6W-05

ミリタリーネットワークのための複数の属性管理機関を持つ属性失効機能付き属性ベース暗号

野村　健太†　　毛利　公美‡　　白石　善明†　　森井　昌克†
神戸大学†　　岐阜大学‡

1　はじめに

ミリタリー環境において作戦の詳細や上官からの命令などの機密情報は厳密にアクセス制御を行う必要がある．また，鍵を生成する管理機関は敵からの侵略に備え，分散させなければならない．

Hur[1]らはミリタリーネットワークへ適用するための次の 3 つの要件を満たす暗号文ポリシ属性ベース暗号（Ciphertext-Policy Attribute-Based Encryption:CP-ABE[2]）を提案している．

【要件 1】複数の鍵管理機関が独立してユーザの属性を管理している
【要件 2】ユーザの属性を即時に失効ができる
【要件 3】結託耐性，データの機密性，前方秘匿性・後方秘匿性を満たしている

Hur らの方式は厳密な安全性証明は与えられていない．またユーザが新たに属性を取得したり，失効したりする際にはユーザ自身が秘密鍵の更新を行う必要がある．しかし，ユーザの秘密鍵は保有する通信機器に埋め込まれ鍵の更新は実際には困難である．さらに通信システムを可能な限り複雑にしないようにストレージからユーザに伝送される暗号文は固定長であることが望ましい．

本稿では，証明可能安全な CP-ABE 方式[3]を拡張し，次の 2 つの要件を追加で満たすことができる CP-ABE 方式を提案する．

【要件 4】秘密鍵の更新が不要
【要件 5】鍵と暗号文が固定長

2　提案方式

2.1　システムの構成

提案方式は図 1 に示すようにユーザ，データ送信者，鍵管理機関（センター，ローカル），ストレージで構成される．

【ユーザ】ストレージに保管されている共有データをダウンロードし，アクセス構造を満たす属性を有するユーザは復号できる．
【データ送信者】データを暗号化し，ストレージにアップロードする．
【鍵管理機関】 センターがユーザを管理し，複数あるローカルがユーザの属性を管理する．センターとローカルが共同で秘密鍵と再暗号化鍵を発行する．
【ストレージ】共有データを保管する．また再暗号化鍵を使って共有データを再暗号化する．

2.2　アクセス構造

提案方式のアクセス構造は，1 つの AND ゲートのみで構成されるものとし，アクセス構造Wを$W = \wedge_{i \in I} \underline{i}$と表す．属性集合$I$は，アクセス構造$W$において考慮する必要がある属性の集合である．

2.3　アルゴリズム

提案方式は Auth.Setup，Auth.Ext，DO.Enc，C.ReEnc，U.Dec の 5 つのアルゴリズムからなる，Auth.Setup，Auth.Ext は鍵管理機関，DO.Enc はデータ送信者，C.ReEnc はストレージ，U.Dec はユーザによって実行される．

-Auth.Setup：2 つのサブアルゴリズム C.Setup.Sub，L.Setup.Sub からなり，セキュリティパラメタλを入力とする．

-C.Setup.Sub：セキュリティパラメタλを入力とする．システム全体の属性空間を$\mathcal{U} = \{1, \dots, n\}$とし，これを m 個の各ローカルに割り振る．ローカル A_k が管理する属性空間を$\mathcal{U}_k = \{1, \dots, n_k | 1 \le k \le m\}$とする．すなわち，$n := \sum_{k=1}^{m} n_k$とする．次に素数$p$，素数位数$p$，生成元$P \in G$，双線形写像$e: G \times G \to G_T$を選ぶ．乱数$\beta \in Z_p$を選び，$h = P^{\beta}$を計算する．以上よりセンターCA の公開鍵$PK_{CA} := (e, P, h)$，マスタ秘密鍵$MK_{CA} := \beta$，ローカル A_k の管理する属性空間を$\mathcal{U}_k$を出力する．

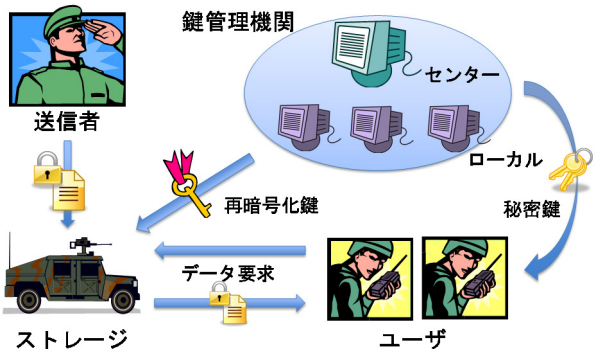

図 1:　提案方式のモデル

-L.Setup.Sub：A_k の管理する属性空間$\mathcal{U}_k$を入力とする．ランダム値$y_k, t_{(k,1)}, \dots, t_{(k,3n_k)}, d_{(k,1)}, \dots, d_{(k,2n_k)} \in Z_p$を選び，以下を計算する．

$$Y_k := e(P, P)^{y_k}$$

$$T_{(k,i)} := \begin{cases} d_{(k,i)} P & (1 \le i \le 2n_k) \\ t_{(k,i)} P & (2n_k + 1 \le i \le 3n_k) \end{cases}$$

$$rk_{(k,i)} := \frac{t_{(k,i)}}{d_{(k,i)}} (1 \le i \le 2n_k)$$

公開鍵$PK_{A_k} := (Y_k, T_{(k,1)}, \dots T_{(k,3n_k)})$，マスタ秘密鍵$MK_{A_k} := (y_k, d_{(k,1)}, \dots d_{(k,2n_k)}, t_{(k,1)}, \dots, t_{(k,3n_k)})$，再暗号化鍵$RK_{A_k} := (rk_{(k,1)}, \dots rk_{(k,2n_k)})$を出力する．

L.Setup.Sub アルゴリズムをローカルの数だけ繰り返し，公開鍵$PK := (PK_{CA}, PK_{A_1}, \dots, PK_{A_m})$，マスタ秘密鍵$MK := (MK_{CA}, MK_{A_1}, \dots, MK_{A_m})$，再暗号化鍵$RK := (RK_{A_1}, \dots, RK_{A_m})$を出力する．

-Auth.Ext：提案方式ではユーザの秘密鍵成分は単一の個人鍵と複数の属性鍵で構成される．各機関のいずれも個別のユーザ鍵の構成要素全体を決定することが出来ないように，供託問題を解決する鍵供託問題算術安全な 2PC プロトコルを利用する．サブアルゴリズム C.Ext.Setup，L.Ext.Per1，C.Ext.Per，L.Ext.Per2，U.Ext.Per，L.Ext.Att，U.Ext.Att と 2PC プロトコルである Auth.Ext.Per.2PC から構成される．

-C.Ext.Setup：属性空間$\mathcal{U}$を入力とする．A_k の管理する各$i \in \mathcal{U}_k$について，$r_{(k,i)} \in Z_p$をランダムに選び，$\gamma_k := \sum_{i=1}^{n_k} r_{(k,i)}$とする．他のローカルで管理する属性でも同様の動作を行い，$r_t := \sum_{i=1}^{m} \gamma_k$とする．この$r_t$が各ユーザのユニークな値となる．またランダム値$r' \in Z_p$を選ぶ．$\left(\left(1/r'\right), \{r' r_{(k,i)} P, \gamma_k | k \in \{1, \dots, m\}, i \in \mathcal{U}\}\right)$を出力とし，$\left(1/r'\right)$をユーザに，$\{r' r_{(k,i)} P | k \in \{1, \dots, m\}, i \in \mathcal{U}\}$を A_k に送信する．

-Auth.Ext.Per.2PC：CA と A_k は 2PC プロトコルを利用し，CA から秘密値(γ_k, β)，A_k から秘密値y_kを入力として，安全な 2PC プロトコルは$x = (y_k + \gamma_k)\beta$を計算する．$x$を出力として A_k に送信する．

-L.Ext.Per1：A_k はランダム値$\tau \in Z_p$を選び，τ, xを入力とする．$T = (x/\tau)P$を計算し，Tを出力として CA に送信する．

-C.Ext.Per：Tとマスタ秘密鍵βを入力とする．$B = (1/\beta^2)P$を計算し，Bを出力として A_k に送信する．

Attribute Revocable Attribute-Based Encryption with Multiple Key Authorities for Military Networks

† Kenta NOMURA, Yoshiaki SHIRAISHI and Masakatu MORII・Kobe University
‡ Masami MOHRI・Gifu University

Copyright ©2015 Information Processing Society of Japan.
All Rights Reserved.

-L.Ext.Per2：τ, Bを入力とする．個人鍵要素$D'_k = \tau B$を計算し，D'_kを出力としてユーザに安全に送る．

-U.Ext.Per：各ローカルから送られてくる個人鍵要素D'_kを入力とする．個人鍵$D' = \prod_{i=1}^{m} D'_k$を計算し，個人鍵$D'$を出力する．

-L.Ext.Att：マスタ秘密鍵MK_{A_k}ユーザの属性集合S_k及び$\{r' r_{(k,i)} P | k \in \{1, \dots, m\}, i \in \mathcal{U}\}$を入力とする．各$i \in \mathcal{U}_k$について属性鍵要素$D_{(k,i)}, F_{(k,i)}$を以下のように計算する．

$$D_{(k,i)} := \begin{cases} \left(1/t_{(k,i)}\right)\left(r' r_{(k,i)} P\right) & (i \in S_k) \\ \left(1/t_{(k,n_k+i)}\right)\left(r' r_{(k,i)} P\right) & (i \notin S_k) \end{cases}$$

$$F_{(k,i)} := \left(1/t_{(k,2n_k+i)}\right)\left(r' r_{(k,i)} P\right)$$

属性鍵要素$D_{(k,i)}, F_{(k,i)}$を出力として，ユーザに送信する．

-U.Ext.Att：$(1/r')$と属性鍵要素$D_{(k,i)}, F_{(k,i)}$を入力として，以下を計算する．

$$D_i := (1/r')\left(D_{(1,i)} \cup \dots \cup D_{(m,i)}\right)$$

$$F_i := (1/r')\left(F_{(1,i)} \cup \dots \cup F_{(m,i)}\right)$$

個人鍵$\{(D_i, F_i) | i \in U\}$を出力する．

以上より，秘密鍵$\mathrm{SK} := (D', \{(D_i, F_i) | i \in U\})$を取得する．

-DO.Enc：公開鍵PKと平文M，アクセス構造Wを入力とする．まず，ランダムに$s \in Z_p$を選ぶ．次に$C = h^s = (P^\beta)^s, \tilde{C} = Me(P,P)^{(y_1+\dots+y_m)s}$を計算する．$\tilde{C}$は$\tilde{C} = M \cdot (Y_1 \times \dots \times Y_m)^s = Me(P,P)^{(y_1+\dots+y_m)s}$として得られる．各$i \in \mathcal{U}_k$について以下のように計算する．

$$C'_{(k,i)} := \begin{cases} sT_{(k,i)} & (i \in I \wedge \underline{i} = i) \\ sT_{(k,n_k+i)} & (i \in I \wedge \underline{i} = \neg i) \\ sT_{(k,2n_k+i)} & (i \notin I) \end{cases}$$

$\mathcal{U}_1$から$\mathcal{U}_m$まで上記計算を行う．以上より暗号文$CT' := \left(W, C, \tilde{C}, \{C'_i | i \in U\}\right)$を出力する．ここで$C'_i = \cup_{k=1}^{m} C'_{(k,i)}$とする．

-C.ReEnc：暗号文CT'，ユーザの属性集合S，再暗号化鍵RKを入力とする．各$i \in \mathcal{U}_k$について以下のように計算する．

$i \in S_k \wedge (i \in I \wedge \underline{i} = i)$の場合

$$C_{(k,i)} := rk_{(k,i)} \cdot C'_{(k,i)} = \frac{t_{(k,i)}}{d_{(k,i)}} \cdot s \cdot d_{(k,i)} P = s \cdot t_{(k,i)} P$$

$i \notin S_k \wedge (i \in I \wedge \underline{i} = \neg i)$の場合

$$C_{(k,i)} := rk_{(k,n_k+i)} \cdot C'_{(k,i)} = \frac{t_{(k,n_k+i)}}{d_{(k,n_k+i)}} \cdot s \cdot d_{(k,n_k+i)} P = s \cdot t_{(k,n_k+i)} P$$

それ以外の属性については$C_{(k,i)} = C'_{(k,i)}$とする．

$\mathcal{U}_1$から$\mathcal{U}_m$まで上記計算を行う．以上より暗号文$CT := \left(W, C, \tilde{C}, \{C_i | i \in U\}\right)$を出力する．ここで$C_i = \cup_{k=1}^{m} C_{(k,i)}$とする．

-U.Dec：秘密鍵SK，再暗号化文CTを入力とする．各$i \in \mathcal{U}_k$について以下のように計算する．

$i \in S_k \wedge \underline{i} = i$の場合

$$e\left(C_{(k,i)}, D_{(k,i)}\right) := e\left(s \cdot t_{(k,i)} P, \frac{r_{(k,i)}}{t_{(k,i)}} P\right) = e(P,P)^{s \cdot r_{(k,i)}}$$

$i \notin S_k \wedge \underline{i} = \neg i$の場合

$$e\left(C_{(k,i)}, D_{(k,i)}\right) := e\left(s \cdot t_{(k,n_k+i)} P, \frac{r_{(k,i)}}{t_{(k,n_k+i)}} P\right) = e(P,P)^{s \cdot r_{(k,i)}}$$

$i \notin I$の場合

$$e\left(C_{(k,i)}, D_{(k,i)}\right) := e\left(s \cdot t_{(k,2n_k+i)} P, \frac{r_{(k,i)}}{t_{(k,2n_k+i)}} P\right) = e(P,P)^{s \cdot r_{(k,i)}}$$

$\mathcal{U}_1$から$\mathcal{U}_m$まで上記計算を行い，以下のように平文Mを復号する．

$$e(P,P)^{\gamma_t s} = \prod_{k=1}^{m} e(P,P)^{\gamma_k s} = \prod_{k=1}^{m} \left(\prod_{i=1}^{n_k} e(P,P)^{s \cdot r_{(k,i)}}\right)$$

$$\frac{\tilde{C}}{\frac{e(C,D')}{e(P,P)^{\gamma_t s}}} = \frac{Me(P,P)^{(y_1+\dots+y_m)s}}{e(P,P)^{(y_1+\dots+y_m)s}} = M$$

平文Mを出力する．属性失効の手順を次節に示す．

2.4 属性の失効手順

属性の失効は以下の手順によって行われる．

1. 失効させたい属性を管理するローカルがユーザの属性情報をストレージに送信する．
2. ストレージはその情報を元にユーザの属性集合を更新する．
3. 属性を失効したユーザがストレージ上のデータにアクセスした際に，ストレージは再暗号化アルゴリズムを実行しデータを再暗号化してユーザに送信する．

表 1: サイズの比較

	Hurらの方式[1]		提案方式
公開鍵のサイズ	2|G|+m|G_T| +log|N|×|K|		(3|U|+2+m) ×|G|＋2m|G_T|
秘密鍵のサイズ	(2|S|+1)×|G|	[max] (2|U|+1)×|G| [min] (2+1)×|G|	(2|U|+1)×|G_T|
暗号文のサイズ	(2|I|+1)×|G| +|G_T|+|G_{tree}|	[max] (2|U|+1)×|G| +|G_T|+|G_{tree}| [min] (2+1)×|G| +|G_T|+|G_{tree}|	(|U|+1) ×|G|+|G_T|
再暗号鍵のサイズ	(2|N|-1)×|K|		2|U|×|Z_p|

\|U\|:システム全体の属性数，m:システム全体のLocal authorities数，
\|S\|:ユーザが有する属性数，\|I\|:アクセス構造に含まれる属性の数
\|N\|:ユーザの総数，\|K\|:共通鍵のサイズ，\|G\|，\|G_T\|:群G，G_Tの要素のビットサイズ
\|G_{tree}\|:アクセスツリーTのビットサイズ

3 性能評価とまとめ

3.1 要件 4について

提案方式における属性失効は2.4節にしたがって行われる．属性失効の際に更新されるのはユーザの属性集合のみであり，ユーザの持つ秘密鍵は更新されない．したがって提案方式は【要件4】を満たす．なお，Hruらの方式ではストレージによる再暗号化に加え，ユーザによる秘密鍵の更新が必要となる．

3.2 要件 5について

鍵と暗号文のサイズについて比較を行った結果を表1に示す．提案方式は鍵，暗号文とも同一システム内では全て固定長である．すなわち提案方式は【要件5】を満たす．

暗号文のサイズはアクセス構造に含まれる属性数がシステム全体の属性数の半数以上ならば，Hurらの方式より提案方式が優れている．再暗号化鍵のサイズはユーザの総数がシステム全体の属性数より多ければ，提案方式の方が優れている．

3.3 計算量と安全性

計算量について比較を行ったところ次のようになった．暗号化，秘密鍵生成の計算量はHurらの方式では実行する際の条件によって異なるが，提案方式では同一システム内で固定されている．秘密鍵生成の計算量は提案方式が大きい．再暗号化の計算量は更新する属性数が同じであるならば等しい．

結託耐性，データの機密性，前方秘匿性・後方秘匿性について，Hurらの方式と同様に満たすことをインフォーマルに確認している．文献[3]と同様にしてフォーマルに証明する予定である．またアクセス構造においてより自由度の高いアクセス構造を扱えるようにすることが今後の課題である．

参考文献

[1] Hur, J. and Kang, K. “Secure Data Retrieval for Decentralized Disruption-Tolerant Military Networks,” IEEE/ACM TRANSACTIONS ON NETWORKING, Vol. 22, No. 1, pp.16-26 (2014).

[2] Bethencourt, J., Sahai, A. and Waters, B.:Ciphertext-Policy Attribute-Based Encryption, Proc. 2007 IEEE Symposium on Security and Privacy, pp.321-334 (2007).

[3] 成瀬 猛,毛利 公美,白石 善明, “前方秘匿性を満たす属性失効機能付き属性ベース暗号, ” 情報処理学会論文誌, Vol.55, No.10, pp.2256-2264 (2014).

Copyright ©2015 Information Processing Society of Japan.
All Rights Reserved.