ヴェイユペアリングを用いたグループ署名方式

岡崎 裕之 * 境 隆一 ** 柴山 潔 * 笠原 正雄 ***
* 京都工芸繊維大学 ** 大阪電気通信大学 *** 大阪学院大学

1 はじめに

グループ署名は，グループに属するメンバの誰もがグループの代表として署名することを可能とするディジタル署名方式である [1]．グループ署名においては，署名がグループのメンバによって正当に署名されたことを誰もが検証できる．ただし，署名者を特定することが可能であるのはグループの管理者のみである．グループ署名方式では，以上のような機能を実現するために，グループのメンバの登録と署名者の特定を行う管理者は非常に大きな権限を有しており，このことが安全性の点において問題をもたらしている．このためグループ署名方式の安全性をより確かにするために，管理者の権限を制限する方法が従来より検討されてきた [2, 3]．しかしながら依然として従来の方式においては管理者が無制限にメンバの追加を行えるという問題が残されたいる [2, 3, 4]．更に従来の方式は，既存のディジタル署名方式を組み合わせることによってグループ署名方式を実現しているため，必ずしも単純ではない．本論文では，ヴェイユペアリングを用いることにより，先に述べたような問題が解決し，更に従来の方式に比べ，より単純でしかも効率の点で優れたグループ署名方式を提案する．

2 準備

2.1 グループ署名

グループ署名はチャウムとヘイストによって初めて提案された署名方式である [1]．このグループ署名方式の機能は以下のとおりである．

1. グループのメンバのみが署名可能である．
2. 誰もがグループの署名文の正当性を検証することは可能であるが，署名文を生成したメンバの特定は不可能である．
3. あらかじめ決められた権限を有する者によってのみ署名者の特定が可能である．

以上のような機能を実現するために，グループ署名方式は少なくとも以下の性質を有する必要がある [3]．

Unforgeability：署名文の偽造は行えない．

Anonymity：いかなる署名文に対しても，あらかじめ決められた管理者以外は署名者を特定することが困難である．

Unlinkability：異なる 2 つの署名文が，同一のメンバによって生成されたか否かを判定することは，あらかじめ決められた管理者以外には困難である．

Identifiability：あらかじめ決められた管理者は，いつでもいかなる署名文に関しても，その署名者を特定できる．

exculpatability : メンバは，自分自身が生成した署名文以外に関して責任を負わされることはない．

一般にグループ署名方式では，メンバの登録や署名者の特定を行う管理者が大きな権限を有している．そのため，管理者からメンバを保護することが重要である．

2.2 ヴェイユペアリング

本節ではヴェイユペアリングについて必要最小限を述べる．なお，ヴェイユペアリングの詳細については文献 [5] を，暗号分野での応用に関しては文献 [6, 7, 8, 9] をそれぞれ参照されたい．

ヴェイユペアリング $e_n(\cdot,\cdot)$ は以下のような写像である．

$$\left.\begin{array}{rcl} E[n]\times E[n] & \to & \mu_n \in \mathbb{F}_{q^\kappa}^* \\ (P,Q) & \mapsto & e_n(P,Q) \end{array}\right\} \tag{1}$$

ただし，κ を $E[n] \subset E(\mathbb{F}_{q^\kappa})$ を満たす最小の整数，μ_n を $\mathbb{F}_{q^\kappa}^*$ 上の 1 の n 乗根の成す群とする．

以下にヴェイユペアリングのよく知られた性質を示す．

性質 1 (非退化)

$$e_n(P,Q) = 1 \quad {}^\forall P \in E[n] \quad \text{iff} \quad Q = \mathcal{O} \tag{2}$$

性質 2 (双線型)

$$\left.\begin{array}{l} e_n(P_1+P_2,Q) = e_n(P_1,Q)e_n(P_2,Q) \\ e_n(P,Q_1+Q_2) = e_n(P,Q_1)e_n(P,Q_2) \end{array}\right\} \tag{3}$$

性質 3 (反対称)

$$e_n(P,Q) = e_n(Q,P)^{-1} \tag{4}$$

式 (2)， (3) 及び (4) より以下の性質を得る．

性質 4

$$e_n(P,P) = 1 \tag{5}$$

性質 5

$$\left.\begin{array}{l} e_n(sP,Q) = e_n(P,Q)^s \\ e_n(P,sQ) = e_n(P,Q)^s \end{array}\right\} \tag{6}$$

更に式 (5) 及び 式 (6) より以下の性質を得る．

性質 6

$$e_n(aP,bP) = e_n(P,P)^{a\cdot b} = 1 \tag{7}$$

性質 7 $R = aP + bQ$ のとき，

$$\left.\begin{array}{l} e_n(R,Q) = e_n(P,Q)^a \\ e_n(P,R) = e_n(P,Q)^b \end{array}\right\} \tag{8}$$

ただし，$a, b \in \mathbb{Z}$ とする．

3　提案方式

3.1　概要

一般にグループ署名方式では，管理者がメンバの登録及び署名者の特定を行う権限を有している．従来のグループ署名方式では，管理者による不正を抑制するために，管理者の権限を制限するさまざまな方法が検討されてきた．例えば [2, 3] では，メンバの登録と署名者の特定を異なる管理者が行えるような方式が提案されている．しかしながらこれらの方式では，メンバの登録を行う管理者が自由にメンバを追加することが可能であるため，メンバの誰のものでもない不正な署名鍵を発行することが可能であった．管理者は，そのような不正な署名鍵を用いて，メンバによって署名されたものではない不正な署名文を生成することが可能であった．本論文では，メンバの登録を行う管理者がまず仮登録を行い，更に別の管理者による承認を得ることによって初めてメンバの登録が行えるような方式を提案する．これにより，管理者が不正な署名文を生成することが困難となる．本方式では，メンバの承認をグループのメンバの一部，あるいは全員が行うようなことも可能であるので，より公正なグループ運営を行うことが期待できる．また，本方式は，他の方式に比べて署名文生成及び検証がより単純に可能である．

本方式では，以下の記号を使用する．

$\mathcal{G}$:グループ

$\mathcal{M}_i$: $\mathcal{G}$ に属する第 i 番目のメンバ

$\mathcal{M}_S$: 署名者

$\mathcal{V}$: 検証者

$\mathcal{RA}$: メンバ登録管理者

$\mathcal{AA}_j$: 第 j 番目のメンバ承認管理者

$\mathcal{IA}$: 署名者特定管理者

本方式ではグループ $\mathcal{G}$ に t 人のメンバ，$\{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_t\}$ が属するものとする．$\mathcal{M}_i$ はグループ $\mathcal{G}$ を代表して署名を行うことが可能である．署名者はグループ $\mathcal{G}$ に属するメンバのいずれかであるが，以下では署名者が特定されていない場合に限り，署名者を $\mathcal{M}_S$ と表記する．メンバ登録管理者 $\mathcal{RA}$ は，グループを運営するためにシステムパラメータを準備し，更にメンバの仮登録を行う．メンバ承認管理者 $\mathcal{AA}$ は，仮登録されたメンバに承認を与える権限を有する．まず $\mathcal{AA}$ は，メンバ登録管理者 $\mathcal{RA}$ が作成した仮公開鍵を変更するとともに，仮登録されたメンバの署名鍵を変更する．本論文では τ 人のメンバ承認管理者 $\{\mathcal{AA}_1, \mathcal{AA}_2, \ldots, \mathcal{AA}_\tau\}$ が存在するとする．本方式では，$\mathcal{RA}$ は $\{\mathcal{AA}_1, \mathcal{AA}_2, \ldots, \mathcal{AA}_\tau\}$ の承認無しには新たなメンバを登録することが不可能である．更に，$\{\mathcal{AA}_1, \mathcal{AA}_2, \ldots, \mathcal{AA}_{\tau-1}\}$ のうちの何人か，あるいは全員を $\mathcal{G}$ のメンバが兼務することが可能である．したがって，本方式では管理者 $\mathcal{RA}$ の権限をメンバが抑制することが可能である．また署名者特定管理者 $\mathcal{IA}$ は署名者の特定を行う権限を有し，問題が起こった署名文の署名者を特定して，署名者と署名文の関係を証明することができる．ただし $\mathcal{IA}$ はメンバ承認管理者のうちの一人で，最後に承認を行うメンバ承認管理者 $\mathcal{AA}_\tau$ が兼務する．

3.2 準備

本方式では，まずメンバ登録管理者 $\mathcal{RA}$ がグループのシステムを設定し，メンバ登録の準備を行う．そして，$\mathcal{RA}$ は各メンバと対話的にメンバの仮署名鍵の生成を行って，メンバの仮登録を行う．メンバ登録管理者 $\mathcal{RA}$ によって仮登録されたメンバは，τ 人のメンバ承認管理者 $\mathcal{AA}_1, \ldots, \mathcal{AA}_\tau$ の承認を受けることによって初めて登録が完了する．

3.2.1 システムパラメータ

まず，$\mathcal{RA}$ は素数 n 及び $E[n] \subset E(\mathbb{F}_{q^\kappa})$ を選び，$E/\mathbb{F}_q$，ハッシュ関数 $h(\cdot)$，ペアリング $e_n(\cdot,\cdot)$ 及び $E(\mathbb{F}_q^\kappa)$ を公開する．次に，$\mathcal{RA}$ は秘密情報として $b_R \in \mathbb{Z}/n\mathbb{Z}$ を選ぶ．$\mathcal{RA}$ は更に 2 つの n ねじれ点 $P \in E[n]$ 及び $Q \in E[n]\backslash\langle P\rangle$ を選ぶ．ただし，以下の条件

条件 1 :$\phi(\cdot)$ を q 乗フロベニウス写像 [5] としたときに $\phi(P) \neq qP, \phi(Q) \neq qQ$ である．

条件 2 :q^κ のサイズ $|q^\kappa|$ が 1024 ビット以上であり，n が 160 ビット以上の素数である．

を満たすものとする*．最後に $\mathcal{RA}$ は

$$\left.\begin{array}{c} g = e_n(P,Q) \\ y_R = e_n(P,Q)^{b_R} \end{array}\right\} \tag{9}$$

を計算し，最後に (g, y_R, P, Q) を $\mathcal{G}$ のメンバと他の管理者に公開する．

3.2.2 メンバ仮登録

$\mathcal{M}_i$ の仮登録のために，$\mathcal{RA}$ と $\mathcal{M}_i$ は対話的に $\mathcal{M}_i$ の仮署名鍵 (K_{Ri}, L_{Ri}) †を以下のように生成する．

Step 1 : $\mathcal{M}_i$ は $r_i \in \mathbb{Z}/n\mathbb{Z}$ を選び，$\mathcal{RA}$ に $\frac{1}{r_i}Q$ を秘密裏に送信する．

Step 2 : $\mathcal{RA}$ は $s_i \in \mathbb{Z}/n\mathbb{Z}$ を選び，$\mathcal{M}_i$ に

$$\left.\begin{array}{l} s_iP + \frac{b_R}{r_i}Q \\ L_{Ri} = \frac{1}{r_i s_i}Q = \frac{1}{a_{Ri}}Q \end{array}\right\} \tag{10}$$

を秘密裏に送信する．ただし $a_{Ri} = r_i s_i$ とする．

Step 3 : $\mathcal{M}_i$ は仮署名鍵

$$\begin{aligned} K_{Ri} &= r_i\left(s_iP + \frac{b_R}{r_i}Q\right) \\ &= r_i s_i P + b_R Q = a_{Ri}P + b_R Q \end{aligned} \tag{11}$$

を計算する．

Step 4 : $\mathcal{M}_i$ は

$$\left.\begin{array}{l} e_n(K_{Ri}, L_{Ri}) = e_n(P,Q) = g \\ e_n(P, K_{Ri}) = e_n(P,Q)^{b_R} = y_R \end{array}\right\} \tag{12}$$

が成り立つ場合に限り (K_{Ri}, L_{Ri}) を受理する．

*条件 1 を満たさない場合には ECD が計算可能であり [10]，条件 2 を満たさない場合には $ECDH$ 若しくは DH が必ずしも困難であるとは言えず，本方式の安全性を保証できなくなる [7, 8].

†K_{Ri} には $\mathcal{M}_i$ と $\mathcal{RA}$ の秘密情報がともに含まれていることに注意したい．

3.2.3 メンバ登録

まず，メンバ承認管理者 $\mathcal{AA}_j(j=1,2,\ldots,\tau)$ は $\mathcal{RA}$ が公開したグループグループ $\mathcal{G}$ の公開鍵を更新する．次に，$\mathcal{AA}_j(j=1,2,\ldots,\tau)$ はメンバの仮署名鍵を更新することによりメンバを承認する．以上によりメンバの登録が完了する．

公開鍵更新： メンバ承認管理者 $\mathcal{AA}_j(j=1,2,\ldots,\tau)$ は以下のようにグループ $\mathcal{G}$ の仮公開鍵 y_R を更新し，公開鍵 y を生成する．

Step 1： $\mathcal{AA}_1$ は $z_1 \in \mathbb{Z}/n\mathbb{Z}$ を選び，$\mathcal{AA}_2$ に $y_1 = {y_R}^{z_1}$ を送信する．$j=2,3,\ldots,\tau$ のとき，以下の Steps 2， 3 を繰り返す．

Step 2： $\mathcal{AA}_j$ は y_{j-1} を $\mathcal{AA}_{j-1}$ から受け取り，$z_j \in \mathbb{Z}/n\mathbb{Z}$ を選んで $y_j = {y_{j-1}}^{z_j}$ を計算する．

Step 3： $j<\tau$ のとき，$\mathcal{AA}_j$ は y_j をに $\mathcal{AA}_{j+1}$ 送信し，Step 2 に戻る．$j=\tau$ のとき，$y=y_\tau$，$b=zb_R$, $z=\prod_{j=1}^{\tau} z_j$ とする．このとき，$y=g^b$ である．$\mathcal{IA}$ は $r \in \mathbb{Z}/n\mathbb{Z}$ を選び，$P_r = rP$ を計算する．$\mathcal{IA}$ は $\mathcal{AA}_\tau$ が兼任することに注意したい．最後に $\mathcal{IA}$ は

$$(E/\mathbb{F}_q, g, y, P_r, n) \tag{13}$$

をグループ $\mathcal{G}$ の公開鍵として公開する．

メンバ承認： $\{\mathcal{AA}_1, \mathcal{AA}_2, \ldots, \mathcal{AA}_\tau\}$ は以下のようにして $\mathcal{M}_i$ の承認を行う．

Step 1： $\mathcal{M}_i$ は $r_i' \in \mathbb{Z}/n\mathbb{Z}$ を選び，$\mathcal{AA}_1$ に $r_i' K_{Ri}$ を送信する．更に $\mathcal{M}_i$ は $g_i = g^{r_i'}$ を公開する．

Step 2： $\mathcal{RA}$ は $\mathcal{AA}_1$ に L_{Ri} を送信する．

Step 3： $\mathcal{AA}_1$ は $w_{i,1} \in \mathbb{Z}/n\mathbb{Z}$ を選び，$\mathcal{AA}_2$ に

$$\left.\begin{aligned} K_{i,1} &= w_{i,1} z_1 r_i' K_{Ri} \\ L_{i,1} &= \tfrac{1}{w_{i,1} z_1} L_{Ri} \\ P_{i,1} &= \tfrac{1}{w_{i,1}} P \\ Q_{i,1} &= w_{i,1} Q \end{aligned}\right\} \tag{14}$$

を送信する．$j=2,3,\ldots,\tau$ のとき Step 3， 4，5 を繰り返す．

Step 4： $\mathcal{AA}_j$ は $w_{i,j} \in \mathbb{Z}/n\mathbb{Z}$ を選び，

$$\left.\begin{aligned} K_{i,j} &= w_{i,j} z_j K_{i,j-1} \\ L_{i,j} &= \tfrac{1}{w_{i,j} z_j} L_{i,j-1} \\ P_{i,j} &= \tfrac{1}{w_{i,1}} P_{i,j-1} \\ Q_{i,j} &= w_{i,1} Q_{i,j-1} \end{aligned}\right\} \tag{15}$$

を計算する．

Step 5： $j<\tau$ のとき，$\mathcal{AA}_j$ は $\mathcal{AA}_{j+1}$ に $(K_{i,j}, L_{i,j}, P_{i,j}, Q_{i,j})$ を送信する．次に Step 3 に戻り $\mathcal{AA}_{j+1}$ が同様の処理を行う．

$j=\tau$ のとき，最後のメンバ承認管理者 $\mathcal{AA}_\tau$，すなわち署名者特定管理者 $\mathcal{IA}$ は $\mathcal{M}_i$ に $(K_{i,\tau}, L_i = L_{i,\tau}, P_i = P_{i,\tau}, Q_i = Q_{i,\tau})$ を送信し，$y_i = e_n(P_i, K_{i,\tau}) = y^{r_i'}$ を $\mathcal{M}_i$ の公開鍵として公開する．$\mathcal{IA}$ は $(K_{i,\tau}, L_i, P_i, Q_i)$ を $\mathcal{M}_i$ の署名者特定用鍵として秘密裏に保持する．

Step 6 : $\mathcal{M}_i$ は K_i を以下のようにして計算する.

$$K_i = \frac{1}{r_i'} K_{i,\tau} \tag{16}$$

$\mathcal{M}_i$ は

$$\left.\begin{array}{l} e_n(K_i, L_i) = g \\ e_n(P_i, Q_i) = g \\ e_n(P_i, K_i) = y \\ y_i = y^{r_i'} \end{array}\right\} \tag{17}$$

が成り立つときにのみ (K_i, L_i, P_i, Q_i) を受理し，そうでなければ Step 1 に戻ってメンバ承認を再び行う.

以上のメンバ承認が完了したときに，$\mathcal{M}_i$ の鍵 (K_i, L_i, P_i, Q_i) は以下のようになる.

$$\left.\begin{array}{l} K_i = w_i\,(a_i P + bQ) \\ L_i = \frac{1}{w_i a_i} Q \\ P_i = \frac{1}{w_i} P \\ Q_i = w_i Q \end{array}\right\} \tag{18}$$

ただし， $a_i = z a_{Ri}$， $w_i = \prod_{j=1}^{\tau} w_{i,j}$ である.

3.3 署名生成

署名者 $\mathcal{M}_S$ はグループ $\mathcal{G}$ のメンバである．$\mathcal{M}_S$ はメッセージ m に対するグループ $\mathcal{G}$ のグループ署名 (R, S) を以下のように生成する.

まず $\mathcal{M}_S$ は $k \in \mathbb{Z}/n\mathbb{Z}$ を選択する．${d_S}^k = 1$ のとき， 署名偽造を抑制するために k を選び直す．次に $\mathcal{M}_S$ は

$$\left.\begin{array}{l} R = k(h({d_S}^k) P_S + L_S) \\ \quad = \frac{k}{w_S}\left(h({d_S}^k) P + \frac{1}{a_S} Q\right) = (X, Y) \\ S = \frac{X}{k} K_S + \frac{h(m)}{k} Q_S \\ \quad = \frac{w_S}{k}\,(X a_S P + (h(m) + Xb) Q) \end{array}\right\} \tag{19}$$

を計算する．ただし，X，Y はそれぞれ点 R の x 座標，y 座標であり，また $d_S = e_n(P_r, L_S) = g^{\frac{r}{w_S a_S}}$ とする.

3.4 署名検証

検証者 $\mathcal{V}$ は以下のように署名検証を行う.

まず $\mathcal{V}$ は

$$v = e_n(P_r, R) = g^{\frac{kr}{w_S a_S}} = {d_S}^k \tag{20}$$

を計算する．$v = 1$ の場合には，署名が偽造されたものとして棄却する．更に

$$e_n(R, S) = g^{h(m)h(v) - X} \cdot y^{h(v)X} \tag{21}$$

が成り立つときにのみ $\mathcal{V}$ は署名を受理する.

3.5 署名者特定

署名者特定管理者 $\mathcal{IA}$ は $L_i(i=1,2,\ldots,t)$ を用いて以下のように署名者特定を行う．

$\mathcal{IA}$ は

$$e_n(P_i,R)^{h(v)} = e_n(R,L_i) \tag{22}$$

が成り立つ場合に $\mathcal{M}_i$ が 署名者 $\mathcal{M}_S$ であると特定する．

このとき， 必要とあらば $\mathcal{IA}$ は $\mathcal{M}_i$ が署名者であることを以下のように証明することが可能である．

まず， $\mathcal{IA}$ は $k' \in \mathbb{Z}/n\mathbb{Z}$ を選択し，

$$\left.\begin{array}{l} R_c = k'\left(h(v)P_i + L_i\right) = \frac{k'}{w_i}\left(h(v)P + \frac{1}{a_i}Q\right) \\ S_c = \frac{1}{k'}\left(K_{i,\tau} + rQ_i\right) \\ \quad = \frac{r'w_i}{k'}\left(a_iP + bQ\right) + \frac{rw_i}{k'}Q \end{array}\right\} \tag{23}$$

を計算する．次に， $\mathcal{IA}$ は $\mathcal{M}_i$ が署名者であることを公表するとともに (R_c,S_c) を公開する．このとき，$\mathcal{V}$ は $\mathcal{M}_i$ が署名者であることを当該の署名 (R,S)，公開鍵 (P_r,Q,g_i,y_i)，及び (R_c,S_c) を用いて以下のように検証する．

更に $\mathcal{V}$ は

$$e_n(R_c,R) = 1 \tag{24}$$

が成り立つか否かを計算する．式 (24) が成り立つ場合には R_c と R が同じ鍵を含むことが判明する．次に $\mathcal{V}$ は

$$e_n(R_c,S_c) = y_i^{h(v)} \cdot g_i^{-1} \cdot e_n(P_r,Q)^{h(v)} \tag{25}$$

が成り立つか否かを計算する．式 (25) が成り立つ場合には R_c が $\mathcal{M}_i$ を含むことが判明する．以上より，式 (24)， (25) が成り立つ場合，$\mathcal{V}$ は $\mathcal{M}_i$ が署名者であったと判断することが可能である．

4 安全性考察

4.1 準備

n を素数， $g \in \mu_n, x,y \in \mathbb{Z}/n\mathbb{Z}$ とする．更に，$P \in E[n], Q \in E[n]\backslash\langle P\rangle$ は $\phi(P) \neq (q \mod n)P$ かつ $\phi(Q) \neq (q \mod n)Q$ を満たすとする．ただし，$\phi(\cdot)$ を q 乗フロベニウス写像とする‡.

定義 1 *"DL"* (Discrete Logarithm) とは
既知の (g^x,g) より， x を計算するアルゴリズムであり，$DL(g^x,g) = x$ と表す.

定義 2 *"ECDL"* (Elliptic Curve Discrete Logarithm) とは
既知の (xP,P) より，x を計算するアルゴリズムであり，$ECDL(xP,P) = x$ と表す.

定義 3 *"ECDH"* (Elliptic Curve Diffie-Hellman) とは
既知の (xP,yP,P) より，xyP を計算するアルゴリズムであり，$ECDH(xP,yP,P) = xyP$ と表す.

‡もし $\phi(P) \neq (q \mod n)P$ または $\phi(Q) = (q \mod n)Q$ かつ，$\kappa \geq 2$，のような特殊な条件を満たしているならば，$ECD(xP+yQ,P,Q)$ が計算可能である [10].

定義 4 *"DDH"*(Decisional Diffie-Hellman) とは
既知の (g^x, g^y, g^z) より，$g^z = g^{xy}$ であるか否かを判定するアルゴリズムであり，

$$DDH(g^x, g^y, g^z) = \begin{cases} 1 & if \quad g^z = g^{xy} \\ 0 & if \quad g^z \neq g^{xy} \end{cases}$$

と表す.

定義 5 *"ECIDH"*(Elliptic Curve Inverse Diffie-Hellman) とは
既知の (xP, P) より，$\frac{1}{x}P$ を計算するアルゴリズムであり，$ECIDH(xP, P) = \frac{1}{x}P$ と表す.

定義 6 *"ECcDH"*(Elliptic Curve co Diffie-Hellman[11]) とは
既知の (xP, P, Q) より xQ を計算するアルゴリズムであり，$ECcDH(xP, P, Q) = xQ$ と表す.

定義 7 *"ECIcDH"* (Elliptic Curve Inverse co Diffie-Hellman) とは
既知の (xP, P, Q) より $\frac{1}{x}Q$ を計算するアルゴリズムであり，$ECIcDH(xP, P, Q) = \frac{1}{x}Q$ と表す.

定義 8 *"ECIcDH′"* とは
既知の (g^x, g, Q) より $\frac{1}{x}Q$ を計算するアルゴリズムであり，$ECIcDH'(g^x, g, Q) = \frac{1}{x}Q$ と表す.

定義 9 *"$ECDS_1$"*(Elliptic Curve Denominator Summation 1) とは
既知の $(\frac{1}{x}P, y, P)$ より $\frac{1}{x+y}P$ を計算するアルゴリズムであり，$ECDS_1(\frac{1}{x}P, y, P) = \frac{1}{x+y}P$ と表す.

定義 10 *"$ECDS_2$"*(Elliptic Curve Denominator Summation 2) とは
既知の $(\frac{1}{x}P, \frac{1}{y}P, P)$ より $\frac{1}{x+y}P$ を計算するアルゴリズムであり，$ECDS_2(\frac{1}{x}P, \frac{1}{y}P, P) = \frac{1}{x+y}P$ と表す.

定義 11 *"$ECcDS_1$"*(Elliptic Curve co Denominator Summation 1) とは
既知の $(\frac{1}{x}P, y, P, Q)$ より $\frac{1}{x+y}Q$ を計算するアルゴリズムであり，$ECcDS_1(\frac{1}{x}P, y, P, Q) = \frac{1}{x+y}Q$ と表す.

定義 12 *"ECD"*(Elliptic Curve Decomposition) とは
既知の $(xP+yQ, P, Q)$ より (xP, yQ) を計算するアルゴリズムであり，$ECD(xP+yQ, P, Q) = yQ$ と表す．ただし xP は $ECD(xP + yQ, P, Q)$ を用いて $xP = xP + yQ - ECD(xP + yQ, P, Q)$ のように計算可能であることに注意したい.

上述の問題に関して以下の定理を得る．ただし，本論文ではアルゴリズム A がアルゴリズム B に多項式時間で帰着されることを "$A \Rightarrow B$" と表記する.

定理 1 $ECDH \Rightarrow ECIDH$

証明 (概略)
以下では光成らによる証明の概要を示す [12][§].
既知の xP, yP, 及び P より $ECIDH$ を用いて以下のように $ECDH$ を得る.

$$\begin{aligned} xyP &= ECDH(xP, yP, P) \\ &= \frac{ECIDH(P, xP+yP)}{2} - \frac{ECIDH(P, xP)}{2} - \frac{ECIDH(P, yP)}{2} \end{aligned} \tag{26}$$

□

[§]光成らは [12] において $ECIDH$ を "$1w - DHA$" と呼称している.

定理 2 $ECDH \Rightarrow ECcDH,\ ECDH \Rightarrow ECIcDH$

証明 (概略)
既知の xP, yP, 及び P より $ECcDH$ または $ECIcDH$ を用いて以下のように $ECDH$ を得る．まず， $Q \in E[n]\backslash\langle P\rangle$ を選んで以下の計算を行う．

$$\begin{aligned} xyP &= ECDH(xP, yP, P) \\ &= ECcDH(ECcDH(xP, P, Q), Q, yP) \\ &= ECIcDH(ECIcDH(xP, P, Q), Q, yP) \end{aligned} \tag{27}$$

□

定理 3 $ECDH \Rightarrow ECcDH$

証明 (概略)
既知の xP, yP, 及び P より $ECcDH$ を用いて以下のように $ECDH$ を得る．まず $Q \in E[n]\backslash\langle P\rangle$ を選んで以下の計算を行う．

$$\begin{aligned} xyP &= ECDH(xP, yP, P) \\ &= ECcDH(ECcDH(xP, P, Q), Q, yP) \end{aligned} \tag{28}$$

□

定理 4 $ECDH \Rightarrow ECIcDH$

証明 (概略)
既知の xP, yP, 及び P より $ECIcDH$ を用いて以下のように $ECDH$ を得る．まず $Q \in E[n]\backslash\langle P\rangle$ を選んで以下の計算を行う．

$$\begin{aligned} xyP &= ECDH(xP, yP, P) \\ &= ECIcDH(ECIcDH(xP, P, Q), Q, yP) \end{aligned} \tag{29}$$

□

系 1 $ECIcDH \Rightarrow ECIcDH'$

証明 (概略)
既知の xP, yP, 及び Q より $ECIcDH'$ を用いて以下のように $ECIcDH$ を得る．

$$\begin{aligned} \frac{1}{x}Q &= ECIcDH(xP, P, Q) \\ &= ECIcDH'(e_n(xP, Q), e_n(P, Q), Q) \end{aligned} \tag{30}$$

□

定理 5 $ECDH \Rightarrow ECDS_1,\ ECDH \Rightarrow ECDS_2$

証明 (概略)
既知の xP, 及び P より $ECDS_1$ または $ECDS_2$ を用いて以下のように $ECIDH$ を得る．

$$\begin{aligned} \frac{1}{x}P &= ECIDH(xP, P) \\ &= P - ECDS_1(xP - P, 1, P) \\ &= P - ECDS_2(xP - P, P, P) \end{aligned} \tag{31}$$

ところが，$ECDH \Rightarrow ECIDH$ であるので，$ECDH \Rightarrow ECDS_1$，$ECDH \Rightarrow ECDS_2$ である．□

定理 6 $ECDH \Rightarrow ECcDS_1$

証明 (概略)
既知の $\frac{1}{x}P, P$， 及び Q より $ECcDS_1$ を用いて以下のように $ECDS_1$ を得る．

$$\begin{aligned} ECDS_1(\frac{1}{x}P, y, P) &= ECcDS_1(ECcDS_1(\frac{1}{x}P, \frac{y}{2}, P, Q), \frac{y}{2}, Q, P) \\ &= \frac{1}{x+y}P \end{aligned} \tag{32}$$

ところが， $ECDH \Rightarrow ECDS_1$ であるので，$ECDH \Rightarrow ECcDS_1$ である． □

定理 7 $ECDH \Rightarrow ECD$

証明 (概略)
既知の $xP, P,$ 及び Q より ECD を用いて以下のように $ECcDH$ を得る．まず $y \in \mathbb{Z}/n\mathbb{Z}$ を選んで以下の計算を行う．

$$\begin{aligned} xQ &= ECcDH(xP, P, Q) \\ &= ECD(xP + yQ, P - Q, Q) - yQ \end{aligned} \tag{33}$$

ところが，$ECDH \Rightarrow ECcDH$ であるので，$ECDH \Rightarrow ECD$ である¶． □

4.2 攻撃者

本方式に対する攻撃者を以下のように分類する．

$\mathcal{ADV}$： グループ $\mathcal{G}$ の内外を問わず，本方式に対する攻撃を試みる者で，攻撃のために，公開鍵と既存の正当な署名を使用可能であると仮定する．

$\mathcal{ADV}(\cdot)$： 本方式に対する攻撃を試みるグループ $\mathcal{G}$ のメンバ，若しくは管理者とする．例えば，$\mathcal{ADV}(\mathcal{M}_i)$ は攻撃を試みるメンバである．$\mathcal{ADV}(\cdot)$ は攻撃のために，公開鍵と攻撃者自身の秘密鍵を使用可能であると仮定する．

$\mathcal{CAD}(\cdot, \cdot, \ldots)$： グループ $\mathcal{G}$ のメンバ，若しくは管理者が結託して本方式に対する攻撃を試みるものとする．例えば，$\mathcal{CAD}(\mathcal{M}_i, \mathcal{RA})$ は公開鍵と，$\mathcal{M}_i$， 及び $\mathcal{RA}$ の秘密鍵を使用可能であると仮定する．

$\mathcal{CEA}(\cdot)$： グループ $\mathcal{G}$ のメンバ，及び管理者が一人を除いて全員で結託して結託して本方式に対する攻撃を試みるものとする．例えば， $\mathcal{CEA}(\mathcal{RA})$ は $\mathcal{CAD}$(全メンバ, 全メンバ承認管理者)である．

なお，攻撃者には本方式に対する攻撃のために多項式時間の計算が許されると仮定する．以下では議論を簡潔に行うために，グループ $\mathcal{G}$ には 2 人のメンバ $\mathcal{M}_1, \mathcal{M}_2$ が属していると仮定する．更に，メンバ承認管理者が $\mathcal{AA}_1, \mathcal{AA}_2$ の 2 人であると仮定する．$\mathcal{AA}_2$ が $\mathcal{IA}$ を兼任することに注意したい．

¶Yoshida らにより， 別の証明が与えられている [13]．また， $\phi(P) = qP$， または $\phi(Q) = qQ$ であるような特殊な場合には $ECD(xP + yQ, P, Q)$ が計算可能である [10].

4.3 仮定

本論文では以下の仮定の下で議論を行う．

仮定 1 $\mu_n \subseteq \mathbb{F}_{q^\kappa}^*$ 上での DL 及び，$E[n]$ 上での $ECDL$ を多項式時間で計算することは困難である．

仮定 2 $E[n]$ 上での $ECDH$ を多項式時間で計算することは困難である．

仮定 3 $\mu_n \subseteq \mathbb{F}_{q^\kappa}^*$ 上での DDH を多項式時間で計算することは困難である．

本方式では上述の仮定を満足させるように 3.2.1 節で示した条件 1，及び条件 2 を満たすパラメータを選ぶ必要がある．なお，以下では DL あるいは $ECDL$ を解くことによる単純な攻撃[‖]に関する考察を割愛する．

4.4 署名変造

本論文では，グループ $\mathcal{G}$ のメンバによって正当に生成された署名を用いて，他のメッセージに対する署名を偽造する攻撃を “署名変造” と呼ぶ．署名変造は主に $\mathcal{ADV}$ が試みる攻撃である．$\mathcal{ADV}$ は既存のメッセージ m に対する署名

$$\left.\begin{aligned} R &= k(h(d_i{}^k)P_i + L_i) = (X, Y) \\ S &= \tfrac{X}{k}K_i + \tfrac{h(m)}{k}Q_i \end{aligned}\right\} \tag{34}$$

を用いて，メッセージ m' に対する以下のような署名

$$\left.\begin{aligned} \tilde{R} &= k(h(d_i^k)P_i + L_i) = (\tilde{X}, \tilde{Y}) \\ \tilde{S} &= \tfrac{\tilde{X}}{k}K_i + \tfrac{h(m')}{k}Q_i \end{aligned}\right\} \tag{35}$$

の偽造を試みると仮定する．$\tilde{R} = R$ であるので，$\tilde{S}$ を偽造できれば攻撃が成功することに注意したい．ここで，既知の S より $\tilde{S}$ を多項式時間で生成するアルゴリズムが存在すると仮定する．このとき，$ECD(S, K_i, Q_i)$ を以下のように計算可能である．

$$ECD(S, K_i, Q_i) = \frac{h(m)}{h(m) - h(m')}\left(S - \tilde{S}\right) \tag{36}$$

仮定 2 の下では，ECD は困難であったので，$\tilde{S}$ を計算する多項式時間アルゴリズムは存在しない．

4.5 署名鍛造

本論文では，検証者 $\mathcal{V}$ による検証においては受理されるにも関わらず，署名者特定管理者 $\mathcal{IA}$ によって署名者を特定できないような署名を偽造する攻撃を “署名鍛造” と呼ぶ．以下では $\mathcal{ADV}(\mathcal{M}_i)$ が $\alpha_P, \beta_P, \alpha_Q, \beta_Q, \gamma_K, \gamma_Q, \delta_P,\ \delta_Q$ を自由に選択できるものと仮定する．$\mathcal{ADV}(\mathcal{M}_i)$ は

$$\left.\begin{aligned} \tilde{R} &= \alpha_P P + \beta_P h(\tilde{v})P_i + \alpha_Q Q + \beta_Q L_i \\ &= \left(\alpha_P + \tfrac{\beta_P h(\tilde{v})}{w_i}\right)P + \left(\alpha_Q + \tfrac{\beta_Q}{w_i a_i}\right)Q = (\tilde{X}, \tilde{Y}) \\ \tilde{S} &= \gamma_K \tilde{X} K_i + \gamma_Q h(m) Q_i + \delta_P P + \delta_Q Q \\ &= (w_i a_i \gamma_K \tilde{X} + \delta_P)P + (w_i b \gamma_K \tilde{X} + w_i \gamma_Q h(m) + \delta_Q)Q \end{aligned}\right\} \tag{37}$$

[‖]例えば公開鍵 $g, y = g^b$ より秘密鍵 b を求める攻撃など．

で表される偽造署名 $\left(\tilde{R},\tilde{S}\right)$ の偽造を試みると仮定する．ただし， $\tilde{v}=e_n(P_r,\tilde{R})$ であり，$\tilde{X},\tilde{Y}$ はそれぞれ点 $\tilde{R}$ の x 座標及び y 座標とする．$\left(\tilde{R},\tilde{S}\right)$ は 3.4 節で述べた署名検証に合格するが，$\mathcal{IA}$ によってさえ署名者特定が行えないような署名である**．$\left(\tilde{R},\tilde{S}\right)$ を式 (21) の (R,S) に代入すると指数部より

$$\begin{aligned}
&\left(\tilde{X}w_i\gamma_K\alpha_P+(\gamma_K\beta_P-1)h(\tilde{v})\tilde{X}\right)b+\alpha_P(w_i\gamma_Q h(m)+\delta_Q)\\
-\quad&\alpha_Q(w_ia_i\gamma_K\tilde{X}+\delta_P)+\beta_P h(\tilde{v})\left(\gamma_Q h(m)+\frac{\delta_Q}{w_i}\right)-\beta_Q\left(\gamma_K\tilde{X}+\frac{\delta_P}{w_ia_i}\right)\\
-\quad&h(m)h(\tilde{v})+\tilde{X}=\epsilon_i b+\epsilon_i'=0 \qquad (38)
\end{aligned}$$

を得る．ただし，

$$\begin{aligned}
\epsilon_i &= (w_i\alpha_P+\beta_P h(\tilde{v}))\,\gamma_K\tilde{X}-h(\tilde{v})\tilde{X} \qquad (39)\\
\epsilon_i' &= \alpha_P(w_i\gamma_Q h(m)+\delta_Q)-\alpha_Q(w_ia_i\gamma_K\tilde{X}+\delta_P)\\
&+\quad \beta_P h(\tilde{v})\left(\gamma_Q h(m)+\frac{\delta_Q}{w_i}\right)-\beta_Q\left(\gamma_K\tilde{X}+\frac{\delta_P}{w_ia_i}\right)-h(m)h(\tilde{v})+\tilde{X} \qquad (40)
\end{aligned}$$

とする．ここで，以下の補題を与える．

補題 1 仮定 *2* の下では

$$\epsilon_i=\epsilon_i'=0 \qquad (41)$$

である．

証明 (概略)
$\epsilon_i\neq 0$ かつ $\epsilon_i'\neq 0$ を仮定する．更に

$$\begin{aligned}
\zeta_i &= -\frac{\beta_Q\delta_P}{w_i} \qquad (42)\\
\zeta_i' &= \beta_P h(\tilde{v})\,(w_i\gamma_Q h(m)+\delta_Q)-\beta_Q\gamma_K\tilde{X}-\alpha_Q\delta_P\\
&+\quad \alpha_P\,(w_i\gamma_Q h(m)+\delta_Q)-h(m)h(\tilde{v})+\tilde{X} \qquad (43)\\
\zeta_i'' &= -\alpha_Q w_i\gamma_K\tilde{X} \qquad (44)
\end{aligned}$$

とする．ただし，$\epsilon_i'=\zeta_i\frac{1}{a_i}+\zeta_i'+\zeta''a_i$ である．このとき，式 (38) は

$$\epsilon_i b+\zeta_i\frac{1}{a_i}+\zeta_i'+\zeta''a_i=0 \qquad (45)$$

となる．このとき，$\mathcal{CEA}(\mathcal{RA})$ は $ECIDH(\frac{1}{x}Q,Q)$ を

$$ECIDH(\frac{1}{x}Q,Q)=xQ=-\frac{\zeta_i}{\zeta_i''}\left(\frac{1}{x}Q\right)-\frac{\epsilon_i b+\zeta_i'}{Q} \qquad (46)$$

のように計算可能である．仮定 2 の下では， $ECIDH$ は困難であったので，$\epsilon_i=\epsilon_i'=0$ である．□
補題 1 より

$$\tilde{X}w_i\gamma_K\alpha_P+(\gamma_K\beta_P-1)h(\tilde{v})\tilde{X}=0 \qquad (47)$$

**$\alpha_P=\alpha_Q=0$ かつ $\beta_P=\beta_Q=k$ であれば，$\left(\tilde{R},\tilde{S}\right)$ は実際には正規の手続きで生成された署名であることに注意したい．

$$\alpha_P\,(w_i\gamma_Q h(m)+\delta_Q)-\alpha_Q(w_i a_i\gamma_K\tilde{X}+\delta_P)+\left(\gamma_Q h(m)+\frac{\delta_Q}{w_i}\right)h(\tilde{v})\beta_P$$
$$-\quad\left(\gamma_K\tilde{X}+\frac{\delta_P}{w_i a_i}\right)\beta_Q = h(m)h(\tilde{v})-\tilde{X} \tag{48}$$

を得る．

$\gamma_K \neq 0$ は明らかであるので，式 (47) は

$$\alpha_P = \frac{(1-\gamma_K\beta_P)h(\tilde{v})}{\gamma_K w_i} \tag{49}$$

となる．式 (49) と $P_i = \frac{1}{w_i}P$ を式 (37) に代入すると，

$$\begin{aligned}\tilde{R} &= (\frac{1}{\gamma_K}-\beta_P)h(\tilde{v})\frac{P}{w_i}+\beta_P h(\tilde{v})P_i+\alpha_Q Q+\beta_Q L_i \\ &= \frac{h(\tilde{v})}{\gamma_K}P_i+\alpha_Q Q+\beta_Q L_i\end{aligned} \tag{50}$$

を得る．式 (37) と 式 (50) を比較して，$\alpha_P = 0$， $\beta_P\gamma_K = 1$ を得る．ここで，以下の補題を与える．

補題 2 仮定 *2* の下では

$$w_i a_i\gamma_K\tilde{X}+\delta_P \neq 0 \tag{51}$$

である．

証明 (概略)
$w_i a_i\gamma_K\tilde{X}+\delta_P = 0$ を仮定する．このとき，

$$\tilde{S} = \gamma_K\tilde{X}w_i bQ + h(m)\gamma_Q Q_i + \delta_Q Q \tag{52}$$

である．式 (52)を満たす$\tilde{S}$が生成可能であるならば，$\tilde{S}$ を用いて，以下のように ECD を計算可能である．

$$ECD(\gamma_K\tilde{X}K_i, P, Q) = \tilde{S}-h(m)\gamma_Q Q_i+\delta_Q Q \tag{53}$$

仮定 2 の下では， ECD は困難であったので，式 (52)を満たす$\tilde{S}$ の生成は困難である．したがって，$w_i a_i\gamma_K\tilde{X}+\delta_P \neq 0$ である． □

式 (48) と補題 2 より

$$\alpha_Q = \frac{\left(\gamma_Q h(m)+\frac{\delta_Q}{w_i}\right)h(\tilde{v})\beta_P - h(m)h(\tilde{v})+\tilde{X}}{w_i a_i\gamma_K\tilde{X}+\delta_P}-\frac{\beta_Q}{w_i a_i} \tag{54}$$

を得る．式 (54) と $L_i = \frac{1}{w_i a_i}Q$ を 式 (37) に代入すると

$$\tilde{R} = \beta_P h(\tilde{v})P_i + \frac{\left\{\left(\gamma_Q h(m)+\frac{\delta_Q}{w_i}\right)\beta_P - h(m)\right\}h(\tilde{v})+\tilde{X}}{w_i a_i\gamma_K\tilde{X}+\delta_P}Q \tag{55}$$

を得る． $\tilde{v}$ は $\tilde{v} = e_n(P_r, \tilde{R}) = e_n(P_r, \alpha_Q Q+\beta_Q L_i)$ で与えられるので，α_Q と β_Q を決定する前に $\tilde{v}$ を決定することは困難である．したがって， $\left(\gamma_Q h(m)+\frac{\delta_Q}{w_i}\right)\beta_P - h(m) \neq 0$ であるならば，$\mathcal{ADV}(\mathcal{M}_i)$ は $\left(\tilde{R}, \tilde{S}\right)$ を無視できるほどの確率でのみ計算可能であると考えられる．以上より

$$\left(\gamma_Q h(m)+\frac{\delta_Q}{w_i}\right)\beta_P - h(m) = 0 \tag{56}$$

との仮定を置くことは妥当である．式 (56) より

$$\delta_Q = w_i h(m) \left(\frac{1}{\beta_P} - \gamma_K \right) \tag{57}$$

を得る．(57), $Q_i = w_i Q$，及び $\gamma_K \beta_P = 1$ を式 (37) に代入すると

$$\begin{aligned} \tilde{S} &= \gamma_K \tilde{X} K_i + \gamma_Q h(m) Q_i + \delta_P P + w_i h(m) \left(\frac{1}{\beta_P} - \gamma_K \right) Q \\ &= \gamma_K \tilde{X} K_i + \gamma_K h(m) Q_i + \delta_P P. \end{aligned} \tag{58}$$

を得る．式 (58) と 式 (37) を比較すると $\gamma_Q = \gamma_K$ と $\delta_Q = 0$ を得る．$\gamma_Q = \gamma_K$ と $\delta_Q = 0$ を 式 (37) に代入すると

$$\tilde{R} = \beta_P h(\tilde{v}) P_i + \frac{1}{w_i a_i \gamma_K + \frac{\delta_P}{\tilde{X}}} Q \tag{59}$$

を得る．$\tilde{X}$ は点 $\tilde{R}$ の x 座標であったので，$\tilde{R}$ を決定する前に $\tilde{X}$ を決定することは困難である．したがって，$\delta_P \neq 0$ であるならば，$\mathcal{ADV}(\mathcal{M}_i)$ は $\left(\tilde{R}, \tilde{S}\right)$ を無視できるほどの確率でのみ計算可能であると考えられる．以上より $\delta_P = 0$ との仮定を置くことは妥当である．$\delta_P = 0$, $L_i = \frac{1}{w_i a_i} Q$, 及び $\gamma_K \beta_P = 1$ を 式 (37) に代入すると

$$\tilde{R} = \beta_P h(\tilde{v}) P_i + \frac{1}{w_i a_i \gamma_K} Q = \beta_P h(\tilde{v}) P_i + \beta_P L_i \tag{60}$$

を得る．式 (60) と 式 (37) を比較して，$\alpha_Q = 0$ を得る．

ここで，$\beta_P = k$ として以上の議論を整理すると

$$\left.\begin{array}{l} \beta_P = \beta_Q = k \\ \gamma_K = \gamma_Q = \frac{1}{k} \\ \alpha_P = \alpha_Q = \delta_P = \delta_Q = 0 \end{array}\right\} \tag{61}$$

となり，更に

$$\left.\begin{array}{l} \tilde{R} = k \left(h(\tilde{v}) P_i + L_i \right) \\ \tilde{S} = \frac{1}{k} \left(\tilde{X} K_i + \gamma_Q h(m) Q_i \right) \end{array}\right\} \tag{62}$$

を得る．したがって，$\left(\tilde{R}, \tilde{S}\right)$ は正規の署名に相違ない．以上より，鍵を偽造することなく署名鍛造を行うことは困難である．

4.6 署名鍵偽造

前節の議論により署名鍵を偽造することなく署名の偽造を行うことが困難であることが明らかになった．そこで，本節では署名鍵を偽造する攻撃に関する考察を行う．以下では署名鍵偽造を “署名鍵特定” と “署名鍵鍛造” に分類して議論を行う．前者は特定のメンバの署名鍵を偽造する攻撃であり，後者は新たな署名鍵を偽造する攻撃，即ち正規の手続きに依らずに，メンバ追加を行う攻撃である．

4.6.1 署名鍵特定

$\mathcal{ADV}(\mathcal{IA})$ は $(K_{i,\tau}, L_i, P_i, Q_i)$ を使用可能であるので，$K_i = \frac{1}{r_i'} K_{i,\tau}$ を計算できれば $\mathcal{M}_i$ の署名鍵特定に成功する．既知の $K_{i,\tau}$, $g_i = g^{r_i'}$， 及び g より K_i を多項式時間で計算するアルゴリズム

$$FKM(K_{i,\tau}, g_i, g) = K_i \tag{63}$$

が存在すると仮定する．このとき，$ECDH(aP, bP, P)$ を以下のように計算可能である．

$$\begin{aligned} & ECDH(aP, bP, P) = abP \\ = \quad & FKM(aP, e_n(P, Q), e_n(bP, Q)) \end{aligned} \tag{64}$$

仮定 2 の下では，$ECDH$ は困難であったので，多項式時間アルゴリズム FKM は存在しない．

4.6.2 署名鍵鍛造

鍛造攻撃 I $\mathcal{CEA}(\mathcal{RA})$ は K_i, z, s_i, 更には $K_{Ri} = (a_{Ri}P + b_RQ)$, $L_{Ri} = \frac{1}{a_{Ri}}Q$, を $i = 1, 2$ について使用可能である．$\mathcal{CEA}(\mathcal{RA})$ にとって，b_R，及び $a_{Ri} = r_is_i$ が未知であることに注意したい．$\mathcal{CEA}(\mathcal{RA})$ が以下のような $\tilde{K}$，及び $\tilde{L}$ を計算できれば，新たな署名鍵 $\left(z\tilde{K}, \frac{1}{z}\tilde{L}, P, Q\right)$ の偽造に成功する．

$$\tilde{K} = (\tilde{a}P + b_RQ) \tag{65}$$

$$\tilde{L} = \frac{1}{\tilde{a}}Q \tag{66}$$

ただし，$\tilde{a} \neq a_{Ri}$ である．以下では攻撃について 2 つ場合に分けて議論を行う．

Attack 1 : $\tilde{a} = a_{Ri} + a'$ とする．まず，$\mathcal{CEA}(\mathcal{RA})$ は a' を選んで

$$\tilde{K} = (a_{Ri} + a')P + b_RQ = K_{Ri} + a'P \tag{67}$$

を計算する．次に $\mathcal{CEA}(\mathcal{RA})$

$$\tilde{L} = \frac{1}{a_{Ri} + a'}Q \tag{68}$$

の計算を試みる．$\mathcal{CEA}(\mathcal{RA})$ が $\tilde{L}$ を計算できれば，署名鍵 $\left(z\tilde{K}, \frac{1}{z}\tilde{L}, P, Q\right)$ の偽造に成功する．

以下では $\tilde{L}$ を計算する 2 つの多項式時間アルゴリズムについて議論する．まず，既知の a', Q, 及び L_{Ri} より $\tilde{L}$ を計算する多項式時間アルゴリズム

$$CER_1(L_{Ri}, a', Q) = \tilde{L} \tag{69}$$

が存在すると仮定する．このとき，$ECDS_1(\frac{1}{x}P, y, P)$ を以下のように計算可能である．

$$\begin{aligned} \frac{1}{x+y}P \quad &= \quad ECDS_1(\frac{1}{x}P, y, P) \\ &= \quad CER_1(\frac{1}{x}P, y, P) \end{aligned} \tag{70}$$

次に，既知の a', P, Q，及び K_{Ri} より $\tilde{L}$ を計算する多項式時間アルゴリズム

$$CER_1'(K_{Ri}, a', P, Q) = \tilde{L} \tag{71}$$

が存在すると仮定する．このとき，$ECIcDH(xP, P, Q)$ を以下のように計算可能である．

$$\begin{aligned} \frac{1}{x}Q \quad &= \quad ECIcDH(xP, P, Q) \\ &= \quad CER_1'(xP - yP + zQ, y, P, Q) \end{aligned} \tag{72}$$

ただし，y 及び z は $\mathbb{Z}/n\mathbb{Z}$ より選ぶものとする．仮定 2 の下では，$ECDS_1$ 及び $ECIcDH$ は困難であったので，$\tilde{S}$ を計算する多項式時間アルゴリズム CER_1 及び CER_1' は存在しない．

Attack 2 : $\tilde{a} = a_{R1} + a_{R2}$ とする．まず $\mathcal{CEA}(\mathcal{RA})$ は

$$\tilde{K} = \frac{1}{2}\left(a_{r1} + a_{R2}\right) P + b_R Q = \frac{K_{R1} + K_{R2}}{2} \tag{73}$$

を計算する．次に $\mathcal{CEA}(\mathcal{RA})$ は

$$\tilde{L} = \frac{1}{a_{R1} + a_{R2}} Q \tag{74}$$

の計算を試みる．$\mathcal{CEA}(\mathcal{RA})$ が $\tilde{L}$ を計算できれば，署名鍵 $\left(z\tilde{K}, \frac{2}{z}\tilde{L}, P, Q\right)$ の偽造に成功する．

以下では $\tilde{L}$ を計算する 2 つの多項式時間アルゴリズムについて議論する．まず，既知の Q, L_{R1} 及び L_{R2} より $\tilde{L}$ を計算する多項式時間アルゴリズム

$$CER_2(\tilde{L}_1, \tilde{L}_2, Q) = \tilde{L} \tag{75}$$

が存在すると仮定する．このとき，$ECDS_2(\frac{1}{x}P, \frac{1}{y}P, P)$ を以下のように計算可能である．

$$\begin{aligned} \frac{1}{x+y}P &= ECDS_2(\frac{1}{x}P, \frac{1}{y}P, P) \\ &= CER_2(\frac{1}{x}P, \frac{1}{y}P, P) \end{aligned} \tag{76}$$

次に，既知の P, Q, K_{R1}，及び K_{R2} より $\tilde{L}$ を計算する多項式時間アルゴリズム

$$CER_2'(K_{R1}, K_{R2}, P, Q) = \tilde{L} \tag{77}$$

が存在すると仮定する．このとき，$ECIcDH(xP, P, Q)$ を以下のように計算可能である．

$$\begin{aligned} \frac{1}{x}Q &= ECIcDH(xP, P, Q) \\ &= 2CER_2'(xP + yQ, xP + zQ, P, Q) \end{aligned} \tag{78}$$

ただし，y 及び z は $\mathbb{Z}/n\mathbb{Z}$ より選ぶものとする．仮定 2 の下では，$ECDS_2$ 及び $ECIcDH$ は困難であったので，$\tilde{S}$ を計算する多項式時間アルゴリズム CER_2 及び CER_2' は存在しない．

鍛造攻撃 II $\mathcal{CEA}(\mathcal{IA})$ は $K_i,\ L_i,\ K_{i,1},\ L_{i,1},\ P_i,\ Q_i$，$z_1,\ w_{i,1},\ a_{Ri}$，及び b_R を使用可能である．更に $\mathcal{CEA}(\mathcal{IA})$ は

$$\tilde{K}' = (\tilde{a}P + z_1 b_R Q) \tag{79}$$

$$\tilde{L}' = \frac{1}{\tilde{a}}Q \tag{80}$$

を計算可能である．ただし $i = 1, 2$ それぞれについて $\tilde{a} \neq z_1 a_{Ri}$ である．$\mathcal{CEA}(\mathcal{IA})$ にとって，z_2 及び $w_{i,2}$ は未知であることに注意したい．$\mathcal{CEA}(\mathcal{IA})$ が $\tilde{K} = z_2\tilde{K}'$ 及び $\tilde{L} = \frac{1}{z_2}\tilde{L}'$ が計算できれば，署名鍵 $\left(\tilde{K}, \tilde{L}, P, Q\right)$ の偽造に成功する．以下では $\tilde{K}$ を計算する 2 つの多項式時間アルゴリズムについて議論する．まず，既知の $\tilde{K}', y_1$ 及び $y_2 = y_1^{z_2}$ より $\tilde{K}$ を計算する多項式時間アルゴリズム

$$CEI(\tilde{K}', y_1, y_2) = \tilde{K} \tag{81}$$

b が存在すると仮定する．$\tilde{L} = CEI(\tilde{L}', y_2, y_1)$ であることに注意したい．このとき，$ECDH(aP, bP, P)$ を以下のように計算可能である．

$$\begin{aligned} abP &= ECDH(aP, bP, P) \\ &= CEI(aP, e_n(P, Q), e_n(bP, Q)) \end{aligned} \tag{82}$$

次に，既知の $\tilde{K}'$, $K_{i,1}$ 及び $K_i = w_{i,2} z_2 K_{i,1}$ より $w_{i,2}\tilde{K}$ を計算する多項式時間アルゴリズム

$$CEI'(\tilde{K}', K_i, K_{i,1}) = w_{i,2}\tilde{K} \tag{83}$$

が存在すると仮定する．$\frac{1}{w_{i,2}}\tilde{L} = CEI'(\tilde{L}', K_{i,1}, K_i)$，$CEI'(\tilde{K}', L_{i,1}, K_i) = w_{i,2}\tilde{K}$ であることに注意したい．CEI' が存在すれば，署名鍵 $\left(w_i\tilde{K}, \frac{1}{w_i}\tilde{L}, P_i, Q_i\right)$ の偽造に成功する．ただし，$w_i = w_{i,1}w_{i,2}$ である．このとき，$ECcDH(xP, P, Q)$ を以下のように計算可能である．

$$\begin{aligned} xQ &= ECcDH(xP, P, Q) \\ &= CEI'(Q, xP, P) \end{aligned} \tag{84}$$

仮定 2 の下では，$ECDH$ 及び $ECcDH$ は困難であったので，$\tilde{S}$ を計算する多項式時間アルゴリズム CEI 及び CEI' は存在しない．

$\mathcal{CEA}(\mathcal{IA})$ による場合と同様に，$\mathcal{CEA}(\mathcal{AA}_1)$ による署名鍵鍛造は困難である．以上より，署名鍵鍛造は管理者全員が結託しない限りは困難であると示された．

4.7 偽証攻撃

本論文では，あるメンバ $\mathcal{M}_i$ が署名 $\{R, S\}$ を生成していないにも関わらず，$\mathcal{M}_i$ が $\{R, S\}$ の署名者であると偽証する攻撃を “偽証攻撃” と呼ぶ．ここでは，$\mathcal{CEA}(\mathcal{M}_2)$ が $\mathcal{M}_2$ が (R, S) の署名者であるとの偽証攻撃を以下のように試みると仮定する．
まず，$\mathcal{CEA}(\mathcal{M}_2)$ は $\mathcal{M}_1$ の署名鍵を用いてメッセージ m に対する署名 (R, S) を生成する．次に，$\mathcal{CEA}(\mathcal{M}_2)$ は

$$e_n(\tilde{R}_c, R) = 1, \tag{85}$$

$$e_n(\tilde{R}_c, \tilde{S}_c) = y_2^{h(v)} \cdot g_2^{-1} \cdot e_n(P_r, Q)^{h(v)} \tag{86}$$

を満たすような $\left(\tilde{R}_c, \tilde{S}_c\right)$ の生成を試みる．もし $\left(\tilde{R}_c, \tilde{S}_c\right)$ が 式 (85) 及び式 (86) を満たすならば，$\left(\tilde{R}_c, \tilde{S}_c\right)$ を用いて 3.5 節で示した検証を行うと，(R, S) の署名者が $\mathcal{M}_1$ であると偽証することに成功する．$\tilde{R}_c \in \langle R \rangle$ 場合に限り，$\tilde{R}_c$ は式 (85) を満たすので，

$$\tilde{R}_c = R_c = \frac{k'}{w_1}\left(h(v)P + \frac{1}{a_1}Q\right) \in \langle R \rangle \tag{87}$$

と仮定する．ただし，k' は $\mathbb{Z}/n\mathbb{Z}$ より選ぶものとする．$\tilde{R}_c$ は

$$e_n(\tilde{R}_c, S_c) = y_1^{h(v)} \cdot g_1^{-1} \cdot e_n(P_r, Q)^{h(v)} \tag{88}$$

を満たす．ただし，S_c は式 (23) によって計算されるものである．式 (86) 及び 式 (88) より

$$\begin{aligned} e_n(\tilde{R}_c, T) &= \left(y^{bh(v)}\right)^{r_1' - r_2'} \left(g^{-1}\right)^{r_1' - r_2'} \\ &= \left(g^{bh(v)-1}\right)^{r_1' - r_2'} \end{aligned} \tag{89}$$

を得る．ただし，$T = S_c - \tilde{S}_c$ とする．ここで，

$$T = \frac{w_1(r_1' - r_2')}{k'}(\alpha P + \beta Q) \tag{90}$$

と仮定する．式 (90) を 式 (89) に代入すると，

$$\left(g^{h(v)\beta-\frac{\alpha}{a_1}}\right)^{r_1'-r_2'} = \left(g^{bh(v)-1}\right)^{r_1'-r_2'} \tag{91}$$

を得る．式 (91) の指数部より

$$h(v)\beta - \frac{\alpha}{a_1} = bh(v) - 1 \tag{92}$$

を得る．$\mathcal{CEA}(\mathcal{M}_2)$ は $b, h(v)$， 及び a_1 を使用できるので，$\mathcal{CEA}(\mathcal{M}_2)$ は式 (92) を満たすような (α, β) を計算可能である．例えば，$(\alpha, \beta) = (a_1, b)$ を式 (90) に代入すると

$$T = \frac{w_1(r_1' - r_2')}{k'}(a_1 P + bQ) = \frac{r_1' - r_2'}{k'} K_1 \tag{93}$$

を得る．既知の $g_2 = g^{r_2'}$, g， 及び K_1 より $\tilde{S}_c$ を計算する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，$ECIcDH'(g_2, g, K_1)$ を以下のように計算可能である．

$$\begin{aligned} r_2' K_1 &= ECIcDH'(g_2', g, K_1) \\ &= k'\left(\tilde{S}_c - S_c\right) + r_1' K_1 \end{aligned} \tag{94}$$

仮定 2 の下では， $ECIcDH$ は困難であったので，$\tilde{S}_c$ を計算する多項式時間アルゴリズムは存在しない．

4.8 署名関連づけ

本論文では， 署名者特定管理者 $\mathcal{IA}$ 以外が 2 つの異なる署名に関して，それらの署名者が同一のメンバであったか否かの判定を試みる攻撃を “署名関連づけ” と呼ぶ．$\mathcal{ADV}$ が (R, S) と (R', S') の関連づけを試みるとする．ここでは，

$$\left.\begin{array}{l} R = k(h(v)P_i + L_i) = (X, Y) \\ S = \frac{X}{k}K_i + \frac{h(m)}{k}Q_i \\ R' = k'(v')P_j + L_j) = (X', Y') \\ S' = \frac{X'}{k'}K_j + \frac{h(m')}{k'}Q_j \end{array}\right\} \tag{95}$$

と仮定する．ただし， $v = e_n(P_r, R)$ かつ $v' = e_n(P_r, R')$ である．$\mathcal{ADV}$ は以下のような単純な攻撃で署名関連づけを行うことは困難である．
まず， $\mathcal{ADV}$ は $l_R = e_n(R, R'), l_S = e_n(S, S')$, 及び $l_T = y^{\left(Xh(m')\right)\cdot\left(h(v)-h(v')\right)}$ を計算する．次に， $\mathcal{ADV}$ は $d_l = DDH(l_R, l_S, l_T)$ を計算する．(R, S) と (R', S') が同一のメンバによって署名されたならば， $d_l = 1$ である．さもなくば， $d_l = 0$ である．ところが，仮定 3 の下では， DDH は困難であったので，$\mathcal{ADV}$ にとってこのような署名関連づけは困難である．

4.9 署名者暴露

本論文では，署名者特定管理者 $\mathcal{IA}$ 以外が署名者の特定を試みる攻撃を “署名者暴露” と呼ぶ．本方式では一度 $\mathcal{IA}$ がある署名とその署名者の関係を公表するまでは，署名と署名者の関係に関する情報は何ら得られない．$\mathcal{IA}$ がある署名とその署名者の関係を公表した後でさえ，もし署名者暴露が可能であるのならば，署名の関連づけが可能となる．したがって，署名者暴露は困難であると思われるが，詳細の検討については今後の課題としたい．

5 効率評価

本節では本方式の公開鍵サイズ及び署名サイズ，並びに計算量に関する考察を行う．以下では，$\kappa = 1$， $q \simeq 2^{1024}$，$n \simeq 2^{160}$ であると仮定して評価を行う．なお，比較のために [3] の方式を，$l_p = 512,\ k = 160$ の場合について評価する．

5.1 公開鍵サイズ及び署名サイズ

本方式の検証に必要な公開鍵サイズ，及び署名サイズについて見積もりを行う．$E[n]$ の点のサイズについては，x 座標の値と y 座標の符号であるとする．本方式のグループ $\mathcal{G}$ の公開鍵は μ_n の元が 2 個と，$E[n] \subset E(\mathbb{F}_q$ の元が 1 個と，楕円曲線 $E/\mathbb{F}_q$ 及び n であり， そのサイズは約 5 K ビットとなる．署名は $E[n] \subset E(\mathbb{F}_{q^\kappa}^*)$ の元が 2 個であり，そのサイズは 約 2 K ビットとなる．

[3] の方式の公開鍵のサイズは 約 6K ビット，署名文のサイズは約 12K ビットであった．本方式の公開鍵サイズ，署名文のサイズは実用上十分に小さく，効率的と言えよう．

5.2 計算量評価

5.2.1 署名生成

本方式の署名生成の計算量は， μ_n の元のべき乗計算が 1 回，$E[n]$ の元のスカラ倍演算が 4 回必要であった．$E[n]$ の元 P の i 倍点 iP を求めるために必要な計算量を

$$\log_2 i \cdot w \cdot s \cdot A_e \tag{96}$$

とする††．ただし，w を最適な表を用いた符号付き 2 進ウインドウ法を用いてスカラ倍演算を行う際に， i の 1 ビットあたりに必要となる P の加算の回数とし，$w = 1.25$ と仮定する．更に同時スカラ倍演算を行うことによって 2 回のスカラ倍演算が 1 回のスカラ倍演算の $s = 1.2$ 倍分の計算量で行えると仮定し，A_e を $E[n]$ の加算 1 回あたりに必要な乗算の回数とし $A_e = 12$ とする．以上により本方式の署名生成を行うには 11520 回程度の 1024 ビットの数の乗算が必要であると見積もることができる．

5.2.2 署名検証

本方式の検証の計算量は μ_n のべき乗計算が 2 回，ペアリングの計算が 2 回必要であった．ペアリングは P の n 倍点を求める過程で，ペアリング値の計算のために $\mathbb{F}_{q^\kappa}^*$ での乗算を行う．ペアリングに必要な計算量を

$$\log_2 n \cdot w \cdot (A_e + M) \tag{97}$$

とする． M はペアリング値を求めるために必要な $\mathbb{F}_{q^\kappa}^*$ での乗算回数であり，12 回の乗算とする．署名生成の場合と同様の仮定を行うことによって，本方式の検証を行うには 9900 回程度の 1024 ビットの数の乗算が必要であると見積もれる．

一方，[3] の方式の計算量は署名生成，検証それぞれについて，21800 回，18900 回程度の 1024 ビットの数の乗算が必要であった．本方式の署名生成及び検証に必要な計算量は実用上十分に小さく，効率的と言えよう．

††本方式においては，n，すなわち $E[n]$ の位数が公開されているために，$i \bmod n$ 倍点を求めればよいことに注意されたい．同様に μ_n の元の j 乗は $j \bmod n$ 乗の計算を行えばよいことにも注意されたい．

6 むすび

本論文では，ヴェイユペアリングを用いたグループ署名方式を提案した．本方式では，グループのメンバの一部，あるいは全員が参加して，管理者の権限を抑制することが可能であるため，より公正なグループ運営を行うことが期待できる．また，従来の方式に比べ，公開鍵サイズ及び署名サイズ，並びに署名生成及び検証に要する計算量それぞれについて，本方式は優れた方式であることを確かめた．今後の課題としては，安全性の更に詳細な考察を行うことが挙げられる．

参考文献

[1] D.Chaum, E.van Heijst, "Group Signature", Advances in Cryptology – EUROCRYPTO '91, pp.257–265, Springer–Verlag, 1991.

[2] J.Camenisch, M.Michels, "Separability and efficiency for generic group signature schemes", In CRYPTO '99, Springer–Verlag, pp.413–430, 1999.

[3] G.Ateniese, J.Camenisch, M.Joye, G.Tsudik, "A Practical and Provably Secure Coalition–Resistant Group Signature Scheme", In CRYPTO '2000, Springer–Verlag, pp.255–270, 2000.

[4] J.Camenisch, M.Stadler, "Efficient Group Signature Schemes for Large Groups", In CRYPTO '97, Springer–Verlag, pp.410–424, 1997.

[5] J.H.Silverman, "The Arithmetic of Elliptic Curves", Springer–Verlag, 1986.

[6] A.J.Menezes, T.Okamoto, S.A.Vanstone, "Reducing elliptic curve logarithms to a finite field", IEEE Trans. Inform. Theory, 39, pp.1639–1646, 1993.

[7] K.Ohgishi, R.Sakai, M.Kasahara, "Notes on ID–Based Key Sharing Systems over Elliptic Curve", Technical Report of IEICE, ISEC99–57, pp.37–42, 1999.

[8] T.Yamanaka, R.Sakai, M.Kasahara, "Fast computation of pairings over elliptic curves", The 2002 Symposium on Cryptography and Infomation Security SCIS '2002, 10B–5, 2002.

[9] R.Sakai, K.Ohgishi, M.Kasahara, "Cryptosystems Based on Pairing", The 2000 Symposium on Cryptography and Information Security SCIS '2000, C20, 2000.

[10] T.Saito, F.Hoshino, S.Uchiyama, T.Kobayashi, "Candidate One–Way Functions on Non–Supersingular Elliptic Curves", Technical Report of IEICE, ISEC2003–65, pp.95–101, 2003.

[11] D.Boneh, C.Gentry, B.Lynn, H.Shacham, "Aggregate and Veryfiably Encrypted Signatures from Bilinear Maps", Proc. of Crypto 2003, LNCS 2656, Springer–Verlag, pp.416–432, 2003.

[12] S.Mitsunari, R.Sakai, M.Kasahara, "A New Traitor Tracing", IEICE Trans. Vol. E85A, No.2, pp.481–484, 2002.

[13] M.Yoshida, S.Mitsunari, T.Fujiwara, "Vector Decomposition Problem and Trapdoor Inseparable Multiplex Transmission Scheme based on the Problem", The 2003 Symposium on Cryptography and Infomation Security SCIS '2003, 7B–1, 2003.