4X-7

メールアドレスを公開鍵とするWebベース機密情報伝送システムの試作

川村 舞† 伴 拓也† 白石 善明† 土井 洋†† 毛利 公美††† 福田 洋治‡ 岩田 彰† 野口 亮司‡‡
名古屋工業大学† 情報セキュリティ大学院大学†† 岐阜大学††† 愛知教育大学‡ (株)豊通シスコム‡‡

1. はじめに

機密情報を含むファイルをやり取りしたい際に，それを安全かつ簡便に行いたいというニーズが存在する．このニーズを満たすために利用できそうなものとして，(1) 公開鍵基盤(PKI), (2)大容量ファイル送受信サービスがあげられる．(1)は，高いレベルのセキュリティが必要なときに利用されている．しかし，送信者が受信者の公開鍵を得るための認証局との予備通信や証明書の管理などの導入と運用のコストがあり，広く使われるという状況にはなっていない． (2)は，Web ベースのシステムが多く，簡便に使うことができる．しかし，通信路は暗号化されているが，ファイル自体は暗号化されていない場合，サービス提供者にファイルの中身を見られてしまう可能性がある．ファイル自体を暗号化するサービスも存在するが，送受信者双方のユーザ登録，パスワードの発行などの導入コストがかかる．

そこで，安全かつ利用者の導入や運用コストの少ないシステムの提案を我々は行っている[1]．また，提案システムに適用する ID ベース暗号を基にした暗号方式を提案[1]している．提案システムの全体像を図 1 に示す．

本研究では，標準化なども進んでおり，ランダムオラクルモデル及び CBDH 仮定の下で IND-ID-CCA 安全性が保証されている Boneh，Franklin らの方式[2] を ID ベース暗号の具体例としている．

文献[3]では，システムの実利用を考え，既存の信頼できるシステムに近い仮定を設定し，システムの処理性能と安全性を考察した．文献[1]のシステムは現実的な運用において，既存の信頼できるシステムを活用できる，(1)と(2)の良さを持ったファイル送受信システムであることを示している．

システムの安全性としては，これまでに，全てのサーバの入力を攻撃者が得られる際の能動的攻撃に対する議論を行い，文献[1]のシステムが CCA 安全性を満たすシステムになるように改良を行い，文献[4]にて提案システムが CCA 安全性を満たすことを示した．本稿では，提案システムを Web ベースのシステムとして試作を行い，性能を確認する．

2. Web ベースファイル送受信システム

試作するシステムを構成するアルゴリズム[4]について説明する．これを二重暗号化 ID ベース方式(DEIBE)と呼ぶ．DEIBE は，以下の 8 つのアルゴリズムからなる．

PKG.Setup: 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 $params$ とマスター鍵 msk を出力する．

PKG.Ext: 公開パラメータ $params$，マスター鍵 msk，ID を入力とし，秘密鍵 d_{ID} を出力する．

RCD.KG: 公開パラメータ $params$ を入力とし，公開鍵 $RCD.PK$，秘密鍵 $RCD.SK$ を出力する．

Prototyping of Web-based Confidential Information Transmission System Using E-mail Address as Public Key
†Mai KAWAMURA and Takuya BAN and Yoshiaki SHIRAISHI and Akira IWATA・Nagoya Institute of Technology
††Hiroshi DOI・Institute of Information Security
†††Masami MOHRI・Gifu University
‡Youji FUKUTA・Aichi University of Education
‡‡Ryoji NOGUCHI・Toyotsu Syscom Corp.

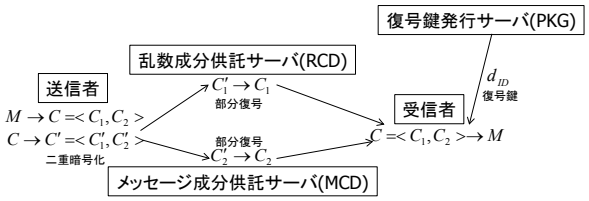

図1 提案システムの全体像

MCD.KG: 公開パラメータ $params$ を入力とし，公開鍵 $MCD.PK$，秘密鍵 $MCD.SK$ を出力する．

Enc: 公開パラメータ $params$，メッセージ M 及び ID を入力とし，暗号文 $C =< C_{RCD}, C_{MCD} >$ を出力する．

RCD.Enc: RCD の公開鍵 $RCD.PK$ 及び部分暗号文 C_{RCD} を入力とし，部分暗号文 C'_{RCD} を出力する．

MCD.Enc: MCD の公開鍵 $MCD.PK$ 及び部分暗号文 C_{MCD} を入力とし，部分暗号文 C'_{MCD} を出力する．

RCD.Dec: RCD の秘密鍵 $RCD.SK$，部分暗号文 C'_{RCD} を入力として，部分復号結果 C_{RCD} もしくは $\perp$ (復号失敗) を出力する．

MCD.Dec: MCD の秘密鍵 $MCD.SK$，部分暗号文 C'_{MCD} を入力として，部分復号結果 C_{MCD} もしくは $\perp$ (復号失敗) を出力する．

Dec: ID に対応する秘密鍵 d_{ID}，部分復号結果 $C =< C_{RCD}, C_{MCD} >$ を入力とし，メッセージ M もしくは $\perp$(復号失敗)を出力する．

ここで，Dec，RCD.Dec 及び MCD.Dec はいずれも確定的な（部分）復号アルゴリズムであり，まず，任意のメッセージ M に対し，$M = Dec(params, Enc(params, M, ID), d_{ID})$ が成り立つ．さらに，$< C_{RCD}, C_{MCD} >= Enc(params, M, ID)$ と置くと，

$C_{RCD} = RCD.Dec(RCD.SK, RCD.Enc(RCD.PK, C_{RCD}))$
$C_{MCD} = MCD.Dec(MCD.SK, MCD.Enc(MCD.PK, C_{MCD}))$

も成り立つ．

3. 提案システムの試作

3.1 実装環境

送信者，受信者が利用するクライアントアプリケーションと乱数成分供託サーバ（RCD），メッセージ成分供託サーバ（MCD），復号鍵発行サーバ（PKG）を作成し，提案システムを試作した．2 つのクライアントアプリケーションを Adobe ActionScript3.0 を利用し，Web ブラウザで実行するアプリケーションである Adobe Flex フレームワークを利用して作成した．3 つのサーバプログラムを Java で作成した．サーバ，クライアント共に同一のマシン上で実行した．実行環境を表 1 に示す．

サーバとクライアントの通信手段として，HTTP の POST メソッドを利用した．提案システムのクライアント側の処理で必要となるペアリング関数として，標数 3 の η_T ペアリング[5]を用いた．楕円曲線上の点を表すのに用いる有限体の要素数 q は 3^{193} に設定した．Adobe ActionScript 3.0 で標数 3 の η_T ペアリング演算を実現するためのライブラリとして，As3Pairing[6], As3Crypto[7]を利用した．

3.2 実装したシステム

システム構成を図 2，実装したシステムの流れを図 3 に示す．

(1) 送信者は，受信者の ID と平文，乱数，公開情報をもとに暗号文 $C' =< C'_{RCD}, C'_{MCD} >$ を作成する．

(2) 送信者は，ID と C'_{RCD} を RCD に，ID と C'_{MCD} を MCD に送信する．

(3) RCD は，C'_{RCD} を部分復号して C_{RCD} を作成し，C_{RCD}^{ID} として，ID と暗号文を関連付けて保管する．

(4) MCD は，C'_{MCD} を部分復号して C_{MCD} を作成し，C_{MCD}^{ID} として，ID と暗号文を関連付けて保管する．

(5) 受信者は，ID，パスワードと受信したい暗号文の情報を RCD，MCD に送信し，RCD，MCD は，それぞれ受信者の認証を行う．認証結果が正しければ，RCD は C_{RCD}^{ID} を，MCD は C_{MCD}^{ID} を受信者に渡す．

(6) 受信者は，ID とパスワードを PKG に送信し，PKG は受信者の認証を行い，認証結果が正しければ，復号鍵 d_{ID} を受信者に渡す．

(7) 受信者は，各サーバから得た C_{RCD}^{ID}，C_{MCD}^{ID}，d_{ID} から M を復号する．

なお，(6)の手続きは初めて受信者が復号するときに行う．事前に復号鍵を取得している場合，もしくは 2 回目以降の復号では(6)の処理をスキップする．

3.3 処理時間

試作したシステムを動作させ，RCD，MCD の部分復号結果 C_{RCD}, C_{MCD} は，送信者が計算した $C =< C_{RCD}, C_{MCD} >$ と一致すること，そして，受信者の復号結果は，送信者が送信した M と一致することを確認した．

試作したシステムで処理時間を計測する．平文 M は，暗号化に用いるセッション鍵を想定し，160 ビットの数値とした．送信者による暗号化の命令を受け取ってから暗号化が終了するまでの時間と，RCD，MCD の部分復号に要した時間と，受信者が暗号文を受け取ってから復号が終了するまでの時間を測定した．10 回の測定結果の平均を表 2 に示す．

試作した環境では，送信者のクライアントでの暗号化は 4 秒程度，受信者のクライアントでの復号は 1 秒以内で処理が可能であった．

4. おわりに

本稿では，文献[4]で提案した CCA 安全が確認されているシステムを Web ベースのシステムとして試作した．提案システムは，Web ベースのアプリケーションとして正しく動作することを確認し，送信者の暗号化に約 5 秒，サーバの部分復号に約 2~5 秒，受信者の復号に約 0.5 秒と現実的な処理時間で動作することが確認できた．

Web ベースで動作させられることから，利用者は特別なソフトのインストールなどの必要がなくなり，システムの利用を容易にできる．

参考文献

[1]川村舞，白石善明，毛利公美，土井洋，“信頼できるメールアドレスを公開鍵とする Web ベース機密情報伝送システムの提案”，情報処理学会第 72 回全国大会，第 3 分冊，p.605，2010.

[2]D. Boneh, M. Franklin, “Identity-based Encryption from the Weil Pairing”, SIAM J. of. Computing, Vol. 32, No. 3, pp.586-615, 2003.

[3]川村舞，白石善明，土井洋，毛利公美，福田洋治，岩田彰，野口亮司，“Web ベースファイル送受信システムの処理性能と安全性に関する考察”，情報処理学会第 73 回全国大会，第 3 分冊，pp.479-480，2011.

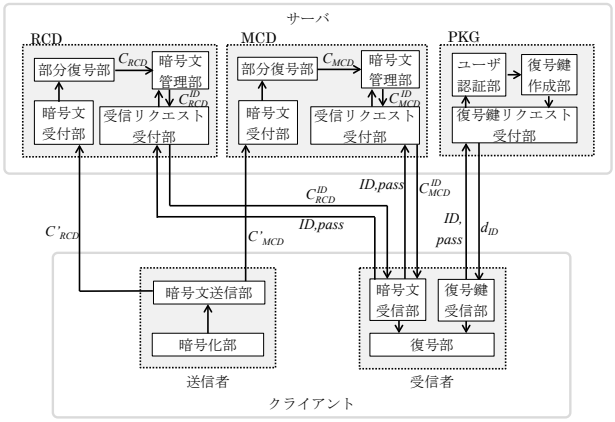

図 2　システム構成

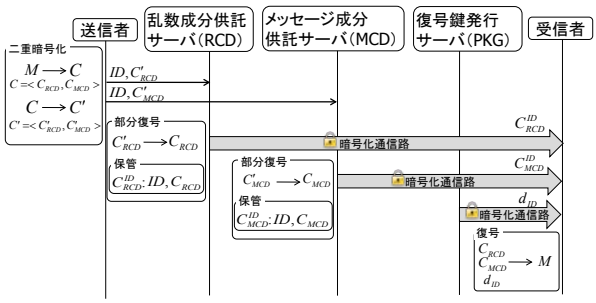

図 3　提案システムの流れ

表 1　試作システムの実行環境

MCD，RCD，PKG，クライアン共通	
CPU	Intel(R) Core i 5 M560 2.67GHz
RAM	4.0 GB
OS	Windows 7 Professional 64bit
MCD，RCD，PKG	
ランタイム	JRE 1.3.1_01
サーバ	Apache Tomcat 7
クライアント	
ブラウザ	Firefox 6.0.2
ランタイム	Adobe Flash Player 11.1.102.55

表 2　暗号化/復号に要した時間[ms]

送信者の暗号化	4431
RCD の部分復号	1746
MCD の部分復号	4316
受信者の復号	402

[4]川村舞，伴拓也，白石善明，土井洋，毛利公美，福田洋治，岩田彰，野口亮司，“ID ベース暗号を用いた複数サーバによる機密情報伝送システム”，SCIS2012，2012.

[5]P. Barreto, S. Galbraith, C. O’hEigeartaigh, M. Scott, “Efficient Pairing Computation on Supersingular Abelian Varieties”, Cryptology ePrint Archive, 2004/375.

[6]伴拓也，毛利公美，白石善明，野口亮司，“ActionScript による η_T ペアリング演算ライブラリ”，DICOMO2011，pp.1285-1295，2011.

[7]H. Torgemane, “as3crypto -Project Hosting on GoogleCode-(online)”，http://code.google.com/p/as3crypto/(accessed 2011-12-7).