社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2003－DPS－113 (7)
2003／6／20

安全で閉じたP2Pネットワークの構成方式の提案

古志智也† 齋藤孝道††

P2Pネットワークモデルにおいて，通信を暗号化するためのグループ鍵（共通鍵）をどのようにして安全に共有するかが問題の一つとしてある．多くのP2Pネットワークモデルの実現方式では，公開鍵の取得をPKI（Public Key Infrastructure）などの利用を前提にしており，さらに，IPアドレスの取得にはDNS（Domain Name System）を利用している．しかし，これらCA（Certificate Authority）やDNSなどの利用，もしくは，処理を一部委託するといった形態は，外部サーバの非依存を特徴とするP2Pの本来の趣旨になじまない．したがって，本論文では，「知り合い」という信頼モデルを導入することにより，外部サーバに依存しないノードだけで構成する安全で閉じたネットワークの構成方式の提案を行い，その安全性に関する議論を行なう．

A Secure Grouping for Pure P2P Network

TOMOYA KOSHI† and TAKAMICHI SAITO††

The P2P network model has a problem how to distribute a group key to proper members. Many P2P network models use server systems, such as PKI's CA and DNS. However, these methods are not applied to P2P feature : independent of external servers. Therefore, in this paper, we propose a secure grouping method, which is independent of external servers, and we discuss its security.

1. はじめに

現在のインターネットにおけるネットワークの通信形態の主流は，クライアント・サーバモデルである．しかし，近年，クライアント側の計算能力の向上やネットワークの高速化といった理由により，サーバを必要とせずノード同士が直接接続してデータの通信を行うネットワーク通信形態であるP2P（Peer-to-Peer）ネットワークモデルが注目されている．P2Pネットワークモデルのアプリケーションが持つべき重要な特徴は，「P2Pネットワークのグループ（以降，グループ）の構成」，「（ユーザが欲する）コンテンツ情報の検索」，「ノード同士がコンテンツを直接交換」の三つである．P2Pネットワークモデルには，様々な技術的課題が存在する．例えば，効率的なコンテンツの検索方法[1]，ノードの参入や脱退の方法[2][3]，グループ内の鍵管理方法[4]などが挙げられる．本論文では，P2Pネットワークモデルの課題であるノードの参入や脱退の方法とグループ内で使用する共通鍵の分配，すなわちグループ構成の方式を提案する．

P2Pネットワークモデルは2つに大別できる．一方は，Hybrid P2Pモデルと呼ばれるものである．このモデルでは，「P2Pネットワークの構成」，「（ユーザが欲する）コンテンツの検索」をサーバが管理するため，グループに存在する全てのノードのIPアドレスや通報を暗号化する鍵と鍵に対応するノードのIDという情報をサーバが一元的に保存する．そのため，ユーザがP2Pアプリケーションを利用する際には，各ユーザは必ずサーバと接続し，グループの既存メンバーや共有したいコンテンツを持つノードのIPアドレスを取得する．そして，データ通信は各ノード同士が直接行う．一般的なHybrid P2Pモデルでは，上述のようにグループを管理するサーバがいるため，新規ノードがグループへ参入する場合，新規ノードはサーバからグループ内で使用する共通鍵を取得できる．このモデルでは，グループ内の全メンバーが特定のサーバを信頼しているため，安全なグループを構成できる．

P2Pネットワークモデルのもう一方は，ノードだけでグループを構成するPure P2Pモデルと呼ばれるものである．しかし，一般的なサーバを利用しないP2Pネットワークモデルの多くの実現方式では，公開鍵の取得にPKI（Public Key Infrastructure）[5]の利用を

† 東京工科大学院 工学研究科 システム電子工学専攻
Toyokyo graduate school of Technology
†† 東京工科大学 コンピュータサイエンス学部
Tokyo University of Technology

前提にしている．このP2Pネットワークモデルにおいて新規ノードがグループへ参入する場合，各々の公開鍵は信頼できるCA（Certificate Authority）が保証するため，新規ノードと既存メンバーは互いの公開鍵を取得できるので，安全なグループを構成できる．

しかし，これらの方式は，以下のような考慮すべき点がある．前者の方式の問題点は，「信頼の集中」，「機能の集中」である．特定のノードを信頼することにより，配布する共通鍵を確実に保証できるが，DoS（Denial of Service）攻撃によって，管理サーバがネットワークの通信障害になる可能性があり，P2Pネットワークモデルの利点であるフォールトトレラントという性質が損なわれる可能性がある．また，P2Pの本来の趣旨である各ノードは等価な関係であるという概念となじまない．後者の方式の問題点もほぼ同様に，外部サーバに処理を一部委託することである．DoS攻撃によって，CAがネットワークの通信障害になる可能性があることが考えれれる．このことは，ドメイン名からIPアドレスを取得する際のDNS（Domain Name System）の利用においても同様である．

そこで，本論文では，「知り合い」という信頼モデルを導入することにより，CAやDNSの役割を各ノードに分散することで，外部サーバに依存しないノードだけで構成する安全で閉じたネットワークの実現方式の提案を行い，その安全性に関する議論を行なう．

2. 準　　備

ここでは，本論文を通して共通に用いる記法と用語を簡潔にまとめる．

2.1 用　　語

以下のように用語を定める．

メンバー：あるグループ識別子（後述）を含むメンバーリスト（後述）を持つノード．

グループ：メンバーから構成されるP2Pネットワーク．

参入者：知り合い（後述）のいるグループに新規参入するノード．

知り合い：参入者がグループに参入する以前に，参入者のIPアドレスと公開鍵の情報をもつ既存メンバー．提案システムでは，知り合いがCAの役割を担う．

交渉者：参入者がグループへ参入する際に接続する，知り合いではないもう一方のメンバー．

脱退者：グループから脱退をするメンバー．

セキュア通信路：2者間で秘密に共有した共通鍵による安全な通信路．共通鍵は，2者間で互いに保持する公開鍵を用いて相互認証をし，生成される．また，共通鍵は定期的に更新される．

固有情報：あるメンバーに対応する識別子（ホスト名），IPアドレス，公開鍵の三つ．

メンバーリスト：各メンバーが保有するグループ内に存在する全メンバーの固有情報から構成されるリスト．これは，現在のグループの状態を把握するために使用される．あるメンバーを基準として，時計回でリング型に構成する（表1参照）．ただし，表中の A，B，Cはグループを構成するメンバーを表している．

表1　メンバーリストの構成の例

Aのホスト名＿AのIPアドレス＿Aの公開鍵
Bのホスト名＿BのIPアドレス＿Bの公開鍵
Cのホスト名＿CのIPアドレス＿Cの公開鍵

グループ識別子：グループを区別するために使用する識別子．グループ識別子は，メンバーリストをハッシュした値とする．

一時保管リスト：JOINプロトコル（後述）やLEAVEプロトコル（後述）の際に使用する参入者や脱退者の固有情報を一時的に保存する際に作成されるリストを指す．JOINおよびLEAVEプロトコルが途中で失敗した際に使用することを考慮している．構成はメンバーリストの構成例と同様である．

2.2 記　　法

本論文中で利用する表記を示す．

ノードの表記：参入者を N，脱退者を L，知り合いを A，交渉者を B，参入者および脱退者と直接接続していないメンバーを Cとする．また，任意のノードを Xとする．

固有情報とリストの表記：メンバー Xのホスト名を ID_X，IPアドレスを IP_X，公開鍵を P_Xとする．このときの Xの固有情報は，(ID_X, IP_X, P_X)である．また，メンバーリストを ML，グループ識別子を GID，一時保管リストを TL_Xと表記する．

通報の表記：通報の流れを $A \rightarrow B : MSG = \{\{Header\},\{Info\}\}$のように表す．これは，メンバー Aがメンバー Bに MSGという通報をセキュア通信路で送信を表し，MSGが $\{Header\}$と $\{Info\}$から構成されることを示している．具体的に $\{Header\}$は，通報の送信者と受信者のホスト名，グループ識別子，通報のタイプで構成され，通報のタイプはTYPE_で始まる文字列である．$\{Info\}$とは，メンバーリストや参入者，脱退者な

どが含まれの固有情報である．また，$\{Info\}_{P_X^{-1}}$ でメンバー X が $Info$ を署名したことを表す．また，通報の表記を説明に応じて単に $A \rightarrow B: MSG$ や $MSG = \{\{Header\},\{Info\}\}$ と表記する場合もある．

3. 提案システム

提案システムでは，リング型にメンバー同士を接続する．また，グループ内の各メンバーは，最終的にメンバーリストによって，自身を含めた全てのメンバーの固有情報を保持する．参入者は，グループ内にいる知り合いと JOIN プロトコルで新たにグループに加わる．また，グループからの脱退には LEAVE プロトコルを利用する．

3.1 提案システムにおける利用の前提

提案システムにおける利用の前提は以下の通りである：

(1) 参入者がグループに参入する場合に，グループ内には既に知り合いが存在する．
(2) 提案システムで使用される全ての通報は，セキュア通信路により暗号化されている．
(3) 各メンバーは自身の知り合いが保証したメンバーを常に信頼する．
(4) 知り合いは，自身が所属するグループのグループ識別子を参入者に公開する．
(5) グループ内のメンバー同士で行なうコンテンツの交換は，メンバーリストを利用したセキュア通信路で行なう．

3.2 通報の種類

ここでは，提案システムで用いられる通報について説明する．

3.2.1 JOIN プロトコルで使用される通報

JOIN プロトコルで使用する通報は 9 種あり，以下で説明する．

- $N \rightarrow A$: $JOIN\_MSG = \{\{ID_N, ID_A, \texttt{TYPE\_JOIN}, GID\}, \{ID_N, P_N, IP_N\}\}$
  グループへの新規の参入要求．
- $A \rightarrow B$: $NOTICE\_MSG = \{\{ID_A, ID_B, \texttt{TYPE\_NOTICE}, GID\}, \{ID_N, P_N, IP_N\}_{P_A^{-1}}\}$
  参入者発生通知．
- $B \rightarrow A$: $ACK\_MSG = \{\{ID_B, ID_A, \texttt{TYPE\_ACK}, GID\}, \{\}\}$
  N の参入許可通知．
- $A \rightarrow N$: $WELCOME\_MSG = \{\{ID_A, ID_N, \texttt{TYPE\_WELCOME}, GID\}, \{ID_B, P_B, IP_B\}_{P_A^{-1}}\}$
  B への接続許可通知．
- $N \rightarrow B$: $HELLO\_MSG = \{\{ID_N, ID_B, \texttt{TYPE\_HELLO}, GID\}, \{\}\}$
  N の接続完了通知．
- $B \rightarrow A$: $NCONNECT\_MSG = \{\{ID_B, ID_A, \texttt{TYPE\_CONNECT}, GID\}, \{\}\}$
  N の接続完了通知．
- $A \rightarrow N$: $NEWMEMBER\_MSG = \{\{ID_A, ID_N, \texttt{TYPE\_NEWMEMBER}, GID\}, \{ML\}_{P_A^{-1}}\}$
  N へ ML の配布．
- $UPLISTS\_MSG = \{\{\texttt{TYPE\_UPLISTS}, GID\}, \{ID_N, P_N, IP_N\}_{P_A^{-1}}\}$
  A から N 以外の全メンバーに送信される，メンバーリスト更新通知（メンバーの追加）．
- $B \rightarrow A$: $FINCONNECT\_MSG = \{\{ID_B, ID_A, \texttt{TYPE\_FINCONNECT}, GID\}, \{\}\}$
  B, A 間の接続終了要求．

3.2.2 LEAVE プロトコルで使用される通報

LEAVE プロトコルで使用する通報は 5 種あり，以下で説明する．

- $L \rightarrow A$: $LEAVE\_MSG = \{\{ID_L, ID_A, \texttt{TYPE\_LEAVE}, GID\}, \{\}\}$
  グループからの脱退要求．
- $A \rightarrow L$: $FINCONNECT\_MSG = \{\{ID_A, ID_L, \texttt{TYPE\_FINCONNECT}, GID\}, \{\}\}$
  A, L 間の接続終了要求．
- $A \rightarrow B$:$REQR\_MSG = \{\{ID_A, ID_B, \texttt{TYPE\_REQR}, GID\}, ID_L\}$
  B に対し，L との接続終了要求．
- $B \rightarrow A$: $CONNECTR\_MSG = \{\{ID_B, ID_A, \texttt{TYPE\_CONNECTR}, GID'\}, \{\}\}$
  A に対し，B の接続完了通知．
- $DELLIST\_MSG = \{\{\texttt{TYPE\_DELLIST}, GID\}, ID_L\}$
  A から L 以外の全メンバーに送信される，メンバーリスト更新通知（メンバーの削除）．

3.3 JOIN プロトコル

JOIN プロトコルは，参入者 N があるグループへ参加するときに用いるプロトコルである．JOIN プロトコルの通報の流れは以下のとおりである（図 1）．初期状態として，A は ML と N の固有情報を，B と C は ML を，N は A の固有情報を保持する．また，N は A の所属する GID の一覧を JOIN プロトコル以前に何らかの方法で獲得し，参入するグループを指定する．この GID は公開情報なので，第三者に漏洩しても問題ない．また，JOIN プロトコル終了後のグループは，図 2 のように構成される．

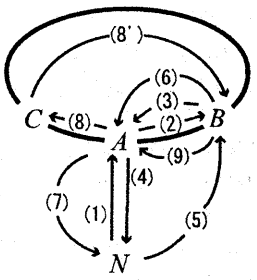

図 1　参入フェーズ時における通報の流れ

通報の流れ（各数字は図 1 に対応）:

(1)　$N \rightarrow A$　:　{*JOIN_MSG*}
(2)　$A \rightarrow B$　:　{*NOTICE_MSG*}
(3)　$B \rightarrow A$　:　{*ACK_MSG*}
(4)　$A \rightarrow N$　:　{*WELCOME_MSG*}
(5)　$N \rightarrow B$　:　{*HELLO_MSG*}
(6)　$B \rightarrow A$　:　{*NCONNECT_MSG*}
(7)　$A \rightarrow N$　:　{*NEWMEMBER_MSG*}
(8)　$A \rightarrow C$　:　{*UPLISTS_MSG*}
(8')　$C \rightarrow B$　:　{*UPLISTS_MSG*}
(9)　$B \rightarrow A$　:　{*FINCONNECT_MSG*}

処理の説明:

(1) Nは Aに対して通報 *JOIN_MSG*を送信する．その際，参入する *GID*を指定する．
(2) A は，その通報から Nの固有情報 (ID_N, P_N, IP_N)を取り出し，自身の TL_Aに保存する．そして，Bに通報 *NOTICE_MSG*を送信する．
(3) Bは，その通報から Nの固有情報 (ID_N, P_N, IP_N)を取り出し，自身の TL_Bに保存する．その後，Aへ通報 *ACK_MSG*を送信する．
(4) A は，自身の *ML* から B の固有情報 (ID_B, P_B, IP_B)を取り出す．その後，Nに通報 *WELCOME_MSG*を送信する．
(5) Nは，その通報から Bの固有情報(ID_B, P_B, IP_B)を取り出し，Bへ通報 *HELLO_MSG*を送信する．
(6) また，Bは Aに通報 *NCONNECT_MSG*を送信し，Aとの接続を終了する．
(7) Aは，(2) で保存した TL_B から Nの固有情報 (ID_N, P_N, IP_N)を取り出し，自身の *ML*に追記することで *ML'* に更新する．そして，自身の所属する *GID*を *ML'* を使用して *GID'*に更新する．その後，Nに通報 *NEWMEMBER_MSG*を送信する．
(8) また，Aは Cに通報 *UPLISTS_MSG*を送信する．
(8') Cは，通報から Nの固有情報を取り出し，自身の *ML*に追記することで *ML'* に更新する．そして，自身の所属する *GID* を *GID'*に変更する．その後，隣接するメンバーである Bへ通報 *UPLISTS_MSG*を送信する．
(9) Bは，通報から Nの固有情報を取り出し，自身の *ML*に追記することで *ML'* に更新する．そして，自身の所属する *GID*を *GID'*に変更する．その後，Aへ通報 *FINCONNECT_MSG*を送信する．この時点でグループの再構築が終了し，Nはグループのメンバーとなる．

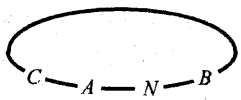

図 2　参入フェーズ終了後のメンバーの位置関係

3.4 LEAVE プロトコル

LEAVE プロトコルの通報の流れは以下の通りである（図 3 参照）．この例では脱退者の両隣が知り合いと交渉者になっているが，隣接するメンバーの参入や脱退によって，参入時とメンバーが変更している場合でも問題なくグループから脱退できる．

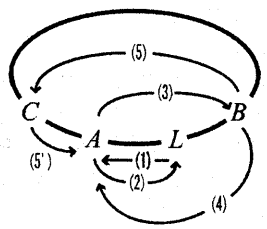

図 3　LEAVE プロトコルにおける通報の流れ

通報の流れ（各数字は図 3 に対応）:

(1)　$L \rightarrow A$　:　{*LEAVE_MSG*}
(2)　$A \rightarrow L$　:　{*FINCONNECT_MSG*}
(3)　$A \rightarrow B$　:　{*REQ_MSG*}
(4)　$B \rightarrow A$　:　{*CONECTR_MSG*}
(5)　$B \rightarrow C$　:　{*DELLISTS_MSG*}
(5')　$C \rightarrow A$　:　{*DELLISTS_MSG*}

処理の説明：

(1) L は A に通報 *LEAVE_MSG* を送信する．

(2) A は，自身の ML から L の固有情報 (ID_L, P_L, IP_L) を取り出して自身の TL_A に保存し，自身の ML を L の固有情報を削除することで ML' に更新する．そして，自身の所属する *GID* を *GID'* に変更する．その後，L に通報 *FINCONNECT_MSG* を送信する．

(3) また，A は，B に通報 *REQ_MSG* を送信する．

(4) B は，その通報から L の固有情報 (ID_L, P_L, IP_L) を取り出して自身の TL_B に保存し，自身の ML を L の固有情報を削除することで ML' に更新する．そして，自身の所属する *GID* を *GID'* に変更する．そして，A に対して通報 *CONNECTR_MSG* を送信する．

(5) その後，B は，C に通報 *DELLISTS_MSG* を送信する．

(5') C は，メンバーリストの更新が終了しているか判断する．C のメンバーリストは更新されていないため，自身の ML を L の固有情報を削除することで ML' に更新する．そして，自身の所属する *GID* を *GID'* に変更する．その後，隣接するメンバーである A へ通報 *DELLISTS_MSG* を送信する．その通報を受け取った A は自身の ML を更新する必要が無いので，L の脱退が終了する．

3.5 実　　装

提案システムは Java 言語（Java 2 SDK, Standard-Edition1.4.1）を使用して実装を行った．また，参入者の認証と認証後の暗号化通路には SSL（Secure Socket Layer）[7] を利用した．SSL とは通信相手の認証と暗号化の機能を提供するプロトコルである．SSL の認証にはサーバ認証モードと相互認証モードがあり，公開鍵を用いて認証する．また，SSL の通報の暗号化は共通鍵暗号化方式を用いる．今回の実装では SSL の相互認証を利用した．また，Java で実装された SSL の API として JSSE（Java Secure Socket Extension）を使用し，共通鍵暗号化方式には Triple-DES，SSL の鍵の共有には RSA 方式を利用している．

4. 議　　論

ここでは，提案システムと一般的な P2P モデルとの比較を，信頼モデル，ネットワーク構成，鍵の管理方法によるストレージと通信のコストという観点から行う．

4.1 信頼モデルによる比較

提案システムは，データの暗号化に 2 者間で秘密に共有する共通鍵を使用するが，2 者の公開鍵を用いて相互認証と共通鍵の交換を行なうため，安全性のために，ID と正しく対応した公開鍵を互いに共有できればよい．多くの P2P ネットワークモデルの実現方式では，他者の公開鍵の取得を PKI などの利用を前提にしている．そのため，各メンバーは信頼できる CA が各公開鍵と ID の対応を保証することにより，グループに参加するノードの公開鍵が正当なものであると判断できる（図 4）．

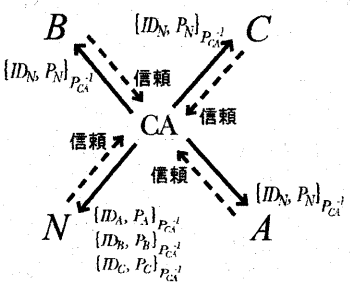

図 4　PKI における CA の信頼関係

これに対し，提案システムでは PGP（Pretty Good Privacy）[6] の信頼モデルのように，全てのメンバーが CA として他のメンバーに対し参入者を保証できるため，信頼の集中や機能の集中を防ぐことができる．また，公開鍵は信頼できるメンバー（知り合い）が署名し配布するため，偽のメンバーが公開鍵を配布しても否認を検出することができる．提案システムでは，図 5（A が CA の役割の時）のようにグループ内の信頼関係が成り立ち，PKI と異なる信頼モデルでグループを構成できる．

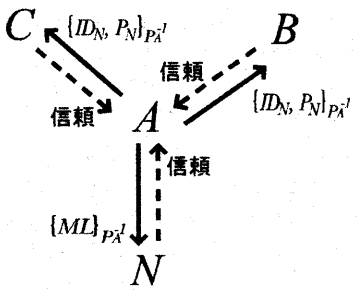

図 5　提案システムの信頼関係

4.2 ネットワーク構成による比較

提案システムでは，各メンバーが自身を含む全てのメンバーの IP アドレスや公開鍵を保持するため，外部の CA や DNS を必要としない．したがって，表 2

に示すように，提案システムではCAやDNSなどの外部サーバに依存しない安全で閉じたネットワークを形成することができる．

表2 ネットワーク構成による比較

	従来のモデル	提案システム
公開鍵	PKIにおけるCAが必要	グルーピングの際に獲得するため，PKIにおけるCAを利用する必要はない
信頼関係	信頼が集中	信頼が分散
位置情報	DNSを利用している	グルーピングの際に獲得するため，DNSを利用する必要はない

4.3 鍵の管理方法によるストレージとグルーピング時の通信のコストの比較

Hybrid P2Pモデルでは，サーバが鍵やIPアドレスを保持していたため，その他のノードには，ストレージコストはあまり要求されない．提案システムでは，グループ内にいる全ノードの公開鍵とIPアドレスを保持しなくてはならないため，Hybrid P2Pモデルに比べ，各ノードにストレージコストが要求される．例えば，公開鍵が1024bit，IPアドレス，ホスト名をそれぞれ32bit，メンバー数を10,000名と仮定した場合，提案システムは各メンバーに約1.36Mbyteのストレージコストを強いる．しかし，グループメンバーの更新によるリストの分配で，グループ内の全てのメンバーのリストを送信するのではなく，新規参入者もしくは脱退者だけの情報を送信したため，現在の計算機の能力やネットワーク環境からすればさほど問題がないといえるだろう．

5. 課　　題

提案システムの課題として，グループ構成時の通信コストの削減，信頼モデルの議論，スケーラビリティの考慮の3つがある．現在の提案システムは，グループ内のメンバー同士で行なうJOINプロトコルとLEAVEプロトコルで使用する通報に2者間で共有している共通鍵を使用している．この方法は，グループ鍵を使用する方法と異なり，鍵の同期をする必要がない．しかし，グループメンバーが変更した際に，全てのメンバーが新しいメンバーリストを作成するまでに複数の暗号化処理が発生し，結果としてグループ構成時の通信コストが増すといった問題がある．また，グループ内に悪意を持つメンバーが存在した場合の議論も必要である．これらを踏まえグループメンバーのスケーラビリティも考慮すべき点である．

6. ま と め

多くのP2Pネットワークモデルの実現方式では，ある一つのノードが鍵と鍵に対応するノードのIDを管理したり，他者の公開鍵の取得にPKIの利用を前提にしている．しかし，これらの方式には，「信頼の集中」，「機能の集中」がおこり，DoS攻撃によって管理者や外部サーバがネットワークの通信障害になる可能性がある．本論文では，「知り合い」という信頼モデルを導入することにより，CAの役割を各ノードに分散し，外部サーバに依存しないノードだけで構成する安全で閉じたネットワークの実現方式の提案を行い，その安全性に関する議論を行なった．

参 考 文 献

1) Ion Stoica, Robert Morris, David Liben Nowell, David R. Karger, M. Frans Kaashoek, Frank Dabek, Hari Balakrishnan：Chord A Scalable Peer-to-peer Lookup Protocol for Internet Applications, the IEEE/ACM Transactions on Networking.
2) Li Gone：Enclaves：Enabling Secure Collaboration over the Internet, JavaSoft, California.
3) Bruno Dutertre, Valentin Crettaz, Victoria Stavridou：Intrusion-Tolerant Enclaves, System Design Labortory, May 2002.
4) Adrian Perrig, Dawn Song, J.D.Tygar：ELK, a New Protocol for Efficient Large-Group Key Distribution.
5) R. Housley, W.Polk, W. Ford, D. Solo, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, RFC 3280, April 2002.
6) S. Garfinkel：PGP Pretty Good Privacy, O'Reilly & Associates, Inc, 1995. 邦訳：山本和彦監訳『PGP 暗号メールと電子署名』, オライリージャパン（1996）.
7) T.Dierks, C.Allen, The TLS Protocol Version 1.0, RFC 2246, January 1999.