

階層的ID-based暗号を用いたグループ鍵管理に関する一考察

毛利 寿志[†] 小野 良司[†]

[†] 三菱電機 情報技術総合研究所

〒247-8501 神奈川県鎌倉市大船5-1-1

E-mail: †{ Mori.Hisashi@cw, Ono.Ryoji@aj } .MitsubishiElectric.co.jp

あらまし 階層化されたグループ構成を有し、各グループが自律的に動作するようなネットワークでは、各グループの判断でグループメンバに対し公開鍵／秘密鍵ペアを付与、または剥奪できることが望ましい。さらに、グループメンバ全員に同一の公開鍵／秘密鍵ペアを持たせ、グループメンバ全員に対して暗号化マルチキャストができることが望ましい。公開鍵／秘密鍵ペアの発行権限を分散できる暗号技術として、階層的ID-based暗号がある。しかし、既存手法では、グループに対する公開鍵／秘密鍵ペアの発行が想定されていない。本研究では、階層的ID-based暗号を用いたグループ鍵配布方法、及びグループ鍵更新方法を提案し、その安全性について考察する。また、提案手法について、通信量、計算量に関する評価を行う。
キーワード 鍵管理方式、暗号化マルチキャスト、ユビキタスシステム

A Group Key Management Scheme Using Hierarchical ID-based Cryptography

Hisashi MOHRI[†] and Ryoji ONO[†]

[†] Information Technology R&D Center, Mitsubishi Electric Corporation

Ofuna 5-1-1, Kamakura, Kanagawa, 247-8501 Japan

E-mail: †{ Mori.Hisashi@cw, Ono.Ryoji@aj } .MitsubishiElectric.co.jp

Abstract This study investigates a key management scheme for hierarchical and dynamic group structures. Existing schemes based on hierarchical identity based cryptography, in which hierarchically arranged Private Key Generators are privileged to issue users with the private keys corresponding to their identities, can be adapted for self-organized networks, but do not consider how to issue and manage group key pairs, derived from group identities and shared by all members of the group. We consider in this paper a new group key management scheme based on the hierarchical identity based cryptography and discuss the security of the scheme. Also, the communication and computation cost of the proposed scheme is investigated analytically.

Key words Key management scheme, Secure multicast, Ubiquitous system

1. ま え が き

現実世界では、ユーザ個人間の通信だけでなく、複数のユーザによってグループが構成され、グループに対してメッセージを送信し情報を共有することが起こり得る。例として、図1のようなネットワークを想定する。図1のネットワークは、複数のユーザ（ノードと呼ぶ）とそれ

らを管理する特別なノード（基地局と呼ぶ）から構成され、ノードはさらにグループA, B, Cを構成する。グループに属するノードをグループメンバ、グループ内でグループメンバに対し命令を出す特別なノードをグループリーダー、どのグループリーダーにもなっていないノードを末端メンバと呼ぶ。また、グループCのように、グループメンバ全員が他のグループBにも属するとき、グループ

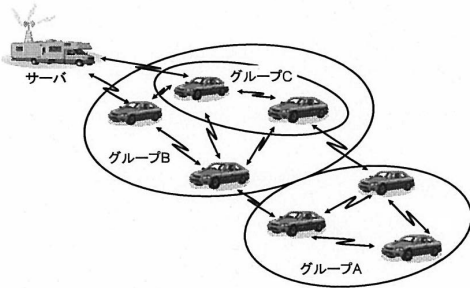


図1 想定するグループ階層

グループCはグループBのサブグループと呼ぶ。

メッセージの内容を宛先のノード以外から秘匿したい場合には、暗号化して送信することが望ましい。グループメンバー全体に対して同一のメッセージを暗号化して送信する場合、グループメンバーのみが保持する秘密情報（グループ鍵と呼ぶ）を導入することで、メンバー毎に暗号化して送信することなく、一度の暗号化と送信で情報を共有できる。また、グループ鍵を用いるには、メンバーの脱退や追加を想定し、グループ構成の変更に沿った鍵更新機能を備える必要がある。

階層構造を有するグループ構成に沿った鍵管理方式としては、Logical Key Hierarchy [4] (LKH) があるが、LKHは木構造に沿って共通鍵を割り当てるため、グループメンバー以外のノードがグループに対して暗号化したメッセージを送信することができない。

グループメンバー以外からグループへの暗号化通信は、公開鍵暗号を適用することで容易に実現できる。階層構造を有する公開鍵暗号系としては、階層的ID-based暗号(HIBE) [1], [3] がある。HIBEでは、公開鍵/秘密鍵ペアの発行権限を階層化された発行局に分散でき、本研究で想定するグループ構造との親和性が高い。しかし、これまでの研究では、HIBEへのグループ鍵の適用や、その際の鍵管理方式に関する議論は行われていない。

本研究では、HIBEを用いた鍵事前配布方法、グループ鍵配布方法、及びグループ鍵更新方法を備えた鍵管理方式を提案する。さらに、提案手法の通信量、計算量とセキュリティに関する評価を行う。

2. 準備：階層的ID-based暗号

本節では、提案手法を述べるための準備として、ID-based暗号（以降、IBE）、及びHIBEについて述べる。

IBE [2] は、公開鍵暗号系の一種であり、公開鍵として任意のビット列を用いることができる暗号である。例えば、メールアドレスなどを公開鍵とすることにより、公開

鍵暗号を運用する際の問題点であった公開鍵の管理が単純化されるという利点がある。IBEでは、任意のビット列に対して秘密鍵を作成するセンターであるPKG (Private Key Generator) と、PKGに対してビット列を渡し、対応する秘密鍵を得るユーザが存在する。

HIBEは、IBEを拡張し、Root PKG, Lower-level PKGという階層型のPKGを導入した暗号技術である。HIBEについては複数の手法が提案されているが、本稿では代表的な方式 [3] を元に、ペアリング写像を元にしたアルゴリズム構成を述べる。

大きな素数 q に対し、位数 q の加法が定義された巡回群を G_1 、乗法が定義された巡回群を G_2 としたとき、以下の関係が成り立つような $\hat{e}: G_1 \times G_1 \rightarrow G_2$ をペアリング写像と呼ぶ。

- 双線形性 (Bilinear) : 任意の $Q, R \in G_1$ かつ任意の $s, r \in \mathbb{Z}$ に対し、 $\hat{e}(sQ, rR) = \hat{e}(Q, R)^{sr}$ 。
- 非縮退性 (Non-degenerate) : $G_1 \times G_1$ のどのペアも、 G_2 の単位元に写像されない。
- 計算可能性 (Computable) : 任意の $Q, R \in G_1$ に対し、 $\hat{e}(Q, R)$ を計算する効率のよいアルゴリズムが存在する。

HIBEでは、Root PKG, Lower-level PKG, ユーザの三者が階層を構成する。Root PKGは全システムパラメータを決定し、任意のビット列（公開鍵）から秘密鍵を生成できる。Lower-level PKGはRoot PKGの下に位置し、同様に任意のビット列から秘密鍵を生成できる。Lower-level PKGは複数の階層を構成してもよい。ユーザは階層構造の末端に位置する。HIBEは、Root_Setup, Lower-level_Setup, Extraction, Encryption, Decryptionの5種類のアルゴリズムより構成される。

- Root_Setup Root PKGは、以下の作業を行う。

(1) セキュリティパラメータ $K(> 0)$ を入力としてBDHパラメータ生成器 [3] を実行し、素数 q 、位数が q である群 G_1 と G_2 、ペアリング写像 $\hat{e}: G_1 \times G_1 \rightarrow G_2$ を生成する。

(2) 任意の生成元 $P_0 \in G_1$ を選択する。

(3) ランダムな値 $s_0 \in \mathbb{Z}/q\mathbb{Z}$ を選択し、 $Q_0 = s_0 P_0$ とおく。

(4) ハッシュ関数 $H_1: \{0, 1\}^* \rightarrow G_1$ と、ある k に対するハッシュ関数 $H_2: G_2 \rightarrow \{0, 1\}^k$ を選択する。

このとき、メッセージ空間 $\mathcal{M} = \{0, 1\}^k$ 、深さ h のメッセージ受信者に対する暗号文空間は $G_1^h \times \{0, 1\}^k$ である。Root PKGは、以下の組 $params$ をシステムパラメータとして公開する。

$$params = (G_1, G_2, \hat{e}, P_0, Q_0, H_1, H_2). \quad (1)$$

Root PKG の保持する情報は s_0 であり、これは下位レベルの秘密鍵を作るためのマスタ鍵となる情報である。

- **Lower-level-Setup** 階層 t に位置する Lower-level PKG は、さらに下位レベルの秘密鍵を作成するためのマスタ鍵 $s_t \in \mathbb{Z}/q\mathbb{Z}$ を作成し、保持する。

- **Extraction** 階層 t に対応する ID_t の ID 組 (これを $IDtuple_t$ とおく) を以下のように表す。

$$IDtuple_t = (ID_1, ID_2, \dots, ID_{(t-1)}, ID_t).$$

ただし、 ID_2 は、 ID_1 の子頂点かつ頂点 ID_t の先祖頂点、 $ID_{(t-1)}$ は、 ID_t の親頂点とする。また、 S_0 を G_1 の単位元とする。ノード ID_t の親頂点は、 $P_t = H_1(IDtuple_t)$ を計算し、 ID_t の秘密鍵 S_t と、暗号文の復号に必要な情報 $Q_{(t-1)}$ をそれぞれ以下のように計算する。

$$S_t = S_{(t-1)} + s_{(t-1)}P_t, \quad (2)$$

$$Q_{(t-1)} = s_{(t-1)}P_0. \quad (3)$$

この S_t と $Q_1, Q_2, \dots, Q_{(t-1)}$ をノード ID_t に格納する。

- **Encryption** メッセージ $M \in \mathcal{M}$ を $IDtuple_t$ で暗号化する。まず、 $ID_1, \dots, ID_t$ を用いて $P_1, \dots, P_t$ を計算する。次に、メッセージごとに異なる乱数 $r \in \mathbb{Z}/q\mathbb{Z}$ を選択し、以下のように暗号文 C を計算する。

$$[rP_0, rP_2, \dots, rP_t, M \oplus H_2(\hat{e}(Q_0, P_1)^r)].$$

- **Decryption** 暗号文 $C = [U_0, U_2, \dots, U_t, M \oplus H_2(\hat{e}(Q_0, P_1)^r)]$ を復号する場合、以下を計算する。

$$M \oplus H_2(\hat{e}(Q_0, P_1)^r) \oplus H_2\left(\frac{\hat{e}(rP_0, S_t)}{\prod_{j=2}^t \hat{e}(Q_{j-1}, U_j)}\right) = M.$$

3. 提案手法

階層化されたグループ構成を有し、それぞれのグループが自律的に動作することが予想されるネットワークでは、各グループの判断でグループメンバに対し公開鍵／秘密鍵ペアを付与、または剥奪できることが望ましい。さらに、グループメンバ全員へのメッセージ送信も想定されるため、グループメンバ全員に公開鍵／秘密鍵ペアを新たに持たせ、暗号化マルチキャストができることが望ましい。本研究では、その一例として、HIBE を用いた鍵管理方式を検討する。既存の HIBE では鍵更新が想定されていないため、メンバの追加／脱退に伴うグループ鍵の更新方法が必要となる。本節では HIBE を用いた鍵事前配布方法、グループ鍵配布方法、及びグループ鍵更新方法を備えた鍵管理方式を提案する。

なお、本稿ではグループメンバが共有する秘密鍵と、個人で保持する秘密鍵を区別するため、それぞれをグループ秘密鍵、個人秘密鍵と呼ぶことにする。

3.1 概要

提案手法の概要を以下に述べる。

事前処理 以下の 3 つのステップで事前処理を行う。

まず第 1 ステップとして、グループの階層構造に沿って、HIBE における階層構造を構成する。すなわち、HIBE における階層構造をグループの階層構造と同型とし、上位から順に基地局 $\rightarrow$ グループリーダー $\rightarrow$ グループメンバとなるように、HIBE における階層構造の各頂点に、各ノードの ID を対応させる。

次に第 2 ステップとして、上記の階層構造に沿って以下の操作を行い、各ノードが保有する公開鍵／秘密鍵ペアを生成する。ここで、基地局は Root PKG と 1 段目の Lower-level PKG を兼ねることに注意されたい。

- **Root-Setup** 基地局は、2. 節と同様の操作を行い、システムパラメータ $params$ と、秘密情報であるマスタ鍵 s_0 を生成する。

- **Lower-level-Setup** 基地局 ID_1 及び各グループリーダー ID_t は、各自が保持するマスタ鍵 s_t を生成する。

- **Extraction** 基地局 ID_1 及び全てのグループリーダー ID_t は、子頂点の個人秘密鍵 S_{t+1} を作成する。さらに、自身がグループリーダーであるグループの ID として $ID_{group(t)}$ (グループ ID と呼ぶ) を用意し、これに対応するグループ秘密鍵 $S_{group(t)}$ を作成する。例えば、 $GROUP(t)$ をグループ固有のビット列、 Ver をバージョン情報とし、 $ID_{group(t)} = (GROUP(t) \parallel Ver)$ とすると、グループ ID が一意に定まり、また鍵更新のためのバージョン管理が可能となる。ここで、 $P_{group(t)} = H_1(IDtuple_{group(t)}) = H_1(ID_1, \dots, ID_t, ID_{group(t)})$ とする。

最後に第 3 ステップとして、基地局 ID_1 及び全てのグループリーダー ID_t は、子頂点に対して必要な秘密鍵を全て格納する。すなわち、ノードの個人秘密鍵 S_{t+1} と、そのノードが属する全てのグループのグループ秘密鍵、つまりそのノードの祖先全てが作成したグループ秘密鍵 $\{S_{group(1)}, \dots, S_{group(t)}\}$ を格納する。

グループ秘密鍵の更新手法 ネットワーク運用後にノードが脱退／追加された際には、グループ鍵の更新が必要である。まず、あるノード (ID_{leave} とする) が脱退したときのグループ鍵更新手順を述べる。

ID_{leave} の脱退を検知した基地局は、自身が管理しているグループ (全ノード及び基地局からなるグループ) のバージョン情報を更新し、新しいバージョン情報からグループ ID を生成して、これに対応するグループ秘密鍵を出力する。新しいバージョン情報とグループ秘密鍵とは、暗号化してグループメンバ全員へ向けて送信する。このとき、 ID_{leave} を含まないサブグループに対しては、サブグループの ID で暗号化し、そのグループメンバ全

員に送信する。 ID_{leave} を含むサブグループに対しては、サブグループのリーダーの ID で暗号化し、そのグループリーダーにのみ送信する。

ID_{leave} をメンバを含むサブグループのリーダーは、受信したバージョン情報とグループ秘密鍵に基づいて、自身が所属するグループの ID とグループ秘密鍵を更新する。また、基地局と同様にして、自身がリーダーであるサブグループのバージョン情報、グループ ID 及びグループ秘密鍵の更新を行う。こうして更新したバージョン情報及びグループ秘密鍵と、上位のグループリーダーから受け取った全てのバージョン情報及びグループ秘密鍵とを合わせて、これも基地局の場合と同様にしてグループメンバへ向けて送信する。

この作業を繰り返し、 ID_{leave} の直接の親ノードまでたどり着く。 ID_{leave} の親ノードはサブグループを持たず、グループメンバがその直下に存在するので、更新された全てのバージョン情報及びグループ秘密鍵を ID_{leave} 以外のノードに対し直接送信する。以上の操作によって、 ID_{leave} が保持していた全てのグループ秘密鍵を、各グループメンバ以外に知られることなく更新できる。

3.2 動作例

前節に述べた提案手法に沿って動作例を述べる。グループ階層構造は階層 4 段の完全二分木と仮定する。また、本節では、具体的な動作例を述べるため、各階層 t ごとの標記ではなく、各頂点 (l, m) ($1 \leq l \leq 4, 1 \leq m \leq 2l$) ごとの標記となっていることに注意されたい。

まず事前処理として、個人秘密鍵とグループ秘密鍵をそれぞれ生成し、格納する。例えば、基地局は自身のマスタ鍵 $s_0, s_{(1)}$ のみを保持するだけで、自身に宛てられた全ての暗号文を復号できる。また、末端メンバ $ID_{(4,5)}$ は、以下全ての個人／グループ秘密鍵を保持する (図 2)。

- 個人秘密鍵

$$(1) \quad s_0 P_{(1)} + s_{(1)} P_{(2,2)} + s_{(2,2)} P_{(3,3)} + s_{(3,3)} P_{(4,5)}.$$

- グループ秘密鍵

$$(1) \quad s_0 P_{(1)} + s_{(1)} P_{group(1)},$$

$$(2) \quad s_0 P_{(1)} + s_{(1)} P_{(2,2)} + s_{(2,2)} P_{group(2,2)},$$

$$(3) \quad s_0 P_{(1)} + s_{(1)} P_{(2,2)} + s_{(2,2)} P_{(3,3)} + s_{(3,3)} P_{group(3,3)}.$$

次に、ノード $ID_{(4,5)}$ が脱退したと仮定したときの鍵更新手順を述べる (図 3)。

$ID_{(4,5)}$ が脱退したとき、これを検知した基地局 $ID_{(1)}$ は、 $\mathbf{Ver}$ を更新し、例えば $ID_{group(1)} = (\text{GROUP}(1) \parallel 2)$ として $P'_{group(1)}, s_1 P'_{group(1)}$ を計算し、グループ秘密鍵 $s_0 P_{(1)} + s_{(1)} P'_{group(1)}$ を出力する。

次に、新しいグループ秘密鍵及びバージョン情報をメッセージ M とし、ランダムな値 r を選択し、 $ID_{(4,5)}$ を含まないサブグループ $ID_{group(2,1)}$ に対して送信する暗号

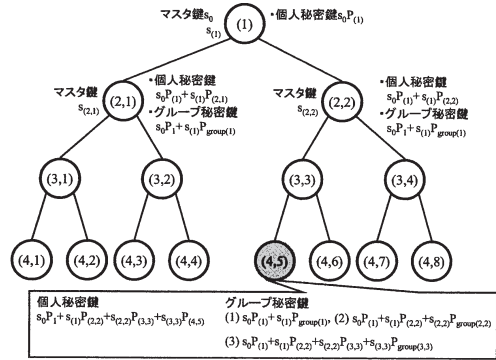


図 2 提案手法におけるグループ／個人鍵割り当て

文 $C_{group(2,1)}$ を以下のように計算する。

$$[r P_0, r P_{(2,1)}, r P_{group(2,1)}, M \oplus H_2(\hat{e}(s_0 P_0, P_{(1)})^r)].$$

ただし、 $M = (s_0 P_1 + s_1 P'_{group(1)}, \mathbf{Ver} = 2)$ である。 $group(2,1)$ に属するすべてのメンバ、かつそのメンバのみが暗号文 $C_{group(2,1)}$ を復号可能であるため、この暗号文からグループ秘密鍵を更新し、対応するバージョン情報を取得することができる。

対して、 $ID_{(4,5)}$ を含むサブグループのリーダー $ID_{(2,2)}$ に対しては、以下の暗号文を $ID_{(2,2)}$ だけに送信する。

$$C_{(2,2)} = [r P_0, r P_{(2,2)}, M \oplus H_2(\hat{e}(s_0 P_0, P_{(1)})^r)].$$

暗号文を受け取った $ID_{(2,2)}$ はさらに、自身のグループ ID のバージョン情報を例えば $ID_{group(2,2)} = (\text{GROUP}(2,2) \parallel 2)$ と更新し、グループ秘密鍵 $s_0 P_{(1)} + s_{(1)} P_{(2,2)} + s_{(2,2)} P'_{group(2,2)}$ を計算して、このグループ秘密鍵と $group(1)$ の新しいグループ秘密鍵を、基地局のときと同様の手順で子に暗号化して配る。以上の操作によって、 $ID_{(4,5)}$ が保持していた全てのグループ秘密鍵が、各グループメンバ以外に知られることなく更新可能となる。

ノードの追加に関しても同様の手順で鍵更新が可能である。すなわち、あるノードの追加に伴い、追加されたノードの属する全てのグループについて、ノードの脱退と同様にして各グループ秘密鍵を更新すればよい。

3.3 利点

提案手法の利点を以下に列挙する。

- 脱退したノードが保持する全てのグループ秘密鍵を、脱退したノードに知られることなく更新できる。これは、グループ公開鍵にバージョン情報を含めることで、陽に公開鍵のビット列に意味を持たせているためである。
- 全ノードに対しユニキャストで送信するよりも効率がよい。これについては 4. 節で詳しく述べる。
- 脱退したノードが保持するグループ秘密鍵のみを

の秘密情報から、以下のアルゴリズムを実行して復号を試みたとする。

$$M \oplus H_2(\hat{e}(s'_0, P_1)^r) \oplus H_2\left(\frac{\hat{e}(rP_0, S_{group(t)})}{\prod_{j=2}^t \hat{e}(Q_{j-1}, U_j)}\right).$$

上記の式を計算すると、 $H_2(\hat{e}(s'_0, P_1)^r)$ の項が消えず、メッセージ M を復号できないことがわかる。よって、システム全体のマスタ鍵 s_0 を変更することによっても、グループ鍵の更新が可能なのかわかる。ただし、システム全体のマスタ鍵を変更することで、全ての個人鍵及びグループ鍵を変更しなければならず、更新に膨大なコストがかかる。

(2) P_0 を変更した場合：公開情報 P_0 を P'_0 と変更した場合、 $Q_0 = s_0 P_0$ も $Q'_0 = s_0 P'_0$ と変更し、システム全体に公開する必要がある。 P'_0 と Q'_0 を用いてメッセージ M を暗号化した場合、脱退した ID_{leave} もまた P'_0 と Q'_0 を用いることができるため、メッセージ M を復号可能となる。よって、 P_0 を変更しても、適切なグループ鍵更新を行ったことにはならない。

(3) s_t を変更した場合：グループリーダーのマスタ鍵を変更した場合を考える。 s_t を s'_t と変更した場合、暗号文は以下のようになる。

$$[rP_0, rP_2, \dots, rP_{group(t)}, M \oplus H_2(\hat{e}(s_0, P_1)^r)].$$

このように、システム全体のパラメータ及び各公開鍵を変更しない限り、暗号文には変化がない。この暗号文を解くために、 ID_{leave} が公開情報と手持ちの秘密情報から以下のアルゴリズムを実行し、暗号文の復号を試みたとする。

$$M \oplus H_2(\hat{e}(s_0, P_1)^r) \oplus H_2\left(\frac{\hat{e}(rP_0, S_{group(t)})}{\prod_{j=2}^{t+1} \hat{e}(Q_{j-1}, U_j)}\right).$$

上記の式を解くと、メッセージ M を復号できる。これは、更新前のグループ秘密鍵 $S_{group(t)}$ 、更新前の公開情報 Q_t 、及び暗号文に付加される $rP_0, rP_2, \dots, rP_{group(t)}$ から $H_2(\hat{e}(s_0, P_1)^r)$ を計算できることが原因である。よって、 s_t を変更しても、適切なグループ鍵更新を行ったことにはならない。

(4) $P_{group(t)}$ を変更した場合：これはすなわち、グループ ID (公開鍵) を更新することを表す。提案手法で用いている方法であり、安全に鍵更新ができることを 3. 節で確認済みである。

以上の考察から、グループ鍵を少ない通信量、計算量で安全に更新するためには、各グループのグループ ID を変更する方法が最も望ましい。提案手法はこの方法に基づいており、少ない通信量、計算量で効率良く鍵更新が可能な手法であると考えられる。

6. ま と め

階層化されたグループ構成を有し、それぞれのグループが自律的に動作することが予想されるネットワークのための鍵管理方式を提案した。本研究では、HIBEを用いた鍵管理方式を検討し、基地局が Root PKG、各グループリーダーが Lower-level PKG の役割を担うような鍵管理方式の導入を検討した。さらに、各階層に対してグループ鍵を割り当て、グループメンバに対する暗号化マルチキャスト手法を提案し、グループメンバの追加/脱退を考慮したグループ鍵の更新方法を合わせて提案した。また、提案手法について、効率とセキュリティに関する評価を行った。さらに、HIBE における鍵更新手法について考察し、提案手法が最も効率よく安全な鍵更新手法であることを確認した。

今後の課題として、提案手法の実装、鍵更新にかかる時間の評価、提案手法にデジタル署名を加えたメッセージ送信方式の検討がある。

文 献

- [1] D. Boneh, X. boyen, and E. J. Goh, "Hierarchical Identity Based Encryption with Constant Size Ciphertext," Proc. of Eurocrypt'05, LNCS 3494, pp.440-456, Springer-Verlag (2005).
- [2] D. Boneh and M. Franklin, "Identity Based Encryption from the Weil Paring," Proc. of Crypto'01, LNCS 2139, pp.213-229, Springer-Verlag (2001).
- [3] C. Gentry and A. Silverberg, "Hierarchical ID-Based Cryptography," Proc. of Asiacrypt'02, LNCS 2501, pp.548-566, Springer-Verlag (2002).
- [4] C. K. Wong, M. Gouda, and S. S. Lam, "Secure Group Communications Using Key Graphs," Proc. of SIGCOMM'98, pp.68-79, ACM Press (1998).