関数型暗号を適用したスマートデバイス機能制御機構の実装

青柳 真紀子† 知加良 盛† 伊坂 広明† 栢口 茂†

スマートフォンやタブレットなどの普及により，スマートデバイスが様々な環境下で利用されるケースが増えている．そのため，スマートデバイスが利用される環境に応じて機能を制御するセキュリティ対策が有効であると考えられる．さらに，環境の変化に応じてそれらの機能が自律的に制御されることが重要である．
我々は上記のような要件を満たす機能制御機構を提案し，実装を通してその評価を行ってきた．しかしながらこれまでの方式では特定の環境属性を利用しており，多様な環境での利用が想定される場合には利便性・柔軟性に欠けるという課題があった．本稿では，その課題を解決するため関数型暗号を応用した方式を提案し，実装による検証と評価を行う．本方式は，従来方式に比べてより柔軟で多様な利用シーンに適用可能であることを示す．

An Implementation of Application/Content Management for Smartdevice with Functional Encryption

Makiko AOYAGI† Sakae CHIKARA† Hiroaki ISAKA† Shigeru KAYAGUCHI†

1. はじめに

近年，スマートフォンやタブレットなどの普及により[1]，スマートデバイスが様々な環境下で利用されるケースが増えている．また，個人所有のスマートデバイスを業務利用する BYOD（Bring Your Own Device）が進むなど[2]，ビジネスシーンも含めて利用形態も多様化している．

このように個人や企業でのスマートデバイスの利用が進むなかで，スマートデバイスを対象としたマルウェアの増加や端末の紛失/盗難など，その利用時のリスクやセキュリティ被害も多く指摘されており，セキュリティ対策の検討が進められている[3][4]．特に，上述した BYOD のような利用形態においては，個人固有のスマートデバイスが企業において利用されるが，その端末におけるセキュリティ担保は個人の意識に依存するところが大きいため，BYOD において企業の機密情報の漏えい防止などに向け，スマートフォンの適切な管理・制御が検討されている[4].

我々はこれまで，スマートフォン利用者の意識レベルや習熟度が多様であることを考慮し，ユーザの操作を介することなく様々な機能が自律的に制御できることが重要と考え，スマートデバイスが利用される場所に応じて，コンテンツの閲覧やアプリケーションの起動といった機能を自律的に制御する機能制御機構を提案してきた[5][6]. この方式では，許可された環境以外でアプリケーションの制御情報やコンテンツを保護するために暗号技術を用いており，前回の提案では ID ベース暗号[7]と呼ばれる暗号方式を応用したシステムを実装・評価した．しかしながら ID ベース暗号は公開されている ID 情報を公開鍵として利用するため，一属性しか扱うことができず条件も値の一致であり，多様な環境での利用が想定されるスマートデバイスに応用するには利便性･柔軟性に欠けるという点が課題であった．

本稿では，前述の課題を解決するために ID ベース暗号に代わり関数型暗号を応用した方式を提案する．関数型暗号とは公開鍵暗号をより高度にした暗号方式であり，暗号文と復号鍵に様々なパラメータ（属性と条件式）を用いることができるため，ID ベース暗号を用いた場合に比べて制御の柔軟性の向上が期待できる．関数型暗号方式は近年新しく開発された暗号技術であるため，本稿ではまず提案方式を実機を用いて実装し，実現性を検証する．

本稿では，2 章では上記プラットフォームに求められる要件を整理し実現方式について述べる．3 章では本提案方式をスマートデバイス上のアプリケーションやサーバアプリケーションとして実装した結果について述べ，4 章において実装した機能制御機構における処理性能と実行性について確認し，5 章においてまとめと今後の課題を述べる．

2. 要件整理と提案方式

2.1 要件整理

Android や iOS，Windows Mobile などのスマートフォンのプラットフォーム OS は，その端末のセキュリティを担保するための機能を備えている．たとえば，Android OS のバージョン 2.2 以降では，Device Administration API が用意されており，パスワードポリシーの強化/強制や端末の遠隔ロックやデータ消去，内部メモリの暗号化などのセキュリティ機能が提供されている[9]. iOS や Windows Mobile でも

† NTT セキュアプラットフォーム研究所
NTT Secure Platform Laboratories

同様の機能が提供されており[10][11]，スマートフォン利用時における各種リスクに対するセキュリティ対策を実現する基本的な機能は OS において用意されている．

これら機能を利用し，各セキュリティベンダや通信キャリアなどは MDM（Mobile Device Management）と呼ばれるソリューションを，主に業務利用向けに提供している．MDM においては，上述の OS が用意するセキュリティ機能を用いた制御アプリケーションを利用者のスマートフォンにインストールする．そして，管理者端末からの制御命令や制御設定ファイルを制御アプリケーションが受信し，それに従った機能制御を実行する．これにより，たとえば，盗難された利用者のスマートフォンを管理者端末から遠隔ロックすることで，その機能を実質的に無効化することができる．現在，業務用のスマートフォンに対しては MDM を利用したセキュリティ対策が一般的である．

一方で，以前の我々の研究においては，スマートフォンの特徴である可搬性や多機能性に加え，その利用者のセキュリティ意識の低さに着目し，多様な利用環境におけるスマートフォンの機能制御機構を設計する要件として以下の 3 点を挙げた[5][6]．

要件 1：動的に変化するスマートフォンの利用環境に応じ，柔軟にその機能制御ができること．

要件 2：高度な通信環境がなくてもスマートデバイスの機能制御ができること．

要件 3：スマートデバイスの機能制御の内容とその実施権限はスマートデバイス利用者以外が保有すること．

MDM を利用した機能制御の場合，管理者端末からの制御命令や制御設定ファイルが送信され，画面転送型の端末制御手法[12]と比較して高速な通信環境を必要としない（要件 2 を充足）．また，機能の制御は管理者端末を操作する管理者の命令により実行される（要件 3 を充足）．しかし，一般的に管理者の操作が必要な MDM は，スマートデバイスの初期設定時や紛失/盗難時での利用が想定されており，スマートデバイスの現在の利用環境に応じた頻繁な制御は想定されていない．前回の提案方式では，ID ベース暗号を応用し位置情報に基づいた制御を行うことで要件 1 を達成した．ただし利用環境として扱う情報が位置情報のみという制限があるため，「多様な利用環境」に対応することで要件 1 をより高度に実現するシステムを検討する．

2.2 実現方式

これまでの報告では特に要件 1，3 を両立するシステム構成を実現するために， ID ベース暗号を用いたフレームワークを導入してきた[13]．ID ベース暗号は，受信者の email アドレスや電話番号，社員番号などの基地の ID を公開鍵として用い，受信者への暗号文を作成する公開鍵暗号系の暗号方式である[8]．しかしながら，ID ベース暗号を用いた方式では公開されたユニークな ID で暗号化するため，一属性しか扱うことができず条件も値の一致のみであるという点で柔軟性に欠けることが課題であった．本稿では特に要件 1 にあたる多様性，柔軟性を強化することを重視し，ID ベース暗号に代わり関数型暗号[16]を導入した方式を提案する．関数型暗号の概要と利用形態については後述するが，暗号文とそれを復号する鍵に様々なパラメータ（属性情報と条件式）を組み込むことができる．この属性情報とは例えば位置情報や時間情報，組織名や役職などで，それら属性情報の AND・OR などを条件式として記述できる．また，RSA などに代表される一般的な公開鍵暗号方式では暗号化を実施する前に公開鍵と秘密鍵の鍵ペアを生成する必要があるが，関数型暗号は対応する秘密鍵を事後に生成できる特徴を持つ．つまり，暗号文を取得した受信者は，暗号化に利用された属性情報を保持していることを鍵管理者へ示し，それを復号するための鍵を取得する．この特徴によりこれまでの報告[5][6]で導入した ID ベース暗号方式と同様，認証と認可のタイミングを独立としたフレームワーク（図 1）が実現できる．

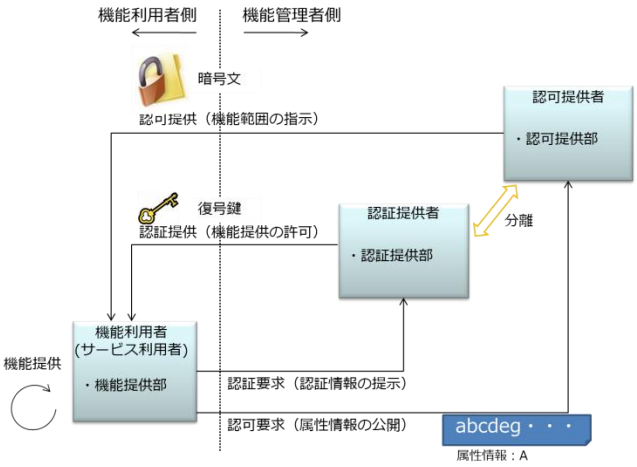


図 1 認証認可独立のフレームワーク

スマートデバイスの利用環境とそれに応じた機能制御は機能利用者の属性とそれに応じた機能の認可に相当すると考えることができる．図 1 において，機能利用者の持つ属性情報を ID とし、機能提供範囲にお制御指示をその ID で暗号化すると，その暗号文は認可情報に相当すると考えられる．この特徴を利用することにより，予めある属性情報を持つユーザに対して認可を与えておき，その属性情報が認証された時点で認可を実行することができる．これにより，認可提供者は認可を発行するたびに認証提供者と連携する必要がなく，システム全体の運用性が向上し，動的に変化する利用環境を認可情報として管理する必要があっても，システムに与える負荷を軽減することが可能である．

スマートデバイスを制御したい管理者は，その利用環境

を表す環境情報と条件式を用いて機能制御情報を暗号化する．利用環境ごとの暗号文をあらかじめ用意することで，動的に変化する利用者の属性情報の管理にかかる運用性を確保した柔軟な機能制御が両立できる．さらに，機能制御情報は暗号化されており，利用者による書き替えのリスクも低減できる．条件式を柔軟に記述できる関数型暗号を適用することにより，複数の条件を復号条件として暗号化しておくことが可能となるため，属性情報の数ごとに暗号化を行う必要がないなど，従来方式と比べて暗号化ファイルのライフサイクル管理などの運用コスト減も期待できる．

2.3 関数型暗号の概要と利用形態

ここで，関数型暗号の概要とその利用形態について簡単に触れる．関数型暗号とは公開鍵暗号をより高度にした暗号方式であり，暗号文と復号鍵に様々なパラメータ（属性情報と条件式）を導入することで「暗号-復号」のロジックを規定することができる．具体的な利用形態として 2 通りあり，「復号鍵に属性情報，暗号文に条件式」あるいは「暗号文に属性情報，復号鍵に条件式」を組み込むことができる．前者を Ciphertext Policy 方式，後者を Key Policy 方式という．本提案で要件とするのは，特定の条件を満たした場合に利用を許可する，ケースのため，前者の方式を採用して実装を行う．

3. 実装

前節で述べた認証認可独立のフレームワークでは，管理者側の運用性の向上が期待されるものの，スマートデバイス上での暗号文の復号処理が必要となる．さらに，今回導入する関数型暗号方式は近年新しく開発された暗号技術であるため，実運用における実績に乏しく，まずは実機での実現性の検証を目的とした．従って，システム構成についてはこれまでの提案方式と大きくは変えずに実現性の評価を行うものとする．そのシステム構成を図 2 に示す．

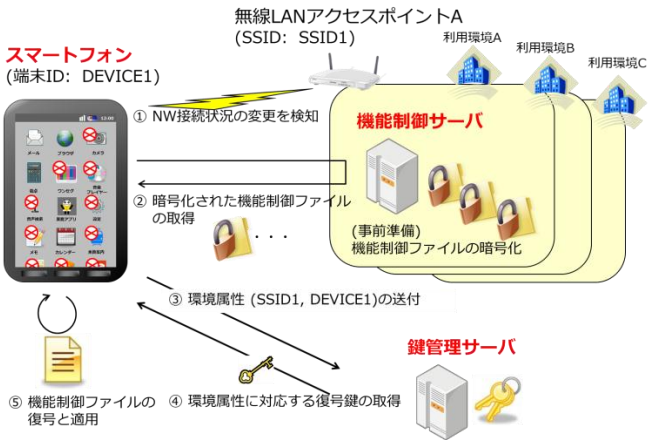


図 2 システム構成図

アプリケーションの起動制御を実現するために，サーバ側には機能制御ファイルの管理機能と復号鍵の生成機能，クライアント側にはアプリ制御機能と暗号化された機能制御ファイルの復号機能が必要である．また，コンテンツの閲覧制御を実現するために，サーバ側にはコンテンツファイルの管理機能と復号鍵の生成機能，クライアント側にはコンテンツデータのビューワ機能と暗号化されたコンテンツファイルの復号機能が必要である．さらに，これらの制御を利用環境に応じて実現するために，クライアント側に利用環境の検知機能が必要である．

スマートデバイスの利用環境としては前回同様に位置を表す情報として無線 LAN アクセスポイントの Service Set Identifier（以下，SSID），その他日時や時間帯などで制御可能とした．図 2 においては，無線 LAN アクセスポイントの SSID の値である「SSID1」が届く範囲において起動を禁止，もしくは起動を許可するアプリケーションを記載した機能制御ファイルを「SSID1」とスマートデバイスの端末に固有の ID（以降，端末 ID）である「DEVICE1」の AND 条件で暗号化を行っている．

機能制御ファイルは上記のような環境属性と条件で予め暗号化したうえで機能制御サーバに格納しておき，端末にインストールされた機能制御アプリは当該端末で利用可能な機能制御ファイルリストを取得しておく．機能制御ファイルは様々な環境属性で暗号化してあるため，本システムではそれらの環境情報を定期的に監視する機能を実装した．さらに，当該端末の環境情報に基づいて暗号化ファイルの復号可否を判定する機能を実装し，環境情報の変化に応じて適用可能な機能制御ファイルをリアルタイムに更新する機能を実装することで，端末の利用環境に適したポリシーが利用できるよう配慮した．「SSID1」の配下に入ったスマートデバイスは，端末 ID である「DEVICE1」と取得した SSID が暗号化ファイルの復号条件を満たすことから当該ファイルを選択可能にする．ユーザが当該機能制御ファイルを選択すると，端末 ID と SSID の組を鍵管理サーバに送信し，対応した復号鍵を取得する．この復号鍵を用いて暗号化された機能制御ファイルを復号し，スマートデバイスに適用する．これにより，機能制御ファイルに記載された制御情報に応じてアプリケーション制御が実現できる．

前述したように，アプリケーション制御とコンテンツ制御を実現するためには，サーバ側の機能として機能制御ファイルの管理機能・コンテンツファイルの管理機能・復号鍵生成機能，クライアント側にはアプリ制御機能・コンテンツのビューワ機能・復号機能・環境情報検知機能が必要となる．実装にあたり，システムの汎用性を高めるために各機能が独立のサービスとして動作可能であることを設計方針とし，クライアントアプリの改造を容易にするためモジュール化を実施した．従って，サーバ機能として機能制

御サーバ，コンテンツ管理サーバ，鍵管理サーバの 3 サーバ，クライアントアプリとして機能制御アプリ，コンテンツビューワアプリ，復号制御アプリ（復号機能，環境情報検知機能）の 3 アプリを実装した．サーバ側の実装環境としては OS として RedHat Enterprise Linux（RHEL） 6.3 とその上に Apache 2.2 系，Tomcat 6 系，OpenSSL 1.0.0 系などを利用した．端末側は Android OS 4.1 以上，iOS7 系を対象とした．クライアントアプリの機能構成を図 3 に示す．

図 3　機能制御クライアントアプリの機能構成

iOS については OS の制限があり，機能制御アプリおよび復号制御アプリを単独サービスとして動作させることが困難であるため，コンテンツビューワアプリに復号機能と環境情報検知機能を組み込んだ一体型のアプリとして実装し，ビューワアプリでの暗号化データの復号制御のみを実現した．

3.1　機能制御サーバ

スマートデバイスの機能を制御する管理者が本サーバを利用する．機能制御ファイルは，指定した状況で特定のアプリケーションを起動可能とするホワイトリスト形式，もしくは特定のアプリケーションを起動停止するブラックリスト形式で記述する．サンプルを図 4 に示す．SSID や端末 ID などの環境条件は環境条件ファイルとして機能制御ファイルの暗号化時にオプションとして添付する．暗号化した機能制御ファイルは機能制御サーバ上で公開する．

また機能制御ファイルは予め利用する環境を想定し，その条件で暗号化を実施しておく．暗号条件のサンプルを図 5 に示す．

```
<policy version="3" >
  <default level="+1" />
  <apps>
    <app
      action="jp.example.launch.action.ACTION"
      category="android.intent.category.DEFAULT"
      class="jp.example.launch.LaunchClass"
      level="+2"
      package="jp.example.launch" />
    <app
      class="jp.example.package.ExampleClass"
      level="-2"
      package="jp.example.package" />
```

図 4 機能制御ファイルの一例

```
<cond>
<and>
 <and>
  <envinfo id="usr_id">userid</envinfo>
  <envinfo id="usr_device">deviceid</envinfo>
 </and>
 <and>
  <or>
   <envinfo id="osi_ssid">SSID01</envinfo>
   <envinfo id="osi_ssid">SSID02>></envinfo>
  </or>
 </and>
</and>
</cond>
```

図 5 暗号化条件の一例

3.2　コンテンツ管理サーバ

機能制御サーバは，アプリケーションの起動可否を記載した機能制御ファイルを暗号化することによって，アプリケーションを制御した．一方，コンテンツ管理サーバにおいては，コンテンツの管理者が閲覧制御をしたいコンテンツに対して直接暗号化を実施する．コンテンツ管理サーバを中心とした機能制御機構は，図 2 において機能制御サーバをコンテンツ管理サーバに，機能制御ファイルをコンテンツに置き換えたものとなる．この場合，アプリケーションに対する制御とは異なり，コンテンツの暗号化された状態がそのまま制御対象を保護している状態になる．

3.3　鍵管理サーバ

スマートデバイスからの鍵生成要求に対し，関数型暗号の復号鍵の生成を行う．

第 2.2 節で述べたとおり，機能制御機構においては認可情報に相当する暗号化ファイルは事前に端末側に取得されており，それを復号するための復号鍵を発行する際に認証を実施する．しかし，認証方式は単純な ID/PW 方式や OpenID[14]や SAML[15]といったシングルサインオンを実現する認証連携方式など，サービスの提供形態により様々な形態が考えられる．従って，本実装においては特定の認証方式を採用せず，認証機能を組み込みやすいモジュール構成のみを考慮した．

3.4　機能制御アプリ

アプリケーションの起動可否の制御を実現するため，常駐型のホームアプリとして実装した．機能制御アプリがもつ機能は主に，機能制御ファイルの取得・管理機能，アプリ制御状態の UI 表示機能，アプリ制御・状態通知機能，の 3 機能である．機能制御アプリを起動すると，機能制御サーバから当該端末で利用可能な機能制御ファイルのリストを取得する．すでに取得済みのリストが存在する場合には差分チェックを行い，利用可能な機能制御ファイルを取得する．機能制御アプリは復号制御アプリと連携すること

で，当該端末の利用環境において各機能制御ファイルが復号可能か否かという情報を取得し，機能制御ファイル一覧として表示する．復号可能と判定された機能制御ファイルをユーザが選択すると，再度復号制御アプリと連携して当該機能制御ファイルを復号し，復号されたファイルを読み込んでその記述に応じて端末内アプリケーションを制御した状態で端末画面に表示する．

3.5 コンテンツビューワアプリ

コンテンツの閲覧制御を実施するアプリケーションとして実装した．主な機能は，コンテンツの取得・管理機能とコンテンツ表示機能の 2 機能である．ビューワアプリを起動すると，予めコンテンツ管理サーバから取得した暗号化コンテンツリストを表示する．ユーザがリストからコンテンツを選択すると，復号制御アプリと連携してコンテンツを復号し，復号されたコンテンツを表示する．

前段は Android OS 版アプリの機能構成であるが, 前述したとおり iOS 版ビューワアプリとしては，暗号化データ復号機能と環境情報管理機能を内包した一体型アプリとして実装した. これら機能は Android OS 版アプリでは復号制御アプリの機能として実装し，概要は次節に示す．

3.6 復号制御アプリ

端末の利用環境に応じて関数型暗号で暗号化されたデータを復号するアプリケーションである．主な機能として，暗号化データ復号機能，暗号化データ復号可否判定機能，環境情報管理機能，の 3 機能である．

環境情報管理機能は，先の例に示した SSID などの利用環境にあたる情報を監視し，最新状態に保つ．環境情報としては SSID のように OS から取得できる情報に加え, 外部アプリや外部サーバと連携して情報を取得することも可能である．環境情報管理機能は，機能制御アプリが機能制御ファイルリストを取得し暗号化データの復号条件にあたる環境情報の監視を依頼されることで当該環境情報の監視を開始する．

暗号化データ復号可否判定機能は，暗号化データの復号を行う前に，事前に取得した暗号化ファイルの復号条件と環境情報管理機能が保持する現在の利用環境を照合し，復号可能か否かを判定する機能である．機能制御アプリが新たな機能制御ファイルを取得した場合や，環境情報管理機能が監視している環境情報を更新した場合に当該機能制御ファイルの復号可否判定を行う．

暗号化データ復号機能は，暗号化されたデータを関数型暗号を用いて復号する機能である．本機能は機能制御アプリ，コンテンツビューワアプリから利用される．各アプリのサービス画面からユーザ操作により暗号化ファイルが選択されることで当該暗号化ファイルの復号を行う．当該暗号化ファイルの復号条件にあたる環境情報を環境情報管理機能から取得し，鍵管理サーバに復号鍵を要求し，取得した秘密鍵で当該ファイルの復号を実施する．

4. 評価

前節で述べた，関数型暗号を応用した機能制御機構のシステムの実行性と優位性について，主に前回の提案方式である ID ベース暗号を応用した方式と比較して評価する．

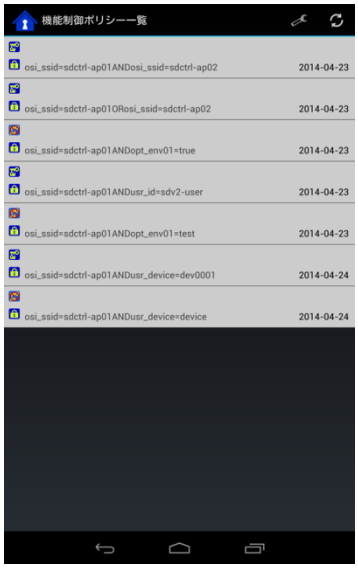
4.1 システムとしての実行性

まずは，機能制御機構として前回の提案方式で示した機能要件が，本方式でも実現できることを示す．前回示した要件に対して機能制御機構として実装した，以下の機能を確認する．

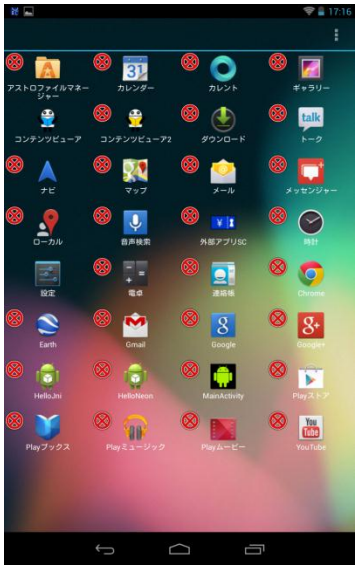
- スマートデバイスの利用環境に応じてアプリケーションの起動可否が制御できること（図 6）
- スマートデバイスの利用環境に応じてコンテンツの閲覧制御ができること（図 7）
- 機能制御ファイルやコンテンツファイルを暗号化して保護し，利用可能な状況でのみ端末側で復号し利用できること

これらを実現するアプリケーションをインストールした端末で起動させ，機能を実現することが確認できた．図 6，7 に実機検証した際の画面キャプチャを示す．図 6 は，取得した機能制御ファイル一覧と各ファイルの復号可否判定結果(1)と，選択した機能制御ファイルを適用し，アプリの起動制御を行った結果画面(2)を示す．図 7 は，暗号化コンテンツファイルを復号した結果画面を示す. 復号結果として，復号対象のデータサイズと復号処理にかかった所要時間，ファイルの復号条件を示している．スマートデバイス端末で複雑な復号処理を行う点は処理時間が懸念されるものの，1MB 相当のファイルでも 2 秒程度で復号できており，体感でもサービス運用として問題がないレベルといえる．

検証端末は Google Nexus7 を用いた. なお, 前回は Android OS のみを対象としたが，今回ビューワアプリにおけるコンテンツの閲覧制御機能に関しては iOS による実装も行い，動作を確認している．



(1) 機能制御ファイル一覧と復号可否判定結果



(2) アプリの起動制御

図 5 機能制御機能の実行結果

図 7 暗号化コンテンツファイルの復号処理

4.2 多様な利用環境に対する柔軟性

次に，前回の提案方式に比べて制御条件の柔軟性などの面で本方式の優位性を示す．

本方式前回の提案である ID ベース暗号を応用した方式における限界の一つとして，公開されたユニークな ID で暗号化するため，一属性しか扱えない点がある．この場合，システムを設計する時点でどの種の公開 ID を使うかを決定する必要があり，前回の提案[6]では SSID の値を鍵として用いた．つまりこの方式では，ほかの環境属性を採用する場合には新たにシステムを設計しなおさなければならない．

今回の提案は ID ベース暗号を関数型暗号に置き換えた点であるが，環境属性自体を暗号化時点で暗号化ファイルに埋め込むことができるため，利用する属性を予め一つに絞る必要がない．実際に今回実装したシステムでは，環境属性として SSID のほかに BSSID（無線アクセスポイントの MAC アドレス），日時，時間帯，ユーザ ID，端末 ID，その他の外部属性情報，の 7 属性について動作検証を行った．関数型暗号を応用することにより，属性数を単純に比較しても 7 倍の環境条件に対応できるようになる．さらに，ID ベース暗号を応用した方式では環境情報を示す ID の一致のみを条件として用いていたが，関数型暗号では属性情報の条件式を組み込むことができる．今回のシステムでは条件式として AND 条件と OR 条件を指定可能であることを確認した．属性数だけでなくその条件式を含めた条件で暗号-復号可能となるため，ID ベース暗号を応用した方式と比べて飛躍的に柔軟な条件のもとに機能制御が実現できることを確認できた．

制御対象であるスマートデバイスは近年高機能化しているとはいえサーバ等に比べればその処理性能には限界がある．本システムでは，端末側での自律制御という要件を重視したため，暗号化ファイルの復号処理を端末側で実施する構成をとった．柔軟な制御情報を扱える点はメリットではあるが，複雑な条件設定を扱うことが端末の処理能力を圧迫することにもなりうる．従って，ある程度安定運用を前提としたシステム設計を実現する場合には、制御対象端末の処理性能を想定して暗号化条件のバリエーションにある程度の制限を設ける必要がある．本稿では関数型暗号を応用したシステムの実行性の検証に焦点を当てたが，本提案システムの処理性能や運用性に関する評価・検討は今後の課題である．

5. まとめ

本稿では，スマートフォンの利用環境に応じた動的な機能制御を実現するため，関数型暗号を応用した機能制御機構を提案し，その実装と実現性の確認を行った．

今回，関数型暗号を応用した方式を Android OS と iOS の端末，各種サーバによってシステムとして実装し，Android OS 4.1 以上，iOS7 系，Linux OS 6.3 系での動作を検証した，前回提案した機能制御機構の要件を満足し，アプリの起動制御・コンテンツの閲覧制御・端末での復号処理を実行するシステムを，関数型暗号方式を応用することでも実現することができた．スマートデバイス端末上での復号処理時間についてもデータサイズ 1MB 相当のファイルが数秒で復号できることを確認した．

機能制御サーバやコンテンツ管理サーバにおいて，NW 接続環境や日時，時間帯，などの様々な利用環境情報に基づいた暗号化を実現できることで，これまでの提案方式である ID ベース暗号を応用した方式よりも制御条件の柔軟性などの面で優位性があることを示した．またそれらの複合的な条件に基づいて制御を行えることからより様々な利用シーンに柔軟に適用可能である．

本稿では実現性の検証と定性評価を中心に行ったが，本提案システムの処理性能や運用性に関する評価・検討は今後の課題である．

参考文献

1) 総務省，平成 25 年版情報通信白書，http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h25/index.html 第 2 部 第 3 節 1-（1） 主な情報通信機器の普及状況（参照 2014-04-24）

2) ARUBA networks, “BYOD in Europe, Middle East and Africa: An overview of adoption, challenges and trends”, http://www.arubanetworks.com/wp-content/uploads/Aruba-Networks-Infographic-v6.jpg, (accessed 2014-04-24)

3) 独立行政法人情報処理推進機構セキュリティセンター，“IPA Technical Watch スマートフォンへの脅威と対策に関するレポート”，https://www.ipa.go.jp/files/000024773.pdf，（参照 2014-04-24）

4) 独立行政法人情報処理推進機構セキュリティセンター，“スマートフォンのセキュリティ<危機回避>対策のしおり”2012 年 6 月 8 日 第 2 版，https://www.ipa.go.jp/files/000011456.pdf，（参照 2014-04-24）

5) 佐藤亮太，知加良盛，奥田哲矢，栢口茂，スマートデバイスにおける利用環境に応じた機能制御機構の提案とその考察，情報処理学会論文誌 第 55 巻 第 1 号，2014 年 1 月発行

6) 佐藤亮太，知加良盛，奥田哲矢，栢口茂：スマートフォンにおける利用環境に応じた機能制御機構の実装と評価，電子情報通信学会技術研究報告，Vol.112, No.466, pp.203-208, Mar. 2013.

7) D. Boneh and M. Franklin, “Identity-based encryption from the Weil pairing, ” CRYPTO 2001, LNCS 2139, Springer Verlag, pp. 213-229, 2001.

8) ID ベース暗号調査 WG，“ID ベース暗号に関する調査報告書”，CRYPTREC，http://www.cryptrec.go.jp/report/c08_idb2008.pdf，（参照 2014-04-30）

9) Android developers, “Device Administration,” Google, http://developer.android.com/guide/topics/admin/device-admin.html, （参照 2014-04-30）

10) Apple, “iPhone と iPad の配備,” Apple, https://ssl.apple.com/jp/iphone/business/docs/iOS_6_Business_Sep12.pdf, （参照 2014-04-30）

11) TechNet, “Security Policies in MDM,” Microsoft, http://technet.microsoft.com/en-us/library/dd261828.aspx, （参照 2014-04-30）

12) Eric Y. Chen and Mitsutaka Itoh, “Virtual Smartphone over IP, ” In proceedings of IEEE WoWMoM 2010, Montreal, QC Canada, June, 2010.

13) D. Boneh and M. Franklin, “Identity-based encryption from the Weil pairing, ” CRYPTO 2001, LNCS 2139, Springer Verlag, pp. 213-229, 2001.

14) OpenID, OpenID Authentication 2.0 – Final: http://openid.net/specs/openid-authentication-2_0.html, (accessed 2014-05-07)

15) OASIS, Security Assertion Markup Language (SAML) v2.0: https://www.oasis-open.org/committees/download.php/35711/sstc-saml-core-errata-2.0-wd-06-diff.pdf, (accessed 2014-05-07)

16) T. Okamoto, K. Takashima, “Fully Secure Functional Encryption with General Relations from the Decisional Linear Assumption”, CRYPTO 2010, LNCS6223, pp.191-208, 2010.