2006－CSEC－35（10）
2006／12／8

P2P セキュアファイル共有システムにおける共有機能の改善

東森ひろこ　宇田隆哉
東京工科大学コンピュータサイエンス学部

本論文で述べるのは，小規模なグループの PC を，安全性を考慮した P2P 技術を用いて連携させることによりファイル管理コストを低減し，ネットワークに参加している各 PC の通信帯域と余剰ディスクスペースをシステム全体で共有するファイル保存システムである．また，複数人数によるファイル共有においては，そのグループのメンバーやファイル更新履歴，メンバーのファイルアクセス権限，メンバーの加入・脱退を常に管理することでグループ外の第三者によるファイルへの不正アクセスを排除し，情報漏洩の憂慮からユーザを解放する．本システムでのファイル保存先は一般の PC であるが，これらのファイルは共通鍵暗号方式を用いて暗号化し，また個人の認証には公開鍵暗号方式を利用するため，ファイルの秘匿性が保たれる．

An Improvement in Sharing Process of Secure File-Sharing System with P2P Technology

Hiroko Tomori　Ryuya Uda
Tokyo University of Technology School of Computer Science

In this thesis, we make an observation on a file preservation system that decreases file management cost by making PC of small-scale group cooperate by using P2P technology that considers safety, and shares communication band of each PC that participates in network and in surplus disk space in the entire system. Moreover, in the file sharing by two or more people, the malicious computer access to the file by the third party outside the group is excluded by always managing the member of the group, file update history, member's file access authority, and the joining and the accession. Hence the user is free from the anxiety about leaking of information. In order to hide files stealthily in general PC, each file on the system is encrypted with common key cryptography and each user on the system is authenticated with a public key cryptography.

1. はじめに

1.1 背景

近年，高度情報化社会の発展に呼応して PC の普及率が高まり，ネットワークの技術も飛躍的に進歩してきている．データを電子的に扱う機会も増え，ユーザ間でファイルの交換や共有を行ったり，個人のファイルを自宅と会社の PC 間でやりとりしたりする技術が進歩し，普及してきた．

また，近年の PC は，ファイルを保存するには十分なほどの大容量ディスクを持っており，余剰ディスクスペースをもてあます人も多い．一方，保存されるデータにも高画質な画像や映像を用いたものが多くなり，個人における効率的なファイル管理は困難になってきている．

1.2 問題点

現在広く普及しているファイルサーバは運用コストが高いか高度な管理知識を必要とし，ファイル共有ソフトはセキュリティ面で脆弱な点をかかえている．ファイル共有ソフトによっては匿名のままでファイルを共有可能なため，著作権と侵害となる違法なファイル共有が行われている場合もある．また，ファイル共有ソフトを利用することで感染するコンピュータワームもあり，セキュリティ意識が低いユーザや知識の乏しいユーザによるソフトの利用で，会社や個人の PC データが流出する事件も見受けられるようになった．

一方，ファイルサーバの利用においては管理コストが問題となっている．セキュリティ面を強化しつつも運用が容易なファイルサーバは高価であり，安価なサーバ高度な知識を持つ管理者を必要とするため人件費によるコストが増加する．大規模な企業や学術機関などであればファイルサーバの導入は比較的容易であるが，大学の研究室やグループ研究，会社でのプロジェクトチーム，開発チームなどの小規模な組織では導入は難しいといえる．

1.3 目標

今回の提案では，小規模なグループの PC を，P2P 技術を用いて安全に連携させることによりファイル管理コストを低減し，ネットワークに参加している各 PC の通信帯域と余剰ディスクスペースをシステム全体で共有するファイル保存システムの構築を目標とする．

また，ファイル共有においては，グループのメンバー内で共有するファイルを効率的かつ安全に管理し，

第三者への情報漏洩を防ぐことに重点を置く．

2. 関連研究

本章では，本研究のベースとなるP2Pファイル共有システム1)，2)，3)における，ファイルの保存・復元・共有についての概要を説明する．このシステムは，ファイルを分割保存するシステム 4)に参加および離脱の概念を加えたものである．

2.1 システム利用環境

既存システムおよび本システムでは図 1 のように，保存対象のファイルを暗号化・分割し，ネットワーク内にあるPCに分散保存する．ユーザは保存対象のファイルに対して共通鍵を用いて暗号化を行い，冗長性を持たせて分割し，ネットワークに参加している各ユーザのPCに保存する．このネットワークは大学の研究室内・社内のプロジェクトチームなどの小規模なグループを対象としており，各ユーザはネットワークへの参加・離脱が自由な環境である．

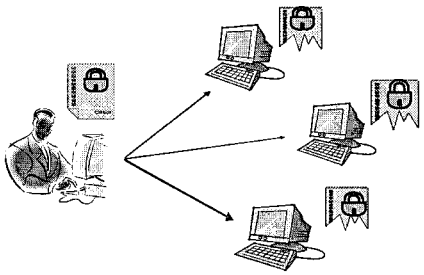

図 1 システム概要

既存システムは IP アドレスが浮動である場合にも対応しており，ユーザも使用するPCを自由に変更できる．既存システムではPCにはPCIDを，ユーザにはユーザIDを割り振っており，PC およびユーザは IP アドレスに依存することなく管理される．これらの IDはアプリケーションの初期設定で決定され，次回以降の利用では同一 IDが使用される．PCIDは16文字のユニークな文字列で構成される．

2.2 ファイルの保存

ユーザはファイルを保存する際，毎回生成される共通鍵で暗号化する．そして，ファイルの重要度に応じた任意の冗長性を持たせてファイルを複数に分割し，ネットワーク上のPCに分散保存する．分割されたそれぞれのファイルにはランダムで128文字の名前がつけられる．次に，ファイルを暗号化した共通鍵を，分割した個々のデータに付加し，そのハッシュ値を計算する．ハッシュアルゴリズムにはSHA-1が用いられている．このハッシュ値は，分割ファイルを取得する際のデータ破損検出とユーザの閲覧権限の有無を確認するために用いられる．同様の手順で，異なる共通鍵を分割ファイルに付加し，そのハッシュ値を計算したものが削除ハッシュである．これは削除用の鍵であり，ファイルを削除する際にハッシュ値との一致を確認する．図2に分割ファイルのデータ形式を示す．図2のとおり，データの1番目に付加される情報が閲覧鍵，2番目に付加される情報が削除鍵となる．

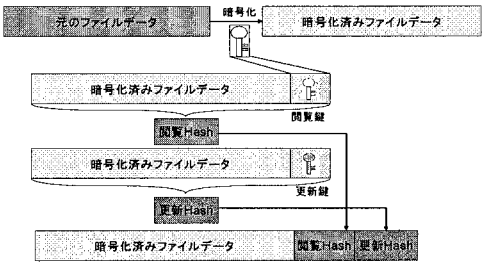

図 2 分割ファイルの形式

次に複数に分割されたそれぞれのファイル断片について，ファイル名，保存先PC，復元に必要な閾値を含む復元情報が記されたプロパティファイルが生成され，ネットワーク上のPCに保存される．プロパティファイルはファイルを保存したユーザ個人の公開鍵で暗号化されており，その公開鍵はユーザの初回ログイン時生成され，以降のログイン認証に利用される．そして最後に，ユーザが保存したファイルと，それを復元させるためのプロパティファイルの情報を記述するリストファイルが生成される．リストファイルもまた，ユーザ個人の公開鍵で暗号化された後に，ネットワーク上のPCに保存される．プロパティファイルとリストファイルは128文字のファイル名を持つが，リストファイルはユーザがシステムにログインする際に最初にアクセスするファイルであるため，図3に示す命名規則によって決定される．

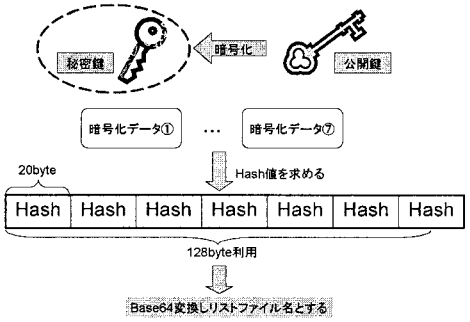

図 3 リストファイル命名規則

まず，ユーザ個人の秘密鍵を公開鍵で暗号化し，得られた暗号化済みデータをインクリメントしながらハッシュ計算(SHA-1)を7回繰り返し，それらのハッシ

ュ値を連結する．結果として得られた 140Byte のデータを BASE64 変換し，その先頭部分から 128Byte 抽出したものをリストファイル名とする．上記手順によりユーザ自身の鍵ペアから一定長のファイル名をもつリストファイルを生成できる．

2.3 ファイルの復元

まずユーザは，自身の公開鍵・秘密鍵のペアを用いてシステムにログインし，2.2 節に示した手法でリストファイルを取得し，秘密鍵で復号してリストファイルの内容を取得する．その後，リストファイルの内容からユーザがネットワーク上に保存しているファイルの一覧が得られるので，取得したいファイルを選択する．このとき同時に復元させるためのプロパティファイルも選択することになるため，プロパティファイルが保存されているネットワーク上の PC から取得される．得られたプロパティファイルをユーザの秘密鍵で復号し，その情報をもとに，ネットワーク上に保存されていたそれぞれの分割ファイルが取得される．分割ファイルが復元の閾値を上回る数だけ集まると，プロパティファイルに含まれる共通鍵を用いて元のファイルが復号される．

2.4 ファイルの共有

ファイルの共有は，2.2 節，2.3 節で述べた手順を基に行われる．

まずユーザは共有したいファイルを保存し，共有するユーザの人数分プロパティファイルを生成する．その際それぞれのプロパティファイルは，各ユーザの公開鍵によって暗号化され，ネットワーク上に保存される．共有ファイルが更新された場合は，各ユーザに共有ファイルが更新された旨を伝える書置きが送信される．その書置きを受け取ったユーザは書置きの内容を解析することで，編集されたファイルに対応するプロパティファイルを更新して，ファイルの更新作業は完了となる．

2.5 問題点

2.2 節に示すとおり，プロパティファイルは共有ファイルの復元情報がしるされたファイルである．そのため，共有ファイルが更新されるたびに，新しく書き換える必要がある．したがって既存のシステムでは，共有したい人数分のプロパティファイルを，その都度暗号化しなければならず，演算処理の回数が増加するため効率が落ちる．

また，プロパティファイルの更新に伴い，それを回収するため各ユーザの持つリストファイルの更新が必要不可欠となる．しかし，リストファイルの更新を行うのは書置きを受信した各ユーザであるため，長期不在や怠慢による更新の滞りが考えられる．このような未更新ユーザが多く発生することにより，円滑なシステムの利用が困難になりうる可能性がある．

そこで本システムでは，既存システムのボトルネックとなるプロパティファイルを共有することで，演算回数の減少を図ると共に，分割保存するファイルと PC への書置きに寿命を持たせることでユーザに対する書置きを不要にし，システム効率の向上を図る．

3. 提案共有システム

3.1 提案手法

本システムではプロパティファイル，リストファイルを共有・自動更新し，書置きの利用方法を変更する．

既存システムでは各ユーザがそれぞれのプロパティファイルを有していたため，ファイルが更新される度に各ユーザのプロパティファイルも更新する必要があった．しかし本システムではプロパティファイルをユーザ間で共有し，同名のファイルには常に同じプロパティファイルを使用し続け，更新があればその都度新しい情報を追記していくことで，常に最新のプロパティファイルを参照することが可能となる．さらに，プロパティファイルとリストファイルを共有することで，更新も自動的に行うことが可能となる．

次に書置きの利用法について変更点を述べる．本システムでは，既存システムでユーザ宛に送信していた書置きを，各 PC への処理命令の書置きとする．これによりユーザが直接旧ファイルを参照・削除を行うことなく，PC が自動的に書置きの内容を参照し，旧ファイルを削除することが可能となる．

また今回は新たに，共有するユーザの名前やファイル更新履歴，ユーザのファイルアクセス権限，ユーザの加入・脱退を常に管理する共有メンバファイルを作成する．これによりグループ外の第三者によるファイルへの不正アクセスを排除し，情報漏洩の憂慮からユーザを解放できると考えられる．加えて，ファイル更新情報内にユーザ名とアクセス時刻を含ませることにより，ユーザの落ち度によりファイルデータの中身が外部漏洩した場合の責任の所在も明確にできる．

3.2 共有システム詳細

まずユーザはファイルを共有するためのグループの公開鍵と秘密鍵を作成する．次に共有したいユーザのユーザ名と，ファイルに対するユーザの権限を，共有メンバファイルを作成し書き込む．この権限とは，共有ファイルを削除可能か否かについての権限である．他は 2.2 節同様，ネットワーク上にファイルを分割し

保存するという作業を行う．ファイルが保存されると，誰がいつファイルを保存したかの履歴が共有メンバファイルに書き込まれることになる．そして，プロパティファイルと共有メンバファイルがネットワーク上に保存され，最後にリストファイルが保存されることになる．ここでのプロパティファイルとリストファイルはすべてグループ専用のものになる．

ファイルを取得する際は図 4 のような手順となる．まずリストファイルから保存してあるファイルを選択し，プロパティファイルと共有メンバファイルをネットワーク上から取得する．このとき，共有メンバファイル内に書いてある，共有ユーザの名前と，ファイルを取得しようとしているユーザの名前が一致すれば，元の共有ファイルが復元されることになる．

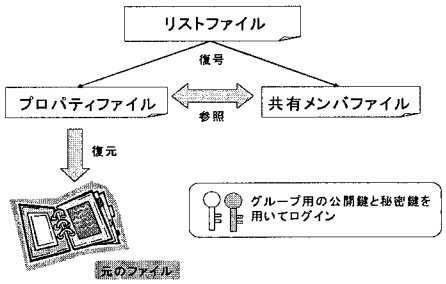

図 4　共有システム概要

3.3　書置き

共有ファイルを保存する際にはプロパティファイルと共有メンバファイルが更新される．しかし更新時にファイルを共有するメンバーが全員オンラインであるとは限らないため，オフライン PC は古いプロパティファイルと共有メンバファイルを持ったままになり，更新作業を正常に完了させることができない．

そこで提案する手法が書置きである．図 5 に書置き方式の概要を示す．まずファイルを更新した PC はオフラインの PC に対して，古いファイルの削除依頼を出す．削除依頼の書置きは，このときオンラインである PC 全てが共有する．書置きを持っている PC は，書置き対象の PC がオンラインになった時点でその情報を通知する．対象の PC は書置きに書かれている内容を解読してファイルを削除する．無事に削除が完了すると，次は削除完了書置きを依頼する．削除完了書置きを受け取った PC は以前の削除書置きを削除して，更新作業は完了となる．本システムでは，書置きは PC に対してのみ必要となり，ユーザに対する書き置きは不要である．

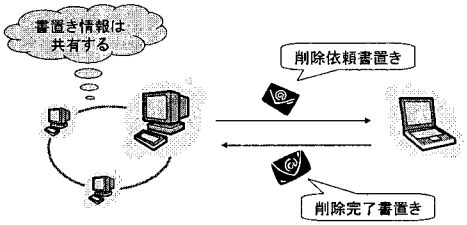

図 5　ペアリング書置き方式

本方式の流れを以下に示す．削除依頼書置きと，削除完了書置きのペアリング書置き方式を用いると，オフライン PC がオンラインになった時点で共有されていた書置きが受信されることになり，最終的に全 PC に書置きが送られ，また不要になった書置きも削除することができる．書置きのペアリングが正しく行われない場合に関しては，次の 2 つが予想される．まず，削除依頼書置きが存在しない状態で削除完了書置きを受信してしまった場合であるが，この場合は受信した削除完了書置きを削除する．次に削除完了書置きの受信タイミングを逃したために削除依頼書置きがいつまでも存在し続ける場合であるが，この場合はあらかじめ定められた一定時間経過後に削除依頼書置きを削除する．これは対象となる PC が長期間ログインできなかったために書置きの依頼が為されなかった，もしくは書置きの依頼が為されたタイミングで自分自身がログインしていなかったかのいずれかであるが，いずれの場合でも削除依頼書置きを削除することで対応する．なお，ファイルの保存期間は書置きの保持時間よりも必ず短く設定することで，不要なファイルが PC に存在したままになることを防ぐ．

今回はこの書置き方式で古いファイルを削除していくため，ユーザに依存することなくファイルの更新・削除を行うことが可能となる．

4.　実装

本システムは暗号化・および分割に crypto++ 5)を用いて実装を行った．使用した PC の OS は Windows XP, CPU は Celeron 2.8GHz, メインメモリは 512MB である．

4.1　共有メンバファイルとプロパティファイル

本システムで共有ファイルを保存する際には，更新情報を含む共有メンバファイルと，復元情報を含むプロパティファイルが必要となる．図 6 に共有メンバファイルとプロパティファイルの構造をまとめる．

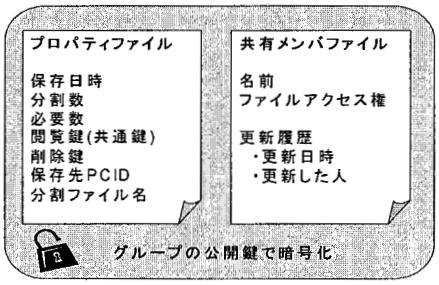

図 6　復元時に必要なファイルの構造

それぞれの項目は以下のように決定する．時間関係項目には標準で用意された localtime 関数(1970 年 1 月 1 日の 00:00:00 からの経過時間)を日時として扱う．保存日時は復元したいファイルの作成日時を判別するために用いる．分割数，必要数はファイルを復元するために必要なファイルサイズを知るために記録しており，共に 4 バイトである．

閲覧鍵と削除鍵については 2.2 節で述べたとおり，共通鍵暗号方式による共通鍵であり 16 バイトの長さを持つ．閲覧鍵はファイルを保存する際に暗号化した鍵であり，ファイル復元時にも利用される．また，分割された個々のデータの閲覧ハッシュを計算する際にも用いられる．上記手順により，ネットワーク上に保存してあるデータの破損検出が行える．削除鍵については 4.3 節で詳細を述べるが，これも閲覧鍵と同様に保存されるファイルに対して一意の鍵となる．

この閲覧鍵と削除鍵は共有メンバファイル内のファイルアクセス権とリンクしている．共有ファイルに対して，「読み，書き，削除」ができるユーザと，「読み，書き」しかできないユーザを分離して管理することで，UNIX と同様のアクセス管理機能を持つファイルシステムが実現できる．共有ユーザ名はシステム加入時に自動で生成され，一律 20 バイトで決定される．保存先 PCID に関しても PC のシステム加入時に一律 16 バイトで決定される．分割ファイル名は Windows で設定することのできる最大長の 128 バイトで一律に自動生成される．上記で示した分割数分だけ完全にランダムなファイル名が生成される．これらの名前データはシステム運用上不変となっており，ユーザの意思で変更することはできない．

4.2　ファイルの更新

本システムでは共有ファイルが更新されても，もとのファイル名とはまったく違うファイル名にしない限り同じプロパティファイルに分散情報が追記されていくことになる．この手順により，2 章で述べた既存のシステムのようにファイルが更新されるたびにプロパティファイル容量が増加する問題は解決する．しかし，ここでまた新たな問題が発生する．ファイルの更新時にオンラインであった PC1~PC3 が，ファイルの保存時にもオンラインであるとは限らない．保存時にもオンラインであるならば，新しく追記されたプロパティファイルの享受を正しく行うことができるが，このシステムはユーザがいつでも自由に参加・離脱できるシステムであるため，保存先 PC がいつでもオンラインであるとは限らない．

そこで今回更新時には，3.3 節で述べた書置き方式を用いる．ユーザはファイルを編集し，プロパティファイルと共有メンバファイルをオンラインの PC に保存した後，オフライン PC に対して削除依頼書置きを送信する．この書置き方式を用いることで，古いプロパティファイルや共有メンバファイルを削除することができ，ユーザは常に最新の状態のファイルにアクセスできると考えられる．

4.3　ファイルの削除

本システムにおけるファイル削除の方法を以下に示す．共有ファイルを削除したい者は，まず削除対象のプロパティファイルと共有メンバファイルを取得する．グループ用鍵ペアの秘密鍵を持たない者は各ファイルへのアクセスが不可能なため，グループ外部からの不正アクセスはできない．取得したファイルはグループ用公開鍵で暗号化されているため，これを対になる秘密鍵で復号する．その後，共有メンバファイル内の，ユーザ名とアクセス権を参照し，削除権限の有無を確認する．権限を持つユーザである場合，このプロパティファイル内に含まれている削除鍵データと削除したい対象ファイル名を，ネットワーク上の PC に対して送信し，削除依頼を行う．こうしてネットワーク上に削除鍵が公開され，指定されたファイルはすべて削除対象となる．そして削除対象のファイルが見つかると，削除鍵をファイルに付加してハッシュ値を取る．このハッシュ値と，分割ファイルに付加されていた削除ハッシュが一致した場合にファイルを削除する．

もしファイルを持つ PC がこの時点でオフラインである場合には3.3節で述べた書置きを行うことになる．以上の手順でファイルの削除は完了する．

5.　考察

5.1　優位性

本提案システムでは，ネットワーク内の PC の余剰ディスクスペースにファイルを分散保存するため，PC の余剰領域が少ないユーザもファイルの保存，バック

アップを行うことができる．また，PCが故障した場合も重要なファイルをあらかじめ分散保存しておくことで，ファイルの復旧が容易になる．

このシステムはネットワークに常時接続しているPCをファイルサーバと見立てて，ファイルを保存するというものではない．保存対象のPCは個人が通常利用しているもので，ネットワーク利用状況は個人に完全に依存するかたちになる．よって常に分割したファイルが保存してあるPCがオンラインであるとは限らないという問題が生じる．しかし，本システムではファイルは冗長性を持たせて分割し，保存するため保存先PCが常に全部オンラインでなければならないという問題は解決する．これによって，分割ファイルの回収が容易になり，PCを管理する必要性もなくなり，コストもかからない．

今回は共有ファイルを管理するための仕組みも新たに整えることによって，ファイルの所在やグループのメンバーが常に最新のものに更新され，グループ外のメンバーによるデータの閲覧，漏洩を防ぐことが可能になる．また，グループ内のメンバーにはUNIXのようなファイルのアクセス制限を設けることによって，ファイルに対する不正なアクセスを防ぐことができる．さらに，閲覧者・更新者も常に管理されるので，ファイルが閲覧者による不正な改ざんがあった場合も責任の所在を明らかにできる．

5.2 問題点

まずこのシステムの問題として挙げられるのが，初期ノードの決定方法である．初期ノードとは，システム起動時に最初にアクセスするノードであり，リストファイル・プロパティファイル・共有メンバファイルの保存場所となる場所である．よってこれらファイルが取得不可能となると，ネットワーク上に保存してある元のオリジナルファイルも復元不可能となってしまう．初期ノードは利用しているPCと同時に稼動していることが多い，つまり相性のよいPCにしなければならない．しかし，本システムでは初期ノードを，相手と自分のPCの稼働時間の累計から算出しているだけなので，稼働時間にばらつきがある場合に初期ノードを決定することは困難となる．

また共有メンバファイルについても問題点が挙げられる．共有メンバファイルには，共有ファイルが更新されると，更新時間とユーザ名が記録されていくことになるが，悪意のあるメンバーがファイルを共有するグループ内にいて，誰かの名前を勝手に利用してファイルを改ざんすると，責任の所在があやふやとなってしまう問題が起こってしまう．そこで解決策として考えているのが，共有メンバファイルの更新情報の部分に，更新者の署名を付加することである．これにより，より確実にファイルを更新でき，かつ問題が発生した場合の責任の所在をはっきりさせられる．

6. まとめ

本システムでは，共有ファイルの情報が記述されているプロパティファイルも含めてネットワーク上に分散・保存されているため，システム全体の設計は少々複雑になっているが，本システムを利用するユーザは簡単にネットワークに参加し，システムを利用できる．また，それぞれのPCに分散保存されているファイルには，それが分割されたファイルなのか，それとも復元に必要なファイルであるのか判別不能な名前が付与されているため，第三者には区別がつかず，公開情報を最小限にとどめている．さらに，保存するファイルに寿命を持たせることにより，ユーザに対する書置きを不要とした．ゆえに，本システムは簡単かつ安全に効率よくファイルを保存する環境を提供できる．

謝辞

本プロジェクトは，2006年度(平成18年度)情報処理推進機構(IPA)未踏ソフトウェア創造事業「未踏ユース」に採択されました．

参考文献

1）大津一樹，宇田隆哉，伊藤雅仁，市村哲，田胡和哉，星徹，松下温,“アクセス制御機構を持つP2Pファイル共有システム”電子情報通信学会SCIS2005論文集 vol.1 pp.13-18,2005

2）鹿島隆行，宇田隆哉，伊藤雅仁，市村哲，田胡和哉，星徹，松下温,“PC共有による安全で低コストなP2Pファイル分散システム”電子情報通信学会SCIS2005論文集 vol.1 pp.1-6,2005

3）大津一樹，宇田隆哉，伊藤雅仁，市村哲，田胡和哉，星徹，松下温,“P2Pファイル共有システムにおける鍵管理効率化手法の検討”情報処理学会 DICOMO2005論文集 pp.609-612,2005

4）石田祐子，井上徹，佐久間隆，下田秦戈，金平恵実，宮前俊一，竹久達也，森本健嗣,“データ分散保存システム”平成13年度情報処理進行事業協会セキュリティセンター：電子政府情報セキュリティ技術開発 http://www.ipa.go.jp/security/fy13/tech/crypto_vss/data_report.pdf

5）Crypto++® Library5.2.1, http://cryptopp.sourceforge.net/docs/ref521/index.html