# 情報紘——情報の安全な共有方法に関する一考察

諸橋 玄武　　小田 哲　　福永 利徳　　冨士 仁

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
〒 180-8585 東京都武蔵野市緑町 3-9-11

{morohashi.gembu, oda.satoshi, fukunaga.toshinori, fuji.hitoshi}@lab.ntt.co.jp

**あらまし**　複数のシステムやネットワーク上に散在する情報を適切に共有するためのアクセス制御方法について検討する．特に，多くの人が情報を共有し編集する場合，複数の作成者や複数の利用者のそれぞれのポリシーに合致した情報の共有が望まれる．本研究では，情報の作成者（送り手）と利用者（受け手）の双方が情報に対してアクセス制御を行える手段を提案する．提案方式によって，送り手にはデータの二次的流通や不正利用の防止といったメリットが得られ，受け手には SPAM や誤情報を受け取ることを抑制することができる．

# A note on secure information sharing systems

Gembu Morohashi　　Satoshi Oda　　Toshinori Fukunaga　　Hitoshi Fuji

NTT Information Sharing Platform Laboratories, NTT Corporation
3-9-11, Midori-Cho Musashino-Shi, Tokyo 180-8585 Japan

{morohashi.gembu, oda.satoshi, fukunaga.toshinori, fuji.hitoshi}@lab.ntt.co.jp

**Abstract**　We discuss the security of information sharing over various systems, then we propose an information sharing scheme that employs cryptograph and signature for providing confidentiality and integrity. Our proposed scheme provides access controls to both senders and receivers. This scheme protects fraudulent partial use of contents and cares receivers from spam.

## 1　はじめに

ブロードバンドの普及等によって情報インフラの整備が進み，近年ネットワーク上で容易にアクセスできるストレージサービスが普及し，従来特定のネットワーク領域に閉じて行われていたようなグループワーク [1, 2] が，インターネット越しに行われる機会が増えてきている．一方で，特定の領域に閉じないような環境では，不特定多数によるアクセスも許容してしまうため，通常は認証を行う等何らかのアクセス制御を伴っている．しかしグループワークの対象となる情報が複数のシステムに分散してしまっているような場合，作業者のアクセス制御をすべてのシステムに設定しなければならない．プロジェクトが大規模になるにつれてシステムの数が増えると，これらの設定が煩雑になったり，プロジェクトが終了後に設定を元に戻すことを忘れてしまう等アクセス制御の管理が十分にできなくなることが懸念される．

また，近年はブログ，SNS や動画投稿サイトの流行により，個人による情報発信が劇的に増加し，情報爆発や情報洪水が生じている情報過剰時代 [3] とも言われている．ネットから様々な情報を瞬時に手に入れることが容易になる一方で，その情報の正確さや真正性がどうであるか，という問題がある．必要な情報を効率よく手に入れるために検索エンジンや掲示板，SNS

等を利用して必要な情報を探し出すといったことがよく行われるが，これらを駆使して情報を探し出しても，二次的な情報であったり編者の主観が入ってしまった情報によって，事実と乖離した情報を信用してしまうことにもなりかねない．

本研究では，このような状況においても適切に情報を管理するために，情報の作成者（送り手）と利用者（受け手）の双方が情報に対してアクセス制御を行うために，暗号によるファイルの保護と署名によるファイルの信頼性の確保を実現する方法を検討した．

## 2　情報とその作成者との関係

情報をやりとりする場合，情報の送り手と受け手が必ず存在し，送り手が作成（あるいは複製や改変）した情報を受け手が取得する．グループワークのように，1つのファイルを複数の人で共有し（同時に）編集するような場合であっても，各人の編集・改変部分ごとに切り出して考えると，送り手と受け手が情報をやりとりしているモデルに置き換えて考えることができる．

**送り手による選択**　送り手が情報を渡すとき，まず誰にどれだけ渡すかということを考える．特定の誰かに渡す場合，既存の方法としては，いわゆる開示制御（アクセス制御）を行うといった方法がある．渡す相手が大勢になったり，ファイルが置いてあるシステムが複雑になってくると，それに伴って開示制御の方法も複雑になってくる．図1のように，複数のシステムが別々に開示制御を行っているような場合，それぞれのシステムにあるファイルの開示制御をそれぞれのシステムに合わせて行わなければならない．例えばあるプロジェクトのメンバーが変更になった場合，そのプロジェクトで利用しているシステムのアクセス制御リストをすべて変更しなければならない．このような課題を解決する方法として，Liberty Alliance [4] のようなシングルサインオンと呼ばれる仕組みを利用し，メンバーの変更を一元管理できるような方法も検討されている．

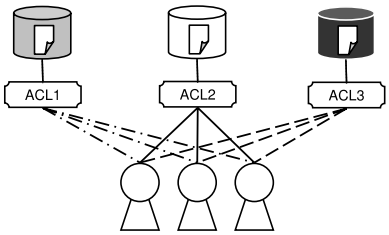


図 1: 一般的なアクセス制御

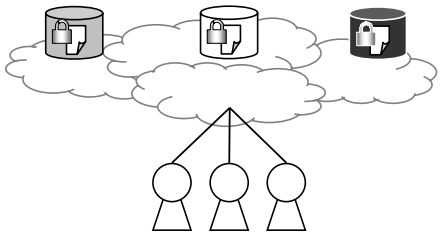

図 2: 提案方式

**受け手による選択**　受け手が情報を得るとき，その情報が正しいものか，信用できるものかどうかを判断するには，受け取った情報だけでなく，その送り手が誰かということも大きく影響する．例えば，電子メールで送られてきた情報は，誰から来たメールであるかを確認することによってその内容の妥当性を判断する．また，Web サイトに掲載されている情報も，全く同じ情報であっても，どの Web サイトにあるかによって中身の信頼性が変わってしまうことが多い．このように，情報は情報の中身だけの信頼性だけでなく，その情報を誰が作成したか，そしてその送り手の評判はどうなのかによって，その情報の信頼性が変わり得ると考えられている．例えば藤村ら [5] は，送り手の評価値によって文章の評価値が変化し，さらにその文章への評価が送り手の評価値にフィードバックされる仕組みを提案している．

本稿ではこれらを踏まえ，情報の真正性を作成者の評判に密着したものであると考え，これが結びつくような方式を提案する．提案方式では，取り扱う情報の秘匿性，完全性およびプライバシーを考慮する．すなわち，

1. データの開示範囲は，そのデータを編集

した人全員が合意した範囲でなければならない.（秘匿性）

2. データがどれほど信頼できるかについては，そのデータを編集したエンティティ全員の信頼性に依存するため，誰がどの部分を書いたかを明らかにしなくてはならない.（完全性）

3. 編集した履歴も含めて，プライバシーの保護を望むエンティティに対しては守られなくてはならない.（プライバシー）

といったことを実現する方式を提案する.

# 3 提案方式

情報を共有して多くの人が参照し，変更を加えた場合，そのデータのアクセス権は誰が設定し，そのデータの真正性は誰が保証するのかが大きな課題となる．本稿では，この問題をシステマティックに考えるために，次のような考え方を導入する.

- あるデータを参照できるかどうかの権限は，そのデータの遍歴に関わったエンティティがそれぞれ与えた権限の AND 条件で決定される.
- あるデータが信頼に足るかどうかは，そのデータの遍歴に関わったエンティティが信頼に足るかどうかの AND 条件で決定される.

直感的には，多くの人がデータの作成に参加した場合，全員が同意しない限り，そのデータを閲覧することができず，参加者の中に信頼に足らない人がいた場合，全体としてのそのデータの真正性も保証することができない，ということを表している．以上の考えに基づき，上記を実現するようなシステムについて考察する.

## 3.1 システム設計

本システムには，4 種類のエンティティとして，ネットワークストレージ，作成者，変更者および閲覧者が存在する．ネットワークストレージは，ネットワーク上でデータを保持する機関であり要求に応じて情報（データ）を受け渡しする機能を有する．データは，作成者によって作られ，変更者の手によって内容を変更され，閲覧者によって見られる．ここで作成者は‘無’の情報の内容を変更したとみなせるので，変更者として扱うこととする.

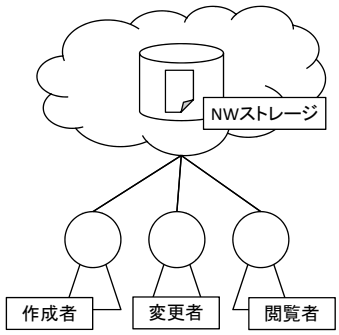


図 3: システム全体図

データは，ひとつ前の世代からの差分に関する編集履歴の組み合わせで管理される．編集履歴とは，具体的に

1. 編集したユーザの名前
2. 編集内容（追加した，削除した等の差分）
3. 一つ前の世代の編集履歴のハッシュ値

で構成される.

各ユーザ（変更者）ごとの暗号化のための鍵，署名のための鍵は事前に作成されており，それぞれの公開鍵は PKI の仕組みなどを利用して，配布されていると仮定する.

編集履歴にある編集したユーザ，編集内容がそれぞれ誰に開示されるか，また閲覧者がどのデータを受け入れるか，といった制御を行うために以下の 3 つのポリシーにまとめる.

1. [アクセスポリシー] あるデータの編集内容を見ることができる閲覧者のリスト
2. [プライバシーポリシー] あるデータを編集した変更者を知ることができる閲覧者のリスト

3. [受け入れポリシー] ある変更者が編集したデータのみ見えるようにするための変更者のリスト

アクセスポリシーおよびプライバシーポリシーはデータごとに与えられ，受け入れポリシーは閲覧者ごとに与えられる．

## 3.2 利用シーケンス

続いて，提案するシステムを用いてデータの編集や閲覧がどのように実現できるかについて，利用シーンごとに具体例を交えて述べる．

### 3.2.1 新規作成

ユーザ $A$ が新たにデータを作成したとする．

1. ユーザ $A$ が新たなデータを作成する．
2. 作成したデータを誰が見られるか，アクセスポリシーを定める．変更者となる $B, C, D$ および閲覧者となるユーザ $E, F$ が閲覧できるものとする．
3. アクセスポリシーに従って，編集内容（空語 $\lambda$ に新たなデータを追加したこと）を暗号化する．ここでは，編集したユーザ $A$ と，これから編集するユーザ $B, C, D$ および閲覧できるユーザ $E, F$ の暗号化鍵でそれぞれ暗号化する．
4. 暗号化した編集内容に対して，ユーザ $A$ の署名を付与する．
5. 変更者の情報をプライバシーポリシーに従って暗号化し，暗号化した編集内容とあわせてネットワークストレージに置いておく．

### 3.2.2 編集

ユーザ $B$ が前節でユーザ $A$ によって作成されたデータを編集したとする．

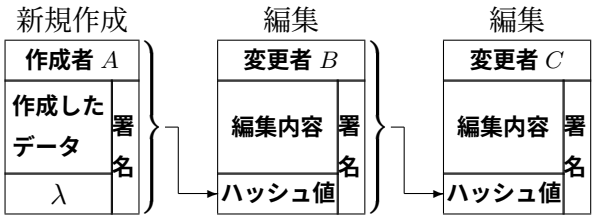


図 4: 編集履歴の推移

1. ユーザ $B$ はユーザ $A$ が作成したデータに対して，追記，削除，書き換えといった編集を行う．
2. $B$ が編集したデータを誰が見られるか，アクセスポリシーを定める．変更者 $A, B, C, D$ および閲覧者となるユーザ $E$ が読めるものとする．
3. 編集する前のデータ（暗号化された編集内容および変更者の情報）のハッシュ値を取る．
4. アクセスポリシーに従い，ユーザ $B$ による編集内容と前述のハッシュ値を暗号化する．
5. 暗号化した編集内容とハッシュ値に対して，ユーザ $B$ の署名を付与する．
6. 変更者の情報をプライバシーポリシーに従って暗号化し，暗号化した編集内容およびハッシュ値とあわせてネットワークストレージに置いておく．

ユーザ $B$ が編集したものをさらに他のユーザ $C, D$ が編集する場合も，同様にして処理され，編集するごとに編集履歴が作成される．ユーザ $A$ が最初に作成したデータに対して，ユーザ $C$ が直接編集することも可能であり，この編集履歴はユーザ $B$ が編集したものとは別の遍歴をたどる（派生する）ことになる．

### 3.2.3 閲覧

ユーザ $A$ が作成し，ユーザ $B$ が編集したデータ（の編集履歴）はネットワークストレージにそれぞれ置かれており，閲覧者（ユーザ $E, F$）は以下の処理を経てデータを閲覧する．

1. （閲覧者ごとに）編集が許容できる変更者は誰かを示す，受け入れポリシーをあらかじめ定めておく．

2. 閲覧したいデータの編集履歴に対して，古い順に以下の処理を行う．

   (a) 暗号化された変更者の情報を復号し，誰が変更したものであるか確認する．復号できない，もしくは受け入れポリシーに含まれていないのであれば，ステップ 3 へ進む．

   (b) データの編集内容とハッシュ値に付与されている署名を検証する．検証が失敗，もしくは受け入れポリシーに含まれていない変更者の署名が施されている場合は，ステップ 3 へ進む．

   (c) 暗号化されたデータの編集内容とハッシュ値を復号する．復号できなればステップ 3 へ進む．

   (d) 1 つ前の編集履歴から得られるハッシュ値と復号したハッシュ値を比較する．一致していれば編集内容を得る．そうでなればステップ 3 へ進む．

3. 得られた編集内容を順番につなぎ合わせ，データ全体を得る．

## 4 ユースケースによる検証

本節では，提案方式が冒頭で挙げた問題に対して有効であることをユースケースを用いて検証する．登場するユーザや各種ポリシーの設定は 3.2 節のものに従う．

図 5 は，データの遍歴を表している．左下の紙のマークが最初に作成されたデータでありユーザ $A$ が作成したことを表している．そこから右への矢印は，そのデータに対して $B$ が何かしら編集した結果を表している．何も制限がない（アクセスポリシーや受け入れポリシーを設定していない）場合，任意の状態のデータを誰でも手に入れることができる．

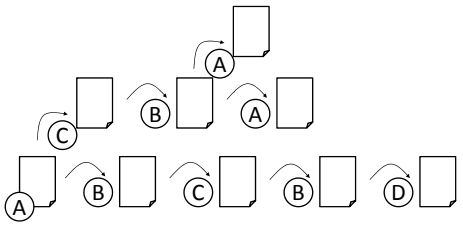


図 5: ポリシーを設定していない場合

次に，アクセスポリシーによりある程度制限されている状況を考える．$E$ がデータを閲覧しようとしているとする．$A$ と $B$ は，$E$ に対して閲覧を許可しているが，$C$ と $D$ が許可していないとする．すなわち，図 6 において破線で示される部分については，アクセスポリシーおよびプライバシーポリシーにより $E$ は見ることができない．$A$ が作成し，$C$ が編集したものをさらに $B$ が編集したデータに着目すると，$E$ は $A, B$ が作成あるいは編集した部分については閲覧できるが，$C$ が編集した部分は閲覧できない．つまり，編集履歴は $A, B$ の部分のみ得られるが，$E$ がデータ全体を復元するとき，$C$ の編集履歴が得られないため，正しく復元することが困難である．実際には，$B$ の編集履歴から（前後の文脈等を類推して）復元を試みることも可能かもしれないが，$C$ の編集の痕跡が $E$ に伝わってしまうことも懸念され，$C$ が $E$ に閲覧させたくないという意図に反してしまう．そこで，$C$ が編集したものをさらに他のユーザが編集したものであっても，$E$ には閲覧を許可しないとする方が妥当であろう．これによって $C$ と $D$ の意志が反映されたまま，データを管理することが可能となる．（秘匿性，プライバシーへの配慮を実現．）

最後に，受け入れポリシーにより閲覧者が信頼しない情報を排除する状況を考える．$F$ がデータを閲覧しようとしているとする．閲覧者 $F$ に対する制限（アクセスポリシー）はないものとし，$F$ の受け入れポリシー，すなわち $F$ が信頼する作成者，変更者は $A, C, D$ に限られているとする．図 7 に示すデータの遍歴において，$F$ は信頼できる作成者，変更者のみが携わった

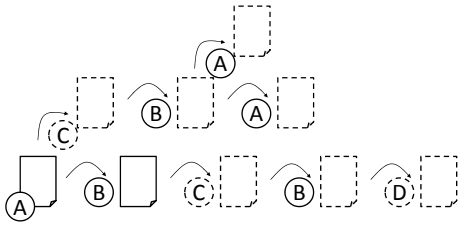


図 6: 一部だけが許可されている場合

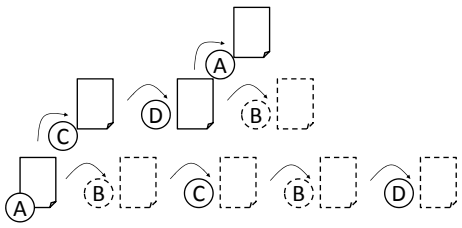


図 7: 一部だけを信用している場合

データから，閲覧するファイルを探すことができる．具体的には，図 7 の左下から上へと向かう四つのデータ（破線以外の箇所）のみが $F$ が信頼できるデータであることをシステマティックに判別できる．先ほどと同様に，信頼できない変更者が編集したものや，そこからさらに編集したものは，信頼できる変更者による編集であっても受け入れない．こうすることによって，SPAM のような不要なデータに惑わされず，探す対象を絞り込むことができる．(完全性の実現.)

なお暗号によるアクセス制限には限界がある．例えば，情報を意味がない状態に破壊することを提案方式で防ぐことはできない．このような攻撃に対しては，システム的に解決することが想定される．具体的には「保管者」のようなエンティティを想定する．保管者は暗号化された編集内容および変更者の情報，編集内容に付与されている署名を検証する．検証が失敗，もしくは受け入れポリシーに含まれていない署名が施されている場合は，最後にバックアップした状態に書き戻すことを実施する．すべての検証が成功した場合は，バックアップする．以上の処理を定期的に実施することによって，情報を破壊するような攻撃に対して対処できる．保管者は中身がどのような状態に書き換えられているかを確認する必要はなく，演算で確認することができるので，システムに組み込んで実施することも可能である．また，他のユーザに対しても許可されていない限り中身を知ることができないため，通常のバックアップよりもメリットがあると考える．

## 5 むすび

本研究では，複数のシステムに散在する情報を適切に共有するためのアクセス制御方法について検討し，暗号によるファイルの保護と保護される領域の細分化によって受け手に応じて必要な情報のみを参照できる方式を提案した．

今後の課題としては，特定の暗号方式を利用した場合のスケーラビリティの考察や，実装上の課題の検討等が挙げられる．

## 参考文献

[1] 村永, 守安: グループワークのための情報共有技術, 情報処理, Vol.34, No.8, pp.1006–1016 (1993).

[2] 中山, 真鍋, 竹林: 知識情報共有システム (Advice/Help on Demand) の開発と実践：知識ベースとノウハウベースの構築, 情処学論, Vol.39, No.5, pp.1186–1194 (1998).

[3] 秋山: 情報大爆発, 宣伝会議 (2007).

[4] Liberty Alliance Project (online), available from ⟨`http://www.projectliberty.org/`⟩ (accessed 2009-09-01).

[5] Fujimura, K., Tanimoto, N. and Iguchi, M.: Calculating Contribution in Cyberspace Community Using Reputation System "RuMoR," *Proc. of the AAMAS Workshop on Trust in Cyber-societies*, pp.40–46 (2004).