SCIS 2012 The 29th Symposium on
Cryptography and Information Security
Kanazawa, Japan, Jan. 30 - Feb. 2, 2012
The Institute of Electronics,
Information and Communication Engineers

関数型暗号:実装の現在と実用化への展望

Functional Encryption: Current State of Implementation and Prospects for Practical Applications

高橋 克巳* Katsumi Takahashi
星野 文学* Fumitaka Hoshino
小林 鉄太郎* Tetsutaro Kobayashi
山本 剛* Go Yamamoto
山本 具英* Tomohide Yamamoto
宮澤 俊之* Toshiyuki Miyazawa
吉田 麗生* Reo Yoshida
冨士 仁* Hitoshi Fuji
横森 正利* Masatoshi Yokomori
永井 彰*† Akira Nagai

あらまし 関数型暗号は公開鍵暗号の一種で，暗号-復号のメカニズムの中にロジック（述語）を組み込むことが可能である特徴がある．現代のクラウドコンピューティングに代表される自由度の高い情報環境において，情報の受け手を「自由に正しく指定」することができると期待されている暗号である．

関数型暗号は属性ベース暗号や述語暗号を含む ID ベース暗号の一般化概念として近年活発に研究されてきた．従来，属性ベース暗号のような任意の述語を記述できる関数型暗号は selective secure のものしか実現出来ず，こうした暗号プリミティブは限られた安全性の下でしか利用できなかったが，Crypto 2010 において岡本らが標準的な暗号学的仮定の下，任意の述語に関して適応的 payload-hiding な関数型暗号を提案した事により，安全性に関する大きな障害が取り除かれた．

本発表の前半では，関数型暗号の概念および構成について概説し，典型的な使用方法とネットワークで必要となる機構を説明する．さらに岡本らの関数型暗号を，x86 プロセッサ上で実装した結果とその評価について報告する．後半では，関数型暗号の応用に関して，どのような応用が可能か具体的に考え，加えて普及に向けて解決すべき課題について考察する．

キーワード Functional Encryption, Attribute-Based Encryption, Predicate Encryption, Implementation

1 はじめに

関数型暗号の実装および実用化に関する現状について，NTT 研究所での取り組みを中心に報告する．

関数型暗号は公開鍵暗号の一種で，暗号-復号のメカニズムの中にロジックを組み込むことができる暗号である．この特徴を用いると，例えば文書の流通において，秘密にしたい文書を，その文書への復号条件，すなわち「その文書を如何なる条件下で開けてよいか」を定めたロジックを含めた形式で暗号化することが，極めて自由度高くかつ安全性を担保した形で可能になる．この性質は従来の暗号の使われ方の枠組みだけでなく，さまざまな目的の情報処理業務に新しい手段を与える可能性を持っていると考えている．

本稿では，関数型暗号に関する一般的な概念，定義，利用のモデルの解説を行い，NTT 研究所で行っている実装と応用の検討状況を報告し，さらに普及への課題について NTT も参加しているフォーラムの活動も参照しながら考察する．本稿で NTT の取り組みとして述べる関数型暗号は，具体的には 2010 年に岡本-高島が提案した [7]，および同年に NTT と三菱電機が共同で報道発表したインテリジェント暗号 [15] に基づいている．

2 概念と定義

2.1 関数型暗号の概念

関数型暗号 (functional encryption) は ID ベース暗号の拡張概念である．ID ベース暗号とは，公開鍵の照会と

* 日本電信電話株式会社 NTT 情報流通プラットフォーム研究所, 〒180-8585 東京都武蔵野市緑町 3-9-11, NTT Information Sharing Platform Laboratories, NTT Corporation, 3-9-11 Midori-cho, Musashino-shi, Tokyo, 180-8585 Japan
† 現 東日本電信電話株式会社

いった予備通信を必要とせず, 受信者識別子 (ID) を公開鍵として暗号化が可能な公開鍵暗号系のことである. ID ベース暗号では, 暗号化時に指定される受信者識別子 id_r と鍵生成時に指定される鍵識別子 id_k が一致するような暗号文と秘密鍵の組み合わせで復号を行なうと元の平文に復号出来る. ID ベース暗号では公開鍵 (受信者識別子) と秘密鍵は必ず 1:1 に対応している.

ここでこの 1:1 の関係ではなく, 文字列 id_k と 文字列 id_r が何らかの特別な関係を満たす場合に復号出来るよう拡張を考える. 例えば一般的な識別子の代わりに, 顔写真のような曖昧なデータを使用したいとする. このような時は $id_k = id_r$ という等式の代わりに $id_k \approx id_r$ (何らかの意味で識別子が近い) という関係が満たされる場合に復号出来る事が望まれる. このとき, 鍵識別子 id_k に対応する受信者識別子 id_r は必ずしも一意ではない.

さらに 1 つの秘密鍵がいろいろな公開鍵 (受信者識別子) に対応したり, 1 つの公開鍵 (受信者識別子) がいろいろな秘密鍵に対応して, 暗号系が構成される拡張を考える. このような一般化を進めていくと, 鍵識別子 id_k と受信者識別子 id_r との間に, ある特別な関係 R が満たされる場合, 即ち $R(id_k, id_r) = \mathrm{True}$ なる場合にのみ復号出来る暗号の概念が生まれる. (関係とは 2 つの文字列を入力とし, 真理値を出力する関数のことを指す.) 関係 R はアプリケーション毎に異なる関数であり, アプリケーションを設計する度に暗号を設計し直す事は現実的ではないので, さまざまな目的に応用可能な万能の R を実現する暗号概念に到達する. このような暗号を実現する関数型暗号の概念が提案されている [9, 4, 8].

2010 年岡本らは, AND，OR，NOT，閾値ゲートにより構成される関係式をすべて含むような, 現時点で考え得る最も一般的な機能を実現できる安全な関数型暗号を提案した [7]. この型の関数型暗号においては, 鍵識別子 id_k あるいは受信者識別子 id_r のどちらか一方を様々な属性の集合と見なす. そして, もう一方の識別子を属性変数で記述された述語と見なす. 関係 R は属性変数の具体的な値の集合が述語を充足する場合にのみ True を返す.

例えば, 暗号文の送信者は受信者を指定する際, 属性の条件式 (述語) を指定して暗号化する事ができる. この時, 暗号文の受信者は自身が持つ属性に対応する秘密鍵を所持しており, 属性が述語を充足すれば暗号文を正しく復号することができる. こうした暗号を用いると, 暗号化するだけで文書に「部長または人事部の課長だけが閲覧できる」といったきめ細かい開示制御の設定をすることが可能となり, クラウドのような共有ストレージ上の情報管理の手間を大幅に単純化する事ができる [5, 1, 4].

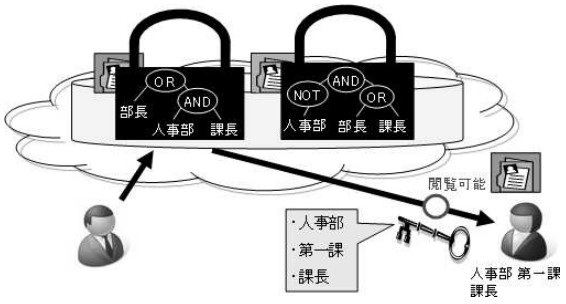

図 1: 企業内機密情報管理システムの利用イメージ

2.2 関数型暗号の定義

関数型暗号が ID ベース暗号の拡張概念であることを述べたが，関数型暗号と ID ベース暗号には構文の違いはない. 即ち関数型暗号も ID ベース暗号と同様に以下の 4 つのアルゴリズム ($\mathtt{Setup}$, $\mathtt{KeyGen}$, $\mathtt{Enc}$, $\mathtt{Dec}$) より構成される.

- $\mathtt{Setup}(1^\lambda) \xrightarrow{\$} (P, K)$: セットアップ— セキュリティパラメータ 1^λ を入力とし公開パラメータ P とマスタ鍵 K を出力する確率的多項式時間アルゴリズム.
- $\mathtt{KeyGen}(K, x) \xrightarrow{\$} k_x$: 鍵生成 — マスタ鍵 K と鍵識別子 x を入力とし, x に対応する秘密鍵 k_x を出力する確率的多項式時間アルゴリズム.
- $\mathtt{Enc}(P, y) \xrightarrow{\$} c_y$: 暗号化 — 公開パラメータ P と暗号文識別子 y を入力とし, 暗号文 c_y を出力する確率的多項式時間アルゴリズム.
- $\mathtt{Dec}(k_x, c_y) \xrightarrow{\$} m$: 復号 — 秘密鍵 k_x と暗号文 c_y を入力とし, 文字列 m を出力する確率的多項式時間アルゴリズム.

関数型暗号では, 正当性の定義が ID ベース暗号から拡張されており, 暗号文の受信者は $\mathtt{KeyGen}$ の入力文字列 x に対応する秘密鍵 k_x と $\mathtt{Enc}$ の入力文字列 y に対応する暗号文 c_y から何らかの関数 $f(x, y)$ を評価する事が出来るようになっている. 即ち, ある関数 $f(\cdot, \cdot)$ が存在し $\forall x, \forall y \in \{0,1\}^{\mathtt{poly}(\lambda)}$ に対して

$$\Pr\left[m = f(x,y) \;\middle|\; \begin{array}{l} (P,K) \xleftarrow{\$} \mathtt{Setup}(1^\lambda); \\ k_x \xleftarrow{\$} \mathtt{KeyGen}(K,x); \\ c_y \xleftarrow{\$} \mathtt{Enc}(P,y); \\ m \xleftarrow{\$} \mathtt{Dec}(k_x,c_y); \end{array} \right]$$

なる確率が λ に関して圧倒的であるとき, 関数型暗号 $(\mathtt{Setup}, \mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Dec})$ は f に関し正当であると云う [4, 8]. 特に, ある関係 $R(\cdot, \cdot)$ が存在し,

$$f(i, j \| m) = \begin{cases} m & (R(i,j) = 1 \text{ の時}) \\ \perp & (R(i,j) = 0 \text{ の時}) \end{cases}$$

なる型の f を持つ関数型暗号類は様々な暗号を包含している [4, 8, 7, 6]. 例えば

$$f(i,j\|m) = \begin{cases} m & (i=j \text{ の時}) \\ \perp & (i \neq j \text{ の時}) \end{cases}$$

を利用して ID ベース暗号を関数型暗号の一種として再定義する事ができる.

関数型暗号は，$R(i,j)$ の引数 i,j のどちらを述語／述語変数の値にするかによって，大きく 2 つに分類される：

i を述語, j を述語変数の具体的な値として,

$$f(i,j\|m) = \begin{cases} m & (j \text{ が述語 } i \text{ を充足する時}) \\ \perp & (j \text{ が述語 } i \text{ を充足しない時}) \end{cases}$$

なる f を持つ関数型暗号は KP-FE (key-policy functional encryption; 復号鍵に属性情報，暗号文に条件式) と呼ばれる. また，j を述語, i を述語変数の具体的な値として,

$$f(i,j\|m) = \begin{cases} m & (i \text{ が述語 } j \text{ を充足する時}) \\ \perp & (i \text{ が述語 } j \text{ を充足しない時}) \end{cases}$$

なる f を持つ関数型暗号は CP-FE (ciphertext-policy functional encryption; 暗号文に属性情報，復号鍵に条件式) と呼ばれる.

関数型暗号の安全性　任意の述語を記述できる関数型暗号は，2009 年以前には選択的安全 (selectivly secure) な方式しか知られていなかった [6, 7].

選択的安全とは，攻撃者が攻撃開始前に攻撃対象の受信者を決定するモデルの下で秘匿性が証明できる安全性である．このモデルは条件が強く，攻撃開始後に受信者を決定可能なモデルの下でも秘匿性を証明できる（適応的安全 : adaptively secure）ことが，より現実的な安全性であろう.

そのため，選択的安全な関数型暗号は攻撃モデルが限定できるアプリケーションでしか利用できず, アプリケーションの部品としては使いにくいという側面があった. しかし，岡本-高島により適応的安全な関数型暗号が提案 [7] されたことにより，この制約は解消され，関数型暗号がより多くのアプリケーションで利用できるようになった.

3　構成モデル

本節では，関数型暗号の構成モデルと，暗号化までの処理フローを説明する．実際の利用を想定した場合，関数型暗号は以下の 3 者からなる構成をとるのが妥当と考えられる.

- **TA(Trusted Authority)**:セットアップ関数 `Setup` と鍵生成関数 `KeyGen` を実行するエンティティ.

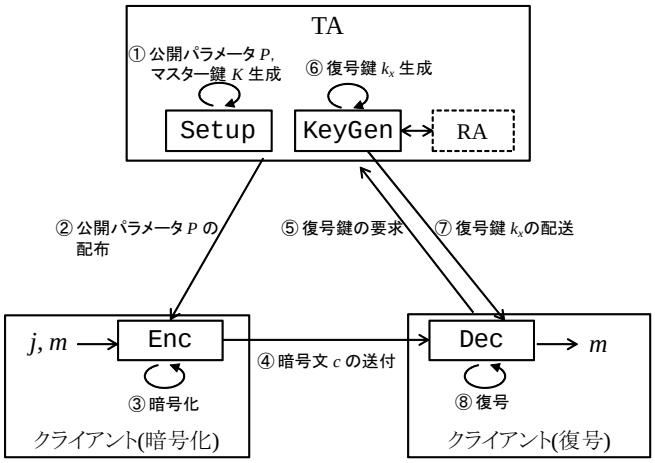

図 2: 関数型暗号の構成モデルと処理フロー

`Setup` の出力である公開パラメータ P を公開し，クライアントからの要求に応じて `KeyGen` を実行し復号鍵を発行する．(以上は鍵発行局に相当)．また，復号鍵を発行する前にクライアントの本人性や属性を確認するエンティティ(登録局：RA) と連携する場合もある.

- **クライアント (送信者)**：暗号化関数 `Enc` を実行するエンティティ．TA から公開パラメータを受領し，復号可能となる属性や条件を定めてから，`Enc` の出力である暗号文を受信者に送付する.
- **クライアント (受信者)**：復号関数 `Dec` を実行するエンティティ．TA から自身の属性や条件に対応する復号鍵を受領し，暗号文に対して `Dec` を実行して平文を復号する.

ただし，この三者は必ずしも別である必要はない．適用するアプリケーションによっては，暗号化と復号を同一のクライアントで行う場合なども考えられる．たとえば，クラウド上のストレージサービスに関数型暗号を用いて暗号化したデータをアップロードし，場所（IP アドレスや GPS の位置）や時刻に応じて復号するようなアプリケーションはその一例である.

上記モデルに沿って，関数型暗号を用いた暗号化・復号の流れを説明する．なお，理解の容易性のため，CP-FE の場合で説明する (図 2 参照).

1. TA が `Setup` 関数を用いて，公開パラメータ P とマスタ鍵 K を作成する.
2. TA が P を公開する.
3. 送信者は復号可能となる条件を定めた述語 j を作成し，P, j および，平文 m から `Enc` 関数により暗号文 c を作成する.

4. 送信者は 暗号文 c を受信者に送信する.

5. 受信者は TA に対して自身の属性 i に関するを復号鍵を要求する.

6. TA は K を用いて KeyGen 関数により，受信者の属性 i に対応する復号鍵 k_x を作成する.

7. TA は受信者に対して，復号鍵 k_x を送付する.

8. 受信者は，P と k_x を用いて，Dec 関数により，c から m を復号する．なお，属性 i が述語 j を充足しない場合は m を復号することはできない.

なお，上記 7 の復号鍵の配送は，TSL を用いた秘匿通信路を使うなど，復号鍵が第三者に渡らないような手段で配送しなくてはならない.

また，関数型暗号における秘密鍵発行のタイミングは，公開鍵暗号のそれとは異なる．公開鍵暗号の場合は，暗号化鍵と復号鍵が同時に生成されるため，送信者は受信者の鍵生成が完了する前には，暗号文を作成することができない．一方，関数型暗号は，公開パラメータが入手できれば，受信者の鍵生成のタイミングとは独立に暗号文を作成することができる．この点は，ID ベース暗号や属性ベース暗号と同じである.

4 実装

ここでは岡本らの関数型暗号 [7] について NTT で行っている実装とその性能に関して簡単に記述する.

岡本らの関数型暗号 [7] には

(1) 属性ベース暗号に基づく述語記述機構
(2) 内積述語暗号に基づく述語記述機構

の二種類の述語記述機構がある．(2) は (1) より低位のレイヤーに位置し, (2) で記述された複数の述語を (1) で合成して一つの大きな述語を構成する．(1) については一般的な属性ベース暗号の設計 [1] と大きく異なる事はない．(2) の部分を自然に (1) に取り込めるよう幾らかの拡張を行い, 概ね次のような機能群を構成する事により C 言語によるソフトウェア実装を行なった.

- 属性・述語を構文木に変換するコンパイラ
- ペアリング等のパラメタ管理
- 変数名等の管理
- 不等号などの述語マクロの展開
- ド・モルガンの法則による単調化
- 内積述語暗号の属性・述語の処理
- 線形スパンプログラム変換
- 復号時の線形代数
- 双対ペアリングベクトル空間の処理
- ペアリング・楕円スカラ倍

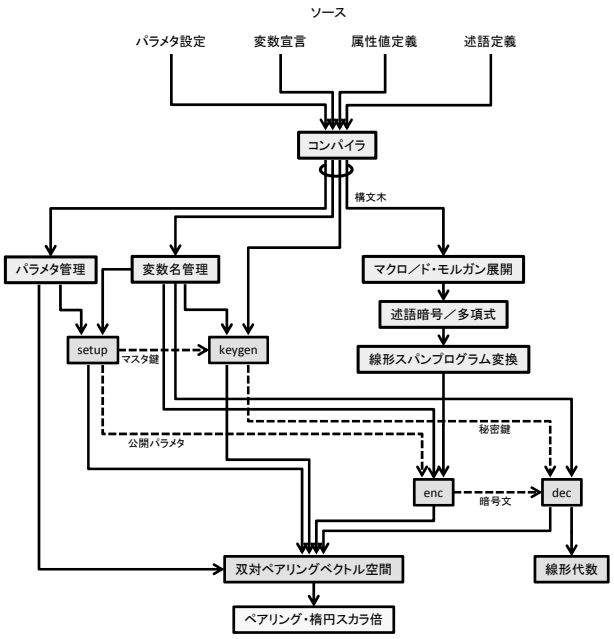

図 3: 機能構成とデータフロー

- Setup, KeyGen, Enc, Dec

モジュール構成とデータフローの概略図を図 3 に記載する．使用するペアリングの位数を q とするとき属性は $\mathbb{F}_q$ 上の元として表現される．述語として, 属性変数と値の等号, 等号否定およびそれらの論理和, 論理積, 否定, 閾値ゲート, および n 個の属性を n ビット整数と見なした等号, 等号否定, 不等号などの演算を利用する事が出来る．CCA 変換には Boneh-Katz 変換 [3] を用いた．それから, 属性や述語の大きさに対する Setup, KeyGen, Enc, Dec の処理時間のグラフを図 4〜図 7 に示す．測定環境等は表 1 に記載する．暗号化や復号では述語式によって処理時間が変動しうるので, 最悪の場合 (原子述語を AND で結合した述語) で時間計測を行なった．全体に属性や述語の大きさに対し, 処理時間が線形となる傾向が見て取れるが, 暗号化 (図 6) では線形スパンプログラムの素朴な処理による高次の項の影響が見て取れる．100 程度の属性や述語の大きさの範囲では KeyGen, Enc, Dec のいずれのアルゴリズムも 1.5 秒以内には完了している．Setup は 1 度しか使用されないので, 他のアルゴリズムより計算コストを多めに取っても構わないが, それでも 5 秒程度で完了しているので, 岡本らの関数型暗号 [7] は PC 上の実装であれば十分実用になり得ると考えられる.

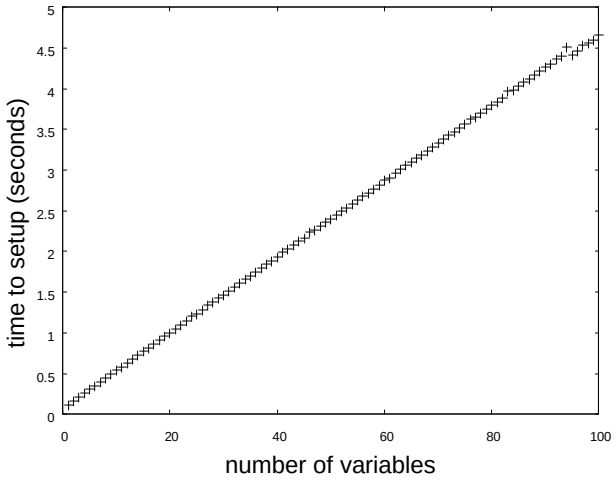

図 4: 変数の数に対する Setup 時間

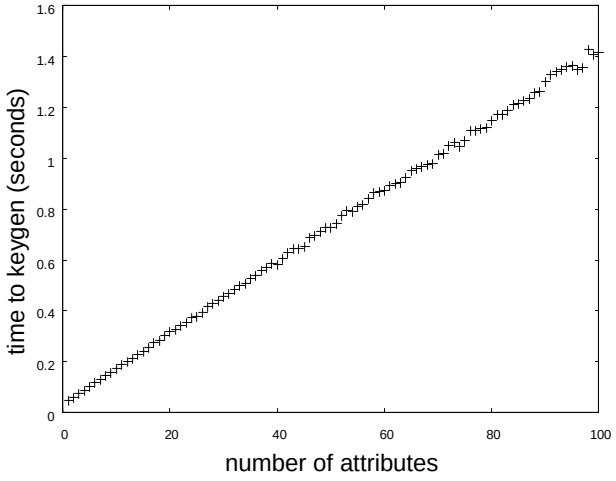

図 5: 属性の数に対する KeyGen 時間

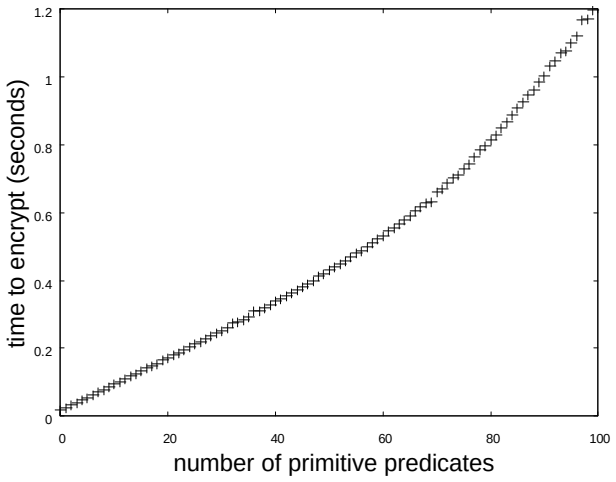

図 6: 述語サイズに対する Enc 時間

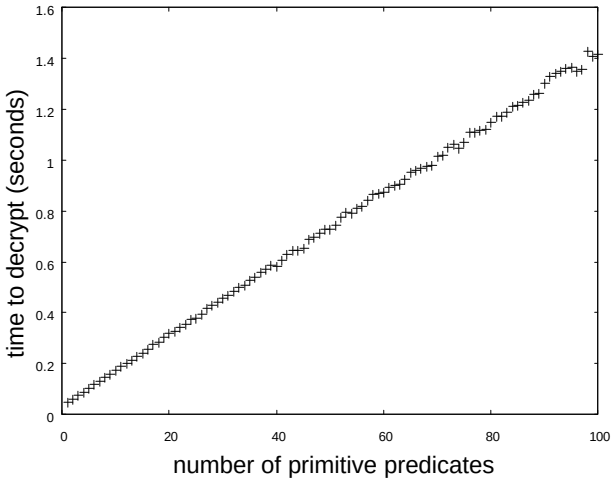

図 7: 述語サイズに対する Dec 時間

表 1: 実測環境・使用パラメータなど

CPU: Intel(R) Core(TM) i7-2600 CPU
測定時クロック数: 3.4GHz
コア (スレッド) 数/内使用数: 4(8)/1(1)
OS: Linux 2.6.32-71.29.1.el6.x86_64 (CentOS r6.0)
コンパイラ: gcc version 4.4.4 20100726
ペアリングパラメータ: 254 bit 素体 BN 曲線 [2]

5　応用

関数型暗号には様々な応用が考えられる．直感的な応用としては，暗号化によるアクセス制御と暗号化された情報の検索が考えられる．本節では NTT における応用の検討状況を報告するが，後者は同じく我々のグループからの報告 [11] [12] に譲ることとし，ここでは前者のアクセス制御について述べる．関数型暗号で暗号文に復号の条件式を指定し，閲覧者の属性を定めた復号鍵を配布することでアクセス制御システムを構成することができる．このシステムは従来のデータの保護方法（例えば，パスワード共有による閲覧制限や，PGP による暗号化メール）ではできなかったアクセス制御，例えば人の所属や時間や場所でファイルの利用を細かく制御すること，が可能になることが期待される．

5.1　情報のアクセス制御システムへの応用

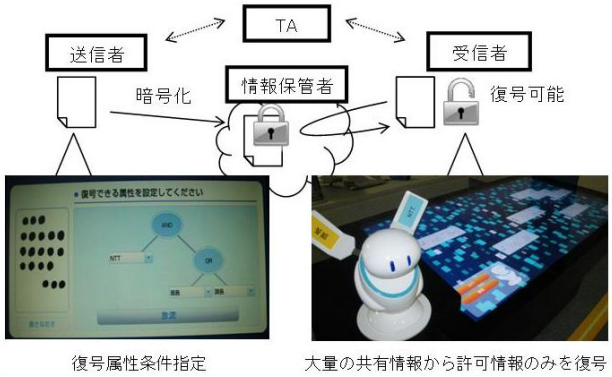

図 8: 情報アクセス制御システムのコンセプトデモ

関数型暗号による情報の開示制御を実現する情報共有システムを検討し，以下の構成要素からなるシステムを開発を行っている．

1. 送信者：情報の開示制御条件を指定し暗号化
2. 受信者：受け取った暗号化情報を属性鍵により復号
3. TA：受信者の認証に基づき属性鍵を発行し受信者へ発行
4. 情報保管者：暗号化された情報を受信・保管し，要求に応じて暗号化された情報を送信

図 8 は，関数型暗号による情報アクセス制御システムのコンセプトデモの概略である．関数型暗号のコンセプトの理解の促進を目的とした本コンセプトデモは，情報の復号属性条件を指定できるユーザインタフェイスを有する送信者，暗号化されたデータを保管するパブリックな情報空間（クラウド）を模した CG による情報保管者，および属性鍵（USB キー）と属性鍵を有する主体（USB ハブである 3 次元ロボット）から構成されている．本デモは NTT R&D フォーラム 2011 (2/22-23)，IEEE ICC 2011 (6/5-9) などで展示した．

5.2　情報アクセス制御システムの概要

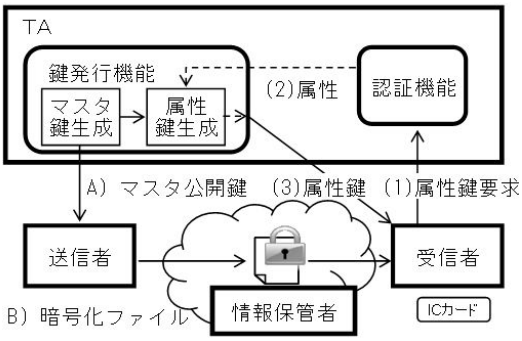

図 9: 情報アクセス制御システムの構成要素と動作概要

前記コンセプトデモシステムに続いて，情報アクセス制御システムのプロトタイプを開発した．動作概要は以下の通りである（図 9）．企業内で機密情報を所属や役職で管理する利用シーン（2.1 節の図 1）が構築可能になる．

動作概要

1. 送信者 (暗号化)
   - A) 鍵発行局のマスタ公開鍵 (暗号パラメータ，属性リスト等）を取得
   - B) 条件を決め暗号化
2. 受信者 (属性鍵取得)
   - (1) 属性鍵を TA に要求，この際認証を実施 (IC カードの ID をあわせて送信)
   - (2) 受信者の属性が鍵発行機能に通知
   - (3) 鍵発行機能は当該属性の属性鍵を生成し，受信者に発行 (この際鍵を IC カードの ID とくくりつける)
3. 受信者 (復号)
   - 暗号化ファイルを受信者の属性鍵で復号
   - 受信者の属性鍵所有の正当性を検証（復号端末の IC カードリーダに IC カードをかざし，鍵と受信者の IC カードの組み合わせを検証）

構成

- サーバ：Linux PC
- クライアント:Windows7(64bit)PC および IC カード

5.3 考察

応用システム構築を通じてわかる課題は，一般的な性能面のものを除くと，TA (Trusted Authority) の設計に関するものと属性鍵の管理に関するものが重要と考えられる．

前者の TA は，鍵発行や属性の管理を行う機関であり，公開鍵暗号系における認証局 (CA) が近い存在になる．応用システムを企業内で閉じて運用する場合は，TA を一括して（例えば社員情報システムと連動させて）企業自身が運用する形態が考えられる．しかし一般には，関数型暗号で属性を扱うとき，その暗号システム的表現の豊かさゆえに，TA の設計も様々な実装形態がありえるので，効率や利便性（わかりやすさ）などの観点から，議論と分析が必要である．

後者の鍵管理は，属性の鍵が不正に用いられない手立てが必要である．今回のプロトタイプでは，鍵を IC カードとくくりつけ，復号時に IC カード所有を確認することで属性鍵の正当性を確認している．鍵の管理は，応用先の要件に応じて様々な解決方法があると考えられる．私たちも鍵を限定して発行する方法（鍵の隔離）[14] や，鍵をクラウドで管理する方法 [13] を提案している．

6 普及への課題

本論文で議論した関数型暗号をオープンなクラウド上で利用するためには，標準化された社会的インフラが必要となる．この節では，関数型暗号実用化のための要素について述べる．

従来の暗号である公開鍵暗号や共通鍵暗号の場合は，

- アルゴリズムやパラメータに関する標準 (CRYPTREC・FIPS など)
- 通信プロトコル仕様 (SSL など)
- 公開鍵認証基盤 (CA など)
- 演算ライブラリ (OpenSSL など)

の 4 点が既に整備されており，多くの者が暗号の実装を行ったり利用することができる．これに対して関数型暗号や ID ベース暗号はこれらの要素が整備されておらず，普及のための障害となっている．

通信プロトコルや TA(Trusted Authority) に関しては，多くの研究者の見識のもとで合意できる仕様を定めていく必要がある．現在，ペアリングフォーラム [10] で，ID ベース暗号に関する通信プロトコルの検討および，鍵配送実験を行っている．

普及のための今後の目標は，上記 4 つの要素がすべて公開鍵暗号と同じ水準になり，誰でも実装できるようになることである．

アルゴリズムや楕円曲線パラメータに関しては，ISO や IEEE で ID ベース暗号に関する標準化が開始されている．関数型暗号における楕円曲線パラメータは，ID ベース暗号と同じものを用いることができる．現時点では関数型暗号が含まれる「属性ベース暗号」や「述語暗号」に関する標準化は行われていないが，今後標準化が開始されると考えられる．

通信プロトコルに関しては，ペアリングフォーラムで設計した仕様の標準化を行っていく予定である．

TA および鍵管理に関しては，属性情報を確認する方法 (RA との連携) と，発行した鍵が危殆化した際に無効化するしくみが必要である．暗号の適用先によっては，一度発行した鍵を無効化する必要がない場合などもありうるが，この 2 点に関してはいまだ一般的な方法が確立されておらず，今後研究を行なっていく必要がある．

演算ライブラリについては，ペアリング演算などについてはフリーソフトが既に存在しているが，今後暗号アルゴリズムや周辺の機能を含めたライブラリが開発されることを期待する．

7 おわりに

関数型暗号の概念，定義，モデルについて説明し，さらに NTT で行っている実装の現状について報告し，またどのような応用が可能か具体的に考え，加えて普及に向けて解決すべき課題について考察した．

現在，関数型暗号ライブラリとしては，PC 上で 100 程度の属性や述語の大きさの範囲においても暗号や復号が 1.5 秒以内に処理をすることができる．応用に関しては，企業内機密情報管理を中心に考察したが，さらにオープンな環境で使用するためには，標準化された社会インフラが必要である．

本稿により関数型暗号の理論や実装に興味を持つ方が一人でも増えることを期待する．また，社会インフラとして仕上げていくための議論が，さらに活発になることを期待するものである．

参考文献

[1] J. Bethencourt, A. Sahai, and B. Waters. Ciphertext-policy attribute-based encryption. In *Proceedings of the 2007 IEEE Symposium on Security and Privacy*, SP '07, pages 321–334, Washington, DC, USA, 2007. IEEE Computer Society.

[2] J.-L. Beuchat, J. E. González-Díaz, S. Mitsunari, E. Okamoto, F. Rodríguez-Henríquez, and T. Teruya. High-speed software implementation of the optimal ate pairing over barreto-naehrig curves. In M. Joye, A. Miyaji, and A. Otsuka, editors, *Pairing*, volume 6487 of *Lecture Notes in Computer Science*, pages 21–39. Springer, 2010.

[3] D. Boneh and J. Katz. Improved efficiency for cca-secure cryptosystems built using identity-based en-

cryption. In A. Menezes, editor, *CT-RSA*, volume 3376 of *Lecture Notes in Computer Science*, pages 87–103. Springer, 2005.

[4] D. Boneh, A. Sahai, and B. Waters. Functional encryption: Definitions and challenges. Cryptology ePrint Archive, Report 2010/543, 2010. `http://eprint.iacr.org/`.

[5] V. Goyal, O. Pandey, A. Sahai, and B. Waters. Attribute-based encryption for fine-grained access control of encrypted data. In A. Juels, R. N. Wright, and S. D. C. di Vimercati, editors, *ACM Conference on Computer and Communications Security*, pages 89–98. ACM, 2006.

[6] A. B. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In H. Gilbert, editor, *EUROCRYPT*, volume 6110 of *Lecture Notes in Computer Science*, pages 62–91. Springer, 2010.

[7] T. Okamoto and K. Takashima. Fully secure functional encryption with general relations from the decisional linear assumption. In T. Rabin, editor, *CRYPTO*, volume 6223 of *Lecture Notes in Computer Science*, pages 191–208. Springer, 2010.

[8] A. O'Neill. Definitional issues in functional encryption. Cryptology ePrint Archive, Report 2010/556, 2010. `http://eprint.iacr.org/`.

[9] A. Sahai and B. Waters. Fuzzy identity-based encryption. In R. Cramer, editor, *EUROCRYPT*, volume 3494 of *Lecture Notes in Computer Science*, pages 457–473. Springer, 2005.

[10] ペアリングフォーラム. `http://www.pairing-forum.jp`.

[11] 吉田麗生, 永井彰, 小林鉄太郎, 冨士仁. 内積述語検索可能暗号のためのバッチ検索アルゴリズム. In *SCIS*, 2011.

[12] 吉田麗生, 小林鉄太郎. ユーザ間で通信不要な暗号学的クラウドストレージ. In *SCIS*, 2012.

[13] 山本剛, 小林鉄太郎, 山本具英, 冨士仁, 高橋克巳. 暗号技術はクラウドで情報を保護できるか. In *SCIS*, 2012.

[14] 星野文学, 藤岡淳. 関数型暗号の応用:鍵隔離暗号の構成. In *SCIS*, 2012.

[15] 日本電信電話株式会社, 三菱電機株式会社. クラウド時代の高度なセキュリティー対策を実現する新世代暗号方式を開発〜最も先進的なインテリジェント暗号を世界で初めて実現〜, 2010. `http://www.ntt.co.jp/news2010/1007/100728a.html`.