sqrt{\sum_{i=0}^{N-1}a_i^2-\frac{1}{N}\left(\sum_{i=0}^{N-1}a_i\right)^2}$$

で定義する．このとき，$\|a\|_\perp=O(\sqrt{N})$ を満たす多項式 $a(x)$ を短い多項式と呼ぶ．

また，$a(x)=\sum_{i=0}^{N-1}a_ix^i, b(x)=\sum_{i=0}^{N-1}b_ix^i\in R$ の積 $a(x)\cdot b(x)(\mathrm{mod}\,x^N-1)$ を $a*b$ で表す．このとき，$c(x)=\sum_{k=0}^{N-1}c_kx^k=a*b\in R$ を，各 k に対して $c_k=\sum_{i+j=k(\mathrm{mod}\,N)}a_ib_j$ としたベクトル $c=(c_0,c_1,\cdots,c_{N-1})\in\mathbf{Z}^N$ とみなすことができ，2つのベクトル $a=(a_0,a_1,\cdots,a_{N-1})$ と $b=(b_0,b_1,\cdots,b_{N-1})$ の積として定義できる．このとき，$\|a*b\|_\perp\approx\|a\|_\perp\cdot\|b\|_\perp$ となる．

2.2 アルゴリズム

適当な自然数 N に対して，q を大きなモジュラス値，p を小さなモジュラス値，x^N-1, p, q は互いに素であるように定め，N, q, p をシステム共通の値とする．また，R_p，R_q をそれぞれ $\mathbf{Z}_p$，$\mathbf{Z}_q$ の元を係数とする $N-1$ 次多項式の集合とする．

【鍵生成】短い多項式 f,g を任意に定めて，秘密鍵とする．この f に関して，$f*F_q=1(\mathrm{mod}\,q)$，$f*F_p=1(\mathrm{mod}\,p)$ となる F_p,F_q を計算した後，公開鍵として，$h=F_q*g(\mathrm{mod}\,q)$ を計算する．

【暗号化】　メッセージ空間を

$$L_m=\{m\in R\mid -1/2(p-1)\le m_i\le 1/2(p-1)\}$$

で定義し，$m\in L_m$ に対して，ランダムな短い多項式 r を定める．このとき，暗号文は，$e=p*r*h+m(\mathrm{mod}\,q)$ となる．

【復号化】秘密鍵 f と暗号文 e から，$a=f*e(\mathrm{mod}\,q)$[1]を計算した後，$b=a(\mathrm{mod}\ p)$ を得て，この b と秘密鍵 f の R_p での逆元 F_p から，復号文 $m=F_p*b(\mathrm{mod}\ p)$[2]を得ることが可能となる．

定義　NTRU 仮定（ラティス問題）

多項式 $h\in R_q$ が与えられたとき，$f*h=g(\mathrm{mod}\,q)$ を満たす短い多項式 f,g を求めることが困難である．

$\deg h=N-1$ のとき，f,g を求める最も効率的な方法は，$2N$ 次元ラティス上の短いベクトルを探す問題を解くことと考えられる．

3　検証可暗号

3.1 集合の定義

最初に2つの集合 $S_1=\{f\in R\mid\|f\|_\perp=O(1)\}$，$S_2=\{f\in R\mid\|f\|_\perp=O(\sqrt{N})\}$ を定義し，各パラメータに対する集合を次のように定義する．

$L_a,L_f\subseteq S_2$，$L_c\subseteq S_1$，$L_x,L_r\subseteq S_2$ を満たす集合に対して，$\boldsymbol{a}\in L_a$，$\boldsymbol{f}\in L_f$，$c\in L_c$，$\boldsymbol{x}\in L_x$，$\boldsymbol{j}\in L_j$ ならば，$x=\boldsymbol{x}+c*\boldsymbol{a}\in L_x$，$r=\boldsymbol{j}+c*\boldsymbol{f}\in L_r$ を満たすような集合 $L_x,L_j\subseteq S_2$ を定める．

例えば，

$$L_a=L_m,\quad L_f=\{f\in R_p\mid\|f\|_\perp=O(\sqrt{N})\}$$

$$L_c=\{f\in R_q\mid\|f\|_\perp=O(1)\}$$

$$L_x=\{f\in R_q\mid\|f\|_\perp=O(\sqrt{N})\},\quad L_r=L_x$$

としたとき，

$$L_x=L_f,\quad L_j=L_f$$

---

[1] $a=f*e=p*r*g+f*m(\mathrm{mod}\,q)=p*r*g+f*m$ である必要がある．

[2] $f(\mathrm{mod}\ p)=1$ を満たす秘密鍵 f を選んだとき，$a(\mathrm{mod}\ p)=m$ となり，F_p による乗算を行う必要がない．

として定義できる．

以下では，サンプル空間を

$$L_a = R_2,\quad L_f = L_a$$

$$L_c = \{c = c_i x^i + c_j x^j + c_k x^k + c_l x^l \in R_5 \mid i, j, k, l \in \{0,1,\cdots,N-1\}, c_i, c_j, c_k, c_l \in \{0,1\}\}$$

$$L_x = R_5 \setminus L_{cf}^*,\quad L_j = L_x$$

$$L_x = R_5,\quad L_r = L_x$$

として定義する．ここで，

$$L_{cf}^* = \{g = c * f \in R_5 \mid c \in L_c, f \in R_2\}$$

である．

3.2 検証可暗号プロトコル

検証可暗号とは，証明者が暗号文に対応する平文を知っていることを，平文に関する情報を与えないで，任意の検証者に証明できる暗号方式である．

証明者は，$a \in L_a$，$f \in L_f$ として，暗号文 $e = p * f * h + a \pmod q$ に対する平文 a を知っていることを以下のプロトコルで示す．

ステップ 1

証明者は，$x \in L_x$，$j \in L_j$ をランダムに選び，$a = p * j * h + x \pmod q$ を検証者に送る．

ステップ 2

検証者は，ランダムな多項式 $c \in L_c$ を証明者に送る．

ステップ 3

証明者は，受け取った c に対して，$x = x + c * a$，$r = j + c * f$ とし，検証者に送る．

ステップ 4

検証者は，$x \in L_x$，$r \in L_r$ が成立するとき，$a = p * r * h + x - c * e \pmod q$ を検証する．

定理

ランダムオラクルモデル上 NTRU 仮定で，証明者は検証者に平文 a に関する情報を与えないで，暗号文 e に対応する平文に関する知識を示すことができる．

証明

付録 A 参照

4 応用

鍵寄託などの目的で，秘密情報を暗号化して登録することを考える．ここで，鍵生成センタ，登録者，登録センタ，復号センタによるモデルを考える．

秘密情報登録時，登録者は，検証可暗号によって，秘密情報を登録センタに登録する．この際，登録センタは，鍵生成センタによって生成された公開鍵によって検証することで，鍵生成センタによる秘密鍵で復号可能な暗号文を登録したことを検証可能である．

一方，秘密情報復元時，付録 B で示す閾値暗号によって復号することで，結託に対する耐性を持たせることが可能となる．鍵生成センタから複数の復号センタに対して，分散した秘密鍵が配布され，閾値以上の復号センタによってのみ秘密情報を復号可能とする．こうすることで，いくつかの復号センタの結託による情報漏洩に対する不安を軽減できる登録及び復元方法となる．

5 まとめ

本稿において，モバイル端末に適した公開鍵暗号 NTRU により検証可暗号を実現する方法について検討した結果をまとめた．今回提案した方式は，NTRU のアルゴリズムに基づく方式であるため，積と和による演算によって可能な方式であり，計算量の少ない方式となる．従って，計算能力の小さいモバイル端末に適した方式となる．

今後，更なる各パラメータに対する評価を行い，安全性に関する検討をより詳細に行う必要がある．

参考文献

[1] M. Stadler, Publicly Verifiable Secret Sharing, In *Proceedings of EUROCRYPT'96*, LNCS 1070, Springer, pp.190-199(1996).

[2] G. Ateniese, Efficient Verifiable Encryption (and Fair Exchange) of Digital Signature, In *Proceeding of the 5th Annual Conference on Computer and Communications of Security*, ACM, pp.138-146(1999).

[3] F. Bao, An Efficient verifiable encryption scheme for encryption of discrete logarithm, In *Proceeding of CARDIS'98*, LNCS 1820, Springer, pp.213-220(2000).

[4] J. Hoffstein, J. Pipher, and J. H. Silverman, NTRU: A ring based public key cryptosystem, In *Proceedings of ANTS'3*, LNCS 1423, Springer, pp.267-288(1998).

[5] A. Shamir, How to share a secret, In *Communication of the ACM*, Vol22, No.11, pp.612-613(1979)

A.定理の証明

補題1～3から従う.

補題1

証明者が正しければ，上記検証可暗号方式の検証は受理される.

証明

$x = x + c*a$ ， $r = j + c*f$ のとき，$x, j \in L_x \setminus L_{cf}^*$ ， $c \in L_c$ ， $a \in L_a$ ， $f \in L_f$ より，$x \in L_x$ ， $r \in L_r$.

また， $p*f*h + a = e + q*E$ ， $p*j*h + x = a + q*A \quad (E, A \in R)$ より，

$$\begin{aligned} & p*r*h + x - c*e \\ &= p*(j + c*f)*h + (x + c*a) - c*e \\ &= (p*j*h + x) + c*(p*f*h + a) - c*e \\ &= a + q*A + c*(e + q*E) - c*e \\ &= a + q*(A + c*E). \end{aligned}$$

よって， $a = p*r*h + x - c*e \pmod q$. □

補題2

証明者が誤っていれば，上記検証可暗号方式の検証は，エラー率 $e \geq 3/\#L_c$ で，拒否される.

証明

エラー率 $e \geq 3/\#L_c$ であるので，$\#L_c$ 通りの質問に対して，$\#L_c e \geq 3$ 個のエラーが存在する．つまり， L_c の中から，検証式を満たす少なくとも3つの異なる c, c', c'' を選ぶことができる.

forking lemma より，公開鍵 h を入力して，

$$\begin{aligned} a &= p*r*h + x - c*e \\ &= p*r'*h + x' - c'*e \\ &= p*r''*h + x'' - c''*e \pmod q \end{aligned}$$

を満たす $e, (x, r, c), (x', r', c'), (x'', r'', c'')$ を出力する確率的多項式時間アルゴリズムが存在すると仮定する．このとき，

$$\begin{aligned} \Delta x_1 &= x - x', \Delta x_2 = x - x'' \\ \Delta r_1 &= r - r', \Delta r_2 = r - r'' \\ \Delta c_1 &= c - c', \Delta c_2 = c - c'' \end{aligned}$$

とおくと，

$$\begin{aligned} p*\Delta r_1*h + \Delta x_1 &= \Delta c_1*e + q*A_1, \; p*\Delta r_2*h + \Delta x_2 \\ &= \Delta c_2*e + q*A_2 \end{aligned}$$

となるので，

$$\begin{aligned} & \Delta c_1*\Delta c_2*e \\ &= p*\Delta c_2*\Delta r_1*h + \Delta c_2*\Delta x_1 - q*\Delta c_2*A_1 \\ &= p*\Delta c_1*\Delta r_2*h + \Delta c_1*\Delta x_2 - q*\Delta c_1*A_2. \end{aligned}$$

ここで， $\Delta X = \Delta c_1*\Delta x_2 - \Delta c_2*\Delta x_1$ ， $\Delta R = \Delta c_2*\Delta r_1 - \Delta c_1*\Delta r_2$ とおくと， $p*\Delta R*h = \Delta X + q*(\Delta c_2*A_1 - \Delta c_1*A_2)$ となり，

$$p\Delta R * h = \Delta X \pmod q.$$

$\|\Delta X\|_\perp = O(\sqrt{N}), \|\Delta R\|_\perp = O(\sqrt{N})$ であるので，多項式時間で NTRU ラティスを解くことができることを意味する．これは，NTRU ラティスを解くことが難しければ，証明者は嘘をつけないことを意味する． □

補題 3

上記検証可暗号方式は，平文に関する情報を与えない．

証明

次のようなシミュレータを作成する．与えられた e に対して，ランダムに $x' \in L_x$ ，$r' \in L_r$ ，$c' \in L_c$ を選択する．そして，$(h, e, p * r' * h + x' - c' * e \pmod q)$ をランダムオラクルに質問する．既に，同じ質問がされている確率は，極めて小さいので，ランダムオラクルは，この質問に対する回答を c' としてリストに入れる．このとき，シミュレータの出力による Proof は (x', r', c') となる．

このとき，$\#L_a = \#L_f = \#R_2 = 2^N$ ，$\#L_c = {}_N C_4 2^4$ ，$\#L_x = \#L_r = \#R_5 = 5^N$ より，$(\#L_c)(\#L_a)(\#L_f)/(\#L_x)(\#L_r) = ({}_N C_4 2^{2N+4})/5^{2N}$ となり，極めて小さいので，上記検証可暗号方式で出現する (x, r, c) の系列が，シミュレータによって生成される (x', r', c') の系列と統計的識別不可能になる．

上記シミュレータは，誰にでも作成できる．このことは，検証者が，平文に関する情報を得ないことを意味する． □

B. 閾値暗号

(k, l) 閾値暗号とは，l 人で秘密鍵を秘密分散法によって分散し，分散鍵の所有者 l 人のうち k 人以上が協力することで，復号できる方法のことをいう．ここでは，Shamir による秘密分散法[5]を用いた NTRU での閾値暗号について説明する．以下では，$f \pmod p = 1$ を満たす秘密鍵について考える．

$f_0 = f$ とおき，ランダムに短い多項式 $f_i \in R$ を定め，$F(X) = \sum_{i=0}^{k-1} f_i * X^i$ とする．各復号センタ I に対して，$F(I)$ を配布する $(I = 1, 2, \cdots, l)$．また，秘密鍵と異なる秘密情報として，$\|d\|_\perp = O(\sqrt{N}), d \pmod p = 0$ を満たす d を選び，秘密鍵の分散と同様に，$\Delta(X) = \sum_{i=0}^{k-1} \delta_i * X^i$ として，復号センタ I に $\Delta(I)$ を配布する $(I = 1, 2, \cdots, l)$．暗号文 e に対して，復号センタ I が，$a_I = F(I) * e + \Delta(I)$ を計算し，各復号センタが計算結果を供出する．k 個以上の $F(I)$ 及び $\Delta(I)$ が揃うことで，$k-1$ 次多項式 $F(X)$ 及び $\Delta(X)$ を復元でき，$f = F(0)$ 及び $\delta = \Delta(0)$ を得ることができる．従って，k 個以上の a_I が揃うことで，ラグランジェの補間法より，

$$\begin{aligned} a &= \sum_{I=0}^{k-1} a_I \prod_{0 \le J \le k-1, J \ne I} \frac{I}{I-J} = f * e + \delta \\ &= p * r * g + f * m + \delta \pmod q \\ &= p * r * g + f * m + \delta \end{aligned}$$

を計算可能である．このとき，$p * r * g + f * m \pmod p = m, \delta \pmod p = 0$ より，$m = a \pmod p$ によって復号文を得ることができる．

上記方法によって，各 a_I から，各復号センタ I が所有する分散秘密鍵に関する情報 (f_I, δ_I) は漏れない．また，秘密通信路によって各 a_I を得ることで，各 a_I を得て a を計算できる人のみが，復号結果を得ることが可能となる．