(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4792609号
(P4792609)
(45)発行日　平成23年10月12日(2011.10.12)　(24)登録日　平成23年8月5日(2011.8.5)

(51)Int.Cl.　F I
G06F 21/24 (2006.01)　G06F 12/14　560C

請求項の数 20　外国語出願　(全 11 頁)

(21)出願番号　特願2007-294389(P2007-294389)
(22)出願日　平成19年11月13日(2007.11.13)
(65)公開番号　特開2008-140384(P2008-140384A)
(43)公開日　平成20年6月19日(2008.6.19)
審査請求日　平成20年1月9日(2008.1.9)
(31)優先権主張番号　11/598,409
(32)優先日　平成18年11月13日(2006.11.13)
(33)優先権主張国　米国(US)

(73)特許権者　500373758
シーゲイト　テクノロジー　エルエルシー
アメリカ合衆国、９５０１４　カリフォルニア州、クパチーノ、サウス・デ・アンザ・ブールバード、１０２００
(74)代理人　100064746
弁理士　深見　久郎
(74)代理人　100085132
弁理士　森田　俊雄
(74)代理人　100083703
弁理士　仲村　義平
(74)代理人　100096781
弁理士　堀井　豊
(74)代理人　100109162
弁理士　酒井　將行

最終頁に続く

(54)【発明の名称】認証済みデータ・ストレージのための方法および装置

(57)【特許請求の範囲】
【請求項１】
データ記憶装置に記憶されたデータの認証のための方法であって、前記データ記憶装置はデータ・ストレージ媒体とリードおよびライト処理を制御するためのコントローラとを含み、
前記コントローラが、前記データ・ストレージ媒体の複数のデータ・ブロック・セットそれぞれについて、第１のメッセージ認証コードを計算するステップを含み、各前記データ・ブロック・セットは、前記データ記憶装置の物理的特徴に基づいて選択され、別途のシーク動作なしにアクセス可能なデータ・ブロックをグループ化するものであり、
前記コントローラが、前記第１のメッセージ認証コードを記憶するステップと、
前記コントローラが、認証されることになる前記データ・ブロック・セットのそれぞれについて第２のメッセージ認証コードを計算すること、および前記記憶された第１のメッセージ認証コードと前記第２のメッセージ認証コードとを比較することによって、前記データ・ブロック・セットを認証するステップと、
を含む方法。
【請求項２】
前記データ・ストレージ媒体は、表面に複数のトラックを有するディスク記録媒体を含み、
前記データ・ブロック・セットのそれぞれは、前記ディスク・ストレージ媒体の１トラック内のデータを含む、請求項１記載の方法。

【請求項３】
前記第１と第２のメッセージ認証コードが異なる場合、前記データ・ブロック・セットが変更されているものとみなされる、請求項１記載の方法。
【請求項４】
前記第１のメッセージ認証コードを記憶するステップは、前記第１のメッセージ認証コードを不揮発性メモリに格納するステップを含む、請求項１記載の方法。
【請求項５】
データ・ブロック・セットに対応する比較結果を示す１つ以上のポインタのテーブルを記憶するステップと、
前記第１および第２のメッセージ認証コードを比較する代わりに、前記テーブルを参照することで当該データ・ブロック・セットが変更されたか否かを判断するステップとをさらに備える、請求項１記載の方法。
【請求項６】
前記第１のメッセージ認証コードのグループに対して追加のメッセージ認証コードを計算するステップと、
前記第１のメッセージ認証コードおよび前記追加のメッセージ認証コードをツリー構造に編成するステップをさらに含む、請求項５記載の方法。
【請求項７】
前記比較結果は所定の期間内に生成される、請求項５記載の方法。
【請求項８】
前記所定の期間は、前記データ記憶装置のパワー・サイクルを含み、前記テーブルの前記ポインタは、前記データ記憶装置のパワー・サイクルに基づいて消去される、請求項７記載の方法。
【請求項９】
認証障害が発生した場合、ユーザに警告を発するステップをさらに含む、請求項１記載の方法。
【請求項１０】
前記第１のメッセージ認証コードが、鍵、入力データ、および前記データ・ブロック・セットのうちの１つの以前に格納されたデータ・ブロックを使用して計算される、請求項１記載の方法。
【請求項１１】
前記データ記憶装置がアイドル状態である間に、生成されたメッセージ認証コードと記憶されたメッセージ認証コードとの比較に基づいて、１つ以上の前記データ・ブロックが変更されたか否かを判断するステップと、
関連するデータブ・ロックが変更されていないと判断された場合、前記関連するデータに対応するポインタを前記テーブルに記憶させ、
前記テーブルを参照して、リードリクエストに対応するデータ・ブロック・セットが変更されたか否かを判断するステップと、
前記リードリクエストに対応するデータ・ブロック・セットと関連するポインタが前記テーブルに含まれる場合、前記リードリクエストに対応する前記データ・ブロック・セットへのアクセスを実行するステップとをさらに含む、請求項５記載の方法。
【請求項１２】
記憶されたデータを認証するための装置であって、
ストレージ媒体と、
前記ストレージ媒体の複数のデータ・ブロック・セットそれぞれについて、第１のメッセージ認証コードを計算し、前記第１のメッセージ認証コードを記憶させるためのプロセッサとを備え、各前記データ・ブロック・セットは、前記装置の物理的特徴に基づいて選択され、別途のシーク動作なしにアクセス可能なデータ・ブロックをグループ化するものであり、
前記プロセッサは、認証されるべき前記データ・ブロック・セットのそれぞれについて

第２のメッセージ認証コードを計算し、前記記憶された第１のメッセージ認証コードと前記第２のメッセージ認証コードとを比較することによって、前記データ・ブロック・セットを認証する、装置。

【請求項１３】

前記ストレージ媒体は、表面に複数のトラックを有するディスク記録媒体を含み、

前記データ・ブロック・セットのそれぞれが、前記ストレージ媒体の１トラック内のデータを含む、請求項１２記載の装置。

【請求項１４】

前記第１と第２のメッセージ認証コードが異なる場合、前記データ・ブロック・セットが変更されているものとみなされる、請求項１２記載の装置。

【請求項１５】

前記第１のメッセージ認証コードを格納するための不揮発性メモリをさらに備える、請求項１２記載の装置。

【請求項１６】

前記プロセッサは、データ・ブロック・セットに対応する比較結果を示す１つ以上のポインタのテーブルを記憶させ、

前記第１および第２のメッセージ認証コードを比較する代わりに、前記テーブルを参照することで当該データ・ブロック・セットが変更されたか否かを判断する、請求項１２記載の装置。

【請求項１７】

前記プロセッサは、前記第１のメッセージ認証コードのグループに対して追加のメッセージ認証コードを計算し、

前記第１のメッセージ認証コードおよび前記追加のメッセージ認証コードがツリー構造に編成される、請求項１６記載の装置。

【請求項１８】

前記プロセッサは、前記比較結果は所定の期間内に生成する、請求項１６記載の装置。

【請求項１９】

前記所定の期間は、前記データ記憶装置のパワー・サイクルを含み、前記テーブルの前記ポインタは、前記データ記憶装置のパワー・サイクルに基づいて消去される、請求項１８記載の装置。

【請求項２０】

前記第１のメッセージ認証コードが、鍵、入力データ、および前記データ・ブロック・セットのうちの１つの以前に格納されたデータ・ブロックを使用して計算される、請求項１２記載の装置。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明はデータ・ストレージ・デバイスに関し、とりわけ、データ・ストレージ・デバイスに格納されたデータを認証するための方法および装置に関する。

【背景技術】

【０００２】

セキュアなデータ・ストレージ・デバイスは、格納済みデータを暗号化し、データへのアクセスを制御するための手段を含む。デバイスは、適切な許可がなければ、たとえ暗号化されたデータに対するアクセスであっても、ストレージ・デバイス・インターフェースを介したアクセスを認可しない。しかしながら、ストレージ・デバイスの物理的保護は非常に費用がかかり、多くの場合非実用的であるため、ディスク・ドライブを分解すること、および微細な磁気検出器を備えたスピン・スタンド上にプラッタを置くこと、または単にヘッド信号を他のディスク・コントローラに接続することなどの、物理的攻撃が依然として可能な場合がある。したがって、暗号化は必ずしも、無許可者によるデータの修正を保護するものではない。

【０００３】
無許可者がストレージ・デバイスへの書き込みアクセスを得ると、たとえば、古い内容をストレージ媒体内の同じ場所に再書き込みすることによって、または他の場所からデータを上書きコピーすることによって、損害を発生させることができる。これを防ぐ安価な物理的方法がないため、データが改ざんされていることをユーザに警告するために、格納済みデータの無許可の変更を検出するための暗号化方法が必要である。
【０００４】
通常は、ユーザにドライブが開かれたかどうかを通知する、ディスク・ドライブ・エンクロージャのシールなどの、高価で脆弱な物理的改ざん検出器が採用される。
【発明の開示】
【発明が解決しようとする課題】
【０００５】
物理的改ざん検出器に依拠しない改ざん検出方法を提供することが望ましい。
【課題を解決するための手段】
【０００６】
第１の態様では、本発明は、データ・ストレージ媒体上の複数のデータ・ブロック・セットそれぞれについて、第１のメッセージ認証コードを計算するステップと、認証されるべきデータ・ブロック・セットのそれぞれについて第２のメッセージ認証コードを計算すること、および第１と第２のメッセージ認証コードを比較することによって、データ・ブロック・セットを認証するステップと、を含む方法を提供する。
【０００７】
データ・ブロック・セットは、第１と第２のメッセージ認証コードが異なる場合、変更されているとみなされる。それぞれのデータ・ブロック・セットが、ディスク・ストレージ媒体の１トラック内のデータを含むことができる。メッセージ認証コードは、ストレージ媒体上または不揮発性メモリのテーブル内に格納可能であり、ツリー構造に編成することができる。
【０００８】
他の態様では、本発明は、データ・ストレージ媒体上の複数のデータ・ブロック・セットそれぞれについて、第１のメッセージ認証コードを計算するため、ならびに、認証されるべきデータ・ブロック・セットのそれぞれについて第２のメッセージ認証コードを計算すること、および第１と第２のメッセージ認証コードを比較することによって、データ・ブロック・セットを認証するための、プロセッサを備える、装置を提供する。
【実施例】
【０００９】
図面を参照すると、図１は、本発明の実施形態に従って動作可能な、ディスク・ドライブ１０の形のデータ・ストレージ・デバイスを示す絵画図である。ディスク・ドライブは、ディスク・ドライブの様々なコンポーネントを含むようにサイズ設定および構成された、ハウジング１２（この図では、上部が取り除かれ、下部が見える）を含む。ディスク・ドライブは、ハウジング内で少なくとも１つのデータ・ストレージ媒体１６、この場合は磁気ディスクを回転させるための、スピンドル・モータ１４も含む。各ハウジング１２内には少なくとも１つのアーム１８が含まれ、それぞれのアーム１８は、記録および／または読み取りヘッドまたはスライダ２２を備えた第１の端部２０と、ベアリング２６によってシャフトに枢動可能（ｐｉｖｏｔａｌｌｙ）に取り付けられた第２の端部２４とを有する。アクチュエータ・モータ２８は、ディスク１６の所望のトラック上にヘッド２２を位置決めするようにアーム１８を枢動させるために、アームの第２の端部２４に配置される。アクチュエータ・モータ２８は、この図には示されていないコントローラによって制御される。データは、ディスク上の同心トラック３０に格納される。複数のディスクが含まれる場合、複数のディスク上の対応するトラックが円柱を形成する。トラックはセクタ３２に分割され、データ・ブロックがセクタに格納される。
【００１０】

図２は、本発明の実施形態に従って構築可能な装置４０を示すブロック図である。装置は、コンピュータ４２と、たとえばディスク・ドライブ、プローブ・ストレージ・デバイス、メモリ・カード、フラッシュ・ドライブ、または他のストレージ・デバイスとすることが可能な、データ・ストレージ・デバイス４４とを含む。コンピュータ４２は、ホスト・オペレーティング・システムに従って動作し、チャネル４８を介してデータ・ストレージ・デバイスと通信する、プロセッサ４６を含む。コンピュータは、一般にコンピュータ内に見られるがこの図には示されていない、入力および出力デバイスならびにメモリ・コンポーネントなどの、他のコンポーネントも含むことになる。プロセッサ４６は、この説明に記載された様々な読み取り、書き込み、および暗号化の機能を実行するように動作可能である。

【００１１】

データ・ストレージ・デバイスは、磁気または光のストレージ媒体、強誘電性ストレージ媒体、あるいはソリッドステート・メモリなどの、何らかのタイプのストレージ媒体５０と、媒体上でのデータのストレージを制御するコントローラ５２と、を含む。コントローラは、データ・ストレージ・デバイスに含まれ、ストレージ媒体へ書き込まれ、および／またはストレージ媒体から読み取られるデータを処理するために使用される、ファームウェア５４を実行することができる。メモリ５６および暗号モジュール５８を含む追加のコンポーネントを、データの処理を容易にするために含めることができる。コントローラは、ファームウェア５４に応答して様々な機能を実行するプロセッサを含むことができる。図２に示されるように、暗号モジュールはストレージ・デバイス内部に含めることができるが、ホスト・コンピュータのソフトウェア内（たとえば、ホスト・オペレーティング・システム内）で実施するか、または、ホストとデータ・ストレージ・デバイスとの間のデータ・パスに挿入することもできる。

【００１２】

一態様では、本発明は、無許可の人物が知らない秘密鍵に依拠する、改ざん検出の暗号化方法（ｃｒｙｐｔｏｇｒａｐｈｉｃ ｍｅｔｈｏｄ ｏｆ ｔａｍｐｅｒ ｄｅｔｅｃｔｉｏｎ）を提供する。この方法は、暗号化メッセージ認証コード（ＭＡＣ）を使用して、秘密鍵に基づくデータ完全性チェック（ｄａｔａ ｉｎｔｅｇｒｉｔｙ ｃｈｅｃｋ）を提供する。暗号化メッセージ認証コードは、データ文字列を認証するために使用される短い情報である。メッセージ認証コードは、メッセージ認証アルゴリズムを通じてデータを渡す結果として生じる、暗号チェックサム（ｃｒｙｐｔｏｇｒａｐｈｉｃ ｃｈｅｃｋｓｕｍ）とみなすことができる。ＭＡＣアルゴリズムは、秘密鍵と認証されるべき任意の長さのデータ文字列とを入力として受け入れ、タグと呼ばれることもあるＭＡＣを出力する。ＭＡＣ値を使用して、検証者－彼もまた秘密鍵を有する－がデータ文字列コンテンツへの変更があれば検出できるようにすることにより、データ文字列の完全性および真正性（ａｕｔｈｅｎｔｉｃｉｔｙ）を保護することが可能である。

【００１３】

ハッシュ関数（ｈａｓｈ ｆｕｎｃｔｉｏｎ）とは、任意長さのデータ文字列を固定長さのデータ文字列にマッピングする数学関数である。ハッシュ関数ベースのＭＡＣは、ハッシュ関数とともに鍵を使用して、データ文字列に付加されるチェックサムを生成する。データのＭＡＣは、秘密鍵の知識なしでは構築不可能である。

【００１４】

一例では、データ認証は、データ文字列のタグまたは署名を生成するために使用される、ガロア・メッセージ認証コード（ＧＭＡＣ）のような、更新可能なキー付きハッシュによって提供することができる。これらのコードは、データ文字列の中ほどが変更された場合、データ文字列全体を読み取らずにタグまたは署名を更新することができる、という特性を有する。

【００１５】

ＧＭＡＣとは、データ認証を提供するために使用可能なブロック暗号化コード（ｂｌｏｃｋ ｃｉｐｈｅｒ ｃｏｄｅ）である。これは、ガロア／カウンタ・モード（ＧＣＭ）

認証済み暗号化操作に関して定義される。ＧＣＭ認証済み暗号化操作は、秘密鍵、初期化ベクトル（ＩＶ）、プレーンテキスト（ｐｌａｉｎｔｅｘｔ）、および追加認証済みデータ（ａｄｄｉｔｉｏｎａｌ　ａｕｔｈｅｎｔｉｃａｔｅｄ　ｄａｔａ：ＡＡＤ）に関する入力の、４つの入力を有する。出力は、プレーンテキストと同じ長さの暗号テキスト、および認証タグの、２つである。ＧＭＡＣは、暗号テキスト出力が無視されるＧＣＭの特殊なケースであるため、関数の出力のみが認証タグである。

【００１６】

図３および図４は、本発明の方法の一実施形態を示す流れ図である。データがストレージ・デバイスに書き込まれると、第１のＭＡＣが計算され、媒体または他の不揮発性メモリに格納される。認証中にデータがストレージ・デバイスから読み取られると、第２のＭＡＣが計算され、格納済みの第１のＭＡＣと比較される。第１および第２のＭＡＣが一致しない場合、データは不法に変更されており、ユーザに警告が出されることになる。ＭＡＣは、たとえばホスト・コンピュータによって、ストレージ・デバイス内で、または独立した暗号モジュール内で、計算することができる。ＭＡＣを計算するために使用される鍵は、暗号鍵が格納されているストア（ｓｔｏｒｅ）と同じ鍵ストア（ｋｅｙ　ｓｔｏｒｅ）内に格納することが可能である。鍵ストアは、認証モジュール内部または外部とすることが可能であり、外部の場合、セキュアな鍵転送プロトコルを実施しなければならない。一実施形態では、ＭＡＣはＧＭＡＣとすることができる。

【００１７】

セキュアなディスク・ドライブに格納されたデータは、ディスク・ドライブの電源が入っている間に変更することができない。したがって、通常は、パワー・サイクル毎に１回のみデータ・ブロック・セットを認証し、適切に認証または変更されたものとしてデータ・ブロックにマーク付けすれば十分である。これらのマークは、揮発性ランダム・アクセス・メモリ内に格納し、電源を切る時に消去することができる。

【００１８】

図３に示されるように、データがストレージ媒体に書き込まれる場合、ＭＡＣ関数を使用して、秘密鍵と入力データとを組み合わせることによって形成される第１のＭＡＣを生成することが可能であり（ブロック６２）、すでに格納済みのすべてのデータ・ブロックは同じセットに属する。第１のＭＡＣは、その後、入力データと共にデータ・ストレージ・デバイスに格納することができる（ブロック６４）。ＭＡＣがデータにリンクされる方式は、実施者に委ねられる。たとえば、ＭＡＣをデータ・ブロックに添付することができるか、またはソフトウェア／ディスク・ファームウェアが、データをＭＡＣにリンクするポインタを維持することができる。

【００１９】

図４に示されるように、格納済みのデータの完全性を検証するために、ストレージ・デバイスは、データが格納された際に使用されたものと同じ鍵およびＭＡＣ関数を使用して、格納済みデータについてＭＡＣを計算し（ブロック６６）、その結果と以前に計算されたＭＡＣとを比較する（ブロック６８）。２つの値が一致する場合、格納済みデータは認証されているものとみなされる。２つの値が一致しない場合、ユーザには、たとえばユーザ・インターフェース上のメッセージによって、データ認証の失敗が通知される。

【００２０】

データ・ストレージ・デバイス内のすべてのデータの完全性を検証するためには、あらゆるブロックを読み取らなければならず、たとえこれがセッションごとに１回だけ、たとえば正常なユーザ認証の後、またはセッション終了前、すなわちユーザのログオフ前に実行しなければならないとしても、遅いプロセスである。

【００２１】

本発明は、ほとんど時間的なオーバヘッドなしにアクセス可能な、データ・ブロック・セット（ｓｅｔｓ　ｏｆ　ｄａｔａ　ｂｌｏｃｋｓ）に対応する多くのメッセージ認証コード（ＭＡＣ）値を維持することによって、プロセスの速度を上げる。たとえば、ディスク・ドライブの各トラックについて、更新可能なＭＡＣを維持することができる。したが

って本発明は、ストレージ・デバイスの物理的特徴を使用してブロックを共にグループ化し（ｔｏ ｇｒｏｕｐ ｂｌｏｃｋｓ ｔｏｇｅｔｈｅｒ）、それは新たなシーク操作なしにアクセス可能であり、これは平均して、単一のブロックを読み取るよりも多くともせいぜい２倍の時間しかかからない（ｔｈａｔ ｉｓ ｏｎｌｙ ａｔ ｍｏｓｔ ｔｗｉｃｅ ｓｌｏｗｅｒ ｔｈａｎ ｒｅａｄｉｎｇ ａ ｓｉｎｇｌｅ ｂｌｏｃｋ）。

【００２２】

本発明の一実施形態では、データ・ブロックはグループ化される（ｔｈｅ ｂｌｏｃｋｓ ｏｆ ｄａｔａ ａｒｅ ｇｒｏｕｐｅｄ）ため、ディスク上で即時に、すなわちシーク（ｓｅｅｋ）操作を実行すること、または、ディスク回転時にヘッドの下に所望のブロックが到達するまで待機することなしに、アクセス可能である。理論的には、トラックの半分、またはトラック内の１つおきのセクタが（データの多くの他の部分またはグループと同様に）可能であるが（Ｈａｌｆ ｔｈｅ ｔｒａｃｋ，ｏｒ ｅｖｅｒｙ ｏｔｈｅｒ ｓｅｃｔｏｒ ｉｎ ａ ｔｒａｃｋ ａｒｅ ｔｈｅｏｒｅｔｉｃａｌｌｙ ｐｏｓｓｉｂｌｅ）、それらはすべてトラック全体のアクセス（ａ ｗｈｏｌｅ ｔｒａｃｋ ａｃｃｅｓｓ）に比べて時間を失う。なぜなら、トラックは任意の開始位置から読み取りまたは書き込みが可能である、すなわちシーク操作後の遅延（ｌａｔｅｎｃｙ）がないためである。

【００２３】

ユーザがデータ・ブロックにアクセスし（すなわち読み取り）、そのデータを含むブロック・セット（ｂｌｏｃｋ ｓｅｔ）がまだ認証されていない場合、そのブロック・セットはディスク・ファームウェアによって認証することが可能であり、認証障害が発生すると、ユーザに警告が発せられる。

【００２４】

ディスク・ドライブがアイドル状態の場合、背景プロセス（ｂａｃｋｇｒｏｕｎｄ ｐｒｏｃｅｓｓ）として格納済みのすべてのＭＡＣ値をテストし、それらを検証することができる。合格したＭＡＣ値は、ＲＡＭに格納可能な揮発性テーブル内で識別可能であるため、ユーザがデータにアクセスしたい場合、このテーブルをチェックして所望のデータが認証されているかどうかを判別することができる。認証されている場合、そのデータへのその後のアクセスはほぼ即時に行われる。ユーザがシステムからログオフするか、または電源がオフになった場合、テーブルは消去される（Ｔｈｅ ｔａｂｌｅ ｃａｎ ｂｅ ｅｒａｃｅｄ）。

【００２５】

メッセージ認証コードを追加することによって、オリジナル・データが修正されているか否かを検出することができる。このコンテキストでは、メッセージ認証コードはオリジナル・データに追加されたビット・シーケンスである。修正は、格納済みデータと、格納済みデータおよび秘密鍵から導出されたＭＡＣチェックサムとを比較することによって検出される。出力が格納済みデータと一致する場合、格納済みデータは真正であると見なされる。

【００２６】

他の例では、ＭＡＣの均衡ツリー（ａ ｂａｌａｎｃｅｄ ｔｒｅｅ ｏｆ ＭＡＣｓ）を利用する。この例では、複数のデータ・ブロック・セットについて複数の第１のＭＡＣが計算され、第１のＭＡＣのグループについて複数の追加のＭＡＣが計算される。次に、第１のＭＡＣおよび追加のＭＡＣが互いにツリー構造に編成される。図５は、こうしたツリー構造７０の概略図である。ツリーは、階層構造に配置構成された、複数のノード７２、７４、７６、７８、８０、８２、８４、８６、８８、９０、および９２を含む。この例では、ノード７２がルート・ノードであり、ノード７４および７６はルート・ノードの子である。ノード７８、８０、８２、および８４はノード７４の子であり、ノード８６、８８、９０、および９２はノード７６の子である。図５は、４つの第１のＭＡＣのグループについて、ノード７４および７６でＭＡＣが計算される例を示すが、異なる数の第１のＭＡＣを使用して追加のＭＡＣを計算することが可能である。

【００２７】
ツリー内の各ノードが、データのＭＡＣまたはその子ノードのＭＡＣを含む。最低レベルでは、ノードは、データ・ブロック・セット９４、９６、９８、１００、１０２、１０４、１０６、および１０８内のデータのＭＡＣを含む。データ・ブロック・セットのうちの１つのデータに書き込み操作が発生した場合、変更されたブロックからルートへのパス上にあるＭＡＣのみを更新（すなわち再計算）しなければならない。
【００２８】
ツリーは、コンピュータ・サイエンスにおける標準的なデータ構造である。これらは、親ノードのアドレスを指すポインタを用いて実施することができる。ストレージ・デバイスは、既知のプログラミング方法に従い、ポインタまたはアドレス・テーブルを使用してデータ・ブロックとＭＡＣとの間の関係を追跡することができる。
【００２９】
まとめて認証されたデータ・ブロック内でデータが変更されていない場合、そのブロックに関するＭＡＣは同じままである。あるノードのすべての子が、ツリー内でそれ以上変更されていない場合、そのノード内のＭＡＣも変更されていない。したがって、変更されたブロックの親、親の親など、ルートまでのノードでＭＡＣを修正するだけでよい。
【００３０】
攻撃の可能性とは、攻撃者がデータ・ブロック・セット全体を、それらの古いコンテンツおよび対応する古いＭＡＣに置き換えられることである。古いＭＡＣは計算された時点で有効であったため、依然として有効である。この種のリプレイ攻撃（ｒｅｐｌａｙ　ａｔｔａｃｋ）を検出するために、対応するブロック・セットのＭＡＣを計算し、ＭＡＣが有効であるかどうかを判別するためにチェックしなければならない。その後、ＭＡＣのツリー内で次に高いレベルがチェックされる、という具合である。ルートのＭＡＣはあらゆるデータ・ブロックに依存するため、ディスク・データを以前の状態に完全に復元することのみが成功であり、それよりも小規模のリプレイ（ｒｅｐｌａｙ）は不可能である。したがって、アクセスされたデータの認証では、現在のブロックからパスに沿ってルートまでのすべてのＭＡＣ値を再計算および検証しなければならない。
【００３１】
ツリー構造にはかなりの柔軟性がある。図５は、より実用的なシステムにおける簡略化されたツリー構造を示すが、ノードの第１のレベルは１６の子を有することが可能であり、より高レベルのノードはそれより多いかまたは少ないものとすることができる。2^{48}の論理ブロック・アドレス（ＬＢＡ）があり、異なるディスク標準によるアクセス可能性が最大であり、あらゆるノードが１６の子を有するものと想定すると、第１レベルの各ノードからルート・ノードへのパスの長さは１２である。書き込み操作が発生すると、１５の他のデータ・ブロック・セットを読み取りおよびハッシュしなければならず、ルート・ノードへのパス上では、キャッシュに入れられたデータに対して１１の他のハッシュ操作が実行されることになる。これには何らかの特別な処理が必要であるが、特に１６のデータ・ブロックが互いに隣り合っている場合、処理時間におけるオーバヘッドは管理可能であるため、高速アクセスが可能である。
【００３２】
本発明を実施する場合、ストレージ・オーバヘッドも小さい。たとえば、１２８ビットＭＡＣ値は、１６ＬＢＡごとに１６Ｂ程度のストレージしか必要とせず、これはＬＢＡあたり１バイト、または５１２Ｂブロックではディスク容量の０．２％未満の損失、または４ＫＢブロックでは８分の１（ｅｉｇｈｔ　ｔｉｍｅｓ　ｌｅｓｓ　ｗｉｔｈ　４ＫＢ　ｂｌｏｃｋｓ）である。ＭＡＣ値はＦＬＡＳＨメモリ、または他の不揮発性ＲＡＭ内に格納することもできる。ＲＡＭはデータ・ストレージ・デバイス内に配置可能であるが、他の場所に配置してもよい。
【００３３】
他のハッシュ関数ベースのメッセージ認証コード、１鍵メッセージ認証コード（ｏｎｅ－ｋｅｙ　ｍｅｓｓａｇｅ　ａｕｔｈｅｎｔｉｃａｔｉｏｎ　ｃｏｄｅｓ）、または並列

化可能（ｐａｒａｌｌｅｌｉｚａｂｌｅ）メッセージ認証コードなどの、他のメッセージ認証コードを使用することができる。

【００３４】

以上、本発明についていくつかの例に関して説明してきたが、当業者であれば、添付の特許請求の範囲に示された本発明の範囲を逸脱することなく、説明された例に様々な変更を加えることが可能なことが明らかとなろう。

【図面の簡単な説明】

【００３５】

【図１】本発明の実施形態に従って構築可能な、ディスク・ドライブを示す絵画図である。

【図２】本発明の実施形態に従って構築可能なデータ・ストレージ・デバイスを含む、システムを示すブロック図である。

【図３】本発明の方法の一実施形態を示す流れ図である。

【図４】本発明の方法の一実施形態を示す流れ図である。

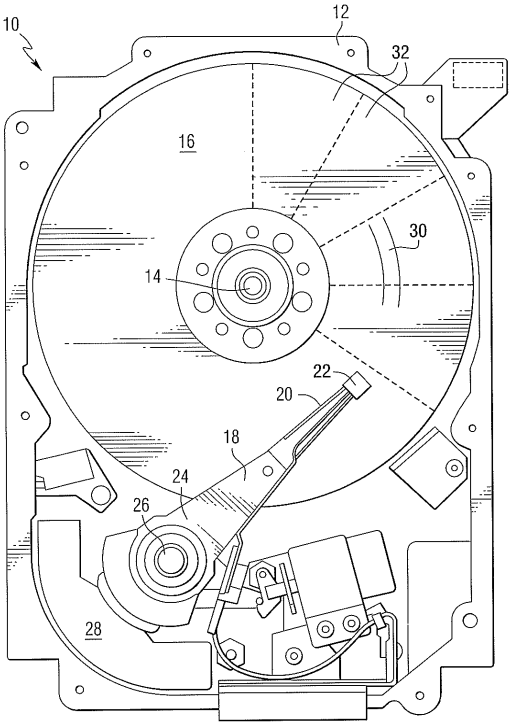
【図５】こうしたメッセージ認証コードのツリー構造を示す概略図である。

【符号の説明】

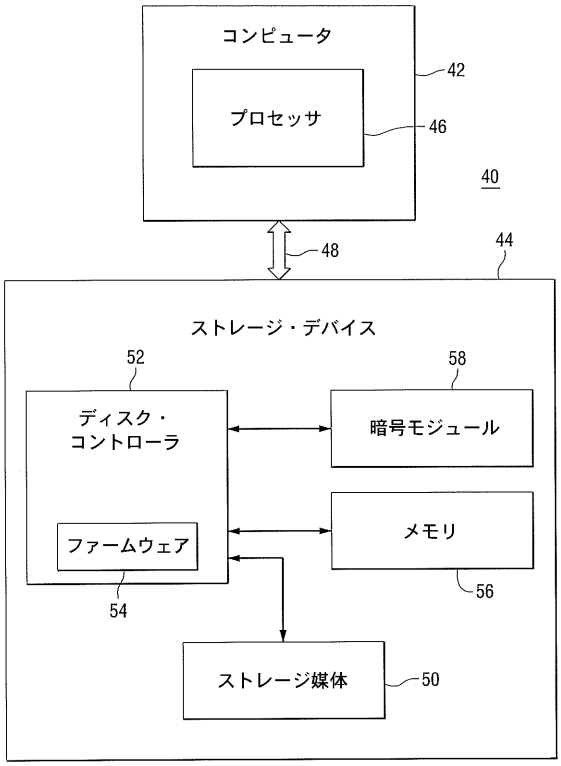
【００３６】

４０　装置
４２　コンピュータ
４４　ストレージ・デバイス
４６　プロセッサ
４８　チャネル
５０　ストレージ媒体
５２　ディスク・コントローラ
５４　ファームウェア
５６　メモリ
５８　暗号モジュール

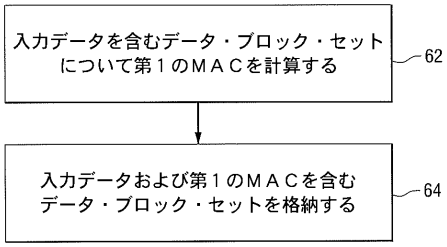
【図１】


10
12
32
16
30
14
22
20
18
24
26
28

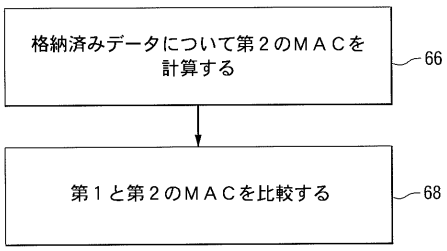
【図２】


コンピュータ
42
プロセッサ
46
40
48
44
ストレージ・デバイス
52
58
ディスク・
コントローラ
暗号モジュール
ファームウェア
メモリ
54
56
ストレージ媒体
50

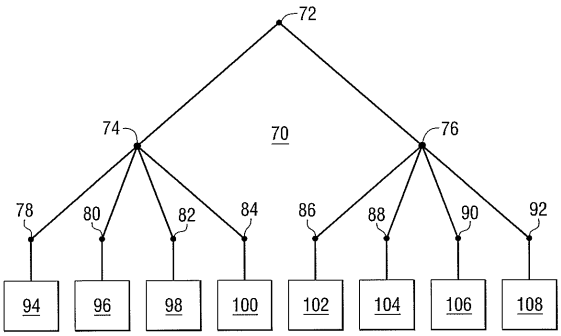
【図３】


入力データを含むデータ・ブロック・セットについて第１のＭＡＣを計算する
62
入力データおよび第１のＭＡＣを含むデータ・ブロック・セットを格納する
64

【図４】


格納済みデータについて第２のＭＡＣを計算する
66
第１と第２のＭＡＣを比較する
68

【図５】


72
74
70
76
78
80
82
84
86
88
90
92
94
96
98
100
102
104
106
108

フロントページの続き

(74)代理人 100111246
弁理士 荒川 伸夫
(74)代理人 100124523
弁理士 佐々木 眞人
(74)代理人 100066692
弁理士 浅村 皓
(74)代理人 100072040
弁理士 浅村 肇
(74)代理人 100091339
弁理士 清水 邦明
(74)代理人 100094673
弁理士 林 鈺三
(72)発明者 ラズロ ハース
アメリカ合衆国、ペンシルヴァニア、クランベリー タウンシップ、 スティープルチェイス ドライブ ３７２

審査官 和田 財太

(56)参考文献 特開２００６－０３９２０６（ＪＰ，Ａ）
特開２００２－０７４８３２（ＪＰ，Ａ）
特表２００８－５１６３１７（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ ２１／２４