

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5424974号
(P5424974)

(45) 発行日 平成26年2月26日(2014.2.26)

(24) 登録日 平成25年12月6日(2013.12.6)

(51) Int.Cl.

F I

G09C 1/00 (2006.01)

G09C 1/00 620Z

G09C 1/00 650B

G09C 1/00 650Z

G09C 1/00 660G

請求項の数 15 (全 85 頁)

(21) 出願番号 特願2010-101657 (P2010-101657)
 (22) 出願日 平成22年4月27日(2010.4.27)
 (65) 公開番号 特開2011-232475 (P2011-232475A)
 (43) 公開日 平成23年11月17日(2011.11.17)
 審査請求日 平成24年11月19日(2012.11.19)

(73) 特許権者 000006013
 三菱電機株式会社
 東京都千代田区丸の内二丁目7番3号
 (73) 特許権者 000004226
 日本電信電話株式会社
 東京都千代田区大手町二丁目3番1号
 (74) 代理人 100099461
 弁理士 溝井 章司
 (74) 代理人 100151220
 弁理士 八巻 満隆
 (72) 発明者 高島 克幸
 東京都千代田区丸の内二丁目7番3号 三
 菱電機株式会社内

最終頁に続く

(54) 【発明の名称】 暗号処理システム、鍵生成装置、暗号化装置、復号装置、署名処理システム、署名装置及び検証装置

(57) 【特許請求の範囲】

【請求項1】

鍵生成装置と暗号化装置と復号装置とを備え、 $t = 0, \dots, d$ (d は1以上の整数) の各整数 t についての基底 B_t と基底 B^*_t とを用いて暗号処理を実行する暗号処理システムであり、

前記鍵生成装置は、

$i = 1, \dots, L$ (L は1以上の整数) の各整数 i についての変数 $p(i)$ であって、識別情報 t ($t = 1, \dots, d$ のいずれかの整数) と、属性ベクトル $v^{\rightarrow}_i := (v_i, i')$ ($i' = 1, \dots, n_t$, n_t は1以上の整数) との肯定形の組 $(t, v^{\rightarrow}_i)$ 又は否定形の組 $\neg(t, v^{\rightarrow}_i)$ のいずれかである変数 $p(i)$ と、 L 行 r 列 (r は1以上の整数) の所定の行列 M とを入力する第1情報入力部と、

r 個の要素を有するベクトル $f^{\rightarrow}$ 及び $w^{\rightarrow}$ と、前記第1情報入力部が入力した行列 M とに基づき生成される列ベクトル $s^{\rightarrow T} := (s_1, \dots, s_L)^T := M \cdot f^{\rightarrow T}$ と、値 $s_0 := w^{\rightarrow} \cdot f^{\rightarrow T}$ と、所定の値 θ_i ($i = 1, \dots, L$) とに基づき、要素 k^*_0 と、 $i = 1, \dots, L$ の各整数 i についての要素 k^*_i とを生成する復号鍵生成部であって、

基底 B^*_0 の基底ベクトル $b^*_{0,p}$ (p は所定の値) の係数として前記値 $-s_0$ を設定し、基底ベクトル $b^*_{0,q}$ (q は前記 p とは異なる所定の値) の係数として所定の値 κ を設定して要素 k^*_0 を生成し、

$i = 1, \dots, L$ の各整数 i について、変数 $p(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ で

10

20

ある場合には、その組の識別情報 t が示す基底 B_t^* の基底ベクトル $b_{t,1}^*$ の係数として $s_i + \theta_i v_{i,1}$ を設定するとともに、前記識別情報 t と $i' = 2, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}^*$ の係数として $\theta_i v_{i,i'}$ を設定して要素 k_i^* を生成し、変数 $\rho(i)$ が否定形の組 $\neg(t, v \rightarrow_i)$ である場合には、その組の識別情報 t と $i' = 1, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}^*$ の係数として $s_i v_{i,i'}$ を設定して要素 k_i^* を生成する復号鍵生成部とを備え、

前記暗号化装置は、

$t = 1, \dots, d$ の少なくとも 1 つ以上の整数 t について、識別情報 t と、属性ベクトル $x \rightarrow_t := (x_{t,i'})$ ($i' = 1, \dots, n_t$, n_t は 1 以上の整数) とを有する属性集合 Γ を入力する第 2 情報入力部と、

10

前記第 2 情報入力部が入力した属性集合 Γ に基づき、要素 c_0 と、前記属性集合 Γ に含まれる各識別情報 t についての要素 c_t とを生成する暗号化データ生成部であって、

基底 B_0 の基底ベクトル $b_{0,p}$ (p は前記 p) の係数として乱数値 δ を設定するとともに、基底ベクトル $b_{0,q}$ (q は前記 q) の係数として所定の値 ζ を設定した要素 c_0 を生成し、

前記属性集合 Γ に含まれる各識別情報 t について、基底 B_t の基底ベクトル $b_{t,i'}$ ($i' = 1, \dots, n_t$) の係数として前記乱数値 δ 倍した $x_{t,i'}$ を設定した要素 c_t とを生成する暗号化データ生成部とを備え、

20

前記復号装置は、

前記暗号化データ生成部が生成した要素 c_0 及び要素 c_t と、前記属性集合 Γ とを含む暗号化データ c を取得するデータ取得部と、

前記復号鍵生成部が生成した要素 k_0^* 及び要素 k_i^* と、前記変数 $\rho(i)$ とを含む復号鍵 sk_s を取得する復号鍵取得部と、

前記データ取得部が取得した暗号化データ c に含まれる属性集合 Γ と、前記復号鍵取得部が取得した復号鍵 sk_s に含まれる変数 $\rho(i)$ とに基づき、 $i = 1, \dots, L$ の各整数 i のうち、変数 $\rho(i)$ が肯定形の組 $(t, v \rightarrow_i)$ であり、かつ、その組の $v \rightarrow_i$ と、その組の識別情報 t が示す Γ に含まれる $x \rightarrow_t$ との内積が 0 となる i と、変数 $\rho(i)$ が否定形の組 $\neg(t, v \rightarrow_i)$ であり、かつ、その組の $v \rightarrow_i$ と、その組の識別情報 t が示す Γ に含まれる $x \rightarrow_t$ との内積が 0 とならない i との集合 I を特定するとともに、特定した集合 I に含まれる i について、 $a_i s_i$ を合計した場合に s_0 となる補完係数 a_i を計算する補完係数計算部と、

30

前記暗号化データ c に含まれる要素 c_0 、要素 c_t と、前記復号鍵 sk_s に含まれる要素 k_0^* 、要素 k_i^* とについて、前記補完係数計算部が特定した集合 I と、前記補完係数計算部が計算した補完係数 a_i とに基づき、数 1 に示すペアリング演算を行い値 $K = g_{T^{\zeta K}}$ を計算するペアリング演算部と

を備えることを特徴とする暗号処理システム。

【数 1】

40

$$K := e(c_0, k_0^*) \cdot \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i} \cdot \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i / (\vec{v}_i \cdot \vec{x}_i)}$$

【請求項 2】

前記暗号処理システムは、少なくとも基底ベクトル $b_{0,i}$ ($i = 1, \dots, 5$) を有する基底 B_0 と、少なくとも基底ベクトル $b_{t,i}$ ($i = 1, \dots, n_t, \dots, n_t + u_t, \dots, n_t + u_t + w_t, \dots, n_t + u_t + w_t + z_t$) (u_t, w_t, z_t は 1 以上の整数、) を有する基底 B_t ($t = 1, \dots, d$) と、少なくとも基

50

底ベクトル $b^*_{0,i}$ ($i = 1, \dots, 5$) を有する基底 B^*_0 と、少なくとも基底ベクトル $b^*_{t,i}$ ($i = 1, \dots, n_t, \dots, n_t + u_t, \dots, n_t + u_t + w_t, \dots, n_t + u_t + w_t + z_t$) を有する基底 B^*_t ($t = 1, \dots, d$) とを用いて暗号処理を実行し、

前記鍵生成装置の前記復号鍵生成部は、乱数値 η_0 と所定の値 κ とに基づき数 2 に示す要素 k^*_0 を生成するとともに、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ である場合には、乱数値 $\theta_i, \eta_{i,i'}$ ($i = 1, \dots, L, i' = 1, \dots, w_t$) とに基づき数 3 に示す要素 k^*_i を生成し、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ である場合には、乱数値 $\eta_{i,i'}$ ($i = 1, \dots, L, i' = 1, \dots, w_t$) とに基づき数 4 に示す要素 k^*_i を生成し、

10

前記暗号化装置の前記暗号化データ生成部は、乱数値 δ, ϕ_0 と所定の値 ζ とに基づき数 5 に示す要素 c_0 を生成するとともに、前記乱数値 $\delta, \phi_{t,i}$ ($i = 1, \dots, z_t$) に基づき数 6 に示す要素 c_t とを生成することを特徴とする請求項 1 に記載の暗号処理システム。

【数 2】

$$k^*_0 := (-s_0, 0, \kappa, \eta_0, 0)_{\mathbb{B}^*_0}$$

【数 3】

$$k^*_i := (\overbrace{s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_t}}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,w_t}}^{w_t}, \overbrace{0^{z_t}}^{z_t})_{\mathbb{B}^*_t}$$

20

【数 4】

$$k^*_i := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_t})}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,w_t}}^{w_t}, \overbrace{0^{z_t}}^{z_t})_{\mathbb{B}^*_t}$$

30

【数 5】

$$c_0 := (\delta, 0, \zeta, 0, \phi_0)_{\mathbb{B}_0}$$

【数 6】

$$c_t := (\overbrace{\delta(x_{t,1}, \dots, x_{t,n_t})}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{0^{w_t}}^{w_t}, \overbrace{\phi_{t,1}, \dots, \phi_{t,z_t}}^{z_t})_{\mathbb{B}_t}$$

40

【請求項 3】

$t = 0, \dots, d$ (d は 1 以上の整数) の各整数 t についての基底 B_t と基底 B^*_t とを用いて暗号処理を実行する暗号処理システムにおいて、復号鍵 sk_s を生成する鍵生成装置であり、

$i = 1, \dots, L$ (L は 1 以上の整数) の各整数 i についての変数 $\rho(i)$ であって、識別情報 t ($t = 1, \dots, d$ のいずれかの整数) と、属性ベクトル $v^{\rightarrow}_i := (v_{i,i'})$ ($i' = 1, \dots, n_t, n_t$ は 1 以上の整数) との肯定形の組 $(t, v^{\rightarrow}_i)$ 又は否定形の組 $\neg(t, v^{\rightarrow}_i)$ のいずれかである変数 $\rho(i)$ と、 L 行 r 列 (r は 1 以上の整数) の所定の行列 M とを入力する第 1 情報入力部と、

r 個の要素を有するベクトル $f^{\rightarrow}$ 及び $w^{\rightarrow}$ と、前記第 1 情報入力部が入力した行列 M と

50

に基づき生成される列ベクトル $s^{\rightarrow T} := (s_1, \dots, s_L)^T := M \cdot f^{\rightarrow T}$ と、
 値 $s_0 := w^{\rightarrow} \cdot f^{\rightarrow T}$ と、所定の値 θ_i ($i = 1, \dots, L$) とに基づき、要素 k^*_0
 と、 $i = 1, \dots, L$ の各整数 i についての要素 k^*_i とを生成する復号鍵生成部で
 あって、

基底 B^*_0 の基底ベクトル $b^*_{0,p}$ (p は所定の値) の係数として前記値 $-s_0$ を設
 定し、基底ベクトル $b^*_{0,q}$ (q は前記 p とは異なる所定の値) の係数として所定の値
 κ を設定して要素 k^*_0 を生成し、

$i = 1, \dots, L$ の各整数 i について、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ で
 ある場合には、その組の識別情報 t が示す基底 B^*_t の基底ベクトル $b^*_{t,1}$ の係数と
 して $s_i + \theta_i v_{i,1}$ を設定するとともに、前記識別情報 t と $i' = 2, \dots, n_t$
 の各整数 i' とが示す基底ベクトル $b^*_{t,i'}$ の係数として $\theta_i v_{i,i'}$ を設定して
 要素 k^*_i を生成し、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ である場合には、その
 組の識別情報 t と $i' = 1, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b^*_{t,i'}$
 の係数として $s_i v_{i,i'}$ を設定して要素 k^*_i を生成する復号鍵生成部と、

前記第1情報入力部が入力した変数 $\rho(i)$ と、前記復号鍵生成部が生成した要素 k^*_0
 及び要素 k^*_i とを含むデータを復号鍵 sk_s として配布する鍵配布部と
 を備えることを特徴とする鍵生成装置。

【請求項4】

$t = 0, \dots, d$ (d は1以上の整数) の各整数 t についての基底 B_t と基底 B^*_t
 とを用いて暗号処理を実行する暗号処理システムにおいて、暗号化データ c を生成する暗
 号化装置であり、

$t = 1, \dots, d$ の少なくとも1つ以上の整数 t について、識別情報 t と、属性ベク
 トル $x^{\rightarrow}_t := (x_{t,i'})$ ($i' = 1, \dots, n_t$, n_t は1以上の整数) とを有
 する属性集合 Γ を入力する第2情報入力部と、

前記第2情報入力部が入力した属性集合 Γ に基づき、要素 c_0 と、前記属性集合 Γ に含
 まれる各識別情報 t についての要素 c_t とを生成する暗号化データ生成部であって、

基底 B_0 の基底ベクトル $b_{0,p}$ (p は所定の値) の係数として乱数値 δ を設定すると
 ともに、基底ベクトル $b_{0,q}$ (q は前記 p とは異なる所定の値) の係数として所定の値
 ζ を設定した要素 c_0 を生成し、

前記属性集合 Γ に含まれる各識別情報 t について、基底 B_t の基底ベクトル $b_{t,i'}$
 ($i' = 1, \dots, n_t$) の係数として前記乱数値 δ 倍した $x_{t,i'}$ を設定した要素
 c_t とを生成する暗号化データ生成部と、

前記第2情報入力部が入力した属性集合 Γ と、前記暗号化データ生成部が生成した要素
 c_0 及び要素 c_t とを暗号化データ c として出力するデータ出力部と
 を備えることを特徴とする暗号化装置。

【請求項5】

$t = 0, \dots, d$ (d は1以上の整数) の各整数 t についての基底 B_t と基底 B^*_t
 とを用いて暗号処理を実行する暗号処理システムにおいて、暗号化データ c を復号鍵 sk_s
 で復号する復号装置であり、

$t = 1, \dots, d$ の少なくとも1つ以上の整数 t について、識別情報 t と、属性ベク
 トル $x^{\rightarrow}_t := (x_{t,i'})$ ($i' = 1, \dots, n_t$, n_t は1以上の整数) とを有
 する属性集合 Γ に基づき生成された要素 c_0 と、要素 c_t (前記属性集合 Γ に含まれる各
 整数) とであって、

基底 B_0 の基底ベクトル $b_{0,p}$ (p は所定の値) の係数として乱数値 δ が設定される
 とともに、基底ベクトル $b_{0,q}$ (q は前記 p とは異なる所定の値) の係数として所定の
 値 ζ が設定された要素 c_0 と、

前記属性集合 Γ に含まれる各識別情報 t について、基底 B_t の基底ベクトル $b_{t,i'}$
 ($i' = 1, \dots, n_t$) の係数として前記乱数値 δ 倍した $x_{t,i'}$ が設定された要
 素 c_t と

を前記属性集合 Γ とともに、暗号化データ c として取得するデータ取得部と、

10

20

30

40

50

r 個 (r は 1 以上の整数) の要素を有するベクトル $f \rightarrow$ 及び $w \rightarrow$ と L 行 r 列の所定の行列 M とに基づき生成される列ベクトル $s \rightarrow^T := (s_1, \dots, s_L)^T := M \cdot f \rightarrow^T$ 及び値 $s_0 := w \rightarrow \cdot f \rightarrow^T$ と、所定の値 θ_i ($i = 1, \dots, L$) と、 $i = 1, \dots, L$ (L は 1 以上の整数) の各整数 i についての変数 $\rho(i)$ であって、識別情報 t ($t = 1, \dots, d$ のいずれかの整数) 及び属性ベクトル $v \rightarrow_i := (v_i, i')$ ($i' = 1, \dots, n_t$, n_t は 1 以上の整数) の肯定形の組 $(t, v \rightarrow_i)$ 又は否定形の組 $\neg(t, v \rightarrow_i)$ のいずれかである変数 $\rho(i)$ とに基づき生成された要素 k^*_0 と、要素 k^*_i ($i = 1, \dots, L$ の各整数) とであって、

基底 B^*_0 の基底ベクトル $b^*_{0,p}$ (p は前記 p) の係数として前記値 $-s_0$ が設定され、基底ベクトル $b^*_{0,q}$ (q は前記 q) の係数として所定の値 κ が設定された要素 k^*_0 と、

10

$i = 1, \dots, L$ の各整数 i について、変数 $\rho(i)$ が肯定形の組 $(t, v \rightarrow_i)$ である場合には、その組の識別情報 t が示す基底 B^*_t の基底ベクトル $b^*_{t,1}$ の係数として $s_i + \theta_i v_{i,1}$ が設定されるとともに、前記識別情報 t と $i' = 2, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b^*_{t,i'}$ の係数として $\theta_i v_{i,i'}$ が設定された要素 k^*_i であって、変数 $\rho(i)$ が否定形の組 $\neg(t, v \rightarrow_i)$ である場合には、その組の識別情報 t と $i' = 1, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b^*_{t,i'}$ の係数として $s_i v_{i,i'}$ が設定された要素 k^*_i とを前記変数 $\rho(i)$ とともに、復号鍵 sk_s として取得する復号鍵取得部と、

前記データ取得部が取得した暗号化データ c に含まれる属性集合 Γ と、前記復号鍵取得部が取得した復号鍵 sk_s に含まれる変数 $\rho(i)$ とに基づき、 $i = 1, \dots, L$ の各整数 i のうち、変数 $\rho(i)$ が肯定形の組 $(t, v \rightarrow_i)$ であり、かつ、その組の $v \rightarrow_i$ と、その組の識別情報 t が示す Γ に含まれる $x \rightarrow_t$ との内積が 0 となる i と、変数 $\rho(i)$ が否定形の組 $\neg(t, v \rightarrow_i)$ であり、かつ、その組の $v \rightarrow_i$ と、その組の識別情報 t が示す Γ に含まれる $x \rightarrow_t$ との内積が 0 とならない i との集合 I を特定するとともに、特定した集合 I に含まれる i について、 $\alpha_i s_i$ を合計した場合に s_0 となる補完係数 α_i を計算する補完係数計算部と、

20

前記暗号化データ c に含まれる要素 c_0 、要素 c_t と、前記復号鍵 sk_s に含まれる要素 k^*_0 、要素 k^*_i について、前記補完係数計算部が特定した集合 I と、前記補完係数計算部が計算した補完係数 α_i とに基づき、数 7 に示すペアリング演算を行い値 $K = g_T^{\sum \kappa}$ を計算するペアリング演算部とを備えることを特徴とする復号装置。

30

【数 7】

$$K := e(c_0, k_0^*) \cdot \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i} \cdot \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i / (\vec{v}_i \cdot \vec{x}_i)}$$

【請求項 6】

鍵生成装置と暗号化装置と復号装置とを備え、 $t = 0, \dots, d$ (d は 1 以上の整数) の各整数 t についての基底 B_t と基底 B^*_t とを用いて暗号処理を実行する暗号処理システムであり、

40

前記鍵生成装置は、

$t = 1, \dots, d$ の少なくとも 1 つ以上の整数 t について、識別情報 t と、属性ベクトル $x \rightarrow_t := (x_t, i')$ ($i' = 1, \dots, n_t$, n_t は 1 以上の整数) とを有する属性集合 Γ を入力する第 1 情報入力部と、

前記第 1 情報入力部が入力した属性集合 Γ に基づき、要素 k^*_0 と、前記属性集合 Γ に含まれる各識別情報 t についての要素 k^*_t とを生成する復号鍵生成部であって、

基底 B^*_0 の基底ベクトル $b^*_{0,p}$ (p は所定の値) の係数として乱数値 δ を設定するとともに、基底ベクトル $b^*_{0,q}$ (q は前記 p とは異なる所定の値) の係数として所

50

定の値 κ を設定した要素 k^*_0 を生成し、

前記属性集合 Γ に含まれる各識別情報 t について、基底 B^*_t の基底ベクトル $b^*_{t,i}$ ($i' = 1, \dots, n_t$) の係数として前記乱数値 δ 倍した $x_{t,i}$ を設定した要素 k^*_t とを生成する復号鍵生成部とを備え、

前記暗号化装置は、

$i = 1, \dots, L$ (L は 1 以上の整数) の各整数 i についての変数 $\rho(i)$ であって、識別情報 t ($t = 1, \dots, d$ のいずれかの整数) と、属性ベクトル $v^{\rightarrow}_i := (v_{i,i'})$ ($i' = 1, \dots, n_t$, n_t は 1 以上の整数) との肯定形の組 $(t, v^{\rightarrow}_i)$ 又は否定形の組 $\neg(t, v^{\rightarrow}_i)$ のいずれかである変数 $\rho(i)$ と、 L 行 r 列 (r は 1 以上の整数) の所定の行列 M とを入力する第 2 情報入力部と、

r 個の要素を有するベクトル $f^{\rightarrow}$ 及び $w^{\rightarrow}$ と、前記第 2 情報入力部が入力した行列 M とに基づき生成される列ベクトル $s^{\rightarrow T} := (s_1, \dots, s_L)^T := M \cdot f^{\rightarrow T}$ と、値 $s_0 := w^{\rightarrow} \cdot f^{\rightarrow T}$ と、所定の値 θ_i ($i = 1, \dots, L$) とに基づき、要素 c_0 と、 $i = 1, \dots, L$ の各整数 i についての要素 c_i とを生成する暗号化データ生成部であって、

基底 B_0 の基底ベクトル $b_{0,p}$ (p は前記 p) の係数として前記値 $-s_0$ を設定し、基底ベクトル $b_{0,q}$ (q は前記 q) の係数として所定の値 ζ を設定して要素 c_0 を生成し、

$i = 1, \dots, L$ の各整数 i について、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ である場合には、その組の識別情報 t が示す基底 B_t の基底ベクトル $b_{t,i}$ の係数として $s_i + \theta_i v_{i,i'}$ を設定するとともに、前記識別情報 t と $i' = 2, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}$ の係数として $\theta_i v_{i,i'}$ を設定して要素 c_i を生成し、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ である場合には、その組の識別情報 t と $i' = 1, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}$ の係数として $s_i v_{i,i'}$ を設定して要素 c_i を生成する暗号化データ生成部とを備え、

前記復号装置は、

前記暗号化データ生成部が生成した要素 c_0 及び要素 c_t と、前記変数 $\rho(i)$ とを含む暗号化データ c を取得するデータ取得部と、

前記復号鍵生成部が生成した要素 k^*_0 及び要素 k^*_i と、前記属性集合 Γ とを含む復号鍵 sk_{Γ} を取得する復号鍵取得部と、

前記データ取得部が取得した暗号化データ c に含まれる $i = 1, \dots, L$ の各整数 i についての変数 $\rho(i)$ と、前記復号鍵取得部が取得した復号鍵 sk_{Γ} に含まれる属性集合 Γ とに基づき、 $i = 1, \dots, L$ の各整数 i のうち、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ であり、かつ、その組の $v^{\rightarrow}_i$ と、その組の識別情報 t が示す Γ に含まれる $x^{\rightarrow}_t$ との内積が 0 となる i と、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ であり、かつ、その組の $v^{\rightarrow}_i$ と、その組の識別情報 t が示す Γ に含まれる $x^{\rightarrow}_t$ との内積が 0 とならない i との集合 I を特定するとともに、特定した集合 I に含まれる i について、 $\alpha_i s_i$ を合計した場合に s_0 となる補完係数 α_i を計算する補完係数計算部と、

前記暗号化データ c に含まれる要素 c_0 、要素 c_t と、前記復号鍵 sk_{Γ} に含まれる要素 k^*_0 、要素 k^*_i とについて、前記補完係数計算部が特定した集合 I と、前記補完係数計算部が計算した補完係数 α_i とに基づき、数 8 に示すペアリング演算を行い値 $K = g_T^{\zeta \kappa}$ を計算するペアリング演算部とを備えることを特徴とする暗号処理システム。

【数 8】

$$K := e(c_0, k_0^*) \cdot \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} e(c_i, k_t^*)^{\alpha_i} \cdot \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} e(c_i, k_t^*)^{\alpha_i / (\vec{v}_i \cdot \vec{x}_t)}$$

【請求項 7】

前記暗号処理システムは、少なくとも基底ベクトル $b_{0, i}$ ($i = 1, \dots, 5$) を有する基底 B_0 と、少なくとも基底ベクトル $b_{t, i}$ ($i = 1, \dots, n_t, \dots, n_t + u, \dots, n_t + u_t + w_t, \dots, n_t + u_t + w_t + z_t$) (u_t, w_t, z_t は 1 以上の整数、) を有する基底 B_t ($t = 1, \dots, d$) と、少なくとも基底ベクトル $b_{0, i}^*$ ($i = 1, \dots, 5$) を有する基底 B_0^* と、少なくとも基底ベクトル $b_{t, i}^*$ ($i = 1, \dots, n_t, \dots, n_t + u_t, \dots, n_t + u_t + w_t, \dots, n_t + u_t + w_t + z_t$) を有する基底 B_t^* ($t = 1, \dots, d$) とを用いて暗号処理を実行し、

10

前記鍵生成装置の前記復号鍵生成部は、乱数値 δ, ϕ_0 と所定の値 κ とに基づき数 9 に示す要素 k_0^* を生成するとともに、前記乱数値 $\delta, \phi_{t, i}$ ($i = 1, \dots, w_t$) に基づき数 10 に示す要素 k_t^* とを生成し、

前記暗号化装置の前記暗号化データ生成部は、乱数値 η_0 と所定の値 ζ とに基づき数 11 に示す要素 c_0 を生成するとともに、変数 $\rho(i)$ が肯定形の組 $(t, \vec{v}_i)$ である場合には、乱数値 $\theta_i, \eta_{i, i'}$ ($i = 1, \dots, L, i' = 1, \dots, z_t$) とに基づき数 12 に示す要素 c_i を生成し、変数 $\rho(i)$ が否定形の組 $\neg(t, \vec{v}_i)$ である場合には、乱数値 $\eta_{i, i'}$ ($i = 1, \dots, L, i' = 1, \dots, z_t$) とに基づき数 13 に示す要素 c_i を生成する

20

ことを特徴とする請求項 6 に記載の暗号処理システム。

【数 9】

$$k_0^* := (\delta, 0, \kappa, \phi_0, 0)_{\mathbb{B}_0^*}$$

30

【数 10】

$$k_t^* := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}))}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{(\phi_{t,1}, \dots, \phi_{t,w_t})}^{w_t}, \overbrace{0^{z_t}}^{z_t})_{\mathbb{B}_t^*}$$

【数 11】

$$c_0 := (-s_0, 0, \zeta, 0, \eta_0)_{\mathbb{B}_0}$$

40

【数 12】

$$c_i := (\overbrace{(s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_t})}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{0^{w_t}}^{w_t}, \overbrace{(\eta_{i,1}, \dots, \eta_{i,z_t})}^{z_t})_{\mathbb{B}_t}$$

【数 1 3】

$$c_i := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_t})}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{0^{w_t}}^{w_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,z_t}}^{z_t})_{\mathbb{B}_t}$$

【請求項 8】

$t = 0, \dots, d$ (d は 1 以上の整数) の各整数 t についての基底 B_t と基底 B^*_t とを用いて暗号処理を実行する暗号処理システムにおいて、復号鍵 sk_Γ を生成する鍵生成装置であり、

$t = 1, \dots, d$ の少なくとも 1 つ以上の整数 t について、識別情報 t と、属性ベクトル $x^{\rightarrow}_t := (x_t, i')$ ($i' = 1, \dots, n_t$, n_t は 1 以上の整数) とを有する属性集合 Γ を入力する第 1 情報入力部と、

前記第 1 情報入力部が入力した属性集合 Γ に基づき、要素 k^*_0 と、前記属性集合 Γ に含まれる各識別情報 t についての要素 k^*_t とを生成する復号鍵生成部であって、

基底 B^*_0 の基底ベクトル $b^*_{0,p}$ (p は所定の値) の係数として乱数値 δ を設定するとともに、基底ベクトル $b^*_{0,q}$ (q は前記 p とは異なる所定の値) の係数として所定の値 κ を設定した要素 k^*_0 を生成し、

前記属性集合 Γ に含まれる各識別情報 t について、基底 B^*_t の基底ベクトル $b^*_{t,i'}$ ($i' = 1, \dots, n_t$) の係数として前記乱数値 δ 倍した x_t, i' を設定した要素 k^*_t とを生成する復号鍵生成部と、

前記第 1 情報入力部が入力した変数 $\rho(i)$ と、前記復号鍵生成部が生成した要素 k^*_0 及び要素 k^*_i とを含むデータを復号鍵 sk_Γ として配布する鍵配布部とを備えることを特徴とする鍵生成装置。

【請求項 9】

$t = 0, \dots, d$ (d は 1 以上の整数) の各整数 t についての基底 B_t と基底 B^*_t とを用いて暗号処理を実行する暗号処理システムにおいて、暗号化データ c を生成する暗号化装置であり、

$i = 1, \dots, L$ (L は 1 以上の整数) の各整数 i についての変数 $\rho(i)$ であって、識別情報 t ($t = 1, \dots, d$ のいずれかの整数) と、属性ベクトル $v^{\rightarrow}_i := (v_i, i')$ ($i' = 1, \dots, n_t$, n_t は 1 以上の整数) との肯定形の組 $(t, v^{\rightarrow}_i)$ 又は否定形の組 $\neg(t, v^{\rightarrow}_i)$ のいずれかである変数 $\rho(i)$ と、 L 行 r 列 (r は 1 以上の整数) の所定の行列 M とを入力する第 2 情報入力部と、

r 個の要素を有するベクトル $f^{\rightarrow}$ 及び $w^{\rightarrow}$ と、前記第 2 情報入力部が入力した行列 M とに基づき生成される列ベクトル $s^{\rightarrow T} := (s_1, \dots, s_L)^T := M \cdot f^{\rightarrow T}$ と、値 $s_0 := w^{\rightarrow} \cdot f^{\rightarrow T}$ と、所定の値 θ_i ($i = 1, \dots, L$) とに基づき、要素 c_0 と、 $i = 1, \dots, L$ の各整数 i についての要素 c_i とを生成する暗号化データ生成部であって、

基底 B_0 の基底ベクトル $b_{0,p}$ (p は前記 p) の係数として前記値 $-s_0$ を設定し、基底ベクトル $b_{0,q}$ (q は前記 q) の係数として所定の値 ζ を設定して要素 c_0 を生成し、

$i = 1, \dots, L$ の各整数 i について、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ である場合には、その組の識別情報 t が示す基底 B_t の基底ベクトル $b_{t,i'}$ の係数として $s_i + \theta_i v_{i,1}$ を設定するとともに、前記識別情報 t と $i' = 2, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}$ の係数として $\theta_i v_{i,i'}$ を設定して要素 c_i を生成し、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ である場合には、その組の識別情報 t と $i' = 1, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}$ の係数として $s_i v_{i,i'}$ を設定して要素 c_i を生成する暗号化データ生成部と、

前記第 2 情報入力部が入力した属性集合 Γ と、前記暗号化データ生成部が生成した要素 c_0 及び要素 c_t とを暗号化データ c として出力するデータ出力部と

10

20

30

40

50

を備えることを特徴とする暗号化装置。

【請求項10】

$t = 0, \dots, d$ (d は1以上の整数)の各整数 t についての基底 B_t と基底 B^*_t とを用いて暗号処理を実行する暗号処理システムにおいて、暗号化データ c を復号鍵 sk_r で復号する復号装置であり、

r 個 (r は1以上の整数)の要素を有するベクトル $f \rightarrow$ 及び $w \rightarrow$ と L 行 r 列の所定の行列 M とに基づき生成される列ベクトル $s \rightarrow^T := (s_1, \dots, s_L)^T := M \cdot f \rightarrow^T$ 及び値 $s_0 := w \rightarrow \cdot f \rightarrow^T$ と、所定の値 θ_i ($i = 1, \dots, L$)と、 $i = 1, \dots, L$ (L は1以上の整数)の各整数 i についての変数 $\rho(i)$ であって、識別情報 t ($t = 1, \dots, d$ のいずれかの整数)と、属性ベクトル $v \rightarrow_{i'} := (v_{i, i'})$ ($i' = 1, \dots, n_t, n_t$ は1以上の整数)との肯定形の組 $(t, v \rightarrow_{i'})$ 又は否定形の組 $\neg(t, v \rightarrow_{i'})$ のいずれかである変数 $\rho(i)$ とに基づき生成された要素 c_0 と、要素 c_i ($i = 1, \dots, L$ の各整数)とであって、

10

基底 B_0 の基底ベクトル $b_{0,p}$ (p は所定の値)の係数として前記値 $-s_0$ が設定され、基底ベクトル $b_{0,q}$ (q は前記 p とは異なる所定の値)の係数として所定の値 ζ が設定された要素 c_0 と、

$i = 1, \dots, L$ の各整数 i について、変数 $\rho(i)$ が肯定形の組 $(t, v \rightarrow_{i'})$ である場合には、その組の識別情報 t が示す基底 B_t の基底ベクトル $b_{t,1}$ の係数として $s_i + \theta_i v_{i,1}$ が設定されるとともに、前記識別情報 t と $i' = 2, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}$ の係数として $\theta_i v_{i,i'}$ が設定された要素 c_i であって、変数 $\rho(i)$ が否定形の組 $\neg(t, v \rightarrow_{i'})$ である場合には、その組の識別情報 t と $i' = 1, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}$ の係数として $s_i v_{i,i'}$ が設定された要素 c_i と

20

を前記変数 $\rho(i)$ とともに、暗号化データ c として取得するデータ取得部と、

$t = 1, \dots, d$ の少なくとも1つ以上の整数 t について、識別情報 t と、属性ベクトル $x \rightarrow_t := (x_{t,i'})$ ($i' = 1, \dots, n_t, n_t$ は1以上の整数)とを有する属性集合 Γ に基づき生成された要素 k^*_0 と、要素 k^*_t (t は前記属性集合 Γ に含まれる各整数)とであって、

基底 B^*_0 の基底ベクトル $b^*_{0,p}$ (p は前記 p)の係数として乱数値 δ が設定されるとともに、基底ベクトル $b^*_{0,q}$ (q は前記 q)の係数として所定の値 κ が設定された要素 k^*_0 と、

30

前記属性集合 Γ に含まれる各識別情報 t について、基底 B^*_t の基底ベクトル $b^*_{t,i'}$ ($i' = 1, \dots, n_t$)の係数として前記乱数値 δ 倍した $x_{t,i'}$ が設定された要素 k^*_t と

を前記属性集合 Γ とともに、復号鍵 sk_r として取得する復号鍵取得部と、

前記データ取得部が取得した暗号化データ c に含まれる変数 $\rho(i)$ と、前記復号鍵取得部が取得した復号鍵 sk_s に含まれる属性集合 Γ とに基づき、 $i = 1, \dots, L$ の各整数 i のうち、変数 $\rho(i)$ が肯定形の組 $(t, v \rightarrow_{i'})$ であり、かつ、その組の $v \rightarrow_{i'}$ と、その組の識別情報 t が示す Γ に含まれる $x \rightarrow_t$ との内積が0となる i と、変数 $\rho(i)$ が否定形の組 $\neg(t, v \rightarrow_{i'})$ であり、かつ、その組の $v \rightarrow_{i'}$ と、その組の識別情報 t が示す Γ に含まれる $x \rightarrow_t$ との内積が0とならない i との集合 I を特定するとともに、特定した集合 I に含まれる i について、 $\alpha_i s_i$ を合計した場合に s_0 となる補完係数 α_i を計算する補完係数計算部と、

40

前記暗号化データ c に含まれる要素 c_0 、要素 c_t と、前記復号鍵 sk_s に含まれる要素 k^*_0 、要素 k^*_i とについて、前記補完係数計算部が特定した集合 I と、前記補完係数計算部が計算した補完係数 α_i とに基づき、数14に示すペアリング演算を行い値 $K = g_T^{\zeta \kappa}$ を計算するペアリング演算部と

を備えることを特徴とする復号装置。

【数 1 4】

$$K := e(c_0, k_0^*) \cdot \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} e(c_i, k_t^*)^{\alpha_i} \cdot \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} e(c_i, k_t^*)^{\alpha_i / (\vec{v}_i \cdot \vec{x}_t)}$$

【請求項 1 1】

鍵生成装置と署名装置と検証装置とを備え、 $t = 1, \dots, d, d+2$ (d は 1 以上の整数) の各整数 t についての基底 B_t と基底 B_t^* とを用いて署名処理を実行する署名処理システムであり、

10

前記鍵生成装置は、

$t = 1, \dots, d$ の少なくとも 1 つ以上の整数 t について、識別情報 t と、属性ベクトル $\vec{x}_t := (x_t, \dots)$ ($i' = 1, \dots, n_t$, n_t は 1 以上の整数) とを有する属性集合 Γ を入力する第 1 情報入力部と、

前記第 1 情報入力部が入力した属性集合 Γ に基づき、前記属性集合 Γ に含まれる各識別情報 t についての要素 k_t^* と、要素 k_{d+2}^* と、要素 k_{d+3}^* とを生成する署名鍵生成部であって、

前記属性集合 Γ に含まれる各識別情報 t について、基底 B_t^* の基底ベクトル $b_{t, i'}^*$ ($i' = 1, \dots, n_t$) の係数として前記乱数値 δ 倍した $x_t, \dots$ を設定した要素 k_t^* を生成し、

20

基底 B_{d+2}^* の基底ベクトル $b_{d+2, p'}^*$ (p' は所定の値) の係数として前記乱数値 δ を設定した要素 k_{d+2}^* を生成し、

基底 B_{d+2}^* の基底ベクトル $b_{d+2, q'}^*$ (q' は前記 p' とは異なる所定の値) の係数として前記乱数値 δ を設定した要素 k_{d+3}^* を生成する署名鍵生成部とを備え、

前記署名装置は、

$i = 1, \dots, L$ (L は 1 以上の整数) の各整数 i についての変数 $\rho(i)$ であって、識別情報 t ($t = 1, \dots, d$ のいずれかの整数) と、属性ベクトル $\vec{v}_i := (v_i, \dots)$ ($i' = 1, \dots, n_t$, n_t は 1 以上の整数) との肯定形の組 $(t, \vec{v}_i)$ 又は否定形の組 $\neg(t, \vec{v}_i)$ のいずれかである変数 $\rho(i)$ と、 L 行 r 列 (r は 1 以上の整数) の所定の行列 M と、メッセージ m とを入力する第 2 情報入力部と、

30

前記署名鍵生成部が生成した要素 k_t^* と、要素 k_{d+2}^* と、要素 k_{d+3}^* と、前記属性集合 Γ とを署名鍵 sk_Γ として取得する署名鍵取得部と、

$i = 1, \dots, L$ の各整数 i についての変数 $\rho(i)$ と、前記署名鍵取得部が取得した署名鍵 sk_Γ に含まれる属性集合 Γ とに基づき、 $i = 1, \dots, L$ の各整数 i のうち、変数 $\rho(i)$ が肯定形の組 $(t, \vec{v}_i)$ であり、かつ、その組の $\vec{v}_i$ と、その組の識別情報 t が示す Γ に含まれる $\vec{x}_t$ との内積が 0 となる i と、変数 $\rho(i)$ が否定形の組 $\neg(t, \vec{v}_i)$ であり、かつ、その組の $\vec{v}_i$ と、その組の識別情報 t が示す Γ に含まれる $\vec{x}_t$ との内積が 0 とならない i との集合 I を特定するとともに、特定した集合 I に含まれる i について、前記第 2 情報入力部が入力した行列 M の i 行目の要素である M_i に基づき、 $\alpha_i M_i$ を合計したした場合に所定のベクトル $\vec{w}$ となる補完係数 α_i を計算する補完係数計算部と、

40

前記変数 $\rho(i)$ と、前記属性集合 Γ と、前記補完係数計算部が特定した集合 I と、前記補完係数計算部が計算した補完係数 α_i とに基づき、 $i = 1, \dots, L$ の各整数 i についての s_i^* と、 s_x^* とを生成する署名生成部であって、

β_i を、行列 M の各行番号を表す各整数 i について、 $\beta_i M_i$ を合計した場合に全ての要素が 0 となる値とし、

行列 M の各行番号を表す各整数 i について、

$i \in I$ であって、変数 $\rho(i)$ が肯定形の組 $(t, \vec{v}_i)$ である場合には、値 $\gamma_i := \alpha_i$ とし、ベクトル $\vec{y}_i := (y_i, \dots)$ ($t := 1, \dots, n_t$) を $\vec{y}_i$ と $\vec{v}$

50

$\vec{v}_i$ との内積が 0 で、 $y_{i,1} = 1$ であるベクトルとし、
 $i \in I$ であって、変数 $\rho(i)$ が否定形の組 $\neg(t, \vec{v}_i)$ である場合には、値 $\gamma_i := \alpha_i / (\vec{v}_i \cdot \vec{x}_t)$ とし、ベクトル $\vec{y}_i := (y_{i,1})$ ($l := 1, \dots, n_t$) を $\vec{y}_i$ と $\vec{v}_i$ との内積が 1 であるベクトルとし、
 $i \in I$ でなく、変数 $\rho(i)$ が肯定形の組 $(t, \vec{v}_i)$ である場合には、値 $\gamma_i := 0$ とし、ベクトル $\vec{y}_i := (y_{i,1})$ ($l := 1, \dots, n_t$) を $\vec{y}_i$ と $\vec{v}_i$ との内積が 0 で、 $y_{i,1} = 1$ であるベクトルとし、
 $i \in I$ でなく、変数 $\rho(i)$ が否定形の組 $\neg(t, \vec{v}_i)$ である場合には、値 $\gamma_i := 0$ とし、ベクトル $\vec{y}_i := (y_{i,1})$ ($l := 1, \dots, n_t$) を $\vec{y}_i$ と $\vec{v}_i$ との内積が 1 であるベクトルとして、
 数 15 に示すように行列 M の各行番号を表す各整数 i についての s^*_i と、 s^*_X とを生成する署名生成部と
 を備え、

10

前記検証装置は、

前記署名生成部が生成した s^*_i 、 s^*_X と、メッセージ m と、変数 $\rho(i)$ と、行列 M とを含む署名データ sig を取得するデータ取得部と、

r 個の要素を有するベクトル $\vec{f}$ と、前記ベクトル $\vec{w}$ と、前記データ取得部が取得した行列 M とに基づき生成される列ベクトル $\vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T$ と、値 $s_0 := \vec{w} \cdot \vec{f}^T$ と、所定の値 θ_i (i は行列 M の各行番号と X) とに基づき、行列 M の各行番号を表す各整数 i についての要素 c_i と、要素 c_X とを生成する暗号化データ生成部であって、

20

前記行列 M の各行番号を表す各整数 i について、変数 $\rho(i)$ が肯定形の組 $(t, \vec{v}_i)$ である場合には、その組の識別情報 t が示す基底 B_t の基底ベクトル $b_{t,1}$ の係数として $s_i + \theta_i v_{i,1}$ を設定するとともに、前記識別情報 t と $i' = 2, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}$ の係数として $\theta_i v_{i,i'}$ を設定して要素 c_i を生成し、変数 $\rho(i)$ が否定形の組 $\neg(t, \vec{v}_i)$ である場合には、その組の識別情報 t と $i' = 1, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}$ の係数として $s_i v_{i,i'}$ を設定して要素 c_i を生成し、

基底 B_{d+2} の基底ベクトル $b_{d+2,p'}$ (p' は前記 p') に $s_0 - \theta_X m$ を設定するとともに、基底ベクトル $b_{d+2,q'}$ (q' は前記 q') に θ_X を設定して要素 c_X を生成する暗号化データ生成部と、

30

行列 M の各行番号と X とを示す各整数 i について、数 16 に示すペアリング演算を行い、前記署名データ sig の正当性を検証するペアリング演算部と
 を備えることを特徴とする署名処理システム。

【数 15】

$$s_i^* := \gamma_i \cdot \xi k_t^* + \beta_i \cdot \left(\sum_{l=1}^{n_t} y_{i,l} \cdot b_{t,l}^* \right) + r_i^*,$$

$$s_X^* := \xi(k_{d+2}^* + m \cdot k_{d+3}^*) + r_X^*$$

40

ここで、

ξ, r_i^*, r_X^* は所定の値

である。

【数 1 6】

$$\prod e(c_i, s_i^*)$$

【請求項 1 2】

前記署名処理システムは、少なくとも基底ベクトル $b_{0,i}$ ($i = 1, \dots, 4$) を有する基底 B_0 と、少なくとも基底ベクトル $b_{t,i}$ ($i = 1, \dots, n_t, \dots, n_t + u_t, \dots, n_t + u_t + w_t, \dots, n_t + u_t + w_t + z_t$) (u_t, w_t, z_t は 1 以上の整数、) を有する基底 B_t ($t = 1, \dots, d$) と、少なくとも基底ベクトル $b_{d+1,i}$ ($i = 1, \dots, 4$) を有する基底 B_{d+1} と、少なくとも基底ベクトル $b_{d+2,i}$ ($i = 1, \dots, 8$) を有する基底 B_{d+2} と、少なくとも基底ベクトル $b_{0,i}^*$ ($i = 1, \dots, 4$) を有する基底 B_0^* と、少なくとも基底ベクトル $b_{t,i}^*$ ($i = 1, \dots, n_t, \dots, n_t + u_t, \dots, n_t + u_t + w_t, \dots, n_t + u_t + w_t + z_t$) を有する基底 B_t^* ($t = 1, \dots, d$) と、少なくとも基底ベクトル $b_{d+1,i}^*$ ($i = 1, \dots, 4$) を有する基底 B_{d+1}^* と、少なくとも基底ベクトル $b_{d+2,i}^*$ ($i = 1, \dots, 8$) を有する基底 B_{d+2}^* とを用いて署名処理を実行し、

10

前記鍵生成装置の前記署名鍵生成部は、

前記属性集合 Γ に含まれる各識別情報 t と、 $t = 1, \dots, w_t$ の各整数 l とについて、要素 $k_{0,l}^*$ と、要素 $k_{t,l}^*$ と、要素 $k_{d+2,l}^*$ と、要素 $k_{d+3,l}^*$ とを、乱数値 δ_j , ϕ_0 , $\phi_{t,l}$, $\phi_{d+2,1}$, $\phi_{d+2,2}$, $\phi_{d+3,1}$, $\phi_{d+3,2}$ に基づき、数 1 7 に示すように生成し、

20

前記署名装置は、さらに、

前記行列 M の $L + 1$ 行目に所定の行ベクトル M_{L+1} を追加する行列生成部を備え

前記署名装置の前記署名生成部は、

要素 $k_{0,l}^*$ と、要素 $k_{t,l}^*$ と、要素 $k_{d+2,l}^*$ と、要素 $k_{d+3,l}^*$ と、乱数値 ξ とに基づき、数 1 8 に示すように、 $i = 1, \dots, L + 1$ の各整数 i についての s_i^* と、 s_x^* として $s_{0,l}^*$ 及び $s_{L+2,l}^*$ とを生成し、

前記検証装置の前記暗号化データ生成部は、

30

$i = 1, \dots, L + 1$ の各整数 i と、 $i' = 1, \dots, z_t$ の各整数 i' とについて、乱数値 θ_i , η_i , $\eta_{L+2,1}$, $\eta_{L+2,2}$ に基づき、数 1 9 に示すように、要素 c_i と、要素 c_x として要素 c_0 及び要素 c_{L+2} とを生成し、

前記ペアリング演算部は、 $i = 1, \dots, L + 2$ の各整数 i について、前記ペアリング演算を行う

ことを特徴とする請求項 1 1 に記載の署名処理システム。

【数 1 7】

$$k_0^* := (\delta, 0, \phi_0, 0)_{\mathbb{B}_0^*},$$

40

$$k_t^* := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}), 0^{u_t}, \phi_{t,1}, \dots, \phi_{t,w_t}, 0^{z_t})}^{n_t, u_t, w_t, z_t})_{\mathbb{B}_t^*},$$

$$k_{d+2}^* := (\delta(1, 0), 0, 0, \phi_{d+2,1}, \phi_{d+2,2}, 0, 0)_{\mathbb{B}_{d+2}^*},$$

$$k_{d+3}^* := (\delta(0, 1), 0, 0, \phi_{d+3,1}, \phi_{d+3,2}, 0, 0)_{\mathbb{B}_{d+2}^*},$$

【数 1 8】

$$c_0 := (-s_0 - s_{L+2}, 0, 0, \eta_0)_{\mathbb{B}_0},$$

$$\text{if } \rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \eta_{i,1}, \dots, \eta_{i,z_t} \xleftarrow{\mathbb{U}} \mathbb{F}_q,$$

$$c_i := (\overbrace{s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_t}}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{0^{w_t}}^{w_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,z_t}}^{z_t})_{\mathbb{B}_t},$$

$$\text{if } \rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \eta_{i,1}, \dots, \eta_{i,z_t} \xleftarrow{\mathbb{U}} \mathbb{F}_q,$$

$$c_i := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_t})}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{0^{w_t}}^{w_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,z_t}}^{z_t})_{\mathbb{B}_t},$$

$$c_{L+2} := (s_{L+2} - \theta_{L+2}m, \theta_{L+2}, 0, 0, 0, 0, \eta_{L+2,1}, \eta_{L+2,2})_{\mathbb{B}_{d+2}}$$

10

【数 1 9】

$$s_0^* := \xi k_0^* + r_0^*,$$

20

$$s_i^* := \gamma_i \cdot \xi k_t^* + \beta_i \cdot (\sum_{l=1}^{n_t} y_{i,l} \cdot b_{t,l}^*) + r_i^*,$$

$$s_{L+2}^* := \xi(k_{d+2}^* + m \cdot k_{d+3}^*) + r_{L+2}^*$$

ここで、

$$\xi, r_0^*, r_i^*, r_{L+2}^* \text{ は所定の値}$$

30

である。

【請求項 1 3】

$t = 1, \dots, d, d+2$ (d は 1 以上の整数) の各整数 t についての基底 B_t と基底 B_t^* とを用いて署名処理を実行する署名処理システムにおいて、署名鍵 sk_Γ を生成する鍵生成装置であり、

$t = 1, \dots, d$ の少なくとも 1 つ以上の整数 t について、識別情報 t と、属性ベクトル $x_{\rightarrow t} := (x_{t,i})$ ($i' = 1, \dots, n_t, n_t$ は 1 以上の整数) とを有する属性集合 Γ を入力する第 1 情報入力部と、

40

前記第 1 情報入力部が入力した属性集合 Γ に基づき、前記属性集合 Γ に含まれる各識別情報 t についての要素 k_t^* と、要素 k_{d+2}^* と、要素 k_{d+3}^* とを生成する署名鍵生成部であって、

前記属性集合 Γ に含まれる各識別情報 t について、基底 B_t^* の基底ベクトル $b_{t,i}^*$ ($i' = 1, \dots, n_t$) の係数として前記乱数値 δ 倍した $x_{t,i}$ を設定した要素 k_t^* を生成し、

基底 B_{d+2}^* の基底ベクトル $b_{d+2,p}^*$ (p' は所定の値) の係数として前記乱数値 δ を設定した要素 k_{d+2}^* を生成し、

基底 B_{d+2}^* の基底ベクトル $b_{d+2,q}^*$ (q' は前記 p' とは異なる所定の値) の係数として前記乱数値 δ を設定した要素 k_{d+3}^* を生成する署名鍵生成部と、

50

前記第1情報入力部が入力した属性集合 Γ と、前記署名鍵生成部が生成した要素 $k^*_{t^*}$ と、要素 k^*_{d+2} と、要素 k^*_{d+3} とを含むデータを署名鍵 sk_{Γ} として配布する鍵配布部と

を備えることを特徴とする鍵生成装置。

【請求項14】

$t = 1, \dots, d, d+2$ (d は1以上の整数)の各整数 t についての基底 B_t と基底 $B^*_{t^*}$ とを用いて署名処理を実行する署名処理システムにおいて、署名データ sig を生成する署名装置であり、

$i = 1, \dots, L$ (L は1以上の整数)の各整数 i についての変数 $\rho(i)$ であって、識別情報 t ($t = 1, \dots, d$ のいずれかの整数)と、属性ベクトル $v^{\rightarrow}_i := (v_{i, i'})$ ($i' = 1, \dots, n_t, n_t$ は1以上の整数)との肯定形の組 $(t, v^{\rightarrow}_i)$ 又は否定形の組 $\neg(t, v^{\rightarrow}_i)$ のいずれかである変数 $\rho(i)$ と、 L 行 r 列(r は1以上の整数)の所定の行列 M と、メッセージ m とを入力する第2情報入力部と、

$t = 1, \dots, d$ の少なくとも1つ以上の整数 t について、識別情報 t と、属性ベクトル $x^{\rightarrow}_t := (x_{t, i'})$ ($i' = 1, \dots, n_t, n_t$ は1以上の整数)とを有する属性集合 Γ に基づき生成された要素 $k^*_{t^*}$ (t は前記属性集合 Γ に含まれる各識別情報)と、要素 k^*_{d+2} と、要素 k^*_{d+3} とであって、

前記属性集合 Γ に含まれる各識別情報 t について、基底 $B^*_{t^*}$ の基底ベクトル $b^*_{t, i'}$ ($i' = 1, \dots, n_t$)の係数として前記乱数値 δ 倍した $x_{t, i'}$ が設定された要素 $k^*_{t^*}$ と、

基底 B^*_{d+2} の基底ベクトル $b^*_{d+2, p'}$ (p' は所定の値)の係数として前記乱数値 δ が設定された要素 k^*_{d+2} と、

基底 B^*_{d+2} の基底ベクトル $b^*_{d+2, q'}$ (q' は前記 p' とは異なる所定の値)の係数として前記乱数値 δ が設定された要素 k^*_{d+3} と

を前記属性集合 Γ とともに、署名鍵 sk_{Γ} として取得する署名鍵取得部と、

$i = 1, \dots, L$ の各整数 i についての変数 $\rho(i)$ と、前記署名鍵取得部が取得した署名鍵 sk_{Γ} に含まれる属性集合 Γ とに基づき、 $i = 1, \dots, L$ の各整数 i のうち、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ であり、かつ、その組の $v^{\rightarrow}_i$ と、その組の識別情報 t が示す Γ に含まれる $x^{\rightarrow}_t$ との内積が0となる i と、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ であり、かつ、その組の $v^{\rightarrow}_i$ と、その組の識別情報 t が示す Γ に含まれる $x^{\rightarrow}_t$ との内積が0とならない i との集合 I を特定するとともに、特定した集合 I に含まれる i について、前記第2情報入力部が入力した行列 M の i 行目の要素である M_i に基づき、 $\alpha_i M_i$ を合計した場合に所定のベクトル $w^{\rightarrow}$ となる補完係数 α_i を計算する補完係数計算部と、

前記変数 $\rho(i)$ と、前記属性集合 Γ と、前記補完係数計算部が特定した集合 I と、前記補完係数計算部が計算した補完係数 α_i とに基づき、行列 M の各行番号を表す各整数 i についての $s^*_{i^*}$ と、 s^*_x とを生成する署名生成部であって、

β_i を、行列 M の各行番号を表す各整数 i について、 $\beta_i M_i$ を合計した場合に全ての要素が0となる値とし、

行列 M の各行番号を表す各整数 i について、

$i \in I$ であって、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ である場合には、値 $\gamma_i := \alpha_i$ とし、ベクトル $y^{\rightarrow}_i := (y_{i, i'})$ ($i' = 1, \dots, n_t$)を $y^{\rightarrow}_i$ と $v^{\rightarrow}_i$ との内積が0で、 $y_{i, 1} = 1$ であるベクトルとし、

$i \in I$ であって、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ である場合には、値 $\gamma_i := \alpha_i / (v^{\rightarrow}_i \cdot x^{\rightarrow}_t)$ とし、ベクトル $y^{\rightarrow}_i := (y_{i, i'})$ ($i' = 1, \dots, n_t$)を $y^{\rightarrow}_i$ と $v^{\rightarrow}_i$ との内積が1であるベクトルとし、

$i \in I$ でなく、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ である場合には、値 $\gamma_i := 0$ とし、ベクトル $y^{\rightarrow}_i := (y_{i, i'})$ ($i' = 1, \dots, n_t$)を $y^{\rightarrow}_i$ と $v^{\rightarrow}_i$ との内積が0で、 $y_{i, 1} = 1$ であるベクトルとし、

$i \in I$ でなく、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ である場合には、値 $\gamma_i :$

10

20

30

40

50

$= 0$ とし、ベクトル $y^{\rightarrow}_i := (y_{i,1}, \dots, y_{i,n_t})$ ($i := 1, \dots, n_t$) を $y^{\rightarrow}_i$ と $v^{\rightarrow}_i$ との内積が 1 であるベクトルとして、

数 20 に示すように行列 M の各行番号を表すの各整数 i についての s^*_i と、 s^*_X とを生成する署名生成部と

前記第 2 情報入力部が入力した変数 $\rho(i)$ と行列 M とメッセージ m と、前記署名生成部が生成した s^*_0 と s^*_i と s^*_X とを含むデータを署名データ sig として出力するデータ出力部と

を備えることを特徴とする署名装置。

【数 20】

$$s_i^* := \gamma_i \cdot \xi k_t^* + \beta_i \cdot \left(\sum_{l=1}^{n_t} y_{i,l} \cdot b_{t,l}^* \right) + r_i^*,$$

$$s_X^* := \xi(k_{d+2}^* + m \cdot k_{d+3}^*) + r_X^*$$

ここで、

ξ, r_i^*, r_X^* は所定の値

である。

【請求項 15】

$t = 1, \dots, d, d+2$ (d は 1 以上の整数) の各整数 t についての基底 B_t と基底 B^*_t とを用いて署名処理を実行する署名処理システムにおいて、署名データ sig を検証する検証装置であり、

$t = 1, \dots, d$ の少なくとも 1 つ以上の整数 t について、識別情報 t と、属性ベクトル $x^{\rightarrow}_t := (x_{t,1}, \dots, x_{t,n_t})$ ($i' = 1, \dots, n_t$, n_t は 1 以上の整数) とを有する属性集合 Γ に基づき生成された要素 k^*_t (t は前記属性集合 Γ に含まれる各識別情報) と、要素 k^*_{d+2} と、要素 k^*_{d+3} とであって、

前記属性集合 Γ に含まれる各識別情報 t について、基底 B^*_t の基底ベクトル $b^*_{t,i'}$ ($i' = 1, \dots, n_t$) の係数として前記乱数値 δ 倍した $x_{t,i'}$ が設定された要素 k^*_t と、

基底 B^*_{d+2} の基底ベクトル $b^*_{d+2,p'}$ (p' は所定の値) の係数として前記乱数値 δ が設定された要素 k^*_{d+2} と、

基底 B^*_{d+2} の基底ベクトル $b^*_{d+2,q'}$ (q' は前記 p' とは異なる所定の値) の係数として前記乱数値 δ が設定された要素 k^*_{d+3} と、

$i = 1, \dots, L$ (L は 1 以上の整数) の各整数 i についての変数 $\rho(i)$ であって、識別情報 t ($t = 1, \dots, d$ のいずれかの整数) と、属性ベクトル $v^{\rightarrow}_i := (v_{i,1}, \dots, v_{i,n_t})$ ($i' = 1, \dots, n_t$, n_t は 1 以上の整数) との肯定形の組 $(t, v^{\rightarrow}_i)$ 又は否定形の組 $\neg(t, v^{\rightarrow}_i)$ のいずれかである変数 $\rho(i)$ と、前記属性集合 Γ とに基づき計算された集合 I と補完係数 α_i とであって、

$i = 1, \dots, L$ の各整数 i のうち、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ であり、かつ、その組の $v^{\rightarrow}_i$ と、その組の識別情報 t が示す Γ に含まれる $x^{\rightarrow}_t$ との内積が 0 となる i と、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ であり、かつ、その組の $v^{\rightarrow}_i$ と、その組の識別情報 t が示す Γ に含まれる $x^{\rightarrow}_t$ との内積が 0 とならない i との集合 I と、

集合 I に含まれる i について、前記第 2 情報入力部が入力した行列 M の i 行目の要素である M_i に基づき、 $\alpha_i M_i$ を合計した場合に所定のベクトル $w^{\rightarrow}$ となる補完係数 α_i と、

前記変数 $\rho(i)$ と、前記属性集合 Γ と、前記集合 I と、前記補完係数 α_i とに基づき

10

20

30

40

50

生成された s^*_i ($i = 1, \dots, L$ の各整数) と、 s^*_X とであって、

β_i を、行列 M の各行番号を表す各整数 i について、 $\beta_i M_i$ を合計した場合に全ての要素が 0 となる値とし、

行列 M の各行番号を表す各整数 i について、

$i \in I$ であって、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ である場合には、値 $\gamma_i := \alpha_i$ とし、ベクトル $y^{\rightarrow}_i := (y_{i,l})$ ($l := 1, \dots, n_t$) を $y^{\rightarrow}_i$ と $v^{\rightarrow}_i$ との内積が 0 で、 $y_{i,1} = 1$ であるベクトルとし、

$i \in I$ であって、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ である場合には、値 $\gamma_i := \alpha_i / (v^{\rightarrow}_i \cdot x^{\rightarrow}_t)$ とし、ベクトル $y^{\rightarrow}_i := (y_{i,l})$ ($l := 1, \dots, n_t$) を $y^{\rightarrow}_i$ と $v^{\rightarrow}_i$ との内積が 1 であるベクトルとし、

$i \in I$ でなく、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ である場合には、値 $\gamma_i := 0$ とし、ベクトル $y^{\rightarrow}_i := (y_{i,l})$ ($l := 1, \dots, n_t$) を $y^{\rightarrow}_i$ と $v^{\rightarrow}_i$ との内積が 0 で、 $y_{i,1} = 1$ であるベクトルとし、

$i \in I$ でなく、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ である場合には、値 $\gamma_i := 0$ とし、ベクトル $y^{\rightarrow}_i := (y_{i,l})$ ($l := 1, \dots, n_t$) を $y^{\rightarrow}_i$ と $v^{\rightarrow}_i$ との内積が 1 であるベクトルとして、

数 2 1 に示すように生成された s^*_i と、 s^*_X と

のうち、 s^*_i と、 s^*_X とメッセージ m と、変数 $\rho(i)$ と、行列 M とを含む署名データ s_{ig} を取得するデータ取得部と、

r 個 (r は 1 以上の整数) の要素を有するベクトル $f^{\rightarrow}$ と、前記ベクトル $w^{\rightarrow}$ と、前記第 2 情報入力部が入力した行列 M とに基づき生成される列ベクトル $s^{\rightarrow T} := (s_1, \dots, s_L)^T := M \cdot f^{\rightarrow T}$ と、値 $s_0 := w^{\rightarrow} \cdot f^{\rightarrow T}$ と、所定の値 θ_i (i は行列 M の各行番号と X) とに基づき、 $i = 1, \dots, L$ の各整数 i についての要素 c_i と、要素 c_X とを生成する暗号化データ生成部であって、

行列 M の各行番号を表す各整数 i について、変数 $\rho(i)$ が肯定形の組 $(t, v^{\rightarrow}_i)$ である場合には、その組の識別情報 t が示す基底 B_t の基底ベクトル $b_{t,1}$ の係数として $s_i + \theta_i v_{i,1}$ を設定するとともに、前記識別情報 t と $i' = 2, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}$ の係数として $\theta_i v_{i,i'}$ を設定して要素 c_i を生成し、変数 $\rho(i)$ が否定形の組 $\neg(t, v^{\rightarrow}_i)$ である場合には、その組の識別情報 t と $i' = 1, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b_{t,i'}$ の係数として $s_i v_{i,i'}$ を設定して要素 c_i を生成し、

基底 B_{d+2} の基底ベクトル $b_{d+2,p'}$ (p' は前記 p') に $s_0 - \theta_X m$ を設定するとともに、基底ベクトル $b_{d+2,q'}$ (q' は前記 q') に θ_X を設定して要素 c_X を生成する暗号化データ生成部と、

行列 M の各行番号と X とを示す各整数 i について、数 2 2 に示すペアリング演算を行い、前記署名データ s_{ig} の正当性を検証するペアリング演算部と

を備えることを特徴とする検証装置。

【数 2 1】

$$s_i^* := \gamma_i \cdot \xi k_t^* + \beta_i \cdot \left(\sum_{l=1}^{n_t} y_{i,l} \cdot b_{t,l}^* \right) + r_i^*,$$

$$s_X^* := \xi (k_{d+2}^* + m \cdot k_{d+3}^*) + r_X^*$$

ここで、

ξ, r_i^*, r_X^* は所定の値

である。

10

20

30

40

50

【数 2 2】

$$\prod e(c_i, s_i^*)$$

【発明の詳細な説明】

【技術分野】

【0001】

この発明は、関数型暗号（Functional Encryption, FE）方式に関するものである。

10

【背景技術】

【0002】

非特許文献3-6, 10, 12, 13, 15, 18には、関数型暗号方式の1つのクラスであるIDベース暗号（Identity-Based Encryption, IBE）方式についての記載がある。また、非特許文献2, 7, 9, 16, 19, 23-26, 28には、関数型暗号方式の1つのクラスである属性ベース暗号（Attribute-Based Encryption, ABE）方式についての記載がある。

【先行技術文献】

【非特許文献】

【0003】

20

【非特許文献1】Beimel, A., Secure schemes for secret sharing and key distribution. PhD Thesis, Israel Institute of Technology, Technion, Haifa, Israel, 1996

【非特許文献2】Bethencourt, J., Sahai, A., Waters, B.: Ciphertext policy attribute-based encryption. In: 2007 IEEE Symposium on Security and Privacy, pp. 321-34. IEEE Press (2007)

【非特許文献3】Boneh, D., Boyen, X.: Efficient selective-ID secure identity based encryption without random oracles. In: Cachin, C., Camenisch, J. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 223-38. Springer Heidelberg (2004)

30

【非特許文献4】Boneh, D., Boyen, X.: Secure identity based encryption without random oracles. In: Franklin, M. K. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 443-59. Springer Heidelberg (2004)

40

【非特許文献5】Boneh, D., Boyen, X., Goh, E.: Hierarchical identity based encryption with constant size ciphertext. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 440-56. Springer Heidelberg (2005)

【非特許文献6】Boneh, D., Franklin, M.: Identity-based encryption from the Weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213-29. Springer Heidelberg (2001)

50

【非特許文献7】Boneh, D., Hamburg, M.: Generalized identity based and broadcast encryption scheme. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 455-70. Springer Heidelberg (2008)

【非特許文献8】Boneh, D., Katz, J., Improved efficiency for CCA-secure cryptosystems built using identity based encryption. RSA-CT 2005, LNCS, Springer Verlag (2005)

【非特許文献9】Boneh, D., Waters, B.: Conjunctive, subset, and range queries on encrypted data. In: Vadhan, S. P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 535-54. Springer Heidelberg (2007) 10

【非特許文献10】Boyen, X., Waters, B.: Anonymous hierarchical identity-based encryption (without random oracles). In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 290-07. Springer Heidelberg (2006) 20

【非特許文献11】Canetti, R., Halevi S., Katz J., Chosen-ciphertext security from identity-based encryption. EUROCRYPT2004, LNCS, Springer-Verlag (2004) 20

【非特許文献12】Cocks, C.: An identity based encryption scheme based on quadratic residues. In: Honary, B. (ed.) IMA Int. Conf. LNCS, vol. 2260, pp. 360-63. Springer Heidelberg (2001)

【非特許文献13】Gentry, C.: Practical identity-based encryption without random oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 445-64. Springer Heidelberg (2006) 30

【非特許文献14】Gentry, C., Halevi, S.: Hierarchical identity-based encryption with polynomially many levels. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 437-56. Springer Heidelberg (2009)

【非特許文献15】Gentry, C., Silverberg, A.: Hierarchical ID-based cryptography. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 548-66. Springer Heidelberg (2002) 40

【非特許文献16】Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: ACM Conference on Computer and Communication Security 2006, pp. 89-8, ACM (2006)

【非特許文献17】Groth, J., Sahai, A.: Efficient 50

non-interactive proof systems for bilinear groups. In: Smart, N. P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 415-32. Springer Heidelberg (2008)

【非特許文献18】Horwitz, J., Lynn, B.: Towards hierarchical identity-based encryption. In: Knudsen, L. R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 466-81. Springer Heidelberg (2002)

【非特許文献19】Katz, J., Sahai, A., Waters, B.: Predicate encryption supporting disjunctions, polynomial equations, and inner products. In: Smart, N. P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 146-62. Springer Heidelberg (2008) 10

【非特許文献20】Lewko, A. B., Waters, B.: Fully secure HIBE with short ciphertexts. ePrint, IACR, <http://eprint.iacr.org/2009/482>

【非特許文献21】Okamoto, T., Takashima, K.: Homomorphic encryption and signatures from vector decomposition. In: Galbraith, S. D., Paterson, K. G. (eds.) Pairing 2008. LNCS, vol. 5209, pp. 57-4. Springer Heidelberg (2008) 20

【非特許文献22】Okamoto, T., Takashima, K.: Hierarchical predicate encryption for Inner-Products, In: ASIACRYPT 2009, Springer Heidelberg (2009)

【非特許文献23】Ostrovsky, R., Sahai, A., Waters, B.: Attribute-based encryption with non-monotonic access structures. In: ACM Conference on Computer and Communication Security 2007, pp. 195-03, ACM (2007) 30

【非特許文献24】Pirretti, M., Traynor, P., McDaniel, P., Waters, B.: Secure attribute-based systems. In: ACM Conference on Computer and Communication Security 2006, pp. 99-12, ACM, (2006)

【非特許文献25】Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 457-73. Springer Heidelberg (2005) 40

【非特許文献26】Shi, E., Waters, B.: Delegating capability in predicate encryption systems. In: Aceto, L., Damgard, I., Goldberg, L. A., Halldosson, M. M., Ingolfsson, A., Walukiewicz, I. (eds.) ICALP (2) 2008. LNCS, vol. 5126, pp. 560-78. Springer Heidelberg (2008) 50

【非特許文献27】Waters, B. : Efficient identity based encryption without random oracles. Eurocrypt 2005, LNCS No. 3152, pp. 443・59. Springer Verlag, 2005.

【非特許文献28】Waters, B. : Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization. ePrint, IACR, <http://eprint.iacr.org/2008/290>

【非特許文献29】Waters, B. : Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 619・36. Springer Heidelberg (2009)

10

【発明の概要】

【発明が解決しようとする課題】

【0004】

この発明は、多機能な暗号機能を有する安全な関数型暗号方式を提供することを目的とする。

【課題を解決するための手段】

20

【0005】

この発明に係る暗号処理システムは、

鍵生成装置と暗号化装置と復号装置とを備え、 $t = 0, \dots, d$ (d は1以上の整数)の各整数 t についての基底 B_t と基底 B^*_t とを用いて暗号処理を実行する暗号処理システムであり、

前記鍵生成装置は、

$i = 1, \dots, L$ (L は1以上の整数)の各整数 i についての変数 $\rho(i)$ であって、識別情報 t ($t = 1, \dots, d$ のいずれかの整数)と、属性ベクトル $\vec{v}_i := (v_{i,1}, \dots, v_{i,n_t})$ ($i' = 1, \dots, n_t$, n_t は1以上の整数)との肯定形の組 $(t, \vec{v}_i)$ 又は否定形の組 $\neg(t, \vec{v}_i)$ のいずれかである変数 $\rho(i)$ と、 L 行 r 列 (r は1以上の整数)の所定の行列 M とを入力する第1情報入力部と、

30

r 個の要素を有するベクトル $\vec{f}$ 及び $\vec{w}$ と、前記第1情報入力部が入力した行列 M に基づき生成される列ベクトル $\vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T$ と、値 $s_0 := \vec{w} \cdot \vec{f}^T$ と、所定の値 θ_i ($i = 1, \dots, L$)とに基づき、要素 k^*_0 と、 $i = 1, \dots, L$ の各整数 i についての要素 k^*_i とを生成する復号鍵生成部であって、

基底 B^*_0 の基底ベクトル $b^*_{0,p}$ (p は所定の値)の係数として前記値 $-s_0$ を設定し、基底ベクトル $b^*_{0,q}$ (q は前記 p とは異なる所定の値)の係数として所定の値 κ を設定して要素 k^*_0 を生成し、

$i = 1, \dots, L$ の各整数 i について、変数 $\rho(i)$ が肯定形の組 $(t, \vec{v}_i)$ である場合には、その組の識別情報 t が示す基底 B^*_t の基底ベクトル $b^*_{t,1}$ の係数として $s_i + \theta_i v_{i,1}$ を設定するとともに、前記識別情報 t と $i' = 2, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b^*_{t,i'}$ の係数として $\theta_i v_{i,i'}$ を設定して要素 k^*_i を生成し、変数 $\rho(i)$ が否定形の組 $\neg(t, \vec{v}_i)$ である場合には、その組の識別情報 t と $i' = 1, \dots, n_t$ の各整数 i' とが示す基底ベクトル $b^*_{t,i'}$ の係数として $s_i v_{i,i'}$ を設定して要素 k^*_i を生成する復号鍵生成部とを備え、

40

前記暗号化装置は、

$t = 1, \dots, d$ の少なくとも1つ以上の整数 t について、識別情報 t と、属性ベクトル $\vec{x}_t := (x_{t,1}, \dots, x_{t,n_t})$ ($i' = 1, \dots, n_t$, n_t は1以上の整数)とを有

50

する属性集合 Γ を入力する第 2 情報入力部と、

前記第 2 情報入力部が入力した属性集合 Γ に基づき、要素 c_0 と、前記属性集合 Γ に含まれる各識別情報 t についての要素 c_t とを生成する暗号化データ生成部であって、

基底 B_0 の基底ベクトル $b_{0,p}$ (p は前記 p) の係数として乱数値 δ を設定するとともに、基底ベクトル $b_{0,q}$ (q は前記 q) の係数として所定の値 ζ を設定した要素 c_0 を生成し、

前記属性集合 Γ に含まれる各識別情報 t について、基底 B_t の基底ベクトル $b_{t,i}$ ($i = 1, \dots, n_t$) の係数として前記乱数値 δ 倍した $x_{t,i}$ を設定した要素 c_t とを生成する暗号化データ生成部と

10

を備え、

前記復号装置は、

前記暗号化データ生成部が生成した要素 c_0 及び要素 c_t と、前記属性集合 Γ とを含む暗号化データ c を取得するデータ取得部と、

前記復号鍵生成部が生成した要素 k_0^* 及び要素 k_i^* と、前記変数 $\rho(i)$ とを含む復号鍵 sk_s を取得する復号鍵取得部と、

前記データ取得部が取得した暗号化データ c に含まれる属性集合 Γ と、前記復号鍵取得部が取得した復号鍵 sk_s に含まれる変数 $\rho(i)$ とに基づき、 $i = 1, \dots, L$ の各整数 i のうち、変数 $\rho(i)$ が肯定形の組 (t, v_i) であり、かつ、その組の v_i と、その組の識別情報 t が示す Γ に含まれる x_t との内積が 0 となる i と、変数 $\rho(i)$ が否定形の組 $\neg(t, v_i)$ であり、かつ、その組の v_i と、その組の識別情報 t が示す Γ に含まれる x_t との内積が 0 とならない i との集合 I を特定するとともに、特定した集合 I に含まれる i について、 $\alpha_i s_i$ を合計した場合に s_0 となる補完係数 α_i を計算する補完係数計算部と、

20

前記暗号化データ c に含まれる要素 c_0 、要素 c_t と、前記復号鍵 sk_s に含まれる要素 k_0^* 、要素 k_i^* について、前記補完係数計算部が特定した集合 I と、前記補完係数計算部が計算した補完係数 α_i とに基づき、数 1 に示すペアリング演算を行い値 $K = g_{T^\zeta}^K$ を計算するペアリング演算部と

を備えることを特徴とする。

【数 1】

30

$$K := e(c_0, k_0^*) \cdot \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i} \cdot \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i / (\vec{v}_i \cdot \vec{x}_i)}$$

【発明の効果】

【0006】

この発明に係る暗号処理システムは、スパンプログラム (Span Program) と内積述語とを用いて関数型暗号を実現しており、多機能な暗号機能を実現している。また、従来の関数型暗号方式に比べ、安全性が高い。

40

【図面の簡単な説明】

【0007】

【図 1】行列 $M^\wedge$ の説明図。

【図 2】行列 M_δ の説明図。

【図 3】 s_0 の説明図。

【図 4】 $s \rightarrow^T$ の説明図。

【図 5】Key-Policy 関数型暗号方式を実行する暗号処理システム 10 の構成図。

【図 6】Key-Policy 関数型暗号方式を実行する暗号処理システム 10 の機能を示す機能ブロック図。

50

【図 7】 S e t u p アルゴリズムの処理を示すフローチャート。

【図 8】 K e y G e n アルゴリズムの処理を示すフローチャート。

【図 9】 E n c アルゴリズムの処理を示すフローチャート。

【図 10】 D e c アルゴリズムの処理を示すフローチャート。

【図 11】 C i p h e r t e x t - P o l i c y 関数型暗号方式のアルゴリズムを実行する暗号処理システム 10 の構成図。

【図 12】 C i p h e r t e x t - P o l i c y 関数型暗号方式を実行する暗号処理システム 10 の機能を示す機能ブロック図。

【図 13】 K e y G e n アルゴリズムの処理を示すフローチャート。

【図 14】 E n c アルゴリズムの処理を示すフローチャート。

10

【図 15】 D e c アルゴリズムの処理を示すフローチャート。

【図 16】 署名処理システム 20 の構成図。

【図 17】 署名処理システム 20 の機能を示す機能ブロック図。

【図 18】 S e t u p アルゴリズムの処理を示すフローチャート。

【図 19】 K e y G e n アルゴリズムの処理を示すフローチャート。

【図 20】 E n c アルゴリズムの処理を示すフローチャート。

【図 21】 D e c アルゴリズムの処理を示すフローチャート。

【図 22】 鍵生成装置 100、暗号化装置 200、復号装置 300、署名装置 400、検証装置 500 のハードウェア構成の一例を示す図。

【発明を実施するための形態】

20

【0008】

以下、図に基づき、発明の実施の形態を説明する。

以下の説明において、処理装置は後述する C P U 911 等である。記憶装置は後述する R O M 913、R A M 914、磁気ディスク 920 等である。通信装置は後述する通信ボード 915 等である。入力装置は後述するキーボード 902、通信ボード 915 等である。つまり、処理装置、記憶装置、通信装置、入力装置はハードウェアである。

【0009】

以下の説明における記法について説明する。

A がランダムな変数または分布であるとき、数 101 は、A の分布に従い A から y をランダムに選択することを表す。つまり、数 101 において、y は乱数である。

30

【数 101】

$$y \xleftarrow{R} A$$

A が集合であるとき、数 102 は、A から y を一様に選択することを表す。つまり、数 102 において、y は一様乱数である。

【数 102】

$$y \xleftarrow{U} A$$

40

数 103 は、y が z により定義された集合であること、又は y が z を代入された集合であることを表す。

【数 103】

$$y := z$$

a が定数であるとき、数 104 は、機械（アルゴリズム）A が入力 x に対し a を出力することを表す。

【数 1 0 4】

$$A(x) \rightarrow a$$

例えば、

$$A(x) \rightarrow 1$$

数 1 0 5、つまり F_q は、位数 q の有限体を示す。

【数 1 0 5】

$$\mathbb{F}_q$$

10

ベクトル表記は、有限体 F_q におけるベクトル表示を表す。つまり、数 1 0 6 である。

【数 1 0 6】

 $\vec{x}$ は、

$$(x_1, \dots, x_n) \in \mathbb{F}_q^n$$

を示す。

20

数 1 0 7 は、数 1 0 8 に示す 2 つのベクトル $\vec{x}$ と $\vec{v}$ との数 1 0 9 に示す内積を表す。

【数 1 0 7】

$$\vec{x} \cdot \vec{v}$$

【数 1 0 8】

$$\vec{x} = (x_1, \dots, x_n) ,$$

30

$$\vec{v} = (v_1, \dots, v_n)$$

【数 1 0 9】

$$\sum_{i=1}^n x_i v_i$$

 X^T は、行列 X の転置行列を表す。

b_i ($i = 1, \dots, n$) が空間 V のベクトルの要素であるとき、つまり、数 1 1 0 であるとき、数 1 1 1 は、数 1 1 2 によって生成される部分空間を表す。

40

【数 1 1 0】

$$b_i \in V \ (i = 1, \dots, n)$$

【数 1 1 1】

$$\text{span}\langle b_1, \dots, b_n \rangle \subseteq V \ (\text{resp. } \text{span}\langle \vec{x}_1, \dots, \vec{x}_n \rangle)$$

【数 1 1 2】

$$b_1, \dots, b_n \text{ (resp. } \vec{x}_1, \dots, \vec{x}_n \text{)}$$

数 1 1 3 に示す基底 B と基底 B^* とに対して、数 1 1 4 である。

【数 1 1 3】

$$\mathbb{B} := (b_1, \dots, b_N),$$

$$\mathbb{B}^* := (b_1^*, \dots, b_N^*)$$

10

【数 1 1 4】

$$(x_1, \dots, x_N)_{\mathbb{B}} := \sum_{i=1}^N x_i b_i,$$

$$(y_1, \dots, y_N)_{\mathbb{B}^*} := \sum_{i=1}^N y_i b_i^*$$

【0 0 1 0】

また、以下の説明において、“ n_t ” が下付き又は上付きで示されている場合、この n_t は n_t を意味する。同様に、“ V_t ” が下付き又は上付きで示されている場合、この V_t は V_t を意味する。同様に、“ $\delta_{i,j}$ ” が上付きで示されている場合、この $\delta_{i,j}$ は、 $\delta_{i,j}$ を意味する。

20

また、ベクトルを意味する“ $\rightarrow$ ” が下付き文字又は上付き文字に付されている場合、この“ $\rightarrow$ ” は下付き文字又は上付き文字に上付きで付されていることを意味する。

また、基底を表す B_0 、 B_{d+1} が下付きで示されている場合、この B_0 、 B_{d+1} はそれぞれ B_0 、 B_{d+1} を表す。同様に、基底を表す B^*_0 、 B^*_{d+1} が下付きで示されている場合、この B^*_0 、 B^*_{d+1} はそれぞれ B^*_0 、 B^*_{d+1} を表す。

【0 0 1 1】

また、以下の説明において、暗号処理とは、鍵生成処理、暗号化処理、復号処理を含むものであり、署名処理とは、鍵生成処理、署名処理、検証処理を含むものである。

30

【0 0 1 2】

実施の形態 1.

この実施の形態では、後の実施の形態で説明する「関数型暗号 (Functional Encryption) 方式」を実現する基礎となる概念と、関数型暗号の 1 つの構成について説明する。

第 1 に、関数型暗号について簡単に説明する。

第 2 に、関数型暗号を実現するための空間である「双対ペアリングベクトル空間 (Dual Pairing Vector Spaces, DPVS)」という豊かな数学的構造を有する空間を説明する。

40

第 3 に、関数型暗号を実現するための概念を説明する。ここでは、「スパンプログラム (Span Program)」、「属性ベクトルの内積とアクセスストラクチャ」、「秘密分散方式 (秘密共有方式)」について説明する。

第 4 に、この実施の形態に係る「関数型暗号方式」を説明する。この実施の形態では、「Key-Policy 関数型暗号 (Key-Policy Functional Encryption, KP-FE) 方式」について説明する。そこで、まず、「Key-Policy 関数型暗号方式」の基本構成について説明する。次に、この「Key-Policy 関数型暗号方式」を実現する「暗号処理システム 10」の基本構成について説明する。そして、この実施の形態に係る「Key-Policy 関数型暗号方式」及び「暗号処理システム 10」について詳細に説明する。

50

【0013】

<第1. 関数型暗号方式>

関数型暗号方式は、暗号化鍵 (e n c r y p t i o n - k e y, e k) と、復号鍵 (d e c r y p t i o n - k e y, d k) との間の関係をより高度化し、より柔軟にした暗号方式である。

関数型暗号方式において、暗号化鍵と復号鍵とは、それぞれ、属性 x と属性 v とが設定されている。そして、関係 R に対して $R(x, v)$ が成立する場合に限り、復号鍵 $dk_v := (dk, v)$ は暗号化鍵 $ek_x := (ek, x)$ で暗号化された暗号文を復号することができる。

関数型暗号方式には、データベースのアクセスコントロール、メールサービス、コンテンツ配布等の様々なアプリケーションが存在する（非特許文献2, 7, 9, 16, 19, 23-26, 28参照）。

10

【0014】

R が等号関係である場合、つまり、 $x = v$ である場合に限り $R(x, v)$ が成立する場合、関数型暗号方式はIDベース暗号方式である。

【0015】

IDベース暗号方式よりも一般化された関数型暗号方式として、属性ベース暗号方式がある。

属性ベース暗号方式では、暗号化鍵と復号鍵とに設定される属性が属性の組である。例えば、暗号化鍵と復号鍵とに設定される属性が、それぞれ、 $X := (x_1, \dots, x_d)$ と、 $V := (v_1, \dots, v_d)$ とである。

20

そして、属性のコンポーネントについて、コンポーネント毎の等号関係（例えば、 $\{x_t = v_t\}_{t \in \{1, \dots, d\}}$ ）がアクセスストラクチャ S に入力される。そして、アクセスストラクチャ S が入力を受理した場合にのみ、 $R(X, V)$ が成立する。つまり、暗号化鍵で暗号化された暗号文を復号鍵で復号することができる。

アクセスストラクチャ S が復号鍵 dk_v に埋め込まれている場合、属性ベース暗号 (ABE) 方式は、Key-Policy ABE (KP-ABE) と呼ばれる。一方、アクセスストラクチャ S が暗号文に埋め込まれている場合、属性ベース暗号 (ABE) 方式は、Ciphertext-Policy ABE (CP-ABE) と呼ばれる。

【0016】

30

非特許文献19に記載された内積述語暗号 (Inner-Product Encryption, IPE) も関数型暗号の1つのクラスである。ここでは、暗号化鍵と復号鍵とに設定される属性がそれぞれ体又は環上のベクトルである。例えば、 $x^{\rightarrow} := (x_1, \dots, x_n) \in F_q^n$ と $v^{\rightarrow} := (v_1, \dots, v_n) \in F_q^n$ とがそれぞれ暗号化鍵と復号鍵とに設定される。そして、 $x^{\rightarrow} \cdot v^{\rightarrow} = 0$ である場合に限り、 $R(x^{\rightarrow}, v^{\rightarrow})$ が成立する。

【0017】

<第2. 双対ペアリングベクトル空間>

まず、対象双線形ペアリング群 (Symmetric Bilinear Pairing Groups) について説明する。

40

対象双線形ペアリング群 (q, G, G^T, g, e) は、素数 q 、位数 q の巡回加法群 G と位数 q の巡回乗法群 G^T と、 $g \neq 0 \in G$ と、多項式時間で計算可能な非退化双線形ペアリング (Nondegenerate Bilinear Pairing) $e: G \times G \rightarrow G^T$ との組である。非退化双線形ペアリングは、 $e(sg, tg) = e(g, g)^{s \cdot t}$ であり、 $e(g, g) \neq 1$ である。

以下の説明において、数115を、 1^λ を入力として、セキュリティパラメータを λ とする双線形ペアリング群のパラメータ $param_G := (q, G, G^T, g, e)$ の値を出力するアルゴリズムとする。

【数 1 1 5】

 $\mathcal{G}_{\text{bpg}}$

【0 0 1 8】

次に、双対ペアリングベクトル空間について説明する。

双対ペアリングベクトル空間 (q, V, G_T, A, e) は、対象双線形ペアリング群 $(\text{param}_G := (q, G, G^T, g, e))$ の直積によって構成することができる。双対ペアリングベクトル空間 (q, V, G_T, A, e) は、素数 q 、数 1 1 6 に示す F_q 上の N 次元ベクトル空間 V 、位数 q の巡回群 G_T 、空間 V の標準基底 $A := (a_1, \dots, a_N)$ の組であり、以下の演算 (1) (2) を有する。ここで、 a_i は、数 1 1 7 に示す通りである。

10

【数 1 1 6】

$$V := \overbrace{\mathbb{G} \times \dots \times \mathbb{G}}^N$$

【数 1 1 7】

$$a_i := (\overbrace{0, \dots, 0}^{i-1}, g, \overbrace{0, \dots, 0}^{N-i})$$

20

【0 0 1 9】

演算 (1) : 非退化双線形ペアリング

空間 V におけるペアリングは、数 1 1 8 によって定義される。

【数 1 1 8】

$$e(x, y) := \prod_{i=1}^N e(G_i, H_i) \in \mathbb{G}_T$$

30

ここで、

$$(G_1, \dots, G_i) := x \in V,$$

$$(H_1, \dots, H_i) := y \in V$$

である。

これは、非退化双線形である。つまり、 $e(sx, ty) = e(x, y)^{st}$ であり、全ての $y \in V$ に対して、 $e(x, y) = 1$ の場合、 $x = 0$ である。また、全ての i と j に対して、 $e(a_i, a_j) = e(g, g)^{\delta_{i,j}}$ である。ここで、 $i = j$ であれば、 $\delta_{i,j} = 1$ であり、 $i \neq j$ であれば、 $\delta_{i,j} = 0$ である。また、 $e(g, g) \neq 1 \in \mathbb{G}_T$ である。

40

【0 0 2 0】

演算 (2) : ディストーション写像

数 1 1 9 に示す空間 V における線形変換 $\phi_{i,j}$ は、数 1 2 0 を行うことができる。

【数 1 1 9】

$$\phi_{i,j}(a_j) = a_i \text{ であり、}$$

$$k \neq j \text{ なら、 } \phi_{i,j}(a_k) = 0 \text{ である。}$$

【数 1 2 0】

$$\phi_{i,j}(x) := (0, \dots, 0, \overbrace{g_j}^{i-1}, \overbrace{0, \dots, 0}^{N-i})$$

10

ここで、

$$(g_1, \dots, g_N) := x$$

である。

ここで、線形変換 $\phi_{i,j}$ をディストーション写像と呼ぶ。

20

【0 0 2 1】

以下の説明において、数 1 2 1 を、 1^λ ($\lambda \in \text{自然数}$)、 $N \in \text{自然数}$ 、双線形ペアリング群のパラメータ $\text{param}_G := (q, G, G_T, g, e)$ の値を入力として、セキュリティパラメータが λ であり、 N 次元の空間 V とする双対ペアリングベクトル空間のパラメータ $\text{param}_V := (q, V, G_T, A, e)$ の値を出力するアルゴリズムとする。

【数 1 2 1】

$$\mathcal{G}_{\text{dpvs}}$$

30

【0 0 2 2】

なお、ここでは、上述した対象双線形ペアリング群により、双対ペアリングベクトル空間を構成した場合について説明する。なお、非対称双線形ペアリング群により双対ペアリングベクトル空間を構成することも可能である。以下の説明を、非対称双線形ペアリング群により双対ペアリングベクトル空間を構成した場合に適用することは容易である。

【0 0 2 3】

< 第 3. 関数型暗号を実現するための概念 >

< 第 3-1. スパンプログラム >

図 1 は、行列 $M^\wedge$ の説明図である。

$\{p_1, \dots, p_n\}$ を変数の集合とする。 $M^\wedge := (M, \rho)$ は、ラベル付けされた行列である。ここで、行列 M は、 F_q 上の (L 行 $\times$ r 列) の行列である。また、 ρ は、行列 M の各列に付されたラベルであり、 $\{p_1, \dots, p_n, \neg p_1, \dots, \neg p_n\}$ のいずれか 1 つのリテラルへ対応付けられる。なお、 M の全ての行に付されたラベル ρ_i ($i = 1, \dots, L$) がいずれか 1 つのリテラルへ対応付けられる。つまり、 $\rho: \{1, \dots, L\} \rightarrow \{p_1, \dots, p_n, \neg p_1, \dots, \neg p_n\}$ である。

40

【0 0 2 4】

全ての入力列 $\delta \in \{0, 1\}^n$ に対して、行列 M の部分行列 M_δ は定義される。行列 M_δ は、入力列 δ によってラベル ρ に値 “1” が対応付けられた行列 M の行から構成される部分行列である。つまり、行列 M_δ は、 $\delta_i = 1$ であるような p_i に対応付けられた行列 M の行と、 $\delta_i = 0$ であるような $\neg p_i$ に対応付けられた行列 M の行とからなる部分行列

50

である。

図2は、行列 M_δ の説明図である。なお、図2では、 $n=7$ 、 $L=6$ 、 $r=5$ としている。つまり、変数の集合は、 $\{p_1, \dots, p_7\}$ であり、行列 M は(6行×5列)の行列である。また、図2において、ラベル ρ は、 ρ_1 が $\neg p_2$ に、 ρ_2 が p_1 に、 ρ_3 が p_4 に、 ρ_4 が $\neg p_5$ に、 ρ_5 が $\neg p_3$ に、 ρ_6 が p_5 にそれぞれ対応付けられているとする。

ここで、入力列 $\delta \in \{0, 1\}^7$ が、 $\delta_1=1$ 、 $\delta_2=0$ 、 $\delta_3=1$ 、 $\delta_4=0$ 、 $\delta_5=0$ 、 $\delta_6=1$ 、 $\delta_7=1$ であるとする。この場合、破線で囲んだリテラル(p_1 、 p_3 、 p_6 、 p_7 、 $\neg p_2$ 、 $\neg p_4$ 、 $\neg p_5$)に対応付けられている行列 M の行からなる部分行列が行列 M_δ である。つまり、行列 M の1行目(M_1)、2行目(M_2)、4行目(M_4)からなる部分行列が行列 M_δ である。

10

【0025】

言い替えると、写像 $\gamma: \{1, \dots, L\} \rightarrow \{0, 1\}$ が、 $[\rho(j)=p_i] \wedge [\delta_i=1]$ 又は $[\rho(j)=\neg p_i] \wedge [\delta_i=0]$ である場合、 $\gamma(j)=1$ であり、他の場合、 $\gamma(j)=0$ であるとする。この場合に、 $M_\delta := (M_j)_{\gamma(j)=1}$ である。ここで、 M_j は、行列 M の j 番目の行である。

つまり、図2では、写像 $\gamma(j)=1$ ($j=1, 2, 4$)であり、写像 $\gamma(j)=0$ ($j=3, 5, 6$)である。したがって、 $(M_j)_{\gamma(j)=1}$ は、 M_1 、 M_2 、 M_4 であり、行列 M_δ である。

すなわち、写像 $\gamma(j)$ の値が“0”であるか“1”であるかによって、行列 M の j 番目の行が行列 M_δ に含まれるか否かが決定される。

20

【0026】

$1^\rightarrow \in \text{span}\langle M_\delta \rangle$ である場合に限り、スパンプログラム $M^\wedge$ は入力列 δ を受理し、他の場合には入力列 δ を拒絶する。つまり、入力列 δ によって行列 $M^\wedge$ から得られる行列 M_δ の行を線形結合して $1^\rightarrow$ が得られる場合に限り、スパンプログラム $M^\wedge$ は入力列 δ を受理する。なお、 $1^\rightarrow$ とは、各要素が値“1”である行ベクトルである。

例えば、図2の例であれば、行列 M の1、2、4行目からなる行列 M_δ の各行を線形結合して $1^\rightarrow$ が得られる場合に限り、スパンプログラム $M^\wedge$ は入力列 δ を受理する。つまり、 $\alpha_1(M_1) + \alpha_2(M_2) + \alpha_4(M_4) = 1^\rightarrow$ となる α_1 、 α_2 、 α_4 が存在する場合には、スパンプログラム $M^\wedge$ は入力列 δ を受理する。

30

【0027】

ここで、ラベル ρ が正のリテラル $\{p_1, \dots, p_n\}$ にのみ対応付けられている場合、スパンプログラムはモノトーンと呼ばれる。一方、ラベル ρ がリテラル $\{p_1, \dots, p_n, \neg p_1, \dots, \neg p_n\}$ に対応付けられている場合、スパンプログラムはノンモノトーンと呼ばれる。ここでは、スパンプログラムはノンモノトーンとする。そして、ノンモノトーンスパンプログラムを用いて、アクセスストラクチャ(ノンモノトーンアクセスストラクチャ)を構成する。アクセスストラクチャとは、簡単に言うと暗号へのアクセス制御を行うものである。つまり、暗号文を復号できるか否かの制御を行うものである。

詳しくは後述するが、スパンプログラムがモノトーンではなく、ノンモノトーンであることにより、スパンプログラムを利用して構成する関数型暗号方式の利用範囲が広がる。

40

【0028】

<第3-2. 属性ベクトルの内積とアクセスストラクチャ>

ここでは、属性ベクトルの内積を用いて上述した写像 $\gamma(j)$ を計算する。つまり、属性ベクトルの内積を用いて、行列 M のどの行を行列 M_δ に含めるかを決定する。

【0029】

U_t ($t=1, \dots, d$ であり $U_t \subset \{0, 1\}^*$)は、部分全集合(sub-universe)であり、属性の集合である。そして、 U_t は、それぞれ部分全集合の識別情報(t)と、 n_t 次元ベクトル($v^\rightarrow$)とを含む。つまり、 U_t は、($t, v^\rightarrow$)である。ここで、 $t \in \{1, \dots, d\}$ であり、 $v^\rightarrow \in F_q^{n_t}$ である。

50

【0030】

$U_t := (t, v^{\rightarrow})$ をスパンプログラム $M^{\wedge} := (M, \rho)$ における変数 p とする。つまり、 $p := (t, v^{\rightarrow})$ である。そして、変数 $(p := (t, v^{\rightarrow}), (t', v'^{\rightarrow}), \dots)$ としたスパンプログラム $M^{\wedge} := (M, \rho)$ をアクセスストラクチャ S とする。

つまり、アクセスストラクチャ $S := (M, \rho)$ であり、 $\rho : \{1, \dots, L\} \rightarrow \{(t, v^{\rightarrow}), (t', v'^{\rightarrow}), \dots, \neg(t, v^{\rightarrow}), \neg(t', v'^{\rightarrow}), \dots\}$ である。

【0031】

次に、 Γ を属性の集合とする。つまり、 $\Gamma := \{(t, x^{\rightarrow}_t) \mid x^{\rightarrow}_t \in \mathbb{F}_q^{n_t}, 1 \leq t \leq d\}$ である。 10

アクセスストラクチャ S に Γ が与えられた場合、スパンプログラム $M^{\wedge} := (M, \rho)$ に対する写像 $\gamma : \{1, \dots, L\} \rightarrow \{0, 1\}$ は、以下のように定義される。 $i = 1, \dots, L$ の各整数 i について、 $[\rho(i) = (t, v^{\rightarrow}_i)] \wedge [(t, x^{\rightarrow}_t) \in \Gamma] \wedge [v^{\rightarrow}_i \cdot x^{\rightarrow}_t = 0]$ 、又は、 $[\rho(i) = \neg(t, v^{\rightarrow}_i)] \wedge [(t, x^{\rightarrow}_t) \in \Gamma] \wedge [v^{\rightarrow}_i \cdot x^{\rightarrow}_t \neq 0]$ である場合、 $\gamma(j) = 1$ であり、他の場合、 $\gamma(j) = 0$ とする。

つまり、属性ベクトル $v^{\rightarrow}$ と $x^{\rightarrow}$ との内積に基づき、写像 γ が計算される。そして、上述したように、写像 γ により、行列 M のどの行を行列 M_{δ} に含めるかが決定される。すなわち、属性ベクトル $v^{\rightarrow}$ と $x^{\rightarrow}$ との内積により、行列 M のどの行を行列 M_{δ} に含めるかが決定され、 $1^{\rightarrow} \in \text{span} \langle (M_i)_{\gamma(i)=1} \rangle$ である場合に限り、アクセスストラクチャ $S := (M, \rho)$ は Γ を受理する。 20

【0032】

<第3-3. 秘密分散方式>

アクセスストラクチャ $S := (M, \rho)$ に対する秘密分散方式について説明する。

なお、秘密分散方式とは、秘密情報を分散させ、意味のない分散情報にすることである。例えば、秘密情報 s を 10 個に分散させ、10 個の分散情報を生成する。ここで、10 個の分散情報それぞれは、秘密情報 s の情報を有していない。したがって、ある 1 個の分散情報を手に入れても秘密情報 s に関して何ら情報を得ることはできない。一方、10 個の分散情報を全て手に入れば、秘密情報 s を復元できる。 30

また、10 個の分散情報を全て手に入れなくても、一部だけ（例えば、8 個）手に入れば秘密情報 s を復元できる秘密分散方式もある。このように、10 個の分散情報のうち 8 個で秘密情報 s を復元できる場合を、8-out-of-10 と呼ぶ。つまり、 n 個の分散情報のうち t 個で秘密情報 s を復元できる場合を、 t -out-of- n と呼ぶ。この t を閾値と呼ぶ。

また、 $d_1, \dots, d_{10}$ の 10 個の分散情報を生成した場合に、 $d_1, \dots, d_8$ までの 8 個の分散情報であれば秘密情報 s を復元できるが、 $d_3, \dots, d_{10}$ までの 8 個の分散情報であれば秘密情報 s を復元できないというような秘密分散方式もある。つまり、手に入れた分散情報の数だけでなく、分散情報の組合せに応じて秘密情報 s を復元できるか否かを制御する秘密分散方式もある。 40

【0033】

図 3 は、 s_0 の説明図である。図 4 は、 $s^{\rightarrow T}$ の説明図である。

行列 M を $(L \text{ 行} \times r \text{ 列})$ の行列とする。 $f^{\rightarrow T}$ を数 122 に示す列ベクトルとする。

【数 122】

$$\overrightarrow{f}^T := (f_1, \dots, f_r)^T \xleftarrow{U} \mathbb{F}_q^r$$

数 123 に示す s_0 を共有される秘密情報とする。

【数 1 2 3】

$$s_0 := \vec{1} \cdot \vec{f}^T := \sum_{k=1}^r f_k$$

また、数 1 2 4 に示す $s \rightarrow^T$ を s_0 の L 個の分散情報のベクトルとする。

【数 1 2 4】

$$\vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T$$

10

そして、分散情報 s_i を $\rho(i)$ に属するものとする。

【0 0 3 4】

アクセスストラクチャ $S := (M, \rho)$ が Γ を受理する場合、つまり $\gamma : \{1, \dots, L\} \rightarrow \{0, 1\}$ について $1 \rightarrow \in \text{span} \langle (M_i)_{\gamma(i)=1} \rangle$ である場合、 $I \subseteq \{i \in \{1, \dots, L\} \mid \gamma(i) = 1\}$ である定数 $\{\alpha_i \in F_q \mid i \in I\}$ が存在する。

これは、図 2 の例で、 $\alpha_1 (M_1) + \alpha_2 (M_2) + \alpha_4 (M_4) = 1 \rightarrow$ となる $\alpha_1, \alpha_2, \alpha_4$ が存在する場合には、スパンプログラム $M^\wedge$ は入力列 δ を受理すると説明したことからも明らかである。つまり、 $\alpha_1 (M_1) + \alpha_2 (M_2) + \alpha_4 (M_4) = 1 \rightarrow$ となる $\alpha_1, \alpha_2, \alpha_4$ が存在する場合には、スパンプログラム $M^\wedge$ は入力列 δ を受理するのであれば、スパンプログラム $M^\wedge$ は入力列 δ を受理する場合には、 $\alpha_1 (M_1) + \alpha_2 (M_2) + \alpha_4 (M_4) = 1 \rightarrow$ となる $\alpha_1, \alpha_2, \alpha_4$ が存在する。

20

そして、数 1 2 5 である。

【数 1 2 5】

$$\sum_{i \in I} \alpha_i s_i := s_0$$

30

なお、定数 $\{\alpha_i\}$ は、行列 M のサイズにおける多項式時間で計算可能である。

【0 0 3 5】

この実施の形態及び以下の実施の形態に係る関数型暗号方式は、上述したように、スパンプログラムに内積述語と秘密分散方式とを適用してアクセスストラクチャを構成する。そのため、スパンプログラムにおける行列 M や、内積述語における属性情報 x 及び属性情報 v (述語情報) を設計するにより、アクセス制御を自由に設計することができる。つまり、非常に高い自由度でアクセス制御の設計を行うことができる。なお、行列 M の設計は、秘密分散方式の閾値等の条件設計に相当する。

例えば、上述した属性ベース暗号方式は、この実施の形態及び以下の実施の形態に係る関数型暗号方式におけるアクセスストラクチャにおいて、内積述語の設計をある条件に限定した場合に相当する。つまり、この実施の形態及び以下の実施の形態に係る関数型暗号方式におけるアクセスストラクチャに比べ、属性ベース暗号方式におけるアクセスストラクチャは、内積述語における属性情報 x 及び属性情報 v (述語情報) を設計の自由度がない分、アクセス制御の設計の自由度が低い。なお、具体的には、属性ベース暗号方式は、属性情報 $\{x \rightarrow_t\}_{t \in \{1, \dots, d\}}$ と $\{v \rightarrow_t\}_{t \in \{1, \dots, d\}}$ とを、等号関係に対する 2 次元ベクトル、例えば $x \rightarrow_t := (1, x_t)$ と $v \rightarrow_t := (v_t, -1)$ とに限定した場合に相当する。

40

また、上述した内積述語暗号方式は、この実施の形態及び以下の実施の形態に係る関数型暗号方式におけるアクセスストラクチャにおいて、スパンプログラムにおける行列 M の設計をある条件に限定した場合に相当する。つまり、この実施の形態及び以下の実施の形

50

態に係る関数型暗号方式におけるアクセスストラクチャに比べ、内積述語暗号方式におけるアクセスストラクチャは、スパンプログラムにおける行列Mの設計の自由度がない分、アクセス制御の設計の自由度が低い。なお、具体的には、内積述語暗号方式は、秘密分散方式を $1-out-of-1$ （あるいは、 $d-out-of-d$ ）に限定した場合である。

【0036】

特に、この実施の形態及び以下の実施の形態に係る関数型暗号方式におけるアクセスストラクチャは、ノンモノトーンスパンプログラムを用いたノンモノトーンアクセスストラクチャを構成する。そのため、アクセス制御の設計の自由度がより高くなる。

具体的には、ノンモノトーンスパンプログラムには、否定形のリテラル ($\neg p$) を含むため、否定形の条件を設定できる。例えば、第1会社には、A部とB部とC部とD部との4つの部署があったとする。ここで、第1会社のB部以外の部署の属するユーザにのみアクセス可能（復号可能）というアクセス制御をしたいとする。この場合に、否定形の条件の設定ができないとすると、「第1会社のA部とC部とD部とのいずれかに属すること」という条件を設定する必要がある。一方、否定形の条件の設定ができるとすると、「第1会社の社員であって、B部以外に属すること」という条件を設定することができる。つまり、否定形の条件が設定できることで、自然な条件設定が可能となる。なお、ここでは部署の数が少ないが、部署の数が多い場合等は非常に有効であることが分かる。

【0037】

<第4．関数型暗号方式の基本構成>

<第4-1．Key-Policy関数型暗号方式の基本構成>

Key-Policy関数型暗号方式の構成を簡単に説明する。なお、Key-Policyとは、復号鍵にPolicyが埋め込まれること、つまりアクセスストラクチャが埋め込まれることを意味する。

Key-Policy関数型暗号方式は、Setup、KeyGen、Enc、Decの4つのアルゴリズムを備える。

(Setup)

Setupアルゴリズムは、セキュリティパラメータ λ と、属性のフォーマット $\mu \rightarrow : = (d; n_1, \dots, n_d)$ とが入力され、公開パラメータpkと、マスター鍵skとを出力する確率的アルゴリズムである。

(KeyGen)

KeyGenアルゴリズムは、アクセスストラクチャ $S := (M, \rho)$ と、公開パラメータpkと、マスター鍵skとを入力として、復号鍵 sk_s を出力する確率的アルゴリズムである。

(Enc)

Encアルゴリズムは、メッセージmと、属性の集合である $\Gamma := \{(t, x \rightarrow_t) \mid x \rightarrow_t \in F_q^{n_t}, 1 \leq t \leq d\}$ と、公開パラメータpkとを入力として、暗号化データcを出力する確率的アルゴリズムである。

(Dec)

Decアルゴリズムは、属性の集合である Γ の下で暗号化された暗号化データcと、アクセスストラクチャSに対する復号鍵 sk_s と、公開パラメータpkとを入力として、メッセージm、又は、識別情報 $\perp$ を出力するアルゴリズムである。

【0038】

Key-Policy関数型暗号方式は、全てのアクセスストラクチャSと属性の集合 Γ と、正しく生成された公開パラメータpkとマスター鍵skと数126に示す暗号文cとに対して、アクセスストラクチャSが属性の集合 Γ を受理する場合、 $m = Dec(pk, sk_s, c)$ であり、アクセスストラクチャSが属性の集合 Γ を拒絶する場合、 $m = Dec(pk, sk_s, c)$ となる確率は無視し得る程度である。

【数 1 2 6】

$$c \xleftarrow{R} \text{Enc}(\text{pk}, m, \Gamma)$$

【0039】

＜第 4-2. 暗号処理システム 10＞

上述した Key-Policy 関数型暗号方式のアルゴリズムを実行する暗号処理システム 10 について説明する。

図 5 は、Key-Policy 関数型暗号方式を実行する暗号処理システム 10 の構成図である。 10

暗号処理システム 10 は、鍵生成装置 100、暗号化装置 200、復号装置 300 を備える。

鍵生成装置 100 は、セキュリティパラメータ λ と、属性のフォーマット $\mu \rightarrow := (d; n_1, \dots, n_d)$ とを入力として Setup アルゴリズムを実行して、公開パラメータ pk とマスター鍵 sk とを生成する。そして、鍵生成装置 100 は、生成した公開パラメータ pk を公開する。また、鍵生成装置 100 は、アクセスストラクチャ S を入力として KeyGen アルゴリズムを実行して、復号鍵 sk_S を生成して復号装置 300 へ秘密裡に配布する。

暗号化装置 200 は、メッセージ m と、属性の集合 Γ と、公開パラメータ pk とを入力として Enc アルゴリズムを実行して、暗号化データ c を生成する。暗号化装置 200 は、生成した暗号化データ c を復号装置 300 へ送信する。 20

復号装置 300 は、公開パラメータ pk と、復号鍵 sk_S と、暗号化データ c とを入力として Dec アルゴリズムを実行して、メッセージ m 又は識別情報 $\perp$ を出力する。

【0040】

＜第 4-3. Key-Policy 関数型暗号方式及び暗号処理システム 10 の詳細＞

図 6 から図 9 に基づき、Key-Policy 関数型暗号方式、及び、Key-Policy 関数型暗号方式を実行する暗号処理システム 10 の機能と動作とについて説明する。

図 6 は、Key-Policy 関数型暗号方式を実行する暗号処理システム 10 の機能を示す機能ブロック図である。暗号処理システム 10 は、上述したように、鍵生成装置 100、暗号化装置 200、復号装置 300 を備える。 30

図 7 と図 8 とは、鍵生成装置 100 の動作を示すフローチャートである。なお、図 7 は Setup アルゴリズムの処理を示すフローチャートであり、図 8 は KeyGen アルゴリズムの処理を示すフローチャートである。図 9 は、暗号化装置 200 の動作を示すフローチャートであり、Enc アルゴリズムの処理を示すフローチャートである。図 10 は、復号装置 300 の動作を示すフローチャートであり、Dec アルゴリズムの処理を示すフローチャートである。

なお、以下の説明では、 $x_{t, 1} := 1$ であるとする。

【0041】

鍵生成装置 100 の機能と動作とについて説明する。鍵生成装置 100 は、マスター鍵生成部 110、マスター鍵記憶部 120、情報入力部 130（第 1 情報入力部）、復号鍵生成部 140、鍵配布部 150 を備える。また、復号鍵生成部 140 は、f ベクトル生成部 141、s ベクトル生成部 142、乱数生成部 143、鍵要素生成部 144 を備える。

【0042】

まず、図 7 に基づき、Setup アルゴリズムの処理について説明する。

（S101：正規直交基底生成ステップ）

マスター鍵生成部 110 は、処理装置により、数 127 を計算して、 $\text{param}_{\mu \rightarrow}$ と、 $t = 0, \dots, d$ の各整数 t について基底 B_t 及び基底 B^*_t とを生成する。

40

【数 1 2 7】

(1)

input $1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d)$

(2)

 $\text{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, g, e) \xleftarrow{R} \mathcal{G}_{\text{bpg}}(1^\lambda)$

(3)

 $\psi \xleftarrow{U} \mathbb{F}_q^\times, N_0 := 5, N_t := 4n_t \text{ for } t = 1, \dots, d$

10

 $t = 0, \dots, d$ の各 t について (4) から (8) の処理を実行する。

(4)

 $\text{param}_{\mathbb{V}_t} := (q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, e) := \mathcal{G}_{\text{dpvs}}(1^\lambda, N_t, \text{param}_{\mathbb{G}})$

(5)

 $X_t := (\chi_{t,i,j})_{i,j} \xleftarrow{U} GL(N_t, \mathbb{F}_q)$

20

(6)

 $(v_{t,i,j})_{i,j} := \psi \cdot (X_t^T)^{-1}$

(7)

 $b_{t,i} := (\chi_{t,i,1}, \dots, \chi_{t,i,N_t})_{\mathbb{A}_t} = \sum_{j=1}^{N_t} \chi_{t,i,j} a_{t,j},$ $\mathbb{B}_t := (b_{t,1}, \dots, b_{t,N_t})$

30

(8)

 $b_{t,i}^* := (v_{t,i,1}, \dots, v_{t,i,N_t})_{\mathbb{A}_t} = \sum_{j=1}^{N_t} v_{t,i,j} a_{t,j},$ $\mathbb{B}_t^* := (b_{t,1}^*, \dots, b_{t,N_t}^*)$

(9)

 $g_T := e(g, g)^\psi,$

40

 $\text{param}_{\vec{\mu}} := (\{\text{param}_{\mathbb{V}_t}\}_{t=0, \dots, d}, g_T)$

【0 0 4 3】

つまり、マスター鍵生成部 1 1 0 は以下の処理を実行する。

(1) マスター鍵生成部 1 1 0 は、入力装置により、セキュリティパラメータ λ (1^λ) と、属性のフォーマット $\vec{\mu} := (d; n_1, \dots, n_d)$ とを入力する。ここで、 d は 1 以上の整数であり、 $t = 1, \dots, d$ までの各整数 t について n_t は 1 以上の整数である。

50

(2) マスター鍵生成部 110 は、処理装置により、(1) で入力したセキュリティパラメータ $\lambda(1^\lambda)$ を入力としてアルゴリズム G_{bpg} を実行して、双線形ペアリング群のパラメータ $param_G := (q, G, G_T, g, e)$ の値を生成する。

(3) マスター鍵生成部 110 は、処理装置により、乱数 ψ を生成するとともに、 N_0 に 5 を、 $t = 1, \dots, d$ の各整数 t について N_t に $4n_t$ を設定する。

【0044】

続いて、マスター鍵生成部 110 は、 $t = 0, \dots, d$ の各整数 t について以下の (4) から (8) までの処理を実行する。

(4) マスター鍵生成部 110 は、処理装置により、(1) で入力したセキュリティパラメータ $\lambda(1^\lambda)$ と、(3) で設定した N_t と、(2) で生成した $param_G := (q, G, G_T, g, e)$ の値とを入力としてアルゴリズム G_{dpvs} を実行して、双対ペアリングベクトル空間のパラメータ $param_{V_t} := (q, V_T, G_T, A, e)$ の値を生成する。

10

(5) マスター鍵生成部 110 は、処理装置により、(3) で設定した N_t と、 F_q とを入力として、線形変換 $X_t := (\chi_{t,i,j})_{i,j}$ をランダムに生成する。なお、GL は、General Linear の略である。つまり、GL は、一般線形群であり、行列式が 0 でない正方行列の集合であり、乗法に関し群である。また、 $(\chi_{t,i,j})_{i,j}$ は、行列 $\chi_{t,i,j}$ の添え字 i, j に関する行列という意味であり、ここでは、 $i, j = 1, \dots, n_t$ である。

(6) マスター鍵生成部 110 は、処理装置により、乱数 ψ と線形変換 X_t とに基づき、 $(v_{t,i,j})_{i,j} := \psi \cdot (X_t^T)^{-1}$ を生成する。なお、 $(v_{t,i,j})_{i,j}$ も $(\chi_{t,i,j})_{i,j}$ と同様に、行列 $v_{t,i,j}$ の添え字 i, j に関する行列という意味であり、ここでは、 $i, j = 1, \dots, n_t$ である。

20

(7) マスター鍵生成部 110 は、処理装置により、(5) で生成した線形変換 X_t に基づき、(4) で生成した標準基底 A_t から基底 B_t を生成する。

(8) マスター鍵生成部 110 は、処理装置により、(6) で生成した $(v_{t,i,j})_{i,j}$ に基づき、(4) で生成した標準基底 A_t から基底 B^*_t を生成する。

(9) マスター鍵生成部 110 は、処理装置により、 g_T に $e(g, g)^\psi$ を設定する。また、マスター鍵生成部 110 は、 $param_{\mu \rightarrow}$ に (4) で生成した $\{param_{V_t}\}_{t=0, \dots, d}$ と、 g_T とを設定する。なお、 $t = 0, \dots, d$ と $i = 1, \dots, N_t$ との各整数 t, i について、 $g_T = e(b_{t,i}, b^*_{t,i})$ である。

30

【0045】

すなわち、(S101) で、マスター鍵生成部 110 は、数 128 に示すアルゴリズム G_b を実行して、 $param_{\mu \rightarrow}$ と、 $t = 0, \dots, d$ の各整数 t について基底 B_t 及び基底 B^*_t とを生成する。

【数 1 2 8】

$$\begin{aligned}
& \mathcal{G}_{\text{ob}}(1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d)) : \text{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, g, e) \xleftarrow{\mathbb{R}} \mathcal{G}_{\text{bpg}}(1^\lambda), \\
& \psi \xleftarrow{\mathbb{U}} \mathbb{F}_q^\times, \\
& N_0 := 5, \quad N_t := 4n_t \quad \text{for } t=1, \dots, d, \\
& \text{For } t=0, \dots, d, \quad \text{param}_{\mathbb{V}_t} := (q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, e) := \mathcal{G}_{\text{dpvs}}(1^\lambda, N_t, \text{param}_{\mathbb{G}}), \\
& X_t := (\chi_{t,i,j})_{i,j} \xleftarrow{\mathbb{U}} GL(N_t, \mathbb{F}_q), \quad (v_{t,i,j})_{i,j} := \psi \cdot (X_t^T)^{-1}, \\
& b_{t,i} := (\chi_{t,i,1}, \dots, \chi_{t,i,N_t})_{\mathbb{A}_t} = \sum_{j=1}^{N_t} \chi_{t,i,j} a_{t,j}, \quad \mathbb{B}_t := (b_{t,1}, \dots, b_{t,N_t}), \\
& b_{t,i}^* := (v_{t,i,1}, \dots, v_{t,i,N_t})_{\mathbb{A}_t} = \sum_{j=1}^{N_t} v_{t,i,j} a_{t,j}, \quad \mathbb{B}_t^* := (b_{t,1}^*, \dots, b_{t,N_t}^*), \\
& g_T := e(g, g)^\psi, \quad \text{param}_{\vec{\mu}} := (\{\text{param}_{\mathbb{V}_t}\}_{t=0, \dots, d}, g_T) \\
& \text{return } (\text{param}_{\vec{\mu}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0, \dots, d}).
\end{aligned}$$

【0 0 4 6】

(S 1 0 2 : 公開パラメータ生成ステップ)

マスター鍵生成部 1 1 0 は、処理装置により、(S 1 0 1) で生成した基底 B_t の部分基底 $B^{\wedge}_0$ と、 $t = 1, \dots, d$ の各整数 t について、部分基底 $B^{\wedge}_t$ とを数 1 2 9 に示すように生成する。

【数 1 2 9】

$$\begin{aligned}
& \hat{\mathbb{B}}_0 := (b_{0,1}, b_{0,3}, b_{0,5}), \\
& \hat{\mathbb{B}}_t := (b_{t,1}, \dots, b_{t,n_t}, b_{t,3n_t+1}, \dots, b_{t,4n_t}) \\
& \quad \text{for } t=1, \dots, d
\end{aligned}$$

マスター鍵生成部 1 1 0 は、生成した部分基底 $B^{\wedge}_0$ 及び部分基底 $B^{\wedge}_t$ と、(S 1 0 1) で入力されたセキュリティパラメータ $\lambda(1^\lambda)$ と、(S 1 0 1) で生成した $\text{param}_{\vec{\mu}}$ とを合わせて、公開パラメータ pk とする。

【0 0 4 7】

(S 1 0 3 : マスター鍵生成ステップ)

マスター鍵生成部 1 1 0 は、処理装置により、(S 1 0 1) で生成した基底 B_t^* の部分基底 $B^{\wedge*}_0$ と、 $t = 1, \dots, d$ の各整数 t について、部分基底 $B^{\wedge*}_t$ とを数 1 3 0 に示すように生成する。

【数 1 3 0】

$$\begin{aligned}
& \hat{\mathbb{B}}_0^* := (b_{0,1}^*, b_{0,3}^*, b_{0,4}^*) \\
& \hat{\mathbb{B}}_t^* := (b_{t,1}^*, \dots, b_{t,n_t}^*, b_{t,2n_t+1}^*, \dots, b_{t,3n_t}^*) \\
& \quad \text{for } t=1, \dots, d,
\end{aligned}$$

マスター鍵生成部 1 1 0 は、生成した部分基底 $B^{\wedge*}_0$ と部分基底 $B^{\wedge*}_t$ とをマスター鍵 sk とする。

【0048】

(S104: マスター鍵記憶ステップ)

マスター鍵記憶部120は、(S102)で $t = 0, \dots, d$ の各整数 t について生成した部分基底 $B^*_{t, \dots}$ を含む公開パラメータ pk を記憶装置に記憶する。また、マスター鍵記憶部120は、(S103)で $t = 0, \dots, d$ の各整数 t について生成した部分基底 $B^*_{t, \dots}$ を含むマスター鍵 sk を記憶装置に記憶する。

【0049】

つまり、(S101)から(S103)において、マスター鍵生成部110は数131に示すSetupアルゴリズムを実行して、公開パラメータ pk とマスター鍵 sk とを生成する。そして、(S104)で、マスター鍵記憶部120が生成された公開パラメータ pk とマスター鍵 sk とを記憶装置に記憶する。

なお、公開パラメータは、例えば、ネットワークを介して公開され、暗号化装置200や復号装置300が取得可能な状態にされる。

【数131】

$\text{Setup}(1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d))$

$(\text{param}_{\vec{\mu}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0, \dots, d}) \xleftarrow{R} \mathcal{G}_{\text{ob}}(1^\lambda, \vec{\mu})$

$\hat{\mathbb{B}}_0 := (b_{0,1}, b_{0,3}, b_{0,5}), \hat{\mathbb{B}}_t := (b_{t,1}, \dots, b_{t,n_t}, b_{t,3n_t+1}, \dots, b_{t,4n_t})$

for $t = 1, \dots, d,$

$\hat{\mathbb{B}}_0^* := (b_{0,1}^*, b_{0,3}^*, b_{0,4}^*), \hat{\mathbb{B}}_t^* := (b_{t,1}^*, \dots, b_{t,n_t}^*, b_{t,2n_t+1}^*, \dots, b_{t,3n_t}^*)$

for $t = 1, \dots, d,$

$sk := \{\hat{\mathbb{B}}_t^*\}_{t=0, \dots, d}, pk := (1^\lambda, \text{param}_{\vec{\mu}}, \{\mathbb{B}_t\}_{t=0, \dots, d}).$

return $sk, pk.$

【0050】

次に、図8に基づき、KeyGenアルゴリズムの処理について説明する。

(S201: 情報入力ステップ)

情報入力部130は、入力装置により、上述したアクセスストラクチャ $S := (M, \rho)$ を入力する。なお、アクセスストラクチャ S の行列 M の設定については、実現したいシステムの条件に応じて設定されるものである。また、アクセスストラクチャ S の ρ は、例えば、復号鍵 sk_s の使用者の属性情報が設定されている。

【0051】

(S202: f ベクトル生成ステップ)

f ベクトル生成部141は、処理装置により、 r 個の要素を有するベクトル $f \rightarrow$ を数132に示すようにランダムに生成する。

【数132】

$\vec{f} \xleftarrow{U} \mathbb{F}_q^r$

【0052】

(S203: s ベクトル生成ステップ)

s ベクトル生成部142は、処理装置により、(S201)で入力したアクセスストラクチャ S に含まれる(L 行 $\times$ r 列)の行列 M と、(S202)で生成したベクトル $f \rightarrow$ と

10

20

30

40

50

に基づき、ベクトル $\mathbf{s}^{\rightarrow T} := (s_1, \dots, s_L)^T$ を数 1 3 3 に示すように生成する。

【数 1 3 3】

$$\vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T$$

また、 $\mathbf{s}$ ベクトル生成部 1 4 2 は、処理装置により、(S 2 0 2) で生成したベクトル $\mathbf{f}^{\rightarrow}$ に基づき、値 s_0 を数 1 3 4 に示すように生成する。

【数 1 3 4】

10

$$s_0 := \vec{1} \cdot \vec{f}^T$$

【0 0 5 3】

(S 2 0 4：乱数生成ステップ)

乱数生成部 1 4 3 は、処理装置により、乱数 η_0 と、 $i = 1, \dots, L$ の各整数 i について、乱数 θ_i とを数 1 3 5 に示すように生成する。

【数 1 3 5】

20

$$\eta_0, \theta_i \xleftarrow{U} \mathbb{F}_q (i = 1, \dots, L)$$

【0 0 5 4】

(S 2 0 5：鍵要素生成ステップ)

鍵要素生成部 1 4 4 は、処理装置により、復号鍵 $\mathbf{sk}_s$ の要素 k_0^* を数 1 3 6 に示すように生成する。

【数 1 3 6】

30

$$k_0^* := (-s_0, 0, 1, \eta_0, 0)_{\mathbb{B}_0^*}$$

なお、上述したように、数 1 1 3 に基底 $\mathbf{B}$ と基底 $\mathbf{B}^*$ とに対して、数 1 1 4 である。したがって、数 1 3 6 は、基底 $\mathbf{B}_0^*$ の基底ベクトル $\mathbf{b}_{0,1}^*$ の係数として $-s_0$ を設定し、基底ベクトル $\mathbf{b}_{0,2}^*$ の係数として 0 を設定し、基底ベクトル $\mathbf{b}_{0,3}^*$ の係数として 1 を設定し、基底ベクトル $\mathbf{b}_{0,4}^*$ の係数として η_0 を設定し、基底ベクトル $\mathbf{b}_{0,5}^*$ の係数として 0 を設定することを意味する。

また、鍵要素生成部 1 4 4 は、処理装置により、 $i = 1, \dots, L$ の各整数 i について、復号鍵 $\mathbf{sk}_s$ の要素 k_i^* を数 1 3 7 に示すように生成する。

40

【数 1 3 7】

$$\begin{aligned}
 &\text{if } \rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \quad \eta_{i,1}, \dots, \eta_{i,n_t} \xleftarrow{\mathcal{U}} \mathbb{F}_q \\
 &k_i^* := (\overbrace{s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_t}}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_t}}^{n_t}, \overbrace{0^{n_t}}^{n_t})_{\mathbb{B}_t^*} \\
 &\text{if } \rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \quad \eta_{i,1}, \dots, \eta_{i,n_t} \xleftarrow{\mathcal{U}} \mathbb{F}_q, \\
 &k_i^* := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_t})}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_t}}^{n_t}, \overbrace{0^{n_t}}^{n_t})_{\mathbb{B}_t^*}
 \end{aligned}$$

10

つまり、数 1 3 7 は、数 1 3 6 と同様に、 $\rho(i)$ が肯定形の組 $(t, \vec{v}_i)$ である場合には、基底 B_t^* の基底ベクトル $b_{t,1}^*$ の係数として $s_i + \theta_i v_{i,1}$ を設定し、基底ベクトル $b_{t,2}^*, \dots, b_{t,n_t}^*$ の係数として $\theta_i v_{i,2}, \dots, \theta_i v_{i,n_t}$ を設定し、基底ベクトル $b_{t,n_t+1}^*, \dots, b_{t,2n_t}^*$ の係数として 0 を設定し、基底ベクトル $b_{t,2n_t+1}^*, \dots, b_{t,3n_t}^*$ の係数として $\eta_{i,1}, \dots, \eta_{i,n_t}$ を設定し、基底ベクトル $b_{t,3n_t+1}^*, \dots, b_{t,4n_t}^*$ の係数として 0 を設定することを意味する。

20

一方、 $\rho(i)$ が否定形の組 $\neg(t, \vec{v}_i)$ である場合には、基底 B_t^* の基底ベクトル $b_{t,1}^*, \dots, b_{t,n_t}^*$ の係数として $s_i v_{i,1}, \dots, s_i v_{i,n_t}$ を設定し、基底ベクトル $b_{t,n_t+1}^*, \dots, b_{t,2n_t}^*$ の係数として 0 を設定し、基底ベクトル $b_{t,2n_t+1}^*, \dots, b_{t,3n_t}^*$ の係数として $\eta_{i,1}, \dots, \eta_{i,n_t}$ を設定し、基底ベクトル $b_{t,3n_t+1}^*, \dots, b_{t,4n_t}^*$ の係数として 0 を設定することを意味する。

【0 0 5 5】

(S 2 0 6 : 鍵配布ステップ)

鍵配布部 1 5 0 は、(S 2 0 1) で入力したアクセスストラクチャ S と、(S 2 0 5) で生成された $k_0^*, k_1^*, \dots, k_L^*$ とを要素とする復号鍵 sk_S を、例えば通信装置によりネットワークを介して秘密裡に復号装置 3 0 0 へ配布する。もちろん、復号鍵 sk_S は、他の方法により復号装置 3 0 0 へ配布されてもよい。

30

【0 0 5 6】

つまり、(S 2 0 1) から (S 2 0 5) において、情報入力部 1 3 0 と復号鍵生成部 1 4 0 とは数 1 3 8 に示す $KeyGen$ アルゴリズムを実行して、復号鍵 sk_S を生成する。そして、(S 2 0 6) で、鍵配布部 1 5 0 が生成された復号鍵 sk_S を復号装置 3 0 0 へ配布する。

【数 1 3 8】

KeyGen(pk, sk, S := (M, ρ))

$$\vec{f} \xleftarrow{\mathcal{U}} \mathbb{F}_q^r, \quad \vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T, \quad s_0 := \vec{1} \cdot \vec{f}^T,$$

$$\eta_0, \theta_i \xleftarrow{\mathcal{U}} \mathbb{F}_q \quad (i = 1, \dots, L),$$

$$k_0^* := (-s_0, 0, 1, \eta_0, 0)_{\mathbb{B}_0^*},$$

for $1 \leq i \leq L$

$$\text{if } \rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \quad \eta_{i,1}, \dots, \eta_{i,n_t} \xleftarrow{\mathcal{U}} \mathbb{F}_q,$$

$$k_i^* := (\overbrace{s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_t}}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_t}}^{n_t}, \overbrace{0^{n_t}}^{n_t})_{\mathbb{B}_i^*}$$

$$\text{if } \rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \quad \eta_{i,1}, \dots, \eta_{i,n_t} \xleftarrow{\mathcal{U}} \mathbb{F}_q,$$

$$k_i^* := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_t})}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_t}}^{n_t}, \overbrace{0^{n_t}}^{n_t})_{\mathbb{B}_i^*}$$

$$\text{sk}_S := (S, k_0^*, k_1^*, \dots, k_L^*).$$

return sk_S.

【0 0 5 7】

暗号化装置 2 0 0 の機能と動作とについて説明する。暗号化装置 2 0 0 は、公開パラメータ取得部 2 1 0、情報入力部 2 2 0（第 2 情報入力部）、暗号化データ生成部 2 3 0、データ送信部 2 4 0（データ出力部）を備える。また、暗号化データ生成部 2 3 0 は、乱数生成部 2 3 1、暗号要素生成部 2 3 2 を備える。

【0 0 5 8】

図 9 に基づき、E n c アルゴリズムの処理について説明する。

（S 3 0 1：公開パラメータ取得ステップ）

公開パラメータ取得部 2 1 0 は、例えば、通信装置によりネットワークを介して、鍵生成装置 1 0 0 が生成した公開パラメータ p k を取得する。

【0 0 5 9】

（S 3 0 2：情報入力ステップ）

情報入力部 2 2 0 は、入力装置により、復号装置 3 0 0 へ送信するメッセージ m を入力する。また、情報入力部 2 2 0 は、入力装置により、属性の集合 $\Gamma := \{ (t, \vec{x}_t : = (x_{t,1}, \dots, x_{t,n_t}) \in \mathbb{F}_q^{n_t}) \mid 1 \leq t \leq d \}$ を入力する。なお、t は、1 以上 d 以下の全ての整数ではなく、1 以上 d 以下の少なくとも一部の整数であってもよい。また、属性の集合 Γ は、例えば、復号可能なユーザの属性情報が設定されている。

【0 0 6 0】

（S 3 0 3：乱数生成ステップ）

乱数生成部 2 3 1 は、処理装置により、乱数 δ 、乱数 $\phi_0, \phi_{t,1}, \dots, \phi_{t,n_t}, \zeta$ を数 1 3 9 に示すように生成する。

10

20

30

40

【数 1 3 9】

$$\delta, \phi_0, \phi_{t,1}, \dots, \phi_{t,n_t}, \zeta \xleftarrow{\cup} \mathbb{F}_q \text{ such that } (t, \vec{x}_t) \in \Gamma$$

【0 0 6 1】

(S 3 0 4 : c ベクトル生成ステップ)

暗号要素生成部 2 3 2 は、処理装置により、暗号化データ c の要素 c_0 を数 1 4 0 に示すように生成する。

【数 1 4 0】

10

$$c_0 := (\delta, 0, \zeta, 0, \phi_0)_{\mathbb{B}_0}$$

また、暗号要素生成部 2 3 2 は、処理装置により、 Γ に含まれる $(t, \vec{x}_t)$ の各整数 t について、暗号化データ c の要素 c_t を数 1 4 1 に示すように生成する。

【数 1 4 1】

$$c_t := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}))}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{(\phi_{t,1}, \dots, \phi_{t,n_t})}^{n_t})_{\mathbb{B}_t}$$

20

また、暗号要素生成部 2 3 2 は、処理装置により、暗号化データ c の要素 c_{d+1} を数 1 4 2 に示すように生成する。

【数 1 4 2】

$$c_{d+1} := g_T^\zeta m$$

【0 0 6 2】

30

(S 3 0 5 : データ送信ステップ)

データ送信部 2 4 0 は、(S 3 0 2) で入力した属性の集合 Γ と、(S 3 0 4) で生成された c_0 , c_t , c_{d+1} とを要素とする暗号化データ c を、例えば通信装置によりネットワークを介して復号装置 3 0 0 へ送信する。もちろん、暗号化データ c は、他の方法により復号装置 3 0 0 へ送信されてもよい。

【0 0 6 3】

つまり、(S 3 0 1) から (S 3 0 4) において、公開パラメータ取得部 2 1 0、情報入力部 2 2 0、暗号化データ生成部 2 3 0 は、数 1 4 3 に示す Enc アルゴリズムを実行して、暗号化データ c を生成する。そして、(S 3 0 5) で、データ送信部 2 4 0 が生成された暗号化データ c を復号装置 3 0 0 へ送信する。

40

【数 1 4 3】

$$\text{Enc}(\text{pk}, m, \Gamma := \{(t, \vec{x}_t := (x_{t,1}, \dots, x_{t,n_t}) \in \mathbb{F}_q^{n_t}) \mid 1 \leq t \leq d\} (x_{t,1} := 1)$$

$$\delta, \phi_0, \phi_{t,1}, \dots, \phi_{t,n_t}, \zeta \xleftarrow{\cup} \mathbb{F}_q \text{ such that } (t, \vec{x}_t) \in \Gamma,$$

$$c_0 := (\delta, 0, \zeta, 0, \phi_0)_{\mathbb{B}_0},$$

$$c_t := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}))}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{(\phi_{t,1}, \dots, \phi_{t,n_t})}^{n_t})_{\mathbb{B}_t} \text{ for } (t, \vec{x}_t) \in \Gamma,$$

10

$$c_{d+1} := g_F^{\zeta} m, \quad c := (\Gamma, c_0, \{c_t\}_{(t, \vec{x}_t) \in \Gamma}, c_{d+1}).$$
return c .

【0 0 6 4】

復号装置 3 0 0 の機能と動作とについて説明する。復号装置 3 0 0 は、復号鍵取得部 3 1 0、データ受信部 3 2 0（データ取得部）、スパンププログラム計算部 3 3 0、補完係数計算部 3 4 0、ペアリング演算部 3 5 0、平文情報計算部 3 6 0 を備える。

20

【0 0 6 5】

図 1 0 に基づき、Dec アルゴリズムの処理について説明する。

（S 4 0 1：復号鍵取得ステップ）

復号鍵取得部 3 1 0 は、例えば、通信装置によりネットワークを介して、鍵生成装置 1 0 0 から配布された復号鍵 $\text{sk}_S := (S, k^*_0, k^*_1, \dots, k^*_L)$ を取得する。また、復号鍵取得部 3 1 0 は、鍵生成装置 1 0 0 が生成した公開パラメータ pk を取得する。

【0 0 6 6】

（S 4 0 2：データ受信ステップ）

データ受信部 3 2 0 は、例えば、通信装置によりネットワークを介して、暗号化装置 2 0 0 が送信した暗号化データ c を受信する。

30

【0 0 6 7】

（S 4 0 3：スパンププログラム計算ステップ）

スパンププログラム計算部 3 3 0 は、処理装置により、（S 4 0 1）で取得した復号鍵 sk_S に含まれるアクセスストラクチャ S が、（S 4 0 2）で受信した暗号化データ c に含まれる Γ を受理するか否かを判定する。アクセスストラクチャ S が Γ を受理するか否かの判定方法は、「第 3．関数型暗号を実現するための概念」で説明した通りである。

スパンププログラム計算部 3 3 0 は、アクセスストラクチャ S が Γ を受理する場合（S 4 0 3 で受理）、処理を（S 4 0 4）へ進める。一方、アクセスストラクチャ S が Γ を拒絶する場合（S 4 0 3 で拒絶）、暗号化データ c を復号鍵 sk_S で復号できないとして処理を終了する。

40

【0 0 6 8】

（S 4 0 4：補完係数計算ステップ）

補完係数計算部 3 4 0 は、処理装置により、数 1 4 4 となる I と、定数（補完係数） $\{\alpha_i\}_{i \in I}$ とを計算する。

【数 1 4 4】

$$s_0 = \sum_{i \in I} \alpha_i s_i, \text{ and } I \subseteq \{i \in \{1, \dots, L\} \mid [\rho(i) = (t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t = 0] \\ \vee [\rho(i) = \neg(t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t \neq 0]\}$$

【0 0 6 9】

10

(S 4 0 5 : ペアリング演算ステップ)

ペアリング演算部 3 5 0 は、処理装置により、数 1 4 5 を計算して、セッション鍵 $K = g_T^{\zeta}$ を生成する。

【数 1 4 5】

$$K := e(c_0, k_0^*) \cdot \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i} \cdot \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i / (\vec{v}_i \cdot \vec{x}_t)}$$

なお、数 1 4 6 に示すように、数 1 4 5 を計算することにより鍵 $K = g_T^{\zeta}$ が得られる。
20

【数 1 4 6】

$$e(c_0, k_0^*) \cdot \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i} \cdot \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i / (\vec{v}_i \cdot \vec{x}_t)} \\ = g_T^{-\delta s_0 + \zeta} \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} g_T^{\delta \alpha_i s_i} \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} g_T^{\delta \alpha_i s_i (\vec{v}_i \cdot \vec{x}_t) / (\vec{v}_i \cdot \vec{x}_t)} \\ = g_T^{\delta(-s_0 + \sum_{i \in I} \alpha_i s_i) + \zeta} = g_T^{\zeta}.$$

30

【0 0 7 0】

(S 4 0 6 : 平文情報計算ステップ)

平文情報計算部 3 6 0 は、処理装置により、 $m' = c_{d+1} / K$ を計算して、メッセージ m' ($=m$) を生成する。なお、 c_{d+1} は数 1 4 2 に示す通り $g_T^{\zeta} m$ であり、 K は g_T^{ζ} であるから、 $m' = c_{d+1} / K$ を計算すればメッセージ m が得られる。

【0 0 7 1】

つまり、(S 4 0 1) から (S 4 0 6) において、公開パラメータ取得部 2 1 0、情報入力部 2 2 0、暗号化データ生成部 2 3 0 は、数 1 4 7 に示す Dec アルゴリズムを実行して、メッセージ m' ($=m$) を生成する。
40

【数 1 4 7】

$\text{Dec}(\text{pk}, \text{sk}_{\mathbb{S}} := (\mathbb{S}, k_0^*, k_1^*, \dots, k_L^*), c := (\Gamma, c_0, \{c_t\}_{(t, \vec{x}_t) \in \Gamma}, c_{d+1}))$

If $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{(t, \vec{x}_t)\}$, then compute I and $\{\alpha_i\}_{i \in I}$ such that

$$s_0 = \sum_{i \in I} \alpha_i s_i, \text{ and } I \subseteq \{i \in \{1, \dots, L\} \mid [\rho(i) = (t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t = 0]$$

$$\vee [\rho(i) = \neg(t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t \neq 0]\}.$$

$$K := e(c_0, k_0^*) \cdot \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i} \cdot \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i / (\vec{v}_i \cdot \vec{x}_t)}$$

10

$$m' = c_{d+1} / K.$$

return m' .

【0 0 7 2】

以上のように、暗号処理システム 1 0 は、スパンプログラムと内積述語と秘密分散とを用いて構成したアクセスストラクチャ $\mathbb{S}$ を用いて、暗号方式（関数型暗号方式）を実現する。したがって、暗号処理システム 1 0 は、非常に高い自由度でアクセス制御の設計を行うことが可能な暗号方式を実現する。

20

また、暗号処理システム 1 0 が実現する暗号方式は、非常に安全性が高い。上述したように、関数型暗号方式の 1 つのクラス（最も限定したクラス）に ID ベース暗号がある。既存の実用的な ID ベース暗号と比較しても、暗号処理システム 1 0 が実現する暗号方式はある意味において安全性が高い。

【0 0 7 3】

なお、上記説明では、(S 1 0 1) の (3) で N_0 を 5 に、 N_t を $4n_t (= n_t + n_t + n_t + n_t)$ に設定した。したがって、基底 A_t 、基底 B_t 、基底 B_t^* は、いずれも $4n_t + 5$ 次元である。

しかし、 $n_t + n_t + n_t + n_t$ を $n_t + u_t + w_t + z_t$ にしてもよい。つまり、1 つ目の n_t をそのまま n_t とし、2 つ目の n_t を u_t とし、3 つ目の n_t を w_t とし、4 つ目の n_t を z_t としてもよい。すなわち、 N_t を $n_t + u_t + w_t + z_t$ に設定してもよい。ここで、 n_t 、 u_t 、 w_t 、 z_t は、それぞれ異なる値であってもよく、 n_t は上述したように 1 以上の整数であり、 u_t 、 w_t 、 z_t はいずれも 0 以上の整数である。

30

この場合、数 1 3 1 に示す Setup アルゴリズムは、数 1 4 8 のように書き換えられる。つまり、基底 B_t と基底 B_t^* の基底ベクトルの添え字が変更される。

【数 1 4 8】

Setup($1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d)$)
$$(\text{param}_{\vec{\mu}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0, \dots, d}) \xleftarrow{R} \mathcal{G}_{\text{ob}}(1^\lambda, \vec{\mu})$$

$$\hat{\mathbb{B}}_0 := (b_{0,1}, b_{0,3}, b_{0,5}), \quad \hat{\mathbb{B}}_t := (b_{t,1}, \dots, b_{t,n_t}, b_{t,n_t+u_t+w_t+1}, \dots, b_{t,n_t+u_t+w_t+z_t})$$

for $t = 1, \dots, d$,

$$\hat{\mathbb{B}}_0^* := (b_{0,1}^*, b_{0,3}^*, b_{0,4}^*), \quad \hat{\mathbb{B}}_t^* := (b_{t,1}^*, \dots, b_{t,n_t}^*, b_{t,n_t+u_t+1}^*, \dots, b_{t,n_t+u_t+w_t}^*)$$

for $t = 1, \dots, d$,

$$\text{sk} := \{\hat{\mathbb{B}}_t^*\}_{t=0, \dots, d}, \quad \text{pk} := (1^\lambda, \text{param}_{\vec{\mu}}, \{\mathbb{B}_t\}_{t=0, \dots, d}).$$

return sk, pk.

10

また、数 1 3 8 に示す Key Gen アルゴリズムは、数 1 4 9 のように書き換えられる。

【数 1 4 9】

20

KeyGen(pk, sk, $\mathbb{S} := (M, \rho)$)
$$\vec{f} \xleftarrow{U} \mathbb{F}_q^r, \quad \vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T, \quad s_0 := \vec{1} \cdot \vec{f}^T,$$

$$\eta_0, \theta_i \xleftarrow{U} \mathbb{F}_q \quad (i = 1, \dots, L),$$

$$k_0^* := (-s_0, 0, 1, \eta_0, 0)_{\mathbb{B}_0^*},$$
for $1 \leq i \leq L$

$$\text{if } \rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \quad \eta_{i,1}, \dots, \eta_{i,w_t} \xleftarrow{U} \mathbb{F}_q,$$

30

$$k_i^* := (\overbrace{s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_t}}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,w_t}}^{w_t}, \overbrace{0^{z_t}}^{z_t})_{\mathbb{B}_i^*}$$

$$\text{if } \rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \quad \eta_{i,1}, \dots, \eta_{i,w_t} \xleftarrow{U} \mathbb{F}_q,$$

$$k_i^* := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_t})}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,w_t}}^{w_t}, \overbrace{0^{z_t}}^{z_t})_{\mathbb{B}_i^*}$$

40

$$\text{sk}_{\mathbb{S}} := (\mathbb{S}, k_0^*, k_1^*, \dots, k_L^*).$$
return $\text{sk}_{\mathbb{S}}$.

また、数 1 4 3 に示す Enc アルゴリズムは、数 1 5 0 のように書き換えられる。

【数 1 5 0】

$\text{Enc}(\text{pk}, m, \Gamma := \{(t, \vec{x}_t := (x_{t,1}, \dots, x_{t,n_t}) \in \mathbb{F}_q^{n_t}) \mid 1 \leq t \leq d\} (x_{t,1} := 1)$
 $\delta, \phi_0, \phi_{t,1}, \dots, \phi_{t,z_t}, \zeta \xleftarrow{\text{U}} \mathbb{F}_q$ such that $(t, \vec{x}_t) \in \Gamma$,
 $c_0 := (\delta, 0, \zeta, 0, \phi_0)_{\mathbb{B}_0}$
 $c_t := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}))}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{0^{w_t}}^{w_t}, \overbrace{(\phi_{t,1}, \dots, \phi_{t,z_t})}^{z_t})_{\mathbb{B}_t}$ for $(t, \vec{x}_t) \in \Gamma$,
 $c_{d+1} := g_T^{\zeta} m, \quad c := (\Gamma, c_0, \{c_t\}_{(t, \vec{x}_t) \in \Gamma}, c_{d+1}).$
 return c .

なお、数 1 4 7 に示す D e c アルゴリズムには変更はない。

【0 0 7 4】

また、 N_0 は 5 ではなく、2 以上の整数であればよい。 N_0 が 2 であると、基底 B_0 と基底 B^*_0 とが 2 次元になる。この場合、K e y G e n アルゴリズムにおいて、 $k^*_0 := (-s_0, 1)_{B^*_0}$ とし、E n c アルゴリズムにおいて、 $c_0 := (\delta, \zeta)_{B_0}$ とすればよい。

【0 0 7 5】

また、上記説明では、K e y G e n アルゴリズムにおいて、 $k^*_0 := (-s_0, 0, 1, \eta_0, 0)_{B^*_0}$ とした。しかし、所定の値 κ を用いて、 $k^*_0 := (-s_0, 0, \kappa, \eta_0, 0)_{B^*_0}$ としてもよい。この場合、D e c アルゴリズムで計算される $K := g^{\zeta \kappa}_T$ となるため、E n c アルゴリズムにおいて、 $c_{d+1} := g^{\zeta \kappa}_T m$ とすればよい。

【0 0 7 6】

また、上記説明では、 v_{i, n_t} の値について特に限定しなかった。しかし、安全性の証明の観点から、 $v_{i, n_t} := 1$ である限定としてもよい。

【0 0 7 7】

また、安全性の証明の観点から、 $i = 1, \dots, L$ の各整数 i についての $\rho(i)$ は、それぞれ異なる識別情報 t についての肯定形の組 $(t, v^{\rightarrow})$ 又は否定形の組 $\neg(t, v^{\rightarrow})$ であると限定してもよい。

言い替えると、 $\rho(i) = (t, v^{\rightarrow})$ 又は $\rho(i) = \neg(t, v^{\rightarrow})$ である場合に、関数 $\rho^{\sim}$ を、 $\rho^{\sim}(i) = t$ である $\{1, \dots, L\} \rightarrow \{1, \dots, d\}$ の写像であるとする。この場合、 $\rho^{\sim}$ が単射であると限定してもよい。なお、 $\rho(i)$ は、上述したアクセスストラクチャ $S := (M, \rho(i))$ の $\rho(i)$ である。

【0 0 7 8】

実施の形態 2.

この実施の形態では、「C i p h e r t e x t - P o l i c y 関数型暗号 (C i p h e r t e x t - P o l i c y F u n c t i o n a l E n c r y p t i o n, C P - F E) 方式」について説明する。なお、この実施の形態で説明する「C i p h e r t e x t - P o l i c y 関数型暗号方式」も、実施の形態 1 で説明した概念に基づき構成される。

この実施の形態では、まず、「C i p h e r t e x t - P o l i c y 関数型暗号方式」の基本構成について説明する。次に、この「C i p h e r t e x t - P o l i c y 関数型暗号方式」を実現する「暗号処理システム 1 0」の基本構成について説明する。そして、この実施の形態に係る「C i p h e r t e x t - P o l i c y 関数型暗号方式」及び「暗号処理システム 1 0」について詳細に説明する。

【0 0 7 9】

<Ciphertext-Policy関数型暗号方式の基本構成>

Ciphertext-Policy関数型暗号方式の構成を簡単に説明する。なお、Ciphertext-Policyとは、暗号文にPolicyが埋め込まれること、つまりアクセスストラクチャが埋め込まれることを意味する。

Ciphertext-Policy関数型暗号方式は、Key-Policy関数型暗号方式と同様に、Setup、KeyGen、Enc、Decの4つのアルゴリズムを備える。

(Setup)

Setupアルゴリズムは、セキュリティパラメータ λ と、属性のフォーマット $\mu \rightarrow : = (d; n_1, \dots, n_d)$ とが入力され、公開パラメータ pk と、マスター鍵 sk とを出力する確率的アルゴリズムである。

10

(KeyGen)

KeyGenアルゴリズムは、属性の集合である $\Gamma := \{(t, x \rightarrow_t) \mid x \rightarrow_t \in F_q^{n_t}, 1 \leq t \leq d\}$ と、公開パラメータ pk と、マスター鍵 sk とを入力として、復号鍵 sk_Γ を出力する確率的アルゴリズムである。

(Enc)

Encアルゴリズムは、メッセージ m と、アクセスストラクチャ $S := (M, \rho)$ と、公開パラメータ pk とを入力として、暗号化データ c を出力する確率的アルゴリズムである。

(Dec)

20

Decアルゴリズムは、アクセスストラクチャ S の下で暗号化された暗号化データ c と、属性の集合である Γ に対する復号鍵 sk_Γ と、公開パラメータ pk とを入力として、メッセージ m 、又は、識別情報 $\perp$ を出力するアルゴリズムである。

【0080】

Ciphertext-Policy関数型暗号方式は、全てのアクセスストラクチャ S と属性の集合 Γ と、正しく生成された公開パラメータ pk とマスター鍵 sk と数151に示す暗号文 c とに対して、アクセスストラクチャ S が属性の集合 Γ を受理する場合、 $m = Dec(pk, sk_\Gamma, c)$ であり、アクセスストラクチャ S が属性の集合 Γ を拒絶する場合、 $m = Dec(pk, sk_\Gamma, c)$ となる確率は無視し得る程度である。

【数151】

30

$$c \xleftarrow{R} Enc(pk, m, S)$$

【0081】

<暗号処理システム10>

上述したCiphertext-Policy関数型暗号方式のアルゴリズムを実行する暗号処理システム10について説明する。

図11は、暗号処理システム10の構成図である。

暗号処理システム10は、鍵生成装置100、暗号化装置200、復号装置300を備える。

40

鍵生成装置100は、セキュリティパラメータ λ と、属性のフォーマット $\mu \rightarrow := (d; n_1, \dots, n_d)$ とを入力としてSetupアルゴリズムを実行して、公開パラメータ pk とマスター鍵 sk とを生成する。そして、鍵生成装置100は、生成した公開パラメータ pk を公開する。また、鍵生成装置100は、属性の集合 Γ を入力としてKeyGenアルゴリズムを実行して、復号鍵 sk_Γ を生成して復号装置300へ秘密裡に配布する。

暗号化装置200は、メッセージ m と、アクセスストラクチャ S と、公開パラメータ pk とを入力としてEncアルゴリズムを実行して、暗号化データ c を生成する。暗号化装置200は、生成した暗号化データ c を復号装置300へ送信する。

復号装置300は、公開パラメータ pk と、復号鍵 sk_s と、暗号化データ c とを入力

50

としてDecアルゴリズムを実行して、メッセージ m 又は識別情報 $\perp$ を出力する。

【0082】

<CipherText-Policy関数型暗号方式及び暗号処理システム10の詳細>

図12から図15に基づき、CipherText-Policy関数型暗号方式、及び、CipherText-Policy関数型暗号方式を実行する暗号処理システム10の機能と動作とについて説明する。

図12は、CipherText-Policy関数型暗号方式を実行する暗号処理システム10の機能を示す機能ブロック図である。暗号処理システム10は、上述したように、鍵生成装置100、暗号化装置200、復号装置300を備える。

10

図13は、鍵生成装置100の動作を示すフローチャートであり、KeyGenアルゴリズムの処理を示すフローチャートである。なお、Setupアルゴリズムは、Key-Policy関数型暗号方式の場合と同様であるため、このでは説明を省略する。図14は、暗号化装置200の動作を示すフローチャートであり、Encアルゴリズムの処理を示すフローチャートである。図15は、復号装置300の動作を示すフローチャートであり、Decアルゴリズムの処理を示すフローチャートである。

なお、以下の説明では、 $x_{t,1} := 1$ であるとする。

【0083】

鍵生成装置100の機能と動作とについて説明する。鍵生成装置100は、マスター鍵生成部110、マスター鍵記憶部120、情報入力部130（第1情報入力部）、復号鍵生成部140、鍵配布部150を備える。また、復号鍵生成部140は、乱数生成部143、鍵要素生成部144を備える。

20

【0084】

上述したように、Setupアルゴリズムは、Key-Policy関数型暗号方式の場合と同様であり、数131に示す通りである。

【0085】

図13に基づき、KeyGenアルゴリズムの処理について説明する。

(S501:情報入力ステップ)

情報入力部130は、入力装置により、属性の集合 $\Gamma := \{(t, \vec{x}_t := (x_{t,1}, \dots, x_{t,n_t}) \in F_q^{n_t}) \mid 1 \leq t \leq d\}$ を入力する。なお、属性の集合 Γ は、例えば、復号鍵 sk_Γ の使用者の属性情報が設定されている。

30

【0086】

(S502:乱数生成ステップ)

乱数生成部143は、処理装置により、乱数 δ 、乱数 $\phi_0, \phi_{t,1}, \dots, \phi_{t,n_t}$ を数152に示すように生成する。

【数152】

$$\delta, \phi_0, \phi_{t,1}, \dots, \phi_{t,n_t} \xleftarrow{U} \mathbb{F}_q \text{ such that } (t, \vec{x}_t) \in \Gamma$$

40

【0087】

(S503:鍵要素生成ステップ)

鍵要素生成部144は、処理装置により、復号鍵 sk_Γ の要素 k_0^* を数153に示すように生成する。

【数153】

$$k_0^* := (\delta, 0, 1, \phi_0, 0)_{\mathbb{B}_0^*}$$

また、鍵要素生成部144は、処理装置により、 Γ に含まれる $(t, \vec{x}_t)$ の各整数

50

t について、復号鍵 sk_Γ の要素 k_t^* を数 154 に示すように生成する。

【数 154】

$$k_t^* := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}), 0^{n_t}, \phi_{t,1}, \dots, \phi_{t,n_t}, 0^{n_t})}^{n_t})_{\mathbb{B}_t^*}$$

【0088】

(S504: 鍵配布ステップ)

鍵配布部 150 は、(S501) で入力した属性の集合 Γ と、(S503) で生成された k_0^* 、 k_t^* とを要素とする復号鍵 sk_Γ を、例えば通信装置によりネットワークを介して秘密裡に復号装置 300 へ配布する。もちろん、復号鍵 sk_Γ は、他の方法により復号装置 300 へ配布されてもよい。

10

【0089】

つまり、(S501) から (S503) において、情報入力部 130 と復号鍵生成部 140 とは数 155 に示す KeyGen アルゴリズムを実行して、復号鍵 sk_Γ を生成する。そして、(S504) で、鍵配布部 150 が生成された復号鍵 sk_Γ を復号装置 300 へ配布する。

【数 155】

$$\text{KeyGen}(\text{pk}, \text{sk}, \Gamma := \{(t, \vec{x}_t := (x_{t,1}, \dots, x_{t,n_t}) \in \mathbb{F}_q^{n_t}) \mid 1 \leq t \leq d\} (x_{t,n_t} := 1))$$

20

$$\delta, \phi_0, \phi_{t,1}, \dots, \phi_{t,n_t} \xleftarrow{\text{U}} \mathbb{F}_q \text{ such that } (t, \vec{x}_t) \in \Gamma,$$

$$k_0^* := (\delta, 0, 1, \phi_0, 0)_{\mathbb{B}_0^*},$$

$$k_t^* := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}), 0^{n_t}, \phi_{t,1}, \dots, \phi_{t,n_t}, 0^{n_t})}^{n_t})_{\mathbb{B}_t^*} \text{ for } (t, \vec{x}_t) \in \Gamma,$$

$$\text{sk}_\Gamma := (\Gamma, k_0^*, \{k_t^*\}_{(t, \vec{x}_t) \in \Gamma})$$

30

return sk_Γ .

【0090】

暗号化装置 200 の機能と動作とについて説明する。暗号化装置 200 は、公開パラメータ取得部 210、情報入力部 220 (第 2 情報入力部)、暗号化データ生成部 230、データ送信部 240 (データ出力部) を備える。また、暗号化データ生成部 230 は、乱数生成部 231、暗号要素生成部 232、f ベクトル生成部 233、s ベクトル生成部 234 を備える。

【0091】

40

図 14 に基づき、Enc アルゴリズムの処理について説明する。

(S601: 公開パラメータ取得ステップ)

公開パラメータ取得部 210 は、例えば、通信装置によりネットワークを介して、鍵生成装置 100 が生成した公開パラメータ pk を取得する。

【0092】

(S602: 情報入力ステップ)

情報入力部 220 は、入力装置により、アクセスストラクチャ $S := (M, \rho)$ を入力する。なお、アクセスストラクチャ S の設定については、実現したいシステムの条件に応じて設定されるものである。また、アクセスストラクチャ S の ρ は、例えば、復号可能なユーザの属性情報が設定されている。

50

また、情報入力部 220 は、入力装置により、復号装置 300 へ送信するメッセージ m を入力する。

【0093】

(S603: f ベクトル生成ステップ)

f ベクトル生成部 233 は、処理装置により、 r 個の要素を有するベクトル $f \rightarrow$ を数 156 に示すようにランダムに生成する。

【数156】

$$\vec{f} \xleftarrow{U} \mathbb{F}_q^r$$

10

【0094】

(S604: s ベクトル生成ステップ)

s ベクトル生成部 234 は、処理装置により、(S602) で入力したアクセスストラクチャ S に含まれる (L 行 $\times$ r 列) の行列 M と、(S603) で生成したベクトル $f \rightarrow$ とに基づき、ベクトル $s \rightarrow^T := (s_1, \dots, s_L)^T$ を数 157 に示すように生成する。

【数157】

$$\vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T$$

20

また、 s ベクトル生成部 234 は、処理装置により、(S603) で生成したベクトル $f \rightarrow$ に基づき、値 s_0 を数 158 に示すように生成する。

【数158】

$$s_0 := \vec{1} \cdot \vec{f}^T$$

【0095】

30

(S605: 乱数生成ステップ)

乱数生成部 231 は、処理装置により、乱数 η_0 と、 $i = 1, \dots, L$ の各整数 i について乱数 θ_i と、乱数 ζ とを数 159 に示すように生成する。

【数159】

$$\eta_0, \theta_i, \zeta \xleftarrow{U} \mathbb{F}_q (i=1, \dots, L)$$

【0096】

(S606: 暗号要素生成ステップ)

40

暗号要素生成部 232 は、処理装置により、暗号化データ c の要素 c_0 を数 160 に示すように生成する。

【数160】

$$c_0 := (-s_0, 0, \zeta, 0, \eta_0)_{\mathbb{B}_0}$$

また、暗号要素生成部 232 は、処理装置により、 $i = 1, \dots, L$ の各整数 i について、暗号化データ c の要素 c_i を数 161 に示すように生成する。

【数 1 6 1】

$$\begin{aligned}
 &\text{if } \rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{F}_q^{n_i}), \quad \eta_{i,1}, \dots, \eta_{i,n_i} \xleftarrow{\mathcal{U}} \mathbb{F}_q, \\
 &c_i := (\overbrace{s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_i}}^{n_i})_{\mathbb{B}_i}, \\
 &\text{if } \rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{F}_q^{n_i}), \quad \eta_{i,1}, \dots, \eta_{i,n_i} \xleftarrow{\mathcal{U}} \mathbb{F}_q, \\
 &c_i := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_i})}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_i}}^{n_i})_{\mathbb{B}_i}
 \end{aligned}$$

10

また、暗号要素生成部 2 3 2 は、処理装置により、暗号化データ c の要素 c_{d+1} を数 1 6 2 に示すように生成する。

【数 1 6 2】

$$c_{d+1} := g_T^{\zeta} m$$

20

【0 0 9 7】

(S 6 0 7 : データ送信ステップ)

データ送信部 2 4 0 は、(S 6 0 2) で入力したアクセスストラクチャ S と、(S 6 0 6) で生成された $c_0, c_1, \dots, c_L, c_{d+1}$ とを要素とする暗号化データ c を、例えば通信装置によりネットワークを介して復号装置 3 0 0 へ送信する。もちろん、暗号化データ c は、他の方法により復号装置 3 0 0 へ送信されてもよい。

【0 0 9 8】

つまり、(S 6 0 1) から (S 6 0 6) において、公開パラメータ取得部 2 1 0、情報入力部 2 2 0、暗号化データ生成部 2 3 0 は、数 1 6 3 に示す Enc アルゴリズムを実行して、暗号化データ c を生成する。そして、(S 6 0 7) で、データ送信部 2 4 0 が生成された暗号化データ c を復号装置 3 0 0 へ送信する。

30

【数 1 6 3】

Enc(pk, m, S := (M, ρ))

$$\vec{f} \xleftarrow{R} \mathbb{F}_q^r, \quad \vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T, \quad s_0 := \vec{1} \cdot \vec{f}^T,$$

$$\eta_0, \theta_i, \zeta \xleftarrow{U} \mathbb{F}_q \quad (i = 1, \dots, L),$$

$$c_0 := (-s_0, 0, \zeta, 0, \eta_0)_{\mathbb{B}_0},$$

for $1 \leq i \leq L$

10

$$\text{if } \rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{F}_q^{n_i}), \quad \eta_{i,1}, \dots, \eta_{i,n_i} \xleftarrow{U} \mathbb{F}_q,$$

$$c_i := (\overbrace{s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_i}}^{n_i})_{\mathbb{B}_i},$$

$$\text{if } \rho(i) = -(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{F}_q^{n_i}), \quad \eta_{i,1}, \dots, \eta_{i,n_i} \xleftarrow{U} \mathbb{F}_q,$$

$$c_i := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_i})}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_i}}^{n_i})_{\mathbb{B}_i},$$

20

$$c_{d+1} := g_T^\zeta m, \quad c := (S, c_0, c_1, \dots, c_L, c_{d+1}).$$

return c.

【0 0 9 9】

復号装置 3 0 0 の機能と動作とについて説明する。復号装置 3 0 0 は、復号鍵取得部 3 1 0、データ受信部 3 2 0（データ取得部）、スパンププログラム計算部 3 3 0、補完係数計算部 3 4 0、ペアリング演算部 3 5 0、平文情報計算部 3 6 0 を備える。

【0 1 0 0】

図 1 5 に基づき、Dec アルゴリズムの処理について説明する。

30

(S 7 0 1：復号鍵取得ステップ)

復号鍵取得部 3 1 0 は、例えば、通信装置によりネットワークを介して、鍵生成装置 1 0 0 から配布された復号鍵 sk_Γ を取得する。また、復号鍵取得部 3 1 0 は、鍵生成装置 1 0 0 が生成した公開パラメータ pk を取得する。

【0 1 0 1】

(S 7 0 2：データ受信ステップ)

データ受信部 3 2 0 は、例えば、通信装置によりネットワークを介して、暗号化装置 2 0 0 が送信した暗号化データ c を受信する。

【0 1 0 2】

(S 7 0 3：スパンププログラム計算ステップ)

40

スパンププログラム計算部 3 3 0 は、処理装置により、(S 7 0 2) で取得した暗号化データ c に含まれるアクセスストラクチャ S が、(S 7 0 1) で受信した復号鍵 sk_Γ に含まれる Γ を受理するか否かを判定する。アクセスストラクチャ S が Γ を受理するか否かの判定方法は、「第 3. 関数型暗号を実現するための概念」で説明した通りである。

スパンププログラム計算部 3 3 0 は、アクセスストラクチャ S が Γ を受理する場合 (S 7 0 3 で受理)、処理を (S 7 0 4) へ進める。一方、アクセスストラクチャ S が Γ を拒絶する場合 (S 7 0 3 で拒絶)、暗号化データ c を復号鍵 sk_s で復号できないとして処理を終了する。

【0 1 0 3】

(S 7 0 4) から (S 7 0 6) とは、実施の形態 1 における図 9 に示す (S 4 0 4) と

50

(S 4 0 6)と同様である。

【0 1 0 4】

つまり、(S 7 0 1)から(S 7 0 6)において、公開パラメータ取得部210、情報入力部220、暗号化データ生成部230は、数164に示すDecアルゴリズムを実行して、メッセージ m' ($=m$)を生成する。

【数164】

$\text{Dec}(\text{pk}, \text{sk}_\Gamma := (\Gamma, k_0^*, \{k_t^*\}_{(t, \vec{x}_t) \in \Gamma}), c := (\mathbb{S}, c_0, c_1, \dots, c_L, c_{d+1}))$

If $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{(t, \vec{x}_t)\}$,

10

then compute I and $\{\alpha_i\}_{i \in I}$ such that

$$s_0 = \sum_{i \in I} \alpha_i s_i, \text{ and}$$

$$I \subseteq \{i \in \{1, \dots, L\} \mid [\rho(i) = (t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t = 0]$$

$$\vee [\rho(i) = \neg(t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t \neq 0]\}.$$

$$K := e(c_0, k_0^*) \cdot \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} e(c_i, k_t^*)^{\alpha_i} \cdot \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} e(c_i, k_t^*)^{\alpha_i / (\vec{v}_i \cdot \vec{x}_t)}$$

20

$$m' = c_{d+1} / K.$$

return m' .

【0 1 0 5】

実施の形態2に係る暗号処理システム10は、実施の形態1に係る暗号処理システム10と同様に、非常に高い自由度でアクセス制御の設計を行うことが可能な暗号方式を実現する。また、実施の形態2に係る暗号処理システム10は、実施の形態1に係る暗号処理システム10と同様に、安全性が高い。

【0 1 0 6】

30

なお、実施の形態1と同様に、 $N_t = n_t + n_t + n_t + n_t$ を $n_t + u_t + w_t + z_t$ にしてもよい。つまり、1つ目の n_t をそのまま n_t とし、2つ目の n_t を u_t とし、3つ目の n_t を w_t とし、4つ目の n_t を z_t としてもよい。すなわち、 N_t を $n_t + u_t + w_t + z_t$ に設定してもよい。ここで、 n_t , u_t , w_t , z_t は、それぞれ異なる値であってもよく、 n_t は上述したように1以上の整数であり、 u_t , w_t , z_t はいずれも0以上の整数である。

この場合、Setupアルゴリズムは、実施の形態1と同様に数148のように書き換えられる。

また、数155に示すKeyGenアルゴリズムは、数165のように書き換えられる。

40

【数 1 6 5】

$$\begin{aligned}
& \text{KeyGen}(\text{pk}, \text{sk}, \Gamma := \{(t, \vec{x}_t := (x_{t,1}, \dots, x_{t,n_t}) \in \mathbb{F}_q^{n_t}) \mid 1 \leq t \leq d\} (x_{t,n_t} := 1) \\
& \delta, \phi_0, \phi_{t,1}, \dots, \phi_{t,w_t} \xleftarrow{\text{U}} \mathbb{F}_q \text{ such that } (t, \vec{x}_t) \in \Gamma, \\
& k_0 := (\delta, 0, 1, \phi_0, 0)_{\mathbb{B}_0^*}, \\
& k_t^* := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}))}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{(\phi_{t,1}, \dots, \phi_{t,w_t})}^{w_t}, \overbrace{0^{z_t}}^{z_t})_{\mathbb{B}_t^*} \text{ for } (t, \vec{x}_t) \in \Gamma, \\
& \text{sk}_\Gamma := (\Gamma, k_0^*, \{k_t^*\}_{(t, \vec{x}_t) \in \Gamma}) \\
& \text{return sk}_\Gamma.
\end{aligned} \tag{10}$$

また、数 1 6 3 に示す Enc アルゴリズムは、数 1 6 6 のように書き換えられる。

【数 1 6 6】

$$\begin{aligned}
& \text{Enc}(\text{pk}, m, \mathbb{S} := (M, \rho)) \\
& \vec{f} \xleftarrow{\text{R}} \mathbb{F}_q^r, \quad \vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T, \quad s_0 := \vec{1} \cdot \vec{f}^T, \\
& \eta_0, \theta_i, \zeta \xleftarrow{\text{U}} \mathbb{F}_q (i = 1, \dots, L), \\
& c_0 := (-s_0, 0, \zeta, 0, \eta_0)_{\mathbb{B}_0}, \\
& \text{for } 1 \leq i \leq L \\
& \text{if } \rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \quad \eta_{i,1}, \dots, \eta_{i,z_t} \xleftarrow{\text{U}} \mathbb{F}_q, \\
& c_i := (\overbrace{(s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_t})}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{0^{w_t}}^{w_t}, \overbrace{(\eta_{i,1}, \dots, \eta_{i,z_t})}^{z_t})_{\mathbb{B}_t}, \\
& \text{if } \rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \quad \eta_{i,1}, \dots, \eta_{i,z_t} \xleftarrow{\text{U}} \mathbb{F}_q, \\
& c_i := (\overbrace{(s_i(v_{i,1}, \dots, v_{i,n_t}))}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{0^{w_t}}^{w_t}, \overbrace{(\eta_{i,1}, \dots, \eta_{i,z_t})}^{z_t})_{\mathbb{B}_t}, \\
& c_{d+1} := g_T^\zeta m, \quad c := (\mathbb{S}, c_0, c_1, \dots, c_L, c_{d+1}). \\
& \text{return } c.
\end{aligned} \tag{20}$$

40

なお、数 1 6 4 に示す Dec アルゴリズムには変更はない。

【0 1 0 7】

また、 N_0 は 5 ではなく、2 以上の整数であればよい。 N_0 が 2 であると、基底 B_0 と基底 B_0^* とが 2 次元になる。この場合、Key Gen アルゴリズムにおいて、 $k_0^* := (\delta, 1)_{B_0^*}$ とし、Enc アルゴリズムにおいて、 $c_0 := (-s_0, \zeta)_{B_0}$ とすればよい。

【0 1 0 8】

また、上記説明では、Key Gen アルゴリズムにおいて、 $k_0^* := (\delta, 0, 1, \phi_0, 0)_{B_0^*}$ とした。しかし、所定の値 κ を用いて、 $k_0^* := (\delta, 0, \kappa, \phi_0, 0)_{B_0^*}$ としてもよい。この場合、Dec アルゴリズムで計算される $K := g^{\zeta \kappa} T$

50

となるため、 Enc アルゴリズムにおいて、 $c_{d+1} := g^{\zeta \kappa_T} m$ とすればよい。

【0109】

また、上記説明では、 v_{i, n_t} の値について特に限定しなかった。しかし、安全性の証明の観点から、 $v_{i, n_t} := 1$ である限定としてもよい。

【0110】

また、安全性の証明の観点から、 $i = 1, \dots, L$ の各整数 i についての $\rho(i)$ は、それぞれ異なる識別情報 t についての肯定形の組 $(t, v^\rightarrow)$ 又は否定形の組 $\neg(t, v^\rightarrow)$ であると限定してもよい。

言い替えると、関数 $\rho^\sim$ を、 $\rho(i) = (t, v^\rightarrow)$ 又は $\rho(i) = \neg(t, v^\rightarrow)$ である場合に $\rho^\sim(i) = t$ である $\{1, \dots, L\} \rightarrow \{1, \dots, d\}$ の写像であるとする。この場合、 $\rho^\sim$ が単射であると限定してもよい。なお、 $\rho(i)$ は、上述したアクセスストラクチャ $S := (M, \rho(i))$ の $\rho(i)$ である。

【0111】

また、実施の形態 1 では、アクセスストラクチャ S を復号鍵 sk_S に持たせ、属性集合 Γ を暗号化データ c に持たせた Key-Policy 関数型暗号について説明した。また、実施の形態 2 では、アクセスストラクチャ S を暗号化データ c に持たせ、属性集合 Γ を復号鍵 sk_Γ に持たせた Ciphertext-Policy 関数型暗号について説明した。

しかし、2 つのアクセスストラクチャ S_1 と S_2 とを用意し、一方のアクセスストラクチャ S_1 を復号鍵に持たせ、他方のアクセスストラクチャ S_2 を暗号化データ c に持たせるとともに、アクセスストラクチャ S_2 に対応する属性集合 Γ_2 を復号鍵に持たせ、アクセスストラクチャ S_1 に対応する属性集合 Γ_1 を暗号化データ c に持たせてもよい。そして、アクセスストラクチャ S_1 が属性集合 Γ_1 を受理し、かつ、アクセスストラクチャ S_2 が属性集合 Γ_2 を受理した場合にのみ、暗号化データ c を復号鍵で復号可能としてもよい。

つまり、 Key-Policy 関数型暗号と Ciphertext-Policy 関数型暗号とを組み合わせた暗号方式としてもよい。

【0112】

実施の形態 3.

この実施の形態では、実施の形態 2 で説明した「 Ciphertext-Policy 関数型暗号方式」を応用した署名方式について説明する。

この実施の形態では、まず、「 Ciphertext-Policy 関数型暗号方式に基づく署名方式」の基本構成について説明する。次に、この「署名方式」を実現する「署名処理システム 20」の基本構成について説明する。そして、この実施の形態に係る「署名方式」及び「署名処理システム 20」について詳細に説明する。

【0113】

< Ciphertext-Policy 関数型暗号方式に基づく署名方式の基本構成>
 Ciphertext-Policy 関数型暗号方式に基づく署名方式は、 Setup 、 KeyGen 、 Sig 、 Ver の 4 つのアルゴリズムを備える。

(Setup)

Setup アルゴリズムは、セキュリティパラメータ λ と、属性のフォーマット $\mu^\rightarrow := (d; n_1, \dots, n_d)$ とが入力され、公開パラメータ pk と、マスター鍵 sk とを出力する確率的アルゴリズムである。

(KeyGen)

KeyGen アルゴリズムは、属性の集合である $\Gamma := \{(t, x^\rightarrow_t) \mid x^\rightarrow_t \in F_q^{n_t}, 1 \leq t \leq d\}$ と、公開パラメータ pk と、マスター鍵 sk とを入力として、署名鍵 sk_Γ を出力する確率的アルゴリズムである。

(Sig)

Sig アルゴリズムは、メッセージ m と、署名鍵 sk_Γ と、アクセスストラクチャ $S := (M, \rho)$ と、公開パラメータ pk とを入力として、署名データ sig を出力する確率

10

20

30

40

50

的アルゴリズムである。

(Ver)

Verアルゴリズムは、メッセージ m と、アクセスストラクチャ $S := (M, \rho)$ と、署名データ sig と、公開パラメータ pk とを入力として、署名の検証に成功したことを示す値“1”、又は、署名の検証に失敗したことを示す値“0”を出力するアルゴリズムである。

【0114】

<署名処理システム20>

上述した署名処理のアルゴリズムを実行する署名処理システム20について説明する。

図16は、署名処理システム20の構成図である。

署名処理システム20は、鍵生成装置100、署名装置400、検証装置500を備える。

鍵生成装置100は、セキュリティパラメータ λ と、属性のフォーマット $\mu \rightarrow := (d; n_1, \dots, n_d)$ とを入力としてSetupアルゴリズムを実行して、公開パラメータ pk とマスター鍵 sk とを生成する。そして、鍵生成装置100は、生成した公開パラメータ pk を公開する。また、鍵生成装置100は、属性の集合 Γ を入力としてKeyGenアルゴリズムを実行して、署名鍵 sk_Γ を生成して署名装置400へ秘密裡に配布する。

署名装置400は、メッセージ m と、アクセスストラクチャ S と、公開パラメータ pk と、署名鍵 sk_Γ とを入力としてSigアルゴリズムを実行して、署名ベクトル $s \rightarrow^*$ を生成する。署名装置400は、生成した署名ベクトル $s \rightarrow^*$ と、メッセージ m と、アクセスストラクチャ S とを検証装置500へ送信する。

検証装置500は、署名ベクトル $s \rightarrow^*$ と、メッセージ m と、アクセスストラクチャ S と、公開パラメータ pk とを入力としてVerアルゴリズムを実行して、値“1”、又は、値“0”を出力する。

【0115】

<署名方式及び署名処理システム20の詳細>

図17から図21に基づき、署名方式、及び、署名処理システム20の機能と動作とについて説明する。

図17は、署名処理システム20の機能を示す機能ブロック図である。署名処理システム20は、上述したように、鍵生成装置100、署名装置400、検証装置500を備える。

図18と図19とは、鍵生成装置100の動作を示すフローチャートである。図18はSetupアルゴリズムの処理を示すフローチャートであり、図19はKeyGenアルゴリズムの処理を示すフローチャートである。図20は、署名装置400の動作を示すフローチャートであり、Sigアルゴリズムの処理を示すフローチャートである。図21は、検証装置500の動作を示すフローチャートであり、Verアルゴリズムの処理を示すフローチャートである。

なお、以下の説明では、 $x_{t, 1} := 1$ であるとする。

【0116】

鍵生成装置100の機能と動作とについて説明する。鍵生成装置100は、マスター鍵生成部110、マスター鍵記憶部120、情報入力部130（第1情報入力部）、署名鍵生成部160、鍵配布部150を備える。また、署名鍵生成部160は、乱数生成部161、鍵要素生成部162、秘匿要素生成部163を備える。

【0117】

まず、図18に基づき、Setupアルゴリズムの処理について説明する。

(S801：正規直交基底生成ステップ)

マスター鍵生成部110は、処理装置により、数167を計算して、 $param_{\mu \rightarrow}$ と、 $t = 0, \dots, d + 2$ の各整数 t について基底 B_t 及び基底 B_t^* とを生成する。

10

20

30

40

【数 1 6 7】

(1)

input $1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d)$

(2)

 $\text{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, g, e) \xleftarrow{R} \mathcal{G}_{\text{bpg}}(1^\lambda)$

(3)

 $\psi \xleftarrow{U} \mathbb{F}_q^\times, n_0 := n_{d+1} := 1, n_{d+2} := 2, N_t := 4n_t \text{ for } t = 0, \dots, d+2$

10

 $t = 0, \dots, d+2$ の各 t について (4) から (8) の処理を実行する。

(4)

 $\text{param}_{\mathbb{V}_t} := (q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, e) := \mathcal{G}_{\text{dpvs}}(1^\lambda, N_t, \text{param}_{\mathbb{G}})$

(5)

 $X_t := (\chi_{t,i,j})_{i,j} \xleftarrow{U} GL(N_t, \mathbb{F}_q)$

20

(6)

 $(\nu_{t,i,j})_{i,j} := \psi \cdot (X_t^T)^{-1}$

(7)

 $b_{t,i} := (\chi_{t,i,1}, \dots, \chi_{t,i,N_t})_{\mathbb{A}_t} = \sum_{j=1}^{N_t} \chi_{t,i,j} a_{t,j},$ $\mathbb{B}_t := (b_{t,1}, \dots, b_{t,N_t})$

30

(8)

 $b_{t,i}^* := (\nu_{t,i,1}, \dots, \nu_{t,i,N_t})_{\mathbb{A}_t} = \sum_{j=1}^{N_t} \nu_{t,i,j} a_{t,j},$ $\mathbb{B}_t^* := (b_{t,1}^*, \dots, b_{t,N_t}^*)$

(9)

 $g_T := e(g, g)^\psi,$

40

 $\text{param}_{\vec{\mu}} := (\{\text{param}_{\mathbb{V}_t}\}_{t=0, \dots, d+2}, g_T)$

【0 1 1 8】

数 1 6 7 について、実施の形態 1 における (S 1 0 1) に示す数 1 2 7 と異なる部分のみ説明する。まず、数 1 6 7 では、(3) で n_0, n_{d+1} に 1 を設定し、 n_{d+2} に 2 を設定する。そして、 $t = 0, \dots, d+2$ の各整数 t について、 N_t に $4n_t$ を設定する。そして、(4) から (8) までの処理が、 $t = 0, \dots, d+2$ の各整数 t について繰り返される。

【0 1 1 9】

50

すなわち、(S801)で、マスター鍵生成部110は、数168に示すアルゴリズム G_{ob} を実行して、 $param_{\mu \rightarrow}$ と、 $t = 0, \dots, d+2$ の各整数 t について基底 B_t 及び基底 B^*_t とを生成する。

【数168】

$$G_{ob}(1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d)) : param_G := (q, G, G_T, g, e) \xleftarrow{R} G_{bpg}(1^\lambda),$$

$$\psi \xleftarrow{U} \mathbb{F}_q^\times,$$

$$n_0 := n_{d+1} := 1, n_{d+2} := 2, N_t := 4n_t \text{ for } t = 0, \dots, d+2,$$

10

For $t = 0, \dots, d+2$,

$$param_{V_t} := (q, V_t, G_T, A_t, e) := G_{dpvs}(1^\lambda, N_t, param_G),$$

$$X_t := (\chi_{t,i,j})_{i,j} \xleftarrow{U} GL(N_t, \mathbb{F}_q), \quad (\nu_{t,i,j})_{i,j} := \psi \cdot (X_t^T)^{-1},$$

$$b_{t,i} := (\chi_{t,i,1}, \dots, \chi_{t,i,N_t})_{A_t} = \sum_{j=1}^{N_t} \chi_{t,i,j} a_{t,j}, \quad \mathbb{B}_t := (b_{t,1}, \dots, b_{t,N_t}),$$

$$b_{t,i}^* := (\nu_{t,i,1}, \dots, \nu_{t,i,N_t})_{A_t} = \sum_{j=1}^{N_t} \nu_{t,i,j} a_{t,j}, \quad \mathbb{B}_t^* := (b_{t,1}^*, \dots, b_{t,N_t}^*),$$

20

$$g_T := e(g, g)^\psi, \quad param_{\mu}^- := (\{param_{V_t}\}_{t=0, \dots, d+2}, g_T)$$

$$\text{return } (param_{\mu}^-, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0, \dots, d+2}).$$

【0120】

(S802：部分基底 $B \wedge_t$ 生成ステップ)

マスター鍵生成部110は、処理装置により、 $t = 1, \dots, d+2$ の各整数 t について、部分基底 $B \wedge_t$ を数169に示すように生成する。

【数169】

$$\hat{\mathbb{B}}_t := (b_{t,1}, \dots, b_{t,n_t}, b_{t,3n_t+1}, \dots, b_{t,4n_t})$$

30

【0121】

(S803：部分基底 $B \wedge^*_t$ 生成ステップ)

マスター鍵生成部110は、処理装置により、 $t = 1, \dots, d+2$ の各整数 t について、部分基底 $B \wedge^*_t$ を数170に示すように生成する。

【数170】

$$\hat{\mathbb{B}}_t^* := (b_{t,1}^*, \dots, b_{t,n_t}^*, b_{t,2n_t+1}^*, \dots, b_{t,3n_t}^*)$$

40

【0122】

(S804：マスター鍵記憶ステップ)

マスター鍵生成部110は、 $t = 0, \dots, d+2$ の各整数 t についての部分基底 $B \wedge_t$ と、 $t = 1, \dots, d+2$ の各整数 t についての部分基底 $B \wedge^*_0$ と、基底ベクトル $b^*_{0,3}$ と、(S101)で入力されたセキュリティパラメータ $\lambda(1^\lambda)$ と、(S101)で生成した $param_{\mu \rightarrow}$ とを合わせて、公開パラメータ p_k とする。

また、マスター鍵生成部110は、基底ベクトル $b^*_{0,1}$ をマスター鍵 s_k とする。

そして、マスター鍵記憶部120は、公開パラメータ p_k とマスター鍵 s_k とを記憶装置に記憶する。

50

【0123】

つまり、(S801)から(S803)において、マスター鍵生成部110は数171に示すSetupアルゴリズムを実行して、公開パラメータpkとマスター鍵skとを生成する。そして、(S804)で、マスター鍵記憶部120が生成された公開パラメータpkとマスター鍵skとを記憶装置に記憶する。

なお、公開パラメータは、例えば、ネットワークを介して公開され、署名装置400や検証装置500が取得可能な状態にされる。

【数171】

Setup($1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d)$)

10

$$(\text{param}_{\vec{\mu}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0, \dots, d+2}) \xleftarrow{R} \mathcal{G}_{\text{ob}}(1^\lambda, \vec{\mu})$$

$$n_0 := n_{d+1} := 1, n_{d+2} := 2,$$

$$\hat{\mathbb{B}}_t := (b_{t,1}, \dots, b_{t,n_t}, b_{t,3n_t+1}, \dots, b_{t,4n_t}) \text{ for } t = 0, \dots, d+2,$$

$$\hat{\mathbb{B}}_t^* := (b_{t,1}^*, \dots, b_{t,n_t}^*, b_{t,2n_t+1}^*, \dots, b_{t,3n_t}^*) \text{ for } t = 1, \dots, d+2,$$

$$\text{sk} := b_{0,1}^*,$$

20

$$\text{pk} := (1^\lambda, \text{param}_{\vec{\mu}}, \{\hat{\mathbb{B}}_t\}_{t=0, \dots, d+2}, \{\hat{\mathbb{B}}_t^*\}_{t=1, \dots, d+2}, b_{0,3}^*).$$

return sk, pk.

【0124】

次に、図19に基づき、KeyGenアルゴリズムの処理について説明する。

(S901：情報入力ステップ)

情報入力部130は、入力装置により、属性の集合 $\Gamma := \{(t, \vec{x}_t := (x_{t,1}, \dots, x_{t,n_t}) \in \mathbb{F}_q^{n_t}) \mid 1 \leq t \leq d\}$ を入力する。なお、属性の集合 Γ は、例えば、署名鍵 sk_Γ の使用者の属性情報が設定されている。

30

【0125】

(S902：乱数生成ステップ)

乱数生成部161は、処理装置により、 $t = 1, \dots, d$ 、 $\iota = 1, \dots, n_t$ の各整数 t と ι について、乱数 δ 、乱数 ϕ_0 、 $\phi_{t,\iota}$ 、 $\phi_{d+2,\iota}$ 、 $\phi_{d+3,\iota}$ を数172に示すように生成する。

【数172】

$$\delta, \phi_0, \phi_{t,\iota}, \phi_{d+2,\iota}, \phi_{d+3,\iota} \xleftarrow{U} \mathbb{F}_q$$

$$\text{for } t = 1, \dots, d, \iota = 1, \dots, n_t$$

40

【0126】

(S903：鍵要素生成ステップ)

鍵要素生成部162は、処理装置により、署名鍵 sk_Γ の要素 k_0^* を数173に示すように生成する。

【数173】

$$k_0^* := (\delta, 0, \phi_0, 0)_{\mathbb{B}_0^*}$$

また、鍵要素生成部162は、処理装置により、 Γ に含まれる $(t, \vec{x}_t)$ の各整数

50

t について、署名鍵 sk_Γ の要素 k_t^* を数 174 に示すように生成する。

【数 174】

$$k_t^* := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}))}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{(\phi_{t,1}, \dots, \phi_{t,n_t})}^{n_t}, \overbrace{0^{n_t}}^{n_t})_{\mathbb{B}_t^*}$$

また、鍵要素生成部 162 は、処理装置により、署名鍵 sk_Γ の要素 k_{d+2}^* と要素 k_{d+3}^* とを数 175 に示すように生成する。

【数 175】

$$k_{d+2}^* := (\delta(1, 0), 0, 0, \phi_{d+2,1}, \phi_{d+2,2}, 0, 0)_{\mathbb{B}_{d+2}^*},$$

$$k_{d+3}^* := (\delta(0, 1), 0, 0, \phi_{d+3,1}, \phi_{d+3,2}, 0, 0)_{\mathbb{B}_{d+2}^*}$$

【0127】

(S904: 鍵配布ステップ)

鍵配布部 150 は、(S901) で入力した属性の集合 Γ と、(S903) で生成された k_0^* , k_t^* , k_{d+2}^* , k_{d+3}^* とを要素とする署名鍵 sk_Γ を、例えば通信装置によりネットワークを介して秘密裡に署名装置 400 へ配布する。もちろん、復号鍵 sk_Γ は、他の方法により署名装置 400 へ配布されてもよい。

【0128】

つまり、(S901) から (S903) において、情報入力部 130 と署名鍵生成部 160 とは数 176 に示す KeyGen アルゴリズムを実行して、署名鍵 sk_Γ を生成する。そして、(S904) で、鍵配布部 150 が生成された署名鍵 sk_Γ を署名装置 400 へ配布する。

【数 176】

$$\text{KeyGen}(\text{pk}, \text{sk}, \Gamma := \{(t, \vec{x}_t := (x_{t,1}, \dots, x_{t,n_t}) \in \mathbb{F}_q^{n_t}) \mid 1 \leq t \leq d\}) (x_{t,1} := 1)$$

$$\delta, \phi_0, \phi_{t,l}, \phi_{d+2,l}, \phi_{d+3,l} \xleftarrow{\cup} \mathbb{F}_q$$

for $t = 1, \dots, d$, $l = 1, \dots, n_t$,

$$k_0^* := (\delta, 0, \phi_0, 0)_{\mathbb{B}_0^*},$$

$$k_t^* := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}))}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{(\phi_{t,1}, \dots, \phi_{t,n_t})}^{n_t}, \overbrace{0^{n_t}}^{n_t})_{\mathbb{B}_t^*} \text{ for } (t, \vec{x}_t) \in \Gamma,$$

$$k_{d+2}^* := (\delta(1, 0), 0, 0, \phi_{d+2,1}, \phi_{d+2,2}, 0, 0)_{\mathbb{B}_{d+2}^*},$$

$$k_{d+3}^* := (\delta(0, 1), 0, 0, \phi_{d+3,1}, \phi_{d+3,2}, 0, 0)_{\mathbb{B}_{d+2}^*},$$

$$T := \{0, d+2, d+3\} \cup \{t \mid 1 \leq t \leq d, (t, \vec{x}_t) \in \Gamma\},$$

$$sk_\Gamma := (\Gamma, \{k_t^*\}_{t \in T})$$

return sk_Γ .

【0129】

署名装置 400 の機能と動作について説明する。署名装置 400 は、署名鍵取得部 41

10

20

30

40

50

0、情報入力部420（第2情報入力部）、補完係数計算部430、行列生成部440、署名生成部450、データ送信部460（データ出力部）を備える。また、署名生成部450は、乱数生成部451、署名要素生成部452を備える。

【0130】

図20に基づき、Sigアルゴリズムの処理について説明する。

（S1001：署名鍵取得ステップ）

署名鍵取得部410は、例えば、通信装置によりネットワークを介して、鍵生成装置100から配布された署名鍵 $sk_\Gamma := (\Gamma, k^*_0, k^*_t, k^*_{d+2}, k^*_{d+3})$ を取得する。また、署名鍵取得部410は、鍵生成装置100が生成した公開パラメータ pk を取得する。

10

【0131】

（S1002：情報入力ステップ）

情報入力部420は、入力装置により、アクセスストラクチャ $S := (M, \rho)$ を入力する。なお、ここで入力されるアクセスストラクチャ S は、（S1001）で入力した署名鍵 sk_Γ に含まれる属性の集合 Γ を受理するものとする。

また、情報入力部220は、入力装置により、署名を付すメッセージ m を入力する。

【0132】

（S1003：補完係数計算ステップ）

補完係数計算部430は、処理装置により、数177となる I と、定数（補完係数） $\{\alpha_i\}_{i \in I}$ とを計算する。

20

【数177】

$$\sum_{i \in I} \alpha_i M_i = \vec{1}, \text{ and}$$

$$I \subseteq \{i \in \{1, \dots, L\} \mid [\rho(i) = (t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t = 0]$$

$$\vee [\rho(i) = \neg(t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t \neq 0]\}$$

なお、 M_i とは、行列 M の i 行目のことである。

30

【0133】

（S1004：行追加ステップ）

行列生成部440は、処理装置により、行列 M に追加するベクトル M_{L+1} とそのラベル $\rho(L+1)$ とを数178に示すように生成する。

【数178】

$$M_{L+1} := \vec{1} \in \mathbb{F}_q^r,$$

$$\rho(L+1) := \neg(d+1, \vec{v}_{L+1})$$

40

ここでは、 $\vec{v}_{L+1} := (1)$ とする。

【0134】

（S1005：乱数生成ステップ）

乱数生成部451は、処理装置により、乱数 ξ を数179に示すように生成する。

【数179】

$$\xi \xleftarrow{U} \mathbb{F}_q$$

50

【0135】

(S1006: 署名要素生成ステップ)

署名要素生成部452は、処理装置により、署名ベクトル $s \rightarrow *$ の要素である s^*_0 を数180に示すように生成する。

【数180】

$$s_0^* := \xi k_0^* + r_0^*$$

ここで、

10

$$r_0^* \leftarrow^U \text{span} \langle b_{0,3}^* \rangle$$

である。

また、署名要素生成部452は、処理装置により、 $i = 1, \dots, L+1$ の各整数*i*について、署名ベクトル $s \rightarrow *$ の要素である s^*_i を数181に示すように生成する。

【数181】

$$s_i^* := \gamma_i \cdot \xi k_t^* + \beta_i \cdot \left(\sum_{l=1}^{n_t} y_{i,l} \cdot b_{t,l}^* \right) + r_i^*$$

20

ここで、 r_i^* は、以下のように定義される。

$$r_i^* \leftarrow^U \text{span} \langle b_{t,2n_t+1}^*, \dots, b_{t,3n_t}^* \rangle$$

また、 γ_i と $\vec{y}_i := (y_{i,1}, \dots, y_{i,n_t})$ は、以下のように定義される。

$$\begin{aligned} &\text{if } i \in I \wedge \rho(i) = (t, \vec{v}_i), \gamma_i := \alpha_i, \vec{y}_i \leftarrow^U \{ \vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 0 \wedge y_{i,1} = 1 \}, \\ &\text{if } i \in I \wedge \rho(i) = \neg(t, \vec{v}_i), \gamma_i := \alpha_i / (\vec{v}_i \cdot \vec{x}_t), \vec{y}_i \leftarrow^U \{ \vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 1 \}, \\ &\text{if } i \notin I \wedge \rho(i) = (t, \vec{v}_i), \gamma_i := 0, \vec{y}_i \leftarrow^U \{ \vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 0 \wedge y_{i,1} = 1 \}, \\ &\text{if } i \notin I \wedge \rho(i) = \neg(t, \vec{v}_i), \gamma_i := 0, \vec{y}_i \leftarrow^U \{ \vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 1 \} \end{aligned}$$

30

また、 β_i は以下のように定義される。

40

$$\{\beta_i\} \leftarrow^U \{ \{\beta_i\} \mid \sum_{i=1}^{L+1} \beta_i M_i = \vec{0} \}$$

また、署名要素生成部452は、処理装置により、署名ベクトル $s \rightarrow *$ の要素である s^*_{L+2} を数182に示すように生成する。

【数 1 8 2】

$$s_{L+2}^* := \xi(k_{d+2}^* + m \cdot k_{d+3}^*) + r_{L+2}^*$$

ここで、

$$r_{L+2}^* \xleftarrow{U} \text{span} \langle b_{d+2,5}^*, b_{d+2,6}^* \rangle$$

である。

10

【0 1 3 6】

(S 1 0 0 7 : データ送信ステップ)

データ送信部 4 6 0 は、(S 1 0 0 2) で入力したメッセージ m と、アクセスストラクチャ $S := (M, \rho)$ と、(S 1 0 0 7) で生成された s^* とを含む署名データ sig を、例えば通信装置によりネットワークを介して検証装置 5 0 0 へ送信する。もちろん、署名データ sig は、他の方法により検証装置 5 0 0 へ送信されてもよい。

【0 1 3 7】

つまり、(S 1 0 0 1) から (S 1 0 0 6) において、署名鍵取得部 4 1 0、情報入力部 4 2 0、補完係数計算部 4 3 0、行列生成部 4 4 0、署名生成部 4 5 0 は、数 1 8 3 に示す Sig アルゴリズムを実行して、署名データ sig を生成する。そして、(S 1 0 0 7) で、データ送信部 4 6 0 が生成された署名データを検証装置 5 0 0 へ送信する。

20

【数 1 8 3】

$\text{Sig}(\text{pk}, \text{sk}_\Gamma, m, \mathbb{S} := (M, \rho))$

If $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{(t, \vec{x}_t)\}$, then compute I and $\{\alpha_i\}_{i \in I}$ such that

$$\sum_{i \in I} \alpha_i M_i = \vec{1}, \text{ and}$$

$$I \subseteq \{i \in \{1, \dots, L\} \mid [\rho(i) = (t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t = 0] \\ \vee [\rho(i) = \neg(t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t \neq 0]\},$$

10

$$M_{L+1} := \vec{1} \in \mathbb{F}_q^r, \vec{v}_{L+1} := (1), \rho(L+1) := \neg(d+1, \vec{v}_{L+1})$$

$$\xi \xleftarrow{\mathbb{U}} \mathbb{F}_q, \{\beta_i\} \xleftarrow{\mathbb{U}} \{(\beta_1, \dots, \beta_{L+1}) \mid \sum_{i=1}^{L+1} \beta_i M_i = \vec{0}\},$$

$$s_0^* := \xi k_0^* + r_0^*, \text{ where } r_0^* \xleftarrow{\mathbb{U}} \text{span}\langle b_{0,3}^* \rangle,$$

for $1 \leq i \leq L+1$,

$$s_i^* := \gamma_i \cdot \xi k_t^* + \beta_i \cdot (\sum_{l=1}^{n_t} y_{i,l} \cdot b_{t,l}^*) + r_i^*,$$

20

$$\text{where } r_i^* \xleftarrow{\mathbb{U}} \text{span}\langle b_{t,2n_i+1}^*, \dots, b_{t,3n_i}^* \rangle,$$

and $\gamma_i, \vec{y}_i := (y_{i,1}, \dots, y_{i,n_i})$ are defined as

$$\text{if } i \in I \wedge \rho(i) = (t, \vec{v}_i), \gamma_i := \alpha_i, \vec{y}_i \xleftarrow{\mathbb{U}} \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 0 \wedge y_{i,1} = 1\},$$

$$\text{if } i \in I \wedge \rho(i) = \neg(t, \vec{v}_i), \gamma_i := \alpha_i / (\vec{v}_i \cdot \vec{x}_t), \vec{y}_i \xleftarrow{\mathbb{U}} \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 1\},$$

$$\text{if } i \notin I \wedge \rho(i) = (t, \vec{v}_i), \gamma_i := 0, \vec{y}_i \xleftarrow{\mathbb{U}} \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 0 \wedge y_{i,1} = 1\},$$

30

$$\text{if } i \notin I \wedge \rho(i) = \neg(t, \vec{v}_i), \gamma_i := 0, \vec{y}_i \xleftarrow{\mathbb{U}} \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 1\},$$

$$s_{L+2}^* := \xi(k_{d+2}^* + m \cdot k_{d+3}^*) + r_{L+2}^*, \text{ where } r_{L+2}^* \xleftarrow{\mathbb{U}} \text{span}\langle b_{d+2,5}^*, b_{d+2,6}^* \rangle$$

$$\vec{s}^* := (s_0^*, \dots, s_{L+2}^*), \text{ sig} := \{m, \mathbb{S}, \vec{s}^*\}$$

return sig.

【0 1 3 8】

40

検証装置 500 の機能と動作とについて説明する。検証装置 500 は、公開パラメータ取得部 510、データ受信部 520（データ取得部）、暗号化データ生成部 530、ペアリング演算部 540 を備える。暗号化データ生成部 530 は、乱数生成部 531、暗号要素生成部 532、f ベクトル生成部 533、s ベクトル生成部 534 を備える。

【0 1 3 9】

図 21 に基づき、Ver アルゴリズムの処理について説明する。

(S1101：公開パラメータ取得ステップ)

公開パラメータ取得部 510 は、例えば、通信装置によりネットワークを介して、鍵生成装置 100 が生成した公開パラメータ pk を取得する。

【0 1 4 0】

50

(S 1 1 0 2 : データ受信ステップ)

データ受信部 3 2 0 は、例えば、通信装置によりネットワークを介して、署名装置 4 0 0 が送信した署名データ s_{ig} を受信する。

【0 1 4 1】

(S 1 1 0 3 : f ベクトル生成ステップ)

f ベクトル生成部 5 3 3 は、処理装置により、r 個の要素を有するベクトル $f \rightarrow$ を数 1 8 4 に示すようにランダムに生成する。

【数 1 8 4】

$$\vec{f} \xleftarrow{R} \mathbb{F}_q^r \quad 10$$

【0 1 4 2】

(S 1 1 0 4 : s ベクトル生成ステップ)

s ベクトル生成部 5 3 4 は、処理装置により、(S 1 1 0 2) で受信したアクセスストラクチャ S に含まれる (L 行 × r 列) の行列 M と、(S 1 1 0 3) で生成したベクトル $f \rightarrow$ とに基づき、ベクトル $s \rightarrow^T := (s_1, \dots, s_L)^T$ を数 1 8 5 に示すように生成する。

【数 1 8 5】

$$\vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T \quad 20$$

また、s ベクトル生成部 5 3 4 は、処理装置により、(S 1 1 0 3) で生成したベクトル $f \rightarrow$ に基づき、値 s_0 と値 s_{L+1} とを数 1 8 6 に示すように生成する。

【数 1 8 6】

$$s_0 := s_{L+1} := \vec{1} \cdot \vec{f}^T \quad 30$$

【0 1 4 3】

(S 1 1 0 5 : 乱数生成ステップ)

乱数生成部 5 3 1 は、処理装置により、 $i = 1, \dots, L + 2$ の各整数 i について、乱数 $\eta_0, \eta_{L+2,1}, \eta_{L+2,2}, \theta_i, s_{L+2}$ を数 1 8 7 に示すように生成する。

【数 1 8 7】

$$\eta_0, \eta_{L+2,1}, \eta_{L+2,2}, \theta_i, s_{L+2} \xleftarrow{U} \mathbb{F}_q \quad 40$$

(S 1 1 0 6 : 暗号要素生成ステップ)

暗号要素生成部 5 3 2 は、処理装置により、暗号化データ c の要素 c_0 を数 1 8 8 に示すように生成する。

【数 1 8 8】

$$c_0 := (-s_0 - s_{L+2}, 0, 0, \eta_0)_{\mathbb{B}_0}$$

また、暗号要素生成部 5 3 2 は、処理装置により、 $i = 1, \dots, L + 1$ の各整数 i について、暗号化データ c の要素 c_i を数 1 8 9 に示すように生成する。

【数 1 8 9】

if $\rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{F}_q^{n_i}), \eta_{i,1}, \dots, \eta_{i,n_i} \xleftarrow{\mathcal{U}} \mathbb{F}_q,$

$$c_i := (\overbrace{s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_i}}^{n_i})_{\mathbb{B}_i},$$

if $\rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{F}_q^{n_i}), \eta_{i,1}, \dots, \eta_{i,n_i} \xleftarrow{\mathcal{U}} \mathbb{F}_q,$

$$c_i := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_i})}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_i}}^{n_i})_{\mathbb{B}_i},$$

10

また、暗号要素生成部 5 3 2 は、処理装置により、暗号化データ c の要素 c_{L+2} を数 1 9 0 に示すように生成する。

【数 1 9 0】

$$c_{L+2} := (s_{L+2} - \theta_{L+2}m, \theta_{L+2}, 0, 0, 0, 0, \eta_{L+2,1}, \eta_{L+2,2})_{\mathbb{B}_{d+2}}$$

20

【0 1 4 4】

(S 1 1 0 7 : ペアリング演算ステップ)

ペアリング演算部 5 4 0 は、処理装置により、数 1 9 1 を計算する。

【数 1 9 1】

$$\prod_{i=0}^{L+2} e(c_i, s_i^*)$$

なお、ここでは、 $\rho(L+1) := \neg(d+1, v \rightarrow_{L+1} := (1))$ であるとする

30

。ペアリング演算部 5 4 0 は、数 1 9 1 を計算した結果が値 “ 1 ” であれば、値 “ 1 ” を出力し、その他の値であれば値 “ 0 ” を出力する。ここで、数 1 9 1 を計算した結果が値 “ 1 ” であれば、署名が検証できたことを示し、その他の値であれば署名が検証できなかったことを示す。

なお、数 1 9 2 に示すように、署名データ s_{ig} が正しいデータであれば、数 1 9 1 を計算することにより値 “ 1 ” が得られる。

【数 1 9 2】

$$\begin{aligned} \prod_{i=0}^{L+2} e(c_i, s_i^*) &= e(c_0, s_0^*) \cdot \prod_{i \in I} e(c_i, k_i^*)^{\xi \gamma_i} \cdot \prod_{i=1}^{L+1} e(c_i, \prod_{t=1}^{n_i} (b_{t,i}^*)^{y_{i,t}})^{\beta_i} \cdot e(c_{L+2}, s_{L+2}^*) \\ &= g_T^{\xi \delta (-s_0 - s_{L+2})} \cdot \prod_{i \in I} g_T^{\xi \delta \alpha_i s_i} \cdot \prod_{i=1}^{L+1} g_T^{\beta_i s_i} \cdot g_T^{\xi \delta s_{L+2}} \\ &= g_T^{\xi \delta (-s_0 - s_{L+2})} \cdot g_T^{\xi \delta s_0} \cdot g_T^{\xi \delta s_{L+2}} = 1 \end{aligned}$$

40

【0 1 4 5】

50

つまり、(S 1 1 0 1) から (S 1 1 0 7) において、公開パラメータ取得部 5 1 0、データ受信部 5 2 0、暗号化データ生成部 5 3 0、ペアリング演算部 5 4 0 は、数 1 9 3 に示す Ver アルゴリズムを実行して、署名データ s i g を検証する。

【数 1 9 3】

Ver(pk,m,S:=(M,ρ), $\vec{s}^*$)

$$\vec{f} \xleftarrow{\mathbb{R}} \mathbb{F}_q^r, \quad \vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T, \quad s_0 := s_{L+1} := \vec{1} \cdot \vec{f}^T,$$

$$\eta_0, \eta_{L+2,1}, \eta_{L+2,2}, \theta_i, s_{L+2} \xleftarrow{\mathbb{U}} \mathbb{F}_q \quad (i=1, \dots, L+2),$$

10

$$\rho(L+1) := \neg(d+1, \vec{v}_{L+1} := (1))$$

$$c_0 := (-s_0 - s_{L+2}, 0, 0, \eta_0)_{\mathbb{B}_0},$$

for $1 \leq i \leq L+1$

if $\rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{F}_q^{n_i})$,

$$\eta_{i,1}, \dots, \eta_{i,n_i} \xleftarrow{\mathbb{U}} \mathbb{F}_q,$$

$$c_i := (\overbrace{s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_i}}^{n_i})_{\mathbb{B}_i},$$

20

if $\rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{F}_q^{n_i})$,

$$\eta_{i,1}, \dots, \eta_{i,n_i} \xleftarrow{\mathbb{U}} \mathbb{F}_q,$$

$$c_i := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_i})}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_i}}^{n_i})_{\mathbb{B}_i},$$

$$c_{L+2} := (s_{L+2} - \theta_{L+2} m, \theta_{L+2}, 0, 0, 0, 0, \eta_{L+2,1}, \eta_{L+2,2})_{\mathbb{B}_{d+2}},$$

30

return 1 if $\prod_{i=0}^{L+2} e(c_i, s_i^*) = 1$, return 0 otherwise.

【0 1 4 6】

以上のように、署名処理システム 2 0 は、スパンプログラムと内積述語と秘密分散とを用いて構成したアクセスストラクチャ S を用いて、署名方式を実現する。

【0 1 4 7】

なお、S i g アルゴリズムにおいて、乱数 ξ や乱数 r を用いて署名ベクトル $s^{\rightarrow *}$ の要素を生成しているのは、署名ベクトル $s^{\rightarrow *}$ の要素の生成元の情報である署名鍵 $s k_{\Gamma}$ の要素 k の変数 δ や変数 ϕ をランダム化するためである。署名鍵 $s k_{\Gamma}$ の要素 k の変数 δ や変数 ϕ をランダム化することにより、署名データの要素である $s^{\rightarrow *}$ からこれらの変数が読み取られることを防止することができる。つまり、署名データがどの署名鍵 $s k_{\Gamma}$ に基づき生成されたものであるかを読み取られることを防止することができる。すなわち、リンクアビリティを高くすることができる。

40

【0 1 4 8】

また、上記説明では、(S 8 0 1) の N_t を $4 n_t (= n_t + n_t + n_t + n_t)$ に設定した。

しかし、 $n_t + n_t + n_t + n_t$ を $n_t + u_t + w_t + z_t$ にしてもよい。つまり、1 つ目の n_t をそのまま n_t とし、2 つ目の n_t を u_t とし、3 つ目の n_t を w_t とし、4

50

つ目の n_t を z_t としてもよい。すなわち、 N_t を $n_t + u_t + w_t + z_t$ に設定してもよい。ここで、 n_t 、 u_t 、 w_t 、 z_t は、それぞれ異なる値であってもよく、 n_t は上述したように 1 以上の整数であり、 u_t 、 w_t 、 z_t はいずれも 0 以上の整数である。

この場合、数 171 に示す Setup アルゴリズムは、数 194 のように書き換えられる。つまり、基底 $B^{\wedge}_t$ と基底 $B^{\wedge*}_t$ の基底ベクトルの添え字が変更される。

【数 194】

Setup($1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d)$)

$$(\text{param}_{\vec{\mu}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0, \dots, d+2}) \xleftarrow{R} \mathcal{G}_{\text{ob}}(1^\lambda, \vec{\mu}) \quad 10$$

$$\hat{\mathbb{B}}_t := (b_{t,1}, \dots, b_{t,n_t}, b_{t,n_t+u_t+w_t+1}, \dots, b_{t,n_t+u_t+w_t+z_t}) \text{ for } t = 1, \dots, d+2,$$

$$\hat{\mathbb{B}}_t^* := (b_{t,1}^*, \dots, b_{t,n_t}^*, b_{t,n_t+u_t+1}^*, \dots, b_{t,n_t+u_t+w_t}^*) \text{ for } t = 1, \dots, d+2,$$

$$\hat{\mathbb{B}}_{d+1} := (b_{d+1,1}, b_{d+1,2}, b_{d+1,4}), \quad \hat{\mathbb{B}}_{d+1}^* := (b_{d+1,1}^*, b_{d+1,2}^*, b_{d+1,3}^*),$$

$$\text{sk} := b_{0,1}^*, \quad \text{pk} := (1^\lambda, \text{param}_{\vec{\mu}}, \{\hat{\mathbb{B}}_t\}_{t=0, \dots, d+2}, \{\hat{\mathbb{B}}_t^*\}_{t=1, \dots, d+2}, b_{0,3}^*).$$

return sk, pk.

20

また、数 176 に示す KeyGen アルゴリズムは、数 195 のように書き換えられる。

【数 195】

KeyGen(pk, sk, $\Gamma := \{(t, \vec{x}_t := (x_{t,1}, \dots, x_{t,n_t}) \in \mathbb{F}_q^{n_t}) \mid 1 \leq t \leq d\} (x_{t,1} := 1)$)

$$\delta, \phi_0, \phi_{t,l}, \phi_{d+2,l}, \phi_{d+3,l} \xleftarrow{U} \mathbb{F}_q$$

for $t = 1, \dots, d$, $l = 1, \dots, w_t$,

$$k_0^* := (\delta, 0, \phi_0, 0)_{\mathbb{B}_0^*},$$

$$k_t^* := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}))}^{n_t}, \overbrace{0^{u_t}}^{u_t}, \overbrace{(\phi_{t,1}, \dots, \phi_{t,w_t})}^{w_t}, \overbrace{0^{z_t}}^{z_t})_{\mathbb{B}_t^*} \text{ for } (t, \vec{x}_t) \in \Gamma,$$

$$k_{d+2}^* := (\delta(1, 0), 0, 0, \phi_{d+2,1}, \phi_{d+2,2}, 0, 0)_{\mathbb{B}_{d+2}^*},$$

$$k_{d+3}^* := (\delta(0, 1), 0, 0, \phi_{d+3,1}, \phi_{d+3,2}, 0, 0)_{\mathbb{B}_{d+3}^*},$$

$$T := \{0, d+2, d+3\} \cup \{t \mid 1 \leq t \leq d, (t, \vec{x}_t) \in \Gamma\},$$

$$\text{sk}_\Gamma := (\Gamma, \{k_t^*\}_{t \in T})$$

return sk_Γ .

30

40

また、数 183 に示す Sig アルゴリズムは、数 196 のように書き換えられる。

【数 1 9 6】

Sig(pk, sk_Γ, m, S := (M, ρ))If S := (M, ρ) accepts Γ := {(t, $\vec{x}_t$)}, then compute I and {α_i}_{i∈I} such that

$$\sum_{i \in I} \alpha_i M_i = \vec{1}, \text{ and}$$

$$I \subseteq \{i \in \{1, \dots, L\} \mid [\rho(i) = (t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t = 0] \\ \vee [\rho(i) = \neg(t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t \neq 0]\},$$

10

$$M_{L+1} := \vec{1} \in \mathbb{F}_q^r, \vec{v}_{L+1} := (1), \rho(L+1) := \neg(d+1, \vec{v}_{L+1})$$

$$\xi \leftarrow \bigcup \mathbb{F}_q, \{\beta_i\} \leftarrow \bigcup \{(\beta_1, \dots, \beta_{L+1}) \mid \sum_{i=1}^{L+1} \beta_i M_i = \vec{0}\},$$

$$s_0^* := \xi k_0^* + r_0^*, \text{ where } r_0^* \leftarrow \bigcup \text{span} \langle b_{0,3}^* \rangle,$$

for 1 ≤ i ≤ L + 1,

$$s_i^* := \gamma_i \cdot \xi k_t^* + \beta_i \cdot (\sum_{l=1}^{n_i} y_{i,l} \cdot b_{t,l}^*) + r_i^*,$$

20

$$\text{where } r_i^* \leftarrow \bigcup \text{span} \langle b_{t,n_i+u_i+1}^*, \dots, b_{t,n_i+u_i+w_i}^* \rangle,$$

and $\gamma_i, \vec{y}_i := (y_{i,1}, \dots, y_{i,n_i})$ are defined as

$$\text{if } i \in I \wedge \rho(i) = (t, \vec{v}_i), \gamma_i := \alpha_i, \vec{y}_i \leftarrow \bigcup \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 0 \wedge y_{i,1} = 1\},$$

$$\text{if } i \in I \wedge \rho(i) = \neg(t, \vec{v}_i), \gamma_i := \alpha_i / (\vec{v}_i \cdot \vec{x}_t), \vec{y}_i \leftarrow \bigcup \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 1\},$$

$$\text{if } i \notin I \wedge \rho(i) = (t, \vec{v}_i), \gamma_i := 0, \vec{y}_i \leftarrow \bigcup \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 0 \wedge y_{i,1} = 1\},$$

30

$$\text{if } i \notin I \wedge \rho(i) = \neg(t, \vec{v}_i), \gamma_i := 0, \vec{y}_i \leftarrow \bigcup \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 1\},$$

$$s_{L+2}^* := \xi(k_{d+2}^* + m \cdot k_{d+3}^*) + r_{L+2}^*, \text{ where } r_{L+2}^* \leftarrow \bigcup \text{span} \langle b_{d+2,5}^*, b_{d+2,6}^* \rangle$$

$$\vec{s}^* := (s_0^*, \dots, s_{L+2}^*), \text{ sig} := \{m, S, \vec{s}^*\}$$

return sig.

40

また、数 1 9 3 に示す Ver アルゴリズムは、数 1 9 7 のように書き換えられる。

【数 1 9 7】

$$\text{Ver}(\text{pk}, m, \mathbb{S} := (M, \rho), \vec{s}^*)$$

$$\vec{f} \xleftarrow{\mathbb{R}} \mathbb{F}_q^r, \quad \vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T, \quad s_0 := s_{L+1} := \vec{1} \cdot \vec{f}^T,$$

$$\eta_0, \eta_{L+1,1}, \eta_{L+1,2}, \theta_i, s_{L+2} \xleftarrow{\mathbb{U}} \mathbb{F}_q \quad (i=1, \dots, L+2),$$

$$c_0 := (-s_0 - s_{L+1}, 0, 0, \eta_0)_{\mathbb{B}_0},$$

for $1 \leq i \leq L+1$

if $\rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{F}_q^{n_i})$,

$$\eta_{i,1}, \dots, \eta_{i,z_i} \xleftarrow{\mathbb{U}} \mathbb{F}_q,$$

$$c_i := (\overbrace{s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_i}}^{n_i}, \overbrace{0^{u_i}}^{u_i}, \overbrace{0^{w_i}}^{w_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,z_i}}^{z_i})_{\mathbb{B}_i},$$

if $\rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{F}_q^{n_i})$,

$$\eta_{i,1}, \dots, \eta_{i,z_i} \xleftarrow{\mathbb{U}} \mathbb{F}_q,$$

$$c_i := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_i})}^{n_i}, \overbrace{0^{u_i}}^{u_i}, \overbrace{0^{w_i}}^{w_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,z_i}}^{z_i})_{\mathbb{B}_i},$$

$$c_{L+1} := (s_{L+2} - \theta_{L+2}m, \theta_{L+2}, 0, 0, 0, 0, \eta_{L+2,1}, \eta_{L+2,2})_{\mathbb{B}_{d+2}},$$

return 1 if $\prod_{i=0}^{L+2} e(c_i, s_i^*) = 1$, return 0 otherwise.

【0 1 4 9】

また、 N_0 や N_{d+1} は 4 ではなく、1 以上の整数であればよい。 N_0 が 1 であると、基底 B_0 と基底 B_0^* とが 1 次元になる。この場合、Key Gen アルゴリズムにおいて、 $k_{0,j}^* := (\delta)_{B_0^*}$ とし、Ver アルゴリズムにおいて、 $c_0 := (-s_0 - s_{L+2})_{B_0}$ とすればよい。

また、 N_{d+2} は 8 ではなく、2 以上の整数であればよい。 N_{d+2} が 2 であると、基底 B_{d+2} と基底 B_{d+2}^* とが 2 次元になる。この場合、Key Gen アルゴリズムにおいて、 $k_{d+2,j}^* := (\delta_j(1, 0))_{B_{d+2}^*}$ 、 $k_{d+3,j}^* := (\delta_j(0, 1))_{B_{d+2}^*}$ とし、Ver アルゴリズムにおいて、 $c_0 := (s_{L+2} - \theta_{L+2}m, \theta_{L+2})_{B_{d+2}}$ とすればよい。

【0 1 5 0】

また、上記説明では、 v_{i,n_i} の値について特に限定しなかった。しかし、安全性の証明の観点から、 $v_{i,n_i} := 1$ である限定としてもよい。

【0 1 5 1】

また、安全性の証明の観点から、 $i = 1, \dots, L$ の各整数 i についての $\rho(i)$ は、それぞれ異なる識別情報 t についての肯定形の組 $(t, v^\rightarrow)$ 又は否定形の組 $\neg(t, v^\rightarrow)$ であるとしてもよい。

言い替えると、関数 $\rho^\sim$ を、 $\rho(i) = (t, v^\rightarrow)$ 又は $\rho(i) = \neg(t, v^\rightarrow)$ である場合に $\rho^\sim(i) = t$ である $\{1, \dots, L\} \rightarrow \{1, \dots, d\}$ の写像であるとする。この場合、 $\rho^\sim$ が単射であるとしてもよい。なお、 $\rho(i)$ は、上述したアクセスストラクチャ $\mathbb{S} := (M, \rho(i))$ の $\rho(i)$ である。

10

20

30

40

50

【0152】

また、上記説明では、S i g アルゴリズムにおいて、行追加ステップ (S 1 0 0 4) で、行列 M に L + 1 行目として、ベクトル M_{L+1} の 1 行を追加した。しかし、行列 M に追加する行は、1 行以上であれば何行であっても構わない。また、追加した $M_{L+1} := 1 \rightarrow$ であるとした。しかし、追加する行は、 $1 \rightarrow$ に限らず他のベクトルであってもよい。

また、上記説明では、追加した L + 1 行目のラベル $\rho (L+1) := \neg (d+1, v \rightarrow_{L+1} := (1))$ とした。しかし、追加する行のラベルの設定は、処理が成立するようになっていけばよく、これに限るものではない。

つまり、追加する行や、追加する行のラベルは、V e r アルゴリズムのペアリング演算ステップ (S 1 1 0 7) でペアリング演算をした場合に、追加した行の情報が 0 になるように設定されていけばよい。

10

なお、追加した行が 2 行以上である場合には、追加した行数に応じて、S i g アルゴリズムや V e r アルゴリズムにおける処理の繰り返し回数を変更する必要がある。

【0153】

また、署名処理の安全性を高めるため、数 1 7 1 に示す署名アルゴリズムを数 1 9 8 のように変更し、数 1 8 3 に示す S i g アルゴリズムを数 1 9 9 のように変更してもよい。

【数 1 9 8】

Setup($1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d)$)

$$(\text{param}_{\vec{\mu}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0, \dots, d+2}) \xleftarrow{R} \mathcal{G}_{\text{ob}}(1^\lambda, \vec{\mu})$$

20

$$n_0 := n_{d+1} := 1, n_{d+2} := 2,$$

$$\hat{\mathbb{B}}_t := (b_{t,1}, \dots, b_{t,n_t}, b_{t,3n_t+1}, \dots, b_{t,4n_t}) \text{ for } t = 0, \dots, d+2,$$

$$\hat{\mathbb{B}}_t^* := (b_{t,1}^*, \dots, b_{t,n_t}^*), \tilde{\mathbb{B}}_t^* := (b_{t,2n_t+1}^*, \dots, b_{t,3n_t}^*) \text{ for } t = 1, \dots, d+2,$$

$$\sigma, \psi_{t,i,l} \xleftarrow{U} \mathbb{F}_q \text{ for } t = 1, \dots, d+1, i = 1, \dots, n_t, l = 1, \dots, n_t,$$

$$p_{t,l}^* := (\overbrace{(0^{t-1}, \sigma, 0^{n_t-t}, 0^{n_t}, \psi_{t,1,l}, \dots, \psi_{t,n_t,l}, 0^{n_t})}^{n_t})_{\mathbb{B}_t^*}$$

for $t = 1, \dots, d+1, l = 1, \dots, n_t,$

30

$$\text{sk} := (b_{0,1}^*, \{\hat{\mathbb{B}}_t^*\}_{t=1, \dots, d+2})$$

$$\text{pk} := (1^\lambda, \text{param}_{\vec{\mu}}, \hat{\mathbb{B}}_0, b_{0,3}, \{\hat{\mathbb{B}}_t, \tilde{\mathbb{B}}_t^*\}_{t=1, \dots, d+2}, \{p_{t,l}^*\}_{t=1, \dots, d+2; l=1, \dots, n_t}).$$

return sk, pk.

40

【数 1 9 9】

$\text{Sig}(\text{pk}, \text{sk}_\Gamma, m, \mathbb{S} := (M, \rho))$

If $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{(t, \vec{x}_t)\}$, then compute I and $\{\alpha_i\}_{i \in I}$ such that

$$\sum_{i \in I} \alpha_i M_i = \vec{1}, \text{ and}$$

$$I \subseteq \{i \in \{1, \dots, L\} \mid [\rho(i) = (t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t = 0] \\ \vee [\rho(i) = \neg(t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t \neq 0]\},$$

10

$$M_{L+1} := \vec{1} \in \mathbb{F}_q^r, \vec{v}_{L+1} := (1), \rho(L+1) := \neg(d+1, \vec{v}_{L+1})$$

$$\xi \xleftarrow{\mathcal{U}} \mathbb{F}_q, \{\beta_i\} \xleftarrow{\mathcal{U}} \{(\beta_1, \dots, \beta_{L+1}) \mid \sum_{i=1}^{L+1} \beta_i M_i = \vec{0}\},$$

$$s_0^* := \xi k_0^* + r_0^*, \text{ where } r_0^* \xleftarrow{\mathcal{U}} \text{span}\langle b_{0,3}^* \rangle,$$

for $1 \leq i \leq L+1$,

$$s_i^* := \gamma_i \cdot \xi k_t^* + \beta_i \cdot (\sum_{l=1}^{n_i} y_{i,l} \cdot p_{t,l}^*) + r_i^*,$$

20

$$\text{where } r_i^* \xleftarrow{\mathcal{U}} \text{span}\langle b_{t,2n_i+1}^*, \dots, b_{t,3n_i}^* \rangle,$$

and $\gamma_i, \vec{y}_i := (y_{i,1}, \dots, y_{i,n_i})$ are defined as

$$\text{if } i \in I \wedge \rho(i) = (t, \vec{v}_i), \gamma_i := \alpha_i, \vec{y}_i \xleftarrow{\mathcal{U}} \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 0 \wedge y_{i,1} = 1\},$$

$$\text{if } i \in I \wedge \rho(i) = \neg(t, \vec{v}_i), \gamma_i := \alpha_i / (\vec{v}_i \cdot \vec{x}_t), \vec{y}_i \xleftarrow{\mathcal{U}} \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 1\},$$

$$\text{if } i \notin I \wedge \rho(i) = (t, \vec{v}_i), \gamma_i := 0, \vec{y}_i \xleftarrow{\mathcal{U}} \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 0 \wedge y_{i,1} = 1\},$$

30

$$\text{if } i \notin I \wedge \rho(i) = \neg(t, \vec{v}_i), \gamma_i := 0, \vec{y}_i \xleftarrow{\mathcal{U}} \{\vec{y}_i \mid \vec{y}_i \cdot \vec{v}_i = 1\},$$

$$s_{L+2}^* := \xi(k_{d+2}^* + m \cdot k_{d+3}^*) + r_{L+2}^*, \text{ where } r_{L+2}^* \xleftarrow{\mathcal{U}} \text{span}\langle b_{d+2,5}^*, b_{d+2,6}^* \rangle$$

$$\vec{s}^* := (s_0^*, \dots, s_{L+2}^*), \text{ sig} := \{m, \mathbb{S}, \vec{s}^*\}$$

return sig.

つまり、Setupアルゴリズムにおいて乱数値が設定された秘匿ベクトル $p_{t,l}^*$ を生成し、Sigアルゴリズムでこの秘匿ベクトル $p_{t,l}^*$ を用いて署名ベクトル s^* の要素 s_i^* を生成している点に変更されている。また、Setupアルゴリズムにおいて、基底 $B^{\wedge^*}_t$ を公開パラメータ pkではなく、マスター鍵 skに含め、秘密にしている点に変更されている。これにより、署名処理の安全性が高くなっている。

40

なお、このようにSetupアルゴリズムと、Sigアルゴリズムとを変更した場合であっても、数199に示すように、署名データ sig が正しいデータであれば、アルゴリズムにおいて、ペアリング演算を計算することにより値“1”が得られる。

【数 2 0 0】

$$\begin{aligned}
\prod_{i=0}^{L+2} e(c_i, s_i^*) &= e(c_0, s_0^*) \cdot \prod_{i \in I} e(c_i, k_i^*)^{\xi \gamma_i} \cdot \prod_{i=1}^{L+1} e(c_i, \prod_{\iota=1}^{n_i} (p_{i,\iota}^*)^{\gamma_{i,\iota}} \beta_i) \cdot e(c_{L+2}, s_{L+2}^*) \\
&= g_T^{\xi \delta(-s_0-s_{L+2})} \cdot \prod_{i \in I} g_T^{\xi \delta \alpha_i s_i} \cdot \prod_{i=1}^{L+1} g_T^{\beta_i s_i} \cdot g_T^{\xi \delta s_{L+2}} \\
&= g_T^{\xi \delta(-s_0-s_{L+2})} \cdot g_T^{\xi \delta s_0} \cdot g_T^{\xi \delta s_{L+2}} = 1
\end{aligned}$$

10

【0 1 5 4】

また、V e r アルゴリズムにおいて、暗号要素生成ステップ (S 1 1 0 6) で、 c_0 を生成することなく、 $c_{L+2} := (s_0 - \theta_{L+2} m, \theta_{L+2}, 0, 0, 0, 0, \eta_{L+2, 1}, \eta_{L+2, 2})$ としてもよい。そして、ペアリング演算ステップ (S 1 1 0 7) で、 $\prod_{i=1}^{L+2} e(c_i, s_i^*)$ を計算するとしてもよい。

つまり、上記説明では、 c_0 と c_{L+2} とを乱数 s_{L+2} で結びつけ、ペアリング演算を行った場合に c_0 と c_{L+2} との間で s_{L+2} が打ち消されるようにしていた。しかし、予め s_{L+2} を用いず、処理を簡略化してもよい。

この場合、K e y G e n アルゴリズムにおいて k^*_0 を生成する必要はない。同様に、S i g アルゴリズムにおいて、 s^*_0 を生成する必要はない。

20

【0 1 5 5】

実施の形態 4.

実施の形態 1 では、安全性の証明の観点から、 $i = 1, \dots, L$ の各整数 i についての $\rho(i)$ は、それぞれ異なる識別情報 t についての肯定形の組 $(t, v^{\rightarrow})$ 又は否定形の組 $\neg(t, v^{\rightarrow})$ であるとしてもよいと説明した。つまり、 $\rho \sim$ が単射であるとしてもよいと説明した。しかし、 $\rho \sim$ が単射でなくてもよい。

この場合、安全性の証明の観点から、実施の形態 1 で説明した K e y - P o l i c y 関数型暗号の S e t u p アルゴリズム、K e y G e n アルゴリズム、E n c アルゴリズムを以下のように変更してもよい。ここでは、実施の形態 1 で説明した K e y - P o l i c y 関数型暗号の各アルゴリズムについての変更部分のみを説明する。

30

【0 1 5 6】

この実施の形態において、 ϕ を数 2 0 1 に示す値であるとする。

【数 2 0 1】

$$\varphi \geq \max_{t=1}^d \# \{i \mid \tilde{\rho}(i) = t\}$$

【0 1 5 7】

S e t u p アルゴリズムについては、 $(1^\lambda, \mu^{\rightarrow} := (d; n_1, \dots, n_d))$ が入力された場合に、S e t u p $(1^\lambda, \mu'^{\rightarrow} := (d; n'_1, \dots, n'_d))$ を実行する。ここで、 $t = 1, \dots, d$ の各整数 t について、 $n'_t := n_t + \phi$ である。

40

【0 1 5 8】

K e y G e n アルゴリズムについては、 k^*_i の生成方法を数 2 0 2 のようにする。

【数 2 0 2】

$$\begin{aligned}
 &\text{if } \rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \quad \eta_{i,1}, \dots, \eta_{i,n'_t} \xleftarrow{\mathbb{U}} \mathbb{F}_q \\
 &k_i^* := (\overbrace{(s_i + \theta_i v_{i,1}, \theta_i v_{i,2}, \dots, \theta_i v_{i,n_t}, 0\varphi)}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n'_t}}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t})_{\mathbb{B}_i^*} \\
 &\text{if } \rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}), \quad \eta_{i,1}, \dots, \eta_{i,n'_t} \xleftarrow{\mathbb{U}} \mathbb{F}_q, \\
 &k_i^* := (\overbrace{(s_i(v_{i,1}, \dots, v_{i,n_t}), 0\varphi)}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n'_t}}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t})_{\mathbb{B}_i^*}
 \end{aligned}$$

10

【0 1 5 9】

Enc アルゴリズムについては、 c_t の生成方法を数 2 0 3 のようにする。

【数 2 0 3】

$$c_t := (\overbrace{(\delta(x_{t,1}, \dots, x_{t,n_t}), 0\varphi)}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t}, \overbrace{(\phi_{t,1}, \dots, \phi_{t,n'_t})}_{n'_t})_{\mathbb{B}_t}$$

20

【0 1 6 0】

ここでは、実施の形態 1 で説明した Key-Policy 関数型暗号についてのみ、 ρ が単射でない場合の処理の変更方法について説明した。しかし、同様の考え方に基づき、他の実施の形態で説明した暗号処理や署名処理を変更してもよい。

【0 1 6 1】

実施の形態 5.

以上の実施の形態では、双対ベクトル空間において暗号処理及び署名処理を実現する方法について説明した。この実施の形態では、双対加群において暗号処理及び署名処理を実現する方法について説明する。

30

【0 1 6 2】

つまり、以上の実施の形態では、素数位数 q の巡回群において暗号処理を実現した。しかし、合成数 M を用いて数 2 0 4 のように環 R を表した場合、環 R を係数とする加群においても、上記実施の形態で説明した暗号処理を適用することができる。

【数 2 0 4】

$$\mathbb{R} := \mathbb{Z}/M\mathbb{Z}$$

40

ここで、

$\mathbb{Z}$: 整数

M : 合成数

である。

【0 1 6 3】

例えば、実施の形態 1 で説明した Key-Policy 関数型暗号を、環 R を係数とする加群において実現すると数 2 0 5 から数 2 0 8 のようになる。

【数 2 0 5】

Setup($1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d)$)

$(\text{param}_{\vec{\mu}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0, \dots, d}) \xleftarrow{\mathbb{R}} \mathcal{G}_{\text{ob}}(1^\lambda, \vec{\mu})$

$\hat{\mathbb{B}}_0 := (b_{0,1}, b_{0,3}, b_{0,5}), \quad \hat{\mathbb{B}}_t := (b_{t,1}, \dots, b_{t,n_t}, b_{t,3n_t+1}, \dots, b_{t,4n_t})$

for $t = 1, \dots, d$,

$\hat{\mathbb{B}}_0^* := (b_{0,1}^*, b_{0,3}^*, b_{0,4}^*), \quad \hat{\mathbb{B}}_t^* := (b_{t,1}^*, \dots, b_{t,n_t}^*, b_{t,2n_t+1}^*, \dots, b_{t,3n_t}^*)$

for $t = 1, \dots, d$,

$\text{sk} := \{\hat{\mathbb{B}}_t^*\}_{t=0, \dots, d}, \quad \text{pk} := (1^\lambda, \text{param}_{\vec{\mu}}, \{\mathbb{B}_t\}_{t=0, \dots, d}).$

return sk, pk.

10

ここで、

$\mathcal{G}_{\text{ob}}(1^\lambda, \vec{\mu} := (d; n_1, \dots, n_d)) : \text{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, g, e) \xleftarrow{\mathbb{R}} \mathcal{G}_{\text{bpg}}(1^\lambda),$

20

$\psi \xleftarrow{\mathbb{U}} \mathbb{R},$

$N_0 := 5, \quad N_t := 4n_t \text{ for } t = 1, \dots, d,$

For $t = 0, \dots, d, \quad \text{param}_{\mathbb{V}_t} := (q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, e) := \mathcal{G}_{\text{dpvs}}(1^\lambda, N_t, \text{param}_{\mathbb{G}}),$

$X_t := (\chi_{t,i,j})_{i,j} \xleftarrow{\mathbb{U}} GL(N_t, \mathbb{R}), \quad (\nu_{t,i,j})_{i,j} := \psi \cdot (X_t^T)^{-1},$

$b_{t,i} := (\chi_{t,i,1}, \dots, \chi_{t,i,N_t})_{\mathbb{A}_t} = \sum_{j=1}^{N_t} \chi_{t,i,j} a_{t,j}, \quad \mathbb{B}_t := (b_{t,1}, \dots, b_{t,N_t}),$

$b_{t,i}^* := (\nu_{t,i,1}, \dots, \nu_{t,i,N_t})_{\mathbb{A}_t} = \sum_{j=1}^{N_t} \nu_{t,i,j} a_{t,j}, \quad \mathbb{B}_t^* := (b_{t,1}^*, \dots, b_{t,N_t}^*),$

30

$g_T := e(G, G)^\psi, \quad \text{param}_{\vec{\mu}}^- := (\{\text{param}_{\mathbb{V}_t}\}_{t=0, \dots, d}, g_T)$

return $(\text{param}_{\vec{\mu}}^-, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0, \dots, d}).$

である。

【数 2 0 6】

KeyGen(pk, sk, $\mathbb{S} := (M, \rho)$)

$$\vec{f} \xleftarrow{\mathbb{U}} \mathbb{R}^r, \quad \vec{s}^T := (s_1, \dots, s_L)^T := M \cdot \vec{f}^T, \quad s_0 := \vec{1} \cdot \vec{f}^T,$$

$$\eta_0, \theta_i \xleftarrow{\mathbb{U}} \mathbb{R} (i = 1, \dots, L),$$

$$k_0^* := (-s_0, 0, 1, \eta_0, 0)_{\mathbb{B}_0^*},$$

for $1 \leq i \leq L$

10

$$\text{if } \rho(i) = (t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{R}^{n_i}), \quad \eta_{i,1}, \dots, \eta_{i,n_i} \xleftarrow{\mathbb{U}} \mathbb{R},$$

$$k_i^* := (\overbrace{s_i + \theta_i v_{i,1}, \dots, \theta_i v_{i,n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i})_{\mathbb{B}_i^*}$$

$$\text{if } \rho(i) = \neg(t, \vec{v}_i := (v_{i,1}, \dots, v_{i,n_i}) \in \mathbb{R}^{n_i}), \quad \eta_{i,1}, \dots, \eta_{i,n_i} \xleftarrow{\mathbb{U}} \mathbb{R},$$

$$k_i^* := (\overbrace{s_i(v_{i,1}, \dots, v_{i,n_i})}^{n_i}, \overbrace{0^{n_i}}^{n_i}, \overbrace{\eta_{i,1}, \dots, \eta_{i,n_i}}^{n_i}, \overbrace{0^{n_i}}^{n_i})_{\mathbb{B}_i^*}$$

20

$$\text{sk}_{\mathbb{S}} := (\mathbb{S}, k_0^*, k_1^*, \dots, k_L^*).$$

return $\text{sk}_{\mathbb{S}}$.

【数 2 0 7】

$$\text{Enc}(\text{pk}, m, \Gamma := \{(t, \vec{x}_t := (x_{t,1}, \dots, x_{t,n_t}) \in \mathbb{F}_q^{n_t}) \mid 1 \leq t \leq d\} (x_{t,1} := 1))$$

30

$$\delta, \phi_0, \phi_{t,1}, \dots, \phi_{t,n_t}, \zeta \xleftarrow{\mathbb{U}} \mathbb{R} \text{ such that } (t, \vec{x}_t) \in \Gamma,$$

$$c_0 := (\delta, 0, \zeta, 0, \phi_0)_{\mathbb{B}_0},$$

$$c_t := (\overbrace{\delta(x_{t,1}, \dots, x_{t,n_t})}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{\phi_{t,1}, \dots, \phi_{t,n_t}}^{n_t})_{\mathbb{B}_t} \quad \text{for } (t, \vec{x}_t) \in \Gamma,$$

$$c_{d+1} := g_{\vec{T}}^{\zeta} m, \quad c := (\Gamma, c_0, \{c_t\}_{(t, \vec{x}_t) \in \Gamma}, c_{d+1}).$$

return c .

40

【数 208】

$\text{Dec}(\text{pk}, \text{sk}_S := (\mathbb{S}, k_0^*, k_1^*, \dots, k_L^*), c := (\Gamma, c_0, \{c_t\}_{(t, \vec{x}_t) \in \Gamma}, c_{d+1}))$

If $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{(t, \vec{x}_t)\}$, then compute I and $\{\alpha_i\}_{i \in I}$ such that

$$s_0 = \sum_{i \in I} \alpha_i s_i, \text{ and } I \subseteq \{i \in \{1, \dots, L\} \mid [\rho(i) = (t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t = 0]$$

$$\vee [\rho(i) = \neg(t, \vec{v}_i) \wedge (t, \vec{x}_t) \in \Gamma \wedge \vec{v}_i \cdot \vec{x}_t \neq 0]\}.$$

10

$$K := e(c_0, k_0^*) \cdot \prod_{i \in I \wedge \rho(i) = (t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i} \cdot \prod_{i \in I \wedge \rho(i) = \neg(t, \vec{v}_i)} e(c_t, k_i^*)^{\alpha_i / (\vec{v}_i \cdot \vec{x}_t)}$$

$$m' = c_{d+1} / K.$$

return m' .

【0164】

ここでは、実施の形態1で説明したKey-Policy関数型暗号についてのみ、環Rを係数とする加群において実現する方法を示した。しかし、原則として、以上の実施の形態で体 F_q として説明した処理を、環Rに置き換えることにより、以上の実施の形態で説明した他の暗号処理や署名処理についても、環Rを係数とする加群において実現できる。

20

【0165】

なお、以上の実施の形態におけるSetupアルゴリズムは、暗号処理システム10あるいは署名処理システム20のセットアップ時に一度実行すればよく、復号鍵を生成する度に実行する必要はない。また、上記説明では、SetupアルゴリズムとKeyGenアルゴリズムとを鍵生成装置100が実行するとしたが、SetupアルゴリズムとKeyGenアルゴリズムとをそれぞれ異なる装置が実行するとしてもよい。

【0166】

また、以上の実施の形態では、スパンプログラム $M^\wedge$ は、入力列 δ によって行列 $M^\wedge$ から得られる行列 M_δ の行を線形結合して $1^\rightarrow$ が得られる場合に限り、入力列 δ を受理するとした。しかし、スパンプログラム $M^\wedge$ は、 $1^\rightarrow$ ではなく、他のベクトル $v^\rightarrow$ が得られる場合に限り、入力列 δ を受理するとしてもよい。

30

この場合、KeyGenアルゴリズムにおいて $s_0 := 1^\rightarrow \cdot f^{\rightarrow T}$ ではなく、 $s_0 := v^\rightarrow \cdot f^{\rightarrow T}$ とすればよい。同様に、Sigアルゴリズムにおいて α_i を計算する際、 $\sum_{i \in I} \alpha_i M_i = v^\rightarrow$ となる α_i を計算すればよい。

【0167】

また、上記説明における暗号処理は、権限の委譲を行うことも可能である。権限の委譲とは、復号鍵を有する者がその復号鍵よりも権限の弱い下位の復号鍵を生成することである。ここで、権限が弱いとは、復号できる暗号化データが限定されるという意味である。

40

例えば、第1階層目（最上位）においては、 $t=1$ の基底 B_t と基底 B_t^* とを用い、第2階層目においては、 $t=1, 2$ の基底 B_t と基底 B_t^* とを用い、 $\dots$ 、第 k 階層目においては、 $t=1, \dots, k$ の基底 B_t と基底 B_t^* とを用いる。用いる基底 B_t と基底 B_t^* とが増える分、属性情報が多く設定されることになる。したがって、より復号鍵の権限が限定されることになる。

【0168】

次に、実施の形態における暗号処理システム10（鍵生成装置100、暗号化装置200、復号装置300、）と、署名処理システム20（鍵生成装置100、署名装置400、検証装置500）のハードウェア構成について説明する。

図22は、鍵生成装置100、暗号化装置200、復号装置300、署名装置400、

50

検証装置５００のハードウェア構成の一例を示す図である。

図２２に示すように、鍵生成装置１００、暗号化装置２００、復号装置３００、署名装置４００、検証装置５００は、プログラムを実行するＣＰＵ９１１（Ｃｅｎｔｒａｌ・Ｐｒｏｃｅｓｓｉｎｇ・Ｕｎｉｔ、中央処理装置、処理装置、演算装置、マイクロプロセッサ、マイクロコンピュータ、プロセッサともいう）を備えている。ＣＰＵ９１１は、バス９１２を介してＲＯＭ９１３、ＲＡＭ９１４、ＬＣＤ９０１（Ｌｉｑｕｉｄ　Ｃｒｙｓｔａｌ　Ｄｉｓｐｌａｙ）、キーボード９０２（Ｋ／Ｂ）、通信ボード９１５、磁気ディスク装置９２０と接続され、これらのハードウェアデバイスを制御する。磁気ディスク装置９２０（固定ディスク装置）の代わりに、光ディスク装置、メモリカード読み書き装置などの記憶装置でもよい。磁気ディスク装置９２０は、所定の固定ディスクインタフェースを介して接続される。

10

【０１６９】

ＲＯＭ９１３、磁気ディスク装置９２０は、不揮発性メモリの一例である。ＲＡＭ９１４は、揮発性メモリの一例である。ＲＯＭ９１３とＲＡＭ９１４と磁気ディスク装置９２０とは、記憶装置（メモリ）の一例である。また、キーボード９０２、通信ボード９１５は、入力装置の一例である。また、通信ボード９１５は、通信装置（ネットワークインタフェース）の一例である。さらに、ＬＣＤ９０１は、表示装置の一例である。

【０１７０】

磁気ディスク装置９２０又はＲＯＭ９１３などには、オペレーティングシステム９２１（ＯＳ）、ウィンドウシステム９２２、プログラム群９２３、ファイル群９２４が記憶されている。プログラム群９２３のプログラムは、ＣＰＵ９１１、オペレーティングシステム９２１、ウィンドウシステム９２２により実行される。

20

【０１７１】

プログラム群９２３には、上記の説明において「マスター鍵生成部１１０」、「マスター鍵記憶部１２０」、「情報入力部１３０」、「復号鍵生成部１４０」、「鍵配布部１５０」、「公開パラメータ取得部２１０」、「情報入力部２２０」、「暗号化データ生成部２３０」、「データ送信部２４０」、「復号鍵取得部３１０」、「データ受信部３２０」、「スパンプログラム計算部３３０」、「補完係数計算部３４０」、「ペアリング演算部３５０」、「平文情報計算部３６０」、「署名鍵取得部４１０」、「情報入力部４２０」、「補完係数計算部４３０」、「署名生成部４５０」、「データ送信部４６０」、「公開パラメータ取得部５１０」、「データ受信部５２０」、「暗号化データ生成部５３０」、「ペアリング演算部５４０」等として説明した機能を実行するソフトウェアやプログラムやその他のプログラムが記憶されている。プログラムは、ＣＰＵ９１１により読み出され実行される。

30

ファイル群９２４には、上記の説明において「公開パラメータｐｋ」、「マスター鍵ｓｋ」、「暗号化データｃ」、「復号鍵ｓｋ_s」、「復号鍵ｓｋ_r」、「アクセスストラクチャＳ」、「属性の集合Γ」、「メッセージｍ」、「署名データｓｉｇ」等の情報やデータや信号値や変数値やパラメータが、「ファイル」や「データベース」の各項目として記憶される。「ファイル」や「データベース」は、ディスクやメモリなどの記録媒体に記憶される。ディスクやメモリなどの記憶媒体に記憶された情報やデータや信号値や変数値やパラメータは、読み書き回路を介してＣＰＵ９１１によりメインメモリやキャッシュメモリに読み出され、抽出・検索・参照・比較・演算・計算・処理・出力・印刷・表示などのＣＰＵ９１１の動作に用いられる。抽出・検索・参照・比較・演算・計算・処理・出力・印刷・表示のＣＰＵ９１１の動作の間、情報やデータや信号値や変数値やパラメータは、メインメモリやキャッシュメモリやバッファメモリに一時的に記憶される。

40

【０１７２】

また、上記の説明におけるフローチャートの矢印の部分は主としてデータや信号の入出力を示し、データや信号値は、ＲＡＭ９１４のメモリ、その他光ディスク等の記録媒体やＩＣチップに記録される。また、データや信号は、バス９１２や信号線やケーブルその他の伝送媒体や電波によりオンライン伝送される。

50

また、上記の説明において「～部」として説明するものは、「～回路」、「～装置」、「～機器」、「～手段」、「～機能」であってもよく、また、「～ステップ」、「～手順」、「～処理」であってもよい。また、「～装置」として説明するものは、「～回路」、「～機器」、「～手段」、「～機能」であってもよく、また、「～ステップ」、「～手順」、「～処理」であってもよい。さらに、「～処理」として説明するものは「～ステップ」であっても構わない。すなわち、「～部」として説明するものは、ROM 913に記憶されたファームウェアで実現されていても構わない。或いは、ソフトウェアのみ、或いは、素子・デバイス・基板・配線などのハードウェアのみ、或いは、ソフトウェアとハードウェアとの組み合わせ、さらには、ファームウェアとの組み合わせで実施されても構わない。ファームウェアとソフトウェアは、プログラムとして、ROM 913等の記録媒体に記憶される。プログラムはCPU 911により読み出され、CPU 911により実行される。すなわち、プログラムは、上記で述べた「～部」としてコンピュータ等を機能させるものである。あるいは、上記で述べた「～部」の手順や方法をコンピュータ等に行わせるものである。

【符号の説明】

【0173】

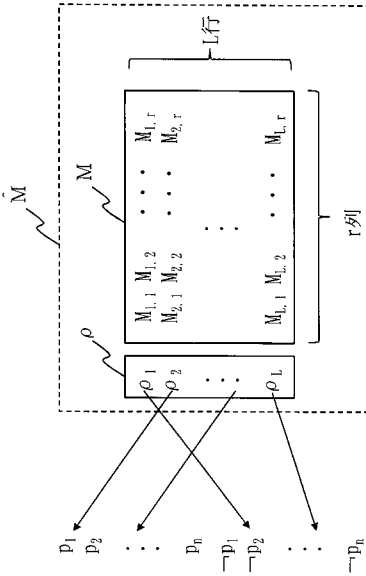
10 暗号処理システム、20 署名処理システム、100 鍵生成装置、110 マスター鍵生成部、120 マスター鍵記憶部、130 情報入力部、140 復号鍵生成部、141 fベクトル生成部、142 sベクトル生成部、143 乱数生成部、144 鍵要素生成部、145 秘匿要素生成部、150 鍵配布部、160 署名鍵生成部、161 乱数生成部、162 鍵要素生成部、200 暗号化装置、210 公開パラメータ取得部、220 情報入力部、230 暗号化データ生成部、231 乱数生成部、232 暗号要素生成部、233 fベクトル生成部、234 sベクトル生成部、240 データ送信部、300 復号装置、310 復号鍵取得部、320 データ受信部、330 スパンプログラム計算部、340 補完係数計算部、350 ペアリング演算部、360 平文情報計算部、400 署名装置、410 署名鍵取得部、420 情報入力部、430 補完係数計算部、440 行列生成部、450 署名生成部、451 乱数生成部、452 署名要素生成部、460 データ送信部、500 検証装置、510 公開パラメータ取得部、520 データ受信部、530 暗号化データ生成部、540 ペアリング演算部、531 乱数生成部、532 暗号要素生成部、533 fベクトル生成部、534 sベクトル生成部。

10

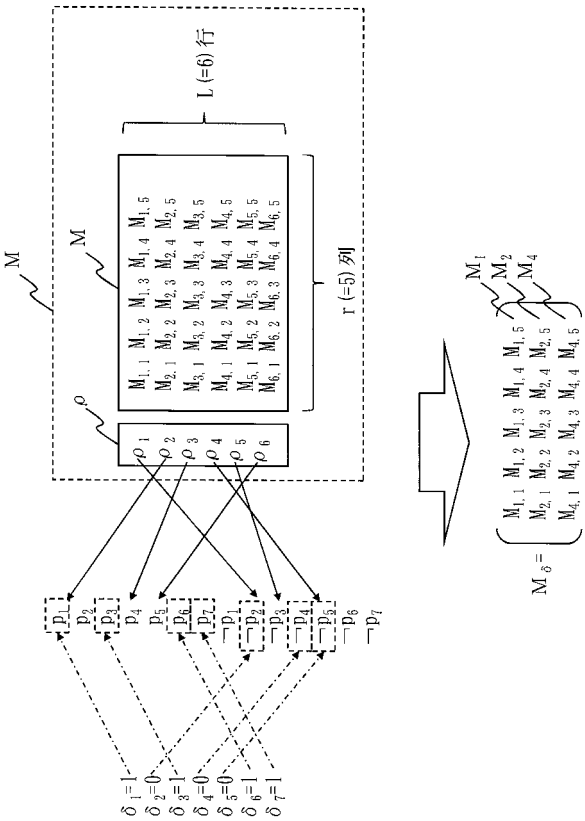
20

30

【図 1】



【図 2】



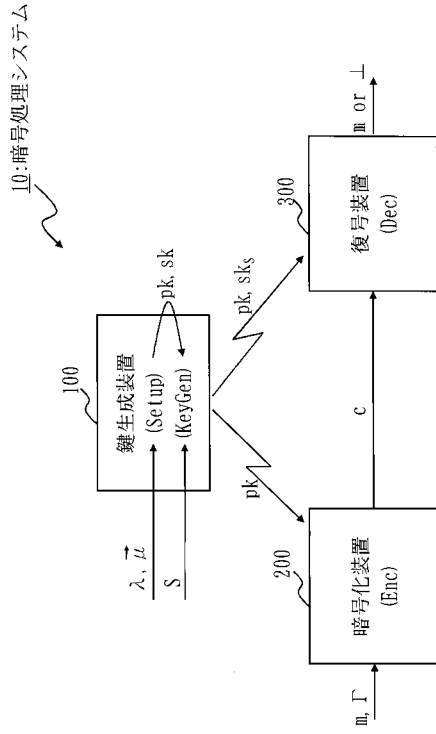
【図 3】

$$s_0 = \left\{ \begin{matrix} f_1 \\ \vdots \\ f_r \end{matrix} \right\} = \sum_{k=1}^r f_k$$

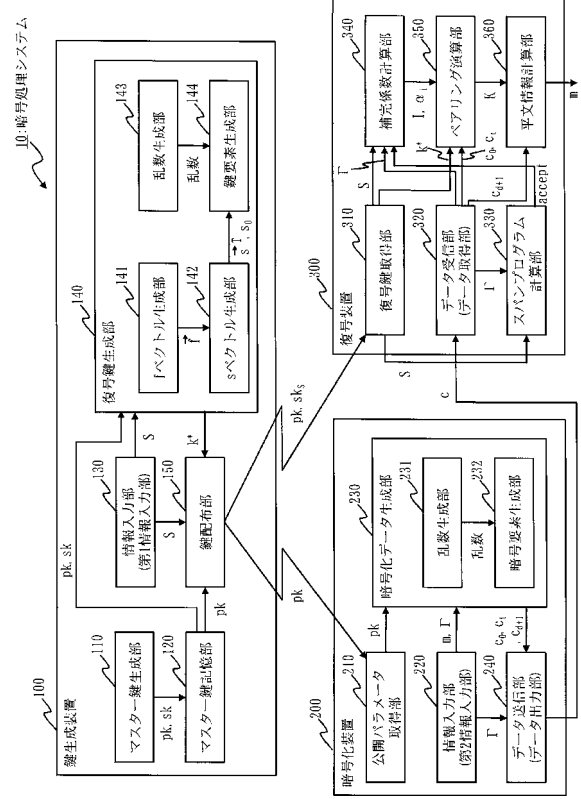
【図 4】

$$s^T = \begin{pmatrix} M_{1,1} & M_{1,2} & \dots & M_{1,r} \\ M_{2,1} & M_{2,2} & \dots & M_{2,r} \\ \vdots & \vdots & \ddots & \vdots \\ M_{L,1} & M_{L,2} & \dots & M_{L,r} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_r \end{pmatrix} = \begin{pmatrix} s_1 \\ \vdots \\ s_r \end{pmatrix}$$

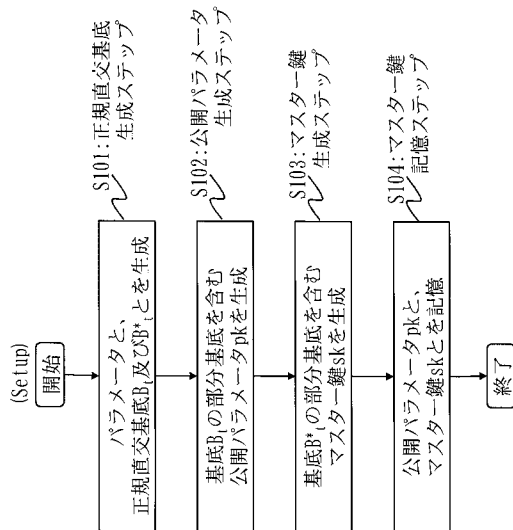
【図 5】



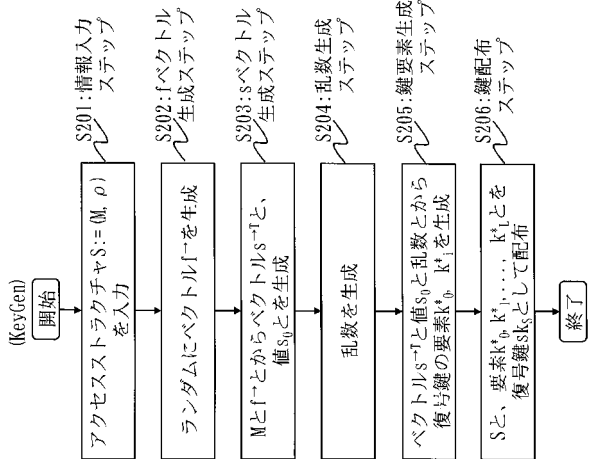
【図 6】



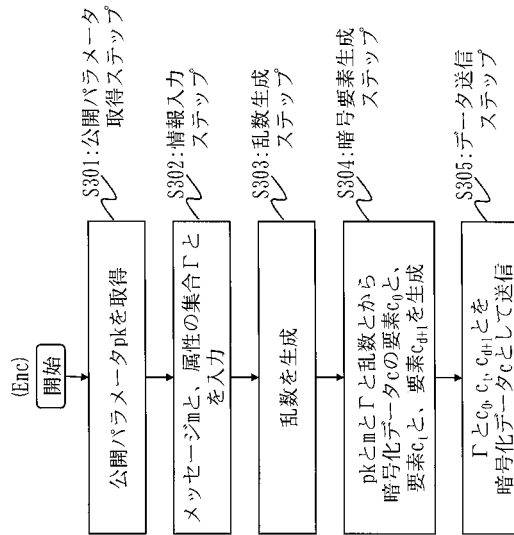
【図 7】



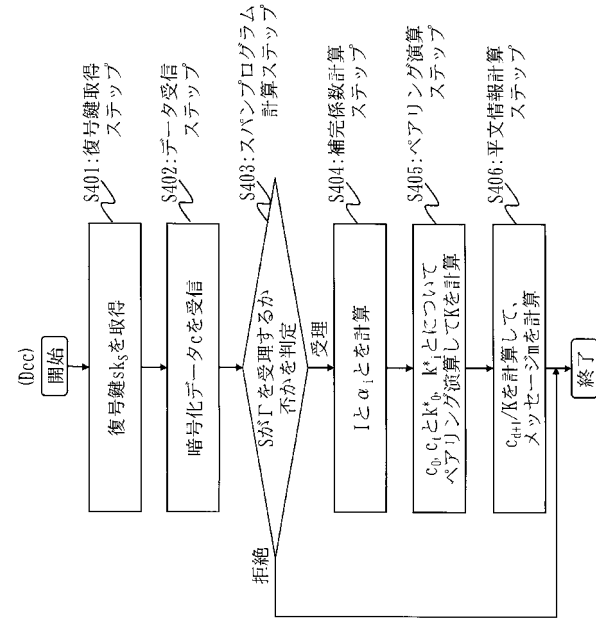
【図 8】



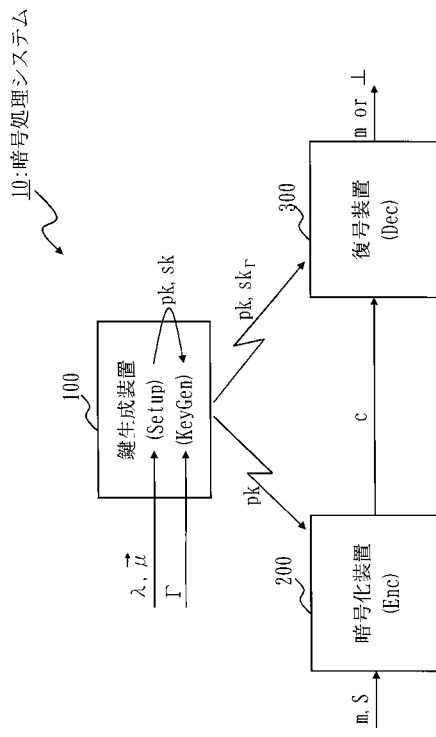
【図 9】



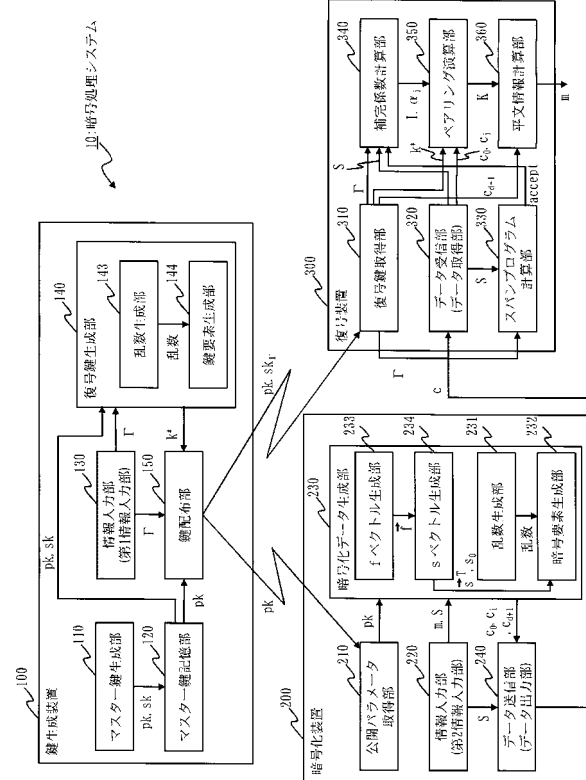
【図 10】



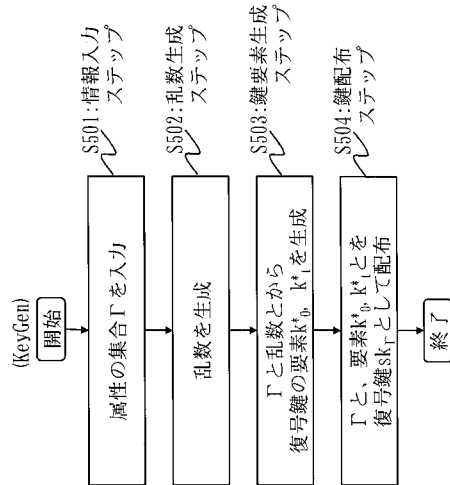
【図 11】



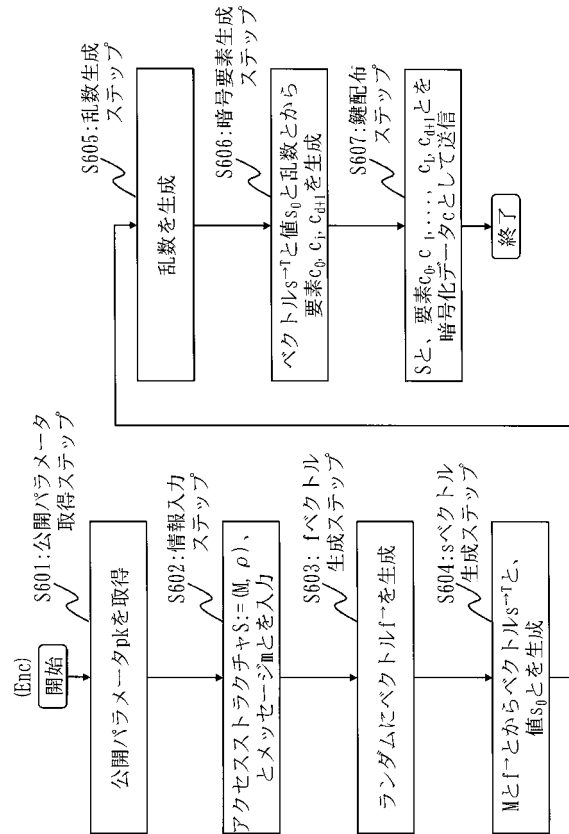
【図 12】



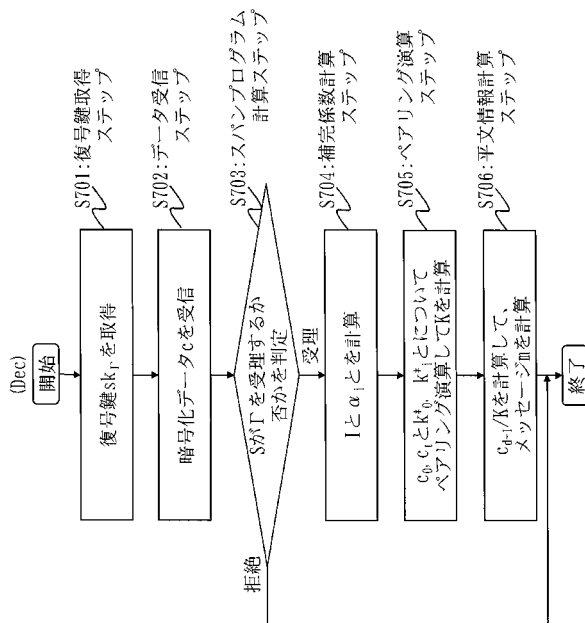
【図 13】



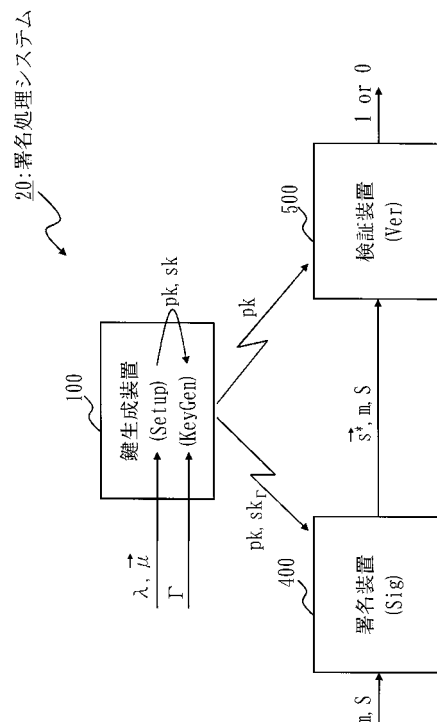
【図 14】



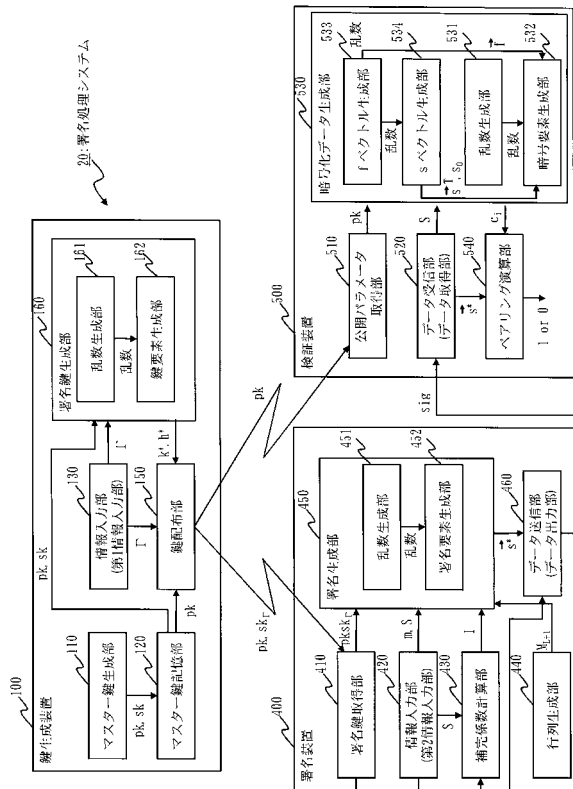
【図 15】



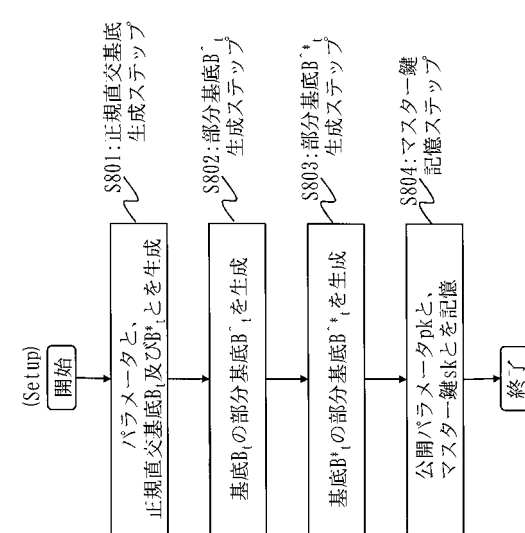
【図 16】



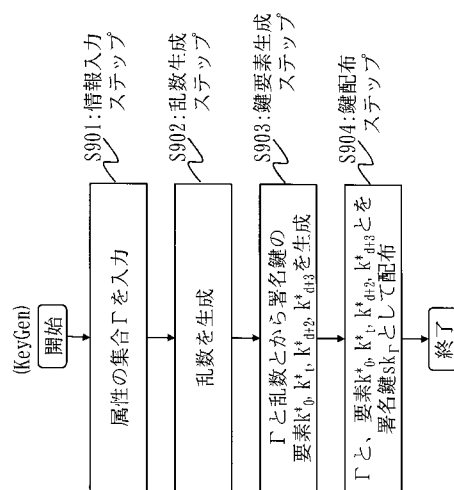
【図 17】



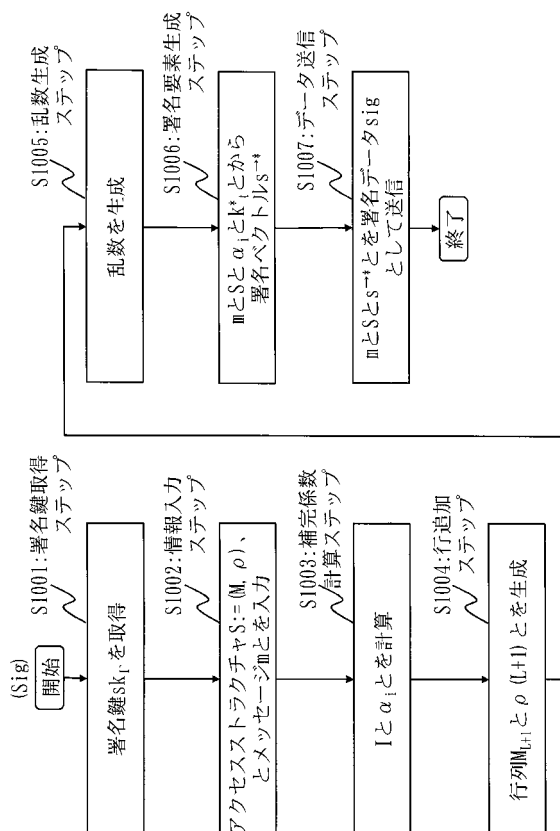
【図 18】



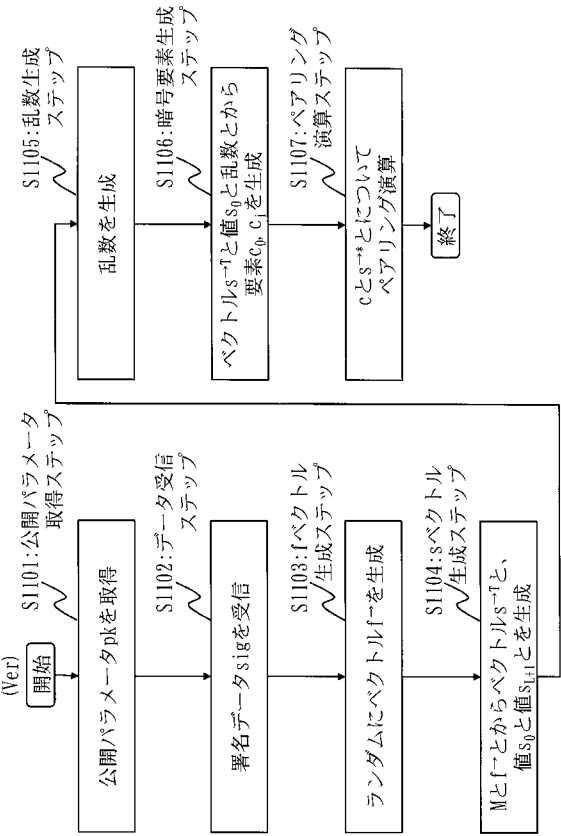
【図 19】



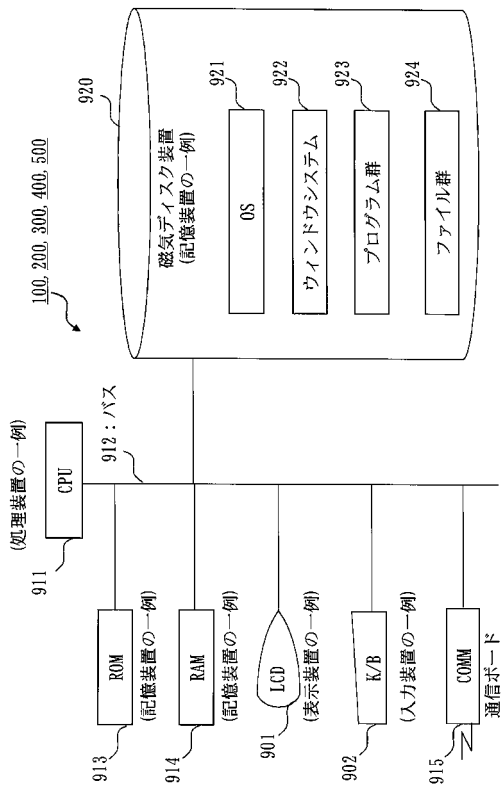
【図 20】



【図 2 1】



【図 2 2】



フロントページの続き

(72)発明者 岡本 龍明

東京都千代田区大手町二丁目3番1号 日本電信電話株式会社内

審査官 青木 重徳

(56)参考文献 Allison Lewko, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, Brent Waters, “Fully Secure Functional Encryption: Attribute-Based Encryption and (Hierarchical) Inner Product Encryption”, Cryptology ePrint Archive: Report 2010/110, [online], 2010年3月29日, Version: 20100329:171109, p.1-56, [retrieved on 2013-10-21]. Retrieved from the Internet, URL, <eprint.iacr.org/eprint-bin/getfile.pl?entry=2010/110&version=20100329:171109&file=110.pdf>

Brent Waters, “Dual System Encryption: Realizing Fully Secure IBE and HIBE under Simple Assumptions”, Cryptology ePrint Archive: Report 2009/385, [online], 2009年8月10日, Version: 20090810:215827, p.1-37, [retrieved on 2013-10-21]. Retrieved from the Internet, URL, <<http://eprint.iacr.org/2009/385.pdf>>

Tatsuaki Okamoto and Katsuyuki Takashima, “Fully Secure Functional Encryption with General Relations from the Decisional Linear Assumption”, Cryptology ePrint Archive: Report 2010/563, [online], 2010年11月5日, Version: 20101105:113344, p.1-49, [retrieved on 2013-10-21]. Retrieved from the Internet, URL, <eprint.iacr.org/eprint-bin/getfile.pl?entry=2010/563&version=20101105:113344&file=563.pdf>

(58)調査した分野(Int.Cl., DB名)

G09C 1/00

JSTPlus/JMEDPlus/JST7580 (JDreamIII)

IEEE Explore