(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2010-61103
(P2010-61103A)
(43)公開日　平成22年3月18日(2010.3.18)

(51)Int.Cl.		F I			テーマコード（参考）
G09C 1/00	(2006.01)	G09C	1/00	610Z	5B017
G06F 21/24	(2006.01)	G06F	12/14	540A	5B075
G06F 17/30	(2006.01)	G09C	1/00	660D	5J104
		G06F	17/30	120B	
		G06F	17/30	320C	

審査請求 有　請求項の数 42　ＯＬ　外国語出願　（全 36 頁）

(21)出願番号　特願2009-128697 (P2009-128697)
(22)出願日　平成21年5月28日 (2009.5.28)
(31)優先権主張番号　200810098359.1
(32)優先日　平成20年5月30日 (2008.5.30)
(33)優先権主張国　中国 (CN)
(31)優先権主張番号　200810145083.8
(32)優先日　平成20年8月1日 (2008.8.1)
(33)優先権主張国　中国 (CN)

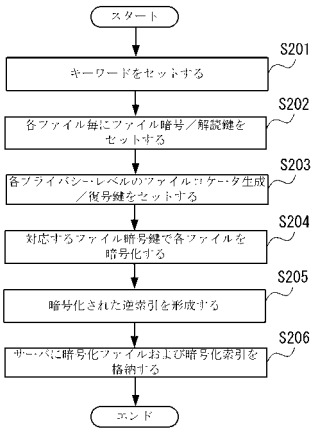
（特許庁注：以下のものは登録商標）
１．フロッピー

(71)出願人　505418870
エヌイーシー（チャイナ）カンパニー，リミテッド
ＮＥＣ（Ｃｈｉｎａ）Ｃｏ．，Ｌｔｄ．
中華人民共和国　１００００７　ベイジン，ドングチェングディストリクト，ドングシシチャオ　エー２２，ナシカング　タワー　ビー，スイート１２２２
(74)代理人　100093595
弁理士　松本　正夫
(72)発明者　レー　ハウ
中華人民共和国　１０００８４　ベイジン，　チンフア　サイエンス　パーク，　イノベーション　プラザ，　ビルディング　エー，　１１エフ

最終頁に続く

(54)【発明の名称】高速検索可能な暗号化のための方法、装置およびシステム

(57)【要約】　（修正有）
【課題】高速検索可能な暗号化のための方法、装置およびシステムを提供する。
【解決手段】データ所有者はファイルを暗号化し、また、ファイルの各キーワードに従って暗号化された索引を生成し、暗号化テキストおよび暗号化された索引をサーバに格納する。索引は、キーワードアイテムセットロケータによって各々識別されており、対応するキーワードに関連したファイルの少なくとも１つ以上のファイルロケータを含むキーワードアイテムセットからなる。ファイルロケータはそれぞれ、暗号化されたファイルの検索のための情報の暗号化テキストを含んでいる。また、正しいファイルロケータ解読鍵でのみ、暗号化テキストは解読することが可能である。データ所有者は、検索者が暗号化索引の上で検索し、かつ一定のキーワードに関するファイルを引き出すことを可能とするため、検索者にファイルロケータ解読鍵だけでなくキーワードアイテムセットロケータも発行する。
【選択図】図４



【特許請求の範囲】
【請求項１】
検索可能な暗号化方法であって、
１つ以上のファイルロケータ生成鍵をセットするステップと、
一意の値に少なくとも１つのキーワードを含む文字列をマッピングすることにより、１つ以上のキーワードアイテムセットロケータを生成するステップと、
少なくとも１つのファイルロケータ生成鍵で複数のファイルのファイル取得情報を暗号化することにより、１つ以上のファイルロケータを生成するステップと、
前記キーワードアイテムセットロケータによって識別され、対応するキーワードに関連したファイルの少なくとも１つ以上のファイルロケータを含む１つ以上のキーワードアイテムセットによって暗号化索引を形成するステップと
を有することを特徴とする検索可能な暗号化方法。
【請求項２】
前記各ファイルについてファイル暗号鍵をセットするステップと、
対応するファイル暗号鍵で各ファイルを暗号化するステップとをさらに含むことを特徴とする請求項１に記載の検索可能な暗号化方法。
【請求項３】
前記ファイル取得情報が、少なくとも１の暗号化されたリソース識別情報および前記ファイルのファイル解読鍵を含むことを特徴とする請求項１に記載の検索可能な暗号化方法。
【請求項４】
前記ファイル取得情報が、確認可能な解読のためのフラグをさらに含むことを特徴とする請求項３に記載の検索可能な暗号化方法。
【請求項５】
キーワードアイテムセット内の前記ファイルロケータがそれぞれ索引ロケータに伴い、
前記ファイルの少なくとも１の暗号化リソース識別情報を含む文字列を一意の値にマッピングすることにより、各ファイルに対する索引位置指標を生成するステップと、
少なくともファイルロケータ、対応するキーワードアイテムセットロケータ、ファイルの索引位置指標を含む文字列を、一意の値にマッピングすることにより、前記キーワードアイテムセット内の各ファイルロケータに対する索引ロケータを生成するステップとを含むことを特徴とする請求項１に記載の検索可能な暗号化方法。
【請求項６】
前記索引位置指標は、少なくとも暗号化リソース識別情報と、秘密鍵を含む文字列のハッシュ値として生成されることを特徴とする請求項５に記載の検索可能な暗号化方法。
【請求項７】
前記キーワードアイテムセットロケータは、少なくとも対応するキーワードとマスター暗号鍵を含む文字列のハッシュ値として生成されることを特徴とする請求項１に記載の検索可能な暗号化方法。
【請求項８】
前記キーワードアイテムセットロケータは、前記ファイルロケータ生成鍵で対応するキーワードを暗号化することにより生成されることを特徴とする請求項１に記載の検索可能な暗号化方法。
【請求項９】
１つ以上のファイルロケータ生成鍵が、１以上のプライバシーレベルに応じてセットされることを特徴とする請求項１に記載の検索可能な暗号化方法。
【請求項１０】
前記ファイルロケータ生成鍵はそれぞれ、少なくとも１つのマスター暗号鍵と前記プライバシーレベルを示す値を含む文字列のハッシュ値であることを特徴とする請求項９に記載の検索可能な暗号化方法。
【請求項１１】

各プライバシー・レベルのファイルロケータ生成鍵は、直前のより高いプライバシーレベルのファイルロケータ生成鍵のハッシュ値であることを特徴とする請求項９に記載の検索可能な暗号化方法。
【請求項１２】
ｄ０が機密キーである場合、各プライバシーレベルのファイルロケータ生成鍵は、直前のより低いプライバシーレベルのファイルロケータ生成鍵のｄ０乗であることを特徴とする請求項９に記載の検索可能な暗号化方法。
【請求項１３】
前記ファイルロケータ生成鍵はそれぞれ、少なくとも１つのキーワードとマスター暗号鍵を含む文字列のハッシュ値であることを特徴とする請求項１に記載の検索可能な暗号化方法。
【請求項１４】
検索可能な暗号化のための装置であって、
１つ以上のファイルロケータ生成鍵をセットする暗号化／解読設定ユニットと、
一意の値に少なくとも１つのキーワードを含む文字列をマッピングすることにより、１つ以上のキーワードアイテムセットロケータを生成するキーワードアイテムセットロケータ生成ユニットと、
少なくとも１つのファイルロケータ生成鍵で複数のファイルのファイル取得情報を暗号化することにより、１つ以上のファイルロケータを生成するファイルロケータ生成ユニットと、
前記キーワードアイテムセットロケータによって識別され、対応するキーワードに関連したファイルの少なくとも１つ以上のファイルロケータを含む１つ以上のキーワードアイテムセットによって暗号化索引を形成する索引形成ユニットと
を備えることを特徴とする装置。
【請求項１５】
前記暗号化／解読設定ユニットが、複数の前記各ファイルについてファイル暗号鍵をセットし、
対応するファイル暗号鍵で各ファイルを暗号化するファイル暗号化ユニットをさらに備えることを特徴とする請求項１４に記載の装置。
【請求項１６】
前記ファイル取得情報が、少なくとも１の暗号化されたリソース識別情報および前記ファイルのファイル解読鍵を含むことを特徴とする請求項１４に記載の装置。
【請求項１７】
前記ファイル取得情報が、確認可能な解読のためのフラグをさらに含むことを特徴とする請求項１６に記載の装置。
【請求項１８】
前記ファイルの少なくとも１の暗号化リソース識別情報を含む文字列を一意の値にマッピングすることにより、各ファイルに対する索引位置指標を生成する索引位置指標生成ユニットと、
少なくともファイルロケータ、対応するキーワードアイテムセットロケータ、ファイルの索引位置指標を含む文字列を、一意の値にマッピングすることにより、前記キーワードアイテムセット内の各ファイルロケータに対する索引ロケータを生成する索引ロケータ生成ユニットを備え、
前記索引形成ユニットが、キーワードアイテムセット内の前記ファイルロケータがそれぞれ索引ロケータに伴うように前記暗号化索引を形成することを特徴とする請求項１４に記載の装置。
【請求項１９】
前記索引位置指標生成ユニットが、少なくとも暗号化リソース識別情報と、秘密鍵を含む文字列のハッシュ値として前記索引位置指標を生成することを特徴とする請求項１６に記載の装置。

【請求項２０】
前記キーワードアイテムセットロケータ生成ユニットが、少なくとも対応するキーワードとマスター暗号鍵を含む文字列のハッシュ値として前記キーワードアイテムセットロケータを生成することを特徴とする請求項１４に記載の装置。
【請求項２１】
前記キーワードアイテムセットロケータ生成ユニットが、前記ファイルロケータ生成鍵で対応するキーワードを暗号化することにより、前記キーワードアイテムセットロケータを生成することを特徴とする請求項１４に記載の装置。
【請求項２２】
前記暗号化／解読設定ユニットが、１つ以上のファイルロケータ生成鍵が、１以上のプライバシーレベルに応じてセットすることを特徴とする請求項１４に記載の装置。
【請求項２３】
前記暗号化／解読設定ユニットが、それぞれ、少なくとも１つのマスター暗号鍵と前記プライバシーレベルを示す値を含む文字列のハッシュ値を前記ファイルロケータ生成鍵としてセットすることを特徴とする請求項２２に記載の装置。
【請求項２４】
前記暗号化／解読設定ユニットが、直前のより高いプライバシーレベルのファイルロケータ生成鍵のハッシュ値を前記各プライバシー・レベルのファイルロケータ生成鍵としてセットすることを特徴とする請求項２２に記載の装置。
【請求項２５】
前記暗号化／解読設定ユニットが、ｄ０が機密キーである場合、直前のより低いプライバシーレベルのファイルロケータ生成鍵のｄ０乗に、各プライバシーレベルのファイルロケータ生成鍵をセットすることを特徴とする請求項２２に記載の装置。
【請求項２６】
前記暗号化／解読設定ユニットが、それぞれ、少なくとも１つのキーワードとマスター暗号鍵を含む文字列のハッシュ値を前記ファイルロケータ生成鍵としてセットすることを特徴とする請求項１４に記載の装置。
【請求項２７】
暗号化ファイル検索に利用する方法であって、
キーワードアイテムセットロケータによって識別され、索引ロケータによって伴われる少なくとも１つ以上のファイルロケータを含む１つ以上のキーワードアイテムセットを備える暗号化索引を格納するステップと、
索引位置指標を受信するステップと、
ファイルロケータを伴う索引ロケータが、少なくともファイルロケータ、キーワードアイテムセットを識別するキーワードアイテムセットロケータ、および受信した索引位置指標を含む文字列のマッピングによって計算した値に等しい場合に、キーワードアイテムセットからファイルロケータを削除するステップと
を有することを特徴とする方法。
【請求項２８】
１つ以上のキーワードアイテムセットロケータを受信するステップと、
受信した１つ以上のキーワードアイテムセットロケータによって識別された１つ以上のキーワードアイテムセットを検索するステップとを有し、
前記削除は前記１つ以上のキーワードアイテムセット内で実行されることを特徴とする請求項２７に記載の方法。
【請求項２９】
キーワードアイテムセットロケータを受信するステップと、
受信したキーワードアイテムセットロケータによって識別されたキーワードアイテムセットを検索するステップと、
前記キーワードアイテムセットに含まれるファイルロケータを出力するステップと、
暗号化ソース識別情報のセットを受信するステップと、

受信した暗号化リソース識別情報と一致する暗号化識別情報によって識別される暗号化ファイルを出力するステップを、さらに有することを特徴とする請求項２７に記載の方法。

【請求項３０】

暗号化リソース識別情報のセット受信後の暗号化リソース識別情報のセットから、検索で除外される暗号化ファイルの暗号化リソース識別情報をフィルタリングするステップをさらに有することを特徴とする請求項２９に記載の方法。

【請求項３１】

暗号化ファイル検索に使用する装置であって、

キーワードアイテムセットロケータによって識別され、索引ロケータによって伴われる少なくとも１つ以上のファイルロケータを含む１つ以上のキーワードアイテムセットを備える暗号化索引を格納するストレージユニットと、

ファイルロケータを伴う索引ロケータが、少なくともファイルロケータ、キーワードアイテムセットを識別するキーワードアイテムセットロケータ、および受信した索引位置指標を含む文字列のマッピングによって計算した値に等しい場合に、キーワードアイテムセットからファイルロケータを削除する索引更新ユニットと

を備えることを特徴とする装置。

【請求項３２】

前記暗号化索引内のキーワードアイテムセットロケータによって識別されるキーワードアイテムセットを検索する索引検索ユニットを備えることを特徴とする請求項３１に記載の装置。

【請求項３３】

暗号化リソース識別情報によって識別される暗号化ファイルを検索するファイル検索ユニットを備えることを特徴とする請求項３１に記載の装置。

【請求項３４】

受信した暗号化リソース識別情報のセットから、検索で除外されるファイルの暗号化リソース識別情報をフィルタリングするフィルタユニットを備えることを特徴とする請求項３３に記載の装置。

【請求項３５】

暗号化ファイル検索のための方法であって、

キーワードアイテムセットロケータおよびファイルロケータ解読鍵を受信するステップと、

前記キーワードアイテムセットロケータで１つ以上のファイルロケータを引き出すステップと、

１つ以上の暗号化リソース識別情報および対応ファイル解読鍵を取得するために前記ファイルロケータ解読鍵で各ファイルロケータを解読するステップと、

１つ以上の暗号化リソース識別情報によって識別された１以上の暗号化ファイルを引き出すステップと、

対応するファイル解読鍵で各暗号化ファイルを解読するステップと

を有することを特徴とする方法。

【請求項３６】

フラグの受信するステップと、

受信したフラグを、ファイルロケータの解読から取得したフラグと比較することにより、各ファイルロケータの解読を確認するステップとを有することを特徴とする請求項３５に記載の方法。

【請求項３７】

より低いプライバシーレベルのファイルロケータ解読鍵を取得するためにファイルロケータ解読鍵のハッシュ値を計算するステップを有することを特徴とする請求項３５に記載の方法。

【請求項３８】

ｅ０が公開鍵である場合に、より低いプライバシーレベルのファイルロケータ解読鍵を取得するために、ファイルロケータ解読鍵のｅ０乗を計算するステップを有することを特徴とする請求項３５に記載の方法。

【請求項３９】

暗号化ファイル検索のための装置であって、

少なくとも１つのキーワードアイテムセットロケータを含む検索要求を生成する検索要求ユニットと、

１つ以上の暗号化リソース識別情報および対応ファイル解読鍵を取得するために前記ファイルロケータ解読鍵で各ファイルロケータを解読するファイルロケータ解読ユニットと、

１つ以上の暗号化リソース識別情報によって識別された１以上の暗号化ファイルを引き出すファイル取得ユニットと、

対応するファイル解読鍵でファイルを暗号化するファイル解読ユニットと

を備えることを特徴とする装置。

【請求項４０】

前記ファイルアロケータ解読ユニットが、受信したフラグを、ファイルロケータの解読から取得したフラグと比較することにより、各ファイルロケータの解読を確認することを特徴とする請求項３９に記載の装置。

【請求項４１】

前記ファイルアロケータ解読ユニットが、より低いプライバシーレベルのファイルロケータ解読鍵を取得するために、ファイルロケータ解読鍵のハッシュ値を計算することを特徴とする請求項３９に記載の装置。

【請求項４２】

前記ファイルアロケータ解読ユニットが、ｅ０が公開鍵である場合に、より低いプライバシーレベルのファイルロケータ解読鍵を取得するために、ファイルロケータ解読鍵のｅ０乗を計算することを特徴とする請求項３９に記載の装置。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は、一般に情報検索方法に関し、特に、高速検索可能な暗号化のための方法、装置及びシステムに関する。

【背景技術】

【０００２】

ネットワークおよびコミュニケーション技術の広範囲にわたる利用と共に、データ記憶と管理サービスは一般的になっている。

ある状況において、利用者は、例えば、利用者端末の記憶容量が限られている、利用者端末ではデータの安定した或いは長時間の連続的なアクセスを提供することができないなどの様々な理由で、重いデータであっても、いくらかのデータを第三者ストレージベンダーによって維持されるリモートサーバーに格納する。一般的に、ストレージ管理のコストを考慮したデータメンテナンスのコストは、データの初期の取得のコストを超えて５～１０倍になる。

【０００３】

しかしながら、ほとんどの第三者ストレージベンダーは、データ機密性および保全性についてしっかりした保証をしない。極秘データが、完全には信頼できない第三者によって維持されたストレージサーバ上に格納されていると、データ機密性およびアクセス・パターン・プライバシー（ａｃｃｅｓｓ　ｐａｔｔｅｒｎ　ｐｒｉｖａｃｙ）の保証をするためにセキュリティーシステムが必要である。

【０００４】

図１は、アリス（データ所有者）が完全には信頼できない第三者（すなわちストレージ

・サービス・プロバイダ）にファイルを外部委託するシナリオを例示している。また、彼女は、さらに特定の検索者（例えば友達、同僚、身内）といくつかのファイルを共有するつもりである。言いかえれば、彼女は、アリス自身に問合せが来る代わりに、検索者にストレージ・サービス上でファイルを直接検索させたいわけである。一方、アリスは、共有ファイル上にアクセス権を規定し強化したい。図１に示される例において、アリスは、ファイルＮｏｖｅｌ．ｐｄｆ、Ｐｅｔｓ．ｊｐｇおよびＦｉｎａｎｃｉａｌ．ｄｏｃについて彼女の身内による検索とアクセスを可能にし、他のファイルについては身内に見えないようにしたい。同様に、アリスは、いくつかのファイルを彼女の友達と同僚による検索とアクセスを可能にし、他のファイルについては見えないようにしたい。このような目的を達成するには、データ機密保護およびアクセス管理基準が必要となる。

【０００５】

ストレージ・サービス・プロバイダが完全には信頼できないので、アリスのファイルをすべて暗号化することが必要とされる。また、ストレージ・サービス・プロバイダは、検索者にファイル解読鍵を流布することはできない。更に、彼女のファイルについてアクセス管理を強化するために、アリスはストレージ・サービス・プロバイダを当てにしない。

【０００６】

上記の状況を考慮すると、次のような解決課題がある。どのようにして検索者がファイルを検索してアクセスすることを可能にするか。どのようにして検索者にファイル解読鍵を配布する。どのようにして異なる検索者の異なるファイルアクセス権を区別するか。ファイルが更新されるか削除される場合に、どのようにしてサービスを維持するか。計算とコミュニケーション消費量の点からどのようにして効率的な解決を行うか。

【０００７】

リモートデータ内で簡単にかつ効率的に検索するための性能は、非常に重要な特徴である。いくつかの効率的なコンテンツベースのキーワード検索の索引付けスキームは最近既に存在する。しかしながら、安全なリモートストレージにおいてプライバシーを保ってコンテンツベースの検索を支援することは困難で、しばしばセキュリティか性能のいずれかを著しく危険にさらす傾向がある。例えば、データがリモートサーバー上に暗号化された形式で格納されている場合、コンテンツベースの検索を実行すると、サーバではそれを解読することができず、また、依頼人に大量の暗号化されたデータを転送できない場合がある。潜在的に完全に信頼できないサーバは解読キーを知る必要があるので、前者はセキュリティを妥協し、また後者は巨大なデータ転送のために性能を妥協する。

【先行技術文献】

【特許文献】

【０００８】

【特許文献１】中国特許公開公報ＣＮ１５８８３６５Ａ

【非特許文献】

【０００９】

【非特許文献１】Ｄ．Ｂｏｎｅｈ、Ｇ．Ｄ．Ｃｒｅｓｃｅｎｚｏ、Ｒ．オストロフスキー、Ｇ．Ｐｅｒｓｉａｎｏ、「キーワード検索とパブリック・キー暗号」、ＥｕｒｏＣｒｙｐｔ　２００４年

【非特許文献２】Ｒ．Ｃｕｒｔｍｏｌａ、Ｊ．Ｇａｒａｙ、Ｓ．嘉間良、「検索可能な対称鍵暗号：改良された定義および効率的な構築」ＣＣＳ　２００６

【発明の概要】

【発明が解決しようとする課題】

【００１０】

「暗号化テキストのグローバル検索技術」中国特許公開公報ＣＮ１５８８３６５Ａにおいて解決策が提案されている。

この暗号化テキストのグローバル検索技術においては、索引付けする工程中に、データ所有者は、すべてのファイルのための索引をまず生成する。その後、暗号索引を生成する鍵を利用して、索引内のキーワードを暗号化し、暗号化ファイルを生成する同じ鍵を利用し

て、ファイルを暗号化し、公開鍵でその鍵を暗号化する。最後に、データ所有者は、暗号化索引、暗号化ファイルおよび暗号化鍵をストレージサーバに格納する。検索工程中に、データ所有者は、第１に、ストレージサーバから暗号化鍵をダウンロードし、検索前に公開鍵に対応する秘密鍵でそれを解読する。第２に、データ所有者は、鍵で問い合わせるキーワードを暗号化し、ストレージサーバに暗号化したキーワードを送る。第３に、ストレージサーバは、同じ暗号化されたキーワードについて暗号索引を調べる。第４に、データ所有者はマッチング結果に従って暗号化されたファイルを引き出し、鍵でそれらを解読する。検索者が暗号索引および暗号化されたファイルについて検索することをデータ所有者が許可したければ、データ所有者は意図する検索者の公開鍵で鍵を暗号化し、検索者のもとへ暗号化した鍵を送る。

【００１１】

上記のような解決策では、データ所有者は、全てのファイルを暗号化するためにただ１つの鍵を利用する。ファイル暗号化はほとんどの場合ストリーム暗号が利用される。しかしながら、単一の鍵で１以上のファイルを暗号化することは不安定な手法として認識されている。さらに、データ所有者は、すべてのファイルおよびすべてのキーワードを暗号化するのに同じ鍵を利用する。したがって、検索者がデータ所有者のファイル上のいくつかのキーワードの検索を繰り返し実行すれば、検索者はデータ所有者の全てのファイルを引き出すことが可能になる。このため、前述の暗号化テキストのグローバル検索技術は、図１に示した適用例においてセキュリティを十分に保証することができない。

【００１２】

より複雑な別の解決策が、Ｄ．Ｂｏｎｅｈ、Ｇ．Ｄ．Ｃｒｅｓｃｅｎｚｏ、Ｒ．オストロフスキー、Ｇ．Ｐｅｒｓｉａｎｏ、「キーワード検索とパブリック・キー暗号」、ＥｕｒｏＣｒｙｐｔ　２００４年や、Ｒ．Ｃｕｒｔｍｏｌａ、Ｊ．Ｇａｒａｙ、Ｓ．嘉間良、「検索可能な対称鍵暗号：改良された定義および効率的な構築」ＣＣＳ　２００６によって提案されている。このような解決策では、索引付けする工程中に、データ所有者は、索引を生成するために、ファイル内のいくつかの特別のフィールド（例えば、電子メール内の「緊急」のようなキーワード）をまず選択する。

具体的には、各ファイルについて、データ所有者はその特別のキーワードを暗号化する。例えば、＜Ａ＝g^r，　Ｂ＝H_2（ｅ（H_1（ＫＷ），h^r））　＞は暗号化されたキーワードである。ここで、ＫＷ　はキーワード、ｅ：　$G_1 \times G_1 \rightarrow G_2$，　ｇ　は$G_1$のジェネレータ、$H_1$　とH_2　は、２つの異なるハッシュ関数、ｒ　はZ^*_p内の乱数、ｈ　はg^xと等しく、ｘ　は秘密鍵であり、Z^*_p内にある。このように、安全な索引は要素の集合からなっており、ｉ番目の組の形式は、＜$ciphertext_i$：（A_1，B_1），…，（A_n，B_n）＞である。ここで、$ciphertext_i$　は、ファイル暗号鍵K_{file_i}．で暗号化された$File_i$の暗号化テキストである。検索工程中に、データ所有者は、まず、検索者にキーワードＫＷに対するｔｒａｐｄｏｏｒを$T_{KW}=H_1^x$（ＫＷ）のように計算し発行することにより、検索者がキーワードを問い合わせることを認可する。それから、検索者はストレージサーバにT_{KW}を送る。各ファイルの各暗号化されたキーワードについて、ストレージサーバは、ファイルがＫＷを含んでいるかどうかを検査するために、Ｂ＝H_2（ｅ（T_{KW}，Ａ））を計算する。Ｂ＝Ｂ’であれば、暗号化されたファイルはマッチング出力となり、そうでなければマッチング出力とならない。検索者が暗号化されたファイルを解読したければ、データ所有者との別のラウンドトリップが対応する解読キーを取り出すのに必要である。

【００１３】

上記の解決策によれば、検索にストレージサーバが費やす計算複雑性は、Ｏ（ｍ×ｎ）で表すことができる。ここで、ｍはファイル数であり、ｎは各ファイル内の異なるキーワードの平均数である。例えば、１０００のファイルおよび１０のキーワードがある場合、８個のＣＰＵを備えたストレージサーバ上で１検索当たり３０秒を必要とする。上記のような解決策の他の問題点は、ストレージサーバがマッチする結果（すなわちキーワードを含む暗号化ファイル）を返した後、検索者は、暗号化ファイルの解読キーのために、デー

タ所有者に連絡をとらなければならないことである。
【課題を解決するための手段】
【００１４】
本発明は、従来技術における問題を考慮してなされたものであり、検索暗号化のための方法、装置およびシステムを提供する。
【００１５】
本発明による検索可能な暗号化方法は、検索可能な暗号化方法であって、
１つ以上のファイルロケータ生成鍵をセットするステップと、
一意の値に少なくとも１つのキーワードを含む文字列をマッピングすることにより、１つ以上のキーワード項目集合ロケータを生成するステップと、
少なくとも１つのファイルロケータ生成鍵で複数のファイルのファイル取得情報を暗号化することにより、１つ以上のファイルロケータを生成するステップと、
前記キーワード項目集合ロケータによって識別され、対応するキーワードに関連したファイルの少なくとも１つ以上のファイルロケータを含む１つ以上のキーワード項目集合によって暗号化索引を形成するステップとを有する。
【００１６】
本発明による検索可能な暗号化のための装置は、検索可能な暗号化のための装置であって、
１つ以上のファイルロケータ生成鍵をセットする暗号化／解読設定ユニットと、
一意の値に少なくとも１つのキーワードを含む文字列をマッピングすることにより、１つ以上のキーワード項目集合ロケータを生成するキーワードアイテムセットロケータ生成ユニットと、
少なくとも１つのファイルロケータ生成鍵で複数のファイルのファイル取得情報を暗号化することにより、１つ以上のファイルロケータを生成するファイルロケータ生成ユニットと、
前記キーワード項目集合ロケータによって識別され、対応するキーワードに関連したファイルの少なくとも１つ以上のファイルロケータを含む１つ以上のキーワード項目集合によって暗号化索引を形成する索引形成ユニットとを備える。
【００１７】
本発明による暗号化ファイル検索に利用する方法は、暗号化ファイル検索に利用する方法であって、
キーワードアイテムセットロケータによって識別され、索引ロケータによって伴われる少なくとも１つ以上のファイルロケータを含む１つ以上のキーワードアイテムセットを備える暗号化索引を格納するステップと、
索引位置指標を受信するステップと、
ファイルロケータを伴う索引ロケータが、少なくともファイルロケータ、キーワードアイテムセットを識別するキーワードアイテムセットロケータ、および受信した索引位置指標を含む文字列のマッピングによって計算した値に等しい場合に、キーワードアイテムセットからファイルロケータを削除するステップとを有する。
【００１８】
本発明による暗号化ファイル検索に使用する装置は、暗号化ファイル検索に使用する装置であって、
キーワードアイテムセットロケータによって識別され、索引ロケータによって伴われる少なくとも１つ以上のファイルロケータを含む１つ以上のキーワードアイテムセットを備える暗号化索引を格納するストレージユニットと、
ファイルロケータを伴う索引ロケータが、少なくともファイルロケータ、キーワードアイテムセットを識別するキーワードアイテムセットロケータ、および受信した索引位置指標を含む文字列のマッピングによって計算した値に等しい場合に、キーワードアイテムセットからファイルロケータを削除する索引更新ユニットとを備える。
【００１９】

本発明による暗号化ファイル検索のための方法は、暗号化ファイル検索のための方法であって、
キーワードアイテムセットロケータおよびファイルロケータ解読鍵を受信するステップと、
前記キーワードアイテムセットロケータで１つ以上のファイルロケータを引き出すステップと、
１つ以上の暗号化リソース識別情報および対応ファイル解読鍵を取得するために前記ファイルロケータ解読鍵で各ファイルロケータを解読するステップと、
１つ以上の暗号化リソース識別情報によって識別された１以上の暗号化ファイルを引き出すステップと、
対応するファイル解読鍵で各暗号化ファイルを解読するステップとを有する。
【００２０】
本発明による暗号化ファイル検索のための装置は、暗号化ファイル検索のための装置であって、
少なくとも１つのキーワードアイテムセットロケータを含む検索要求を生成する検索要求ユニットと、
１つ以上の暗号化リソース識別情報および対応ファイル解読鍵を取得するために前記ファイルロケータ解読鍵で各ファイルロケータを解読するファイルロケータ解読ユニットと、
１つ以上の暗号化リソース識別情報によって識別された１以上の暗号化ファイルを引き出すファイル取得ユニットと、
対応するファイル解読鍵でファイルを暗号化するファイル解読ユニットとを備える。
【発明の効果】
【００２１】
本発明によれば、データ所有者は、暗号化された逆索引上で属性ベースとマルチレベル検索を適用することが可能となる。
サーバのもとに送られる前に、データおよび関連するメタデータはすべて暗号化を利用して、データ所有者側では暗号化される。データはサーバではその一生を通じて暗号化され続ける。暗号化されたデータ上のコンテンツベースの検索を可能にするために、格納されたどのファイルも、データ所有者のサイトでは索引付けする工程において安全に索引付けされる。これはサーバ側では、索引構造の機密の記憶装置に帰着し、将来の安全な依頼人アクセスについて利用可能とする。仮想削除は検索結果においてろ過することによって保証される。マルチレベル検索は、プライバシーレベルあるいはキーワードのいずれかに従って、制限、および検索者に対応する解読キーの配備によって達成される。
【００２２】
本発明は文書とキーワードの大量の検索を計るために効率的なサーチアルゴリズムを採用する。
この発明によって、検索時間は、Ｏ（ｌｏｇ（Ｎ））　ｔｏ　Ｏ（１）　となる。ここで、Ｎはファイルの全体集合における異なるキーワードの合計の数である。したがって、Ｏ（ｍ×ｎ）を必要とする先行技術と比較して、この発明は効率的で実行可能な解決策を提供する。
【図面の簡単な説明】
【００２３】
【図１】ストレージサービスの利用例を示す図である。
【図２】本発明が適用されるシステムの構成例を示す概略形である。
【図３】本発明の１実施の形態によるデータ所有者端末の構成例を示すブロック図である。
【図４】本発明の１実施の形態によるデータ所有者端末の動作の概要を説明するフローチャートである。
【図５】本発明の１実施の形態による暗号化された逆索引を生成する処理の例を説明する

フローチャートである。
【図６】本発明の１実施の形態による索引付け工程におけるデータの流れを説明する図である。
【図７】本発明の１実施の形態によるサーバの構成例を示すブロック図である。
【図８】本発明の１実施の形態による検索者端末の構成例を示すブロック図である。
【図９】本発明の１実施の形態による検索処理の例を説明するフローチャートである。
【図１０】本発明の１実施の形態による検索工程におけるデータの流れを説明する図である。
【図１１】本発明の１実施の形態による検索工程におけるフィルタリング処理のデータの流れを説明する図である。
【図１２】本発明の１実施の形態によるデータ所有者端末の構成例を示すブロック図である。
【図１３】本発明の１実施の形態による索引付け工程におけるデータの流れを説明する図である。
【図１４】本発明の１実施の形態によるサーバの構成例を示すブロック図である。
【図１５】本発明の１実施の形態による、暗号化ファイルが削除されることになっている場合に暗号化索引を更新するサーバの処理を説明するフローチャートである。
【図１６】本発明の１実施の形態による暗号化索引の更新におけるデータの流れを説明する図である。
【図１７】本発明の１実施の形態による暗号化された索引の更新におけるデータの流れを説明する図である。
【発明を実施するための形態】
【００２４】
以下、図面を参照して本発明について説明する。以下の詳細な説明においては、多くの特定の細部が、本発明を十分に理解するために提供される。しかしながら、これらの特定の細部のうちのいくつかがなくとも本発明を実行できることは、当業者にとって明白である。図面および以下の説明において、周知の構成および技術については、本発明を不必要に不明瞭にしないために詳細には説明しない。
【００２５】
図２は本発明が適用されるシステムの概略構成を例示するブロック図である。
このシステムには、３種類の当事者が含まれる。少なくとも１人のデータ所有者、少なくとも１つのサービスプロバイダおよび一人以上の検索者である。
図２に示すように、データ所有者の装置或いは端末、サービスプロバイダによって管理されるサーバおよび１以上の検索者の装置或いは端末が、通信ネットワークによって接続され、互いに通信することができる。
データ所有者と検索者の装置或いは端末は、それぞれ、情報を処理することができかつ情報を通信することができる装置として実現される。例えば、パーソナルコンピュータ（ＰＣ）、携帯情報端末（ＰＤＡ）、スマートフォン（ｓｍａｒｔ　ｍｏｂｉｌｅ　ｐｈｏｎｅ）或いはその他のデータ処理装置等である。
一般に、サーバは、データを格納し維持することが可能であり、また、端末によるデータに対する限定受信を可能にし、サービスプロバイダによって管理される単一の装置或いは装置の集合として実現される。
【００２６】
本発明のシステムにおいて、データ所有者は自身のファイルおよび関連するメタデータを暗号化し、暗号化テキストをサーバに格納する。サーバにおいて、ファイルはその寿命を通じて暗号化されたままである。暗号化されたファイルについてコンテンツベースの検索を可能にするために、データ所有者（装置又は端末）が、ファイルの各キーワードによって暗号化された索引を生成し、サーバに暗号化された索引を格納する。索引は逆索引であり、それがサーバに格納されるとともに暗号化されたままとなる。
検索者が暗号化された索引を検索し、１つ以上の指定されたキーワードを含む一定のフ

ファイルを取り出すことを許可するために、データ所有者は、検索者に特定の復号キーを含む必要なデータを発行する。その後、データ所有者によって発行されたデータを用いて、検索者は、検索要求によってサーバ上に格納された暗号化されたファイルを検索する。その結果、サーバから関連する暗号化されたファイルを取り出し、発行された復号キーで復号することによりファイルの平文を取得する。

【００２７】

本発明によれば、暗号化されたファイルは、１つ以上のキーワードアイテムセット（ＫＩＳ）からなる新しい暗号化された逆索引と索引付けされる。

サーバ上に格納されているデータは、クライアントサーバ・トランシット（ｃｌｉｅｎｔ－ｓｅｒｖｅｒ　ｔｒａｎｓｉｔ）間やサーバ側はもちろん、悪意のあるサーバによってさえ復号することができない。

全ての特定の検索者は、その検索者に発行されたあるプライバシーレベルのファイルロケータ復号キーに一致する暗号化ファイルを検索し復号することができるだけである。暗号化された逆索引の実際の更新がその後都合で実行されているならば、暗号化ファイルは、削除された後に検索から除外することが可能である。

【００２８】

本発明の種々の特徴および典型的な実施の形態については、以下にその詳細を説明する。実施の形態の以下の説明は、本発明の例を示すことにより、本発明をよりよく理解するためになされている。本発明は、以下に説明されたどんな特定の構成およびアルゴリズムにも限定されない。本発明の範囲から外れない限り、構成要素、構成部品およびアルゴリズムのいかなる変形、代替案および改良も包含する。

（暗号化および検索）

【００２９】

図３は、本発明の１実施の形態によるデータ所有者端末の構成例を示すブロック図である。

図３に示すように、データ所有者端末１００は、主に、キーワードユニット１０１、暗号化／復号設定ユニット１０２、ファイル暗号化ユニット１０３、ＫＩＳロケータ生成ユニット１０４、ファイルロケータ生成ユニット１０５および索引形成ユニット１０６を含む。

【００３０】

本実施の形態によるデータ所有者端末１００の動作について、図４および図５を参照して説明する。

図４は、データ所有者端末の動作の概略を説明するフローチャート、また、図５は、暗号化された逆索引を生成する処理の例を示すフローチャートである。

【００３１】

図４に示すように、ステップＳ２０１で、キーワードユニット１０１は、各ファイルと、そのファイルに含まれるか或いはそのファイルに関連した１つ以上のキーワードとの関係を設定する。

これは、ファイルからキーワードを抽出することにより、あるいは利用者からの入力により行われる。

また、ファイルとキーワードの関係は、データ所有者によってあらかじめセットされ、テーブルとしてデータ所有者端末の記憶手段に格納され、あるいは遠隔地から受信することが可能である。そのような状況において、キーワードユニット１０１は、データ所有者端末の構成に必ずしも必要ない。

【００３２】

ステップＳ２０２で、暗号化／復号設定ユニット１０２は、各ファイルについてファイル暗号鍵と復号鍵をセットする。

ファイル暗号鍵は対応するファイルを暗号化するために利用される。また、ファイル復号鍵は対応する暗号化されたファイルを復号するために利用される。

ファイル暗号化／復号キーは、任意の暗号化方法によって任意にセットすることが可能

である。
本発明において、ファイルのファイル暗号鍵およびファイル復号鍵は、非対称の暗号化スキームによって別々にセットされる。
しかしながら、対称な暗号化スキームによる本発明におけるファイルのファイル暗号鍵とファイル復号鍵の両方として、単一の鍵を利用すること可能である。その場合、同じファイルのファイル復号鍵とファイル暗号鍵は、以下の説明において同じとなる。
【００３３】
ステップＳ２０３で、暗号化／復号設定ユニット１０２は、さらに、検索に利用されるファイルロケータ生成／復号鍵をセットし割り当てる。これについては、以下に詳細に説明する。
【００３４】
ファイルロケータ生成鍵は、暗号化された索引（これについては後述する）のファイルロケータを生成するファイルのファイル取得情報を暗号化するために用いられる。また、ファイルロケータ復号鍵は、暗号化された索引のファイルロケータを復号するために利用される。
本実施の形態において、複数のファイルロケータ生成鍵と暗号鍵のペアは、異なるプライバシーレベルに合わせてセットすることが可能である。
【００３５】
例えば、図１に示す状況においては、身内に対するレベル１、友達に対するレベル２および同僚に対するレベル３の、３つのプライバシーレベルが必要である。
以下に述べるように、異なるプライバシー・レベルの検索者は、自身のプライバシー・レベルで明かせるファイルを検索して復号することができるが、自身のプライバシー・レベルで明かせないファイルについては不明なままである。
上記の例において、ファイルロケータ生成／復号鍵の３つのペアは、各々３つのプライバシー・レベルのうちの１つにセットされる。

$EKey_1$／$DKey_1$（レベル１用），$EKey_2$／$DKey_2$（レベル２用）、$EKey_3$／$DKey_3$（レベル３用）

以降の説明で、ＥＫｅｙはファイルロケータ生成キーを表わし、ＤＫｅｙはファイルロケータ復号鍵を表わす。
【００３６】
また、ファイルロケータ生成鍵および対応するファイルロケータ復号鍵は、任意の暗号化方法によって任意にセットすることができる。
それらは、非対称の暗号化スキームで別々にセットするか、あるいは対称な暗号化スキームで同じものにセットすることが可能である。
対称な暗号化スキームでは、同じペアのファイルロケータ復号鍵とファイルロケータ生成鍵は、同じ物である。
【００３７】
例えば、プライバシーレベルｍのファイルロケータ生成／復号鍵は、以下のように生成される。

$$EKey_m = DKey_m = Hash(MEK||m) \quad (式1)$$

ここで、Ｈａｓｈ（ＭＥＫ||ｍ）は、鍵ＭＥＫを持つハッシュ関数である。“||”は、所定の順の文字列か数字の組み合わせを表わす。また、ＭＥＫはデータ所有者のマスター暗号鍵である。それは、暗号化／復号設定ユニット１０２によって選択されるか、或いは他の信用機関から発行される。
言うまでもなく、他の類似するアルゴリズムの値も、ファイルロケータ生成／復号鍵として利用することができる。

【００３８】
データ所有者は、ファイルロケータ生成／復号鍵の後の計算のために、ファイルロケータ生成／復号鍵を計算するのに必要なアルゴリズムおよび関連するパラメータを、例えば、暗号化／復号設定ユニット１０２に、保持する。
例えば、データ所有者端末は、マスター暗号鍵ＭＥＫを格納し、暗号化された索引が設定された後、後の段階において特定のプライバシーレベルで検索者に権限を与える場合、式によってファイルロケータ生成／復号鍵を計算する。
このように、データ所有者は、暗号化された索引が設定された後、全てのファイルロケータ生成／復号鍵を格納することを要求されない。
あるいは、データ所有者端末は、例えば、暗号化／復号設定ユニット１０２に、ローカルにマッピングテーブルを格納することができる。
後の段階において、特定のプライバシー・レベルのファイルロケータ生成／復号鍵が必要になれば、データ所有者端末が、対応する鍵を見つけ出すために単にマッピングテーブルを調べる。
【００３９】
ここで、図４に戻る。各ファイルについてファイル暗号鍵と復号鍵がセットされた後、ファイル暗号化ユニット１０３は、ステップＳ２０４で、対応するファイル暗号鍵で各ファイルを暗号化する。
【００４０】
ステップＳ２０５で、索引形成ユニット１０６は、ファイルのキーワードに基づいて、１つ以上のキーワードアイテムセット（ＫＩＳｅｓ）からなる暗号化された逆索引を形成する。
本実施の形態によるキーワードアイテムセットＫＩＳは、それぞれ１つのキーワードに対応する。
本実施の形態による独自のインデックス生成方法について、図５を参照して説明する。
【００４１】
図５は、本実施の形態による暗号化逆索引を生成する処理の例を示している。
ステップＳ３０１で、ＫＩＳロケータ生成ユニット１０４は、キーワードKW_iについて、キーワードKW_iのキーワードアイテムセットＫＩＳの一意の識別子として一意のＫＩＳロケータＫＬｉを生成する。
ＫＩＳロケータKL_iは、それがキーワードKW_iに一意に対応する限り、任意に生成され、データ所有者の支援なしでは、その他の誰もKL_iからのキーワードKW_iを計算することは不可能である。
一般に、各キーワードについてＫＩＳロケータを生成するために、ＫＩＳロケータ生成ユニット１０４は任意の利用可能なアルゴリズムによってユニークな値に各キーワードをマッピングする。
例えば、ＫＩＳロケータＫＬｉは、以下のように生成される。

$$KL_i = Hash(MEK||KW_i) \qquad (式２)$$

【００４２】
この説明において利用されるハッシュ関数が、当業者に知られている多くの写像アルゴリズムのうちの単なる１つの例であることに留意すべきであり、本発明はそのようなアルゴリズムに限定されない。
【００４３】
ステップＳ３０２で、ファイルロケータ生成ユニット１０５は、ファイルを見せることができる１つ以上のプライバシーレベルに従って各ファイルについて１つ以上のファイルロケータを生成する。
詳しくは、ファイル$FILE_j$をプライバシーレベルｍで見せるなら、ファイルロケータ生成ユニット１０５は、プライバシーレベルｍに割り当てられたファイルロケータ生成

鍵$EKey_m$でファイル$FILE_j$のファイル取得情報を暗号化することにより、ファイル$FILE_j$のファイルロケータを生成する。

ファイルを複数のプライバシーレベルで見せるなら、ファイルロケータ生成ユニット１０５は、ファイルについて複数のファイルロケータを生成する。各ファイルアロケータは、複数のプライバシーレベルのうちの１つに対応し、それぞれのファイルロケータ生成鍵で生成される。

【００４４】

例えば、図１に示す状況において、アリスは、ファイルＮｏｖｅｌ．ｐｄｆ，　Ｐｅｔｓ．ｊｐｇ　と　Ｆｉｎａｎｃｉａｌ．ｄｏｃをプライバシーレベル１で見ること、ファイルＮｏｖｅｌ．ｐｄｆ　と　Ｐｅｔｓ．ｊｐｇをプライバシーレベル２で見ること、また、ファイルＲｅｓｅａｒｃｈ．ｐｐｔと　Ｐｅｔｓ．ｊｐｇをプライバシーレベル３で見ることを望む。この例において、各ファイルを見ることができるレベルはテーブル１に示される。

【表１】

	Level 1	Level 2	Level 3
Research.ppt	No	No	Yes
Novel.pdf	Yes	Yes	No
Pets.jpg	Yes	Yes	Yes
Financial.doc	Yes	No	No

【００４５】

一例として、プライバシーレベル１およびプライバシーレベル２で見ることができるファイルＮｏｖｅｌ．ｐｄｆを引用すると、ファイルロケータ生成ユニット１０５は、ファイルロケータ$FL_{novel.pdf,\ 1}$を生成するためにプライバシーレベル１のファイルロケータ生成鍵$EKey_1$でファイルＮｏｖｅｌ．ｐｄｆのファイル取得情報を暗号化し、ファイルロケータ$FL_{novel.pdf,\ 2}$を生成するためにプライバシー・レベル２のファイルロケータ生成鍵$EKey_2$でファイル取得情報を暗号化する。

【００４６】

ファイル取得情報は、サーバから暗号化ファイルを取り出すために必要な情報と、暗号化ファイルを解読するための情報を含んでいる。

例えば、ファイル$FILE_j$のファイル取得情報が、$CFN_j||K_{file_j}$である場合、CFN_jはファイル$FILE_j$の暗号化ファイルを識別するための暗号化されたリソース識別情報でり、K_{file_j}は暗号化／解読設定ユニット１０２によってセットされたファイル$FILE_j$のファイル解読鍵である。

暗号化されたリソース識別情報CFN_jは、ファイル$FILE_j$の暗号化ファイル名、あるいはファイル$FILE_j$の暗号文のＵＲＬなどである。

【００４７】

本実施の形態によれば、プライバシーレベルｍのファイル$FILE_j$についてのファイルロケータ$FL_{j,\ m}$は、以下のように生成される。

$$FL_{j,\ m} = E(EKey_m,\ CFN_j||K_{file_j}) \quad \text{（式３）}$$

ここで、Ｅ（Ｘ（Ｙ））は、Ｘによって暗号化するＹを表わす暗号化関数である。

【００４８】

図５へ戻ると、ＫＩＳロケータ生成ユニット１０４が各キーワードKW_iについてＫＩＳロケータKL_iを生成し、ファイルロケータ生成ユニット１０５が、すべてのファイルについてファイルロケータを生成した後、索引形成ユニット１０６は、ステップＳ３０３で、対応するＫＩＳロケータKL_iおよびそのキー　ワードに関するファイルのすべてのファイルロケータによって各キーワードKW_iについてＫＩＳを形成する。

【００４９】

一例として、図１およびテーブル１において示される状況を引用し、かつファイルＲｅｓｅａｒｃｈ．ｐｐｔとＮｏｖｅｌ．ｐｄｆがキーワードKW_aに関係していると想定すると、本実施の形態によれば、キーワードKW_aのＫＩＳは、要素＜KL_a：$FL_{Research.ppt,3}$＝Ｅ（$EKey_3$，$CFN_{Research.ppt}||K_{Research.ppt}$），$FL_{Novel.pdf,1}$＝Ｅ（$EKey_1$，$CFN_{Novel.pdf}||K_{Novel.pdf}$），$FL_{Novel.pdf,2}$＝Ｅ（$EKey_2$，$CFN_{Novel.pdf}||K_{Novel.pdf}$）＞として生成される。

【００５０】

索引形成ユニット１０６は、各キーワードについて、キーワードアイテムセットＫＩＳを形成する。また、ステップ３０４で、索引形成ユニット１０６は、すべてのキーワードアイテムセットＫＩＳによって暗号化索引を形成する。

【００５１】

ＫＩＳロケータがキーワードアイテムセットＫＩＳと別に設定され、単にキーワードアイテムセットＫＩＳｅｓの識別情報として構成されて処理される場合があることに留意するべきである。

その場合、マッピング関係は、キーワードアイテムセットＫＩＳの一部としてＫＩＳロケータを生成する代わりに、各ＫＩＳロケータと対応するＫＩＳの間に生成される。

暗号化索引は、一意のＫＩＳロケータによる標準データ構造（例えば、ツリーベース構造）に構成することが可能である、

また、ＫＩＳロケータは暗号化索引内の正確な位置を指定する。したがって、サーバは対数時間内に解読されたデータを見つけ出すことが可能である。

【００５２】

図４へ戻ると、ステップＳ２０６で、データ所有者端末１００は、サーバに、暗号化されたファイルおよび暗号化された索引を格納する。

検索者と同様に、データ所有者端末とサーバの間の通信は、図示しない通信ユニットによって実行される。

ここで使用される用語「サーバ」は、記憶装置と検索サービスの両方を提供する単一の装置、あるいは互いに隣接し又は隔たっている複数の装置の集合を意味する。各サーバは、記憶装置、データ探索、ユーザー管理などの種々のサービスに責任を負うか、あるいはサービスの負担を共有する。

例えば、データ所有者端末１００は、ストレージサーバ上に暗号化されたファイルを格納し、ストレージサーバと通信し合うファイル検索サーバ上に暗号化された索引を格納することが可能である。

説明を簡単にするために、サービスを提供するそのような装置をすべて「サーバ」と呼ぶことにする。

【００５３】

本実施の形態による索引付け段階の処理に対する理解を助けるために、図６は、上述した例におけるデータの流れを示している。

【００５４】

本発明の１実施の形態による索引付けする工程におけるデータ所有者端末の処理は、上

述した通りである。
検索工程における処理とサーバおよび検索者端末の構成について、図７～図９を参照して説明する。
【００５５】
図７は、本発明の１実施の形態によるサーバの構成例を示し、また、図８は、本発明の１実施の形態による検索者端末の構成例を示している。
【００５６】
図７に示すように、サーバ４００は、暗号化されたファイルおよびデータ所有者から受信した暗号化された索引を格納する記憶ユニット４０１、検索者の要求に応じて暗号化された索引内の検索を実行するための索引検索ユニット４０２、特定の暗号化されたリソース識別情報によって識別された暗号化されたファイルを検索するファイル検索ユニット４０３とを含む。
【００５７】
図８に示すように、検索者端末５００は、主に、検索要求の生成のために主に検索要求ユニット５０１と、ファイルロケータを復号するためのファイルアロケータ復号ユニット５０２と、ファイル取得要求を生成するためのファイル取得ユニット５０３と、取得した暗号化されたファイルを復号するためのファイル復号ユニット５０４とを含む。
【００５８】
本発明の本実施の形態による検索の処理例について、図９を参照して説明する。
【００５９】
まず、ステップＳ６０１で、データ所有者が検索者に対してキーワードについて検索することを可能にしたければ、データ所有者は、検索者に権限を与えられた適切なプライバシーレベルのファイルロケータ解読鍵を発行すると共に、安全な方法で、検索者に対してキーワードのＫＩＳロケータを発行する。
データ所有者は、様々な方法によってＫＩＳロケータおよびファイルロケータ解読鍵のそれぞれを各検索者に通知することができる。例えば、データ所有者端末と検索者端末の間の通信ネットワークによって送られた電子メッセージによって自動的に通知してもよいし、口頭あるいは書面形式で通知してもよい。
認証は検索者の要求に応じて実行することができる。
例えば、検索者は、検索資格要求ユニット（図示せず）等によって、彼が検索したい１つ以上のキーワードを含んでいる要求をデータ所有者に対して送信する。
検索者の身元を確認した後に、データ所有者は検索者に適したプライバシーレベルを決定し、要求されたキーワードのＫＩＳロケータおよび決定したプライバシーレベルのファイルロケータ解読鍵を検索者に対して発行する。
ＫＩＳロケータおよびファイルロケータ解読鍵は、データ所有者端末に格納されたテーブルから検索され、あるいは、格納されたセキュリティパラメータに従ってデータ所有者端末によってオンラインで計算される。
認証の処理は、例えば、データ所有者端末の認証ユニット（図示せず）によって実行することが可能である。
ある状況では、セキュリティ認証は検索者がデータ所有者からの認証を取得するために要求されるかもしれない。
【００６０】
検索工程において、ステップＳ６０２に示すように、検索者端末は、検索要求ユニット５０１によってＫＩＳロケータを含む検索要求を生成し、サーバに検索要求を送信する。
【００６１】
サーバが検索者端末からのＫＩＳロケータを含む要求を受信した後、サーバは、ステップＳ６０３に示すように、要求において受信したＫＩＳロケータと同じＫＩＳロケータのＫＩＳを見つけ出すために、索引検索ユニット４０２によって記憶ユニット４０１に格納された暗号化索引内の検索を実行する。
その後、サーバは、ステップＳ６０４で、一致したＫＩＳに含まれるファイルロケータ

を検索者端末に送信する。
上述したように、これらのファイルロケータの各々は、ＫＩＳに対応するキーワードに関連したファイルのファイル取得情報を、ファイルロケータ生成鍵によって暗号化することにより生成される。
【００６２】
サーバからファイルロケータを受信した後に、ステップＳ６０５に示すように、検索者端末は、各ファイルのファイル取得情報を取得するために、ファイルアロケータ解読ユニット５０２によって、データ所有者によって発行されたファイルロケータ解読鍵で各ファイルロケータを解読する。ファイル取得情報は、暗号化されたリソース識別情報およびファイルの対応ファイル解読鍵を含んでいる、
上述したように、各ファイルロケータは、データ所有者によってあるプライバシーレベルのファイルロケータ生成鍵でファイル取得情報を暗号化することにより生成される。
特定のプライバシーレベルのファイルロケータ解読鍵では、検索者は、他のプライバシーレベルの他のファイルロケータ生成鍵で暗号化されたファイルロケータを解読することができない。
これは、検索者が、データ所有者によって認証されたプライバシーレベルで見ることができるファイルの暗号化されたリソース識別情報および対応ファイル解読鍵を取得することができることを保証するが、そのプライバシーレベルで見ることのできないファイルの正確な暗号化されたリソース識別情報およびファイル解読鍵を取得することはできない。
【００６３】
その後、検索者端末は、ステップＳ６０６で、ファイル取得ユニット５０３によって、ステップＳ６０５で取得した暗号化されたリソース識別情報を含むファイル取得要求を生成し、サーバへそのファイル取得要求を送信する。
【００６４】
検索者からの暗号化されたリソース識別情報を含むファイル取得要求を受信した後に、ステップＳ６０７で、サーバのファイル検索ユニット４０３は、格納された暗号化ファイルのうち、受信した暗号化リソース識別情報とマッチする暗号化ファイルを見つけ出す。マッチする暗号化ファイルを見つけると、サーバは、検索者端末に対してマッチした暗号化ファイルを送信する。
【００６５】
暗号化されたファイルを受信すると、検索者端末は、ステップＳ６０８で、ファイル解読ユニット５０４によって対応ファイル解読鍵を用いて暗号化ファイルを解読する。
それにより、検索者は検索結果としてファイルを取得することができる。
【００６６】
ステップＳ６０５では、検索者が、データ所有者がセットしたプライバシーレベルで現せないファイルの適正な暗号化されたリソース識別情報およびファイル解読鍵を取得できないことに注目すべきである。
検索者が他のプライバシーレベルのファイルロケータを不正に解読し、サーバのもとへ取得した正しくない暗号化リソース識別情報を送っても、サーバは正しい暗号化ファイルを捜し出さない。したがって、他のプライバシーレベルでのみ現せる暗号化ファイルは、検索者に提供されない。
検索者がサーバからのそのような暗号化ファイルをたまに取得しても、検索者はこれらのファイルを正確に解読することができない。このことは、検索者が、特定のキーワードを含みかつデータ所有者によってセットされた特定のプライバシーレベルで現せるファイルを検索して見ることができるだけであることを保証する。また、全工程の間、すべてのファイルがサーバに示されるとは限らないことに注目すべきである。
【００６７】
フローチャートには示さなかったが、ステップＳ６０５で検索者によって取得した１つ以上の暗号化されたリソース識別情報が、上述したようなＵＲＬである場合、検索者のもとへこれらのＵＲＬを送るのではなく、検索者がこれらのＵＲＬによって暗号化ファイル

を直接取得するようにすることも可能である。
あるいは、検索者がさらにサーバのもとへこれらのＵＲＬを送り、検索者のファイル検索ユニット４０３がこれらのＵＲＬによって識別されたネットワーク位置から暗号化ファイルを取り出すようにすることも可能である。
【００６８】
上に述べられた例では、検索者は１検索でサーバのもとに１つのＫＩＳロケータを送る。データ所有者が複数のＫＩＳロケータを検索者に発行する場合には、検索者は、複数のキーワードについて検索を実行するために、サーバのもとへ検索要求で複数のＫＩＳロケータを送るようにすることも可能である。

（確認可能な解読）
【００６９】
上記実施の形態において、別のプライバシーレベルのファイルロケータが、検索者によって不正に解読される可能性がある。また、無効な情報が転送されて処理されるかもしれない。
ところが、本発明の他の実施の形態においては、無効な暗号化されたリソース識別情報の転送およびサーバ側で無効な暗号化リソース識別情報によって暗号化されたファイルを探し出す処理を回避するために、検索者がサーバのもとへファイル取得要求を送る前に、各ファイルロケータの解読の正しいことを検索者側でチェックする。
確認可能な解読は、ファイルロケータが生成される時、ファイル取得情報と一緒に暗号化された既知の値（例えば、ファイル取得情報に加えるフラグ）を確認することにより実現される。
そのような１実施例について以下に説明する。
【００７０】
この実施例において、ファイル$FILE_j$のファイル取得情報は、$FLAG||CFN_j||K_{file_j}$に拡張される。ここで、ＦＬＡＧは、データ所有者によって選択された任意の値あるいは他の文字である。

【００７１】
索引付けする工程の処理は、基本的に、式２の代わりの式を除いて、上述した実施の形態と同じである。データ所有者端末は、ステップ３０４で、以下のように、ファイル$FILE_j$のファイルロケータを生成する。

$$FL_{j,m} = E(EKey_m, FLAG||CFN_j||K_{file_j}) \quad （式４）$$

【００７２】
検索工程で、データ所有者端末は、ステップＳ６０１で、ＫＩＳロケータおよびファイルロケータ解読鍵に加えてフラグＦＬＡＧを検索者端末に送信する。
【００７３】
サーバからファイルロケータを取得する検索者端末の処理は、上述した実施の形態と同様である。
受信したファイルロケータを解読する際に、検索者端末のファイルアロケータ解読ユニット５０２は、解読されたファイルロケータに包含されるフラグがデータ所有者から受信したフラグと同じであるかどうかをチェックする。
マッチしていれば、それはファイルロケータの解読が適正であることを示しており、正しいファイル取得情報が取得される。
そうでなければ、それは、不適当なファイルロケータ解読鍵あるいは他の理由によりファイルロケータの解読が失敗することを示す。
それにより、確認可能な解読がフラグの利用により実現される。

この実施の形態による検索工程処理の理解を助けるために、図１０はそのような事例のデータの流れを例示している。

【００７４】

上述したような確認によって、検索者端末は、対応暗号化ファイルを取り出しかつ受信したファイルの解読に適正なファイル解読鍵を利用するために、適正な暗号化リソース識別情報を選択しサーバに送信することが可能である。

【００７５】

この実施の形態におけるフラグのチェックによれば、不当な暗号化リソース識別情報をサーバに転送することを防止することができる。また、サーバは、暗号化ファイルをより効果的に捜し出すことが可能となる。

【００７６】

フラグは、データ所有者端末の暗号化／解読設定ユニット１０２によって最初に選択され、それから、検索者に通知してもよい。あるいは、データ所有者および検索者の両方で認識されている数字を、フラグとしてあらかじめセットすることも可能である。
他の実施例では、種々のフラグを、種々のプライバシーレベルあるいは種々のファイル用に利用することも可能である。

他の種類のパラメータおよびアルゴリズムを確認可能な解読のために本発明に適用することができることは、当業者によって十分に理解できるであろう。

（仮想削除）

【００７７】

よく知られているように、削除そのものの動作は比較的高速でかつ実行が容易であるが、１以上のファイルの削除の後に索引を更新することは、比較的複雑でありかつ一般に多くの計算機のリソースと時間を必要とする。

この点を考慮すると、暗号化されたファイルが削除された直後に、暗号化された索引を更新することは非能率的である。

索引の更新がより低い頻度では実行されることが好ましい。

例えば、更新を、１日毎、１週間毎あるいは１ヶ月毎に実行することや、所定数の暗号化ファイルが削除された後に一回実行することなどが考えられる。

また、索引の更新が使用停止の継続および影響を軽減するために予定されることが好ましい。

例えば、検索サービスにアクセスする検索者の数がより少なくなる時間（真夜中のある時間など）に索引の更新を実行するようにする。

【００７８】

しかしながら、１以上の暗号化ファイルがストレージ・サービスから削除された後、検索の正確さを保証するために、暗号化索引が更新される前に、検索結果から削除された暗号化ファイルを隠すことが必要である。このような動作を仮想削除と称する。

【００７９】

検索者に暗号化されたファイルを供給する際に一定の条件に従っていくつかのファイルをフィルタリングことによって、サーバには、本発明における仮想削除の性能が提供される。

例えば、データ所有者は、削除される暗号化されたファイルの暗号化されたリソース識別情報｛CFN_2，　CFN_4｝のリストをサーバに送信し、サーバは、対応暗号化ファイルを削除する。

その後、サーバが、検索者から、暗号化リソース識別情報｛CFN_1，　CFN_2，　CFN_3，　CFN_4，　CFN_5｝のリストを受信すると、サーバのファイル検索ユニット４０３は、まず削除されたファイルをフィルタリングする。すなわち、｛CFN_1，　CFN_2，　CFN_3，　CFN_4，　CFN_5｝－｛CFN_2，　CFN_4｝　＝　｛CFN_1，　CFN_3，　CFN_5｝のようにフィルタリングする。

その後、サーバは、単に、フィルタリング結果｛CFN_1，　CFN_3，　CFN_5｝

に対応する暗号化ファイルを捜し出し、検索者に返す。
図１１は、そのような例のデータの流れを例示している。
【００８０】
仮想削除において、削除される暗号化ファイルは、実際に削除されたのではなく、ある特殊記号によって分類される。
データ所有者からの確認指示の受信あるいは他の決められた条件が満たされた後、サーバは、暗号化ファイルの実際の削除を実行する。
【００８１】
仮想削除に加えて、フィルタリングも他の状況において適用することが可能である。フィルタリングの条件は、任意の特定用途に従って設計される。

（暗号化索引の発見と更新）
【００８２】
本発明によれば、暗号化された索引内の各ＫＩＳの拡張によって、特定のファイルに関するファイルロケータを捜し出す機能が提供される。
例えば、暗号化ファイルがサーバから削除された後、この暗号化ファイルに関するファイルロケータは暗号化索引から除去されるべきである。
本発明に従って各ＫＩＳに加えられた付加的なパラメータにより、ファイルの内容とそこに含まれているキーワードがサーバに知らせられないうちに、サーバは、データ所有者の助けを借りて指定されたファイルに関するファイルロケータを捜し出すことが可能となる。
本発明のそのような実施の形態について、図１２～図１７を参照して以下に説明する。
【００８３】
図１２は、本発明の１実施の形態によるデータ所有者端末７００の構成例を示している。
図１２に示すように、データ所有者端末７００は、図３に示したユニットをすべて含み、さらに、索引位置指標を生成するための索引位置指標生成ユニット７０１、およびファイルロケータに関連した索引ロケータを生成するための索引ロケータ生成ユニット７０２を含んでいる。
本実施の形態のキーワードユニット１０１、暗号化／解読設定ユニット１０２、ファイル暗号化ユニット１０３、ＫＩＳロケータ生成ユニット１０４およびファイルロケータ生成ユニット１０５の機能および動作は、前述した内容と同じである。
以下、本実施の形態が前述した実施の形態と相違する部分に絞って説明する。

【００８４】
本実施の形態において、暗号化された索引内の各ＫＩＳは、データ所有者端末によって生成されたファイルロケータ、対応するＫＩＳロケータおよび索引位置指標からマッピングされる索引ロケータを各ファイルロケータに加えることにより拡張される。
【００８５】
特に、索引付けする工程において、データ所有者端末７００の索引位置指標生成ユニット７０１は、ファイルの暗号化リソース識別情報を一意の値にマッピングすることにより、各ファイルの索引位置指標を生成する。
例えば、ファイル$FILE_j$について、索引位置指標生成ユニット７０１は、以下のように、索引位置指標x_jを生成する。

$$x_j = Hash(CFN_j || sk) \qquad (式5)$$

ここで、CFN_jはファイル$FILE_j$の暗号化リソース識別情報であり、ｓｋは、データ所有者によって保持された秘密鍵（例えば、データ所有者によって保持されたプライ

ベート鍵）である。
上述したように、どのような単方向のマッピング方法でもハッシュ関数の代わりに利用することが可能である。
【００８６】
ＫＩＳロケータおよびファイルロケータに加えて、この実施の形態によるデータ所有者端末７００は、索引ロケータ生成ユニット７０２によって、さらに、ＫＩＳに含まれる各ファイルロケータについて索引ロケータを生成する。
各索引ロケータは、対応するファイルロケータ、ＫＩＳロケータおよび索引位置指標生成ユニット７０１によって生成された索引位置指標の組み合わせをある値にマッピングすることにより生成される。
例えば、ＫＩＳロケータKL_iを有するＫＩＳ内のファイル$FILE_j$に関するファイルロケータ$FL_{j,m}$について、索引ロケータ生成ユニット７０２は、以下のように、索引ロケータ$IL_{i,j,m}$を生成する。

$$IL_{i,j,m} = Hash(KL_i||FL_{j,m}||x_j) \quad （式６）$$

ここで、x_jはＦＩＬＥｊのための索引位置指標であり、それは索引位置指標生成ユニット７０１によって生成される。

【００８７】
その後、データ所有者端末７００の索引形成ユニット１０６は、各々がＫＩＳロケータを含む１つ以上のＫＩＳと、上記の実施の形態で生成された１つ以上のファイルロケータと、各々が対応するファイルロケータを伴う１つ以上の索引ロケータによって暗号化索引を形成する。
図１とテーブル１に示された状況を例にとり、そして、ファイルＲｅｓｅａｒｃｈ．ｐｐｔおよびＮｏｖｅｌ．ｐｄｆがキーワードKW_aに関係していると想定すると、本実施の形態によれば、キーワードKW_aに対するＫＩＳは、要素＜KL_a：$FL_{Research.ppt,3}$，$IL_{a,Research.ppt,3} = Hash(KL_a||FL_{Research.ppt,3}||x_{Research.ppt})$，$FL_{Novel.pdf,1}$，$IL_{a,Novel.pdf,3} = Hash(KL_a||FL_{Novel.pdf,3}||x_{Novel.pdf})$，$FL_{Novel.pdf,2}$，$IL_{a,Novel.pdf,3} = Hash(KL_a||FL_{Novel.pdf,3}||x_{Novel.pdf})$＞として生成される。
このように生成された暗号化索引は、サーバに送られ、サーバ上に格納される。
【００８８】
本実施の形態による索引付け工程のデータの流れが図１３に概念的に示されている。
【００８９】
暗号化ファイルが削除された後に暗号化索引を更新する処理について以下に説明する。
【００９０】
図１４は、本実施の形態によるサーバの構成例を示す図である。図１４に示すように、サーバ８００は、図７に示したユニットをすべて含み、さらに、格納された暗号化索引の更新のために索引更新ユニット８０１を含む。
本実施の形態内の記憶ユニット４０１、索引検索ユニット４０２およびファイル検索ユニット４０３の機能と動作は、前述した内容と同じである。
以下、本実施の形態が前述した実施の形態と相違する部分に絞って説明する。
【００９１】
図１５は、暗号化ファイルが削除された後に暗号化索引を更新するサーバの処理の例を

示すフローチャートである。

【００９２】

暗号化ファイルＦＩＬＥ$_a$が暗号化索引から削除される場合、例えば、ファイルＦＩＬＥ$_a$がサーバ上のストレージサービスから削除され、索引を更新する必要がある場合、データ所有者端末７００は、索引位置指標生成ユニット７０１によって計算したファイルＦＩＬＥ$_a$の索引位置指標ｘ$_a$を含むメッセージをサーバ８００に送信する。

ステップＳ９０１で、サーバ８００はデータ所有者端末８００から索引位置指標ｘ$_a$を受信する。

【００９３】

その後、格納された暗号化索引内の各ＫＩＳ内の各ファイルロケータについて、サーバ８００の索引更新ユニット８０１は、暗号化索引を生成するのにデータ所有者端末によって利用される同じマッピング方法で、受信した索引位置指標ｘ$_a$を使用することにより索引ロケータを計算する。

ＫＩＳロケータＫＬ$_i$を有するＫＩＳ内のファイルロケータＦＬ$_{j,\ m}$について、索引更新ユニット８０１は、前述と同じハッシュ関数を利用することにより、

$$ＩＬ_{i,\ j,\ m} ＝ Ｈａｓｈ （ＫＬ_i ｜｜ ＦＬ_{j,\ m} ｜｜ ｘ_a）$$

を計算する。

その後、索引更新ユニット８０１は、計算されたＩＬ$_{i,\ j,\ m}$がＫＩＳを含まれるファイルロケータＦＬ$_{j,\ m}$を伴う索引ロケータＩＬ$_{i,\ j,\ m}$と等しいかどうかチェックする。

２つの値がマッチする場合、それは対応ファイルロケータを削除すべきであることを示している。

これにより、ステップＳ９０２で、索引更新ユニット８０１は、削除すべきファイルロケータをすべて見つけ出す。

【００９４】

その後、ステップＳ９０３で、サーバ８００の索引更新ユニット８０１は、暗号化索引を更新するために、記憶ユニット４０１に格納された暗号化索引から附随する索引ロケータだけでなくマッチするファイルロケータをすべてを削除する。

【００９５】

上述したような暗号化索引の更新におけるデータの流れが図１６に概念的に示されている。

【００９６】

上記の実施例において、サーバは、暗号化索引内のすべてのＫＩＳｅｓにおけるファイルロケータをチェックする。

あるいは、データ所有者は、検索範囲をマッチするＫＩＳロケータを有するＫＩＳｅｓに縮小してサーバを支援するために、削除されたファイルに関するすべてのＫＩＳｅｓのＫＩＳロケータを送信する。

【００９７】

ファイルに関するＫＩＳｅｓのＫＩＳロケータは、索引付けする工程においてデータ所有者端末にはじめから格納されていてもよいし、あるいは、データ所有者端末が、各ファイルのキーワードの情報をあらかじめ保持し、更新する工程においてＫＩＳロケータを計算してもよい。

また、暗号化されたファイルがサーバから削除される前に、データ所有者が、暗号化リソース識別情報によって識別された暗号化ファイルを取り出し、暗号化ファイルを解読し、解読されたファイルからキーワードを抽出し、削除すべきファイルに関するＫＩＳロケータを計算しサーバに送ることも可能である。

その場合、データ所有者端末は、また検索者端末として動作し、図８に示されるような関連するユニットを含む。

【００９８】

データ所有者端末からＫＩＳロケータおよび索引位置指標を取得すると、サーバは、受

信したＫＩＳロケータによって識別されるＫＩＳ内のファイルロケータを単にチェックすることも可能である。
それにより、計算量が非常に軽減される。
【００９９】
本実施例の暗号化索引の更新のデータの流れが図１７に概略的に示されている。
【０１００】
上述したのは索引からファイルを削除する例である。
本発明によれば、暗号化索引も、後で１以上のファイルを追加する時に容易に更新することが可能である。
例えば、暗号化された索引が設定された後、データ所有者がストレージサービスに追加の暗号化ファイルを追加すれば、データ所有者端末は、上述した方法と同じ方法で新しく追加したファイルと関連したＫＩＳロケータおよびファイルロケータ（索引ロケータを伴って、あるいは索引ロケータなしで）を計算し、サーバにそれらを送信する。
サーバでは、索引検索ユニット４０２が、受信したＫＩＳロケータに対応するＫＩＳを捜し出し、また、索引更新ユニット８０１は、対応するＫＩＳに受信したファイルロケータ（索引ロケータを伴って、あるいは索引ロケータなしで）を単に追加することにより、暗号化索引を更新する。
これにより、追加ファイルの情報が更新された索引に組み入れられる。

（きめの細かい認証）
【０１０１】
プライバシーレベルに関連して、そして特定のキーワードと無関係に、ファイルロケータ生成／解読鍵の各ペアが生成されることは、上述した実施の形態で説明した。
発行されたファイルロケータ解読鍵を持つ検索者が、データ所有者によって当該検索者に発行されていないＫＩＳロケータを取得すると、検索者は、さらにこのＫＩＳロケータによる検索を実行し、対応するＫＩＳ内のファイルロケータを解読することができてしまうという懸念がある。
【０１０２】
本発明の１実施の形態によれば、認証の管理を高めるために、ファイルロケータ生成／解読鍵の各ペアを、プライバシーレベルと特定のキーワードの両方に関連して生成することが可能である。
例えば、ファイルロケータ生成／解読鍵は、キーワードKW_iとプライバシーレベルｍに関連して以下のように生成することが可能である：

$$EKey_{i,m} = DKey_{i,m} = Hash(MEK||KW_i||m)$$
（式７）

あるいは、ファイルロケータ生成／解読鍵は、対応するキーワードと鍵の少なくとも１つの組み合わせを一意の値にマッピングする他のアルゴリズムによって生成することが可能である。
このような拡張したファイルロケータ生成／解読鍵によれば、きめの細かい権限付与の制御が、プライバシーレベルだけでなくキーワードに基づいて提供される。
【０１０３】
このような実施の形態によれば、各ファイルのファイルロケータは、ファイルとそのファイルを見せることができるプライバシーレベルに関連付けられたキーワードに関する各拡張ファイルロケータ生成鍵でファイル取得情報を暗号化することにより、索引付けする工程において生成される。
【０１０４】
ファイル$FILE_j$のファイル取得情報が$CFN_j||K_{file_j}$の書式をとると想定すると、ファイルロケータを計算するための特定のアルゴリズムは、上述した式３と比

較して下のように与えられる。
すなわち、ファイル$FILE_j$とそのファイル$FILE_j$を見ることができるプライバシーレベルmに関連付けらたキーワードKW_iについて、ファイル$FILE_j$に対するファイルロケータ$FL_{i,j,m}$は、以下のように生成される。

$$FL_{i,j,m} = E(EKey_{i,m}, CFN_j || K_{filej}) \quad (式8)$$

【0105】
このような実施の形態によれば、キーワードのＫＩＳはそれぞれ、そのキーワードに関する拡張ファイルロケータ生成鍵で生成されたファイルロケータをすべて含む。
すなわち、ファイルのすべてのファイルロケータの中で、特定のキーワードに関する拡張ファイルロケータ生成鍵で生成されたものだけが、そのキーワードのＫＩＳに加えられ、他のキーワードに関する拡張ファイルロケータ生成鍵で生成されたものは加えられない。
このことは、そのキーワードに関する適正な拡張ファイルロケータ解読鍵を所有しなければ、誰もキーワードのＫＩＳ内のファイルロケータを正確に解読することができないことを保証する。
他の処理は、前述の実施の形態に述べた処理の内容と同じである。
【0106】
検索工程において、データ所有者が検索者がキーワードについて検索することを可能にしたければ、データ所有者は、適切なプライバシーレベルに対応する拡張ファイルロケータ解読鍵とキーワードのＫＩＳロケータを安全な方法で検索者に発行する。
検索者による拡張ファイルロケータ解読鍵の使用については、上記の実施の形態に述べられたファイルロケータ解読鍵の使用と同じである。
【0107】
この実施の形態によれば、各拡張ファイルロケータ解読鍵は、各検索者で秘密に保持され、サーバに示されないであろう。
従って、ＫＩＳロケータが他のものに示されても、他のキーワードに関する任意のファイルロケータ解読鍵で対応するＫＩＳ内のファイルロケータを解読することはできない。
【0108】
確認可能な解読、仮想削除、発見と更新のような本発明の他の特徴は、同様にこの実施の形態に適用することが可能である。
ファイルロケータ生成／解読鍵が拡張ファイルロケータ生成／解読鍵に置き換えられることを除いて。処理は基本的に同じである。
【0109】
また、本発明はプライバシーレベルを区別する必要がないという場合にも適用可能であることに注目すべきである。その場合、ファイルロケータ生成／解読鍵は異なるキーワードに関連して生成される。例えば、ファイルロケータ生成／解読鍵は、以下のように生成される。

$$EKey_i = DKey_i = Hash(MEK || KW_i) \quad (式9)$$

【0110】
インデックス付け、検索および更新の処理については、上述した処理と同様である。特定の処理は、ただ１つのプライバシーレベルがあると想定することで理解できるので、その説明は繰り返されない。

（連鎖認証）
【0111】
上記の実例となる実施の形態において、様々なプライバシー・レベルのファイルロケー

タ生成／解読鍵は、異なるパラメータで別々に生成され、互いとの計算上の関係を有しない。

【０１１２】

実際には、異なるプライバシーレベル間に支配関係がある可能性はある。すなわち、より高いプライバシーレベルがより低いプライバシーレベルを支配する。言いかえれば、あるプライバシーレベルでの検索において、そのプライバシーレベルより低いプライバシーレベルで支配されたファイルと、そのプライバシーレベルで支配されるが、他のより低いプライバシーレベルでは支配されないファイルを検索することが可能である。例えば、データ所有者ボブは、ファイルの検索を実行する検索者を、種々の関係に従って種々のレベルに分類する。例えば、家族は最も高いプライバシーレベル（１レベル）を有し、親しい友達は中間のプライバシーレベル（２レベル）を有し、また、共通の友達は最低のプライバシーレベル（３レベル）を有する。同時に、ファイル上の検索の性能は、低いプライバシーレベルで支配されたファイルはすべてより高いプライバシーレベルでも支配されるという規則に従う。すなわち、共通の友達によって検索できるファイルは全て親しい友達および家族によって検索することができ、一方、親しい友達によって検索できるファイルはすべて家族によって検索することができる。

【０１１３】

本発明において、連鎖認証は、認証と管理をより簡単かつ効率的に行うといった状況で利用される。本発明によって連鎖認証に適用される１実施例を以下に説明する。

【０１１４】

ｎのプライバシーレベルがあり、最も高いプライバシーがレベル１であり、また、プライバシーレベルｍが他のより低いプライバシーレベル（プライバシーレベルｍ＋１、・・・、ｎ、ここで、ｍはｎ未満の自然数である）を支配するということが想定される。

【０１１５】

この実施の形態によれば、索引付け工程においてファイルロケータ生成／解読鍵をセットする際に、データ所有者は、ハッシュ関数を利用することにより、最も高いプライバシーレベルでファイルロケータ生成／解読鍵をまずセットする。

例えば、次のものに続くように、ファイルロケータ生成鍵$EKey_1$および最も高いプライバシーレベルのファイルロケータ解読鍵$DKey_1$は、以下のように生成される。

$$EKey_1 = DKey_1 = H^1(z) \qquad (式１０)$$

ここで、$H^1(z)$は１回のハッシュ演算（Ｈａｓｈ（ｚ））を表わす、また、ｚは任意の文字列（例えば、ＭＥＫ、ＭＥＫと任意数の組み合わせ、ＭＥＫ｜｜KW_iなど）である。

好ましくは、ｚはデータ所有者によって簡単に記憶され或いは引き出される文字列である。

【０１１６】

その後、他のプライバシーレベルのファイルロケータ生成／解読鍵は、$EKKey_1$と$DKey_1$に基づいてハッシュチェーン方式で生成される。

特に、ファイルロケータ生成鍵$EKey_m$およびプライバシーレベルｍのファイルロケータ解読鍵$DKey_m$は、以下のように生成される。

$$EKey_m = DKey_m = H^m(z) \qquad (式１１)$$

ここで、$H^m(z)$は、ｍ回のハッシュ演算（

【数１】

$$\underbrace{Hash(Hash...Hash(z)...)}_{m}$$

）を示す。

【０１１７】
すなわち、ファイルロケータ生成鍵ＥＫｅｙ$_{m}$およびプライバシーレベルｍのファイルロケータ解読鍵ＤＫｅｙ$_{m}$は、次の漸化式によって生成される。

ＥＫｅｙ$_{m}$ ＝ ＤＫｅｙ$_{m}$ ＝ Ｈａｓｈ（ＥＫｅｙ$_{m-1}$）＝ Ｈａｓｈ（ＤＫｅｙ$_{m-1}$）　（式１２）

【０１１８】
上記の計算は、例えばデータ所有者端末の暗号化／解読設定ユニットによって実行される。

【０１１９】
認証する場合、データ所有者はそれぞれのレベルで検索者に異なるプライバシーレベルのファイルロケータ解読鍵を発行する。他の処理は上述した実施の形態のそれと同様である。

【０１２０】
ＤＫｅｙ$_{m}$が発行されたプライバシーレベルｍの検索者は、より低いプライバシーレベルでファイルロケータを解読するために、データ所有者よって知られ或いは公表されているハッシュアルゴリズムに従って、より低いプライバシーレベルのファイルロケータ解読鍵を容易に判別することが可能であることが理解できる。ハッシュ関数の非可逆特性のために、プライバシーレベルｍの検索者は、より高いプライバシーレベルのファイルロケータ解読鍵を判別することができない。そして、これにより、非可逆連鎖認証が保証される。

【０１２１】
上記の実施の形態の連鎖認証では、任意のプライバシーレベルの検索者は、より低いプライバシーレベルの性能を取得するために、計算によって、任意のより低いプライバシーレベルのファイルロケータ解読鍵を取得することができる。これにより、簡単で便利な連鎖認証が実現される。

【０１２２】
本発明に適用可能な連鎖認証の方法は、前述のハッシュ・チェーン・アルゴリズムに制限されないが、任意の非可逆認証技術である。例えば、Ｆｏｒｗａｒｄ Ｋｅｙ Ｒｏｔａｔｉｏｎ（ＦＫＲ）ｔｅｃｈｎｏｌｏｇｙ ｐｒｏｐｏｓｅｄ ｂｙ マヘシュ・カラハラ他、「Ｐｌｕｓｔｕｓ：信頼されていない安全なファイル共有ストレージを拡張」、カナダ、バークレイ、ユーズニックス協会によって公開のファイルと記憶の技術についての第２回会議（２００３年3月３１日～４月２日、カナダ、サンフランシスコ）の議事録、２９－４２頁（Ｆｏｒｗａｒｄ Ｋｅｙ Ｒｏｔａｔｉｏｎ（ＦＫＲ）ｔｅｃｈｎｏｌｏｇｙ ｐｒｏｐｏｓｅｄ ｂｙ Ｍａｈｅｓｈ Ｋａｌｌａｈａｌｌａ, ｅｔｃ. ｉｎ “Ｐｌｕｓｔｕｓ: Ｓｃａｌａｂｌｅ ｓｅｃｕｒｅ ｆｉｌｅ ｓｈａｒｉｎｇ ｏｎ ｕｎｔｒｕｓｔｅｄ ｓｔｏｒａｇｅ”, ｉｎ ｔｈｅ Ｐｒｏｃｅｅｄｉｎｇｓ ｏｆ ｔｈｅ ２ｎｄ Ｃｏｎｆｅｒｅｎｃｅ ｏｎ Ｆｉｌｅ ａｎｄ Ｓｔｏｒａｇｅ Ｔｅｃｈｎｏｌｏｇｉｅｓ（ＦＡＳＴ’０３）, ｐｐ. ２９－４２（３１ Ｍａｒ － ２ Ａｐｒ ２００３, Ｓａｎ Ｆｒａｎｃｉｓｃｏ, ＣＡ）, ｐｕｂｌｉｓｈｅｄ ｂｙ ＵＳＥＮＩＸ, Ｂｅｒｋｅｌｅｙ, ＣＡ）において提案される技術を利用することができる。本発明の他の実施の形態では、その

ような技術が適用される。

【０１２３】

e_0がデータ所有者の公開鍵であり、また、d_0がデータ所有者の秘密鍵であると想定する。データ所有者は、公開鍵e_0を公表し、d_0を秘密に保持する。

【０１２４】

索引付けする工程においてファイルロケータ生成／解読鍵をセットする際に、データ所有者は、任意の整数$k_0 \in Z_p^*$を選択し、最低のプライバシーレベルｎのファイルロケータ生成鍵$EKey_n$および ファイルロケータ暗号鍵$DKey_n$を以下のようにセットする。

$$EKey_n = DKey_n = k_0^{d_0} \quad \text{（式１３）}$$

【０１２５】

他のプライバシーレベルｍ（ｍはｎ未満の自然数である）のファイルロケータ生成および解読キーは、次の漸化式に従って計算される。

$$EKey_m = DKey_m = (EKey_{m+1})^{d_0} = (DKey_{m+1})^{d_0} = \quad \text{（式１４）}$$

【０１２６】

上述した計算は、例えば、データ所有者端末の暗号化／解読設定ユニットによって実行される。

【０１２７】

認証する場合、データ所有者はそれぞれのレベルで検索者に異なるプライバシーレベルのファイルロケータ解読鍵を発行する。

$DKey_m$が発行されたプライバシーレベルｍの検索者は、以下の漸化式によってデータ所有者によって公表された公開鍵e_0に従って他のより低いプライバシーレベルのファイルロケータ解読鍵を容易に判別することが可能である。

$$Dkey_{l+1} = (DKey_l)^{e_0}, \quad l = m, \cdots, n-1 \quad \text{（式１５）}$$

【０１２８】

上述した計算は、例えば、検索者端末のファイルアロケータ解読ユニットによって実行される。

【０１２９】

一方、プライバシーレベルｍでの検索は、より高いプライバシーレベルのファイルロケータ解読鍵を判別することができない。それにより、非可逆認証が実現される。

（変形例）

【０１３０】

本発明によるいくつかの特定の実施の形態について、図面を参照して説明した。しかしながら、本発明は、上記の実施の形態に述べられた特定の構成および処理に限定されるものではない。上述した構成、アルゴリズム、動作及び処理について、本発明の精神の範囲内で、種々の代替案、変更あるいは変形が可能であることは、当業者によって十分に理解できるであろう。

【０１３１】

例えば、キーワードがそれぞれ暗号化された逆索引内の１つのＫＩＳを有すること、また、各ＫＩＳのＫＩＳロケータがキーワードにユニークに対応するものとして生成されることは、上記の典型的な実施の形態に述べられている。

しかしながら、各ＫＩＳがキーワードだけでなくプライバシーレベル（すなわち、ファイルロケータ生成或いは解読鍵）に対応するように、索引を生成することも可能である。
すなわち、同じプライバシーレベルのファイルであって、同じキーワードに関係するファイルは、１つのＫＩＳに索引付けされ、また、異なるプライバシーレベルのファイルは、このファイルが同じキーワードに関連するかどうかに関係なく、異なるＫＩＳに索引付けされる。
言い換えれば、ＫＩＳはそれぞれ、ただ１つのファイルロケータ生成（あるいは解読）鍵および１つのキーワードに対応する。
その場合、キーワードKW_iとファイルロケータ生成鍵$EKey_m$（あるいは、ファイルロケータ解読鍵$DKey_m$）に対応するＫＩＳのＫＩＳロケータ$KL_{i,m}$は、以下のように生成される。

$$KL_{i,m} = E(EKey_m, KW_i) \quad （式１６）$$

あるいは

$$KL_{i,m} = E(DKey_m, KW_i) \quad （式１７）$$

【０１３２】
本発明は、図面中で示される特定の構成および処理に限定されない。上述したような本発明の種々の形態を具体化する実施例は、その用途に応じて組み合わせることが可能である。
例えば、暗号化された索引は、解読の正確さを確認するためのフラグおよびファイルロケータを捜し出すための索引ロケータの両方を含み、また、データ所有者端末、サーバおよび検索者端末は、２つの場合に対応する構成部品を含むことが可能である。
【０１３３】
さらに、前述した処理の順序については、合理的な範囲で変更することが可能である。
例えば、図４に示されるステップＳ２０１およびＳ２０２の順序については、逆にすることが可能であるし、これらのステップを同時に実行することも可能である。
【０１３４】
この説明において利用される「ファイル」は、広い意味に解釈されるべきである。また、それは、例えば、テキストファイル、ビデオ／オーディオ・ファイル、画像／図表、およびその他のデータあるいは情報を含むが、それに限定されるものではない。
【０１３５】
データ所有者端末、検索者端末およびサーバの典型的な構成と、相互につながれたいくつかのユニットが、図面中で示されている。これらのユニットは、相互間で信号を送信するために、バスあるいは他の信号ライン、あるいは任意の無線接続によって互いに接続することが可能である。しかしながら、各装置の中に含まれる構成部品は、上述したユニットに制限されていない。また、特定の構成は修正され、或いは交換することが可能である。
各装置はそれぞれ、装置のオペレータに情報を表示するためのディスプレイ装置、オペレータの入力を受信するための入力装置、各ユニットの動作を制御するためのコントローラ、任意の必要なストレージ手段などのような他のユニットを含むことが可能である。そのような構成部品はこの技術分野において既に知られているので、それらについての細に述べない。また、当業者であれば、装置に上記のようなユニットを追加することは容易に理解できるであろう。
さらに、上述したユニットは、図面内で個別のブロックとして示されているが、それらのうちのどれでも他のユニットと組み合わせて１つの構成部品とすることが可能であり、あるいはそれらをいくつかの構成部品に分割することが可能である。
例えば、図３に示されるＫＩＳロケータ生成ユニット、ファイルロケータ生成ユニットおよび索引形成ユニットは、相互に組み合わせて索引作成ユニットとすることが可能である。

あるいは、上に述べられた暗号化／解読設定ユニットは、暗号化／解読のための鍵を選択するためのユニットと、他の保護パラメーターを選択するためのユニットに分割することが可能である。

【０１３６】

さらに、データ所有者端末、検索者端末およびサーバは、上記の実施例において個別の装置として示され説明された。それらは通信ネットワークにおいて互いに遠隔に位置することもできる。しかしながら、それらを、強化機能のために１つの装置として組み合わせることも可能である。

例えば、ある場合にはデータ所有者端末として機能し、他のある場合には検索者端末として検索を実行する新たな装置を生成するために、データ所有者端末と検索者端末を組み合わせることも可能である。

他の例では、サーバとデータ所有者端末あるいはサーバと検索者端末は、それらがある適用例において２つの役割を実行するならば、互いに組み合わせることができる。また、装置は、異なる処理でデータ所有者端末、検索者端末およびサーバの役割を実行できるように作成することも可能である。

【０１３７】

上述した通信ネットワークは、電気通信ネットワークあるいはコンピュータネットワークを含む何れかの種類のネットワークである。また、例えば、データ所有者端末、検索者端末およびサーバが単一の装置の部品として実現される場合、通信ネットワークは、データバスあるいはハブなどの内部データ転送メカニズムも含むことも可能である。

【０１３８】

本発明に要素は、ハードウェア、ソフトウェア、ファームウェアあるいはそれの組み合わせにおいて実現され、それのシステム、サブシステム、構成部品あるいはサブコンポーネントにおいて利用することができる。ソフトウェア中で実現された時、本発明に要素は、必要なタスクを実行するためのプログラム、あるいはコードセグメントである。プログラムまたはコードセグメントは、コンピュータ読み取り可能な媒体に格納するか、あるいは伝送ケーブルか通信リンク上の搬送波に包含されたデータ信号によって送信することが可能である。コンピュータ読み取り可能な媒体には、情報を格納するか転送することが可能であるすべての媒体を含む。コンピュータ読み取り可能な媒の具体例は、電子回路、半導体記憶装置、ＲＯＭ、フラッシュ・メモリー、消去可能ＲＯＭ（ＥＲＯＭ）、フロッピー・ディスク、ＣＤ－ＲＯＭ光ディスク、ハードディスク、光ファイバー媒体、無線周波数（ＲＦ）リンクなどを含む。コードセグメントは、インターネット、イントラネットなどのようなコンピュータネットワークを経由してダウンロードすることも可能である。

【０１３９】

本発明は、本発明の精神および本質的な機能から外れずに、他の特定の形態で実装可能である。例えば、特徴が本発明に基本的な範囲から外れない限り、特定の実施の形態で述べられたアルゴリズムは修正することが可能である。従って、現在の実施の形態は、全ての点において例示でありかつ限定的でないとして考慮されるべきである。本発明の範囲は、前述の説明によってではなく添付された請求項によって示される。また、したがって、請求項と同等の意味と範囲の内で生ずる変更は全て本発明の範囲に包含される。

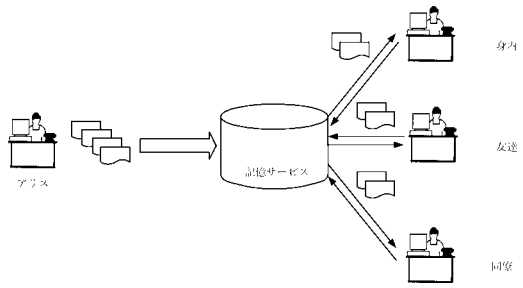
【符号の説明】

【０１４０】

１０１：キーワードユニット
１０２：暗号／解読設定ユニット
１０３：ファイル暗号化ユニット
１０４：ＫＩＳロケータ生成ユニット
１０５：ファイルロケータ生成ユニット
１０６：索引形成ユニット
４００：サーバ
４０１：記憶ユニット

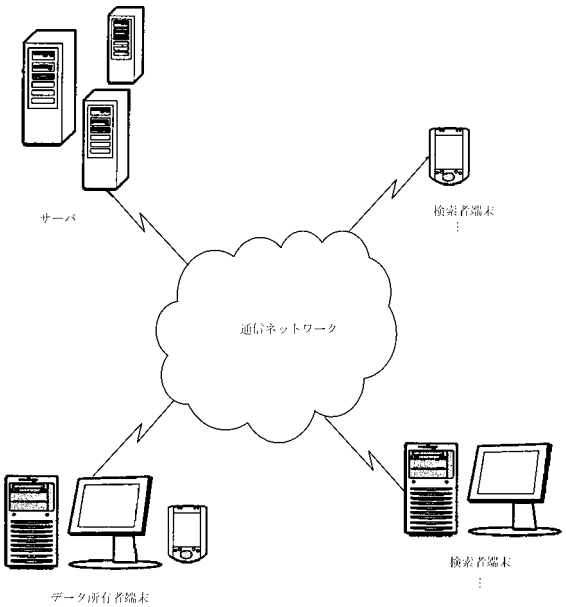
４０２：索引検索ユニット
４０３：ファイル検索ユニット
５００：検索者端末
５０１：検索要求ユニット
５０２：ファイルアロケータ解読ユニット
５０３：ファイル取得ユニット
５０４：ファイル解読ユニット
７０１：索引位置指標生成ユニット
７０２：索引ロケータ生成ユニット
８０１：索引更新ユニット

【図１】

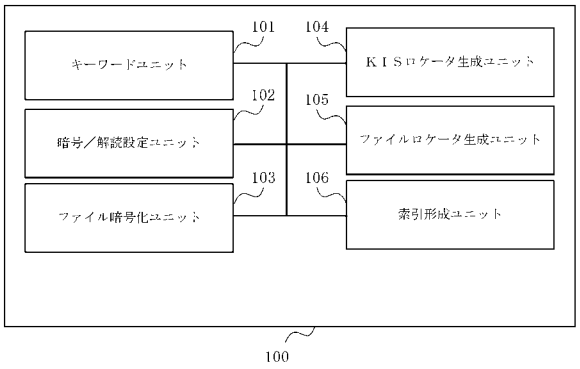


ファイル名	身内	友達	同僚
Research.ppt	No	No	Yes
Novel.pdf	Yes	Yes	No
Pets.jpg	Yes	Yes	Yes
Financial.doc	Yes	No	No

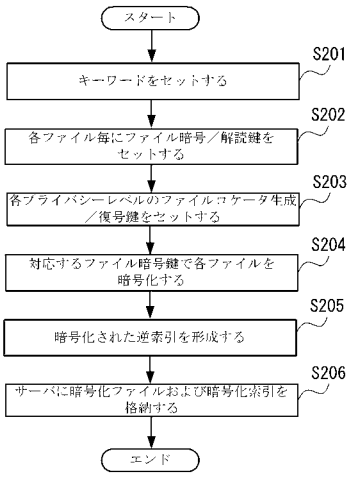
【図２】



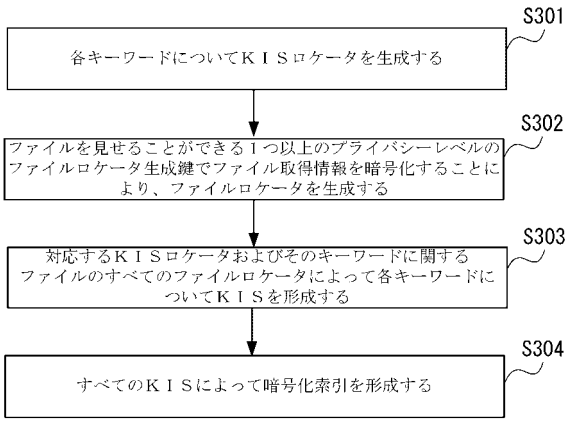
【図３】



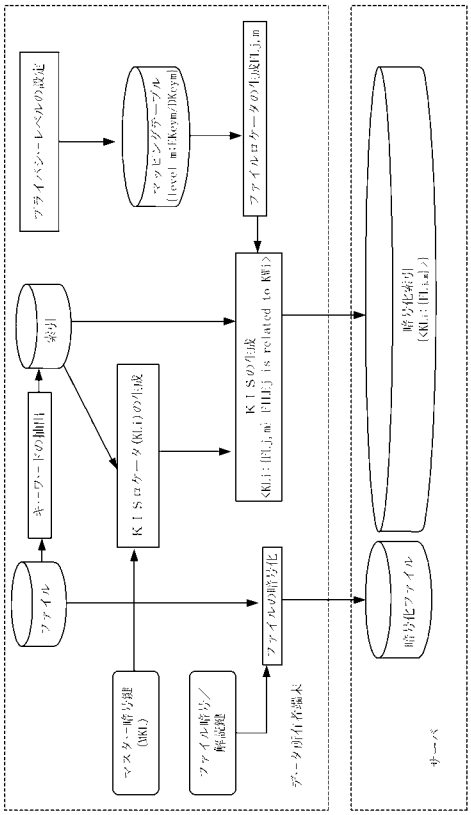
【図４】



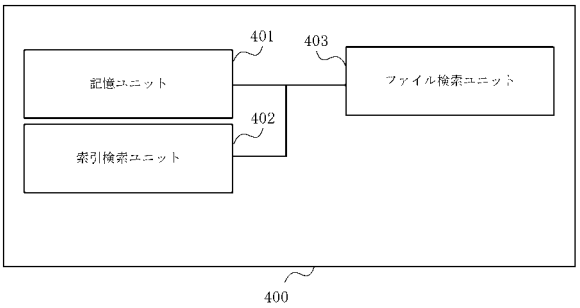
【図５】



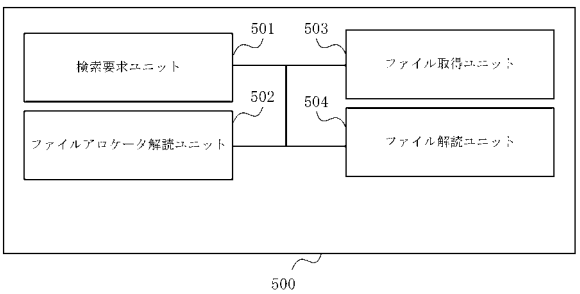
【図６】



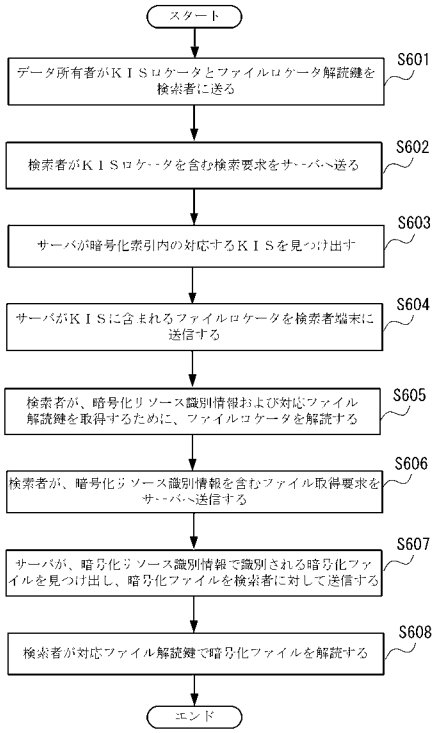
【図７】



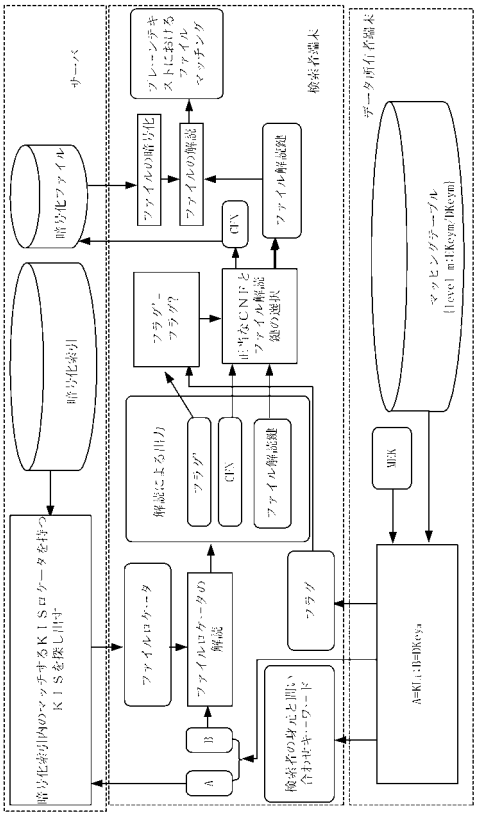
【図８】



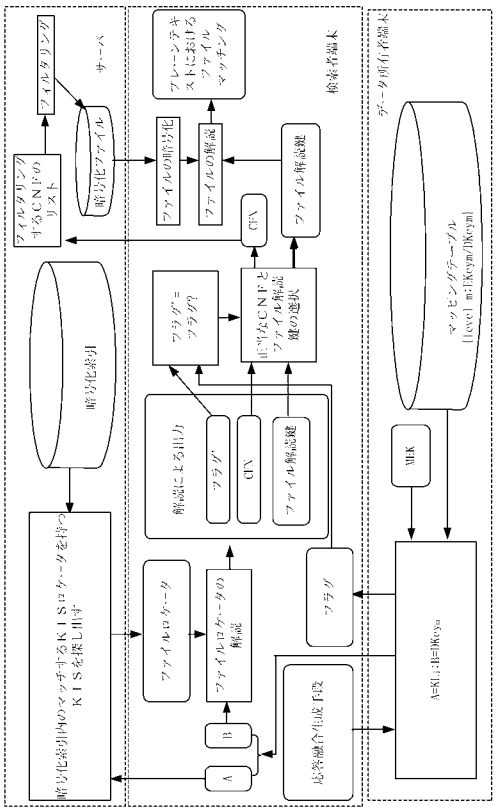
【図９】



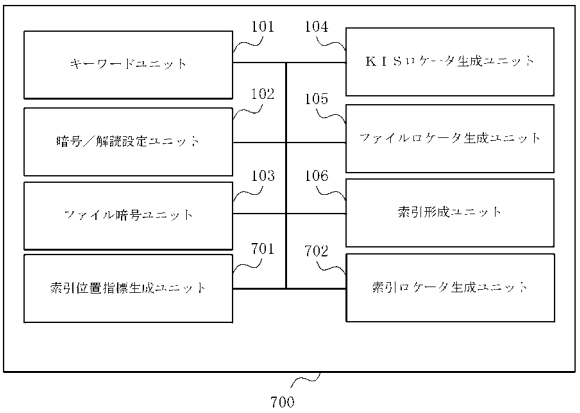
【図１０】



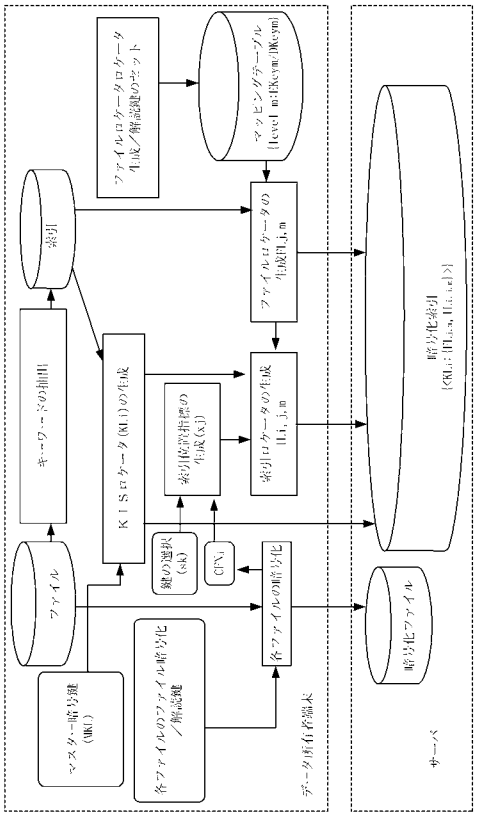
【図１１】



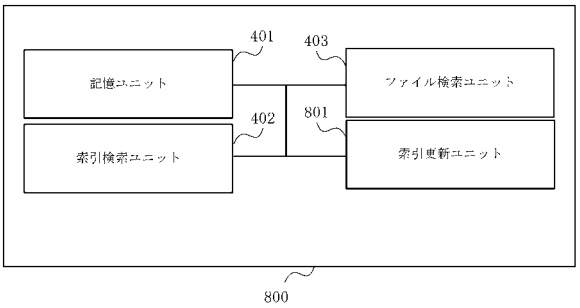
【図１２】



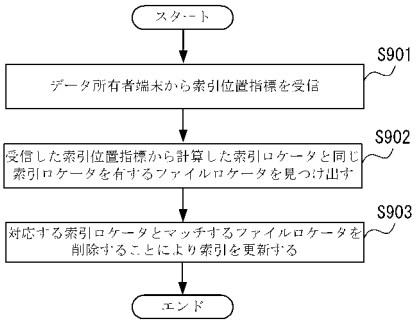
【図１３】



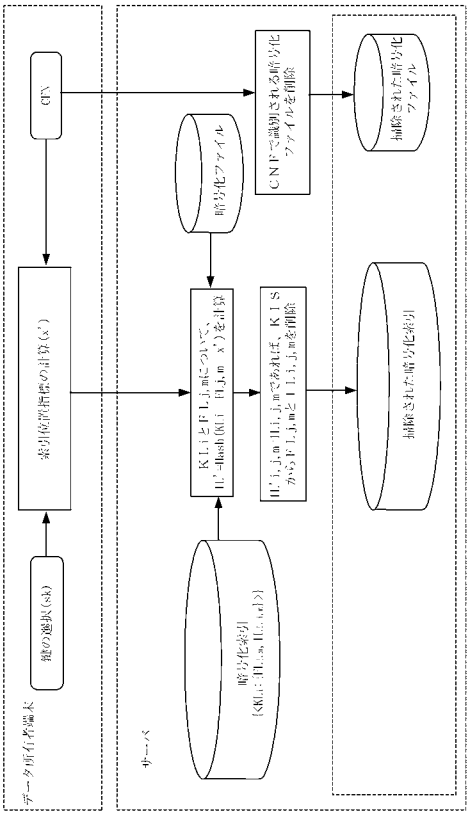
【図１４】



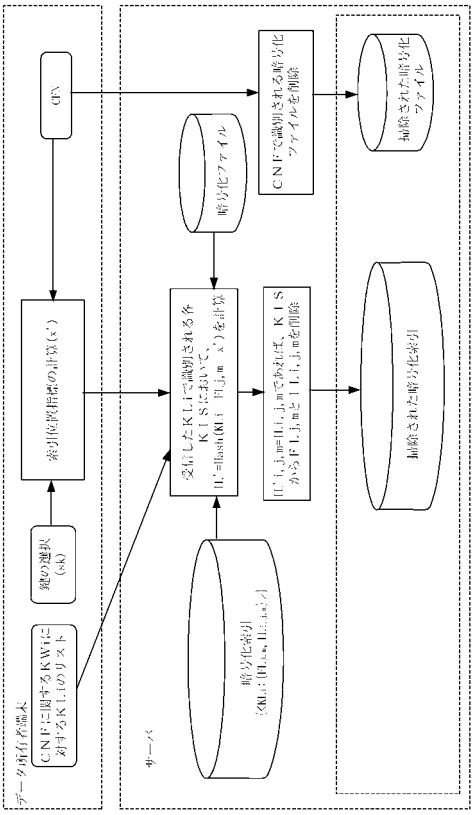
【図１５】



【図１６】



【図１７】



フロントページの続き

(72)発明者　テン　イェ
中華人民共和国　１０００８４　ベイジン，　チンフア　サイエンス　パーク，　イノベーション　プラザ，　ビルディング　エー，　１１エフ
(72)発明者　ゼン　カ
中華人民共和国　１０００８４　ベイジン，　チンフア　サイエンス　パーク，　イノベーション　プラザ，　ビルディング　エー，　１１エフ
(72)発明者　ワン　リーミン
中華人民共和国　１０００８４　ベイジン，　チンフア　サイエンス　パーク，　イノベーション　プラザ，　ビルディング　エー，　１１エフ
(72)発明者　フクシマ　トシカズ
中華人民共和国　１０００８４　ベイジン，　チンフア　サイエンス　パーク，　イノベーション　プラザ，　ビルディング　エー，　１１エフ

Ｆターム(参考)　5B017 AA03 BA07 CA16
5B075 KK53 KK54 NK06
5J104 AA16 AA32 EA04 EA15 EA16 JA03 MA05 NA02 NA27 NA37
PA14

【外国語明細書】
2010061103000001.pdf