M-51

ピアグループにおける鍵共有*

陳 志松† 山本雅基† 西野 晴久††† 神保雅一††
†(株)デンソークリエイト ††慶応義塾大学理工学部数理科学科 ††† (財) ソフトピアジャパン

1. はじめに

P2P(Peer to Peer)と呼ばれるネットワークシステムが注目を集めている。P2P ネットワークシステムとは、サーバとクライアントの両方の機能を備える複数の情報端末（ピア）が相互接続されることによって構成されるネットワークシステムである。P2P 通信を安全に行うために鍵共有(Key Contribution あるいは Key Agreement)プロトコルが提案されている[1][2][3]。グループに属する複数のピアが情報を秘密に交換するための共通鍵を、通常の通信回線を使って安全に生成するプロトコルである。

鍵共有プロトコルは、IKA(Initial Key Agreement)とAKA(Auxiliary Key Agreement)に分けることができる。IKA とは、グループを最初に形成するときに用いられるプロトコルである。一方、AKA とは、すでに存在しているグループに対して、ピアの追加や削除、鍵の更新といった操作を行うときに用いられるプロトコルである。

我々は、IKA プロトコルとして、「改良 IKA.1 プロトコル」を提案した[3]。本発表では、AKA に適している修正「改良 IKA.1 プロトコル」（以下、Waterfall プロトコルと呼ぶ）と、新しい AKA プロトコルを提案する。

2. Waterfall プロトコル

「改良 IKA.1 プロトコル」は、従来の IKA プロトコルである IKA.1[2]と比べ、ピアごとの計算量が均等でない問題を解消し、全体のべき乗計算量も $O(n^2)$ から $O(n\log n)$に減少した。また、ブロードキャストが不要のため、幅広い用途に用いることができると考えている。Waterfall プロトコルは、「改良 IKA.1 プロトコル」を AKA に適用できるように修正を加えた。その概要を以下に説明する。

本発表で使う記法を以下でまとめて示す：

n	:	グループに属するピアの数
i,j,m	:	ピアの順番を示す添え字
P	:	ピア、Pi は i 番目のピア
q	:	素数
α	:	GF(q)の原始元
N	:	秘密鍵、Ni は i 番目のピアの秘密鍵
K,K'	:	共通鍵

Waterfall プロトコルは、下記の２つのフェーズから構成される：

フェーズ１：ピア Pi が $\alpha^{\prod_{j=1}^{n} N_j/N_i}$ を得る。この情報より、ピアが秘密鍵 N_i を用いて共有鍵 K を得ることができる。

フェーズ２：ピア全員が $\{\alpha^{\prod_{j=1}^{n} N_j/N_i} ; i=1,\cdots n\}$ を得る。この情報は後述の AKA プロトコルに用いられる。

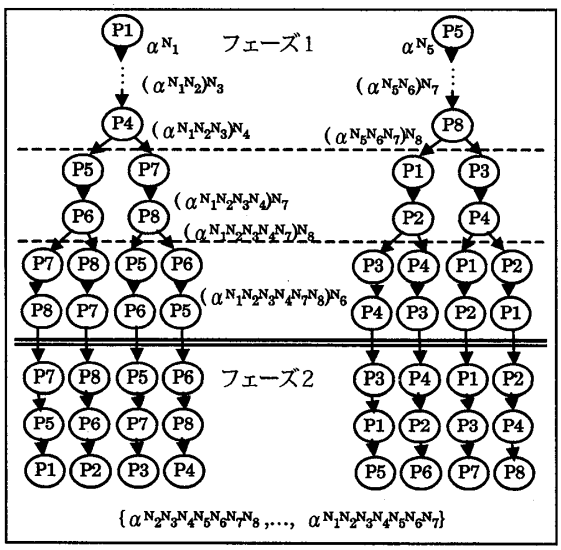

図１ Waterfall プロトコル（$n=2^3$ の場合）

$n=2^3$ の場合の Waterfall プロトコルを図１で示す。「改良 IKA.1 プロトコル」はフェーズ１にあたる。フェーズ２では、(P1,P2),...,(P7,P8)がペアになって各自が持っている情報を交換し、さらに

---

*Key Agreement in Peer Group.

†Zhisong Chen, †Masaki Yamamoto, †††Haruhisa Nishino, ††Masakazu Jimbo
†Denso Create Inc., ††Faculty of Science and Technology, Keio University, †††Softpia Japan

(P1,P3),(P2,P4),...(P5,P7),(P6,P8)がペアになって情報交換を行い、最後にピア全員が $\{\alpha^{\prod_{j=1}^{n} N_j/N_i} ; i=1,\cdots n\}$ を持つことになる。フェーズ2は計算なしで、ユニーキャストのみで実現できる。

3. 提案するAKAプロトコル

IKAプロトコルにより、すでに共有鍵を持つピアグループにおいて、新たなピアの追加または削除するたびに、元の鍵が使えなくなり、新たに鍵を生成する必要が生じる。また、安全上の理由で定期的に鍵を更新したいことも考えられる。以下はピアの追加、削除、鍵更新のプロトコルを提案する。

3.1 ピアの追加

3.1.1 単独ピアの追加

以下、単独ピア追加の手順を示す：

(a) 元のグループのピア P_i が秘密鍵 $\tilde{N}$ を選び、追加するピア P'_1 に $\{\alpha^{(\prod_{j=1}^{n} N_j/N_i)\tilde{N}} ; i=1,\cdots n\}$ と $K^{\tilde{N}}$ (K：元のグループの共有鍵) をユニーキャストする。

(b) P'_1 が $\{\alpha^{(\prod_{j=1}^{n} N_j/N_i)\tilde{N}} ; i=1,\cdots n\}$ に自分の秘密鍵 N'_1 でべき乗し、$K^{\tilde{N}}$ と一緒に元のグループにブロードキャストする。

(c) ピア全員が秘密鍵を用いて新しい共有鍵 $K'=K^{\tilde{N}N'_1}$ を得る。

3.1.2 複数ピアの追加

以下、複数ピアの追加手順を示す：

(a) 元のグループのピア P_i と追加ピアの P'_1,P'_2 の間でBurmester-Desmedtプロトコルを用いて共有鍵K*を生成する。

(b) P_i が K^{K*} を P'_1,P'_2 にマルチキャストし、P'_1,P'_2 がK*を用いてKを得る。

(c) 追加ピアが P'_1,P'_2 を先頭とし、Waterfallプロトコルのフェーズ1を適用し、新しい共有鍵 $K'=K^{N_1N'_1\cdots N'_m}$ を生成する

(d) P'_1 が K^{K*} を Pi にユニーキャストし、P_i がK*を用いてK'を得る。

(e) P_i が K^K を元のグループにブロードキャストし、元のグループのピアがKを用いてK'を得る。

最後にWaterfallプロトコルのフェーズ2を適用する必要があるが、紙面の都合で詳細は省略する。Waterfallプロトコルを用いることで、IKA時と同様に追加ピアごとの計算量が均等になり、追加ピア全体のべき乗計算量も従来の方法より減少した。

3.2 ピアの削除

以下、ピア削除の手順を示す：

(a) 元のグループのピア P_i が秘密鍵 $\tilde{N}$ を選び、$\{\alpha^{\prod_{j=1}^{n} N_j/N_i} ; i=1,\cdots n\}$ から削除しようとするピアの情報を除いた後、$\tilde{N}$ べき乗してグループにブロードキャストする。

(b) ピア全員が自分の秘密鍵を用いて、新しい共有鍵 $K'=K^{\tilde{N}}$ を得る。

3.3 共有鍵の更新

鍵更新はピア削除の特別ケース(0 ピアの削除)として考えられるので、ピア削除のプロトコルを利用できる。

4. 今後の課題

今回はIKAプロトコルとAKAプロトコルについて検討を行い、Waterfallプロトコルおよびピアの追加、削除する場合の新しいAKAプロトコルを提案した。提案プロトコルにより、ピアグループに対して実用的な鍵共有が可能となった。今後は認証付きのグループ鍵生成プロトコルの検討を行い、これらのプロトコルを適用したアプリケーションを開発したい。

参考文献

[1] M. Burmester and Y. Desmedt, "A secure and effcient conference key distribution system", in Advances in Cryptology EUROCRYPT '94, 1994.

[2] M. Steiner, G. Tsudik, and M. Waidner, "Key agreement in dynamic peer groups", IEEE Transactions on Parallel and Distributed Systems, 11(08), August 2000.

[3] 陳,山本,矢尻,石田,神保, "グループ鍵生成プロトコルに関する一考察", 情報処理学会第64回全国大会, 2002