様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

平成　２９　年　　６　月　１１　日現在

機関番号：　１３７０１
研究種目：　若手研究(B)
研究期間：　2010～2016
課題番号：　２２７０００６７

研究課題名（和文）情報漏えい耐性と簡便性の高い情報交換サービス提供基盤の開発

研究課題名（英文）Study on Data Exchange Service with Confidentiality and Usability

研究代表者
毛利　公美（Mohri, Masami）

岐阜大学・総合情報メディアセンター・准教授

研究者番号：５０２８１６９７

交付決定額（研究期間全体）：（直接経費）　　2,100,000円

研究成果の概要（和文）：本研究は高い情報漏えい耐性を持ちながら簡便な情報交換サービスを実現する基盤の開発を目的としている．具体的には，センターからの情報漏えいリスクが生じないような情報交換サービスのための暗号化通信方式を開発し，さらにそれをWebアプリケーションシステムとして提供するためのスクリプト系言語による暗号化・通信ライブラリの開発を行った．

研究成果の概要（英文）：One of important issues is to provide data sharing platform with confidentiality and usability for constructing trusted cloud services. This research proposes an identity-based encryption for cloud service providers of data exchange and an encryption library based on a script language for providing Web applications.

研究分野：　情報通信工学

キーワード：　セキュリティ　暗号

１．研究開始当初の背景

ペアリング暗号を用いた情報交換サービスのモデルでは，PKG（Private Key Generator）と呼ばれるセンターを設け，暗号文を復号しょうとするユーザからの要求に対して，公開パラメータからユーザの ID に対応する秘密鍵を生成・配送する，という仕組みになっている．しかしながら，このモデルは PKG がユーザの秘密鍵を生成・保有するため，センターからの情報漏えいに対する耐性がないことが指摘されている．この課題を改善するために，センターには秘密鍵の一部（部分鍵）のみを置き，センターが単独では復号できないようにするような方式がいくつか示されているが，暗号化・復号処理や鍵の管理・更新手続きが複雑になるなど，サービス提供のためのシステムを構築する側の立場からは，運用・管理コストが高いモデルになっている．また，従来の公開鍵認証基盤に基づく情報交換・蓄積サービスモデルでは，公開鍵の正当性を証明するための認証局を含めたモデルを考える必要があり，導入コストやサービスの提供範囲に関するスケーラビリティの観点から不特定多数に対する情報交換サービス提供基板として最適であるとは言い難い．

２．研究の目的

本研究で開発する情報交換サービス提供基盤を実現するための要素として，以下の二つの研究項目を設定する．

1. センターからの情報漏えいに対して耐性のある情報交換サービス提供方式
2. Web アプリケーションシステムの形での情報交換サービス提供基盤

本研究で開発する情報交換サービス提供方式は，センターに秘密情報を持たせることなく安全な鍵交換を行うことによって，情報漏えいリスクがなく，かつ公開鍵の正当性を保証するための認証局を必要としない簡便かつ柔軟な情報交換サービス提供基盤を目指す．

本研究で開発する暗号化・通信ライブラリ群は，バーチャルマシン上で動作するスクリプト系言語を用いて開発する．これは，情報交換サービスモデルを Web サービスとして提供することで，クライアントに別途ソフトウェアをインストールすることなく安定したサービスを受けることが可能となるためである．

３．研究の方法

本研究では，(1) センターにユーザの秘密情報を保有させることなく，ユーザの公開鍵の正当性を保証する仕組み，(2) ユーザの存在証明を保証する ID に基づく鍵交換方式によって，認証局を設けることなくユーザ公開鍵の正当性を保証する鍵交換方式の二点を満たす方式を検討する．そして，(3) バーチャルマシン上で動作するスクリプト系言語による暗号化・通信ライブラリ群を開発する．

(1)，(2) に対しては，ID ベース暗号を応用して設計をする．

(3) について，既に標数 2 および 3 の有限体 F_{2^m} or F_{3^m} 上のペアリングに関しては，その演算ライブラリが C++や Java 言語など，いくつかの言語によって提供されているが，ペアリングによる暗号化をクライアント PC で行うような Web アプリケーションを想定した場合に，クライアントの動作環境に影響されにくいスクリプト系言語での開発が望ましい．特に，Flash コンテンツのコントロールに使われる ActionScript による公開ライブラリは，現在のところ提供されていない．ActionScript は仮想マシンのバージョンが同じ，つまりインストールされている Adobe FlashPlayer のバージョンが同じならば，クライアントの環境に影響されず同じ動作をするアプリケーションを用意に開発することができる．

４．研究成果

情報交換サービスの例として，ファイル配送を考える．ファイルの要求者 A がファイルの保持者 B に対して送信要求を出す場合，この要求情報（要求元：A の ID，要求先：B の ID）のみを接続仲介サーバが保持し，その情報を参照した B は接続仲介サーバから要求元となる A の情報を入手し，直接 A に対してファイルを送信するというホスト間直接通信を想定する．

一般に，宛先として利用できる ID は，ユーザ ID とホスト ID が考えられるが，ホスト ID を用いて暗号化する場合は，どのホスト間で暗号通信が行われたかが暗号文から追跡可能になる．このことは，社外へのファイルの持ち出しなど，どのホストにファイルが保存されたのか，ファイルがどのように散らばったのかを管理者が把握できることを意味しており，機密情報管理という観点からユーザ ID を用いるよりも有用である．

しかし，その反面，接続仲介サーバがファイル要求元／要求先の両方の ID を保持するため，もし，攻撃者にこのリストを入手されるとファイルの所在が漏えいすることになる．これを避けるためには，ホスト ID をシステム起動ごとに変更し（ID 発行サーバの役割），接続仲介サーバから情報が漏えいした場合でも，ID 発行サーバと結託しない限りファイルの所在が漏えいしないようにすればよい．これにより，ホスト ID が漏えいしても攻撃者は実際に通信したホストを特定することができず，ファイルを保持しているホストへの直接的な攻撃を回避できる．一方，すべてのサーバ・ユーザ・ホストから情報を取得可能な権限を有する管理者は，ID 発行サーバ・ユーザ・ホストから情報を取得可能な権限を有する管理者は，ID 発行サーバの情報，接続仲介サーバの情報，ユーザの情報，暗号文を用いることで，ファイルの所在を追跡す

ることができる．そこで，(a)ホスト ID を用いた暗号化，(b)ホスト ID をシステム起動ごとに変更する，(c)ホスト情報とユーザ ID を個別にホスト ID と紐付ける，という特徴を持つファイル配送を設計する．

このとき，まず，暗号化に関する秘密情報を仲介サーバに保持させた場合，その秘密情報を使ってサーバ側でファイルの内容が復元できてしまい情報漏えいにつながる，ホストの ID が変更されるたびにサーバがすべての秘密情報を生成しなおす必要がある．したがって，暗号化に関する秘密情報をすべてホスト側に移し，ホストだけで秘密情報を生成・管理できることが要件 1 としてあげられる．次に，通信の宛先を表すホスト ID を用いて簡便に暗号化でき，かつ管理者によってファイルの配送先が追跡できる暗号化方式とする場合，ホストが自身のホスト ID に対応した公開鍵（ホスト ID とユーザの秘密情報を用いた鍵）を生成できること，およびホストが自身の秘密鍵を生成できることが要件 2 としてあげられる．最後に，システム起動ごとにホスト ID が変更されても暗号化する場合，ホスト ID が変更されるたびにホストが自身の公開鍵を生成できることが要件 3 としてあげられる．

提案するシステムのエンティティは以下のように定義する．

[ユーザ管理サーバ（UserManagement Server）] ユーザのアカウントの登録と登録されたユーザを認証してシステムへのアクセス許可証を発行する処理を行う．このときユーザアカウントは不正に改変されない．

[ID 発行サーバ（IdentityIssueServer）] 正しいユーザの使用するホストに ID を発行する．この ID はホストを一意に特定できるものであり，不正に改変されない．

[接続仲介サーバ（ConnectedMediation Server）] ホストを認証し，接続要求を受け付けて公開する．このとき接続用情報は不正に改変されない．

[長期鍵公開サーバ（LongTermKeyServer）] アカウント登録時にユーザの長期鍵を登録し，登録された長期鍵の公開を行う．このとき長期鍵は不正に改変されない．

[短期鍵公開サーバ（ShortTermKeyServer）] ユーザの短期鍵の登録と公開を行う．

[ユーザ（要求側／所持側）] ユーザ管理サーバに登録されたユーザであり，秘密鍵は漏えいしない．正しく認証が行われたユーザを正しいユーザとする．

[ホスト（要求側／所持側）]システムが正しくインストールされた端末であり，正しいユーザに使われている．

[サーバとの通信] ユーザ管理サーバとの通信は安全に行われ，通信情報の盗聴も入れ替えも行われない．また，IIS・LTKS・CMS との通信内容は入れ替えられない．

提案方式の安全性は Computational Bilinear Diffie-Hellman(CBDH) 仮定，l-Strong Diffie-Hellman(l-SDH)仮定に基づく．CBDH 問題が困難であればセッション鍵の復元は困難であること，l-SDH 問題が困難であれば短期鍵偽造によるなりすましは困難であることを示している．また，ID のすり替えによるホストのなりすまし，長期鍵の入れ替えによるユーザのなりすまし，ユーザの秘密鍵を利用したユーザのなりすまし，短期鍵の入れ替えによるユーザのなりすましについてはシステムの仮定から成り立たない．

本研究では，安全なホスト間直接通信型ファイル配送システムのモデルでシステム内でのホスト間直接通信を暗号化する方式を提案した．提案システムは，暗号化に必要な秘密をホスト側に移したことで，長期鍵公開サーバと短期鍵公開サーバは復号に必要な秘密情報を保持しないモデルになっている．また，短期鍵登録時にホスト自身がホスト ID と長期秘密鍵・短期秘密鍵による短期鍵の生成を行う．そして，このモデルではシステム起動時に毎回 ID が変更されるが，短期鍵の生成はその後に行われる．以上のようにすべての要件を満たすことを確認した．

ペアリング演算を行うためには，ペアリング関数の引数となる楕円曲線上の点や戻り値となる拡大体を扱えなければならない．さらに，楕円曲線や拡大体を扱うためには素体となる有限体上の演算の実装が必要となる．その演算に関わるコードをセキュアな Web アプリケーション開発に利用するためには，再利用が容易な形でまとめたライブラリとすることが望ましい．ペアリング演算に必要な関数をまとめてライブラリとすることで，開発者は有限体や楕円曲線，ペアリング演算のための複雑なアルゴリズムなど，高度な理論的知識を有することなく，プロトコルや方式を Web アプリケーションへ適用することができる．

C 言語，C++言語によるペアリング演算ライブラリが公開されているが，一般的な Web アプリケーションではこれらの言語は使用されていない．Java 言語のライブラリは C，C++ に比べて Web アプリケーション開発に適しているが，Java では Java 実行環境（JRE）のバージョンによりアプリケーションの動作が異なる．Web アプリケーション開発ではスクリプト言語がよく利用されており，その代表的な言語に JavaScript がある．JavaScript は Web ブラウザがソースコードを解釈して実行することから，JavaScript の解釈が Web ブラウザで同一の動作をするアプリケーションを開発するのは容易ではない．スクリプト言語のひとつに Flash コンテンツのコントロールに用いられる ActionScript がある．この言語は SWF 形式でコンパイルされ，クライアント側の仮想マシン上で実行される．この仮想マシン ActionScript Virtual Machine 2（AVM2）は Adobe Flash Player に組み込まれている．ActionScript は仮想マシンのバー

ジョンが同じ，つまり，インストールされている Adobe Flash Player のバージョンが同じならば，クライアントの環境に影響されずに同じ動作をするアプリケーションを容易に開発できる．Adobe Flash Player は Java と比較して広く普及していることからセキュアな Web アプリケーションを開発する言語として ActionScript に注目し，標数 3 の η_T ペアリング演算ライブラリを開発した．

本ライブラリの使用例として，最も基本的な方式である Boneh-Franklin（BF）方式を Web ブラウザ上で暗号化／復号する ID ベース暗号として実装した．ID ベース暗号を実現するには PKG（鍵生成センタ）と呼ばれる公開パラメータを配付する機能とユーザ ID をクライアントから受け取り，ユーザの復号鍵の生成・配布する機能を備える主体が必要となる．BF 方式のクライアントアプリケーションを実装するにあたり，BF 方式に対応した PKG を REST スタイルで実装した．サービス提供者，利用者を特定する ID として，サービス提供者が生成したサービス ID，利用者のメールアドレスをそれぞれ用いる．利用者はサービスを利用する前に，サービス利用登録を行い，サービス提供者にメールアドレスを渡す．利用者が暗号文を送信する場合，サービスサーバにアクセスし，公開パラメータとサービス ID，Web アプリケーションクライアントを受け取り，サービス ID で暗号化する．また，サービス提供者が暗号文を送信する場合，登録されているユーザのメールアドレスで暗号化する．以上のようにすることで双方が相手の ID を確認できるサービスとして実装した．送信者は PKG に対して公開パラメータが記述されたファイルの URL に GET メソッドによるリクエストを送信し，公開パラメータを取得する．その後，公開鍵生成と暗号化を行う．本ライブラリを用いることで，公開鍵の生成を 1 行，暗号化を 6 行のコードで実装できる．その他に，本ライブラリと外部ライブラリのインポート宣言を 2 行，PKG から公開パラメータを取得するコードを 27 行，文字列を多倍長整数型に変換するコードを 12 行記述し，合計 48 行で送信者のプログラムが実装可能であった．受信者は，ID を含む復号鍵リクエストを XML で記述して POST メソッドで PKG に送信し，復号鍵を含む XML レスポンスを受け取る．その後，復号鍵を使って復号する．本ライブラリを利用して暗号文を復号するのには 3 行のコードで実装可能であった．復号鍵取得の際の認証にはいくつかの方法が考えられるが，ここでは Email Answerback 認証と呼ばれる Email による本人確認を利用した認証を実装した．その部分のコード数は 32 行で，受信者のプログラムは合計 82 行で実装可能であった．

場所や端末を問わずにサービスを利用できるなどの利点から，クラウドコンピューティングの利用が広まっている．一方では，組織外へ情報を出すことの不安から利用をためらう企業もある．その不安の一因は，サービス提供者を信頼できるか判断できないことである．また，サービス提供者が自身の運用の品質を対外的に示すことは顧客の獲得につながるという考え方もある．そこで，安心・安全にクラウドを利用するために中立な第三者機関による監査を行うことが考えられる．利用者にとっての不安である，ファイルを組織外に出すことに関しては，利用者および提供者以外の監査人により運用の信頼性を保証されていれば，保証のない知恵強者よりも利用時の不安が緩和されると考えられる．また，責任の所在を明らかにする際にも，当事者同士だけでは意見がまとまらない場合に，第三者が提出された証拠を基にして，客観的立場から判断することで決着がつく．しかしながら，クラウドサービスでは一般的に暗号通信でアクセスするため，あるサービスに対する利用者が誰であるか，第三者により特定することは困難である．クラウド利用に対する不安を増やさないために，監査であってもできるだけ部外者に情報を知られないことが望ましい．そこで，クラウドサービス提供者のサービスの運用および利用者の利用の実績を第三者に示すことを目的にするならば，普段は監査人が平文を知ることなく監査ができ，問題発生時には平文とあわせることで通信内容を確認できる暗号方式が必要となる．本研究ではサービス利用及び運用時の通信内容を秘匿できること（要件 a），普段は監査人によってサービスの利用および運用の実績を第三者に示すことができ（要件ｂ），さらに問題発生時には通信された内容も確認できる（要件 c）クラウドサービスの利用者が特定できる ID ベース暗号方式を提案する．提案方式の主体は以下の通りである．

[サービス利用者] 信頼できる組織によって固有の ID が登録・発行されており，サービス利用のログを保存している．サービスを利用する主体であり，秘密情報を漏えいしない．

[サービス提供者] 信頼できる組織によって固有の ID が登録・発行されており，サービス提供のログを安全に保存する．サービスを提供する主体であり，秘密情報を漏えいしない．

[KGC] 暗号化と署名に必要な情報を公開しており，サービス利用者，提供者および定期監査人の ID 情報に対する秘密鍵・定期監査鍵の発行と，臨時監査人から渡された情報に対する臨時監査鍵を発行する主体である．プロトコルを遵守し，秘密情報の漏えいなどいかなる不正も行わない Trusted Third Party（TTP）である．

[定期監査人] 信頼できる組織によって固有の ID が登録・発行されており，KGC から発行される自身の ID に対する定期監査鍵を用いて，サービス提供者から受け取ったログによって利用及び運用の実績の監査を

行う.

[臨時監査人] 問題発生時にサービス提供者／要求者からログを受け取り，KGC に問題のログに対応した臨時監査鍵を要求する．入手した臨時監査鍵で暗号文を復号することで通信内容の監査を行う主体であり，プロトコルを遵守し，秘密情報の漏えいなどいかなる不正も行わない TTP である．

提案方式は，ID ベースの証明書不要で送信元の認証が 1 回の通信で可能な鍵確立方式 ID-based one-pass AKE（Authenticated Key Establishment）と，ID ベースでの検証者指定署名 ID-based strong designated verifier signature（ID-SDVS）を用いて構成する．ID-based one-pass AKE が安全であれば，提案方式の暗号化は安全である．また，ID-SDVS の偽造不可能性が破られないならば，提案方式の署名は安全である．

５．主な発表論文等

〔雑誌論文〕（計　1 件）

1. 毛利 公美，伴 拓也，白石 善明，“ActionScript によるηT ペアリング演算ライブラリー”，電子情報通信学会論文誌 D，Vol.J95-D，No.4，pp.799-811，2012 年（査読有）http://id.nii.ac.jp/1476/00005625/

〔学会発表〕（計　5 件）

1. 福田 洋治，白石 善明，毛利 公美，“電子鑑識の動向とネットワークフォレンジック”，電気関係学会関西連合大会（招待講演），2016 年 11 月，大阪府立大学（大阪府堺市）．
2. 野村 健太，毛利 公美，白石 善明，森井 昌克，“近隣サービスで同時検証するためのマルチグループ署名”，情報処理学会 コンピュータセキュリティシンポジウム（CSS2016），2016 年 10 月，秋田キャッスルホテル（秋田県秋田市）．
3. 本郷 考一，毛利 公美，白石 善明，“ActionScript による GF(3)上のηT ペアリング演算ライブラリ”，情報処理学会第 73 回全国大会，2011 年 3 月，東京工業大学（東京都目黒区）．
4. 宇都宮 秀利，毛利 公美，白石 善明，土井 洋，“ペアリングを用いた大小比較の秘匿計算の一手法”，情報処理学会第 73 回全国大会，2011 年 3 月，東京工業大学（東京都目黒区）．
5. 疇地 悠，毛利 公美，白石 善明，土井 洋，“信頼できる ID と一時的に発行された ID を結合した事後追跡可能性を有するペアリングに基づく鍵確立プロトコル”，情報処理学会 コンピュータセキュリティシンポジウム（CSS2010），2010 年 10 月，岡山コンベンションセンター（岡山県岡山市）

６．研究組織

(1)研究代表者

毛利　公美（Mohri，Masami）

岐阜大学・総合情報メディアセンター・准教授

研究者番号：５０２８１６９７