楕円曲線を利用したネットワークセキュリティシステム構築の検討

Invesigaton of Network Security Systems applyining Elliptic Curves

株式会社　メビウス　扇　裕和
Mebius Corporation　Hirokazu Ougi
e-mail: h_ougi@mebius.co.jp (URL http://www.mebius.co.jp)

概要

現代の情報社会の必須の基盤技術である公開鍵暗号システムは、RSA システムばかりでなく、楕円曲線でもほぼ同等のものを実現することが可能である[1][2][3]。アーベル群を構成する楕円曲線の離散対数問題を利用した場合には、公開鍵を、**Q** = d1・P1 + d2・P2 とさらに和の形に表現することも可能である。本論文では、このように秘密鍵を複数の秘密鍵で構成することに着目し、より安全なネットワークシステムの構築法について検討した。

1.　はじめに

現代の情報社会の必須の基盤技術である公開鍵暗号システムは、RSA システムばかりでなく、楕円曲線でもほぼ同等のものを実現することが可能である。RSA システム、および楕円曲線を利用したシステムは、それぞれ特徴あるシステムを構成しており、組み込まれるシステムの要求に応じて使いわけられている。一般に公開鍵暗号システムは、公開鍵 Q と秘密鍵 d とを組にして構成されるシステムであり、これをアーベル群を構成する楕円曲線の離散対数問題を利用して構成した場合、公開鍵Qは、楕円曲線上の点 P (一般にベースポイントとよばれる。) により Q = d・P (ここで d; 秘密鍵、ドット・は楕円曲線上の演算) という関係で表現される。

ネットワークセキュリティシステムは、特定の情報の送信者がある定まった通信相手に対して、情報データを暗号化して送信し、受信者がこれを復号し、必要に応じて相手を認証する 1 対 1 の暗号化通信が基本である。

また、ある特定の N 人で情報データを共有する必要が生じた場合などは、この 1 対 1 の暗号化通信をもとに、これを繰り返し使用する 1 対 N のシステムが構築される。

楕円曲線暗号に着目した場合、楕円曲線上の点はアーベル群を構成するため、公開鍵を Q = d1・Go + d2・Ge （ここで d1 , d2 ; 整数、Go , Ge ; 楕円曲線上の点）とさらに和の形に表現することができる。本論文では、この公開鍵を和の形に表現する方式を用いて、効率的な 1 対 N の暗号化通信システムの構築法について検討を行ったが、この公開鍵を和の形に表現する方法は、1 個の公開鍵 Q と 2 個の秘密鍵の組{ d1 , d2 }とを対応させたものと考えることができる。常識的に考えれば、2 個またはそれ以上複数個の秘密鍵でシステムを運用すると、ユーザの鍵管理が複雑になり、これに伴って不便さが増すと考えられる。しかし、2 個の秘密鍵を通信ネットワーク上、別の場所に格納し、これを連動させて運用することを考えた場合、物理的に別の場所に保管された秘密鍵の両方を盗むことは難しく鍵管理の安全性はきわめて高くなるものと考えられる。近年、ユビキタスが唱えられ、モバイルコミュニケーションが広く普及するようになったが、たとえば、移動体端末側に 1 個の秘密鍵 d1 を持たせ、もう 1 個の秘密鍵 d2 を自宅もしくは移動体端末を管理する信頼のおける別の固定端末に持たせて利用すると有効ではないかと考えられる。

また、SET などのシステムにみられるように、ネットワーク上のセキュリティを考えた場合、鍵管理は信頼のおける第 3 者が携わるものである。そこで、公開鍵 Q、及び 1 個の秘密鍵 d1 を信頼のおける第 3 者が管理し、ユーザがもう 1 個の秘密鍵 d2 を利用するといったシステムの構築も有効と考えることができる。このように本論文は、楕円曲線を利用したシステムの応用として、

(1)　公開鍵に対応する秘密鍵を分散して管理する方法
(2)　1 対 N の方式でデータを共有する方法

について述べたものである。

本論文を通して、楕円曲線上の公開鍵は、Q = d1・Go + d2・Ge と和の形式で表現することで議論が進められている。アーベル群の構造定理によれば、Go , Ge は異なる巡回群の生成元として定められる可能性がある。暗号化通信に使用できる安全な楕円曲線を生成するには、虚数乗法論を利用するなどいくつかの方法が知られている[4]。本論文では、まず Go をこの安全な楕円曲線の生成元とし、この Go をもとに別の生成元 Ge を定める方法、および得られる楕円曲線の安全性について述べる。

次に、2 個の秘密鍵{ d1 , d2 }を利用した暗号化、復号化、及び電子署名の構成法について述べる。そしてこの暗号方式を応用した、鍵管理の方法、データアクセスの管理方法について述べる。また、導入した異なる生成元をもとに、双線型形式Φ(Q1 , Q2)を利用した暗号化、復号化について検討し、1 対 N のデータ共有化方式について述べ、さらに、ユーザの結託攻撃からの安全性について述べる。

一般に、楕円曲線を利用したシステムは、極めて難解な数論的アルゴリズムを用いて構成されるものであるが、本論文は、既存の安全な楕円曲線上にシステムを構築したものであり、ここで述べた応用はかなり初歩的な議論で導出されたものである。なお、この応用は有名な RSA システムにはない、新しい利用方法ではないかと考える。

2. 複数個の生成元を持つ楕円曲線の生成

本論文では、楕円曲線上の公開鍵を Q = d1・Go + d2・Ge と和の形式で表現することで議論を進めている。

まず、暗号論的に安全な有限体上の楕円曲線の生成元として Go を定め、これをもとに有限体を代数拡大し、Go とは異なる巡回群の生成元として Ge を設定する方法について述べる。楕円曲線の記号については、一般的なものを使用することとし、たとえば、CRYPTREC Report[1] 、筑波大学集中講義資料[3]などを参考にした。

2.1. 1 個の生成元（ベースポイント）を持つ楕円曲線上の暗号システム

まず､暗号論的に安全な有限体上の楕円曲線を定める。暗号論的に安全な楕円曲線を生成する方式としては、虚数乗法論を利用した方法等が知られている。ここで定める楕円曲線は、MOV 攻撃や、anomalous 攻撃等に耐えるものである[4][5]。

(1) 暗号に使用する有限体上の楕円曲線

(i) 暗号を構成する楕円曲線

$Y^2 = X^3 + a*X + b$ ； $a, b \in Fq$

楕円曲線の定義体 Fq

(Fq ；q = p^n ; 素数 p の n べきによる有限体)

(ii) 楕円曲線の位数 $\#E(Fq) = u*s$

u ; 小さい整数 、s ; 大きい素数[4]

(iii) 素数位数 s のベースポイント Go

(楕円曲線上の点、生成元)

(iv) 秘密鍵 $d \in Zs$ （ 素数位数 s の有限体 ）

(iv) 公開鍵 Q = d・Go

(ﾄﾞｯﾄ・；楕円曲線上の加法演算)

(2) 楕円 ElGamal 暗号の暗号化、復号化方式

図１に暗号化、復号化の方式を示す。

(i) 多項式時間で計算可能な関数

π：Fq ⟶ Z/Zq

ユーザの平文 m ∈Z/Zq (情報データ)

(ii) 情報データ（平文 m）の暗号化

乱数 $\kappa \in Zs\ (0 \leqq \kappa \leqq s-1)$ を生成する。

R = κ・Go = (Rx , Ry) を計算する。

T = κ・Q = (Tx, Ty) を計算し

c = π(Tx) ⊕ m を求め、

暗号データとして { R , c } を作成する。

(iii) 情報データの復号化

秘密鍵 d より

$d \cdot R = d \cdot (\kappa \cdot Go) = \kappa \cdot (d \cdot Go)$

$= \kappa \cdot Q = (Tx, Ty) = T$

この κ・Q = T をもとに、暗号化された情報データ c を復号化し、

平文 m = π(Tx) ⊕ c を取得する。

(3) ECDSA による署名作成[1]

(i) 情報データ(平文 m) の電子署名の作成

平文 m のハッシュ値 e = H(m) を求める。

(H ; ﾊｯｼｭ関数 SHA-1)

乱数 $\kappa\ (0 \leqq \kappa \leqq s-1)$ を生成し、

R = κ・Q = (Tx , Ty) を求め、

$S = (e + d*r) / \kappa \pmod s$ (ここで $r = Tx \pmod s$)

これより、電子署名データ { S , R } を作成する。

(4) 通信相手ユーザによる電子署名の検証

(i) R = κ・Q = (Tx, Ty) により

$r = Tx \pmod s$ を求める。

(ii) $C = S^{-1} \pmod s$ と情報データ(平文 m) のハッシュ値 e = H(m) を求める。

(iii) $C \cdot (e \cdot Go + r \cdot Q) = C \cdot (e \cdot Go + r \cdot (d \cdot Go))$

$= C \cdot (e + r*d) \cdot Go = C \cdot (S*\kappa) \cdot Go$

$= \kappa \cdot Go = R$

この計算結果が、R = κ・Go と一致した場合、平文 m の正当性を確認したものとする。

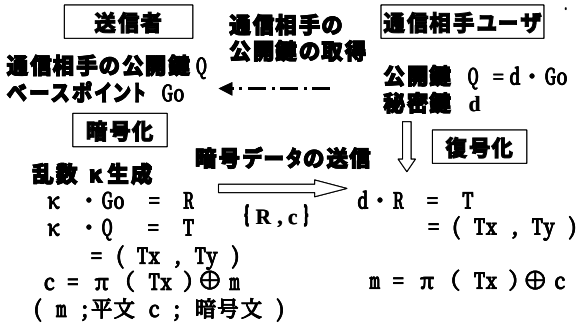

図１ 暗号化、復号化の方式

2.2. 2 個の生成元を持つ楕円曲線

(1) 拡大体 Fq(√τ)上の楕円曲線

2.1 項では、有限体 Fq 上に暗号論的に安全な楕円曲線を定めた。ここで、Fq 上の関数 f(X)を

$Y^2 = f(X) = X^3 + a*X + b$

と定め、さらに、次の性質を持つ $\tau \in Fq$ を定める。

(a) $xe \in Fq$ より $\tau = f(xe)$

(b) $\tau = t^2$ となる $t \in Fq$ が存在しない。

即ち、τはある Fq の要素の平方とはならない要素とする。この要素τにより、有限体 Fq に√τを付加して得られる代数拡大体 Fq(√τ)を考える。

τの定め方から 2 次の多項式（$X^2 - \tau$）は、多項式環 Fq[X] の既約多項式となり、従って Fq (√τ) は、剰余環 Fq[X] ／($X^2 - \tau$）と同型となる。

$\rho = X \bmod (X^2 - \tau)$ とすると、

Fq(√τ) の要素は、$\sigma = a*\rho + b$ $(a, b \in Fq)$ と表示される。

このように､Fq(√τ) は､Fq の 2 次の拡大体であり、位数は、#Fq(√τ) = q^2 となっている。

また、$\rho = \sqrt{\tau}$ であり、$\rho^2 = \tau = f(xe)$ であるから、Ge = (xe , ρ) は､拡大された体 Fq(ρ)を定義体とする楕円曲線上の点と考えることができる。

次に、Ge の整数倍について考える。

任意の整数 $m \in Z\ (m > 0)$ に対して、m 倍は

m・Ge = (xem , ρ* sm) (ここで xem , sm ∈ Fq) という表現となることを帰納的に確認することができる。

一方、Go は、定義体を Fq とする楕円曲線上の点であるから、任意の整数 n ∈ Z (n > 0) に対して

n・Go = (xon , yn) (ここで xon , yn ∈ Fq)

と表現される。

この整数倍の楕円曲線上の点の座標要素の構造から、任意の整数 n , m ∈ Z (n > 0 , m>0) に対して、

n・Go ≠ m・Ge

となっていることがわかる。これは、Go , Ge が定義体を Fq(ρ)とする楕円曲線上のアーベル群の異なる巡回群の生成元であることを示している。

(3) 有限体 Fq(ρ)上の楕円曲線の安全性

有限体 Fq 上の楕円曲線の位数を＃E(Fq)とすると、

2.1 項の仮定から、

＃E(Fq) ＝ u＊s （ s；大きな素数 ）である。

有限体上の楕円曲線の位数に対して、ハッセの定理が成立する[6]。有限体 Fq の位数は、q = p^n であり、有限体 Fq(ρ) の位数は、q^2 であるから、

$1+q-2\sqrt{q} \leqq \#E(Fq) \leqq 1+q+2\sqrt{q}$

$1+q^2-2q \leqq \#E(Fq(\rho)) \leqq 1+q^2+2q$

となり、ほぼ、＃E(Fq(ρ)) ＃E(Fq) * ＃E(Fq) --- ①

と考えることができる。

Go の生成する群の位数を＃<Go> = s とし、Ge の生成する群の位数を＃<Ge>とすると、

＃E(Fq) ≧＃<Go>

＃E(Fq(ρ)) ≧＃<Go>＊＃<Ge>

であるから、①式を考慮すると＃<Ge>が、＃<Go> = s と同じくらい大きな位数のものが存在するとしても不合理ではない。任意に設定したτから、ただちに暗号論的に使用可能な高位数の生成元 Ge が求められるとは限らないが、τ = f (xe)を計算し、試行錯誤的に位数を評価すれば、高位数の生成元 Ge が求められるものと考えられる。なお、定義体の拡大につてはいくつかの検討がなされているようである[7]。関係 τ = f (xe) は、求め方のひとつの例であって、Ge は別の方法で求められる、定義体 Fq(ρ)の楕円曲線上の点であっても良いと考えられる。

以下、アーベル群の構造定理により、Go , Ge は、定義体を Fq(ρ)とし、大きな素数位数 s を持つ異なる巡回群の生成元と仮定して議論を進める。

3. 2 次元ベクトル空間上での暗号化通信方式の構成

2 項の定めに従って、Go , Ge を定義体 Fq(ρ)の楕円曲線上の大きな素数位数 s を持つ、相異なる巡回群の生成元とする。任意の整数 m , n ∈ Z に対して、Q = n・Go + m・Ge は楕円曲線の有限部分群を構成し、

E = { Q | Q = n・Go +m・Ge ; m、n ∈ Zs }

とすると、E は、有限体 Zs 上の 2 次元ベクトル空間と考えることができる。次にこの 2 次元ベクトル空間 E に対して、暗号化通信システムを構成する。図 2 に 2 次元ベクトル空間上の暗号化、復号化の方式を示す。

3.1. 暗号化、復号化方式

(i) 鍵の構成

f , g ∈ Zs をもとに、Q = f・Go + g・Ge ;

Q1 = f ・Go , Q2 = g・Ge を定める。

Q が、公開鍵であり、{ f , g }の組が対応する秘密鍵である。なお、Q1、f は、代数拡大する前の定義体 Fq の楕円曲線上の公開鍵、および秘密鍵と考えることができる。

(ii) 多項式時間で計算可能な関数

π：Fq(ρ) ⟶ Z/Zq^2

ユーザの平文 m ∈ Z/Zq^2 (情報データ)

(iii) 情報データ(平文 m) の暗号化

乱数κ, τ∈ Zs (0≦κ, τ≦s－1)を生成する。

R1 = κ・Go , R2 = τ・Ge を計算する。

Q1 = f ・Go , Q2 = g・ Ge より、

T1 = κ・Q1 , T2 = τ・Q2 を求め

T = T1 + T2 = (Tx , Ty) を計算し、

c = π(Tx)⊕ m を求め

暗号データとして{ R1 , R2 , c } を作成する。

(iv) 情報データの復号化

ユーザが所有する秘密鍵 { f , g } により、

f・R1 + g・R2 = f・(κ・Go) + g・(τ・Ge)

= κ・Q1 + τ・Q2

= T1 + T2 = T = (Tx , Ty)

このようにして求めた T = κ・Q1 + τ・Q2

をもとに、受信した暗号化された情報データ c より

平文 m = π（Tx）⊕ c を取得する。

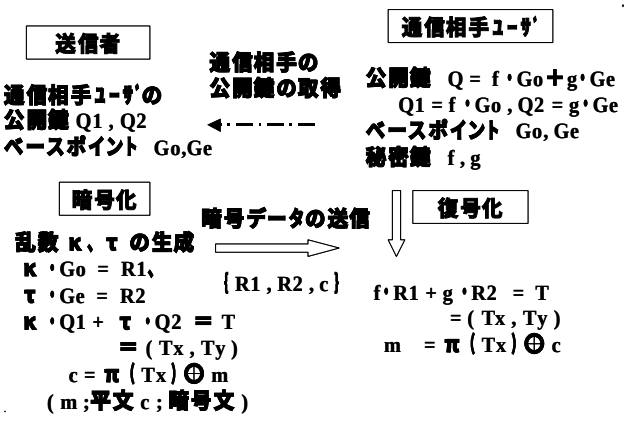

図 2 2 次元ベクトル空間上の暗号化、復号化の方式

3.2. 電子署名方式

2 次元ベクトル空間 W 上の暗号化通信システムにおいて、ユーザには、秘密鍵 { f , g } と対応する公開鍵 Q が割り当てられている。この公開鍵と秘密鍵をもとに、平文 m の電子署名を作成することができる。電子署名は、平文 m のハッシュ値を用いて作成するが、この平文 m を平文 m1 と平文 m2 の 2 つの部分に分割し、それぞれの個別のハッシュ値を用いて作成することも可能である。

(1)　電子署名の作成

(i) 乱数κ,τ∈ Zs（0≦κ,τ≦ s－1）を生成する。

(ii) この乱数をもとに楕円曲線上の点を定める。

R1 = κ・Go = (x1 , y1)

R2 = τ・Ge = (x2 , y2)

(iii) 楕円曲線の定義体はFq(ρ)であり、Go , Ge の定め方、および整数倍の座標構成より

x1 = xe1 ,　(y1 = ye1)

x2 = xe2 ,　(y2 = ρ*yi2)

(ここで xe1 , xe2 , ye1 , yi2 ∈ Fq)

と表現される。

Fq は、位数 q = p^n の有限体である。

一般に、この有限体の要素を　u ∈ Fq とすると

u = (a0 , a1, ・・・, an-1)

と表現される。 (0 ≦ ai ≦ p-1 ,　0 ≦ i ≦ n-1)

このu に対応する整数 nu を

$$nu = \sum_{i=0}^{n-1} ai * p^{i} \in Z$$

と定めることができる。

このようにして、xe1 , xe2 , (ye1 , yi2) ∈ Fq に対応する整数　nxe1 , nxe2 , (nye1 , nyi2) ∈ Z が定まる。この整数をもとに

r1 = nxe1 (mod s)　(または、r1 = nye1 (mod s))

r2 = nxe2 (mod s)　(または、r2 = q*nyi2 (mod s))

を求める。

(vi) 電子署名を作成する平文 m が、平文 m1 と平文 m2 とに分割されているものとする。各平文のハッシュ値

e1 = H(m1)　;　e2 = H(m2)　(H ; ﾊｯｼｭ関数　SHA-1)

を求める。

なお、平文 m を分割しない場合は、e1 = e2 = e = H(m) と、同じ平文 m のハッシュ値をとるものとする。

(v) これより、2 行 2 列のマトリックス

$$S = \begin{pmatrix} \frac{e1 + f}{\kappa} & \frac{r1}{\kappa} \\ \frac{g}{\tau} & \frac{e2 + r2}{\tau} \end{pmatrix}$$

が、定まる。(ﾏﾄﾘｯｸｽの各要素は mod s で定める。)

(vi) 以上より、{平文 m (m1 , m2) , ﾏﾄﾘｯｸｽ S , 公開鍵 Q , R1 , R2 } の組を電子署名データとして定める。

(2) 情報データ（平文 m1 , m2 ）の電子署名の検証

(1) 項で与えられる電子署名データに対して、下記の手順で、署名検証を実施することができる。

(i) R1 = κ・Go　= (x1 , y1)

R2 = τ・Ge　= (x2 , y2)

より、x1 , x2 (y1 , y2) ∈ Fq(ρ)を求め、(1) 項と同じ手順で、x1 , x2 (y1 , y2) に対応する整数 nxe1 , nxe2 , (または、nye1 , nyi2) を求め、

r1 = nxe1 (mod s)　(または、r1 = nye1 (mod s))

r2 = nxe2 (mod s)　(または、r2 = q*nyi2 (mod s))

を計算する。

(ii) 楕円曲線上のベースポイント Go , Ge より

R = r1・Go + r2・Ge を求める。

(iii) 平文 m1 , m2 のハッシュ値 e1 = H(m1) 、e2 = H(m2) を求め、マトリックス S の逆行列　C = S^{-1} を求める。

(iv) R、C、Q、およびハッシュ値 e1、e2、をもとに、次に示す、電子署名検証演算を実施する。

$$\begin{aligned} &(e1\cdot Go ,\ e2\cdot Ge)\,C + (Q , R)\,C \\ &= (Go , Ge)\begin{pmatrix} e1 & 0 \\ 0 & e2 \end{pmatrix} C + (Go , Ge)\begin{pmatrix} f & r1 \\ g & r2 \end{pmatrix} C \\ &= (Go , Ge)\begin{pmatrix} e1 + f & r1 \\ g & e2 + r2 \end{pmatrix} C \end{aligned}$$

ここで、マトリックス S の定め方から

$$S = \begin{pmatrix} \frac{1}{\kappa} & 0 \\ 0 & \frac{1}{\tau} \end{pmatrix}\begin{pmatrix} e1 + f & r1 \\ g & e2 + r2 \end{pmatrix}$$

よって

$$\begin{aligned} &(e1\cdot Go , e2\cdot Ge)\,C + (Q , R)\,C \\ &= (Go , Ge)\begin{pmatrix} \kappa & 0 \\ 0 & \tau \end{pmatrix} S\,C \\ &= (\kappa\cdot Go , \tau\cdot Ge) = (R1 , R2) \end{aligned}$$

(e1・Go, e2・Ge) C + (Q, R) C = (κ・Go , τ・Ge)

= (R1 , R2) が成立すれば、平文 m (m1 , m2) の正当性を確認したものとする。

3.3 電子署名の分割による作成

3.2 項において 2 次元のベクトル空間上、秘密鍵{ f , g } を用いて、平文 m (m1 , m2) に対して電子署名を与える 2 行 2 列のマトリックス S を与えた。このマトリックス S を以下の通り、S1、S2 の 2 行に分けて考える。

$$S = \begin{pmatrix} S1 \\ S2 \end{pmatrix} \qquad S1 = \begin{pmatrix} \frac{e1 + f}{\kappa} & \frac{r1}{\kappa} \end{pmatrix}$$

$$S2 = \begin{pmatrix} \frac{g}{\tau} & \frac{e2 + r2}{\tau} \end{pmatrix}$$

各 S1、S2 は、

S1：秘密鍵 f、ハッシュ値 e1、ベースポイント Go、乱数κ　で作成される。

S2 : 秘密鍵 g、ハッシュ値 e2、ベースポイント Ge、乱数τ　で作成される。

これは、S1、S2 それぞれ独立に作成できることを示している。即ち、秘密鍵 {f , g } を各々別の場所に格納し、そこで個別に、情報データ m1 , m2 に関する電子署名 S1 , S2 を作成し、作成した S1 , S2 を合成することで、電子署名を与えるマトリックス S が作成できることを示している。

4. 1対Nのデータ共有方式

以上、楕円曲線上において公開鍵を Q = f・Go + g・Ge と 2 個の秘密鍵 { f, g } を用いて表現することに着目し、1対1の場合の暗号化通信の方式について述べた。

さらに、この表現を1対N に適用した場合の暗号化通信の方式について述べる。

4.1. 結託攻撃の存在する1対N の暗号方式

まず簡単な、1 対 N の暗号方式から説明する。

ここでは、ベースポイントを Go とし、公開鍵を Q、秘密鍵を d とし、Q = d・Go とする。

次に秘密鍵 d に関する不定方程式を考える。

d = a*x + b*y　(a, b, x, y ∈ Zs)

これは、d, a, b を定数（固定要素）とし、x, y を未知数とする 1 次元の極めて単純な不定方程式である。

この解は多数存在するが、この解の組のひとつを

x = ξ, y = η とし、G1 = a・Go, G2 = b・Go とすると、

ξ・G1 + η・G2 = (a*ξ + b*η)・Go

= d・Go = Q

となる、これは、G1 , G2 と組み合わせれば、{ ξ, η } の値を用いて、公開鍵 Q で暗号化されたデータを復号化できることを示している。解の組{ ξi , ηi }(1≦ i ≦ N)を N 組定めれば、N のユーザが、それぞれ異なる秘密鍵 { ξi , ηi } を持ち、公開鍵 Q で暗号化されたデータを共有する（復号化する）ことが可能となる。

しかし、この方式には簡単な結託攻撃が存在する。

{σ, τ} をこの不定法方程式の他の解すると、

d = a*ξ + b*η

d = a*σ ＋b*τ

これより、{ρ, λ} を

ρ = α*ξ+β*σ , λ = α*η+β*τ

(ここで、α , βは、α＋β =1 を満足するものとする。) と定めれば、d = a＊ρ＋b＊λ　となり、a , b の値は求めなくとも、2 ユーザが結託すれば、公開鍵 Q を復号する別の秘密鍵の組 {ρ, λ} を構成できることを示している。このように簡単に結託攻撃が存在する原因は、秘密鍵 d とユーザに配布される秘密鍵{ x, y } とが、a, b の値が共通に固定されたまま、線型関係を持つ為である。

そこで、この結託攻撃を回避するため、秘密鍵 d とユーザに配布する秘密鍵 { x, y } との間に、線型関係が成立しないような鍵の構成について検討する。

4.2 双線形演算を利用した暗号化、復号化の方式

(1) 楕円曲線上の点に対する双線型形式

3 項において、素数を s とし有限体 Zs 上の 2 次元ベクトル空間 E を導入した。Go, Ge は素数位数 s の異なる巡回群の生成元であり、有限体 Zs 上のベクトル空間の1次独立な基底となっている。楕円曲線上の双線型形式としては、weil-paring または、Neron-Tate paring などが存在するが[8]、このような対称な双線型形式Φが、有限体上の楕円曲線に存在するとして、このベクトル空間 E 上の双線型形式Φを利用した暗号方式について述べる。

(2) システム側の鍵の構成

ベクトル空間 E において

f, g, c, d ∈ Zs　とし、

Q = f・Go + g・Ge

W = c・Go + d・Ge

Qa = (f*c)・Go , Qb = (g*d)・Ge

とする。ベクトル空間 E 上の対称な双線型形式Φをもとに、次の関数Ψを考える

Ψ(Q, W, Qa, Qb, Go, Ge)

= Φ(Q, W) - Φ(Qa, Go) - Φ(Qb, Ge)

= Φ(f・Go + g・Ge , c・Go + d・Ge)

- Φ((f*c)・Go , Go) - Φ((g*d)・Go , Go)

= (f*d + g*c)Φ(Go, Ge)

= (u + v) Lg

ここで、f*d = u ; g*c = v ; Φ(Go, Ge) = Lg とする。これより { f, g, c, d, u, v } をシステム側（管理者側）で鍵として保管し管理するものとする。次に、これらの鍵の値をもとにユーザに配布する鍵を構成する。

(3) ユーザ i に配布する鍵の構成

(i) まず、ai, bi ∈ Zs を任意に定める。

次に di, ci ∈ Zs　を

ai*di = u , bi * ci = v

を満足するように定める。このようにすることにより、4.1 項の方式と比較すると、a, b の値がユーザごとに変化する方式とすることができる。この値をもとに、

Qi = ai・Go + bi・Ge ; Si = ci・Go + di・Ge

Qai = (ai*ci)・Go , Qbi = (bi*di)・Ge

と定める。

(ii) 図 3 に示すように、ユーザ i には、鍵の組として { ai, bi, Si, Qai, Qbi }を割り当てて配布する。なお、公開鍵は、Qi = ai・Go + di・Ge となり、公開鍵 Qi と秘密鍵 { ai, bi } とを対応させて単独で運用することも可能である。

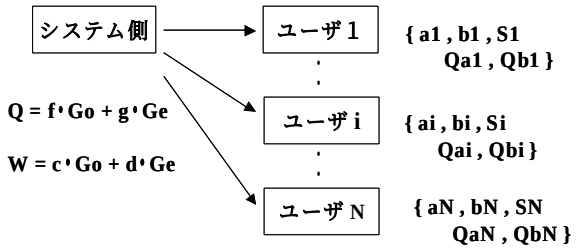

図 3 ユーザ i への鍵の配布

(4) 情報データ（平文 m）の暗号化

(i) 乱数κ, τ∈ Zs（0≦κ, τ≦ s－1）を生成する。

(ii) R1 = κ・Go 、R2 = τ・Ge を求め

Rs = f・R1 + g・R2

= (κ*f)・Go + (τ*g)・Ge を計算する。

Ψ(Rs, W, Qa, Qb, R1, R2)

= Φ(Rs, W) - Φ(Qa, R1) - Φ(Qb, R2)

$= (\kappa * f * d + \tau * g * c)\Phi(Go, Ge)$

$= (\kappa * u + \tau * v)\,Lg$　（ ここで、$Lg = \Phi(Go, Ge)$ ）

この $(\kappa * u + \tau * v)\,Lg$ の値をもとに、情報データ（平文 m）を暗号化して暗号文 c を作成し、

暗号データ｛ c , R1 , R2 ｝を作成する。

(5) 情報データの復号化

(i) ユーザ i に配布された秘密鍵 ｛ ai , bi ｝と R1, R2 より

$Ru = ai \cdot R1 + bi \cdot R2$

$= (ai * \kappa) \cdot Go + (bi * \tau) \cdot Ge$

を求める。次に、配布されている Si , Qai , Qbi より

$\Psi(Ru, Si, Qai, Qbi, R1, R2)$

$= \Phi(Ru, Si) - \Phi(Qai, R1) - \Phi(Qbi, R2)$

$= \Phi((ai * \kappa) \cdot Go + (bi * \tau) \cdot Ge,\ ci \cdot Go + di \cdot Ge)$

$-\Phi((ai*ci) \cdot Go,\ \kappa \cdot Go) - \Phi((bi*di) \cdot Ge,\ \tau \cdot Ge)$

$= (\kappa * ai * di + \tau * bi * ci)\ \Phi(Go, Ge)$

$= (\kappa * u + \tau * v)Lg$　（ ここで、$Lg = \Phi(Go, Ge)$ ）

(ii) $(\kappa * u + \tau * v)Lg$ が求まり、この値をもとに、暗号文 c を復号化し平文 m を取得することができる。

(6) 結託攻撃からの安全性について

楕円曲線の離散対数問題より、ユーザに配布される鍵 $Si = ci \cdot Go + di \cdot Ge$, $Qai = (ai*ci) \cdot Go$, および $Qbi = (bi*di) \cdot Ge$ の関係から、Si , Qai , Qbi をもとに、ci , di を求めることはできない。

この暗号システムでの結託攻撃とは、複数のユーザの秘密鍵{ ai , bi }の組から未知数として u , v, ci , di を求めることである。ここで、

$\Psi(Qi, Si, Qai, Qbi, Go, Ge)$

$= \Phi(Qi, Si) - \Phi(Qai, Go) - \Phi(Qbi, Ge)$

$= \Phi(ai \cdot Go + bi \cdot Ge,\ ci \cdot Go + di \cdot Ge)$

$-\Phi((ai*ci) \cdot Go,\ Go) - \Phi((bi*di) \cdot Ge,\ Ge)$

$= (ai*di + bi*ci)\Phi(Go, Ge)$

$= (u + v)Lg = z$　（ここで、$Lg = \Phi(Go, Ge)$ ）

となっている。

di を未知数 xi 、ci を未知数 yi として鍵の関係から

$ai * xi = u$　---- ①

$bi * yi = v$　---- ②

$(u + v)\,Lg = z$　---- ③

となる。未知数が u , v , xi , yi の 4 個に対して、方程式は、①，②，③ の 3 個が定まっている。結託するユーザの数が増加しても、未知数の数が方程式の数より 1 個多くなり、u , v の値を求めることはできない。

次に(1)項に述べた線型和による結託攻撃について検討する。ユーザに配布する鍵の構成法から、

$\Psi(Qi, Si, Qai, Qbi, Go, Ge)$

$= \Psi(Qj, Sj, Qaj, Qbj, Go, Ge)\ (i \neq j)$ となる。

$\alpha, \beta \in Zs$ とすると、

｛ $(\alpha \cdot Qi + \beta \cdot Qj)$, $(\alpha \cdot Si + \beta \cdot Sj)$ ｝が、(1) の結託攻撃の組合せとなるが、Ψは双線型形式のため、{ Qi , Sj } { Qj , Si } に対する Φの値 (Φ(Qi, Sj) , Φ(Qi, Sj))が、Ψの $\alpha * \beta$ の項に存在するため、Qi , Qj の線型和を用いた簡単な結託攻撃を実施することはできない。

なお、$\{(\alpha \cdot Qi), (\alpha^{-1} \cdot Si)\}$ の組は、双線型形式の性質から、別の鍵の組として複製し定めることができる。そこで、ユーザの秘密鍵を構成する場合、任意の $\alpha \in Zs$ に対して（ ai , bi ）$\neq \alpha$（ aj , bj ）

即ち、2 次元の射影平面で考えた場合、秘密鍵{ ai , bi } が異なる点となるように定めれば、このような複製を判定することができ悪用を防止することができる。

(7) 楕円曲線上の双線形形式についての検討

Silverman[8] 等[9]には、楕円曲線上の点 P に対して、標準的高さ（ canonical height ）[9]という演算　h(P) が定義され、これにより、楕円曲線上の点 P , Q に対して Neron-Tate paring とよばれる対称な双線型形式が与えられている。

$< P, Q> = h(P+Q) - h(P) - h(Q)$

有限体上の楕円曲線上にこのような、対称な双線型形式Φが存在するとした場合、この双線型形式Φは、テンソル空間 $E \otimes E$ からの線型写像 T で表現される。

即ち、$f, g, c, d \in Zs$　として、

$Q = f \cdot Go + g \cdot Ge$

$W = c \cdot Go + d \cdot Ge$

$\Phi(Q, W) = \Phi(W, Q) = T(Q \otimes W) = T(W \otimes Q)$

そこで、テンソル空間 $E \otimes E$ の $Ge \otimes Ge$, $Go \otimes Ge$ で生成される部分空間 Z への射影を Pz とし、T と Pz の合成写像を $L = T \circ Pz$ とすると、写像 L は、

$Pz(Q \otimes W) = (f * d)\,Go \otimes Ge + (g * c)\,Ge \otimes Go$

$L(Q \otimes W) = T(Pz(Q \otimes W))$

$= (f * d)\,T(Go \otimes Ge) + (g * c)\,T(Ge \otimes Go)$

$= (f * d + g * c)\Phi(Ge, Go)$

この L の値は、Ψの値 ((2)項で求めたΨの値) と同じものである。このようにΨの値を、Q, W だけから直接求める関数 L をプログラム化して作成できるなら、ユーザに Qai , Qbi の付加情報を与える必要はなく、より安全と考えられる。双線型型式Φが交代型式の場合も同様に、

$\Phi(Q, W) = (f * d - g * c)\,T(Ge \Lambda Go)$　（Λ ； 外積演算 ）

$= (f * d - g * c)\Phi(Go, Ge)$

となり、1 対 N の暗号方式を定めることができる。

また、付加情報も別の形式の与え方で定めたほうが、より安全とも考えられるが、これらの検討は、有限体上の楕円曲線の対称、または交代の双線型形式Φのプログラム化に向けた定式化とも併せて今後の課題としたい。

5. ネットワークセキュリティシステムへの応用

以上、楕円曲線上において公開鍵を $Q = f \cdot Go + g \cdot Ge$ と 2 個の秘密鍵 ｛ f , g ｝ を用いて表現し、1 対 1 で通信する場合の、暗号化、復号化、電子署名の作成、および検証の方式について述べ、また公開鍵 Q , W の双線型形式を利用した 1 対 N の暗号化通信の方式について述べた。

次に、セキュリティシステムへの応用として、秘密鍵 { f , g } を分散して管理する方式、1 対 N の暗号通信方式によるデータの共有方式、および電子署名を分割して作成する方式の利用法について述べる。

5.1. ネットワーク上のアクセス管理システム

(1) ユーザのアクセス管理

本提案の暗号化通信方式を通信ネットワーク上のユーザアクセス管理に適用した例について、図4を参照し説明する。この図4は、(無線)LANなどの一般的なネットワークを示している。ユーザN人がPCなどの端末で接続し、ネットワーク管理サーバが、秘密鍵等ユーザのアクセス管理を実施しているものとする。

各ユーザi(1≦i≦N)に割り当てられる秘密鍵の組を{ai, bi}とし、対応する公開鍵をQi = Qi1 + Qi2 (Qi1 =ai・Go, Qi2 = bi・Ge)とする。Qiで暗号化された情報データcは、秘密鍵{ai, bi}で復号化され平文mを取得することが出来る。

本システムでは、各ユーザiに、両方の秘密鍵を配布するのではなく、一方の秘密鍵aiだけを配布し、もう片方の秘密鍵biは、ネットワーク管理サーバが管理するものとする。次に、各ユーザiが、公開鍵Qiで暗号化された情報データを復号化する手順について説明する。

なお、暗号化データは、{平文mの暗号文c, R1=κ・Go, R2 = τ・Ge (κ, τ ;乱数)}の構成とする。

(i) ユーザiは、所有している秘密鍵aiにより、ai・R1を計算するとともに、ネットワーク管理サーバに秘密鍵操作要求を送出する。

(ii) ネットワーク管理サーバは、ある条件、たとえばユーザiの秘密鍵が紛失されるなどの不正がなく有効であると確認できる場合、bi・R2を計算し、ユーザiに配送する。また、bi・R2を発行するイベントにより、確実なユーザのアクセス履歴管理を実施することができる。

(iii) ユーザiは、取得したai・R1、bi・R2をもとに Ti = ai・R1＋bi・R2 を求め、暗号文cから平文mを復号し取得する。

通常、公開鍵Qiに関して、不正が発生した場合、棄却リストに載せて、公開鍵Qiの使用を防止するが、本システムでは、ユーザiにbi・R2の計算値の配送を停止することにが可能であり、物理的に公開鍵Qiで暗号化されたデータの復号を停止することが可能である。

(2) 階層構造を持たせたアクセス管理

(i) ユーザに配布する鍵の構成

4.2項の方式によるシステム側の鍵をQ、Wとし、Ψ(Q, W, Qa, Qb, Go, Ge) = (u + v) Lg とする。

ユーザiには ai*di = u, bi*ci = v を満足する{ci, di}の組で定められる公開鍵{Si = ci・Go + di・Ge, Qai = (ai*ci)・Go, Qbi = (bi*di)・Ge}が配布されており、(1)項で配布された秘密鍵と組み合わせた{ai, bi, Si, Qai, Qbi}により、システム側の鍵Q、Wで暗号化されたデータを復号化することができる。このようにして、複数のユーザi (1≦i≦N)が、相異なる秘密鍵を有し、システム側の一組の鍵Q, Wで暗号化されたデータを共有することができる。

(ii) 階層化した鍵の構成

システム側の鍵は、任意に複数個設定することができる。これを例えば、{Qα,Wα}{Qβ,Wβ}{Qγ,Wγ}の3組、および対応するΨの値をΨα, Ψβ, Ψγとし、α, β, γの記号で識別するものとする。

Ψα = (uα+vα)Lg

Ψβ = (uβ+vβ)Lg

Ψγ = (uγ+vγ)Lg

とすると、ユーザiには、配布する秘密鍵{ai, bi}は共通とし、{uα, vα}{uβ, vβ}{uγ, vγ}の値をもとに、α、β、γそれぞれを復号する鍵の組{Sαi, Qαai, Qαbi},{Sβi, Qβai, Qβbi},{Sγi, Qγai, Qγbi}を設定することができる。これは、ネットワーク上で暗号化するデータをα, β, γの3階層に分類し、αのデータを共有するユーザーに、鍵{Sαi, Qαai, Qαbi}を配布し、同様にβ、γのユーザに対応する鍵を配布することにより、図4に示すようにユーザをα, β, γ重複を許してグループ化し階層化できることを示している。

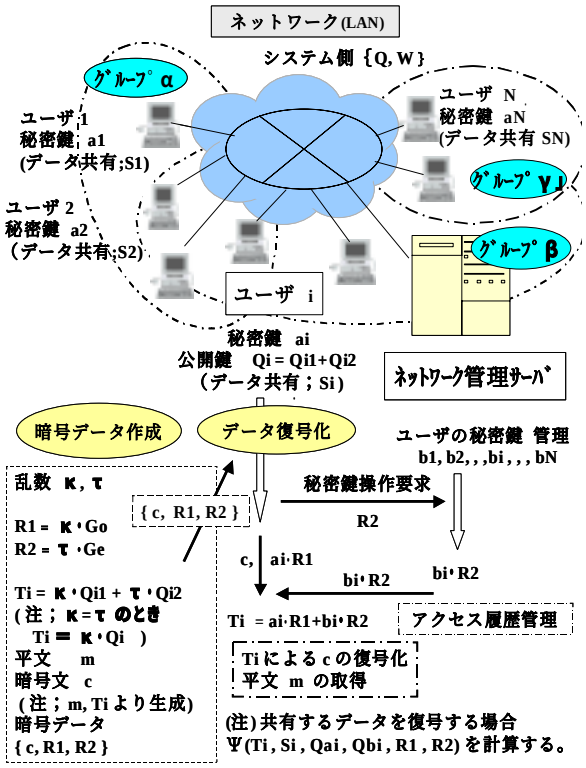

図4 ネットワーク上のアクセス管理システム

5.2. 分割して作成する電子署名の応用

3.2、3.3項において、公開鍵Q = f・Go + g・Ge、に対応する秘密鍵{f, g}を用い、平文m(平文 m1, m2)に対して、電子署名を構成するマトリックスSの要素を別々に作成し合成する方法について示した。ここでは、図5を参照し、移動体端末での電子決済の運用例について説明する。

この例では、ユーザ i の秘密鍵を{ fi , gi }, 公開鍵を Qi = fi・Go + gi・Ge、とし平文 m は、m1、m2 と 2 つの部分に分割されているものとする。

5.1 項と同様、インターネット上にネットワーク管理サーバを設置し、ユーザ i には一方の秘密鍵 fi のみが配布され、もう片方の秘密鍵 gi はネットワーク管理サーバ側で運用管理するものとする。また平文 m1 は移動端末側に、平文 m2 は、当該サーバ側で保有するものとする。

(i) ユーザ側での電子署名の作成

電子署名のマトリックス S1 を、秘密鍵 fi、ハッシュ値 e1 = H(m1)、ベースポイント Go、乱数κで作成する。

(ii) ネットワーク管理サーバ側での電子署名の作成

ユーザからの電子署名作成要求により、電子署名のマトリックス S2 を秘密鍵 gi、ハッシュ値 e2 = H(m2)、ベースポイント Ge 乱数τで作成する。

ここで、m2 は秘匿性の高い個人の共通情報、たとえば個人の銀行口座のなどが、適用できると考えるが、ユーザの証明書の代行として利用することも可能ではないかと考えられる。

このようにすれば、m2 の情報は移動体端末に保管する必要がなく、またこの端末が盗難に会った場合、ネットワーク管理サーバ側での電子署名 S2 の作成を停止すれば、電子署名の悪用を物理的に防止することができる。

別の運用例として、移動体端末側を A 社、サーバ側を B 社など、異なる組織とし、何らかの契約書、文書などを別々に承認する場合も、A 社 B 社別々に電子署名 S1、S2 を作成し、これらを合成することができる。

このように、本署名方式は分散して存在する情報データに対して、関連付けの正当性を確認する有効な手段を与えるものと考えられる。

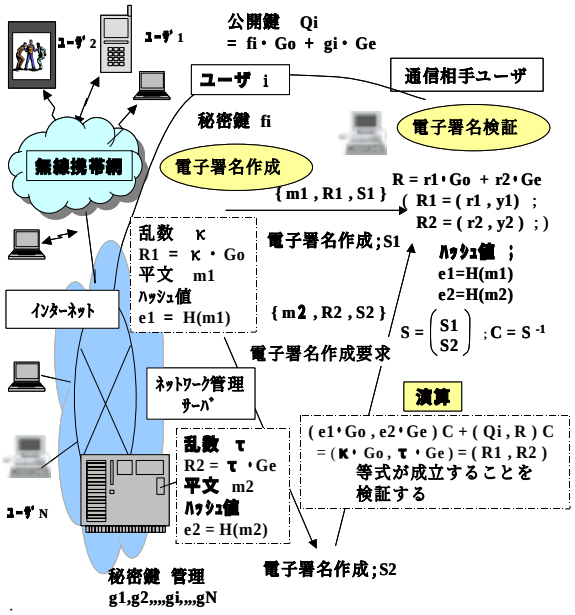

図 5　移動体端末での電子決済の運用例

6.　まとめ

本論文では、楕円曲線上において、公開鍵 Q が 2 個の秘密鍵 f、g により Q = f・Go +g・Ge と表現されることを利用した、暗号化通信方式について検討したものである。全体を通して鍵はシステム側（信頼のおける第 3 者）が管理するほうが適切だという立場をとり、

(1) 公開鍵に対応する秘密鍵を分散して管理する方法

(2) 1 対 N の方式でデータを共有する方法

などについて述べた。利用する楕円曲線は、既存の暗号論的に安全な楕円曲線を用いて構成する方式であるため、かなり安全なシステムでないかと考えられる。

公開鍵 Q を 2 個の秘密鍵 f、g で表現できることは、RSA システムにはない楕円曲線暗号の特徴だと考えられるが、本提案の方式が有効であれば、さらに具体的なプログラム化や様々な応用について検討して行きたい。

謝辞

提案者は、（株）メビウスという創立後まもない小さな企業に属している。それにもかかわらず本論文の執筆の機会を与えていただいた坂本社長に深く感謝致します。

また、社外の専門家の方々にも貴重な御指導をいただいておりまして、心より感謝致します。

参考文献

[1] 「暗号技術評価報告書　CRYPTREC　Report 2002」
平成 15 年 3 月　情報処理振興事業協会
通信・放送機構

[2] 「暗号と情報セキュリティ」
昭晃堂　辻井重男　笠原　正雄　編著

[3] 「筑波大学集中講義資料」
北陸先端科学技術大学院大学　宮地　充子：
(http://www.risk.sie.tsukuba.ac.jp/~kame/topics_1/risk_eng_topics1_2001.pdf)

[4] 「楕円曲線暗号」　ピアソン・エデユケーション
（イアン・F・ブラケ、カディエル・セロッシ、
ナイジェル・P・スマート＝著　鈴木　治郎＝訳

[5] 数学アルゴリズム問題の研究調査
(http://www.ipa.go.jp/security/enc/CRYPTREC/fy15/cryptrec20030424_outrep01.html)

[6] 「代数曲線入門」日本評論社　（梶原　健　著）

[7] 「代数体の拡大で rank が増える楕円曲線について]
東京都立大学大学院理学研究科　松野　一夫　：
(http://www.math.is.tohoku.ac.jp/~taya/sendaiNT/2000/matsuno.pdf)

[8] Joseph H.Silverman「The Arithmetic of Elliptic Curves」
GTM 106, Springer-Verlag, New York 1986

[9] 「楕円曲線上の標準的高さについて」
中央大学　土屋和由：
(http://www.chuo-u.gr.jp/papers/tsuchiya_2002_ichigaya.pdf)