

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7511630号
(P7511630)

(45)発行日 令和6年7月5日(2024.7.5)

(24)登録日 令和6年6月27日(2024.6.27)

(51)国際特許分類		F I	
G 0 6 F	16/174 (2019.01)	G 0 6 F	16/174
G 0 9 C	1/00 (2006.01)	G 0 9 C	1/00 6 6 0 D
G 0 6 F	21/64 (2013.01)	G 0 9 C	1/00 6 6 0 F
G 0 6 F	21/62 (2013.01)	G 0 9 C	1/00 6 6 0 G
G 0 6 F	21/60 (2013.01)	G 0 6 F	21/64
請求項の数 13 (全39頁) 最終頁に続く			
(21)出願番号	特願2022-507917(P2022-507917)	(73)特許権者	390009531
(86)(22)出願日	令和2年8月10日(2020.8.10)	インターナショナル・ビジネス・マシー ンズ・コーポレーション	
(65)公表番号	特表2022-545351(P2022-545351 A)	INTERNATIONAL BUSI NESS MACHINES CORPO RATION	
(43)公表日	令和4年10月27日(2022.10.27)	アメリカ合衆国10504 ニューヨー ク州 アーモンク ニュー オーチャード ロード	
(86)国際出願番号	PCT/IB2020/057505	New Orchard Road, A rmonk, New York 105 04, United States of America	
(87)国際公開番号	WO2021/033072	(74)代理人	100112690
(87)国際公開日	令和3年2月25日(2021.2.25)	弁理士 太佐 種一	
審査請求日	令和5年1月20日(2023.1.20)	最終頁に続く	
(31)優先権主張番号	16/544,708		
(32)優先日	令和1年8月19日(2019.8.19)		
(33)優先権主張国・地域又は機関	米国(US)		

(54)【発明の名称】 データ重複排除のための不透明な暗号化

(57)【特許請求の範囲】

【請求項1】

コンピュータによって実行されるコンピュータ実装方法であって、前記コンピュータが、ストレージ・システムで重複排除情報を受信することであって、前記重複排除情報が、動作を実行するために前記ストレージ・システムによってアクセス可能であり、前記重複排除情報が、クライアント・データの一部に関連付けられたシグネチャを含む、前記受信することと、

第1のクライアントの秘密鍵を使用して暗号化された前記クライアント・データを受信することであって、前記第1のクライアントの秘密鍵が前記ストレージ・システムには使用することができない、前記受信することと、

前記クライアント・データのチャンクに対して、前記ストレージ・システムに格納されたデータ・チャンクを重複排除することであって、前記重複排除情報を使用して、重複排除のために前記クライアント・データのチャンクが前記クライアント・データから選択される、前記重複排除することを含む、コンピュータ実装方法。

【請求項2】

前記重複排除情報が、共有鍵を使用して暗号化され、前記ストレージ・システムが、前記共有鍵を使用して暗号化されたデータ・チャンクを含み、前記共有鍵が、前記ストレージ・システムによって使用可能であり、前記重複排除することは、第1のクライアントの秘密鍵および共有鍵から成る群から選択された鍵を使用して暗号化されたデータ・チャンクに対して実行される、請求項1に記載のコンピュータ実装方法。

【請求項 3】

重複排除されていないクライアント・データのチャンクが、前記第 1 のクライアントの秘密鍵を使用して暗号化されて前記ストレージ・システムに格納される、請求項 2 に記載のコンピュータ実装方法。

【請求項 4】

前記ストレージ・システムが前記データ・チャンクのメタデータを格納し、前記メタデータが、前記データ・チャンクを暗号化するために使用された前記第 1 のクライアントの秘密鍵を一意に識別する鍵識別子を含む、請求項 1 ないし 3 のいずれかに記載のコンピュータの実装方法。

【請求項 5】

第 2 のクライアントの秘密鍵を使用して暗号化されたデータを受信することと、
データ・チャンクを重複排除することとを含み、1 つまたは複数のデータ・チャンクが、前記第 1 のクライアントの秘密鍵を使用して暗号化され、前記 1 つまたは複数のデータ・チャンクと異なる別の 1 つまたは複数のデータ・チャンクが、前記第 2 のクライアントの秘密鍵を使用して暗号化される、請求項 1 ないし 4 のいずれかに記載のコンピュータ実装方法。

【請求項 6】

コンピュータによって実行されるコンピュータ実装方法であって、前記コンピュータが、
重複排除情報を生成することであって、前記重複排除情報が、クライアント・データの一部に関連付けられたシグネチャを含む、前記生成することと、

共有鍵を使用して前記重複排除情報を暗号化することと、
クライアントの秘密鍵を使用して前記クライアント・データを暗号化することと、
前記重複排除情報および前記クライアント・データを重複排除のためにストレージ・システムに送信することと、

前記ストレージ・システムに対して前記クライアント・データを要求することと、
前記クライアント・データをデータ・チャンクとして受信することであって、前記データ・チャンクのうちの 1 つまたは複数の前記クライアントの秘密鍵を使用して暗号化され、前記データ・チャンクのうちの 1 つまたは複数の前記共有鍵を使用して暗号化される、前記受信することとを含む、コンピュータ実装方法。

【請求項 7】

前記ストレージ・システムに対して前記クライアント・データを要求することが、署名されたアクセス制御を使用して前記要求を検証することを含む、請求項 6 に記載のコンピュータ実装方法。

【請求項 8】

前記ストレージ・システムに対して前記共有鍵を要求することを含む、請求項 6 または 7 のいずれかに記載のコンピュータ実装方法。

【請求項 9】

デチャンカが前記データをデータ・チャンクとして受信し、前記デチャンカが、どの鍵を使用して各データ・チャンクが暗号化されたかを決定する、請求項 6 ないし 8 のいずれかに記載のコンピュータ実装方法。

【請求項 10】

前記クライアント・データを圧縮することと、
前記クライアントの秘密鍵を使用して前記クライアント・データを暗号化することとを含む、請求項 6 ないし 9 のいずれかに記載のコンピュータ実装方法。

【請求項 11】

コンピュータ可読媒体に格納され、デジタル・コンピュータの内部メモリに読み込み可能なコンピュータ・プログラムであって、前記プログラムがコンピュータ上で実行された場合に請求項 1 ないし 10 のいずれかに記載の前記方法を実行するためのソフトウェア・コード部分を含む、コンピュータ・プログラム。

【請求項 12】

プロセッサと、

前記プロセッサと統合されたか、前記プロセッサによって実行可能であるか、または前記プロセッサと統合され、前記プロセッサによって実行可能である論理とを備えているシステムであって、前記論理が、

ストレージ・システムで重複排除情報を受信することであって、前記重複排除情報が、動作を実行するために前記ストレージ・システムによってアクセス可能であり、前記重複排除情報が、クライアント・データの一部に関連付けられたシグネチャを含む、前記受信することと、

クライアントの秘密鍵を使用して暗号化された前記クライアント・データを受信することであって、前記クライアントの秘密鍵が前記ストレージ・システムには使用することができない、前記受信することと、

10

前記クライアント・データのチャンクに対して、前記ストレージ・システムに格納されたデータ・チャンクを重複排除することであって、前記重複排除情報を使用して、重複排除のために前記クライアント・データのチャンクが前記クライアント・データから選択される、前記重複排除することとを実行するように構成される、システム。

【請求項 13】

プロセッサと、

前記プロセッサと統合されたか、前記プロセッサによって実行可能であるか、または前記プロセッサと統合され、前記プロセッサによって実行可能である論理とを備えているシステムであって、前記論理が、

20

重複排除情報を生成することであって、前記重複排除情報が、クライアント・データの一部に関連付けられたシグネチャを含む、前記生成することと、

共有鍵を使用して前記重複排除情報を暗号化することと、

クライアントの秘密鍵を使用して前記クライアント・データを暗号化することと、

前記重複排除情報および前記クライアント・データを重複排除のためにストレージ・システムに送信することと、

前記ストレージ・システムに対して前記クライアント・データを要求することと、

前記クライアント・データをデータ・チャンクとして受信することであって、前記データ・チャンクのうちの1つまたは複数が前記クライアントの秘密鍵を使用して暗号化され、前記データ・チャンクのうちの1つまたは複数が前記共有鍵を使用して暗号化される、前記受信することとを実行するように構成される、システム。

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明はデータ重複排除に関連しており、より詳細には、本発明はクラウド・ストレージ・システムおよびネットワークにおけるデータ重複排除のための不透明な暗号化に関連している。

【背景技術】

【0002】

重複排除または圧縮あるいはその両方などの従来のデータ削減技術は、暗号化データに適用された場合、意味のある削減を提供しない。さまざまな暗号化アルゴリズムが、従来の重複排除プロセスが重複するデータ・チャンクを識別するのを妨げる場合、一意の暗号鍵を使用してそれぞれ暗号化されたデータの複数のセットの重複排除は、失敗に終わる。従来のデータ削減技術は、クライアントとストレージ・システムの間の十分なデータの機密性も提供しない。

40

【0003】

例えば、1つの既知の独自の鍵使用（BYOK：bring your own key）暗号化技術は、マルチパーティ信用システム（multi-party trust system）を含む。すべてのデータ削減機能は、すべてのデータにアクセスできるストレージ・システムによって提供されることがあるが、ストレージ・システムがクライアントの鍵にアクセスすることができるた

50

め、従来のB Y O Kシステムは、ストレージ・システムとクライアントの間のデータの機密性を提供しない。サード・パーティの鍵サービスも、クライアント・データの暗号化に使用される共有暗号鍵にアクセスすることができる。この形態のB Y O K暗号化の場合、データの機密性はユーザ間のみに存在する。

【0004】

従来の保管時の暗号化は、ストレージ・システムに知られている鍵を使用して、暗号化されていない入力データを暗号化する。ストレージ・システムは、すべてのデータを復号し、システム内のすべてのデータに対して重複排除を実行することができる。しかし、保管時の暗号化はデータの機密性を提供しない。

【0005】

従来の完全にクライアント側の暗号化は、ストレージ・システムに知られていない鍵を使用してデータを暗号化する。ストレージ・システムは、共通鍵を使用して暗号化されたデータのみを重複排除する。完全にクライアント側の重複排除は、相対的に高いデータの機密性を提供するが、重複排除の効率を妨げる。

【0006】

したがって、当技術分野において、前述の問題に対処する必要がある。

【発明の概要】

【0007】

第1の態様から見ると、本発明は、ストレージ・システムで重複排除情報を受信することであって、重複排除情報が、動作を実行するためにストレージ・システムによってアクセス可能であり、重複排除情報が、クライアント・データの一部に関連付けられたシグネチャを含む、受信することと、クライアントの秘密鍵を使用して暗号化されたクライアント・データを受信することであって、クライアントの秘密鍵がストレージ・システムには使用することができない、受信することと、クライアント・データのチャンクに対して、ストレージ・システムに格納されたデータ・チャンクを重複排除することであって、重複排除情報を使用して、重複排除のためにクライアント・データのチャンクがクライアント・データから選択される、重複排除することとを含んでいる、コンピュータ実装方法を提供する。

【0008】

さらなる態様から見ると、本発明は、不透明な暗号化のためのコンピュータ・プログラム製品を提供し、コンピュータ・プログラム製品は、プログラム命令が具現化されているコンピュータ可読ストレージ媒体を備え、プログラム命令は、コンピュータによって実行可能であり、コンピュータに、コンピュータによってストレージ・システムで重複排除情報を受信することであって、重複排除情報が、動作を実行するためにストレージ・システムによってアクセス可能であり、重複排除情報が、クライアント・データの一部に関連付けられたシグネチャを含む、受信することと、コンピュータによって、クライアントの秘密鍵を使用して暗号化されたクライアント・データを受信することであって、クライアントの秘密鍵がストレージ・システムには使用することができない、受信することと、コンピュータによって、クライアント・データのチャンクに対して、ストレージ・システムに格納されたデータ・チャンクを重複排除することであって、重複排除情報を使用して、重複排除のためにクライアント・データのチャンクがクライアント・データから選択される、重複排除することとを実行させる。

【0009】

さらなる態様から見ると、本発明は、プロセッサ、およびプロセッサと統合された論理、プロセッサによって実行可能な論理、またはプロセッサと統合され、プロセッサによって実行可能な論理を備えるシステムを提供し、この論理は、ストレージ・システムで重複排除情報を受信することであって、重複排除情報が、動作を実行するためにストレージ・システムによってアクセス可能であり、重複排除情報が、クライアント・データの一部に関連付けられたシグネチャを含む、受信することと、クライアントの秘密鍵を使用して暗号化されたクライアント・データを受信することであって、クライアントの秘密鍵がスト

10

20

30

40

50

レージ・システムには使用することができない、受信することと、クライアント・データのチャンクに対して、ストレージ・システムに格納されたデータ・チャンクを重複排除することとあって、重複排除情報を使用して、重複排除のためにクライアント・データのチャンクがクライアント・データから選択される、重複排除することとを実行するように構成される。

【0010】

さらなる態様から見ると、本発明は、重複排除情報を生成することとあって、重複排除情報が、クライアント・データの一部に関連付けられたシグネチャを含む、生成することと、共有鍵を使用して重複排除情報を暗号化することと、クライアントの秘密鍵を使用してクライアント・データを暗号化することと、重複排除情報およびクライアント・データを重複排除のためにストレージ・システムに送信することと、ストレージ・システムに対してクライアント・データを要求することと、クライアント・データをデータ・チャンクとして受信することとあって、データ・チャンクのうちの1つまたは複数がクライアントの秘密鍵を使用して暗号化され、データ・チャンクのうちの1つまたは複数が共有鍵を使用して暗号化される、受信することとを含む、コンピュータ実装方法を提供する。

10

【0011】

さらなる態様から見ると、本発明は、プロセッサ、およびプロセッサと統合された論理、プロセッサによって実行可能な論理、またはプロセッサと統合され、プロセッサによって実行可能な論理を備えるシステムを提供し、この論理は、重複排除情報を生成することとあって、重複排除情報が、クライアント・データの一部に関連付けられたシグネチャを含む、生成することと、共有鍵を使用して重複排除情報を暗号化することと、クライアントの秘密鍵を使用してクライアント・データを暗号化することと、重複排除情報およびクライアント・データを重複排除のためにストレージ・システムに送信することと、ストレージ・システムに対してクライアント・データを要求することと、クライアント・データをデータ・チャンクとして受信することとあって、データ・チャンクのうちの1つまたは複数がクライアントの秘密鍵を使用して暗号化され、データ・チャンクのうちの1つまたは複数が共有鍵を使用して暗号化される、受信することとを実行するように構成される。

20

【0012】

さらなる態様から見ると、本発明は、クラウド・ストレージ・システムおよびネットワークにおけるデータ重複排除のためのコンピュータ・プログラム製品を提供し、コンピュータ・プログラム製品は、処理回路によって読み取り可能な、本発明のステップを実行するための方法を実行するために処理回路によって実行される命令を格納している、コンピュータ可読ストレージ媒体を備える。

30

【0013】

さらなる態様から見ると、本発明は、コンピュータ可読媒体に格納された、デジタル・コンピュータの内部メモリに読み込み可能なコンピュータ・プログラムを提供し、このコンピュータ・プログラムは、コンピュータ上で実行された場合に本発明のステップを実行するためのソフトウェア・コード部分を含む。

【0014】

コンピュータ実装方法は、1つの方法によれば、重複排除情報をストレージ・システムで受信することを含む。重複排除情報は、動作を実行するためにストレージ・システムによってアクセス可能である。重複排除情報は、クライアント・データの一部に関連付けられたシグネチャを含む。この方法は、クライアントの秘密鍵を使用して暗号化されたクライアント・データを受信することも含む。クライアントの秘密鍵は、ストレージ・システムには使用することができない。この方法は、クライアント・データのチャンクに対して、ストレージ・システムに格納されたデータ・チャンクを重複排除することを含み、重複排除情報を使用して、重複排除のためにクライアント・データのチャンクがクライアント・データから選択される。この方法は、従来技術と比較して、データの機密性の改善、データの削減の改良、およびクライアントのオーバーヘッドの削減を実現する。

40

【0015】

50

この方法は、第2のクライアントの鍵を使用して暗号化されたデータを受信することと、データ・チャンクを重複排除することとを任意選択的に含んでよく、1つまたは複数のデータ・チャンクが、第1のクライアントの秘密鍵を使用して暗号化され、1つまたは複数のデータ・チャンクが、第2のクライアントの秘密鍵を使用して暗号化される。前述の方法は、さまざまな鍵を使用して暗号化されたデータ・チャンクの重複排除を可能にし、ユーザ間およびユーザとストレージ・システムの間のデータの機密性を提供する。

【0016】

不透明な暗号化のためのコンピュータ・プログラム製品は、1つの方法によれば、プログラム命令が具現化されているコンピュータ可読ストレージ媒体を含む。プログラム命令は、コンピュータによって実行可能であり、コンピュータに前述の方法を実行させる。

【0017】

システムは、1つの方法によれば、プロセッサ、およびプロセッサと統合された論理、プロセッサによって実行可能な論理、またはプロセッサと統合され、プロセッサによって実行可能な論理を含む。この論理は、前述の方法を実行するように構成される。

【0018】

コンピュータ実装方法は、1つの方法によれば、重複排除情報を生成することを含む。重複排除情報は、クライアント・データの一部に関連付けられたシグネチャを含む。この方法は、共有鍵を使用して重複排除情報を暗号化すること、およびクライアントの秘密鍵を使用してクライアント・データを暗号化することを含む。この方法は、重複排除情報およびクライアント・データを重複排除のためにストレージ・システムに送信することを含む。この方法は、ストレージ・システムに対してクライアント・データを要求することと、クライアント・データをデータ・チャンクとして受信することとを含み、データ・チャンクのうちの1つまたは複数がクライアントの秘密鍵を使用して暗号化され、データ・チャンクのうちの1つまたは複数が共有鍵を使用して暗号化される。この方法は、従来技術と比較して、データの機密性の改善、データの削減の改良、およびクライアントのオーバーヘッドの削減を実現する。

【0019】

この方法は、ストレージ・システムに対して共有鍵を要求することを任意選択的に含む。この方法は、ストレージ・システムとクライアントの間のデータの機密性を維持しながら、ストレージ・システムによってアクセスできる暗号鍵を使用して重複排除情報を暗号化することを可能にする。

【0020】

システムは、1つの方法によれば、プロセッサ、およびプロセッサと統合された論理、プロセッサによって実行可能な論理、またはプロセッサと統合され、プロセッサによって実行可能な論理を含む。この論理は、前述の方法を実行するように構成される。

【0021】

本発明のその他の態様および構成は、本発明の原理を図面と併せて例として説明する、以下の詳細な説明から明らかになるであろう。

【0022】

以下では、次の図に示された好ましい実施形態を単に例として参照し、本発明が説明される。

【図面の簡単な説明】

【0023】

【図1】本発明の実施形態に従って、クラウド・コンピューティング・ノードを示す図である。

【図2】本発明の1つの構成に従って、クラウド・コンピューティング環境を示す図である。

【図3】本発明の1つの構成に従って、抽象モデル・レイヤを示す図である。

【図4】本発明の1つの構成に従う、高レベルのアーキテクチャの図である。

【図5】本発明の1つの構成に従う、高レベルのアーキテクチャの図である。

10

20

30

40

50

【図6】本発明の1つの構成に従う、高レベルのアーキテクチャの図である。

【図7】本発明の1つの構成に従う、高レベルのアーキテクチャの図である。

【図8】本発明の1つの構成に従う、方法のフローチャートである。

【図9】本発明の1つの構成に従う、方法のフローチャートである。

【発明を実施するための形態】

【0024】

以下の説明は、本発明の一般的原理を説明する目的で行われるのであり、本明細書において請求される本発明の概念を制限するように意図されていない。さらに、本明細書に記載された特定の特徴を、さまざまな可能な組合せおよび並べ替えの各々において、その他の説明された特徴と組み合わせて使用することができる。

【0025】

本明細書では、特に具体的に定義されない限り、すべての用語には、本明細書から示される意味、および当業者によって理解される意味、または辞書、論文などにおいて定義された意味、あるいはその両方を含む、最も広い可能な解釈が与えられる。

【0026】

本明細書および添付の特許請求の範囲において使用されるとき、単数形「a」、「an」および「the」が、特に規定されない限り、複数の指示対象を含むということにも注意しなければならない。「備える」または「備えている」あるいはその両方の用語は、本明細書で使用される場合、記載された機能、整数、ステップ、動作、要素、またはコンポーネント、あるいはその組合せの存在を示すが、1つまたは複数のその他の機能、整数、ステップ、動作、要素、コンポーネント、またはこれらのグループ、あるいはその組合せの存在または追加を除外していないということが、さらに理解されるであろう。

【0027】

以下の説明は、不透明な暗号化を使用するデータ重複排除の複数の構成を開示する。

【0028】

1つの一般的な構成では、コンピュータ実装方法は、ストレージ・システムで重複排除情報を受信することを含む。重複排除情報は、動作を実行するためにストレージ・システムによってアクセス可能である。重複排除情報は、クライアント・データの一部に関連付けられたシグネチャを含む。この方法は、クライアントの秘密鍵を使用して暗号化されたクライアント・データを受信することを含む。クライアントの秘密鍵は、ストレージ・システムには使用することができない。この方法は、クライアント・データのチャンクに対して、ストレージ・システムに格納されたデータ・チャンクを重複排除することを含み、重複排除情報を使用して、重複排除のためにクライアント・データのチャンクがクライアント・データから選択される。

【0029】

別の一般的な構成では、不透明な暗号化のためのコンピュータ・プログラム製品は、プログラム命令が具現化されているコンピュータ可読ストレージ媒体を含む。プログラム命令は、コンピュータによって実行可能であり、コンピュータに前述の方法を実行させる。

【0030】

さらに別の一般的な構成では、システムは、プロセッサ、およびプロセッサと統合された論理、プロセッサによって実行可能な論理、またはプロセッサと統合され、プロセッサによって実行可能な論理を含む。この論理は、前述の方法を実行するように構成される。

【0031】

1つの一般的な構成では、コンピュータ実装方法は、重複排除情報を生成することを含む。重複排除情報は、クライアント・データの一部に関連付けられたシグネチャを含む。この方法は、共有鍵を使用して重複排除情報を暗号化すること、およびクライアントの秘密鍵を使用してクライアント・データを暗号化することを含む。この方法は、重複排除情報およびクライアント・データを重複排除のためにストレージ・システムに送信することを含む。この方法は、ストレージ・システムに対してクライアント・データを要求することと、クライアント・データをデータ・チャンクとして受信することとを含み、データ・

10

20

30

40

50

チャンクのうちの1つまたは複数がクライアントの秘密鍵を使用して暗号化され、データ・チャンクのうちの1つまたは複数が共有鍵を使用して暗号化される。

【0032】

別の一般的な構成では、システムは、プロセッサ、およびプロセッサと統合された論理、プロセッサによって実行可能な論理、またはプロセッサと統合され、プロセッサによって実行可能な論理を含む。この論理は、前述の方法を実行するように構成される。

【0033】

本開示にはクラウド・コンピューティングに関する詳細な説明が含まれているが、本明細書において示された内容の実装は、クラウド・コンピューティング環境に限定されないと理解されるべきである。本発明の構成は、現在既知であるか、または今後開発される任意のその他の種類のコンピューティング環境と組み合わせて実装できる。

【0034】

クラウド・コンピューティングは、構成可能な計算リソース（例えば、ネットワーク、ネットワーク帯域幅、サーバ、処理、メモリ、ストレージ、アプリケーション、仮想マシン、およびサービス）の共有プールへの便利なオンデマンドのネットワーク・アクセスを可能にするためのサービス提供モデルであり、管理上の手間またはサービス・プロバイダとのやりとりを最小限に抑えて、これらのリソースを迅速にプロビジョニングおよび解放することができる。このクラウド・モデルは、少なくとも5つの特徴、少なくとも3つのサービス・モデル、および少なくとも4つのデプロイメント・モデルを含むことができる。

【0035】

特徴は、次のとおりである。

オンデマンドのセルフ・サービス：クラウドの利用者は、サーバの時間、ネットワーク・ストレージなどの計算能力を一方的に、サービス・プロバイダとの人間的なやりとりを必要とせず、必要に応じて自動的にプロビジョニングすることができる。

幅広いネットワーク・アクセス：クラウドの能力は、ネットワークを経由して利用可能であり、標準的なメカニズムを使用してアクセスできるため、異種のシン・クライアントまたはシック・クライアント・プラットフォーム（例えば、携帯電話、ラップトップ、およびPDA）による利用を促進する。

リソース・プール：プロバイダの計算リソースは、プールされ、マルチテナント・モデルを使用して複数の利用者に提供される。さまざまな物理的および仮想的リソースが、要求に従って動的に割り当ておよび再割り当てされる。場所に依存しないという感覚があり、利用者は通常、提供されるリソースの正確な場所に関してどうすることもできないし、知っているわけでもないが、さらに高い抽象レベルでは、場所（例えば、国、州、またはデータセンター）を指定できる場合がある。

迅速な順応性：能力は、迅速かつ柔軟に、場合によっては自動的にプロビジョニングされ、素早くスケールアウトし、迅速に解放されて素早くスケールインすることができる。プロビジョニングに使用できる能力は、利用者には、多くの場合、任意の量をいつでも無制限に購入できるように見える。

測定されるサービス：クラウド・システムは、計測機能を活用することによって、サービスの種類（例えば、ストレージ、処理、帯域幅、およびアクティブなユーザのアカウント）に適した抽象レベルで、リソースの使用を自動的に制御および最適化する。リソースの使用量は監視、制御、および報告することができ、利用されるサービスのプロバイダと利用者の両方に透明性が提供される。

【0036】

サービス・モデルは、次のとおりである。

SaaS（Software as a Service）：利用者に提供される能力は、クラウド・インフラストラクチャ上で稼働しているプロバイダのアプリケーションの利用である。それらのアプリケーションは、Webブラウザ（例えば、Webベースの電子メール）などのシン・クライアント・インターフェイスを介して、さまざまなクライアント・デバイスからアクセスできる。利用者は、ネットワーク、サーバ、オペレーティング・システム、ストレ

10

20

30

40

50

ージ、またはさらには個々のアプリケーション機能を含む基盤になるクラウド・インフラストラクチャを、限定的なユーザ固有のアプリケーション構成設定を行う可能性を除き、管理することも制御することもない。

P a a S (Platform as a Service)：利用者に提供される能力は、プロバイダによってサポートされるプログラミング言語およびツールを使用して作成された、利用者が作成または取得したアプリケーションをクラウド・インフラストラクチャにデプロイすることである。利用者は、ネットワーク、サーバ、オペレーティング・システム、またはストレージを含む基盤になるクラウド・インフラストラクチャを管理することも制御することもないが、デプロイされたアプリケーション、および場合によってはアプリケーション・ホスティング環境の構成を制御することができる。

10

I a a S (Infrastructure as a Service)：利用者に提供される能力は、処理、ストレージ、ネットワーク、およびその他の基本的な計算リソースのプロビジョニングであり、利用者は、オペレーティング・システムおよびアプリケーションを含むことができる任意のソフトウェアをデプロイして実行できる。利用者は、基盤になるクラウド・インフラストラクチャを管理することも制御することもないが、オペレーティング・システム、ストレージ、およびデプロイされたアプリケーションを制御することができ、場合によっては、選択されたネットワーク・コンポーネント（例えば、ホスト・ファイアウォール）を限定的に制御できる。

【0037】

デプロイメント・モデルは、次のとおりである。

20

プライベート・クラウド：このクラウド・インフラストラクチャは、ある組織のためにのみ運用される。この組織またはサード・パーティによって管理することができ、オンプレミスまたはオフプレミスに存在することができる。

コミュニティ・クラウド：このクラウド・インフラストラクチャは、複数の組織によって共有され、関心事（例えば、任務、セキュリティ要件、ポリシー、およびコンプライアンスに関する考慮事項）を共有している特定のコミュニティをサポートする。これらの組織またはサード・パーティによって管理することができ、オンプレミスまたはオフプレミスに存在することができる。

パブリック・クラウド：このクラウド・インフラストラクチャは、一般ユーザまたは大規模な業界団体が使用できるようになっており、クラウド・サービスを販売する組織によって所有される。

30

ハイブリッド・クラウド：このクラウド・インフラストラクチャは、データとアプリケーションの移植を可能にする標準化された技術または独自の技術（例えば、クラウド間の負荷バランスを調整するためのクラウド・バースト）によって固有の実体を残したまま互いに結合された2つ以上のクラウド（プライベート、コミュニティ、またはパブリック）の複合である。

【0038】

クラウド・コンピューティング環境は、ステートレス、疎結合、モジュール性、および意味的相互運用性に重点を置いたサービス指向の環境である。クラウド・コンピューティングの中心になるのは、相互接続されたノードのネットワークを含んでいるインフラストラクチャである。

40

【0039】

ここで図1を参照すると、クラウド・コンピューティング・ノードの例の概略図が示されている。クラウド・コンピューティング・ノード10は、適切なクラウド・コンピューティング・ノードの一例に過ぎず、本明細書に記載された本発明の実施形態の使用または機能の範囲に関して、いかなる制限を示唆することも意図されていない。いずれにせよ、クラウド・コンピューティング・ノード10は、前述した機能のいずれかを実装すること、または実行すること、あるいはその両方を行うことができる。

【0040】

クラウド・コンピューティング・ノード10内には、他の多数の汎用または専用のコン

50

ピューティング・システム環境または構成で運用できるコンピュータ・システム／サーバ 12 が存在する。コンピュータ・システム／サーバ 12 と共に使用するのに適した周知のコンピューティング・システム、環境、または構成、あるいはその組合せの例は、パーソナル・コンピュータ・システム、サーバ・コンピュータ・システム、シン・クライアント、シック・クライアント、ハンドヘルドまたはラップトップ・デバイス、マイクロプロセッサ・システム、マイクロプロセッサベース・システム、セット・トップ・ボックス、プログラマブル・コンシューマ・エレクトロニクス、ネットワーク PC、マイクロコンピュータ・システム、メインフレーム・コンピュータ・システム、およびこれらのシステムまたはデバイスのいずれかを含む分散クラウド・コンピューティング環境などを含むが、これらに限定されない。

10

【0041】

コンピュータ・システム／サーバ 12 は、コンピュータ・システムによって実行されているプログラム・モジュールなどの、コンピュータ・システムによって実行可能な命令との一般的な関連において説明されてよい。通常、プログラム・モジュールは、特定のタスクを実行するか、または特定の抽象データ型を実装するルーチン、プログラム、オブジェクト、コンポーネント、論理、データ構造などを含んでよい。コンピュータ・システム／サーバ 12 は、通信ネットワークを介してリンクされたりリモート処理デバイスによってタスクが実行される、分散クラウド・コンピューティング環境で実行されてよい。分散クラウド・コンピューティング環境において、プログラム・モジュールは、メモリ・ストレージ・デバイスを含む、ローカルおよびリモートの両方のコンピュータ・システム・ストレージ媒体に配置されてよい。

20

【0042】

図 1 に示すように、クラウド・コンピューティング・ノード 10 内のコンピュータ・システム／サーバ 12 は、汎用コンピューティング・デバイスの形態で示されている。コンピュータ・システム／サーバ 12 のコンポーネントは、1 つまたは複数のプロセッサまたはプロセッシング・ユニット 16、システム・メモリ 28、およびシステム・メモリ 28 を含むさまざまなシステム・コンポーネントをプロセッサ 16 に接続するバス 18 を含んでよいが、これらに限定されない。

【0043】

バス 18 は、メモリ・バスまたはメモリ・コントローラ、ペリフェラル・バス、アクセラレーテッド・グラフィックス・ポート、およびさまざまなバス・アーキテクチャのいずれかを使用するプロセッサまたはローカル・バスを含む、複数の種類のバス構造のいずれかのうちの 1 つまたは複数を表す。例として、そのようなアーキテクチャは、ISA (Industry Standard Architecture) バス、MCA (Micro Channel Architecture) バス、EISA (Enhanced ISA) バス、VESA (Video Electronics Standards Association) ローカル・バス、および PCI (Peripheral Component Interconnects) バスを含むが、これらに限定されない。

30

【0044】

コンピュータ・システム／サーバ 12 は、通常、さまざまなコンピュータ・システム可読媒体を含む。そのような媒体は、コンピュータ・システム／サーバ 12 によってアクセスできる任意の使用可能な媒体であってよく、揮発性および不揮発性媒体、取り外し可能および取り外し不可の媒体を含む。

40

【0045】

システム・メモリ 28 は、ランダム・アクセス・メモリ (RAM: random access memory) 30 またはキャッシュ・メモリ 32 あるいはその両方などの、揮発性メモリの形態でのコンピュータ・システム可読媒体を含むことができる。コンピュータ・システム／サーバ 12 は、その他の取り外し可能／取り外し不可、揮発性／不揮発性のコンピュータ・システム・ストレージ媒体をさらに含んでよい。単に例として、取り外し不可、不揮発性の磁気媒体 (図示されておらず、通常は「ハード・ドライブ」と呼ばれる) に対する読み取りと書き込みを行うために、ストレージ・システム 34 を提供することができる。図示

50

されていないが、取り外し可能、不揮発性の磁気ディスク（例えば、「フロッピー・ディスク」）に対する読み取りと書き込みを行うための磁気ディスク・ドライブ、およびCD-ROM、DVD-ROM、またはその他の光媒体などの取り外し可能、不揮発性の光ディスクに対する読み取りと書き込みを行うための光ディスク・ドライブを提供することができる。そのような例では、それぞれを、1つまたは複数のデータ媒体インターフェイスによってバス18に接続することができる。下で詳細に示され、説明されているように、メモリ28は、本発明の実施形態の機能を実行するように構成された一連の（例えば、少なくとも1つの）プログラム・モジュールを備える少なくとも1つのプログラム製品を含んでよい。

【0046】

例えば、一連の（少なくとも1つの）プログラム・モジュール42を含んでいるプログラム／ユーティリティ40がメモリ28に格納されてよいが、これに限定されず、オペレーティング・システム、1つまたは複数のアプリケーション・プログラム、その他のプログラム・モジュール、およびプログラム・データも格納されてよい。オペレーティング・システム、1つまたは複数のアプリケーション・プログラム、その他のプログラム・モジュール、およびプログラム・データまたはこれらの組合せの各々は、ネットワーク環境の実装を含んでよい。プログラム・モジュール42は、通常、本明細書に記載された本発明の実施形態の機能または方法あるいはその両方を実行する。

【0047】

コンピュータ・システム／サーバ12は、キーボード、ポインティング・デバイス、ディスプレイ24などの1つまたは複数の外部デバイス14、ユーザがコンピュータ・システム／サーバ12と情報をやりとりできるようにする1つまたは複数のデバイス、またはコンピュータ・システム／サーバ12が1つまたは複数の他のコンピューティング・デバイスと通信できるようにする任意のデバイス（例えば、ネットワーク・カード、モデムなど）、あるいはその組合せと通信してもよい。そのような通信は、入出力（I/O：Input/Output）インターフェイス22を介して行うことができる。さらに、コンピュータ・システム／サーバ12は、ローカル・エリア・ネットワーク（LAN：local area network）、一般的な広域ネットワーク（WAN：wide area network）、またはパブリック・ネットワーク（例えば、インターネット）、あるいはその組合せなどの1つまたは複数のネットワークと、ネットワーク・アダプタ20を介して通信することができる。図示されているように、ネットワーク・アダプタ20は、バス18を介してコンピュータ・システム／サーバ12の他のコンポーネントと通信する。図示されていないが、その他のハードウェア・コンポーネントまたはソフトウェア・コンポーネントあるいはその両方を、コンピュータ・システム／サーバ12と併用できるということが理解されるべきである。その例として、マイクロコード、デバイス・ドライバ、冗長プロセッシング・ユニット、外部ディスク・ドライブ・アレイ、RAIDシステム、テープ・ドライブ、およびデータ・アーカイブ・ストレージ・システムなどが挙げられるが、これらに限定されない。

【0048】

ここで図2を参照すると、例示的なクラウド・コンピューティング環境50が示されている。図示されているように、クラウド・コンピューティング環境50は、クラウドの利用者によって使用されるローカル・コンピューティング・デバイス（例えば、パーソナル・デジタル・アシスタント（PDA：personal digital assistant）または携帯電話54A、デスクトップ・コンピュータ54B、ラップトップ・コンピュータ54C、または自動車コンピュータ・システム54N、あるいはその組合せなど）が通信できる1つまたは複数のクラウド・コンピューティング・ノード10を含んでいる。ノード10は、互いに通信してよい。ノード10は、本明細書において前述されたプライベート・クラウド、コミュニティ・クラウド、パブリック・クラウド、またはハイブリッド・クラウド、あるいはこれらの組合せなどの1つまたは複数のネットワーク内で、物理的または仮想的にグループ化されてよい（図示されていない）。これによって、クラウド・コンピューティング環境50は、クラウドの利用者がローカル・コンピューティング・デバイス上でリソース

10

20

30

40

50

を維持する必要のないインフラストラクチャ、プラットフォーム、またはSaaS、あるいはその組合せを提供できる。図1に示されたコンピューティング・デバイス54A～Nの種類は、例示のみが意図されており、コンピューティング・ノード10およびクラウド・コンピューティング環境50は、任意の種類のネットワークまたはネットワーク・アドレス可能な接続（例えば、Webブラウザを使用した接続）あるいはその両方を経由して任意の種類のコンピュータ制御デバイスと通信することができると理解される。

【0049】

ここで図3を参照すると、クラウド・コンピューティング環境50（図2）によって提供される機能的抽象レイヤのセットが示されている。図3に示されたコンポーネント、レイヤ、および機能は、例示のみが意図されており、本発明の構成がこれらに限定されないということが、あらかじめ理解されるべきである。図示されているように、次のレイヤおよび対応する機能が提供される。

【0050】

ハードウェアおよびソフトウェア・レイヤ60は、ハードウェア・コンポーネントおよびソフトウェア・コンポーネントを含む。ハードウェア・コンポーネントの例としては、メインフレーム61、RISC（Reduced Instruction Set Computer）アーキテクチャベースのサーバ62、サーバ63、ブレード・サーバ64、ストレージ・デバイス65、ならびにネットワークおよびネットワーク・コンポーネント66が挙げられる。一部の構成では、ソフトウェア・コンポーネントは、ネットワーク・アプリケーション・サーバ・ソフトウェア67およびデータベース・ソフトウェア68を含む。

【0051】

仮想化レイヤ70は、仮想サーバ71、仮想ストレージ72、仮想プライベート・ネットワークを含む仮想ネットワーク73、仮想アプリケーションおよびオペレーティング・システム74、ならびに仮想クライアント75などの仮想的実体を提供できる抽象レイヤを備える。

【0052】

一例を挙げると、管理レイヤ80は、以下で説明される機能を提供することができる。リソース・プロビジョニング81は、クラウド・コンピューティング環境内でタスクを実行するために利用される計算リソースおよびその他のリソースの動的調達を行う。計測および価格設定82は、クラウド・コンピューティング環境内でリソースが利用される際のコスト追跡、およびそれらのリソースの利用に対する請求書の作成と送付を行う。一例を挙げると、それらのリソースは、アプリケーション・ソフトウェア・ライセンスを含んでよい。セキュリティは、クラウドの利用者およびタスクのID検証を行うとともに、データおよびその他のリソースの保護を行う。ユーザ・ポータル83は、クラウド・コンピューティング環境へのアクセスを利用者およびシステム管理者に提供する。サービス・レベル管理84は、必要なサービス・レベルを満たすように、クラウドの計算リソースの割り当てと管理を行う。サービス水準合意（SLA：Service Level Agreement）計画および実行85は、今後の要求が予想されるクラウドの計算リソースの事前準備および調達を、SLAに従って行う。

【0053】

ワークロード・レイヤ90は、クラウド・コンピューティング環境で利用できる機能の例を示している。このレイヤから提供されてよいワークロードおよび機能の例としては、マッピングおよびナビゲーション91、ソフトウェア開発およびライフサイクル管理92、仮想クラスルーム教育の配信93、データ解析処理94、トランザクション処理95、および不透明な暗号化を使用するデータ重複排除96が挙げられる。

【0054】

重複排除または圧縮あるいはその両方などの従来のデータ削減技術は、暗号化データに適用された場合、意味のある削減を提供しない。さまざまな暗号化アルゴリズムが、従来の重複排除プロセスが重複するデータ・チャンクを識別するのを妨げる場合、一意の暗号鍵を使用してそれぞれ暗号化されたデータの複数のセットの重複排除は、失敗に終わる。

従来のデータ削減技術は、クライアントとストレージ・システムの間の十分なデータの機密性も提供しない。

【0055】

例えば、1つの既知の独自の鍵使用（BYOK）暗号化技術は、マルチパーティ信用システムを含む。クライアントは、サード・パーティの鍵サービスに対して暗号鍵を要求し、この鍵がクライアントに返される。ストレージ・システムは、同じ暗号鍵をチェックアウトしてよい。クライアントは、共有暗号鍵を使用してデータを暗号化し、そのデータをストレージ・システムに送信する。ストレージ・システムは、共有暗号鍵を使用してデータを復号することができ、データを格納するために、ストレージ・システムの鍵を使用してデータを再び暗号化することができる。クライアントの読み取り動作時に、ストレージ・システムは、ストレージ・システムの鍵を使用してデータを復号し、共有暗号鍵を使用してデータを再び暗号化し、そのデータをクライアントに返す。すべてのデータ削減機能は、すべてのデータ（例えば、暗号化されていないデータ、ストレージ・システムの鍵を使用して暗号化されたデータ、共有暗号鍵を使用して暗号化されたデータなど）にアクセスできるストレージ・システムによって提供されることがあるが、ストレージ・システムがクライアントの鍵（例えば、共有暗号鍵）にアクセスすることができるため、従来のBYOKシステムは、ストレージ・システムとクライアントの間のデータの機密性を提供しない。サード・パーティの鍵サービスも、クライアント・データの暗号化に使用される共有暗号鍵にアクセスすることができる。この例示的なBYOK暗号化技術の場合、データの機密性はユーザ間のみに存在する。

【0056】

別のBYOK技術では、サード・パーティのサービスがデータの交換に関与せず、暗号鍵がストレージ・システムと直接共有される。この技術は、ユーザとストレージ・システムの間のデータの機密性を提供しない。別の例示的な従来のBYOK技術は、ストレージ・システムで「信頼できるデクリプタ」機能を使用するが、データの機密性に関してまだ不足している。

【0057】

従来の保管時の暗号化は、ストレージ・システムに知られている鍵を使用して、暗号化されていない入力データを暗号化する。ストレージ・システムは、すべてのデータを復号し、システム内のすべてのデータに対して重複排除を実行することができる。保管時の暗号化は、データの機密性を提供しない。

【0058】

従来の完全にクライアント側の暗号化は、ストレージ・システムに知られていない鍵を使用してデータを暗号化する。ストレージ・システムは、共通鍵を使用して暗号化されたデータのみを重複排除する。完全にクライアント側の重複排除は、相対的に高いデータの機密性を提供するが、重複排除の効率を妨げる。

【0059】

全く対照的に、本明細書に記載されたさまざまな方法は、ストレージ・システムがクライアントの秘密暗号鍵または秘密復号鍵あるいはその両方にアクセスすることがないデータ圧縮または重複排除あるいはその両方のために、暗号化データをストレージ・システムに配信する能力をクライアントに提供する。本明細書で開示された方法の少なくとも一部は、ストレージ・システムにとって不透明な方法でデータを暗号化し、ストレージ・システム内の他のデータに対してデータを重複排除する。不透明な暗号化は、BYOK技術を含む従来の暗号化技術と比較して、データの機密性を向上させる。不透明な暗号化のさまざまな動作は、従来の完全にクライアント側の暗号化よりも相対的に良いデータ削減およびクライアント側の重複排除より少ないクライアントのオーバーヘッドを実現する。

【0060】

本明細書において提示される構成の少なくとも一部は、クライアント・システムで圧縮を実行して重複排除メタデータを生成し、ストレージ・システムで重複排除が実行される。クライアント・データは、1つの方法では、データがストレージ・システムに送信され

10

20

30

40

50

る前にクライアントの秘密暗号鍵を使用して暗号化される。既知のB Y O K技術とは対照的に、好ましい方法では、クライアント・データが、クライアントの秘密暗号鍵を使用して暗号化されてストレージ・システムに書き込まれ、ストレージ・システムは、クライアントの秘密復号鍵にアクセスすることができないため、クライアント・データを復号することができない。本明細書に記載された動作の少なくとも一部は、対称鍵暗号化または非対称暗号化（例えば、公開鍵基盤（P K I : public key infrastructure））あるいはその両方と共に使用されてよい。従来技術において知られた任意の構成に従ってP K I暗号化が実行されてよいということが、当業者によって理解されるべきである。例えば、P K Iにおける公開鍵は秘密鍵ではなく、公開鍵を使用してデータを暗号化するには、復号するための対応する秘密鍵（secret private key）が必要である。

10

【0061】

図4は、さまざまな構成に従う高レベルのアーキテクチャである。アーキテクチャ400は、さまざまな構成において、特に図1～図3および図5～図9に示された環境のいずれかにおいて、本発明に従って実装されてよい。当然、当業者が本説明を読んだときに理解するであろうように、図4において具体的に説明された要素よりも多いか、または少ない要素がアーキテクチャ400に含まれてよい。

【0062】

アーキテクチャ400は、対称鍵暗号化を使用する書き込み動作の例示的な実装を示している。アーキテクチャ400は、クライアント1側402を含んでいる。図に示されているように、クライアント1側402は、暗号化データをストレージ・システム420に送信する（下で詳細に説明される）。

20

【0063】

クライアント1側402は、クライアントの秘密暗号鍵404を含んでいる。好ましい方法では、クライアントの秘密暗号鍵404はクライアント1側402から離れない。この好ましい方法では、ストレージ・システム420は、データ転送のどの時点でも、暗号化データを暗号化されていない状態で見ることがない。好ましい方法では、クライアントの秘密暗号鍵404は、ストレージ・システム420、クライアント2側436（下で詳細に説明される）、任意のその他のクライアント、任意のその他のストレージ・システムなどと共有されない。クライアントの秘密暗号鍵404は、対称鍵、非対称鍵などであってよい。図に示されているように、クライアントの秘密暗号鍵404は対称鍵である。

30

【0064】

クライアント1側402は、メタ生成器（meta-generator）406を含んでいる。好ましい方法では、メタ生成器406は、データに関連付けられた重複排除メタデータを計算する。さまざまな構成では、メタ生成器406は、クライアント1側402またはストレージ・システム420あるいはその両方に配置されてよい。

【0065】

好ましい方法では、メタ生成器406は、クライアント1側402でデータに対してメタデータ動作を実行し、データ内のデータ重複排除の機会を識別する。メタ生成器406は、従来技術において知られた任意のデータ重複排除技術を使用してデータ重複排除の機会を識別する。1つの態様では、メタ生成器406は、スライド・ウィンドウ可変チャンク重複排除技術（sliding window variable chunk deduplication technique）を使用してデータ重複排除の機会を識別する。メタ生成器406は、重複排除シグネチャ（例えば、ローリング最小ハッシュ（rolling min hashes）、暗号ハッシュなど）を識別するか、または計算するか、あるいはその両方を実行する。メタ生成器406は、ポインタ、チャンク長、チャンク・シグネチャなどを含む、データ重複排除の機会に関連付けられたメタデータ（例えば、重複排除情報）を生成する。重複排除情報は、以下で詳細に説明されるように、ストレージ・システムによって提供された共有一時鍵（shared ephemeral key）を使用して暗号化されてよい。

40

【0066】

1つの方法では、メタ生成器406は、従来技術において知られた任意のデータ圧縮技

50

術を使用してデータを圧縮する。一部の方法では、チャンキングの前または後あるいはその両方で、さまざまな圧縮技術が適用されてよい。1つの構成では、チャンキング前の圧縮は、チャンキングの性能を改善する種類の圧縮であってよい。別の構成では、チャンキング後の圧縮は、結果として得られるチャンク・サイズを最小化することに向けて調整されてよい。

【0067】

シグネチャの計算は、圧縮の前または後あるいはその両方で、データ・チャンクに対して実行されてよい。圧縮前のシグネチャの計算は、読み取り時の復元後にシグネチャを検証できるようにする。圧縮後のシグネチャの計算は、チャンクが圧縮された状態にある間に、シグネチャをチェックできるようにする。

10

【0068】

クライアント1側402は、クライアントの秘密鍵エンクリプタ408を備えている。当業者が本開示を読むときに理解するであろうように、クライアントの秘密鍵エンクリプタ408は、クライアントの秘密暗号鍵404を使用してデータを暗号化してよい。

【0069】

クライアント1側402は、一時鍵エンクリプタ (ephemeral key encrypter) 410を備えている。当業者が本開示を読むときに理解するであろうように、一時鍵エンクリプタ410は、ストレージの一時鍵440を使用してデータを暗号化してよい (下で詳細に説明される)。

【0070】

20

クライアント1側402は、デチャンカ (dechunker) 412を備えている。デチャンカ412は、ストレージ・システムから受信されたデータ・チャンクをチャンク解除し、データ・チャンクは、複数の暗号鍵を使用して暗号化される。好ましい方法では、複数の暗号鍵を使用して暗号化されたデータ・チャンクとは、複数のチャンクの暗号鍵を使用して暗号化されたデータ・チャンクのことを指す。例えば、書き込みデータ444 (下で詳細に説明される) はデータ・チャンクのセットを含んでよく、データ・チャンクのサブセットまたは個別のデータ・チャンクあるいはその両方は、異なる (例えば、一意の) 鍵を使用して暗号化される。具体的には、1つの例示的な構成では、少なくとも一部のデータ・チャンクが第1の鍵を使用して暗号化され、少なくとも一部の他のデータ・チャンクが第2の鍵を使用して暗号化され、第1の鍵および第2の鍵は異なる鍵である。複数の暗号鍵を使用して暗号化されたデータ・チャンクは、本明細書で明示的に開示されない限り、複数の暗号化の対象になるデータ・チャンクとして解釈されるべきではない。例えば、複数の暗号化の対象になるデータ・チャンクとは、第1の鍵を使用して暗号化され、2回目に異なる鍵を使用して暗号化されたデータ・チャンクなどのことを指す。

30

【0071】

デチャンカ412は、どの暗号鍵を使用してデータ・チャンクが暗号化されているかを識別し、そのデータ・チャンクを対応するデクリプタ (例えば、クライアントの秘密鍵デクリプタ414、一時鍵デクリプタ416など) に転送してよい。当業者が本開示を読むときに理解するであろうように、各デクリプタは、対応する鍵を使用してデータを復号してよい。

40

【0072】

アーキテクチャ400は、ストレージ・システム420を含んでいる。ストレージ・システム420は、従来技術において知られた任意の種類のストレージ・システムであってよい。ストレージ・システム420が、本明細書において示されたコンポーネントよりも多いかまたは少ないコンポーネントを含んでよいということが、当業者によって理解されるべきである。ストレージ・システム420は、本明細書に記載されたさまざまな重複排除動作を実行するのが好ましい。

【0073】

ストレージ・システム420は、ストレージの秘密暗号鍵422を含んでいる。好ましい方法では、ストレージの秘密暗号鍵422はストレージ・システム420から離れない

50

。この好ましい方法では、ストレージの秘密暗号鍵 4 2 2 は、クライアント 1 側 4 0 2、クライアント 2 側 4 3 6、任意のその他のクライアント、任意のその他のストレージ・システムなどと共有されない。ストレージの秘密暗号鍵 4 2 2 は、対称鍵、非対称鍵などであってよい。図に示されているように、ストレージの秘密暗号鍵 4 2 2 は対称鍵である。

【0074】

ストレージ・システム 4 2 0 は、重複排除マネージャ 4 2 4 を備えている。重複排除マネージャ 4 2 4 は、クライアント 1 側 4 0 2 によって提供された重複排除情報に基づいて、重複するデータを位置特定するか、または識別するか、あるいはその両方を実行する。チャンキングの重複排除の場合、重複しているデータ・チャンクを識別するために、重複排除情報がチャンク・メタデータと共に使用される。当業者が本開示を読むときに理解するであろうように、重複排除マネージャ 4 2 4 は、任意の関連する重複排除動作を実行する。データ・チャンクを重複排除するために、任意の既知の重複排除技術が使用されてよい。

10

【0075】

ストレージ・システム 4 2 0 は、チャンク・ストレージ 4 2 6 を備えている。チャンク・ストレージ 4 2 6 は、従来技術において知られた任意のストレージ・コンポーネントを備えてよい。チャンク・ストレージ 4 2 6 は、従来技術において知られた任意の方法で、データ・チャンク、重複排除されたデータ、重複排除されていないデータ、暗号化データ、暗号化されていないデータ、重複排除情報などを格納してよい。1つの方法では、チャンク・ストレージ 4 2 6 は、データ・チャンク、重複排除されたデータ、重複排除されていないデータ、暗号化データ、暗号化されていないデータ、重複排除情報などをテーブルに格納する。

20

【0076】

ストレージ・システム 4 2 0 は、チャンク・メタデータ 4 2 8 を備えている。チャンク・メタデータ 4 2 8 は、ポインタ、チャンク長、チャンク・シグネチャなどを含む、重複排除されていないデータ・チャンクまたは重複排除されたデータ・チャンクあるいはその両方に関連付けられた任意のメタデータを含んでよい。チャンク・メタデータ 4 2 8 は、重複排除マネージャ 4 2 4 から新しいデータを受信することに応答して更新されてよい。

【0077】

ストレージ・システム 4 2 0 は、ストレージの秘密鍵エンクリプタ 4 3 0 を備えている。当業者が本開示を読むときに理解するであろうように、ストレージの秘密鍵エンクリプタ 4 3 0 は、ストレージの秘密暗号鍵 4 2 2 を使用してデータを暗号化してよい。

30

【0078】

ストレージ・システム 4 2 0 は、ストレージの秘密鍵デクリプタ 4 3 2 を備えている。当業者が本開示を読むときに理解するであろうように、ストレージの秘密鍵デクリプタ 4 3 2 は、ストレージの秘密暗号鍵 4 2 2 を使用して暗号化されたデータを復号してよい。

【0079】

ストレージ・システム 4 2 0 は、ストレージの一時鍵エンクリプタ 4 3 4 を備えている。当業者が本開示を読むときに理解するであろうように、ストレージの一時鍵エンクリプタ 4 3 4 は、ストレージの一時鍵 4 4 0 を使用してデータを暗号化してよい（下で詳細に説明される）。

40

【0080】

アーキテクチャ 4 0 0 は、クライアント 2 側 4 3 6 を含んでいる。図に示されているように、クライアント 2 側 4 3 6 は、保管時の暗号化ストレージを使用し、暗号化されていないデータをストレージ・システム 4 2 0 に送信する。少なくとも一部の方法では、クライアント 2 側 4 3 6 は、データの機密性に関する懸念を持っていないため、暗号化されていないデータを送信してよい。暗号化されていないデータとは、ストレージ・システムが暗号化されていない状態で受信するデータのことを指してよい。さまざまな方法では、従来技術において知られた任意の通信暗号化が、さまざまなコンポーネント間で実装されてよい。

50

【0081】

クライアント1側402は、ストレージ・システム420からの一時鍵に関する要求438を、ストレージ・システム420に発行してよい。1つの方法では、クライアント1側402は、一時鍵に関するストレージ・システム420に対する要求438によって、ストレージ・システム420とのセッションを確立する。一時鍵は、ストレージ・システム420がクライアント1側402から送信されたデータを復号できるように、データを暗号化するためにあってよい。ストレージの一時鍵440は、要求438に応答(442)して返されてよい。

【0082】

1つの方法では、クライアント1側402は、書き込みデータ444をメタ生成器406に送信する前に、書き込みデータ444を暗号化してよい。クライアント1側402は、書き込みデータ444、および書き込みデータ444を暗号化するために使用されるクライアントの秘密鍵(例えば、クライアントの秘密暗号鍵404)を示す鍵識別情報(ID: identification)を、メタ生成器406に送信する。

【0083】

好ましい方法では、メタ生成器406は、書き込みデータ444に関連付けられた重複排除情報(例えば、メタデータ)446を計算する。当業者が本開示を読むときに理解するであろうように、重複排除情報446は、重複排除方法の動作の詳細を含んでよい。1つの構成では、チャンキングの重複排除の場合、重複排除情報446は、チャンク開始位置、チャンク長、チャンク・シグネチャ、類似性情報などを含んでよい。1つの方法では、チャンク・シグネチャは、従来技術において知られた任意の方法で計算されてよい暗号ハッシュである。重複排除情報446は、一時鍵エンクリプタ410によってストレージの一時鍵440を使用して暗号化され、ストレージ・システム420(例えば、ストレージ・システム420内の重複排除マネージャ424)に送信されてよい。

【0084】

1つの方法では、書き込みデータ444は、メタ生成器406によって圧縮される。従来技術において知られた任意の形態の圧縮が使用されてよい。圧縮データ448が、クライアントの秘密鍵エンクリプタ408に送信され、クライアントの秘密暗号鍵404を使用して暗号化されてよい。クライアントの鍵によって暗号化されたデータ450およびクライアントの秘密暗号鍵404に関連付けられた鍵IDが、ストレージ・システム420(例えば、重複排除マネージャ424)に送信されてよい。別の方法では、書き込みデータ444が、クライアントの秘密鍵エンクリプタ408に送信され、クライアントの秘密暗号鍵404を使用して暗号化される。書き込みデータ444およびクライアントの秘密暗号鍵404に関連付けられた鍵IDが、ストレージ・システム420(例えば、重複排除マネージャ424)に送信されてよい。データ450は、ストレージ・システム420にとって不透明である(例えば、クライアントの秘密暗号鍵404は、ストレージ・システム420には使用することができない)。書き込みデータ444は、ストレージ・システム420にとって不透明である(例えば、クライアントの秘密暗号鍵404は、ストレージ・システム420には使用することができない)。

【0085】

さまざまな方法では、重複排除情報446および書き込みデータ444が、関連付けられているとしてタグ付けされるか、またはその他の方法で示されるか、あるいはその両方である。1つの態様では、重複排除情報446は、暗号化されていないデータに対して計算される。暗号化されていないデータは、チャンク・シグネチャを含んでよい。重複排除情報446は、暗号化データに対して計算されてよい。暗号化データに対して計算された重複排除情報446は、暗号化されたチャンクのシグネチャを含んでよい。

【0086】

好ましい方法では、重複排除マネージャ424は、暗号化された書き込みデータ444(例えば、ストレージ・システム420には使用することができない鍵を使用して暗号化されたデータ)および暗号化された重複排除情報446(例えば、ストレージ・システム

10

20

30

40

50

420が使用できる鍵を使用して暗号化されたメタデータ)を受信する。

【0087】

1つの構成では、重複排除マネージャ424は、一時鍵デクリプタ(ephemeralkey decrypter)を含む。当業者が本開示を読むときに理解するであろうように、一時鍵デクリプタは、ストレージの一時暗号鍵440を使用して暗号化されたデータを復号してよい。

【0088】

重複排除マネージャ424は、重複排除情報446に基づいて、重複するデータを位置特定するか、または識別するか、あるいはその両方を実行する。チャンキングの重複排除の場合、重複しているデータ・チャンクを識別するために、重複排除情報446がチャンク・メタデータ428と共に使用される。例えば、重複排除マネージャ424は、復号された重複排除情報446からのポインタ、シグネチャ、チャンク長などを、チャンク・メタデータ428に格納されたポインタ、シグネチャ、チャンク長などと比較する。当業者が本開示を読むときに理解するであろうように、重複排除マネージャ424は、任意の関連する重複排除動作を実行する。

【0089】

関連する重複排除動作を実行することに対応して、チャンク・メタデータ428に対して更新452が実行されてよい。更新452は、任意の重複排除されたチャンクの使用、残りすべてのチャンクに関連する情報の追加などに起因する変更を反映する。チャンク・メタデータ428は、特定のデータ・チャンクを暗号化する鍵IDに関する情報を含む。重複排除されていない新しいシグネチャ(例えば、チャンク・メタデータ428内にまだ存在していないシグネチャ)に関連付けられた残りのすべてのデータ・チャンクが、チャンク・ストレージ426に書き込まれる(454)。任意の新しい重複排除されていないデータ・チャンクが、チャンク・ストレージ426に書き込まれる(454)。

【0090】

好ましい構成では、重複排除マネージャ424が、クライアント(例えば、クライアント1側402)が復号することができるデータ・チャンクに対して、クライアントのために重複排除する。クライアント1側402が復号することができるデータ・チャンクは、クライアント1側402に属している鍵(例えば、クライアントの秘密暗号鍵404)を使用して暗号化されたデータ・チャンク、ストレージ・システム420に属している鍵(例えば、ストレージの秘密暗号鍵422)、暗号化されていないデータなどを含む。クライアント1側402は、クライアント1側402がアクセスすることができない一意のクライアントの秘密暗号鍵を使用して暗号化されたデータを、復号することができなくてよい。

【0091】

メタデータの更新452に関連付けられた残りのすべての重複排除されていないデータ・チャンク456は、ストレージの秘密鍵エンクリプタ430で、ストレージの秘密暗号鍵422を使用して暗号化されてよい。メタデータの更新452に関連付けられた重複排除されていないデータ・チャンク456とは、チャンク・メタデータ428内にすでに存在するデータ・チャンクのことを指してよく、メタデータの更新452は、対応する格納位置への追加のポインタを含む。

【0092】

当業者が本開示を読むときに理解するであろうように、好ましい方法では、ストレージ・システム420は、保管時の暗号化データ、暗号化されていないデータ、共有鍵によって暗号化されたデータなどのようなデータを復号することができるため、そのようなデータにアクセスしてよい。

【0093】

クライアント2側436は、図に示されているように、保管時の暗号化を使用して、暗号化されていないデータ458をストレージ・システム420に送信する。暗号化されていないデータ458は、暗号化されてチャンク・ストレージ426に格納されてよい。暗号化されていないデータ458は、読み取り要求に対応して、暗号化されずにクライアン

10

20

30

40

50

ト 2 側 4 3 6 で受信されてよい。好ましい方法では、重複排除マネージャ 4 2 4 は、クライアント 1 側 4 0 2 に関して実行される重複排除に実質的に類似する方法で重複排除を実行する。暗号化されていないデータ 4 5 8 は、ストレージ・システムの鍵（例えば、ストレージの秘密暗号鍵 4 2 2）を使用して暗号化されたデータ・チャンクまたは暗号化されていないデータ・チャンクあるいはその両方に対して、重複排除されてよい。

【0094】

一部の構成では、重複排除マネージャ 4 2 4 は、メタ生成器 4 0 6 の機能に実質的に類似する機能を実行するメタ生成器（図示されていない）を備えてよい。残りのすべての重複排除されていないデータ・チャンク 4 5 6 は、ストレージの秘密鍵エンクリプタ 4 3 0 で、ストレージの秘密暗号鍵 4 2 2 を使用して暗号化されてよい。暗号化された重複排除されていないデータ・チャンク 4 6 0 が、チャンク・ストレージ 4 2 6 に書き込まれてよい。ストレージ・システム 4 2 0 は、本明細書に記載されたプロセスに従って配信された任意のデータを復号してよい。データ・チャンク 4 6 0 はすべて、任意の鍵を使用して暗号化された任意のデータ・チャンクに対して重複排除するために使用されてよい。

【0095】

図 5 は、さまざまな構成に従う高レベルのアーキテクチャである。アーキテクチャ 5 0 0 は、さまざまな構成において、特に図 1～図 4 および図 6～図 9 に示された環境のいずれかにおいて、本発明に従って実装されてよい。当然、当業者が本説明を読んだときに理解するであろうように、図 5 において具体的に説明された要素よりも多いか、または少ない要素がアーキテクチャ 5 0 0 に含まれてよい。

【0096】

アーキテクチャ 5 0 0 は、対称鍵暗号化を使用する読み取り動作の例示的な実装を示している。好ましい構成では、読み取り動作は、少なくとも 2 つの鍵（例えば、クライアントの秘密暗号鍵およびストレージの一時鍵）を使用して暗号化されたデータ・チャンクを受信することを含む。例えば、読み取り動作は、第 1 の鍵を使用して暗号化された 1 つまたは複数のデータ・チャンクおよび第 2 の鍵を使用して暗号化された 1 つまたは複数のデータ・チャンクを受信することを含む。アーキテクチャ 5 0 0 は、クライアント 1 側 4 0 2 を含んでいる。

【0097】

クライアント 1 側 4 0 2 は、クライアントの秘密暗号鍵 4 0 4 を含んでいる。好ましい方法では、クライアントの秘密暗号鍵 4 0 4 はクライアント 1 側 4 0 2 から離れない。この好ましい方法では、ストレージ・システム 4 2 0 は、データ転送のどの時点でも、クライアント側の暗号化データを暗号化されていない状態で見ることはない。この好ましい方法では、クライアントの秘密暗号鍵 4 0 4 は、ストレージ・システム 4 2 0、クライアント 2 側 4 3 6（下で詳細に説明される）、任意のその他のクライアント、任意のその他のストレージ・システムなどと共有されない。クライアントの秘密暗号鍵 4 0 4 は、対称鍵、非対称鍵などであってよい。図に示されているように、クライアントの秘密暗号鍵 4 0 4 は対称鍵である。

【0098】

クライアント 1 側 4 0 2 は、メタ生成器 4 0 6 を含んでいる。好ましい方法では、メタ生成器 4 0 6 は、データに関連付けられた重複排除メタデータを計算する。さまざまな構成では、メタ生成器 4 0 6 は、クライアント 1 側 4 0 2 またはストレージ・システム 4 2 0 あるいはその両方に配置されてよい。

【0099】

好ましい方法では、メタ生成器 4 0 6 は、クライアント 1 側 4 0 2 でデータに対してメタデータ動作を実行し、データ内のデータ重複排除の機会を識別する。メタ生成器 4 0 6 は、従来技術において知られた任意のデータ重複排除技術を使用してデータ重複排除の機会を識別する。1 つの態様では、メタ生成器 4 0 6 は、スライド・ウィンドウ可変チャンク重複排除技術を使用してデータ重複排除の機会を識別する。メタ生成器 4 0 6 は、重複排除シグネチャ（例えば、ローリング最小ハッシュ、暗号ハッシュなど）を識別するか、

または計算するか、あるいはその両方を実行する。メタ生成器 406 は、ポインタ、チャンク長、チャンク・シグネチャなどを含む、データ重複排除の機会に関連付けられたメタデータ（例えば、重複排除情報）を生成する。重複排除情報は、以下で詳細に説明されるように、ストレージ・システムによって提供された一時鍵を使用して暗号化されてよい。

【0100】

1つの方法では、メタ生成器 406 は、従来技術において知られた任意のデータ圧縮技術を使用してデータを圧縮する。一部の方法では、チャンキングの前または後あるいはその両方で、さまざまな圧縮技術が適用されてよい。1つの構成では、チャンキング前の圧縮は、チャンキングの性能を改善する種類の圧縮であってよい。別の構成では、チャンキング後の圧縮は、結果として得られるチャンク・サイズを最小化することに向けて調整されてよい。

10

【0101】

シグネチャの計算は、圧縮の前または後あるいはその両方で、データ・チャンクに対して実行されてよい。圧縮前のシグネチャの計算は、読み取り時の復元後にシグネチャを検証できるようにする。圧縮後のシグネチャの計算は、チャンクが圧縮された状態にある間に、シグネチャをチェックできるようにする。

【0102】

クライアント 1 側 402 は、クライアントの秘密鍵エンクリプタ 408 を備えている。当業者が本開示を読むときに理解するであろうように、クライアントの秘密鍵エンクリプタ 408 は、クライアントの秘密暗号鍵 404 を使用してデータを暗号化してよい。

20

【0103】

クライアント 1 側 402 は、一時鍵エンクリプタ 410 を備えている。当業者が本開示を読むときに理解するであろうように、一時鍵エンクリプタ 410 は、一時鍵 504 を使用してデータを暗号化してよい（下で詳細に説明される）。

【0104】

クライアント 1 側 402 は、デチャンカ 412 を備えている。デチャンカ 412 は、ストレージ・システムから受信されたデータ・チャンクをチャンク解除し、データ・チャンクは、複数の暗号鍵を使用して暗号化される。複数の暗号鍵を使用して暗号化されたデータ・チャンクとは、1つまたは複数のデータ・チャンクが第1の鍵を使用して暗号化され、1つまたは複数のデータ・チャンクが第2の鍵を使用して暗号化される、データ・チャンクのセットのことを指すのが好ましい。デチャンカ 412 は、どの暗号鍵を使用してデータ・チャンクが暗号化されているかを識別し、そのデータ・チャンクを対応するデクリプタ（例えば、クライアントの秘密鍵デクリプタ 414、一時鍵デクリプタ 416 など）に転送してよい。当業者が本開示を読むときに理解するであろうように、各デクリプタは、対応する鍵を使用して暗号化されたデータ・チャンクを復号してよい。

30

【0105】

アーキテクチャ 500 は、ストレージ・システム 420 を含んでいる。ストレージ・システム 420 は、従来技術において知られた任意の種類のストレージ・システムであってよい。ストレージ・システム 420 が、本明細書において示されたコンポーネントよりも多いかまたは少ないコンポーネントを含んでよいということが、当業者によって理解されるべきである。ストレージ・システム 420 は、本明細書に記載されたさまざまな重複排除動作を実行するのが好ましい。

40

【0106】

ストレージ・システム 420 は、ストレージの秘密暗号鍵 422 を含んでいる。好ましい方法では、ストレージの秘密暗号鍵 422 はストレージ・システム 420 から離れない。この好ましい方法では、ストレージの秘密暗号鍵 422 は、クライアント 1 側 402、クライアント 2 側 436、任意のその他のクライアント、任意のその他のストレージ・システムなどと共有されない。ストレージの秘密暗号鍵 422 は、対称鍵、非対称鍵などであってよい。図に示されているように、ストレージの秘密暗号鍵 422 は対称鍵である。

【0107】

50

ストレージ・システム 420 は、重複排除マネージャ 424 を備えている。重複排除マネージャ 424 は、クライアント 1 側 402 によって提供された重複排除情報に基づいて、重複するデータを位置特定するか、または識別するか、あるいはその両方を実行する。チャンキングの重複排除の場合、重複しているデータ・チャンクを識別するために、重複排除情報がチャンク・メタデータと共に使用される。当業者が本開示を読むときに理解するであろうように、重複排除マネージャ 424 は、任意の関連する重複排除動作を実行する。従来技術において知られた任意の重複排除技術が使用されてよい。

【0108】

ストレージ・システム 420 は、チャンク・ストレージ 426 を備えている。チャンク・ストレージ 426 は、従来技術において知られた任意のストレージ・コンポーネントを備えてよい。チャンク・ストレージ 426 は、従来技術において知られた任意の方法で、データ・チャンク、重複排除されたデータ、重複排除されていないデータ、暗号化データ、暗号化されていないデータ、重複排除情報などを格納してよい。1つの方法では、チャンク・ストレージ 426 は、データ・チャンク、重複排除されたデータ、重複排除されていないデータ、暗号化データ、暗号化されていないデータ、重複排除情報などをテーブルに格納する。

10

【0109】

ストレージ・システム 420 は、チャンク・メタデータ 428 を備えている。チャンク・メタデータ 428 は、ポインタ、チャンク長、チャンク・シグネチャなどを含む、重複排除されていないデータ・チャンクまたは重複排除されたデータ・チャンクあるいはその両方に関連付けられた任意のメタデータを含んでよい。

20

【0110】

ストレージ・システム 420 は、ストレージの秘密鍵エンクリプタ 430 を備えている。当業者が本開示を読むときに理解するであろうように、ストレージの秘密鍵エンクリプタ 430 は、ストレージの秘密暗号鍵 422 を使用してデータを暗号化してよい。

【0111】

ストレージ・システム 420 は、ストレージの秘密鍵デクリプタ 432 を備えている。当業者が本開示を読むときに理解するであろうように、ストレージの秘密鍵デクリプタ 432 は、ストレージの秘密暗号鍵 422 を使用して暗号化されたデータを復号してよい。

【0112】

ストレージ・システム 420 は、ストレージの一時鍵エンクリプタ 434 を備えている。当業者が本開示を読むときに理解するであろうように、ストレージの一時鍵エンクリプタ 434 は、ストレージの一時暗号鍵 504 を使用してデータを暗号化してよい。

30

【0113】

アーキテクチャ 500 は、クライアント 2 側 436 を含んでいる。クライアント 2 側 436 は、図に示されているように、保管時の暗号化を使用して、暗号化されていないデータ 458 をストレージ・システム 420 に送信する。暗号化されていないデータ 458 は、暗号化されてチャンク・ストレージ 426 に格納されてよい。暗号化されていないデータ 458 は、読み取り要求に応答して、暗号化されずにクライアント 2 側 436 で受信されてよい。好ましい方法では、重複排除マネージャ 424 は、クライアント 1 側 402 に関して実行される重複排除に実質的に類似する方法で重複排除を実行する。暗号化されていないデータ 458 は、ストレージ・システムの鍵（例えば、ストレージの秘密暗号鍵 422）を使用して暗号化されたデータ・チャンクまたは暗号化されていないデータ・チャンクあるいはその両方に対して、重複排除されてよい。

40

【0114】

クライアント 1 側 402 は、ストレージ・システム 420 からの一時鍵に関する要求 502 を、ストレージ・システム 420 に発行してよい。1つの方法では、クライアント 1 側 402 は、一時鍵に関するストレージ・システム 420 に対する要求 502 によって、ストレージ・システム 420 とのセッションを確立する。一時鍵は、返されたデータ・チャンクを暗号化するためであってよい。一時鍵 504 は、要求 502 に応答（506）し

50

て返されてよい。一時鍵 5 0 4 は、読み取り時に、重複排除および少なくとも一部のデータ・チャンクに適用されてよい。クライアント 1 側 4 0 2 は、読み取り要求 5 0 8 をストレージ・システム 4 2 0 に送信する。

【0115】

好ましい方法では、重複排除マネージャ 4 2 4 は、チャンク・メタデータ 4 2 8 を使用して、読み取り要求 5 0 8 に関連付けられたデータ・チャンクを識別する。識別されたデータ・チャンクが、チャンク・ストレージ 4 2 6 から読み取られる (5 1 0)。重複排除マネージャ 4 2 4 は、読み取り要求 5 0 8 からのポインタ、シグネチャ、チャンク長などを、チャンク・メタデータ 4 2 8 に格納された任意のポインタ、シグネチャ、チャンク長などと比較する。好ましい方法では、チャンク・メタデータ 4 2 8 は、各データ・チャンクを暗号化するために使用される鍵の識別子を含む。クライアントの鍵を使用して暗号化されたデータ・チャンクが、デチャンカ 4 1 2 に送信されてよい (5 1 2)。クライアントの鍵を使用してデータ・チャンクが暗号化されたことのインジケータも送信されてよい (5 1 2)。

10

【0116】

ストレージの秘密暗号鍵 4 2 2 を使用して暗号化されたデータ・チャンク 5 1 4 が、ストレージの秘密鍵デクリプタ 4 3 2 に送信されてよい。ストレージの秘密鍵デクリプタ 4 3 2 は、ストレージの秘密暗号鍵 4 2 2 を使用して暗号化されたデータを復号してよい。ストレージの秘密鍵デクリプタ 4 3 2 の出力 5 1 6 が、ストレージの一時鍵エンクリプタ 4 3 4 に送信されてよい。当業者が本開示を読むときに理解するであろうように、ストレージの一時鍵エンクリプタ 4 3 4 は、ストレージの一時暗号鍵 5 0 4 を使用してデータを暗号化してよい。

20

【0117】

1つの構成では、ストレージの一時鍵エンクリプタ 4 3 4 の出力 5 1 8 が、デチャンカ 4 1 2 に送信されてよい。一時鍵 5 0 4 を使用して出力 5 1 8 が暗号化されたことのインジケータが、デチャンカ 4 1 2 に送信されてよい。デチャンカ 4 1 2 は、ストレージ・システムから受信されたデータ・チャンクをチャンク解除し、データ・チャンクは、複数の暗号鍵を使用して暗号化される。複数の暗号鍵を使用して暗号化されたデータ・チャンクとは、1つまたは複数のデータ・チャンクが第1の鍵を使用して暗号化され、1つまたは複数のデータ・チャンクが第2の鍵を使用して暗号化される、データ・チャンクのセットのことを指すのが好ましい。デチャンカ 4 1 2 は、どの暗号鍵を使用してデータ・チャンクが暗号化されているかを識別し、そのデータ・チャンクを対応するデクリプタ (例えば、クライアントの秘密鍵デクリプタ 4 1 4、一時鍵デクリプタ 4 1 6 など) に転送してよい。具体的には、デチャンカ 4 1 2 が、クライアントの鍵によって暗号化されたデータ・チャンク 5 2 0 をクライアントの秘密鍵デクリプタ 4 1 4 に送信するか、またはデチャンカ 4 1 2 が、一時鍵によって暗号化されたデータ・チャンク 5 2 2 を一時鍵デクリプタ 4 1 6 に送信するか、あるいはその両方が実行される。当業者が本開示を読むときに理解するであろうように、各デクリプタは、対応する鍵を使用して暗号化されたデータ・チャンクを復号してよい。さまざまな方法では、デチャンカ 4 1 2 は、従来技術において知られた任意の復元動作を実行する。デチャンカ 4 1 2 は、データ・チャンクをマージして、クライアント 1 側 4 0 2 のためのデータ 5 2 4 を生成してよい。

30

40

【0118】

当業者が本開示を読むときに理解するであろうように、クライアントの秘密鍵デクリプタ 4 1 4 は、クライアントの秘密暗号鍵 4 0 4 を秘密に保つためにデチャンカ 4 1 2 から分離しているクライアント 1 側 4 0 2 によって提供されてよい。

【0119】

クライアントの鍵を使用して暗号化されたデータ・チャンク、およびストレージ・システム 4 2 0 から送信されたストレージの一時鍵エンクリプタ 4 3 4 の出力 5 1 8 は、各チャンクが正しく配信され、適切な鍵を使用して復号されたことを検証するためのチャンク・シグネチャを含んでいるデータ・チャンク・メタデータを含んでよい。データ・チャン

50

ク・メタデータは、エンドツーエンドのチェックを提供してよい。一部の方法では、クライアントの秘密鍵が使用されるときに、メタ生成器 406 によって、データ・チャンクごとのシグネチャが、暗号化されていないデータに対して計算されてよい。データ・チャンクごとのシグネチャは、重複排除マネージャ 424 内のメタ生成器（図示されていない）によって、保管時の暗号化されたデータ・チャンクまたは暗号化されていないデータ・チャンクあるいはその両方に対して計算されてよい。

【0120】

アーキテクチャ 500 の好ましい構成では、クライアントの秘密暗号鍵 404 がクライアント 1 側 402 から離れることなく、さまざまな圧縮技術または重複排除技術あるいはその両方が実行されてよい。クライアントの秘密暗号鍵 404 は、ストレージ・システム 420、クライアント 2 側 436、任意のその他のクライアント、任意のサード・パーティのサービス、任意のその他のストレージ・システムなどには、使用することができない。重複排除プロセスの一部（例えば、アーキテクチャ 500 におけるチャンキング・プロセス）は、クライアント側で実行されるのが好ましく、それによって、クライアントの秘密暗号鍵 404 の外部の可視性を大幅に低減するか、または除去する。

【0121】

前述したように、一時鍵 504 は、読み取り時に、データの機密性に役立つように、重複排除および少なくとも一部のデータ・チャンクに適用されてよい。一時鍵 504 は、スタンド・トランスポート暗号化（stand transport encryption）（例えば、トランスポート層セキュリティ（TLS：transport layer security））であってよい。重複排除メタデータおよびチャンク・データは、暗号化されていない状態でトランスポート層に提供されてよい。この方法では、クライアントの鍵によって暗号化されたデータが、一時鍵 504 を使用して再び暗号化されてよい。

【0122】

好ましい方法では、従来の重複排除動作とは対照的に、暗号化されているチャンクの鍵を示すために、追加のチャンク・メタデータが各データ・チャンクに関連付けられる。異なるクライアントが、同じである（例えば、同一のシグネチャで重複する）データ・チャンクを含んでよい。しかし、異なるクライアントからの重複するデータ・チャンクが、異なる鍵を使用して暗号化される場合、クライアント 2 のデータ・チャンクに対するクライアント 1 のデータ・チャンクの従来の重複排除は、鍵の共有を必要とする。全く対照的に、本開示のさまざまな構成は、データ・チャンク、インデックス、タグ、シグネチャ、鍵 ID などを暗号化するために使用される鍵を識別する情報を含んでいるチャンク識別子を提供する。クライアントが 2 つ以上の鍵を使用している場合、クライアント 1 側 402 によって、任意の入出力（I/O：input/output）動作（例えば、ストレージの一時鍵 440 に対する要求 502 およびストレージの一時鍵 440 を含む応答 506 を含んでいる鍵の交換を含む）の前に、鍵 ID に関する情報がストレージ・システム 420 に提供されてよい。クライアントがすべてのデータ暗号化に単一の鍵を使用する場合、鍵 ID は、クライアントを一意に識別する。

【0123】

データの機密性の文脈において、鍵 ID は鍵自体ではない。例えば、提供された鍵 ID は、クライアントの秘密暗号鍵 404 の SHA 256 などの暗号ハッシュであってよい。別の例では、鍵 ID は、クライアントの秘密暗号鍵 404 の一意のインデックス番号であってよい。ストレージ・システム 420 は、内部参照鍵 ID（internal reference key ID）を作成してよい。例えば、ストレージ・システム 420 は、クライアントの秘密暗号鍵 404 の ID のための鍵のインデックス・テーブルを作成してよい。

【0124】

共通のシグネチャおよび異なる鍵 ID を含んでいる少なくとも 2 つのクライアントからの暗号化データに応答して、重複排除マネージャ 424 は、重複排除を実行するにはシグネチャと鍵 ID の両方が一致しなければならないため、暗号化データを重複排除しない。当業者が本開示を読むときに理解するであろうように、ストレージの鍵を使用して暗号化

10

20

30

40

50

されたデータ・チャンクまたは暗号化されていないデータ・チャンクあるいはその両方が、いずれかのクライアントの鍵を使用して暗号化されたデータ・チャンクに対して重複排除されてよい。

【0125】

重複排除ストレージ（例えば、チャンク・ストレージ426）内のチャンク・メタデータ428は、さまざまな方法では、データのいくつかの異なる部分がデータ・チャンクを指し示しているかを示すために、参照カウンタを含んでよい。ソース鍵IDおよび各鍵IDの参照カウンタを示すメタデータが、各データ・チャンクに追加されてよい。要求の鍵IDのゼロ以外の参照カウンタを含まないチャンクを読み取ろうとする試みが、傍受されてよい。この傍受は、重複排除されたデータ・チャンクに対するアクセス制御を提供する。例えば、クライアントの秘密暗号鍵404に関連付けられた鍵IDを使用するクライアント1側402は、重複排除されたチャンクAを含んでいるデータを含んでよい。チャンクAは、チャンクAに対して重複排除されたクライアントの秘密暗号鍵404を使用して暗号化されたデータの部分のみであってよい。クライアント1側402は、重複排除されたチャンクAを含んでいるデータを削除してよい。秘密暗号鍵404に関連付けられたクライアント1側402の鍵IDの参照カウンタが、0にデクリメントされる。クライアント1側402の秘密暗号鍵404に関連付けられた鍵IDを使用してチャンクAを読み取ろうとするその後のすべての試みが阻止されてよく、ゼロ以外の関連する参照カウンタを含むすべての要求が許可されてよい。

10

【0126】

クライアントの秘密暗号鍵404を使用して暗号化されたデータが最初に重複排除されない場合、このデータは、一致するシグネチャを含むストレージの秘密暗号鍵422を使用して暗号化されたデータ・チャンクが書き込まれるか、または暗号化されていないデータ・チャンクが書き込まれるか、あるいはその両方に応答して、重複排除されてよい。最初に重複排除されないデータ・チャンクを重複排除するためのプロセスは、本明細書に記載されたさまざまな構成において、バックポーティングと呼ばれてよい。バックポートは、（例えば、ストレージ・システム420が使用できる鍵を使用して暗号化された）新しいデータ・チャンクを格納する重複排除動作を含んでよい。バックポートは、一致するシグネチャを含んでいるが、異なる鍵IDを含んでいるすべての既存のチャンクを削除することを含む。格納されたデータ・チャンクまたは削除されたデータ・チャンクあるいはその両方のメタデータが、新しいデータ・チャンクを指し示すように（例えば、新しいデータ・チャンクが最初に存在していたかのように）変換されてよい。異なる鍵でデータ・チャンクが書き込まれる順序は、最終的な重複排除の効率に影響を与えない。データ・チャンクの多くのプライベート・コピーが存在する場合、データ・チャンクのコピーのその後のすべての保管時の書き込みまたは暗号化されない書き込みあるいはその両方は、本明細書に記載されたバックポーティング技術を使用して、システム上に格納される容量を低減する。

20

30

【0127】

本明細書に記載されたさまざまな構成は、非対称暗号鍵（例えば、PKI）を使用して実現されてよい。対称鍵暗号化は、相対的に大きいデータ・セットの場合、非対称鍵暗号化よりも好ましいことがある。非対称鍵暗号化は、ストレージ・システムと共有される公開鍵、およびストレージ・システムと共有されない秘密鍵を含んでいるクライアントを含む。ストレージ・システムは、クライアントと共有される公開鍵、およびクライアントと共有されない秘密鍵の両方を含んでよい。PKIにおける公開鍵は秘密ではないが、公開鍵でデータを暗号化するには、データを復号するための対応する秘密鍵が必要である。

40

【0128】

図6は、さまざまな構成に従う高レベルのアーキテクチャである。アーキテクチャ600は、さまざまな構成において、特に図1～図5および図7～図9に示された環境のいずれかにおいて、本発明に従って実装されてよい。当然、当業者が本説明を読んだときに理解するであろうように、図6において具体的に説明された要素よりも多いか、または少な

50

い要素がアーキテクチャ 600 に含まれてよい。

【0129】

アーキテクチャ 600 は、非対称鍵暗号化を使用する書き込み動作の例示的な実装を示している。アーキテクチャ 600 は、クライアント 1 側 602 を含んでいる。

【0130】

クライアント 1 側 602 は、クライアントの秘密暗号鍵 604 および関連するクライアントの公開鍵 606 を含んでいる。クライアント 1 側 602 は、データをストレージ・システム 608 に送信する前に、クライアントの公開鍵 606 を使用してデータを暗号化する。対称鍵暗号化の場合とは対照的に、ストレージ・システム 608 は、クライアントの公開鍵 606 を使用して暗号化されたすべてのデータをクライアント 1 側 602 に返してよい。

10

【0131】

アーキテクチャ 600 は、ストレージ・システム 608 を含んでいる。ストレージ・システム 608 は、ストレージの秘密鍵 610 および関連するストレージの公開鍵 612 を含んでいる。

【0132】

ストレージ・システム 608 は、クライアントの公開鍵 606 を使用して暗号化されたすべてのデータをクライアント 1 側 602 に返してよい。ストレージ・システム 608 は、デチャンカ 614 を備えている。デチャンカ 614 は、クライアントの秘密暗号鍵 604 の知識を持たずに動作してよい。

20

【0133】

好ましい方法では、データを書き込む前に、クライアント 1 側 602 およびストレージ・システム 608 が、公開鍵を交換する。クライアントの公開鍵 606 が、ストレージ・システム 608 に送信されてよい (616)。ストレージの公開鍵 612 が、クライアント 1 側 602 に送信されてよい (618)。クライアント 1 側 602 は、書き込みデータ 620 をメタ生成器 622 に送信する。

【0134】

好ましい方法では、メタ生成器 622 は、書き込みデータ 620 に関連付けられた重複排除情報 (例えば、メタデータ) 624 を計算する。当業者が本開示を読むときに理解するであろうように、重複排除情報 624 は、重複排除方法の動作の詳細を含んでよい。1つの構成では、チャンキングの重複排除の場合、重複排除情報 624 は、チャンク開始位置、チャンク長、チャンク・シグネチャ、類似性情報などを含んでよい。1つの方法では、チャンク・シグネチャは、従来技術において知られた任意の方法で計算されてよい暗号ハッシュである。重複排除情報 624 は、ストレージの公開鍵エンクリプタ 626 によって、ストレージの公開鍵 612 を使用して暗号化されてよい。重複排除情報 624 が、ストレージ・システム 608 に送信されてよい。

30

【0135】

1つの方法では、書き込みデータ 620 は、メタ生成器 622 によって圧縮される。従来技術において知られた任意の形態の圧縮が使用されてよい。圧縮データ 628 が、クライアントの公開鍵エンクリプタ 630 に送信され、クライアントの公開鍵 606 を使用して暗号化されてよい。クライアントの公開鍵によって暗号化されたデータ 632 が、ストレージ・システム 608 (例えば、重複排除マネージャ 634) に送信されてよい。クライアントの公開鍵によって暗号化されたデータ 632 および重複排除情報 624 が、ストレージ・システム 608 (例えば、重複排除マネージャ 634) に送信されてよい。データ 632 は、ストレージ・システム 608 にとって不透明である (例えば、クライアントの秘密暗号鍵 604 は、ストレージ・システム 608 には使用することができない)。重複排除情報 624 は、ストレージの公開鍵エンクリプタ 626 によって、ストレージの公開鍵 612 を使用して暗号化されてよい。別の方法では、書き込みデータ 620 は、ストレージの公開鍵エンクリプタ 626 によって、ストレージの公開鍵 612 を使用して暗号化されてよい。暗号化された書き込みデータ 620 および重複排除情報 624 が、ストレ

40

50

ージ・システム 6 0 8（例えば、重複排除マネージャ 6 3 4）に送信されてよい。さまざまな方法では、重複排除情報 6 2 4 および書き込みデータ 6 2 0 が、関連付けられているとしてタグ付けされるか、またはその他の方法で示されるか、あるいはその両方である。

【0 1 3 6】

1つの構成では、重複排除マネージャ 6 3 4 は、ストレージの秘密鍵デクリプタを含む。当業者が本開示を読むときに理解するであろうように、ストレージの秘密鍵デクリプタは、ストレージの秘密鍵 6 1 0 を使用して暗号化されたデータを復号してよい。

【0 1 3 7】

重複排除マネージャ 6 3 4 は、重複排除情報 6 2 4 に基づいて、重複するデータを位置特定するか、または識別するか、あるいはその両方を実行する。チャンキングの重複排除の場合、重複しているデータ・チャンクを識別するために、重複排除情報 6 2 4 がチャンク・メタデータ 4 2 8 と共に使用される。例えば、重複排除マネージャ 6 3 4 は、復号された重複排除情報 6 2 4 からのポインタ、シグネチャ、チャンク長などを、チャンク・メタデータ 4 2 8 に格納された任意のポインタ、シグネチャ、チャンク長などと比較する。当業者が本開示を読むときに理解するであろうように、重複排除マネージャ 6 3 4 は、任意の関連する重複排除動作を実行する。従来技術において知られた任意の重複排除技術が使用されてよい。

【0 1 3 8】

関連する重複排除動作を実行することに対応して、チャンク・メタデータ 4 2 8 に対して更新 4 5 2 が実行されてよい。更新 4 5 2 は、任意の重複排除されたチャンクの使用、残りのすべてのチャンクに関連する情報の追加などに起因する変更を反映する。重複排除されていない残りのすべてのデータ・チャンクが、チャンク・ストレージ 4 2 6 に書き込まれる（4 5 4）。チャンク・メタデータ 4 2 8 は、特定のデータ・チャンクを暗号化する鍵 ID に関する情報を含む。

【0 1 3 9】

好ましい構成では、重複排除マネージャ 6 3 4 が、クライアント（例えば、クライアント 1 側 6 0 2）が復号することができるデータ・チャンクに対して、クライアントのために重複排除する。クライアント 1 側 6 0 2 が復号することができるデータ・チャンクは、クライアント 1 側 6 0 2 に属している鍵（例えば、クライアントの秘密暗号鍵 6 0 4）を使用して暗号化されたデータ・チャンク、ストレージ・システム 6 0 8 に属している鍵（例えば、ストレージの秘密鍵 6 1 0）、暗号化されていないデータなどを含む。残りのすべての重複排除されていないデータ・チャンク 4 5 6 は、ストレージの秘密鍵エンクリプタ 6 3 6 で、ストレージの秘密鍵 6 1 0 を使用して暗号化されてよい。

【0 1 4 0】

当業者が本開示を読むときに理解するであろうように、好ましい方法では、ストレージ・システム 6 0 8 は、保管時の暗号化データ、暗号化されていないデータ、共有鍵によって暗号化されたデータなどのようなデータを復号することができるため、そのようなデータにアクセスしてよい。

【0 1 4 1】

ストレージ・システム 6 0 8 は、ストレージの秘密鍵デクリプタ 6 3 8 を備えている。当業者が本開示を読むときに理解するであろうように、ストレージの秘密鍵デクリプタ 6 3 8 は、ストレージの秘密暗号鍵 6 1 0 を使用して暗号化されたデータを復号してよい。

【0 1 4 2】

ストレージ・システム 6 0 8 は、クライアントの公開鍵エンクリプタ 6 4 0 を備えている。当業者が本開示を読むときに理解するであろうように、クライアントの公開鍵エンクリプタ 6 4 0 は、クライアントの公開鍵 6 0 6 を使用してデータを暗号化してよい。

【0 1 4 3】

1つの方法では、ストレージ・システム 6 0 8 は、デチャンカ 6 1 4 を介して、従来技術において知られた任意の方法でクライアントの秘密鍵デクリプタ 6 4 2 によって復号されるデータ・チャンクを返してよい。

【0144】

クライアント2側436は、図に示されているように、保管時の暗号化を使用して、暗号化されていないデータ458をストレージ・システム608に送信する。暗号化されていないデータ458は、暗号化されてチャンク・ストレージ426に格納されてよい。暗号化されていないデータ458は、読み取り要求に応答して、暗号化されずにクライアント2側436で受信されてよい。好ましい方法では、重複排除マネージャ634は、クライアント1側602に関して実行される重複排除に実質的に類似する方法で重複排除を実行する。暗号化されていないデータ458は、ストレージ・システムの鍵（例えば、ストレージの秘密鍵610）を使用して暗号化されたデータ・チャンクまたは暗号化されていないデータ・チャンクあるいはその両方に対して、重複排除されてよい。残りのすべての重複排除されていないデータ・チャンク456は、ストレージの秘密鍵エンクリプタ636で、ストレージの秘密鍵610を使用して暗号化されてよい。暗号化された重複排除されていないデータ・チャンク460が、チャンク・ストレージ426に書き込まれてよい。ストレージ・システム608は、前述の動作に従って配信された任意のデータを復号してよい。データ・チャンク460はすべて、本明細書において暗号化された任意のチャンクに対して重複排除するために使用されてよい。

10

【0145】

図7は、さまざまな構成に従う高レベルのアーキテクチャである。アーキテクチャ700は、さまざまな構成において、特に図1～図6および図8～図9に示された環境のいずれかにおいて、本発明に従って実装されてよい。当然、当業者が本説明を読んだときに理解するであろうように、図7において具体的に説明された要素よりも多いか、または少ない要素がアーキテクチャ700に含まれてよい。

20

【0146】

アーキテクチャ700は、非対称鍵暗号化を使用する読み取り動作の例示的な実装を示している。アーキテクチャ600は、クライアント1側602を含んでいる。クライアント1側602は、クライアントの秘密暗号鍵604および関連するクライアントの公開鍵606を含んでいる。アーキテクチャ700は、ストレージ・システム608を含んでいる。ストレージ・システム608は、ストレージの秘密鍵610および関連するストレージの公開鍵612を含んでいる。

【0147】

好ましい方法では、データを書き込む前に、クライアント1側602およびストレージ・システム608が、公開鍵を交換する。クライアントの公開鍵606が、ストレージ・システム608に送信されてよい（616）。ストレージの公開鍵612が、クライアント1側602に送信されてよい（618）。クライアント1側602は、書き込みデータ620をメタ生成器622に送信する。

30

【0148】

クライアント1側602は、読み取り要求702をストレージ・システム608（例えば、重複排除マネージャ634）に送信する。重複排除マネージャ634は、チャンク・メタデータ428を使用して、読み取り要求702に関連付けられたデータ・チャンクを位置特定するか、または識別するか、あるいはその両方を実行する。識別されたデータ・チャンクが、チャンク・ストレージ426から読み取られてよい。チャンク・メタデータ428は、データ・チャンクを暗号化するために使用される鍵の識別子を含む。クライアントの秘密暗号鍵604を使用して暗号化されたデータ・チャンクが、デチャンカ614に送信されてよい。クライアントの秘密暗号鍵604を使用してデータ・チャンクが暗号化されたことのインジケータが、デチャンカ614に送信されてよい。

40

【0149】

ストレージの秘密鍵610を使用して暗号化されたとして識別されたデータ・チャンクが、ストレージの秘密鍵デクリプタ638に送信されてよい（704）。当業者が本開示を読むときに理解するであろうように、ストレージの秘密鍵デクリプタ638は、ストレージの秘密暗号鍵610を使用して暗号化されたデータを復号してよい。ストレージの秘

50

密鍵デクリプタ 638 の出力 706 が、クライアントの公開鍵エンクリプタ 640 に送信されてよい。当業者が本開示を読むときに理解するであろうように、クライアントの公開鍵エンクリプタ 640 は、クライアントの公開鍵 606 を使用してデータを暗号化してよい。クライアントの公開鍵エンクリプタ 640 の出力 708 が、デチャンカ 614 に送信されてよい。対称鍵の実装とは対照的に、デチャンカ 614 に送信されるすべてのデータ・チャンクが、クライアントの公開鍵 606 を使用して暗号化される。デチャンカ 614 は、データ・チャンクに対して任意の復元を実行してよい。好ましい方法では、デチャンカ 614 は、すべてのデータ・チャンクをマージされたデータ・セット 710 にマージし、マージされたデータ・セット 710 をクライアントの秘密鍵デクリプタ 642 に送信する。当業者が本開示を読むときに理解するであろうように、クライアントの秘密鍵デクリプタ 642 は、マージされたデータ・セット 710 を復号してよい。クライアントの秘密鍵デクリプタ 642 の出力 712 は、最初の読み取り要求 702 に関連付けられたいずれかのデータ・チャンクを含む。

【0150】

ここで図 8 を参照すると、1 つの構成に従って、方法 800 のフローチャートが示されている。方法 800 は、さまざまな構成において、特に図 1 ～図 7 および図 9 に示された環境のいずれかにおいて、本発明に従って実行されてよい。当然、当業者が本説明を読んだときに理解するであろうように、図 8 において具体的に説明された動作よりも多いか、または少ない動作が方法 800 に含まれてよい。

【0151】

方法 800 のステップの各々は、動作環境の任意の適切なコンポーネントによって実行されてよい。例えば、さまざまな構成では、方法 800 は、コンピュータ、または 1 つまたは複数のプロセッサを含んでいるその他のデバイスによって、部分的に、または全体的に実行されてよい。ハードウェアまたはソフトウェアあるいはその両方において実装され、好ましくは少なくとも 1 つのハードウェア・コンポーネントを含んでいるプロセッサ（例えば、処理回路、チップ、またはモジュール、あるいはその組合せ）が、方法 800 の 1 つまたは複数のステップを実行するために任意のデバイス内で利用されてよい。例示的なプロセッサは、中央処理装置（CPU：central processing unit）、特定用途向け集積回路（ASIC：application specific integrated circuit）、フィールド・プログラマブル・ゲート・アレイ（FPGA：field programmable gate array）など、これらの組合せ、または従来技術において知られた任意のその他の適切なコンピューティング・デバイスを含むが、これらに限定されない。

【0152】

図 8 に示されているように、方法 800 は、重複排除情報をストレージ・システムで受信することを含んでいる動作 802 を含み、重複排除情報は、クライアント・データの一部に関連付けられたシグネチャを含む。好ましい方法では、重複排除情報が、共有一時鍵を使用して暗号化される。1 つの方法では、ストレージ・システムは、共有一時鍵を使用して暗号化された重複排除情報を重複排除マネージャで受信する。1 つの構成では、重複排除マネージャは、一時鍵デクリプタを含む。当業者が本開示を読むときに理解するであろうように、一時鍵デクリプタは、ストレージの一時暗号鍵を使用して暗号化されたデータを復号してよい。好ましい構成では、重複排除情報は、ストレージ・システムによってアクセス可能である（例えば、ストレージ・システムによって暗号化されていない状態で受信される、ストレージ・システムが情報に対して動作できる方法で受信される、などである）。

【0153】

当業者が本開示を読むときに理解するであろうように、重複排除情報は、重複排除方法の動作の詳細を含んでよい。1 つの構成では、チャンキングの重複排除の場合、重複排除情報は、チャンク開始位置、チャンク長、チャンク・シグネチャ、類似性情報などを含んでよい。1 つの方法では、チャンク・シグネチャは、従来技術において知られた任意の方法で計算されてよい暗号ハッシュである。重複排除情報は、ストレージの一時鍵（例えば

、共有一時鍵)を使用して暗号化されるのが好ましい。

【0154】

動作804は、クライアントの秘密鍵を使用して暗号化されたクライアント・データを受信することを含み、クライアントの秘密鍵は、ストレージ・システムには使用することができない。クライアント・データは、ストレージ・システムにとって不透明である(例えば、クライアントの秘密鍵は、ストレージ・システムには使用することができない)。ストレージ・システムには使用することができない鍵は、ストレージ・システムが復号できないデータを暗号化する鍵である。

【0155】

動作806は、クライアント・データのチャンクに対して、ストレージ・システムに格納されたデータ・チャンクを重複排除することを含み、重複排除情報を使用して、重複排除のためにクライアント・データのチャンクがクライアント・データから選択される。1つの方法では、重複排除マネージャは、重複排除情報に基づいて、重複するデータを位置特定するか、または識別するか、あるいはその両方を実行する。チャンキングの重複排除の場合、重複しているデータ・チャンクを識別するために、重複排除情報がチャンク・メタデータと共に使用される。例えば、重複排除マネージャは、復号された重複排除情報からのポインタ、シグネチャ、チャンク長などを、ストレージ・システム上のチャンク・メタデータ・リポジトリに格納された任意のポインタ、シグネチャ、チャンク長などと比較する。当業者が本開示を読むときに理解するであろうように、重複排除マネージャは、任意の関連する重複排除動作を実行する。好ましい方法では、ストレージ・システムはデータ・チャンクのメタデータを格納し、メタデータは、データ・チャンクを暗号化するために使用された鍵を一意に識別する鍵識別子を含む。

【0156】

関連する重複排除動作を実行することに応答して、チャンク・メタデータ・リポジトリに対して更新が実行されてよい。更新は、任意の重複排除されたチャンクの使用、残りのすべてのチャンクに関連する情報の追加などに起因する変更を反映する。チャンク・メタデータ・リポジトリは、特定のデータ・チャンクを暗号化する鍵IDに関する情報を含む。重複排除されていない新しいシグネチャ(例えば、チャンク・メタデータ・リポジトリ内にまだ存在していないシグネチャ)に関連付けられた残りのすべてのデータ・チャンクが、ストレージ・システムに書き込まれる。任意の新しい重複排除されていないデータ・チャンクが、ストレージ・システムに書き込まれる。

【0157】

好ましい構成では、重複排除マネージャが、クライアントが復号することができるデータ・チャンクに対して、クライアントのために重複排除する。クライアント側が復号することができるデータ・チャンクは、クライアントに属している鍵でのデータ・チャンク(例えば、クライアントの秘密鍵を使用して暗号化されたデータ・チャンク)、ストレージ・システムに属している鍵(例えば、ストレージの秘密暗号鍵)、暗号化されていないデータなどを含む。クライアントは、クライアントがアクセスすることができない一意のクライアントの秘密暗号鍵を使用して暗号化されたデータを、復号することができなくてよい。

【0158】

メタデータの更新に関連付けられた残りのすべての重複排除されていないデータ・チャンクは、ストレージの秘密鍵エンクリプタで、ストレージの秘密暗号鍵を使用して暗号化されてよい。メタデータの更新に関連付けられた重複排除されていないデータ・チャンクとは、チャンク・メタデータ・リポジトリ内にすでに存在するデータ・チャンクのことを指してよく、メタデータの更新は、対応する格納位置への追加のポインタを含む。さまざまな方法では、重複排除されていない任意のクライアント・データのチャンクが、クライアントの秘密鍵を使用して暗号化されてストレージ・システムに格納される。

【0159】

当業者が本開示を読むときに理解するであろうように、好ましい方法では、ストレージ

10

20

30

40

50

・システムは、保管時の暗号化データ、暗号化されていないデータ、共有鍵によって暗号化されたデータなどのようなデータを復号することができるため、そのようなデータにアクセスしてよい。

【0160】

1つの方法では、ストレージ・システムは、共有一時鍵を使用して暗号化されたデータ・チャンクを含む。共有一時鍵は、ストレージ・システムによって使用可能である。ストレージ・システムは、クライアントの秘密鍵を使用して暗号化されたデータ・チャンクに対して、および共有一時鍵を使用して暗号化されたデータ・チャンクに対して、共有一時鍵を使用して暗号化されたデータ・チャンクに対する重複排除動作を実行してよい。

【0161】

1つの方法では、ストレージ・システムは、第2のクライアントの秘密鍵を使用して暗号化されたデータを受信してよい。ストレージ・システムは、前述したバックポータリング動作に従って、複数の鍵を使用して暗号化されたデータ・チャンクに対して、第2のクライアントの秘密鍵を使用して暗号化されたデータ・チャンクを重複排除してよい。

【0162】

ここで図9を参照すると、1つの構成に従って、方法900のフローチャートが示されている。方法900は、さまざまな構成において、特に図1～図8に示された環境のいずれかにおいて、本発明に従って実行されてよい。当然、当業者が本説明を読んだときに理解するであろうように、図9において具体的に説明された動作よりも多いか、または少ない動作が方法900に含まれてよい。

【0163】

方法900のステップの各々は、動作環境の任意の適切なコンポーネントによって実行されてよい。例えば、さまざまな構成では、方法900は、コンピュータ、または1つまたは複数のプロセッサを含んでいるその他のデバイスによって、部分的に、または全体的に実行されてよい。ハードウェアまたはソフトウェアあるいはその両方において実装され、好ましくは少なくとも1つのハードウェア・コンポーネントを含んでいるプロセッサ（例えば、処理回路、チップ、またはモジュール、あるいはその組合せ）が、方法900の1つまたは複数のステップを実行するために任意のデバイス内で利用されてよい。プロセッサの例は、中央処理装置（CPU）、特定用途向け集積回路（ASIC）、フィールド・プログラマブル・ゲート・アレイ（FPGA）など、これらの組合せ、または従来技術において知られた任意のその他の適切なコンピューティング・デバイスを含むが、これらに限定されない。

【0164】

図9に示されているように、方法900は、重複排除情報を生成することを含んでいる動作902を含み、重複排除情報は、クライアント・データの一部に関連付けられたシグネチャを含む。好ましい方法では、メタ生成器は、データに関連付けられた重複排除メタデータを計算する。さまざまな構成では、メタ生成器は、クライアント側またはストレージ・システムあるいはその両方に配置されてよい。

【0165】

好ましい方法では、メタ生成器は、クライアント側でデータに対してメタデータ動作を実行し、データ内のデータ重複排除の機会を識別する。メタ生成器は、従来技術において知られた任意のデータ重複排除技術を使用してデータ重複排除の機会を識別する。1つの態様では、メタ生成器は、スライド・ウィンドウ可変チャンク重複排除技術を使用してデータ重複排除の機会を識別する。メタ生成器は、重複排除シグネチャ（例えば、ローリング最小ハッシュ、暗号ハッシュなど）を識別するか、または計算するか、あるいはその両方を実行する。メタ生成器は、ポインタ、チャンク長、チャンク・シグネチャなどを含む、データ重複排除の機会に関連付けられたメタデータ（例えば、重複排除情報）を生成する。重複排除情報は、共有一時鍵に対するクライアント側からの要求に応答してストレージ・システムによって提供された共有一時鍵を使用して、暗号化されてよい。

【0166】

当業者が本開示を読むときに理解するであろうように、重複排除情報は、重複排除方法の動作の詳細を含んでよい。1つの構成では、チャンキングの重複排除の場合、重複排除情報は、チャンク開始位置、チャンク長、チャンク・シグネチャ、類似性情報などを含んでよい。1つの方法では、チャンク・シグネチャは、従来技術において知られた任意の方法で計算されてよい暗号ハッシュである。

【0167】

動作904は、共有一時鍵を使用して重複排除情報を暗号化することを含む。クライアント側は、一時鍵エンクリプタを備えてよい。当業者が本開示を読むときに理解するであろうように、一時鍵エンクリプタは、ストレージの一時鍵を使用してデータを暗号化してよい。

【0168】

動作906は、クライアントの秘密鍵を使用してクライアント・データを暗号化することを含む。クライアント側は、クライアントの秘密鍵エンクリプタを備えてよい。当業者が本開示を読むときに理解するであろうように、クライアントの秘密鍵エンクリプタは、クライアントの秘密鍵を使用してデータを暗号化してよい。好ましい方法では、クライアントの秘密鍵は、クライアント側から離れることがない。この好ましい方法では、ストレージ・システムは、データ転送のどの時点でも、暗号化データを暗号化されていない状態で見ることがない。この好ましい方法では、クライアントの秘密鍵が、ストレージ・システム、他の任意のクライアント、他の任意のストレージ・システムなどと共有されない。クライアントの秘密鍵は、対称鍵、非対称鍵などであってよい。1つの態様では、クライアントの秘密鍵は対称鍵である。

【0169】

一部の方法では、チャンキングの前または後あるいはその両方で、さまざまな圧縮技術がクライアント・データに適用されてよい。1つの構成では、チャンキング前の圧縮は、チャンキングの性能を改善する種類の圧縮であってよい。別の構成では、チャンキング後の圧縮は、結果として得られるチャンク・サイズを最小化することに向けて調整されてよい。

【0170】

シグネチャの計算は、圧縮の前または後あるいはその両方で、データ・チャンクに対して実行されてよい。圧縮前のシグネチャの計算は、読み取り時の復元後にシグネチャを検証できるようにする。圧縮後のシグネチャの計算は、チャンクが圧縮された状態にある間に、シグネチャをチェックできるようにする。

【0171】

さまざまな方法では、重複排除情報およびクライアント・データが、関連付けられているとしてタグ付けされるか、またはその他の方法で示されるか、あるいはその両方である。好ましい方法では、重複排除情報は、暗号化されていないデータに対して計算される。暗号化されていないデータは、チャンク・シグネチャを含んでよい。重複排除情報は、暗号化データに対して計算されてよい。暗号化データに対して計算された重複排除情報は、暗号化されたチャンクのシグネチャを含んでよい。

【0172】

動作908は、重複排除情報およびクライアント・データを重複排除のためにストレージ・システムに送信することを含む。重複排除情報およびクライアント・データは、従来技術において知られた任意の方法で、重複排除のためにストレージ・システムに送信されてよい。

【0173】

動作910は、ストレージ・システムに対してクライアント・データを要求することを含む。クライアント・データは、従来技術において知られた任意の方法で要求されてよい。

【0174】

1つの方法では、ストレージ・システムに対するクライアント・データの要求は、署名されたアクセス制御を使用して要求を検証することを含む。一部の方法では、非対称鍵暗

10

20

30

40

50

号化の署名が、対称鍵暗号化のデータ転送のためのセキュアなアクセス制御として使用されてよい。クライアントは、クライアントの秘密暗号鍵（例えば、クライアントの秘密 P K I 鍵）を使用してすべての I / O 要求に署名してよい。例えば、クライアント・データの対称鍵暗号化の場合、クライアントの秘密 P K I 鍵が本人確認に使用されてよい。ストレージ・システムは、公開鍵を使用してシグネチャを検証してよい。同様に、ストレージ・システムは、クライアントによって公開鍵を使用して検証されることができるストレージの秘密暗号鍵を使用して、I / O 応答に署名してよい。

【0175】

動作 912 は、クライアント・データをデータ・チャンクとして受信することを含み、データ・チャンクの少なくとも一部が、クライアントの秘密鍵を使用して暗号化され、データ・チャンクの少なくとも一部が、共有一時鍵を使用して暗号化される。データ・チャンクとしてのクライアント・データは、従来技術において知られた任意の方法で受信されてよい。

【0176】

1つの方法では、クライアント側はデチャンカを備える。デチャンカは、ストレージ・システムから受信されたデータ・チャンクをチャンク解除し、データ・チャンクは、複数の暗号鍵を使用して暗号化される。複数の暗号鍵を使用して暗号化されたデータ・チャンクとは、1つまたは複数のデータ・チャンクが第1の鍵を使用して暗号化され、1つまたは複数のデータ・チャンクが第2の鍵を使用して暗号化される、データ・チャンクのセットのことを指すのが好ましい。デチャンカは、どの暗号鍵を使用してデータ・チャンクが暗号化されているかを識別し、そのデータ・チャンクを対応するデクリプタ（例えば、クライアントの秘密鍵デクリプタ、共有一時鍵デクリプタなど）に転送してよい。当業者が本開示を読むときに理解するであろうように、各デクリプタは、対応する鍵を使用して暗号化されたデータ・チャンクを復号してよい。

【0177】

本明細書で開示されたさまざまな構成では、クライアントの秘密暗号鍵がクライアント側から離れることなく、さまざまな圧縮技術または重複排除技術あるいはその両方が実行される。好ましい構成では、ストレージ・システムは、クライアントの秘密暗号鍵にアクセスすることができず、サード・パーティは、クライアントの秘密暗号鍵を見ることができない。

【0178】

他の方法では、対称暗号化と非対称暗号化のハイブリッドが使用されてよい。例えば、ストレージ・システムは、ストレージの鍵でチャンクを格納するために、対称鍵暗号化を使用してよく、非対称鍵がクライアント・データに使用されてよい。

【0179】

1つの方法では、クライアントまたはストレージ・システムあるいはその両方が、2つ以上の鍵を含んでよい。本明細書で開示されたさまざまな構成は、ブロック・ストレージ・システム、オブジェクト・ストレージ・システム、ファイル・ストレージ・システムなどに適用されてよい。暗号化されていないデータが、ストレージ・システムによって使用できる鍵を使用して暗号化されたデータのこと（例えば、ストレージ・システムが、暗号化されていない状態でデータを見ることができること）を指すことということが、本開示を読むときに当業者によって理解されるはずである。

【0180】

一部の方法では、メタ生成器によって作成されるメタデータ（例えば、重複排除情報）は、重複排除のためのメタデータよりも、相対的にさらに一般的であってよい。例えば、テキスト・データのキーワード・インデックスを作成するために、重複排除情報が使用されてよい。キーワード・インデックスは、ストレージ・システムによって、復号される任意の発見された情報をクライアントに返すことを含む検索動作を暗号化データに対して実行するために、使用されてよい。別の例では、重複排除情報は、ビデオ情報用のイベント検出器を作成するために使用されてよい。イベント検出器は、検出されたイベントに関連

10

20

30

40

50

するポインタまたは情報あるいはその両方を作成してよい。イベント検出器は、検出されたイベントに関する追加の分類を提供してよい。ストレージ・システムは、このメタデータに対して検索動作を実行し、特定の基準を満たすデータの部分を識別して、I/Oワークロードを減らしてよい。さまざまな他の分類、タグ付けの位置決めなどの技術が、重複排除機能を含むか、または含まない、これらの動作から恩恵を受けることができる。

【0181】

本発明は、任意の可能な統合の技術的詳細レベルで、システム、方法、またはコンピュータ・プログラム製品、あるいはその組合せであってよい。コンピュータ・プログラム製品は、プロセッサに本発明の態様を実行させるためのコンピュータ可読プログラム命令を含んでいるコンピュータ可読ストレージ媒体を含んでよい。

【0182】

コンピュータ可読記憶媒体は、命令実行デバイスによって使用するための命令を保持および格納できる有形のデバイスであることができる。コンピュータ可読ストレージ媒体は、例えば、電子ストレージ・デバイス、磁気ストレージ・デバイス、光ストレージ・デバイス、電磁ストレージ・デバイス、半導体ストレージ・デバイス、またはこれらの任意の適切な組合せであってよいが、これらに限定されない。コンピュータ可読ストレージ媒体のさらに具体的な例の非網羅的リストは、ポータブル・コンピュータ・ディスク、ハード・ディスク、ランダム・アクセス・メモリ（RAM：random access memory）、読み取り専用メモリ（ROM：read-only memory）、消去可能プログラマブル読み取り専用メモリ（EPROM：erasable programmable read-only memoryまたはフラッシュ・メモリ）、スタティック・ランダム・アクセス・メモリ（SRAM：static random access memory）、ポータブル・コンパクト・ディスク読み取り専用メモリ（CD-ROM：compact disc read-only memory）、デジタル・バーサタイル・ディスク（DVD：digital versatile disk）、メモリ・スティック、フロッピー・ディスク、パンチカードまたは命令が記録されている溝の中の隆起構造などの機械的にエンコードされるデバイス、および前述の任意の適切な組合せを含む。本明細書において使用されるとき、コンピュータ可読ストレージ媒体は、それ自体が、電波またはその他の自由に伝搬する電磁波、導波管またはその他の送信媒体を伝搬する電磁波（例えば、光ファイバ・ケーブルを通過する光パルス）、あるいはワイヤを介して送信される電気信号などの一過性の信号であると解釈されるべきではない。

【0183】

本明細書に記載されたコンピュータ可読プログラム命令は、コンピュータ可読ストレージ媒体から各コンピューティング・デバイス／処理デバイスへ、またはネットワーク（例えば、インターネット、ローカル・エリア・ネットワーク、広域ネットワーク、または無線ネットワーク、あるいはその組合せ）を介して外部コンピュータまたは外部ストレージ・デバイスへダウンロードされ得る。このネットワークは、銅伝送ケーブル、光伝送ファイバ、無線送信、ルータ、ファイアウォール、スイッチ、ゲートウェイ・コンピュータ、またはエッジ・サーバ、あるいはその組合せを備えてよい。各コンピューティング・デバイス／処理デバイス内のネットワーク・アダプタ・カードまたはネットワーク・インターフェイスは、コンピュータ可読プログラム命令をネットワークから受信し、それらのコンピュータ可読プログラム命令を各コンピューティング・デバイス／処理デバイス内のコンピュータ可読記憶媒体に格納するために転送する。

【0184】

本発明の動作を実行するためのコンピュータ可読プログラム命令は、アセンブラ命令、命令セット・アーキテクチャ（ISA：instruction-set-architecture）命令、マシン命令、マシン依存命令、マイクロコード、ファームウェア命令、状態設定データ、集積回路のための構成データ、あるいは、Smalltalk（R）、C++などのオブジェクト指向プログラミング言語、および「C」プログラミング言語または同様のプログラミング言語などの手続き型プログラミング言語を含む1つまたは複数のプログラミング言語の任意の組合せで記述されたソース・コードまたはオブジェクト・コードであってよい。コン

コンピュータ可読プログラム命令は、ユーザのコンピュータ上で全体的に実行すること、ユーザのコンピュータ上でスタンドアロン・ソフトウェア・パッケージとして部分的に実行すること、ユーザのコンピュータ上およびリモート・コンピュータ上でそれぞれ部分的に実行すること、あるいはリモート・コンピュータ上またはサーバ上で全体的に実行することができる。後者のシナリオでは、リモート・コンピュータは、ローカル・エリア・ネットワーク（LAN：local area network）または広域ネットワーク（WAN：wide area network）を含む任意の種類のネットワークを介してユーザのコンピュータに接続されてよく、または接続は、（例えば、インターネット・サービス・プロバイダを使用してインターネットを介して）外部コンピュータに対して行われてよい。一部の実施形態では、本発明の態様を実行するために、例えばプログラマブル・ロジック回路、フィールドプログラマブル・ゲート・アレイ（FPGA：field-programmable gate arrays）、またはプログラマブル・ロジック・アレイ（PLA：programmable logic arrays）を含む電子回路は、コンピュータ可読プログラム命令の状態情報を利用することによって、電子回路をカスタマイズするためのコンピュータ可読プログラム命令を実行してよい。

【0185】

本発明の態様は、本明細書において、本発明の実施形態に従って、方法、装置（システム）、およびコンピュータ・プログラム製品のフローチャート図またはブロック図あるいはその両方を参照して説明される。フローチャート図またはブロック図あるいはその両方の各ブロック、ならびにフローチャート図またはブロック図あるいはその両方に含まれるブロックの組合せが、コンピュータ可読プログラム命令によって実装され得るということが理解されるであろう。

【0186】

これらのコンピュータ可読プログラム命令は、コンピュータまたはその他のプログラム可能なデータ処理装置のプロセッサを介して実行される命令が、フローチャートまたはブロック図あるいはその両方のブロックに指定される機能／動作を実施する手段を作り出すべく、コンピュータ、または他のプログラム可能なデータ処理装置のプロセッサに提供されてマシンを作り出すものであってよい。これらのコンピュータ可読プログラム命令は、命令が格納されたコンピュータ可読ストレージ媒体がフローチャートまたはブロック図あるいはその両方のブロックに指定される機能／動作の態様を実施する命令を含んでいる製品を含むように、コンピュータ可読ストレージ媒体に格納され、コンピュータ、プログラム可能なデータ処理装置、または他のデバイス、あるいはその組合せに特定の方式で機能するように指示できるものであってもよい。

【0187】

コンピュータ可読プログラム命令は、コンピュータ上、その他のプログラム可能な装置上、またはその他のデバイス上で実行される命令が、フローチャートまたはブロック図あるいはその両方のブロックに指定される機能／動作を実施するように、コンピュータ、その他のプログラム可能なデータ処理装置、またはその他のデバイスに読み込まれてもよく、それによって、一連の動作可能なステップを、コンピュータ上、その他のプログラム可能な装置上、またはコンピュータ実装プロセスを生成するその他のデバイス上で実行させる。

【0188】

図内のフローチャートおよびブロック図は、本発明のさまざまな実施形態に従って、システム、方法、およびコンピュータ・プログラム製品の可能な実装のアーキテクチャ、機能、および動作を示す。これに関連して、フローチャートまたはブロック図内の各ブロックは、規定された論理機能を実装するための1つまたは複数の実行可能な命令を備える、命令のモジュール、セグメント、または部分を表してよい。一部の代替の実装では、ブロックに示された機能は、図に示された順序とは異なる順序で発生してよい。例えば、連続して示された2つのブロックは、実際には、含まれている機能に応じて、1つのステップとして実行されるか、同時に実行されるか、時間的に部分的または完全に重複する方法で実質的に同時に実行されるか、または場合によっては逆の順序で実行されてよい。ブロッ

ク図またはフローチャート図あるいはその両方の各ブロック、ならびにブロック図またはフローチャート図あるいはその両方に含まれるブロックの組合せは、規定された機能または動作を実行するか、または専用ハードウェアとコンピュータ命令の組合せを実行する専用ハードウェアベースのシステムによって実装され得るということにも注意する。

【0189】

さらに、さまざまな実施形態に従うシステムは、プロセッサ、およびプロセッサに統合された論理またはプロセッサによって実行可能な論理あるいはその両方を含んでよく、この論理は、本明細書において列挙された処理ステップのうちの1つまたは複数を実行するように構成される。統合されるということは、つまり、プロセッサに論理が、特定用途向け集積回路（ASIC：application specific integrated circuit）、FPGAなどのハードウェア論理として組み込まれるということである。プロセッサによって実行可能であるということは、つまり、論理が、プロセッサによってアクセス可能なハードウェア論理、ソフトウェア論理（ファームウェア、オペレーティング・システムの一部、アプリケーション・プログラムの一部など）など、またはハードウェア論理とソフトウェア論理の何らかの組合せであり、プロセッサによって実行されたときになんらかの機能をプロセッサに実行させるように構成されているということである。ソフトウェア論理は、従来技術において知られた任意のメモリ・タイプの、ローカルまたはリモートあるいはその両方のメモリに格納されてよい。ASIC、FPGA、中央処理装置（CPU：central processing unit）、集積回路（IC：integrated circuit）、グラフィック処理装置（GPU：graphics processing unit）といった、ソフトウェア・プロセッサ・モジュールまたはハードウェア・プロセッサあるいはその両方などの、従来技術において知られた任意のプロセッサが使用されてよい。

【0190】

上で提示された説明から複数の組合せを作成して、前述のシステムまたは方法あるいはその両方のさまざまな特徴が任意の方法で組み合わせられてよいということが明らかである。

【0191】

本発明の実施形態が、サービスをオンデマンドで提供するように顧客のために展開されるサービスの形態で提供されてよいということがさらに理解されるであろう。

【0192】

本発明のさまざまな実施形態の説明は、例示の目的で提示されているが、網羅的であることは意図されておらず、開示された実施形態に限定されない。説明された実施形態の範囲を逸脱することなく多くの変更および変形が可能であることは、当業者にとって明らかである。本明細書で使用された用語は、実施形態の原理、実際の適用、または市場で見られる技術を超える技術的改良を最も適切に説明するため、または他の当業者が本明細書で開示された実施形態を理解できるようにするために選択されている。

10

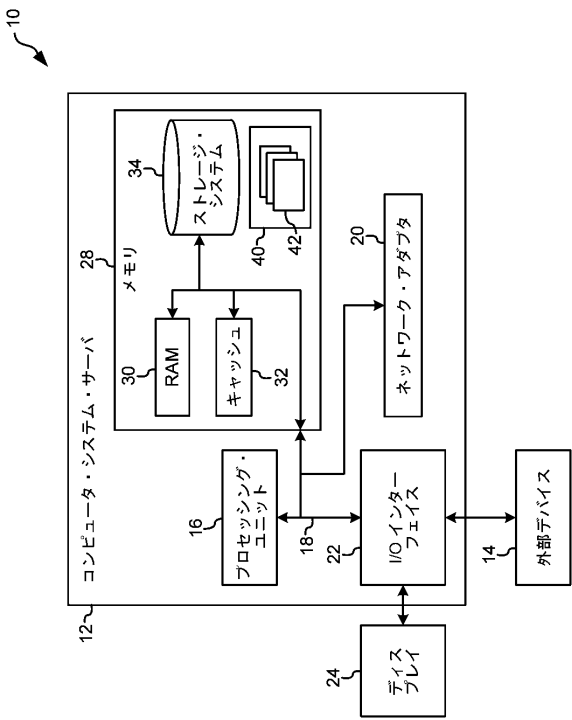
20

30

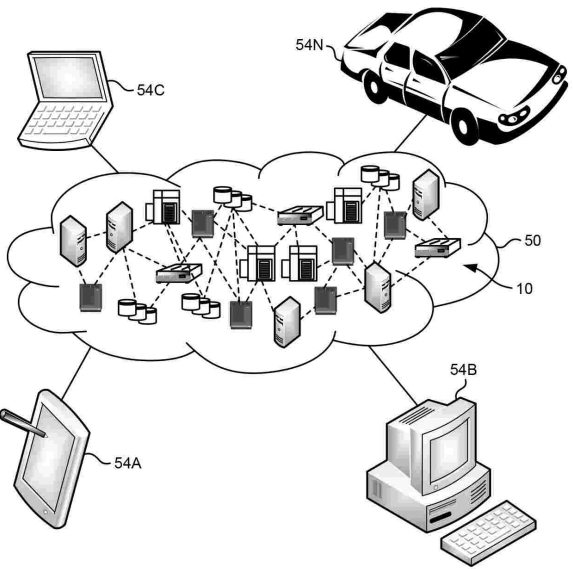
40

50

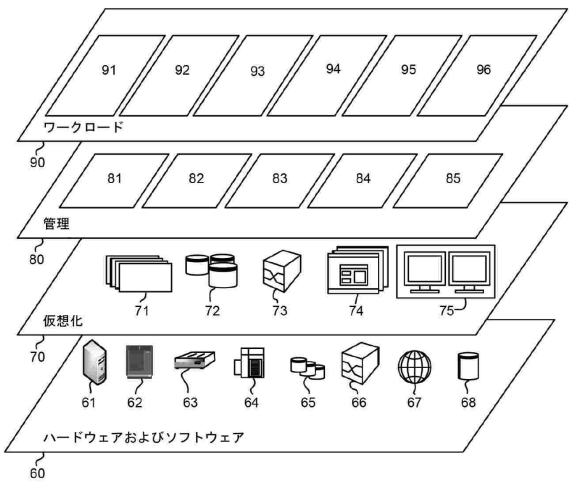
【図面】
【図 1】



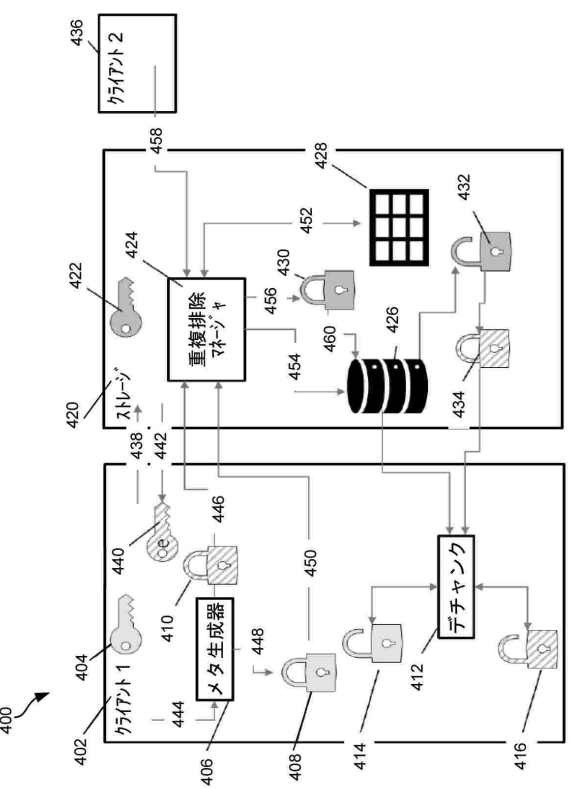
【図 2】



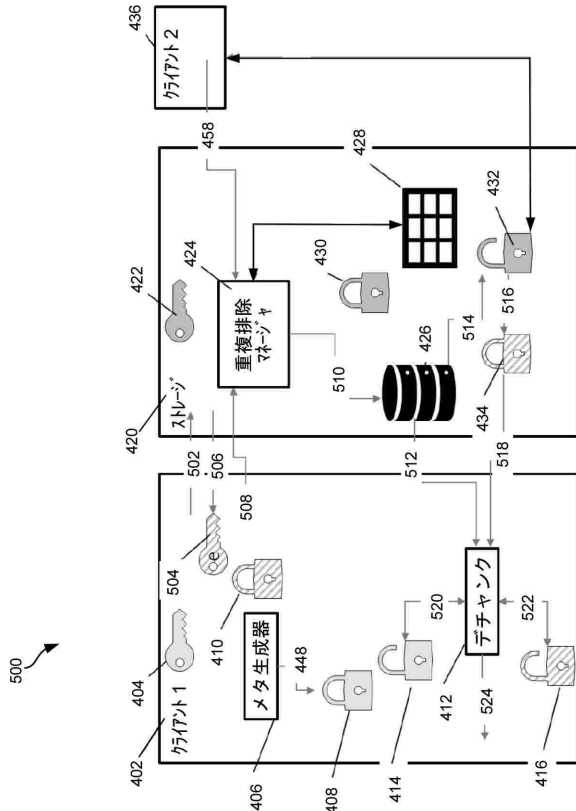
【図 3】



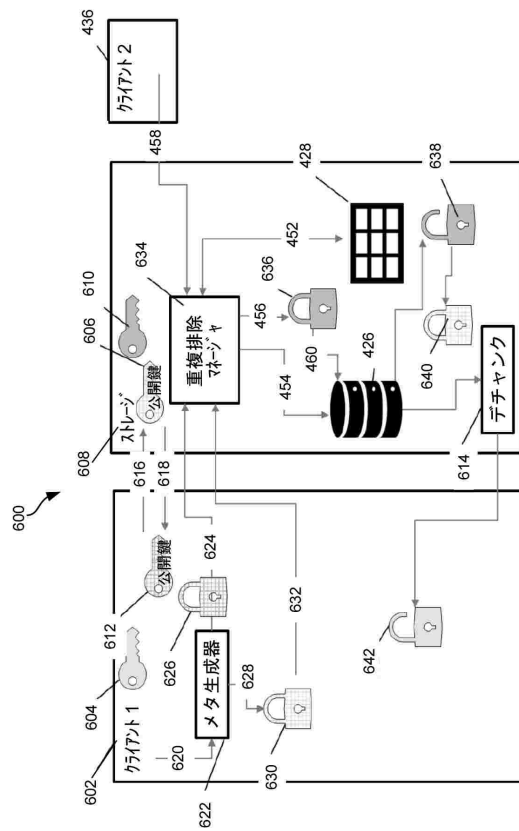
【図 4】



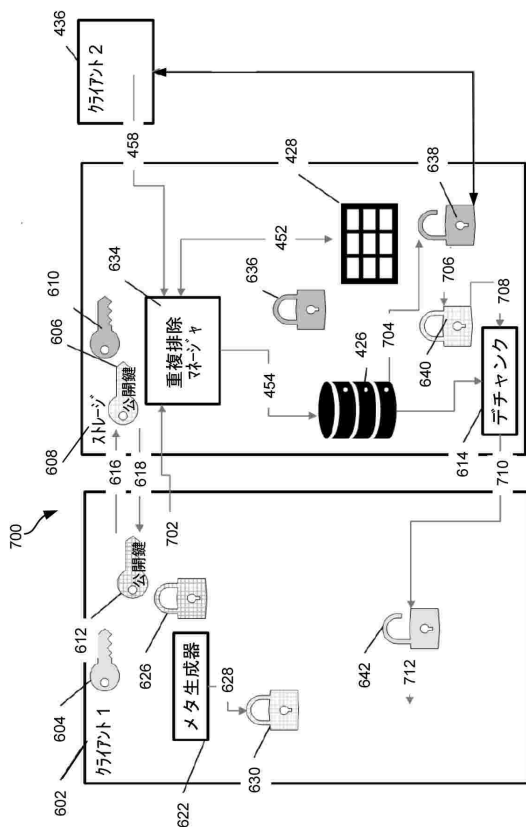
【図 5】



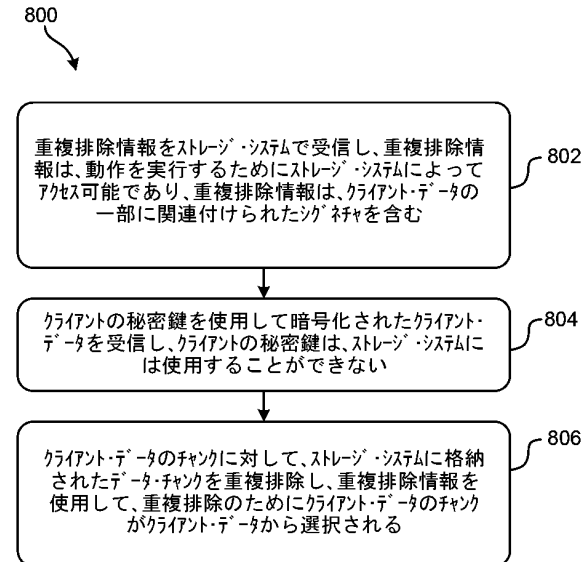
【図 6】



【図 7】



【図 8】



10

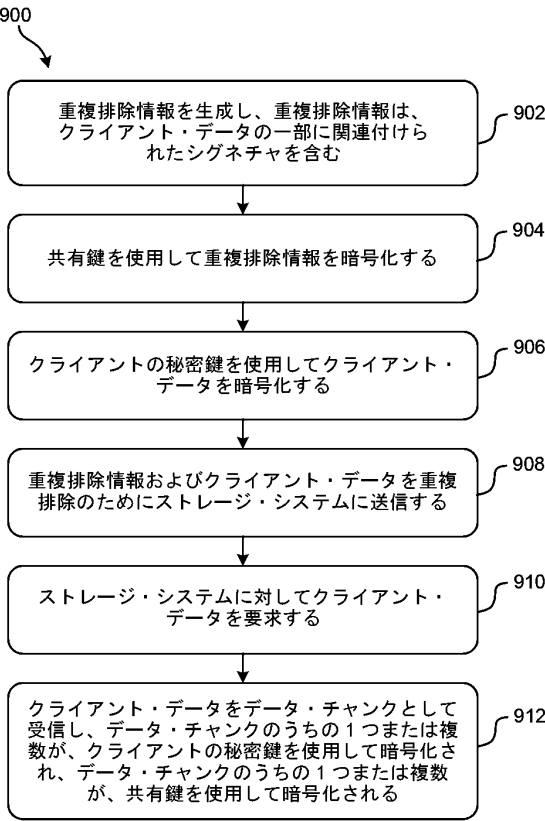
20

30

40

50

【図 9】



10

20

30

40

50

フロントページの続き

- (51)国際特許分類

G 0 6 F

16/28 (2019.01)

F I

G 0 6 F21/623 1 8

G 0 6 F21/603 2 0

G 0 6 F16/28
- (72)発明者

ヘツラー、スティーブン、ロバート

アメリカ合衆国 9 5 1 2 0 - 6 0 9 9 カリフォルニア州サンノゼ アルマデン・リサーチ・センタ
ー ハリー・ロード 6 5 0
- 審査官

早川 学
- (56)参考文献

米国特許出願公開第 2 0 1 5 / 0 2 2 7 7 5 7 (U S , A 1)

特表 2 0 1 2 - 5 3 3 1 2 6 (J P , A)

米国特許出願公開第 2 0 1 6 / 0 0 6 5 5 4 0 (U S , A 1)

米国特許出願公開第 2 0 1 4 / 0 0 2 5 9 4 8 (U S , A 1)
- (58)調査した分野

(Int.Cl., D B 名)

G 0 6 F16 / 0 0 - 1 6 / 9 5 8

G 0 9 C1 / 0 0

G 0 6 F21 / 1 0 - 2 1 / 8 8