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

2006－DBS－138（19）
2006－GN－58（19）
2006／1／27

端末分散型ファイル共有システムにおけるアクセス可用性の検討

鷲尾 知暁　荒金 陽助　塩野入 理　金井 敦
日本電信電話株式会社 NTT 情報流通プラットフォーム研究所

概要：我々は，ファイルの機密性を確保しつつ，負荷やリスクの低減を実現するために，秘密分散を適用した端末分散型のファイル共有システムの検討を行っている．本論文では，ネットワーク上の端末にシェアを格納することにより生じるシェア所在情報の管理の問題について言及し，これを解決するための方法として，ブローカレスシステムでのシェア収集方式について提案を行う．また，端末の起動･停止に伴うファイルへのアクセス可用性の低下について検討を行い，可用性を確保･維持するためのシェア配置･再配置方式を提案する．

Data Availability in a Distributed File Sharing System

Tomoaki Washio,　Yosuke Aragane,　Osamu Shionoiri,　Atsushi Kanai
NTT Information Sharing Platform Laboratories, NTT Corporation

Abstract　To design a secure file sharing system, we have examined a distributed file sharing system using (k, n) threshold secret sharing scheme. In this paper, we discussed about the share management method. To avoid the illegal share stealings, we proposed a share collection method with brokerless file sharing system. Also, we proposed a share relocation method for increasing the access availability of shared files. Finally, we discussed the effectiveness of our proposal methods.

1. はじめに

企業における情報共有を目的として，企業内ネットワークなどを利用したファイル共有が広く行われている．ファイル共有は，情報の有効利用を促進するが，情報の機密性を低下させる危険性を有している．特に近年，企業からの情報漏洩が社会的な問題となっており，オフィス環境での情報の取り扱いが重要視されている．

従来から広く利用されているファイル共有手法としては，ファイルサーバを配置した集中管理型が挙げられるが，処理負荷や情報漏洩リスクがサーバに集中する問題がある．また，ファイルサーバは，運用コストも求められる．これに対し，近年では，端末分散型（P2P）のファイル共有方式の研究，及び実用化が進んできている．これらは，集中管理型システムの欠点であった負荷集中を解消するとともに，維持･管理コストの削減が期待できる．しかし，端末に情報を格納した場合，保管情報の機密性を如何に確保するかといった問題や，ブローカが存在しない中で統一的なアクセス制御が実現できるかなどの問題を残している．

我々は，格納情報の機密性を確保しつつ，負荷やリスクの分散を実現するために，秘密分散を適用した端末分散型のファイル共有システムの検討を行っている．保管状態での機密性を確保するために，各端ファイルを，秘密分散法[1]により分割した断片情報（シェア）として各端末に格納することとした．

本論文では，このような端末分散型ファイル共有システムを実現する際に課題となる，シェア所在情報の秘匿化とアクセス可用性の向上について検討することを目的とする．以下，秘密分散法を用いた端末分散型ファイル共有システムについて説明し，本論文における課題である，シェア所在情報，及びアクセス可用性について述べ，これらの課題を解決するための手法を提案する．

2. 秘密分散法によるファイル共有

2.1. (k, n)閾値秘密分散法

秘密分散法は, 秘密情報から複数の断片情報（シェア）を生成する. 各シェアからは元の秘密情報を類推することはできないため, 分割したシェアを複数の場所で分散して管理することにより, 保管状態での情報の機密性を確保できる.

秘密分散法には, 秘密情報の復元に生成した全てのシェアを必要とする満場一致法, 生成したシェア（n）のうち所定数（k）のシェアから元の秘密情報を復元することが可能な（k, n）閾値法などがある[1-3]. （k, n）閾値秘密分散法では, n-k を大きくとることにより冗長性を高めることが可能である.

さて, オフィス環境において端末分散型ファイル共有システムを利用するケースでは, シェアの破損などの障害は一定の確率で起こりえると考えられる. また, シェアをサーバ外の端末に格納することを想定しているため, 端末の故障だけでなく, 日常的な端末のシャットダウンによる停止も考えられる. そのため, 本論文では, 冗長性を表現できる（k, n）閾値秘密分散法採用する. 以下に（k, n）閾値秘密分散法の概略について説明する.

（k, n）閾値秘密分散法では, 秘密情報 s, ランダムな係数 r_i を用いて k-1 次多項式

$$f(x) = \sum_{i=0}^{k-1} r_i x^i$$

を作成する. ここで, $r_0 = s$ とする. 作成した k-1 次多項式を用いて n 個の異なる x_j に対し,

$$w_j = f(x_j)$$

を計算する. (x_j, w_j) の組を 1 つのシェアとし, n 個のシェアを生成する. $f(x)$ は k-1 次の多項式であるため, k 個の (x_j, w_j) の組から各係数を求めることが可能であり, $f(x)$ を特定することができる. 秘密情報 s は, 特定した $f(x)$ をもとに $s = f(0)$ を計算することにより求めることができる.

2.2. 秘密分散法を利用したストレージシステム

閾値秘密分散法をストレージシステムに適用した技術, 及び製品が存在する[4,5]. これらはサーバベースのシステムであり, 保管時の機密性確保とともに秘密分散法の冗長性を活かし, 複数のサーバにシェアを分散格納することにより, サーバ障害によるデータの破損･損失に対し強靭なストレージを構築することを目的としている. また, 端末分散型のアプローチをとるファイル共有システムについても研究が行われている[6]. ここでは, ブローカを擁するハイブリット型の端末分散型ファイル共有システムについて概要を説明する[7].

このシステムへのファイル格納は, （k, n）閾値秘密分散法により分割したシェアをネットワーク上の端末に分散格納することにより実現されている. また, システムからのファイル取り出しは, 各端末から k 個のシェアを収集し復元処理を行うことにより実現されている. このとき, 所望のシェアがどの端末に格納されているかを知る必要があるが, シェア所在情報をブローカで管理することにより, これを解決している. ファイルからシェアへの分割処理, シェアからファイルへの復元処理は各端末で行っており, シェアの受け渡しについても各端末間で直接通信することにより負荷分散を実現している. ブローカはシェア所在情報を一元管理し, 端末がシェア分散を行う際の格納先候補情報の提供, 及びシェア収集を行う際の収集元情報の提供などの役目を果たす.

各端末にはシェアの形で情報が格納されているため, k 台未満の端末から不正にシェアが流出したとしても, 元の情報が外部に漏れることはなく, 保管状態での情報の機密性を高めることが可能である.

2.3. シェア所在と可用性

本節では, 従来の端末分散型ファイル共有システムにおける 2 つの課題について述べる.

ハイブリッド型システムでは, システム中で共有される全ファイルの識別子と当該ファイルのシェアがどの端末に格納されているかを示すシェア所在情報との対応付けをブローカで一括管理していた. シェア所在

情報はファイルを復元するのに必須の情報である．必要数（k 個）のシェアを取得することでファイル復元が可能となるため，システムのセキュリティを確保するためにはシェア所在情報についても安全に管理する必要がある．しかし，全てのシェア所在情報がブローカで管理されている状況では，ブローカへの局所的な攻撃やブローカ（サーバ）管理者の不正などにより，全てのシェア所在情報が一度に流出してしまう危険性がある．シェア所在情報が漏れた場合，システムに格納されている情報が直接露呈することはないが，不正なシェア取得に繋がる可能性もあり，システムのセキュリティを維持するためにも対策が必要である．

2 つ目の課題として，システムに格納されているファイルへのアクセス可用性の問題が挙げられる．可用性(availability)とは，一般的にシステムの壊れ難さを意味し，ディスク故障やサーバダウンなどの障害の発生の頻度，及び障害発生時に復旧までに要する時間などをもとに数値が算出され，システム利用の安定性を示す指標として用いられる．一般的な可用性に関する議論では，発生頻度の低い故障等を対象として，長期間のシステム運用における可用性について議論されるが，本論文では，日常的に行われる端末の起動・停止に伴うファイルへのアクセス可能性を可用性と定義し議論する．

端末分散型のシステムでは，オフィス環境等における各ユーザが使用している PC へのシェア格納を想定している．オフィス環境等で稼動している PC は，日常的に電源の on/off が行われ，停止状態になることもあるため，必要なシェアを格納している端末が停止している場合には，復元必要数（k）のシェアを収集できず，所望のファイルにアクセスできないこともありえる．利便性の観点から，日常的な端末の停止による可用性の低下を最低限に抑えるための検討が必要である．

そこで本論文では，従来システムでの課題であったシェア所在情報の管理の問題を解決することを目的として，システムのブローカレス化について検討を行う．さらに，端末分散型システムでの問題であった端末の停止に伴う可用性の低下を抑制するためのシェア配置・再配置方式についても検討を行う．

3. 提案方式

3.1. シェア所在の隠蔽

ハイブリッド型の従来システムでは，シェア所在情報をブローカで集中管理していたため，当該ブローカがセキュリティ上のボトルネックとなっていた．そこで，システムをブローカレスとして構成し，シェア所在情報の管理を必要としないシェア収集方式を採用することにより，シェア所在情報の漏洩の可能性を排除する．また，シェア収集のシーケンスに認可判定を組み込むことにより，不正ユーザ/端末に対してシェアの所在を隠蔽する．

提案方式では，システムからファイルを取り出す際に，以下に示すフローに基づきネットワーク上の端末からシェアを収集する（図 1）．

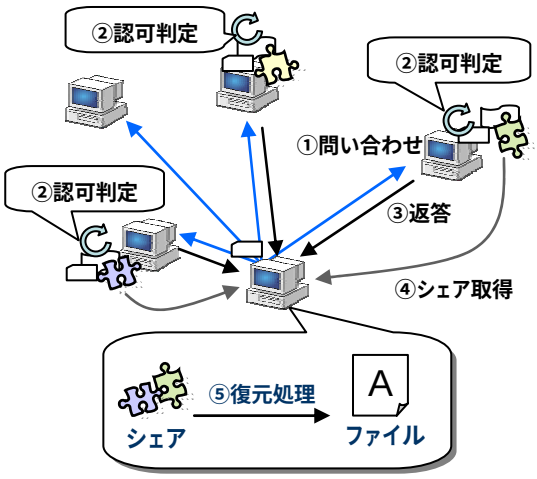

図 1 ファイル取り出し（シェア収集）

① システムからのファイル取り出す．つまり，シェア収集を行う端末は，事前にオーナから渡されたハッシュ値を識別子に他端末に対しシェアの問い合わせを行う．このとき，要求元のユーザ情報等を問い合わせに付加しておく．

② 問い合わせを受け取った端末では，指定されたハッシュ値をもとに自端末に当該ハッシュ値と対応付けられて管理されているシェアが存在するかを確認し，存在する場合は，当該シェアに付与されているポリシに基づき，要求の正当性を確認するための認可判定を実施する．また，該当シェアが存在しない場合は，問い合わせを無視する．

③ 要求の正当性が確認された場合には，シェアを保有している旨を要求元に通知する．正当性が確認できない場合は，問い合わせを無視する．

④ シェアの存在通知を受けた要求元端末は，通知元の中のk台の端末に対し，シェアの提供を改めて要求しk個のシェアを収集する．

⑤ 集まった k 個のシェアをもとに復元処理を行い，所望のファイルを得る．

3.2. 可用性の確保

(k, n)閾値秘密分散法により分割したシェアを各端末に分散格納する場合，当該ファイルに関するシェアがその時点で起動している端末に k 個以上存在することが，ファイルへのアクセスを可能とする条件，つまり，可用性を維持するための条件となる．仮に，同一ファイルのシェアを 1 台の端末に複数格納することを制限した場合，可用性を維持するための最も単純な方法は，ファイルの分割数 n をシステムを構成する端末数と同数にし，全端末にシェアを格納することである．この場合，最低 k 台の端末が起動していれば，確実にファイルへのアクセスが可能となる．さらに，復元必要数 k を小さく設定する(n-k を大きくとることにより冗長性を持たせる)ことにより，可用性を高く保つことが可能である．しかし，Shamir の(k, n)閾値秘密分散法では，シェアのサイズは元ファイルのサイズと同等か，それ以上となるため，単純に全端末にシェアを格納した場合，サイズ fsize のファイルを格納するためには，システム全体で〔fsize × (全端末数)〕以上の領域を必要とすることとなり，ストレージの使用効率は極めて悪くなる．

そこで，オフィス環境において各ユーザが使用している PC のように常時起動を仮定できない端末を格納領域として用いること想定し，シェア分割数 n が小さい場合でも，できるだけファイルへのアクセス可用性を維持する方式について検討する．

システムへのファイル格納時に行われるシェアの分散(初期分散)において，その時点で起動している端末からランダムにnヵ所を選びシェアを配置した場合，配置されたシェアは固定的に当該端末に格納され続けるため，ファイルへのアクセス可用性は，初期分散時に選ばれた端末の起動状態に依存することになる．そこで提案方式では，最適なシェア配置先の選別，及びシェアの動的な再配置を実施することにより，可用性を高める配置状態を作り出すことを考える．

提案方式では，次の 2 つの前提を設ける

前提 1. シェア分散時に可用性を優先的に確保する端末(優先端末)を指定し，分散するシェアのメタ情報としてこの情報を付与する．

前提 2. システムに参加している各端末の起動･停止タイミングは，日によって大きく変化することがなく，比較的一定であるとする．

オフィスにおけるファイル共有においては，ファイルの重要度や関わっているプロジェクトなどに応じて，ファイルにアクセス可能なユーザを限定することが行われている．従って，端末分散型ファイル共有システムにおいても，ネットワーク内の全てのユーザの可用性を高めるのではなく，特定のユーザの可用性を考慮すれば良いこととなる．本論文では，各ユーザは特定の端末を使う環境を想定して，特定の端末の可用性について検討する．また，オフィス環境で使用される端末は，起動･停止タイミングがある程度パターン化されると考えられる．本システムは，オフィス環境での利用を想定していることから，この特徴を有効活用することとした．

提案方式では，初期分散時，シェア受け取り時，停止時の各場面における動作ルールとして，以下を定義する．

① 初期配置時：優先端末の起動以前から起動している可能性の高い端末にシェアの初期配置を行う

② シェア受け取り時：シェアを受け取った端末は優先端末の起動よりも前から起動している可能性の高い端末へのシェアの再配置を検討し，必要に応じて再配置を実施する

③ 停止時：優先端末が起動している場合は，その時点で起動している他端末に対しシェアのレプリカの再配置を実施する．また，レプリカを保持していた場合は，他端末にレプリカの再配置を行った後，自端末中のレプリカは削除してしまう．

提案方式では, 指定された優先端末に応じて, 初期分散時のシェア配置先の選別や, 再配置時の配置先の選別を行う. 配置先を選別する際の判断材料として, 各端末が知りうる周辺端末の起動･停止パターンの情報を用いる. 各端末は, 起動･停止時に他端末に対し自らの起動･停止を通知することとし, また, 起動通知を受け取った端末は既に自らが起動していることを返答として通知元の端末に返すこととする. これにより, 各端末は, 周辺の他端末の起動状況を把握することができる. これらの情報を蓄積し日々更新することにより, 他端末の起動状況の傾向を把握する. シェア初期配置時や再配置時の配置先選別では, これらの情報を参照し, 優先端末の可用性を高めると思われる端末を選別する.

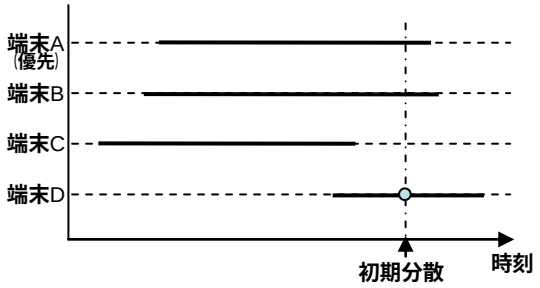

図 2 端末の起動状況

まず, 初期分散時の動作について説明する. 例として, 一日の端末の起動状況が図 2 のタイムテーブルで表わされるとし, 優先端末を“端末 A”として, “端末 D”が矢印で示した時間に k=3, n=3 のシェアを分散する場面を考える. ここで“端末 D”は, 日々の周辺端末の起動状況の把握により, 図中の網掛けのない部分に相当する情報を持っているとする. “端末 A”を優先端末として 3 ヶ所のシェアの配置先を考える場合, 優先端末である“端末 A”は自分の起動よりも前から起動している傾向があり, “端末 A”の起動以前から起動している可能性のある端末として“端末 A, B, C”を考える. しかし, “端末 C”は現在起動していないことから, 3 ヶ所の配置先として“端末 A, B, D”を選択しそれぞれにシェアを送信する.

次にシェア受け取り時の動作について説明する. 前述の初期配置後を例にとると, “端末 D”からシェアを受け取った“端末 A, B, D”がそれぞれ再配置の要/不要を判断することになる. “端末 A, B”は, 日々の周辺端末の起動状況の把握により, ともに優先端末である“端末 A”の起動以前から自分が起動している傾向があると判断し, 受け取ったシェアを自ら保持する. 一方, “端末 D”は, 自分が優先端末である“端末 A”の起動以前に起動する傾向がないと判断し, 他の“端末 A, B, C”への再配置を考える. “端末 A, B”は既にシェアを保持していることから, 翌日“端末 C”が起動している時間帯で再配置が行われ最終的に“端末 C”で保持される.

最後に, 停止時の動作について説明する. 前述の例において“端末 A, B, C”でシェアが保持された後に“端末 C”が停止する場面を例にとる. “端末 C”は優先端末である“端末 A”がまだ起動していることから, 自らが保持しているシェアのレプリカを他端末への再配置する必要があると判断する. “端末 A, B”は既にシェアを保持していることから, 最終的に“端末 D”にレプリカが配置されることになる.

4. 考察

本論文で提案した端末分散型のファイル共有システムについて, その有効性を議論する.

4.1. シェア所在の隠蔽

ブローカレスシステムとした場合, ファイル復元のためにシェア収集を行う際には, 必要なシェアの所在を如何にして知るかが問題となる. ブローカレスシステムにおいてシェア所在情報を管理する方法としては, システムに参加している端末の中から任意の端末を選び, ハイブリッド型システムにおけるブローカと同等の役割を果たすスーパーノードとして機能させる方法や, P2P におけるファイル検索手法として用いられる DHT (分散ハッシュテーブル) を応用し, ファイル識別子とシェア所在情報を複数端末で分担して管理する方法等が考えられる. しかし, これらの方法においてもシェア所在情報がスーパーノード端末に集中的に存在, もしくは各端末に分散して存在することとなるため, 当該情報の流出の可能性は否定できない.

そこで本論文では，シェア所在に関する情報の管理をシステムとしては行わない手法を採用した．その効果を以下に示す．

① ファイルと当該ファイルの復元に必要なシェアの所在の対応を管理する必要がなく，システムにおいてシェア所在情報が存在しないため，端末への攻撃等によるシェア所在情報の不正流出は起こりえない．

② シェア所在を確認する問い合わせに対して，シェアを保管している各端末は認可判定を行い，問い合わせの正当性が確認できない限り応答を返さない．仮に，認可判定の結果を返すとした場合，例え判定結果が「不許可」であっても，判定結果を返してきた端末にシェアが存在することが露呈してしまう．しかし，正当性が確認できない限り返答を返さないことにより，権限を持たないユーザ/端末に対しては，シェアを提供しないだけでなく，シェアの存在を間接的に知られてしまう情報すら提供することがない．これにより，不正ユーザに対しシェア所在を隠蔽することが可能となる．

これにより，2.3 節で挙げたシェア所在情報の管理に伴う問題を解決することが可能となる．

4.2. 可用性の確保

提案方式として定義した動作ルールに基づき，シェアの初期分散，再配置が行われることにより，優先端末の起動以前から起動している端末までシェアが伝播され，これらの端末で保持される．さらに，これらの端末が優先端末よりも早く停止した場合でも，他の端末へのシェアのレプリカの再配置が行われることにより，優先端末が起動している時間帯ではファイルの復元に必要な k 個以上のシェアを収集できる可能性が高くなる．その結果，優先端末のファイルへのアクセス可用性を確保することが可能となる．また，可用性を確保するために全端末にシェアを配置する場合に比べ，ストレージ使用量が少なく，効率がよいといえる．

5. まとめ

本論文では，秘密分散法を用いた端末分散型ファイル共有システムについて検討を行い，ネットワーク上の端末にシェアを格納することにより生じるシェア所在情報の管理の問題について言及し，これを解決するための方法として，システム構成のブローカレス化とブローカレスシステムでのシェア収集方式について提案を行った．また，同じく端末をシェア格納領域とすることにより生じる端末の起動･停止に伴うファイルへのアクセス可用性の低下について検討を行い，可用性を確保･維持するためのシェア配置･再配置方式を提案した．

今後は，提案したブローカシステムでのシェア収集方式，及びシェア配置･再配置方式のフィージビリティの確認を行う予定である．

参考文献

[1] Shamir,A: “How to share a secret,” Commun.ACM, Vol.22, No.11, pp.612-613 (1979)

[2] G.R.Blakley: “Security of Ramp Scheme,” Proc. Of Crypt’84, Lecture Note in Computer Science, vol.196, pp.242-268 (1984)

[3] 尾形わかは: “秘密分散共有法とその応用,” 情報処理学会誌, Vol.82, No.12, pp.1228-1236 (1999)

[4] 宮本: “秘密分散共有法を利用した自律分散ストレージシステム,” 信学会論文誌，Vol.J87-D-Ⅰ，No.10, pp.899-906 (2004)

[5] トラステッドソリューションズ Web ページ http://trusted-solutions.jp/

[6] 鹿島: “PC 共有による安全で低コストな P2P 分散ファイルシステムの提案,” DICOMO2004, pp.261-264 (2004)

[7] 大嶋: “安全･便利な情報流通を実現する分散型セキュアファイル共有システム,” DICOMO2005, pp.285-288 (2004)