秘密分散を用いたセキュアなクラウドストレージシステムの実装方式

SHIRAYAMA, Tomoyasu / 白山, 智康

(出版者 / Publisher)
法政大学大学院理工学・工学研究科
(雑誌名 / Journal or Publication Title)
法政大学大学院紀要. 理工学・工学研究科編 / 法政大学大学院紀要. 理工学・工学研究科編
(巻 / Volume)
57
(開始ページ / Start Page)
1
(終了ページ / End Page)
6
(発行年 / Year)
2016-03-24
(URL)
https://doi.org/10.15002/00013277

秘密分散を用いたセキュアなクラウドストレージシステムの実装方式

IMPLEMENTATION AND EVALUATION OF SECURE CLOUD STORAGE SYSTEM USING A SECRET SHARING SCHEME

白山智康
Tomoyasu Shirayama
指導教員　金井敦

法政大学大学院理工学研究科応用情報工学専攻修士課程

Cloud services are becoming more popular. As their price has been decreasing, the opportunity to use them has been increasing. However, with the popularity of public cloud services, the concern of confidentiality is recognized as the largest problem. We propose has approached for multiple clouds to maintain confidentiality using a secret sharing scheme. With our proposed approach, secret information is encrypted, and encrypted data is distributed using a secret sharing scheme. In this paper, we implement the prototype system of proposed approach and evaluate its performance.

***Key Words** : Clouds computing, Security, Multi-cloud,*

１．はじめに

近年，スマートフォンやタブレットが普及してきており，その間のデータ交換や共有手段としてクラウドストレージサービスの利用が急速に広まっている．クラウドストレージサービスは高い可用性を持ち，無料で利用できるサービスが数多くある．しかし，これらのサービスは第三者のクラウドサービスプロバイダが管理と運用をおこなうため，プロバイダによるデータの悪用や情報漏えいなどの機密性に問題がある．最近では，複数のクラウドサービスを利用して機密性の向上を図る方式[1][2][3]やクラウドのトラストモデル[4][5]の提案がされている．

データの機密性や可用性を向上でき，クラウドストレージサービスに適した技術として秘密分散法がある．秘密分散法は，秘密情報と呼ばれる秘密にしたいデータから分散情報と呼ばれる複数のデータを生成し，そのデータをあるしきい値以上の数集めれば元のデータを復元できるという特性を持つ符号化技術である．あるしきい値未満の数の分散データからは元のデータに関する情報が一切得られないため，クラウドサービスプロバイダによる悪用や情報漏えいを防ぐことができる．また，分散情報分散したデータからあるしきい値以上のデータから元のデータを復元することができるため，分散して保管したすべてのクラウドサービスが稼働している必要がないため可用性を向上させることができる．また，秘密分散法以外にも暗号化やハッシュアルゴリズムを利用することでセキュリティを向上させる研究も行われている．

そこで本稿では，”秘密分散を用いた複数クラウドのデータ管理方式”[3]で提案された方式をベースとしたシステムを実際のクラウドストレージサービスを複数利用して実装し，システムの性能評価を行い，その有用性を示す．

２．関連研究

（１）秘密分散法

秘密分散法の代表例として(k, n)しきい値秘密分散法がある．(k, n)しきい値秘密分散法[6][7]は，A.Shamir と Blakley が同時期に別々に考案した秘密分散法の原型とも呼べる手法であり，秘密情報を n 個の分散情報に分割して，そのうちの k 個を取得すれば秘密情報を復元できる手法である．しかし，この手法は分散情報のデータータサイズが肥大化し，ストレージの容量を圧迫するというデメリットを内容している．そこで，(k, n)しきい値秘密分散法の欠点である分散情報の肥大化を克服するために (k, l, n) ランプ型秘密分散法[8][9]が提案されている．(k, l, n) ランプ型秘密分散法は，特定の条件下において元の情報が一部復元できるが，分散情報のデータサイズは小さくすることができる方式である．(k, L, n)ランプ型秘密分散法における各分散情報の最低データサイズは秘密情報の$1/L$倍となる．このため，全ての分散情報集めた場合でも，秘密情報のn/L 倍に抑えることができる．この方式には分散情報がk 個以上集まった際の復元時にk - 1次の多項式を処理する必要があり，その計算負荷の大きさが問題となっているが，排他論理和(XOR)

を用いることで秘密分散処理を高速化するアルゴリズム[10][11]が提案されている．本稿では，XORを用いた高速な(k, L, n)しきい値秘密分散法をクラウドストレージ上で保存するファイルに対して使用する．

（２）複数クラウドを用いたセキュアなクラウドストレージシステム

クラウドサービスは高い可用性を持ちコストが少ない利点があり，広く普及しているが，機密性に問題を有している．そこで，可用性を保ちつつも機密性を向上させる目的で，秘密分散を用いた複数クラウドのデータ管方式[3]が提案されている．この方式では，秘密情報を暗号化し，その暗号文および鍵情報を共に秘密分散して複数クラウド，複数クライアントに保存し，機密性を向上させる．また，複数クライアントの同意により秘密情報を復元する方法を取ることにより，人的ミスや不正な情報持ち出しによる情報漏えいを防ぐことを可能にする．

また，複数のクラウドサービスを利用するマルチクラウド環境では，ユーザが自らの手でサービスを選定する必要があるため煩雑となっている．そこでマルチクラウドにおける最適なクラウド決定方式[2]が提案されている．ユーザの要求の定量化およびクラウドサービスの定量的な評価を行い，自動的にマッチングを行う方式である．さらに，クラウドストレージに保存するデータごとに動的に分散方法を変化させることで，セキュリティやコストなどの最適化を目指している．

３．提案システム

（１）システム構成

提案するセキュアなクラウドストレージシステムの構成を図１に示す. 本システムは”秘密分散を用いた複数クラウドのデータ管理方式”[3]で提案された方式をベースに個人が新たなストレージを用意することなく，複数のクラウドストレージサービスを利用し，データを安全に保存することを目的としている．

ユーザは秘密情報を暗号化し，暗号文情報を分散処理サーバに送信する．暗号化はローカルで行うため秘密情報の漏洩を防ぐことができる．さらに，分散処理サーバは受け取った暗号文情報を(k, l, n)秘密分散法で分散し，各クラウドサービスプロバイダに保管する．また，分散処理サーバは検証用のハッシュ値と復元用の識別情報を生成し，復号時に改ざんの検知やデータの完全性を確保する．

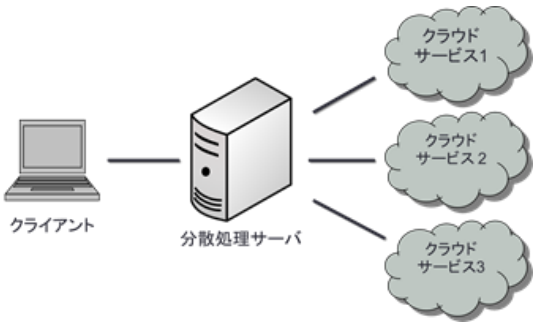

図１　セキュアなクラウドストレージシステム

本システムでは, 表 １のトラストモデルに基づいて安全性を確保する．分散処理サーバはクライアントから知られていて，仮に分散処理サーバ自身が何らかの情報を持っていた場合にクライアント側から取得される可能性があり，またクラウドサービスとも通信を行うため情報漏洩の危険があり，分散処理サーバのデータストレージは信頼しない．しかし，分散処理サーバ上で動作するプログラムは改ざんされないものとする．クライアントのデータストレージおよびプログラムは信頼する．クラウドサービスは単体でのデータストレージは信頼しない．信頼しないストレージであっても，秘密分散法によって生成されたデータ分散情報をしきい値未満保管する場合は利用できるものとする．

表 １　トラストモデル

名称	信頼性
クライアント	データストレージとプログラムを信頼する
分散処理サーバ	データストレージは信頼しない プログラムは改ざんされない
クラウドサービス	単体でのデータストレージを信頼しない

（２）データ管理方式

本システムの具体的な手順をについて述べる．分散過程と復号過程について下記に示す．文献[3]の方式では秘密情報を暗号化し，その暗号文および鍵情報を共に秘密分散して暗号文の分散情報は複数クラウド，鍵情報の分散情報は組織内のユーザに分配して保存する．秘密情報の復元時には複数クライアントの同意により秘密情報を復元する方法を取っている．しかし，本稿では個人の利用を想定しているため暗号化に用いた鍵は秘密分散を行わずそのままクライアント端末で保管する.あとは文献[3]の方式と同様である．

分散過程

1. データ M を保存したいユーザは，M を共通鍵 K を用いて暗号化（AES）し，秘密情報 S を作成する．
2. 秘密情報 S を分散処理サーバに送信する．
3. データ分散サーバは(k，L，n)しきい値法を用いて秘密情報 D から n 個の分散情報を作成する．
4. 分散処理サーバは暗号文情報の各分散情報からハッシュ値を生成する．
5. 分散処理サーバはハッシュ値情報と秘密分散の識別情報をユーザ端末のストレージに保管する．
6. 分散処理サーバは暗号文の各分散情報を各クラウドストレージサービスに保管する．

復号過程

1. 分散過程で秘密情報 S をアップロードしたユーザはストレージにあるハッシュ値情報と秘密分散の識別情報を分散処理サーバにアップロードする．
2. 分散処理サーバは任意の k 個のクラウドストレージサービスを選択し，分散情報をダウンロードする．
3. 復号過程の手順 1 でアップロードされたハッシュ値情報を用いて分散情報を検証する．正しくない分散情報があった場合，他の分散情報を取得する．
4. 分散処理サーバは(k，l，n)秘密分散法を適用して秘密情報 S を復元する
5. ユーザは分散処理サーバから秘密情報 S をダウンロードして共通鍵 K を用いてデータ M を復元する

（３）システム安全性

本システムにおける，盗聴，改ざんに対する安全性をトラストモデルに基づき検証する．

・盗聴

分散処理終了時の状態において，攻撃者もしくはクラウドサービスプロバイダ自身がクラウドストレージ上に保管しているデータ分散情報を盗聴によりしきい値k 個以上取得できたと仮定する．攻撃者はしきい値k 個以上取得したデータ分散情報より暗号化ファイルを取得することは可能である．しかし，クライアントに保管している鍵データを取得しない限り，暗号化ファイルから元データを復元することはできない．したがって，トラストモデルよりクライアントのストレージは信頼され安全であるため，攻撃者は鍵分散情報を取得することはできないため，盗聴に対しては安全であると言うことができる．

・改ざん

分散処理終了時の状態において，攻撃者もしくはクラウドサービスプロバイダ自身によって，クラウドストレージ上に保存しているデータ分散情報が改ざんまたは誤りが発生したと仮定する．提案手法では，分散処理サーバで識別情報ファイルに記録されたハッシュ値を利用して分散情報の正当性の検証が行われる．この手順を行うことで，改ざんや誤りが発生した正当ではない分散情報を除外することができる．そのため，攻撃者もしくはクラウドサービスプロバイダによって改ざんされたデータ分散情報の数が$n-k$ 個以下だった場合は，改ざんされていないk 個以上のデータ分散情報で暗号化ファイルを復号することが可能である．また，改ざんされたデータ分散情報の数が$n-k$ 個より大きかった場合は，改ざんされていないデータ分散情報がk 個に満たないので暗号化ファイルを復号することができない．

４．実装

本システムを実環境上で実装し，システムの検証を行った．開発言語は Java である．実装における全体のシステム構成を図 2 に示す．クライアントとしてノート PC，分散処理サーバとしてサーバ PC，クラウドストレージとして一般に利用されている Dropbox，GoogleDrive，OneDrive，Box の 4 つクラウドストレージサービスを用意した．

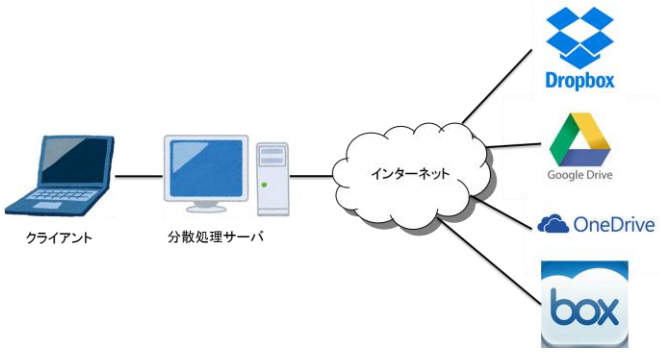

図 2．実装における全体のシステム構成

クライアントとしてのノート PC と分散処理サーバとしてのサーバ PC のスペックは表 2 に示す．

表 2．実装環境

	クライアント	分散処理サーバ
CPU	Intel Core　i3 1.7GHz	IntelCore　i7 2.8GHz
メモリ	4GB	4GB
OS	Windows7	Windows10

プログラムの構成は表 3 に示す．クライアントのプログラムはファイル読込と書込，AES によるファイルの暗号化と復号化，分散処理サーバとの通信を行うプログラムがある．またサーバのプログラムはクライアントとクラウドサービスとの通信，秘密分散による符号化と復号化，メッセージダイジェスト，識別情報ファイルの作成などを行うプログラムがある．

表 3. 各部位の機能

名称	機能
クライアント	データの共通鍵暗号化 データ分散サーバとの通信
分散サーバ	(k, L, n)しきい値法による符号化 ハッシュ値の生成 クライアントとの通信 クラウドサービスとの通信
クラウドサービス	データの保存

今回の実装における秘密分散は，東京理科大学の高荒らが提案した”XOR を用いた高速なランプ型秘密分散法”[11]を用いる．クライアントと分散処理サーバ間の通信はJava ソケット通信を利用して行いTCP プロトコルを利用している．また，クラウドストレージサービスとの通信は，それぞれのサービスプロバイダが公式提供しているAPI を利用し通信を行う．

(k, l, n)ランプ型秘密分散法により各クラウドサービスにアップロードするデータサイズは元データのサイズより小さくすることができるがクラウドストレージサービス全体にアップロードするデータサイズは元データのサイズより大きくなるので各クラウドストレージサービスと通信は Java のマルチスレッドの機能を利用し，パフォーマンスの向上を図っている．マルチスレッド仮想的に複数の処理を同時に実行することが可能なので，それぞれのクラウドストレージサービスの通信の完了を待たずに通信を開始することができる．

プログラムは実行可能 JAR ファイルとして保存し，クライアント PC ではコマンドプロンプトを利用し，サーバ PC では端末を利用してプログラムを実行する．

実装したプログラムの分散処理時と復号処理時の処理手順の詳細について述べる．

・分散処理時

まず，クライアントプログラムでは引数を入力し起動する．引数は秘密分散のパラメータ，分散処理サーバのIP アドレス，アップロードするファイルのパスを指定する．プログラム起動時はファイルのバイト配列読込，AES での鍵生成，ファイルの暗号化が行われる．次に入力された IP アドレスとソケット通信が実行される．通信が確立できない場合は例外が発生し終了する．サーバとの通信が確立された後，引数で指定された秘密分散用のパラメータ，暗号化ファイルがサーバ PC に送信される．サーバ PC ではクライアント PC から送られてきた情報を基に，暗号化ファイルに対して秘密分散で符号化を行う．またメッセージダイジェストによりそれぞれの分散情報のハッシュ値を取得し，データ分散情報用の識別情報ファイルを作成する．識別情報ファイルには，分散情報のパスとハッシュ値が記録されている．秘密分散処理が終わった後各クラウドストレージサービスとの通信を行い，データ分散情報を各クラウドサービスアップロードする．また，2 つの識別情報ファイルをクライアント PC に送信する．クライアント PC 上で 2 つの識別情報ファイルを保存する．また秘密分散用のパラメータと入力ファイルの拡張子を記録した設定情報ファイルを作成し，同じくクライアント PC 上で保存する．以上で一連の分散処理が終了する．

・復号処理時

クライアントプログラムが起動すると設定情報ファイルより秘密分散用パラメータと拡張子が読み込まれ，サーバ PC とのソケット通信が行われる．サーバとの通信が確立された後，引数で指定された設定情報ファイルから読み込んだ秘密分散用のパラメータ，識別情報ファイルがサーバ PC に送信される．また各クラウドストレージサービスからデータ分散情報のダウンロードを行う．次に秘密分散での復号を行うとともに，ハッシュ値を利用しての改ざんの確認を行う．復号終了後，データ分散情報から得た暗号化ファイルをクライアント PC に送信する．クライアント PC 上で鍵を利用して暗号化ファイルの復号を行い，設定情報ファイルの拡張子のファイル形式で元のファイルを復元する.そしてクライアントPC 上に復元されたデータが保存される．以上で一連の復号処理が終了である．

５．評価

実装したシステムを評価するために暗号化や秘密分散の処理時間とクラウドストレージサービスと通信時間を測定した．各測定値は 10 回試行した平均値である．(k, L, n)しきい値秘密分散法のパラメータは k = 3, l = 2, n = 4 である．

（１）暗号化や秘密分散の処理時間

クライアント上のデータを暗号化(AES)と分散処理サーバの秘密分散のそれぞれの分散時の処理時間を図 3, 復号時の処理時間を図 4 に示す.

図 3 より，分散処理時間はデータサイズに比例して AES による暗号化時間および秘密分散処理の時間が増えていることがわかる．秘密分散処理にかかる時間は暗号化処理に比べて，処理時間上昇の傾きが大きいことから，あまりに大きな情報を扱う際には非常に時間がかかってしまうと考えられる．しかし，100MB もの容量のデータでさえも分散保存にかかる時間が 5 秒足らずで完了していることから，高い安全性と高い可用性を得るためには十分実用レベルであると考えられる．

図 4 より，復号処理時間も分散処理時間同様にデータサイズに比例して処理時間が増えていることがわかる．分散処理のときと異なる点は秘密分散処理の時間とAES による復号化処理の時間が等しいことである．AES

は非常に高速で動作することが知られている共通鍵暗号方式であるが，秘密分散の復号処理もそれに匹敵する速度が出ることから実装した秘密分散方式は十分実用レベルと言える．

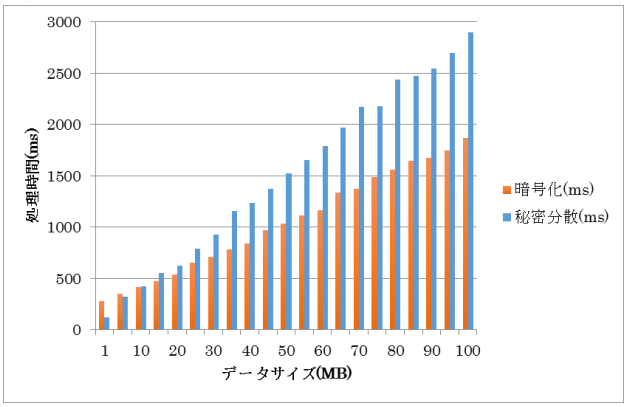

図 3.　データサイズ毎の分散処理時間

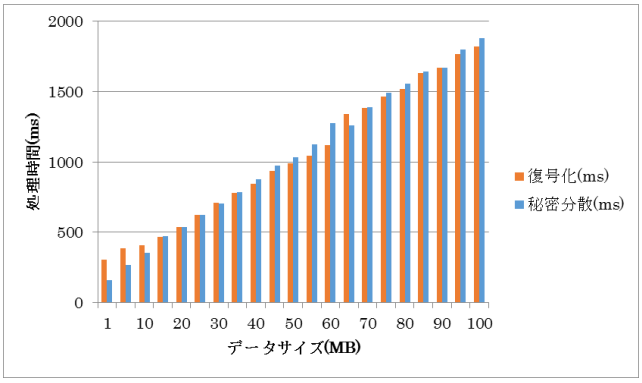

図 4.　データサイズ毎の復号処理時間

（2）クラウドストレージサービスとの通信

クラウドストレージサービスとの通信時間の測定結果について述べる．分散処理サーバからクラウドサービスへのアップロード時間は図 5，クラウドサービスから分散処理サーバへのダウンロード時間は図 6 に示す．この通信時間は 5.1 の暗号化や秘密分散の処理時間は含まれず，単純に分散処理サーバとクラウドストレージサービス間の通信時間である．A は Box，B は OneDrive，C は GoogleDrive，D は Dropbox，E は複数クラウドを利用したマルチクラウドの測定結果である．

A～D の単体のクラウドストレージサービスの通信時間は元データを暗号化も秘密分散しないで平文のままアップロードした時の通信時間を表している．また，E のマルチクラウドの通信時間は本方式の分散処理サーバ上で秘密分散されたデータを各クラウドサービスに送信したときの通信時間を表している．

この時のクラウドサービスの数を n 個，元のデータサイズを DS とするとクラウドサービス全体に保存される分散情報のサイズはおよそ n ×(DS / L)となる． 今回は，(k, L, n)しきい値秘密分散法のパラメータは k = 3, l = 2, n = 4 に設定して測定を行っているため，元のデータの

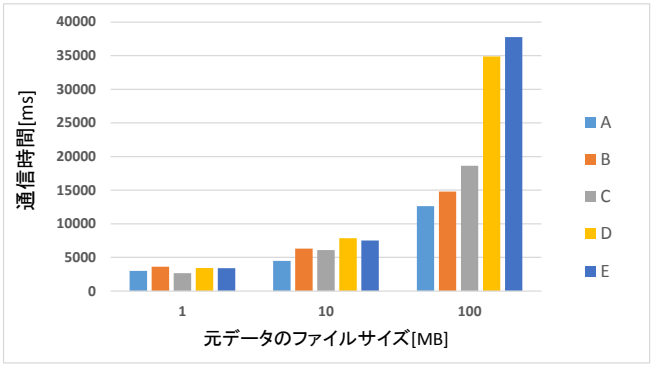

図 5.　分散処理サーバからクラウドストレージサービスへのアップロード時間

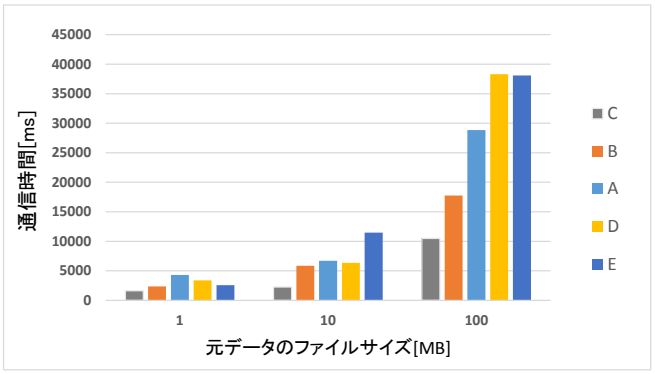

図 6.　クラウドサービスから分散処理サーバへのダウンロード時間

ファイルサイの 1 / 2 のサイズを GoogleDrive，Dropbox，OneDrive，Box の 4 つのクラウドストレージサービスにアップロードした通信時間である．

元データのファイルサイズは 1～10MB まではマルチクラウドと単体で各クラウドサービスを利用したときの通信時間はほぼ等しくなっている．元データのファイルサイズが 100MB となると各クラウドサービスで差が見られるようになった．そのため，マルチクラウドの通信時間は通信速度の遅いクラウドサービスに引っ張られて通信時間時間は増えているが，単体で利用した時の通信時間が一番遅いクラウドサービスに匹敵した通信時間なので十分実用レベルと言える．

復元処理は，4 つのクラウドサービスにアップロードした分散情報から 3 つの分散情報を集めることができれば元のデータを復元することができる．そのため今回は通信速度の速い 3 つのクラウドを静的に選択し，その 3 つクラウドサービスから分散情報をダウンロードした通信時間となっている．

ダウンロード時の通信時間もアップロード時と同様な結果となった．ダウンロード時の通信時間も十分実用レベルと言える．

６．おわりに

本研究では，セキュアなクラウドストレージシステムの方式提案と実際のクラウドサービスを複数利用して実装したシステムの性能評価を行った．

複数のクラウドストレージサービスを利用し，秘密分散，暗号化，メッセージダイジェストなどを利用することで高い可用性と高い機密性を実現し，システム全体として有効な方法であることを証明することができた．

今後の課題として秘密分散による分散情報の復号時に複数のクラウドストレージサービスから静的に選択して分散情報のダウンロードを行っているが，複数クラウドサービスから動的に最適なサービスを選択する機構が必要となる．また，本稿では特に取り上げていなかった各クラウドストレージサービスとの SSO 機能やの検討が課題となる．

謝辞：**本稿の作成にあたりご協力頂いた皆様に深く感謝いたし ます．本研究は JSPS 科研費 15H02783 の助成を受けたものです**．

参考文献

1）Y. Kajiura, A. Kanai, S. Tanimoto and H. Sato, “A File-Distribution Approach to Achieve High Availability and Confidentiality for Data Storage on Multi-cloud,” Proc. of The Fith IEEE International Workshop on Security Aspects in Processes and Service Engineering (SAPSE2013), pp. 212- 217, 2013.

2）Yuuki Kajiura, Shohei Ueno1, Atsushi Kanai, Shigeaki Tanimoto, Hiroyuki Sato, “An Approach to Selecting Cloud Services for Data Storage in Heterogneous-multicloud Environment with High Availability and Confidentiality,” Proc.of IEEE ISADS2015 The First International Workshop on Service Assurance in System Wide Information Management(SASWIM2015), pp. 205-210, March 2015.

3）Atsushi Kanai, Naoya Kikuchi, Shigeaki Tanimoto, Hiroyuki Sato, “Data Management Approach for Multiple Clouds using Secret Sharing Scheme,” 8th International Workshop on Advanced Distributed and Parallel Network Applications (ADPNA-2014), pp. 432-437, 2014.

4）H. Sato, A. Kanai, S. Tanimoto, “A Cloud Trust Model in Security Aware Cloud,” Proceedings of 10th International Symposium on Applications and the Internet (SAINT 2010), pp. 121-124, 2010.

5）H. Sato, A. Kanai, S. Tanimoto, “Building a Security Aware Cloud by Extending Internal Control to Cloud,” Proc. 10th Int'l Symposium on Autonomous Decentralized Systems (ISADS 2011), pp. 323-326., 2011.

6）A. Shamir,“How to share a secret,”Common. ACM, Vol.22, No.11, pp.612-613, 1979.

7）G. R. Blakley,“Safeguarding cryptographic keys,”Afips 1979 Nat. Computer Conf, Vol. 48, Afips Press, pp.313-317, 1979.

8）G. R. Blakley,“Security of ramp schemes,”Crypto’84, pp.242-268, 1984.

9）山本博資，“(k, L, n) しきい値秘密分散システム，”電子通信学会論文誌, vol.J68-A, no.9, pp.945-952, 1985.

10）栗原淳，清本晋作，福島和英，田中俊昭，“排他的論理和を用いた高速な (4, n) 閾値秘密分散法と (k, n) 閾値法への拡張” ISEC2007-4, pp.23-30 (2007).

11）高荒亮 ，岩村恵市，“XOR を用いた高速な(k, L, n) ランプ型秘密分散法に関する研究，” コンピューターセキュリティシンポジウム 2009 B9-3, 2009.