Vol.2016-DPS-166 No.6
Vol.2016-CSEC-72 No.6
2016/3/3

秘密分散法と匿名通信による秘匿性に優れたP2P型ストレージ技術の提案

福光 正幸[1]　長谷川 真吾[2]　岩崎 淳也[2]　酒井 正夫[2]　高橋 大樹[3]

概要：商用オンラインストレージサービスに保存されたユーザデータは，ターゲット広告配信や犯罪捜査の目的でサービス事業者側で解析されたり，第三者からの攻撃により改変/漏洩される恐れがある．その対策として，ユーザ側でデータを暗号化し，さらに，秘密分散法により分割して，複数のオンラインストレージに分散保存して守る技術が提案されている．しかし，強力な攻撃者から標的にされた場合，この対策だけでは十分ではない．攻撃者は，標的ユーザの通信を傍受することで分散保存先の特定が可能であり，さらに，その全ての攻撃に成功した場合は，保存データを消失させたり，暗号データを奪うことも可能である．また，奪われた暗号データが，オフラインのブルートフォース攻撃により解読される恐れもある．そこで，本稿では，秘密分散法に匿名通信技術を援用したP2P型ストレージ技術を提案する．提案法では，P2Pノードが相互に匿名通信を用いてデータを送受信して保存し合うことにより，攻撃者の通信傍受による分散保存先の特定が不可能になり，分散保存先への攻撃を困難にできる．ところで，提案法のようにデータを不特定なP2Pノード群に分散保存して秘匿する場合，一般的には，その分散保存先の情報を復元用メタ情報としてユーザ端末などに残す必要がある．しかし，そのメタ情報は格好の標的であり，攻撃者によるユーザ端末の盗難やハッキングを受ける恐れがある．そこで，本稿では，ブロックチェーンの技術を援用することで，メタ情報もP2Pストレージ上に秘匿し，ユーザ端末でのメタ情報の保存を不要にし，復元時にはユーザ名とパスワードのみを用いて，P2Pストレージ上の保存データに安全にアクセスできる仕組みも提案する．

キーワード：秘密分散法，匿名通信，P2P (Peer to Peer)，オンラインストレージ，ブロックチェーン

1. はじめに

Dropbox[1], Google ドライブ [2], OneDrive[3] などの個人ユーザを対象とした商用オンラインストレージに保存されたユーザデータは，サービス事業者によりターゲット広告配信を目的に解析されたり，公的な捜査機関により犯罪捜査を理由に合法かつ強制的に覗かれる恐れがある [4], [5], [6]．このような不特定多数のユーザを対象とする保存データの解析や覗き見には，ユーザ側での保存データの暗号化が有効な対策になりえる．しかし，暗号データとは言え，それを完全な形で一つのオンラインストレージに保存すれば，そのオンラインストレージの管理者により，オフラインでのブルートフォース攻撃を受けて解読される恐れが残る．そこで，暗号データを秘密分散法 [7] などにより複数に分割し，その分割片（以後，シェア）を異なるオンラインストレージに分散保存することで，暗号データ自体を秘匿化する技術も提案されている [8], [9], [10], [11]．

しかし，強力な攻撃者から標的とされた場合には，これらの対策でも十分とは言えない．一般的な個人ユーザが利用可能な分散保存先候補のオンラインストレージの選択肢は多くなく，また，攻撃者が標的ユーザの通信を傍受すれば，その分散保存先候補を絞り込むことも難しくない．そして，攻撃者がその分散保存先候補への全攻撃に成功すれば，標的ユーザの保存データが危険に晒されることになる．

本稿では，そのような強力な攻撃者から標的とされた場合にも耐性のある，秘匿性に優れたP2P (Peer to Peer) 型のオンラインストレージを実現する技術を提案する．提案技術は，P2Pストレージを構成するP2Pノード群が相互にTor[12]やI2P[13]のような匿名通信を用いてデータを送受信して保存し合うことで，攻撃者が分散保存先候補を絞りこむことを不可能にする．さらに，ノード数が十分に大きいと，無差別攻撃により標的ユーザの保存データを探すことも実質不可能になる．

[1] 北海道情報大学
[2] 東北大学
[3] ニッセイ情報テクノロジー株式会社
メールアドレスは共通：sss@isl.is.tohoku.ac.jp

ところで，データを暗号化と分割をして，ランダムに選択したP2Pノード群に分散保存する場合，その復元処理に必要なメタ情報（復号用パスワード，結合方法，分散保存先など）がユーザ側に残る．そのメタ情報は攻撃の格好の標的であるため，メタ情報が保存されるユーザ端末やUSBメモリなどの外部メモリは，攻撃者による盗難やハッキングの被害を受ける恐れがある．強力な攻撃者から標的にされた場合を想定すると，一般のユーザがそれを適切に防ぐのは容易ではない．すなわち，安全なP2Pストレージの実現のためには，ユーザ端末や外部メモリにメタ情報を残さないことが重要である．

そこで，本稿では，データ保存時にメタ情報もP2Pストレージに分散保存して秘匿し，データ復元時にはユーザ名とパスワードのみを用いてメタ情報を復元する技術も提案する．本研究では，中央集権的なサーバの存在は仮定せず，信頼性にバラつきがある不特定多数のノード群のみでP2Pストレージが構成されることを前提とする．すなわち，P2Pネットワークの全体の構成ノード数や個々のノードの稼働状態は，時間経過により変化し，メタ情報の保存時と復元時で異なる．提案法では，そのような前提においても，保存時に信頼性の高いP2Pノード群を分散保存先として選択しつつ，復元時にはユーザ名とパスワードのみを用いて，保存時に選択したのと同じP2Pノード群を分散保存先として再選択できる．

提案法では，その実現のために，おおよそ一定時間の間隔で継続的に作成されるストレージノードリストの存在を仮定する．このリストには，その作成日時において，他ノードに対して「ストレージ（データの保存/復元）」のサーバ機能を提供可能なP2Pノード群の情報が掲載される．そして，メタ情報の保存時には，ユーザが任意に選択した作成日時のリストと，ユーザ名とパスワードを用いて，決定論的アルゴリズムに従い，そのリストの中から分散保存先とするP2Pノード群を一意に決める．また，メタ情報の復元時には，保存時と同じ作成日時のリストと，ユーザ名とパスワードを用いることで，P2Pネットワークの状態の変動に影響を受けることなく，同じ分散保存先を選択することが可能になる．なお，メタ情報の復元を成功させるためには，リストが改変/消失しないことが重要である．提案法では，P2Pノード群が，おおよそ一定時間の間隔で継続的にそのようなリストを作成し，また，改変/消失することなく共有情報として確実に保存するために，暗号通貨Bitcoinシステムにおけるブロックチェーンの技術[14]を援用する．

ところで，前述した通り，メタ情報の復元時には，ユーザ名とパスワードだけでなく，保存時にユーザが任意選択した作成日時のリストを必要とする．リスト本体はP2Pノード群により共有情報として保存されているため，ユーザはメタ情報の保存時に選択したリストの作成日時のみを秘密情報として記憶すれば良い．なお，ユーザは，作成日時を厳密に記憶する必要はない．もしユーザが作成日時を完全に忘れてしまっても，ユーザ名とパスワードが正しければ，全てのリストを試すことで，いずれは復元に成功する．その復元処理に必要な計算コストは，時間の経過とともに増大するが，たかだか線形である．また，おおよその日時（例えば，何年何月）を覚えていれば，その日時を含む範囲を期間指定することで，復元処理を試す必要のあるリストが限定されて，計算コストを削減可能である．すなわち，リストの作成日時を記憶する負担と，復元処理時間にはトレード・オフの関係があり，ユーザは任意に調整可能である．

本稿で提案するP2Pストレージでは，そのユーザ認証用の秘密情報として，通常のユーザ名とパスワードだけでなく，リストの作成日時を補助的に用いる．不特定多数の一般ユーザの利用を想定する場合，全ユーザに秘匿性の高い安全なユーザ名とパスワードの使用を強制することは困難である[15], [16], [17]．この一般ユーザのユーザ名とパスワードが脆弱な問題は，携帯電話などを利用した所有物認証を組み合わせる二段階認証[18]で対策されるのが一般的である．提案法のように，リストの作成日時を補助的な秘密情報として使用すれば，携帯電話などの特別な機器を必要とせずに，オンラインのブルートフォース攻撃を効果的に防ぐことができる．

以降では，第2節で，提案法の主要な構成要素を定義し，第3節では，動作手順を説明する．第4節では，提案法の安全性について評価する．第5節はまとめである．

2. 構成要素

本節では，提案法で用いる主な構成要素であるP2Pノード，ユーザアカウント，ストレージノードリストについて説明する．

2.1 P2Pノード

本研究では，中央集権的なサーバを必要としない，不特定多数のP2Pノードにより構成されるP2Pストレージを想定している．個々のP2Pノードは，それぞれ，通信用のIPアドレス/ポート番号と，P2Pノード間での暗号通信と署名に用いるユニークな公開鍵/秘密鍵ペアを保持し，公開鍵のハッシュ値をP2Pノードの識別子（ノードID）として利用する．なお，データサイズ削減の必要がなければ，公開鍵をノードIDとしてもよい．

P2Pノードは，以下のサーバ機能を相互に提供し合う．

- ストレージノードリスト作成
- 掲示板（共有情報の公開）
- 匿名通信中継
- ストレージ（データの保存/復元）

なお，各サーバ機能の提供は任意であり，一切提供しな

かったり，その一部のみを提供しても良い．

2.2 ユーザアカウント

ユーザアカウントは，P2P ストレージの使用時に求められるユーザ認証用のユーザ名とパスワードにより構成される．ユーザ名はユーザが任意に設定可能であり，その作成時に他ユーザが作成済みのユーザ名との衝突確認は行わない（中央集権的なサーバが存在しないので不可能）．そのため，低い確率であるが，ユーザ名とパスワード，そして補助的な秘密情報である「データ保存時に選択したストレージノードリストの作成日時」の３つ全てが一致した場合，異なるユーザ同士でのメタ情報の復元/上書き（すなわち保存データの漏洩/消失）が起こりえる．実用上は，メールアドレスなどのユーザ固有のユニークな情報をユーザ名として使用することで，そのリスクを大幅に削減できる．

2.3 ストレージノードリスト

ストレージノードリストには，そのリストが作成された日時において，「ストレージ」のサーバ機能を提供している P2P ノード（以後，ストレージノード）群の中から，相対的に信頼性の高いストレージノードの情報（ノード ID，IP アドレス/ポート番号，公開鍵と，その署名）が，その需要に応じて一定数選抜されて，ランダムに順序付けられたリスト形式で記載される．また，リストには，識別子として，その作成日時がラベル付けされる．

リストは，「ストレージノードリスト作成」のサーバ機能を提供する P2P ノード群により，おおよそ一定時間の間隔で継続的に作成され続ける．そして，作成されたリストは，「掲示板」のサーバ機能を提供する P2P ノード群により保存/蓄積され，過去の作成分を含めて，他の P2P ノードに対して公開される．

ところで，リスト（の作成日時）の違いにより，リストに掲載されるストレージノード群の構成と，構成ノードのリスト順序が，不規則に変化するが，その不規則性が，メタ情報の分散保存先の秘匿性を向上させる上で重要である．そのため，新規作成リストには，過去に作成されたリストと比較して一定以上の不規則性を有することを，その作成要件とする．

また，P2P ストレージの保存データの復元を保証するためには，これまでに作成された全てのリストが改変/消失されずに存在し続けることが重要である．提案法では，暗号通貨 Bitcoin システムにおけるブロックチェーンの技術を援用することで，リストの改変/消失を回避する．

3. 動作手順

以下では，新規 P2P ノード作成時の初期設定，P2P ノード群の情報共有，ストレージノードリストの作成と承認，匿名通信中継，さらに，ユーザデータの保存と復元の動作手順について順に説明する．なお，前提条件として，既に十分な数の P2P ノード群により P2P ネットワークが構築され，P2P ストレージとして安定的に動作中であるとする．

3.1 初期設定

P2P ネットワークに，はじめて接続するユーザ端末は，P2P ノードとなるための初期設定が必要である．初期設定では，ユニークな公開鍵/秘密鍵ペアを作成し，通信に使用する IP アドレスとポート番号を確認し，他ノードに提供するサーバ機能を選択する．

3.2 情報共有

提案法では，中央集権的なサーバの存在を仮定しないため，P2P ノード群の情報共有には，ブロードキャスト通信を主に使用する．例えば，P2P ネットワークを構成する P2P ノード群の中で，各種のサーバ機能を提供する P2P ノードは，自身のノード情報（ノード ID，IP アドレス/ポート番号，公開鍵，提供するサーバ機能情報）を定期的かつ継続的にブロードキャスト発信して，自身が現在，サーバ機能を提供中であることを，P2P ネットワーク全体に通知する．また，全ての P2P ノードは，その動作の過程で不正な P2P ノードの存在を検知した場合，その不正ノード情報（ノード ID，IP アドレス，不正内容）をブロードキャスト発信し，P2P ネットワーク全体に注意喚起する．

そして，全ての P2P ノードは，前述のブロードキャスト通信を受信することで，各種サーバ機能を提供する P2P ノード群の情報や，危険な不正ノードの情報を共有する．また，「掲示板」のサーバ機能を提供する P2P ノード群は，これまでに発信されたブロードキャストを収集/蓄積して公開する．すなわち，P2P ノードは，自身がオフライン中に，ブロードキャスト通信により発信された共有情報も，「掲示板」を参照することで，後から入手可能である．

3.3 ストレージノードリストの作成と承認

ストレージノードリストの作成と承認の手順について説明する．はじめに，「ストレージノードリスト作成」のサーバ機能を提供する P2P ノード（以後，リスト作成ノード）が，ブロードキャスト通信や「掲示板」の公開情報を収集して解析することで，「ストレージ」のサーバ機能を提供する P2P ノード（以後，ストレージノード）群の，最新の信頼性をノード毎に評価する．そして，その得られた評価結果を基に，2.3 節で説明した形式と要件を満たすリストを，リスト作成ノード毎に独自に作成する．

つぎに，リスト作成ノードは，直近の承認済みリストと独自作成したリストを用いて，Bitcoin システムにおけるブロックチェーンのマイニング処理 [14] のような計算競争に参加する．おおよその一定時間毎に，その勝利ノードが決定され，その勝利ノードが，自身で独自作成したリストと

勝利の証拠（Bitcoin システムにおける nonce）を，ブロードキャスト発信する．ブロードキャスト通信を受信した他のリスト作成ノードは，そのリストと勝利の証拠を検証し，問題ないことを確認すると，そのリストを承認する．リスト作成ノード群の過半数から承認された段階で，そのリストが正式承認されたと実施的に見なすことができる．以上の作成と承認の手順が，おおよそ一定時間の間隔で継続的に繰り返される，

3.4 匿名通信

提案法では，P2P ノード群がユーザデータを相互に送受信し合う際に，不特定多数の中継ノードを中継する匿名通信を用いる．匿名通信の場合，各中継ノードは，直接的に通信を行う直前/直後のノード情報（ノード ID, 通信用 IP アドレス/ポート番号，公開鍵）しか把握できず，また，その直前/直後のノードが起点/終点なのかを判別できない．同様に，終点ノードは，直前のノード情報しか把握できず，また，その直前ノードが起点なのかを判別できない．そのため，起点ノードは，中継ノードと終点ノードに対して，自身のノード情報を秘匿した状態で，終点ノードとの暗号通信が可能になる．さらに，不正な P2P ノードが，起点ノードの通信傍受を局所的に行ったとしても，その全ての中継を追跡することは困難である．すなわち，標的ユーザの端末の通信傍受を行なう攻撃者に対しても，匿名通信を用いることで，シェアの分散保存先を秘匿することができる．

匿名通信を行なう場合，はじめに，「掲示板」を参照して，「匿名通信中継」のサーバ機能を提供する P2P ノード（すなわち，中継ノード）群の情報を入手する．つぎに，その中継ノード群の情報から，実際の中継に使用する中継ノードをランダムに複数選択する．そして，その選択した中継ノードを順に中継させて、終点ノードとの通信を行う．

3.5 ユーザデータの保存

ユーザデータを P2P ストレージに保存する手順は，ユーザデータ本体を P2P ストレージに保存して，その復元に必要なメタ情報を作成するまでの前半と，そのメタ情報を P2P ストレージに保存する後半に大別される．以下に，前半部分と後半部分を分けて，それぞれの手順を記述する．

ユーザデータ本体の保存とそのメタ情報の作成

(1) ユーザ端末を P2P ネットワークに接続する．
(2) 保存するユーザデータ M を確定する．
(3) ランダム作成したユーザデータ本体用ワンタイムパスワード P^M_{otp} を用いて，M を暗号化し，暗号データ C を作成する．
(4) 秘密分散法を用いて，C を N 個のシェア $c_1, c_2, \cdots, c_N$ に分割する．
(5) 「掲示板」を参照して，「ストレージ」のサーバ機能を提供するストレージノード群の情報を入手する．
(6) 入手したストレージノード群の情報から，シェア $c_1, c_2, \cdots, c_N$ の分散保存先のストレージノード $u^M_1, u^M_2, \cdots, u^M_N$ をそれぞれランダムに選択する．
(7) シェア $c_1, c_2, \cdots, c_N$ のハッシュ値を計算し，それぞれのシェアの復元鍵 $k^M_1 = H(c_1), k^M_2 = H(c_2), \cdots, k^M_N = H(c_N)$ とする．ここで，復元鍵とは，ストレージノードに対してシェアの復元を依頼する際に，シェアの正当な所有ユーザであることを証明するために用いる秘密情報である．
(8) ストレージノード $u^M_1, u^M_2, \cdots, u^M_N$ に，シェアとその復元鍵の対 $\left(c_1, k^M_1\right), \left(c_2, k^M_2\right), \cdots, \left(c_N, k^M_N\right)$ を，それぞれ匿名通信を用いて送信して，保存を依頼する．依頼を受けたストレージノードは，シェアとその復元鍵の対を保存する．
(9) ユーザデータ本体用ワンタイムパスワード P^M_{otp} と，シェアの分散保存先とその復元鍵の対 $\left(u^M_1, k^M_1\right), \left(u^M_2, k^M_2\right), \cdots, \left(u^M_N, k^M_N\right)$ を，メタ情報 E として作成する．

メタ情報の保存

(1) 「掲示板」を参照して，「ストレージノードリスト」の情報を入手する．
(2) 保存するメタ情報 E と紐付けるユーザアカウントのユーザ名 ID とパスワード PASS を任意に決定する．
(3) 入手したリストの情報から，メタ情報の保存時に使用するリストの作成日時 T を任意に選択する．なお，そのリストを $L(T)$ とする．
(4) T と ID と PASS を引数とするハッシュ値 $H(T, \mathrm{ID}, \mathrm{PASS})$ を計算し，メタ情報用ワンタイムパスワード $P^E_{otp} = H(T, \mathrm{ID}, \mathrm{PASS})$ とする．
(5) パスワード P^E_{otp} を用いて，メタ情報 E を暗号化し，暗号データ Q を作成する．
(6) 秘密分散法を用いて，Q を J 個のシェア $d_1, d_2, \cdots, d_J$ に分割する．
(7) T と ID と PASS を用いて，$L(T)$ に掲載のストレージノード群から，シェア $d_1, d_2, \cdots, d_J$ の分散保存先のストレージノード $u^E_1, u^E_2, \cdots, u^E_J$ を，決定論的アルゴリズムに従い，それぞれ一意に選択する．

例えば，リスト $L(T)$ の掲載ノード数 $|L(T)|$ を法とする，ID と PASS と，インデックス番号 $j \in \{1, 2, \cdots, J\}$ を引数とするハッシュ値に，1 を加算した

$$H(\mathrm{ID}, \mathrm{PASS}, j) \bmod |L(T)| + 1$$

番目の掲載順位のノードを，分散保存先のストレージノード u^E_j として選択するなどのアルゴリズムが考えられる．

(8) T と ID と PASS を引数とするハッシュ値 $H(T, \mathrm{ID}, \mathrm{PASS})$ を計算し，メタ情報用シェアの復元鍵 $k^E = H(T, \mathrm{ID}, \mathrm{PASS})$ とする．

(9) ストレージノード $u_1^E, u_2^E, \cdots, u_J^E$ に，復元鍵 k^E とリストの作成日時 T の対 $\left(k^E, T\right)$ を匿名通信を用いて送信して，シェアの保存を仮依頼する．

仮依頼を受けたストレージノードは，受信した復元鍵 k^E と T の組み合わせとなるシェアを既に保存していないかどうかを確認し，保存していない場合は承諾を，保存している場合は拒否の回答をする．

ユーザ端末は，ストレージノード $u_1^E, u_2^E, \cdots, u_J^E$ の中に一つでも拒否するノードが存在した場合は，T を変更して，(4) からやり直す．全ノードが承諾した場合は，$u_1^E, u_2^E, \cdots, u_J^E$ に，シェア $d_1, d_2, \cdots, d_J$ を，それぞれ送信して，保存を正式依頼する．

正式依頼を受けたストレージノード $u_1^E, u_2^E, \cdots, u_J^E$ は，そのシェア d_j と復元鍵 k^E とリストの作成日時 T の組 $\left(d_1, k^E, T\right), \left(d_2, k^E, T\right), \cdots, \left(d_J, k^E, T\right)$ を，それぞれ保存する．

(10) ユーザは，ID と PASS と T を，メタ情報の復元に必要な秘密情報として記憶する．また，ユーザ端末への盗難やハッキングのリスクを回避するために，ユーザデータ M と，その M の P2P ストレージへの保存の過程で作成されたメタ情報などの関連情報を，ユーザ端末から削除する．

3.6 ユーザデータの復元

ユーザデータを P2P ストレージから復元する手順は，3.5 節で説明した保存の手順とは逆になり，メタ情報を復元するまでの前半と，そのメタ情報を用いてユーザデータを復元する後半に大別される．以下に，前半部分と後半部分を分けて，それぞれの手順を記述する．

メタ情報の復元

(1) ユーザ端末を P2P ネットワークに接続する．

(2) 「掲示板」を参照して，「ストレージノードリスト」の情報を入手する．

(3) メタ情報の保存時に紐付けたユーザアカウントのユーザ名 ID とパスワード PASS を入力する．

(4) メタ情報の保存時に選択したリストの作成日時 T を入力する．その作成日時のリストを $L(T)$ とする．

(5) T と ID と PASS を用いて，$L(T)$ に掲載のストレージノード群から，メタ情報のシェアの分散保存先のストレージノード $u_1^E, u_2^E, \cdots, u_J^E$ を，決定論的アルゴリズムに従い，それぞれ一意に決定する．

(6) T と ID と PASS を引数とするハッシュ値 $H(T, \mathrm{ID}, \mathrm{PASS})$ を計算し，メタ情報用シェアの復元鍵 $k^E = H(T, \mathrm{ID}, \mathrm{PASS})$ を作成する．

(7) ストレージノード $u_1^E, u_2^E, \cdots, u_J^E$ に，シェアの復元鍵と T の対 $\left(k^E, T\right)$ と，ユーザ端末のノード情報（ノード ID，IP アドレス/ポート番号，公開鍵と，その署名）を，ユーザ端末を起点とする匿名通信を用いて送信し，メタ情報のシェアの復元を依頼する．

復元依頼を受けたストレージノードは，一旦，ユーザ端末を起点とする匿名通信を終了し，今度は，自身を起点とする（すなわち，ユーザ端末を終点とする）匿名通信を用いて，ユーザ端末（の IP アドレス）が復元依頼ノードであることを確認する．その確認後，依頼時に受信した復元鍵 k^E と T の組み合わせとなるシェアを保存しているかどうかを確認し，保存していた場合は，そのシェアを，ユーザ端末に送信する．

ユーザ端末は，ストレージノード u_j^E から受信したシェアを，d_j として保存する．

(8) ストレージノード群から受信したシェア $d_1, d_2, \cdots, d_J$ を結合して，メタ情報の暗号データ Q を作成する．

(9) T と ID と PASS を引数とするハッシュ値 $H(T, \mathrm{ID}, \mathrm{PASS})$ を計算し，メタ情報用ワンタイムパスワード $P_{otp}^E = H(T, \mathrm{ID}, \mathrm{PASS})$ を作成する．

(10) パスワード P_{otp}^E を用いて，暗号データ Q からメタ情報 E を復号する．

メタ情報を用いたユーザデータの復元

(1) メタ情報 E より，ユーザデータ本体用ワンタイムパスワード P_{otp}^M と，ユーザデータ本体のシェアの分散保存先と復元鍵の対 $\left(u_1^M, k_1^M\right), \left(u_2^M, k_2^M\right), \cdots, \left(u_N^M, k_N^M\right)$ を作成する．

(2) ストレージノード $u_1^M, u_2^M, \cdots, u_N^M$ に，シェアの復元鍵 $k_1^M, k_2^M, \cdots, k_N^M$ を，それぞれ匿名通信を用いて送信して，シェアの復元を依頼する．

復元依頼を受けたストレージノードは，受信した復元鍵と対となるシェアを保存している場合，そのシェアをユーザ端末に送信する．

ユーザ端末は，ストレージノード u_i^M から受信したシェアを c_i として保存する．

(3) ストレージノード群から受信したシェア $c_1, c_2, \cdots, c_N$ を結合して，ユーザデータの暗号データ C を作成する．

(4) パスワード P_{otp}^M を用いて，暗号データ C からユーザデータ M を復号する．そして，復号に成功した場合，ストレージノード $u_1^M, u_2^M, \cdots, u_N^M$ に復元鍵 $k_1^M, k_2^M, \cdots, k_N^M$ を，また，ストレージノード $u_1^E, u_2^E, \cdots, u_J^E$ に復元鍵 k^E とリストの作成日時 T を，匿名通信を用いてそれぞれ送信し，対応するシェアの削除を依頼する．

削除依頼を受けたストレージノードは，対応するシェ

アを保存していた場合，それを削除する．

4. 安全性評価

攻撃者が，P2P ストレージに保存された他人のユーザデータに対して覗き見や改変などの攻撃を試みる場合，その攻撃対象は，ユーザデータやメタ情報のシェアが保存されている P2P ネットワーク，ユーザデータやメタ情報が生成/復元されるユーザ端末，そして，ユーザが記憶するユーザ認証用の秘密情報（ユーザ名/パスワード/リストの作成日時）の３つに大別される．以下では，各対象への攻撃についての安全性を評価する．

4.1 P2P ネットワークの安全性

提案法では，ユーザデータとメタ情報を，ワンタイムパスワードで暗号化し，さらに，秘密分散法により複数のシェアに分割して，P2P ネットワーク（P2P ノード群）に分散保存する．したがって，P2P ネットワークへの攻撃により，攻撃者が特定のユーザの保存データにアクセス可能になるまでには，(1) シェアの分散保存先を特定し，(2) その分散保存先からシェアを奪い，(3) 奪ったシェアからユーザデータまたはメタ情報の暗号データを復元し，さらに，(4) その暗号データを解読するという４つの段階での攻撃を全て成功させる必要がある．以下では，それぞれの段階での攻撃の危険性について考察する．

(1) シェアの分散保存先が特定される危険性

ユーザデータ本体のシェアの分散保存先はランダム選択されるため，その秘匿性は最大である．したがって，その保存先が記録されたメタ情報を用いることなく，攻撃者が特定のユーザのユーザデータ本体のシェアの分散保存先を特定するのは困難である．

メタ情報のシェアの分散保存先は，ユーザ名とパスワード，さらに，保存時に用いたストレージノードリスト（の作成日時）の３つの秘密情報の組み合わせにより選択される．リストの多様性により，メタ情報のシェアの分散保存先も多様になる．したがって，ユーザ名とパスワードとリストの作成日時のいずれも知らない攻撃者が，メタ情報のシェアの分散保存先を特定することは困難である．

また，P2P ノード間でのシェアの送受信には匿名通信が用いられるため，攻撃者が局所的な通信傍受を行うことによる，シェアの分散保存先の特定も困難である．

(2) 分散保存先からシェアが奪われる危険性

シェアの保存先ノードから，正規の手順により，そのシェアの復元（送信）を受けるためには，そのシェアの正当な所有者ユーザであることを証明する復元鍵が必要である．

ユーザデータ本体のシェアの復元鍵は，そのシェアのハッシュ値である．そのため，シェアの実体を持たない攻撃者が，そのハッシュ値を計算することは不可能である．すなわち，その復元鍵が記録されたメタ情報を入手しない限りは，攻撃者がユーザデータ本体のシェアの復元鍵を知ることができず，そのシェアの復元を受けることも不可能である．

メタ情報のシェアの復元鍵は，ユーザ名，パスワード，保存時に用いたリストの作成日時の３つ全ての秘密情報を引数とするハッシュ値である．したがって，攻撃者が，正当のユーザと同等にユーザ認証用の秘密情報を知らない限りは，メタ情報のシェアの復元鍵を作成できず，そのシェアの復元を受けることも不可能である．

(3) 暗号データが復元される危険性

シェアを奪った攻撃者が，そのシェアを用いてユーザデータまたはメタ情報の暗号データを復元する危険性について考察する．

シェアを結合して，元の暗号データを復元するためには，秘密分散法での分割時に設定した閾値以上のシェアが必要である．すなわち，暗号データを復元させないためには，閾値以上のシェアが奪われるのを防ぐことが重要である．

提案法では，シェアの分散保存先を不特定多数の P2P ノードが選択可能である．その特性を活かし，その閾値（すなわち，暗号データの復元に必要なシェア数）を十分に大きくすることで，攻撃者が暗号データを復元する危険性を低減させることが可能である．

(4) 暗号データが解読される危険性

ユーザデータの暗号データは，ランダムに作成したワンタイムパスワードを用いて暗号化される．したがって，そのパスワードが記録されたメタ情報を入手しない限りは，攻撃者が暗号データを解読することは困難である．

メタ情報の暗号データは，ユーザ名，パスワード，保存時に用いたリストの作成日時の３つ全ての秘密情報を引数とするハッシュ値をワンタイムパスワードとして用いて暗号化される．したがって，攻撃者が，正当のユーザと同等にユーザ認証用の秘密情報を知らない限りは，暗号データを解読することは困難である．

ただし，暗号データが奪われるということは，それが乱数列のような強力なパスワードで暗号化されていたとしても，オフラインのブルートフォース攻撃により，将来的には解読される恐れが残り危険である．しかし，提案法では，メタ情報の暗号データに限れば，そのようなオフラインのブルートフォース攻撃に対しても耐性がある．ユーザデータを改変する度に，そのユーザデータの暗号データのシェアの分散保存先を変更して，メタ情報を更新することで，古いメタ情報を無効化することができる．したがって，メタ情報の暗号データが攻撃者に奪われたとしても，それが

オフラインのブルートフォース攻撃で解読される前に，メタ情報（ユーザデータの暗号データの分散保存先）を更新すれば，ユーザデータが奪われるのを回避できる．

以上の説明の通り，P2P ネットワークは厳重な 4 段階で守られており，P2P ネットワークへの攻撃により，攻撃者が特定のユーザの保存データにアクセスすることは容易ではない．

4.2 ユーザ端末の安全性

攻撃者により，ユーザ端末の盗難やハッキングの被害を受けた場合の危険性について考察する．

はじめに，ユーザ端末が盗難の被害を受けた場合を考える．ユーザ端末が盗難の被害を受けた場合，そのユーザ端末内のデータが OS レベルのユーザ認証で保護されていたとしても，一般的ユーザのパスワード強度では，オフラインのブルートフォース攻撃を受けて解読されていまう可能性が高い [19]．そのため，ユーザ端末内の保存データは，全て攻撃者に奪われると考えるのが妥当である．

ユーザ端末には，初期設定時に作成した公開鍵/秘密鍵ペアの情報と，収集済みの共有情報が保存されている．また，ストレージノードとして「ストレージ」のサーバ機能を提供していた場合，他ユーザからの依頼を受けて保存しているシェアとその復元鍵の対が保存されている．

公開鍵/秘密鍵ペアは，ユーザ自身のユーザデータやメタ情報には紐付かないため，公開鍵/秘密鍵ペアが奪われたり失われたりした場合でも，P2P ストレージに保存されたユーザデータに危険が及ぶことはない．また，共有情報は公開情報であるため，漏洩による悪影響は無く，「掲示板」を参照することで再収集が可能である．さらに，他ユーザからの依頼を受けて保存しているシェアは，秘密分散法で分割されているため，攻撃者がその一つのみを奪っても，分割前の暗号データを復元することは不可能である．

復元鍵に関しては，ユーザデータ本体のシェアの復元鍵は，そのシェアのハッシュ値であるため，その所有ユーザの情報が漏洩することはない．一方，メタ情報のシェアの復元鍵は，リスト作成日時とユーザ名を引数とするハッシュ値であり，しかも，復元鍵にはリスト作成日時の情報も組み合わされているため，オフラインのブルートフォース攻撃を受ければ，比較的簡単に，ユーザ名の情報が漏洩する可能性がある．しかし，提案法では，メタ情報のシェアの復元鍵のサイズを制限することで，ハッシュ値の衝突を意図的に起こりやすくしており，ユーザ名が明確に漏洩するのを回避する工夫を行っている．

また，攻撃者が，シェアや復元鍵を消失させたとしても，シェアは秘密分散法により冗長化されているため，そのユーザデータが直ちに復元不可能になるわけではない．そもそも，提案技術は，信頼性にバラつきがある不特定多数の P2P ノード群による P2P ストレージを想定しており，シェアの一部が損なわれることは想定の範囲内である．以上のように，ユーザ端末が盗難の被害を受けた場合の危険性は低い．

つぎに，ユーザ端末がハッキングの被害を受けた場合を考える．ユーザが P2P ストレージを使用中は，ユーザ自身のユーザデータまたはメタ情報が，ユーザ端末内に一時保存されて存在する．現状，提案法では，ハッキング対策が存在しない．そのため，そのタイミングで，ユーザ端末がハッキングを受けた場合，ユーザ自身のユーザデータまたはメタ情報が奪われる恐れがある．

4.3 ユーザ認証用の秘密情報の安全性

攻撃者により，ユーザ認証用の秘密情報（ユーザ名/パスワード/リストの作成日時）に関する攻撃を受けた場合の危険性について考察する．

本稿で提案した P2P ストレージは，一般的なウェブサービスと同様に，ユーザ名とパスワードのみを用いて，どこからでも保存データにアクセス可能である（前述した通り，リストの作成日時は必須ではない）．このようなユーザ認証方式の場合，ユーザは，自身のユーザ名とパスワードを厳重に管理する必要があるが，一般的なユーザは，ユーザ名の秘匿性が低く，また，脆弱なパスワードを使用する者が少なくない．また，攻撃者が，標的ユーザの個人情報を調べて，ユーザ名とパスワードを推測する恐れもある．そのため，一般的には，ユーザ名とパスワードだけで実現される秘匿性は十分とは言えない．

提案法では，リストの作成日時を補助的な秘密情報として使用することで，その秘匿性を向上させている．リストの作成日時は，ユーザの個人情報からは推測困難な秘密情報である．しかも，通常，リストの作成日時は毎回選択し直されるため，頻繁に変更される．すなわち，攻撃者がオンラインのブルートフォース攻撃により，標的ユーザのユーザ名とパスワードを探索する場合，膨大なリストの分だけ探索範囲が広がることになる．また，攻撃者は，リストの作成日時が再変更される前に，オンラインのブルートフォース攻撃をやり遂げる必要がある．

さらに，提案法には，オンラインのブルートフォース攻撃を検知/排除する対策もなされている．提案法では，メタ情報のシェアの復元依頼時に，復元依頼元のユーザ端末の IP アドレスと，復元鍵と，用いたストレージノードリストの作成日時の情報を依頼先ノードに通知する必要がある．攻撃者が，ブルートフォース攻撃を行う場合，使用が推定される「ユーザ名」と「パスワード」と「リストの作成日時」の全ての組み合わせで，繰り返しメタ情報の復元を試すことになる．すなわち，ブルートフォース攻撃の場合は，同じ IP アドレスの依頼元ノードから，一つの「リストの作成日時」に対して，異なる複数の復元鍵（「ユー

ザ名」と「パスワード」の違いにより変化する）を組み合わせた復元依頼が多数行われることになる．正規ユーザでも，「リストの作成日時」が不明の場合，「リストの作成日時」の全ての組み合わせで，繰り返しメタ情報の復元を試すことになるが，「ユーザ名」と「パスワード」が固定されため，一つの「リストの作成日時」に対して，１つの復元鍵での復元依頼を１回のみしか行わない．そのため，同じIP アドレスから，同じ「リストの作成日時」に対して異なる復元鍵を組み合わせた復元依頼を一定回数以上受けた場合，その復元依頼がブルートフォース攻撃によるものと検知できる．そして，ブルートフォース攻撃を検知したストレージノードが，その IP アドレスを不正ノード情報としてブロードキャスト通信で注意喚起することで，そのようなオンラインのブルートフォース攻撃を排除できる．

以上の説明のように，攻撃者がユーザ認証用の秘密情報に関する攻撃を行なうことで，ユーザデータにアクセスすることは困難である．

5. まとめ

本稿では，秘匿性の優れた P2P 型ストレージを実現する技術を提案した．提案法では，暗号化と秘密分散法を用いる従来技術に，匿名通信技術を援用し，さらに，P2P 型へ一般化することで，保存データの秘匿性を従来技術に比べて大幅に向上することに成功している．また，P2P ノード群により一定時間毎に継続的に生成され，永続的に改変/消失されない共有情報（いわゆる，ブロックチェーン）を活用することで，保存データのメタ情報まで P2P ストレージ上に秘匿し，ユーザはユーザ名とパスワードのみを用いて、P2P ストレージ上の保存データに，どこからでも安全にアクセスできる仕組みも実現した。

本稿では，一般的な攻撃に対する安全性の定性的な評価を行い，提案技術が，P2P ストレージ使用中のハッキング攻撃を除き，多様な攻撃に対して優れた安全性を有することを示した．しかし，提案法が，各種パラメータの変化により，実際にどの程度の性能を実現できるかを調べる定量的な評価は今後の課題である．

ところで，提案法では，P2P ノード群の大多数が，プロトコルに従い，正しく動作することを前提としており，それが安全性の根拠にもなっている．しかし，現段階では，正しい動作に対して報酬が支払われる仕組みが実現されおらず，不特定多数の端末で構成される P2P ノード群が，その前提通りに動作するかは疑問が残る．その前提をより確かなものにするためには，P2P 型仮想通貨の発行し，P2P ノード間で報酬授受の仕組みを組み込むなどの改良が必要である．そのための技術の開発も今後の課題である．

参考文献

[1] Dropbox : トップページ, https://www.dropbox.com/
[2] Google : Google ドライブ, https://www.google.co.jp/intl/ja/drive/
[3] Microsoft: OneDrive, https://onedrive.live.com/about/ja-jp/
[4] Dropbox : Dropbox の安全性について, 入手先 ⟨https://www.dropbox.com/ja/help/27⟩
[5] Google : Google はどのようなデータを収集していますか？, 入手先 ⟨https://privacy.google.com/data-we-collect.html⟩.
[6] Leo Kelion: *Microsoft tip leads to child porn arrest in Pennsylvania*, BBC News, available from ⟨http://www.bbc.com/news/technology-28682686⟩, (2014/9/6).
[7] A. Shamir: *How to Share a Secret*, Commun. ACM, vol. 22, pp. 612-613, (1979).
[8] 堀内 公平 : 複数ベンダーのクラウドを用いた秘密分散ストレージ「MyCloud」の開発 - 安全で高速なクラウドのある未来の為に - , 入手先 ⟨http://www.ipa.go.jp/files/000007202.pdf⟩.
[9] 福光, 長谷川, 岩崎, 酒井, 高橋 : 秘密分散法によりクラウドストレージを安全に活用する技術の実用化研究, 2014 年暗号と情報セキュリティシンポジウム (SCIS2014), (2014).
[10] 福光, 長谷川, 岩崎, 酒井, 高橋 : 秘密分散法とサーバアプリを用いた安全性と利便性を両立するパスワードマネージャの提案, コンピュータセキュリティシンポジウム 2014 論文集, vol.2014, no.2, pp. 619-626, (2014-10-15)
[11] 白山, 金井, 谷本, 佐藤 : 秘密分散法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と評価，2016 年暗号と情報セキュリティシンポジウム (SCIS2016), (2016).
[12] Tor Project: Anonymity Online, https://www.torproject.org/
[13] The Invisible Internet Project: I2P 匿名ネットワーク, https://geti2p.net/ja/links
[14] Satoshi Nakamoto: *Bitcoin: A Peer-to-Peer Electronic Cash System*, available from ⟨https://bitcoin.org/bitcoin.pdf⟩, (2008/10/31).
[15] CSID:*Consumer Survey: Password Habits*, available from ⟨http://www.csid.com/wp-content/uploads/2012/09/CS_PasswordSurvey_FullReport_FINAL.pdf⟩,(2012/9).
[16] P. Ducklin:*Anatomy of a Password Sisaster - Adobe's Giant-Sized Cryptographic Blunder*, available from ⟨https://nakedsecurity.sophos.com/2013/11/04/anatomy-of-a-password-disaster-adobes-giant-sized-cryptographic-blunder/⟩ , Neked Security, (2013).
[17] Z. Liu:*A Large-Scale Study of Web Password Habits of Chinese Network Users*, Journal of Software, vol. 9, no. 2, pp. 293-297, (2014).
[18] Google : Google 2 段階認証プロセス, https://www.google.co.jp/intl/ja/landing/2step/.
[19] D. Goodin:*25-GPU Cluster Cracks Every Standard Windows Password in <6 Hours*, available from ⟨http://arstechnica.com/security/2012/12/25-gpu-cluster-cracks-every-standard-windows-password-in-6-hours/⟩, (2012),