様式 C-19

科学研究費補助金研究成果報告書

平成２３年６月１日現在

機関番号：13903
研究種目：若手研究（B）
研究期間：2009～2010
課題番号：21700072
研究課題名（和文）セキュアグループ通信における多重帰属の鍵管理を容易化する秘密分散方式に関する研究
研究課題名（英文）A secret sharing scheme for multiple associating group key management in secure group communications
研究代表者
白石　善明（SHIRAISHI YOSHIAKI）
名古屋工業大学・工学研究科・准教授
研究者番号：70351567

研究成果の概要（和文）：
本研究では、秘密分散技術により鍵は分割され、各メンバとサーバが分割された異なる鍵（メンバ部分秘密鍵、サーバ部分秘密鍵）を所持し、サーバ部分秘密鍵とある一つのメンバ部分秘密鍵によりグループ公開鍵で暗号化されたメッセージを復号できるようなグループ通信システムを想定する。あるメンバが複数のグループに多重帰属しているとき、そのメンバは復号に必要なメンバ部分秘密鍵を複数所有する。多重帰属のために複数の鍵を利用者が管理することは負担であることから、本研究では単一の部分秘密鍵で多重帰属できるセキュアグループ通信方式の開発をした。

研究成果の概要（英文）：
Shared file is encrypted and stored in server then group member shares its decryption key as an implementation of secure group communication. User has the same number of group decryption key as multiple associating groups in the communication. This research proposes a group file sharing protocol and group management protocols using ElGamal threshold cryptosystem and secret sharing scheme, which can decrypt file decryption key shared in any associated groups by one group key.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
２００９年度	1,300,000	390,000	1,690,000
２０１０年度	1,100,000	330,000	1,430,000
年度			
年度			
年度			
総　計	2,400,000	720,000	3,120,000

研究分野：総合領域
科研費の分科・細目：情報学・計算機システム・ネットワーク
キーワード：グループ通信、鍵管理、多重帰属、秘密分散

１．研究開始当初の背景

グループ内のメンバでセキュアに通信を行うとき、一般に、メンバはグループ通信のための鍵を所持する。メンバが複数のグループに多重帰属する場合、メンバはグループごとに異なる鍵を所有しなければならない。

多重帰属環境の鍵管理に関する課題をソフトウェアベースのシングルサインオンのようなシステムによるのではなく、秘密分散技術に基づいて理論的に解決するアプローチは見られない。本研究では、メンバの所有するただ一つの鍵により多重帰属しているグループのそれぞれで通信ができる方式の開発を目指した。

２．研究の目的

グループ内のメンバでセキュアに通信を行うとき、一般に、メンバはグループ通信のための鍵を所持する。本研究では、秘密分散技術により鍵は分割され、各メンバとサーバが分割された異なる鍵（メンバ部分秘密鍵、サーバ部分秘密鍵）を所持し、サーバ部分秘密鍵とある一つのメンバ部分秘密鍵によりグループ公開鍵で暗号化されたメッセージを復号できるようなグループ通信システムを想定する。

あるメンバが複数のグループに多重帰属しているとき、そのメンバは復号に必要なメンバ部分秘密鍵を複数所有する。多重帰属のために複数の鍵を利用者が管理することは負担であり、本研究では単一の部分秘密鍵で多重帰属できるセキュアグループ通信方式の要素技術の開発を行う。

３．研究の方法

本研究ではセキュアグループ通信の一例として、グループファイル共有システムの実装を目標にして方式設計を行う。

目標とするグループファイル共有システムは、ファイル復号鍵を取り出すメンバ部分復号鍵はメンバごとに異なり、また、多重帰属するメンバはいずれのグループでも同一のメンバ部分秘密鍵で復号できるように単一化されているという特徴を持つ。

閾値暗号技術、分散秘密情報の再分散技術、分散秘密情報のリフレッシュ技術を組み合わせて、グループの初期構築と動的な構成変更ができ、かつ、多重帰属するメンバの部分秘密鍵を単一化する機能を有したサーバとメンバの段階的復号を実現する秘密分散方式を検討する。

４．研究成果

グループ秘密鍵をサーバ部分復号鍵とメンバ部分復号鍵に(2,2)閾値秘密分散法を繰り返し適用することで分割する。分散させた一方をメンバに、他方をファイル管理サーバに割り当てる。すなわち、メンバはメンバ部分復号鍵を持ち、サーバは n+1 個（メンバ数 n の場合）のサーバ部分復号鍵を持つ。(2,2)閾値秘密分散により、グループ秘密鍵の復元には、各メンバとサーバの二者が協力する必要があるため、サーバは１グループにつき複数のサーバ部分復号鍵を持っているものの、単独でグループ秘密鍵を復元できない。また、この構造はメンバにとってファイルを復号するために他の複数のメンバと協調する必要がなく、実用上都合がよい。さらに、メンバが複数のグループに多重帰属するときに、従来であればメンバは所属グループと同数の部分復号鍵を管理しなければならないが、次に述べるシェア・コントロールと呼ぶ技術により単一の鍵を管理すればよい仕組みを実現できる分散構造となっている。

秘匿したい情報 K を(k,n)閾値秘密分散を用いて n 人のメンバに分配して、各々が一つの分散情報を管理する状況とする。このとき、メンバのうち一人の分散情報を更新するのがシェア・コントロールと呼んでいるものである。これは分散情報を更新する前後で情報 K は変わらず、また情報 K を用いずに実行できる。更新後の分散情報を指定するある一人のメンバが主体となって処理を行うものの、すべてのメンバはシェア・コントロール前後で、互いの分散情報、および秘匿情報 K を知ることはできない。

グループ秘密鍵を先に述べた方法で分散管理し、各メンバとサーバはその分散情報である部分復号鍵を保持する場合、閾値復号を用いたファイル復号鍵の取り出しには現実的かつ効率がよい。閾値復号を用いた復号手順は次の通りである。まず、ファイル管理サーバは、メンバからファイルの入手リクエストを受け取ると、サーバ上に保管されている暗号化された共有ファイルとファイル復号鍵のうち、ファイル復号鍵をサーバ部分復号鍵を用いて部分復号する。次に、部分復号したファイル復号鍵をメンバ自身のメンバ部分復号鍵を使って完全復号する。最後に復号されたファイル復号鍵を使って共有ファイルを復号する。以上の手順において、ファイル復号鍵を復号するための二段階の復号処理として閾値復号を用いている。第一段階で、ファイル管理サーバ側で部分復号を行うので、サーバ管理者はその結果を知ることが可能だが、部分復号した結果からはファイル復号鍵に関する一切の情報を得ることはできないため、復号処理に参加しているにもかかわらず、ファイルを復号することはできない。また、この復号処理は、処理中にグループ秘密鍵が現れず、直接復号対象であるファイル復号鍵が出力されるため、各メンバとサーバはグループ秘密鍵を知らないままに復号処理を実行可能となる。当然、離脱メンバにもグループ秘密鍵を知られていないため、メンバが離脱するたびにグループ秘密鍵を生成する必要はない。

ファイルを共有するグループを管理するプロトコルは、グループ構築プロトコル、メンバ追加プロトコル、メンバ離脱プロトコル、鍵更新プロトコルの４つからなる。

グループ構築プロトコルは、サーバとグループメンバ一人が分散情報生成プロトコルを実行してグループ秘密鍵を生成する。グループ秘密鍵の精製後、グループ公開鍵を生成する。

メンバ追加プロトコルは、鍵分散木のリーフノードメンバのメンバ部分復号鍵に(2,2)

閾値秘密分散法を適用し、サーバと新規メンバに分散することで、新規メンバをグループに追加する。新規メンバが既に他のグループに所属している場合は、シェア・コントロールを新規メンバが実行し、サーバ部分復号鍵との関係を維持しながら、現在のメンバ部分復号鍵でファイル取得が可能になる。

メンバ離脱プロトコルは、離脱メンバの親ノードのメンバ部分復号鍵を、サーバと離脱メンバの子ノードメンバに再分散して離脱メンバのメンバ部分復号鍵をそのグループでは使用できないようにする。

鍵更新プロトコルは、サーバと全グループメンバが協力し、鍵分散木の各ノード間の関係を維持しながらメンバ部分復号鍵を更新する。関係を維持するために、ルートメンバから順に実行し、子ノードメンバに自身のメンバ部分復号鍵の更新量を伝播させる。

提案プロトコルの安全性を評価したところ次のようになった。グループ構築プロトコルはサーバとメンバにグループ秘密鍵を分散するためのやりとりをしており、それ以外のメンバ追加プロトコル、メンバ離脱プロトコル、鍵更新プロトコルでは既に所持しているサーバ／メンバ部分秘密鍵の情報を用いており、サーバ、メンバ、および第三者に鍵が漏れないことを確認した。すなわち、サーバにグループ秘密鍵がもれないこと、サーバにメンバ部分復号鍵がもれないこと、メンバにグループ秘密鍵がもれないこと、メンバにサーバ部分復号鍵がもれないこと、他のメンバにメンバ部分復号鍵がもれないこと、第三者にグループ秘密鍵がもれないこと、第三者にメンバ部分復号鍵がもれないこと、第三者にサーバ部分復号鍵がもれないこと、以上の 8 つについて、離散対数問題、検証可能秘密分散、(k, n) 閾値秘密分散、プロアクティブ秘密分散などの安全性に根拠を持つプロトコルとなっていることを確認した。

以上のプロトコルのスケーラビリティを評価するためにマルチエージェントシミュレーションを行った。本シミュレーションはターン毎に 1）新規タスク生成フェーズ、2）鍵更新プロトコル実行の判定フェーズ、3）ホスト行動フェーズという順に実行する。新規タスク生成フェーズでは、ホスト一台に対して、鍵更新プロトコル以外のプロトコルを生成する。鍵更新プロトコル実行の判定フェーズでは、最後に更新が行われてから一定ターンが経過していれば所属メンバに対して鍵更新プロトコルを生成する。ホスト行動フェーズではホストがプロトコルを実行する。

本シミュレータはホスト、サーバ、グループ、ファイル、ホスト管理者により構成される。

ホストは、サービス利用者が使用する端末、利用者は各端末を利用してファイル共有などを行う。ホストは複数のタスクを同時に処理することはできず、処理しきれないタスクがある場合はキューに保持する。ホスト行動フェーズでは、プロトコル実行の対象グループが更新中であればプロトコルを実行できない。この場合、ホストは実行中のタスクをキューへ戻し、次のタスクをキューから取り出して別のプロトコルの実行を試みる。ただし、キューからタスクを取り出せるのは 1 ターンにつき 1 タスクまでとする。ホストがとりうる状態は、各プロトコルの実行中である、グループ構築プロトコル実行中、メンバ追加プロトコル実行中・メンバ離脱プロトコル実行中・鍵更新プロトコル実行中・ファイル取得中・ファイル保管中、プロトコルを実行していない待機中、サーバのコネクション確立ができなかったときに遷移するコネクション確立待ちの 8 状態である。

サーバは、サービス提供者のファイル管理サーバで、各グループのサーバが管理する鍵とファイルの保管、ホストからリクエストのあったプロトコルを実行する。また、グループ情報を管理する。

グループは、ホストによって形成されるグループで、この単位でファイル共有を行う。ファイルは、グループで共有されるファイルで、そのファイルサイズはファイル共有プロトコル生成時に決定され、サイズの分布は Double Pareto Distributions に従う。シミュレーションコントローラは、新規タスクの生成など、シミュレータの各要素の動作管理を行う。

構築したシミュレータにより、鍵分散木の深さに応じて鍵更新プロトコルの実行時間がどのように変化するかを調べた。シミュレーションに登場するホスト数は 2048 台で、通信スループット、通信遅延はすべて 10Mbps と 30ms とした。各プロトコルの生起確率は、ファイル共有プロトコルを 1 日 100 回実行し、グループ生成プロトコル・メンバ追加プロトコル・メンバ離脱プロトコルは 1 か月に 3 回の頻度で実行されると考え、ファイル共有プロトコルを 0.7、他の 3 つのプロトコルを 0.0007 とし、残りの 0.2979 を待機に割り当てた。鍵更新プロトコルは一定ターン毎に実行される。サーバの同時最大接続数は、ホストのリクエストには全て応答可能となるよう 2048 とし、最大ファイルサイズは 100GB、最大グループ数は 100 グループとした。

事前に構成要素ごとにプロトコルの処理時間を測定し、各プロトコルの処理を 1 ターンで完了することができるように 1 ターンを 150 ミリ秒と定義した。シミュレーション実行前に、メンバ数が 2048 人のグループが 1 グループ存在する状況でシミュレータを 1 ヶ月分実行しておき、システムがある程度使用されている状況を再現しておく。この状態を

シミュレーションの初期状態として評価を行った。このとき，グループ数は 100 とし、2048 人のグループ以外のメンバ数は数人程度である。

鍵分散木の深さが 2000 のとき、鍵更新プロトコルが実行に要するターンは約 7000 ターンであることがわかった。これを実時間に換算すると約 20 分間である。鍵更新プロトコルの実行はシステムが利用されない時間帯に行えばシステムとして問題なく利用できることが確認できた。あわせて、メンバ数が 2000 人を超えるグループが存在することは考えにくく、通常は高々40 人程度であろうと考えられる。このようなグループが 100 グループ存在する場合の鍵更新プロトコルの実行時間は 15 秒間程度であり、もし鍵更新プロトコル実行中にシステムを利用する場合でも支障をきたさないと考えられる。

以上のプロトコルを用いたグループファイル共有システムの実装を行った。まず、ファイル共有、グループ作成／削除、グループへのメンバ追加／離脱、鍵更新のシステムの機能について設計を行った。システムはファイル、グループ、ユーザの 3 種類のデータ群があり、適切に動作するようにデータベースの設計を行った。以上の設計に基づき、デスクトップアプリケーションとして実装し、各プロトコルの機能と動作を確認した。

５．主な発表論文等

〔雑誌論文〕（計 0 件）

〔学会発表〕（計 2 件）

1．佐々木啓，長澤悠貴，毛利公美，福田洋治，白石善明，野口亮司，“単一の鍵で多重帰属できるグループファイル共有システムの実装”，情報処理学会第 73 回全国大会，第 4 分冊，pp. 57-58，2Z-9，2011 年 3 月 2 日．

2．長澤悠貴，白石善明，毛利公美，福田洋治，“単一の鍵で多重帰属できるグループファイル共有プロトコルの評価”，情報処理学会第 72 回全国大会，第 3 分冊，pp. 667-668，4ZE-5，2010 年 3 月 10 日．

〔図書〕（計 0 件）

〔産業財産権〕
○出願状況（計 0 件）
○取得状況（計 0 件）

〔その他〕
なし

６．研究組織

(1)研究代表者
白石　善明（SHIRAISHI YOSHIAKI）
名古屋工業大学・工学研究科・准教授
研究者番号：70351567

(2)研究分担者
なし

(3)連携研究者
なし