(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第5496663号
(P5496663)
(45)発行日　平成26年5月21日(2014.5.21)　(24)登録日　平成26年3月14日(2014.3.14)

(51)Int.Cl.　　F I
H04L 9/32 (2006.01)　　H04L 9/00 675A
G09C 1/00 (2006.01)　　G09C 1/00 640D

請求項の数 12　(全 21 頁)

(21)出願番号　特願2009-519016 (P2009-519016)
(86)(22)出願日　平成19年6月27日 (2007.6.27)
(65)公表番号　特表2009-543498 (P2009-543498A)
(43)公表日　平成21年12月3日 (2009.12.3)
(86)国際出願番号　PCT/IB2007/052496
(87)国際公開番号　WO2008/010119
(87)国際公開日　平成20年1月24日 (2008.1.24)
審査請求日　平成22年6月11日 (2010.6.11)
(31)優先権主張番号　06117041.1
(32)優先日　平成18年7月12日 (2006.7.12)
(33)優先権主張国　欧州特許庁 (EP)

(73)特許権者　598036964
イルデト・コーポレート・ビー・ヴイ
オランダ・ＮＬ－２１３２・エルエス・ホーフドルプ・タウルサヴェンウー・１０５
(74)代理人　100108453
弁理士　村山　靖彦
(74)代理人　100064908
弁理士　志賀　正武
(74)代理人　100089037
弁理士　渡邊　隆
(74)代理人　100110364
弁理士　実広　信哉

最終頁に続く

(54)【発明の名称】デジタルデータ処理装置の耐改竄性

(57)【特許請求の範囲】
【請求項１】
第一のユニット（９０１）と第二のユニット（９０２）とを有するデジタルデータ処理ユニットの改竄に対する抵抗力を増加するシステムであって、
前記第一のユニット（９０１）は、
前記第二のユニット（９０２）からデジタルデータを受信する入力（９０４）と、
受信されたデジタルデータにおける値に基づいた少なくとも１つのルックアップテーブル（９１６）内で発見された予め決定された値に依存して前記受信されたデジタルデータを処理する処理手段（９０６）であって、受信した前記デジタルデータが少なくとも１つの値を含むものであって、受信した前記デジタルデータを処理するとき、前記少なくとも１つの値が、前記第一のユニット（９０１）に前記ルックアップテーブル（９１６）中の少なくとも１つの所定の値を発見させる、前記処理手段とを有し、
前記第二のユニット（９０２）は、
前記少なくとも１つの値を計算する手段（９１２）と、
前記デジタルデータに前記少なくとも１つの値を含む挿入手段（９１０）と、
前記少なくとも１つの値を含む前記デジタルデータを前記第一のユニットに送出する出力手段（９０８）とを有する
ことを特徴とするシステム。
【請求項２】
前記第二のユニットは、前記デジタルデータに暗号化されたコンテンツを含む暗号化手

段を有し、
前記挿入手段は、前記暗号化されたコンテンツに前記少なくとも１つの値を含むために構成され、
前記処理手段は、前記少なくとも１つのルックアップテーブル内で発見された予め決定された値に基づいて前記暗号化されたコンテンツを復号するために構成される、
請求項１記載のシステム。
【請求項３】
前記処理手段の出力を参照値と比較する検証手段を更に有する、
請求項１記載のシステム。
【請求項４】
前記処理手段は、予め決定された順序で前記デジタルデータを処理するために構成され、
前記発見された値は、その処理が発見する動作後に開始するデータの処理の結果に影響を及ぼす、
請求項１記載のシステム。
【請求項５】
前記挿入手段は、前記デジタルデータの予め決定されたブロックを処理する前に、前記処理手段に前記少なくとも１つの予め決定された値を前記ルックアップテーブルで発見させるため、前記デジタルデータに前記少なくとも１つのデータを配置するために構成される、
請求項１記載のシステム。
【請求項６】
前記第一のユニットは、前記第一のユニットにより実行されるソフトウェアのビット表現を記憶するメモリを有し、
前記ソフトウェアのビット表現の少なくとも１部は、前記処理手段によるルックアップテーブルの少なくとも１部として使用され、
前記ルックアップテーブルにおける少なくとも１つの予め決定された値は、前記ソフトウェアのビット表現の少なくとも１部で生じる、
請求項１記載のシステム。
【請求項７】
前記ソフトウェアのビット表現の少なくとも１部は、前記処理手段の動作の一部として実行される命令を含む、
請求項６記載のシステム。
【請求項８】
デジタルデータ処理ユニットの改竄に対する抵抗力を増加するシステム（９０２）であって、
デジタルデータに含むための少なくとも１つの値を計算する手段（９１２）を有し、この値は、前記デジタルデータ処理ユニットが前記少なくとも１つの値を含む前記デジタルデータを処理するとき、前記デジタルデータ処理ユニットに、ルックアップテーブル内で少なくとも１つの予め決定された値を発見させ、
更に、前記デジタルデータに前記少なくとも１つの値を含む挿入手段（９１０）と、
前記少なくとも１つの値を含む前記デジタルデータを前記デジタルデータ処理ユニットに送出する出力手段（９０８）と、
を有することを特徴とするシステム。
【請求項９】
デジタルデータ処理ユニットの改竄に対する抵抗力を増加するシステム（９０１）であって、
デジタルデータを受信する入力（９０４）と、
受信されたデジタルデータにおける値に基づいた少なくとも１つのルックアップテーブル（９１６）内で発見された予め決定された値に依存して前記受信されたデジタルデータ

を処理する処理手段（９０６）とを有し、
前記デジタルデータは、少なくとも１つの値を含み、この値は、前記デジタルデータを処理するとき、前記処理手段に前記ルックアップテーブル内で少なくとも１つの予め決定された値を発見させる、
ことを特徴とするシステム。
【請求項１０】
デジタルデータ処理ユニットの改竄に対する抵抗力を増加する方法であって、
第一のユニット（９０１）において、
第二のユニット（９０２）からデジタルデータを受信する（９０４）ステップと、
受信されたデジタルデータにおける値に基づいた少なくとも１つのルックアップテーブル（９１６）で発見された予め決定された値に依存して前記受借されたデジタルデータを処理する（９０６）ステップであって、受信した前記デジタルデータが少なくとも１つの値を含むものであって、受信した前記デジタルデータを処理するとき、前記少なくとも１つの値が、前記第一のユニット（９０１）に前記ルックアップテーブル（９１６）中の少なくとも１つの所定の値を発見させる、ステップと、を有し、
前記第二のユニット（９０２）において、
前記少なくとも１つの値を計算する（９１２）ステップと、
前記デジタルデータに前記少なくとも１つの値を含む（９１０）ステップと、
前記少なくとも１つの値を含む前記デジタルデータを前記第一のユニットに送出する（９０８）ステップと、を有することを特徴とする方法。
【請求項１１】
プロセッサに、デジタルデータ処理ユニットの改竄に対する抵抗力を増加する方法を実行させるコンピュータプログラムで、あって、前記方法は、
デジタルデータに含むための少なくとも１つの値を計算する（９１２）ステップを有し、この値は、前記デジタルデータ処理ユニットが前記少なくとも１つの値を含むデジタルデータを処理するとき、前記デジタルデータ処理ユニットにルックアップテーブル内で少なくとも１つの予め決定された値を発見させ、
更に、前記デジタルデータに前記少なくとも１つの値を含む（９１０）ステップと、
前記少なくとも１つの値を含む前記デジタルデータを前記デジタルデータ処理ユニットに送出する（９０８）ステップと、を有することを特徴とするコンピュータプログラム。
【請求項１２】
プロセッサに、デジタルデータ処理ユニットの改竄に対する抵抗力を増加する方法を実行させるコンピュータプログラムであって、前記方法は、
デジタルデータを受信する（９０４）　ステップと、
受信されたデジタルデータにおける値に基づいた少なくとも１つのルックアップテーブル内で発見された予め決定された値に依存して前記受信されたデジタルデータを処理する（９０６）ステップと、を有し、
前記デジタルデータは、少なくとも１つの値を含み、この値は、前記デジタルデータを処理するとき、前記プロセッサに前記ルックアップテーブル内で少なくとも１つの予め決定された値を発見させる、
ことを特徴とするコンピュータプログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、デジタルデータ処理装置の改竄に対する抵抗力を増加することに関する。
【背景技術】
【０００２】
インターネットは、デジタルコンテンツへの便利且つユビキタスなアクセスをユーザに提供する。パワフルな分散チャネルとしてインターネットのポテンシャルのため、多くのコンシューマエレクトロニクス（ＣＥ）の製品は、インターネットに直接アクセスするか

、又はＰＣプラットフォーム－インターネットへの主要なポータルと相互運用するのを試みる。ＣＥ製品は、限定されるものではないが、デジタルセットトップボックス、デジタルＴＶ、ゲームコンソール、ＰＣ、及び、さらに、ＰＤＡ、移動電話のようなハンドヘルド装置、及びアップル社のｉＰｏｄのようなモバイルストレージ及びレンダリング装置を含む。著作権利化されたコンテンツの流通媒体としてのインターネットの使用により、コンテンツプロバイダの関心を確実にするために、強力な課題が形成される。特に、コンテンツプロバイダの著作権及びビジネスモデルを正当化することが要求される。次第に、ＣＥプラットフォームは、適切なソフトウェアがロードされたプロセッサを使用して動作される。かかるソフトウェアは、オーディオ及び／又はビデオのようなデジタルコンテンツのレンダリング（プレイバック）の機能の主要部分を含む場合がある。プレイバックソフトウェアの制御は、コンテンツが使用される諸条件を含めたコンテンツオーナの利益を強化する１つの方法である。慣習的に、（ＰＣ及びＰＤＡを除いて）多くのＣＥプラットフォームがクローズされるために使用される場合、今日、益々多くのプラットフォームが少なくとも部分的にオープンにされる。特に、ＰＣプラットフォームについて、幾つかのユーザは、コンテンツへのアクセス、及びコンテンツプロテクションメカニズムをアタック及びバイパスする大量の時間及びリソースを提供するハードウェア及びソフトウェアを通して完全な制御を有することが想定される。結果として、コンテンツプロバイダは、全てのユーザ又は装置を信頼することができないコミュニティに敵対するネットワークを通して合法的なユーザにコンテンツを伝達する必要がある。

【０００３】

典型的に、デジタル著作権管理システムは、ラウンドと呼ばれる暗号／復号ステップのシーケンスを使用してブロックでデータストリームを処理するブロック暗号に基づいて暗号化技術を使用する。それぞれのラウンドの間、ラウンドに特化した機能が実行される。ラウンドに特化した機能は、ラウンドに特化したサブキーの制御下で実行される同じラウンド機能に基づく場合がある。多くの暗号化システムについて、ラウンド機能は、マッピングテーブル又はルックアップテーブルを使用して規定される。明示的なテーブルが使用されない場合であっても、暗号／復号機能のソフトウェアにおける有効な実行のため、異なる部分の機能についてテーブルが頻繁に使用される。コンピュータコードは、テーブルの値を機能のレンジの値にアクセス又は結合する。ユーザに特化したキーを分配する代わりに、暗号又は復号アルゴリズムのためのキーの代わりにユーザに特化したアルゴリズムを配信する関心が益々増えている。これらのアルゴリズムは、大部分は機能（マッピング）であり、キーのようなエレメントの再設計を防止するか又は再計算を禁止するために難読化される（隠される）。コンピュータでは、幾つかのコンピュータコードを伴うテーブルは、これらの機能を表すことがある。

【０００４】

コンテンツプロバイダは、全てのユーザ又は装置を信頼することができないコミュニティに敵対するネットワークを通して合法的なユーザにコンテンツを伝達する必要がある。特に、ＰＣプラットフォームについて、ユーザは、コンテンツへのアクセス、及びコンテンツプロテクションメカニズムをアタック及びバイパスするための制限されない量の時間及びリソースを提供するハードウェア及びソフトウェアの完全な制御を有することが想定される。コンテンツが使用される諸条件を実施するソフトウェアコードは、改竄されてはならない。ＰＣに配信されるプロテクトされたコンテンツのデジタル著作権の管理における一般的なアプローチは、たとえばＤＥＳ（Data Encryption Standard）、ＡＥＳ（Advanced Encryption Standard）、又はWO9967918で開示される方法を使用してデジタルコンテンツを暗号化し、復号キーを使用することである。

【０００５】

暗号化に頼るデジタル著作権管理の脆弱性の２つの主要な領域は、コンテンツが使用される諸条件を実施するソフトウェアプラグイン、並びに、キーディストリビューション及びハンドリングである。

【０００６】

典型的に、プラグインは、コンテンツが使用されるべき諸条件を実施する。これらの諸条件を取り除こうとするアタッカーは、ソフトウェアのプラグインに含まれるプログラムコードの改竄を通して、これを達成しようとする。

【０００７】

キーハンドリングに関連して、プレイバックについて、メディアプレーヤは、ライセンスデータベースから復号キーを取り出す必要がある。次いで、メディアプレーヤは、暗号化されたコンテンツの復号のため、メモリにおける何処かにこの復号キーを記憶する必要がある。これは、アタッカーにキーへのアタックのための２つのオプションを残す。はじめに、ライセンスデータベースアクセス機能のリバースエンジニアリングにより、ブラックボックスソフトウェアとなり（すなわちアタッカーはソフトウェアの機能の内部動作を理解する必要がない）、アタッカーは全てのライセンスデータベースからアセットキーを取り出すことができる。第二に、コンテンツの復号の間にメモリへのアクセスの観察により、アセットキーを取り出すことが可能である。両者のケースでは、キーは信用できなくなったと考慮される。

【０００８】

改竄に対する抵抗があるソフトウェアは、ソフトウェアとの目標指向の改竄が複雑にされるためにそのように呼ばれる。ソフトウェアアプリケーションの改竄に対する抵抗を増加する様々な技術が存在する。これらの技術の大部分は、ソフトウェアアプリケーションの制御及びデータパスの両者におけるランダム性及び複雑さのベールを追加することでアプリケーションの埋め込まれた情報を隠すことに基づいている。この背後にある考えは、コードの調査により情報を抽出することが更に困難となることである。したがって、たとえばアプリケーションのアクセス及び許可制御に対処するコードを発見し、結果的に、それを変えることが更に困難になる。

【０００９】

“White-Box Cryptography and an AES Implementation”, by Stanley Chow, Philip Eisen, Harold Johnson, and Paul C. Van Oorschot, in Selected Areas in Cryptography: 9th Annual International Workshop, SAC2002, St. John’s, Newfoundland, Canada, August 15-16, 2002は以下で“Chow 1”と呼ばれ、“A White-Box DES Implementation for DRM Applications”, by Stanley Chow, Phil Eisen, Harold Johnson, and Paul C. van Oorshot, in Digital Rights Management: ACM CCS-9 Workshop, DRM 2002, Washington, DC, USA, November 18, 2002は以下で“Chow 2”と呼ばれ、これらは、個々のステップではなくコンポジションを表すランダムな全単射によりそのテーブルをエンコードすること、暗号化の境界を閉鎖したアプリケーションに押し出すことで暗号化の境界を拡張することの組み合わせによりキーを隠す方法を開示する。

【００１０】

WO2006/046187は、システムにおいて、サーバが実行装置に暗号化関数Ｆを難読化された形式でどのように提供するかを開示する。関数Ｆは、アベールグループ演算子

【００１１】

【外１】

【００１２】

$\otimes$

を使用して複数のマッピングテーブルT_i（０≦ｉ≦ｎ；ｎ≧１）の出力を構成する。プロセッサは、

【００１３】

【数１】

$$C[x] \otimes O[x] = 0, \forall x \in D_i$$

となるようにテーブルＯ及びＣを選択し、テーブルT'_i，０≦ｉ≦ｍ；ｎ≦ｍ≦ｎ＋１

を形成し、ここで０≦ｉ≦ｎである。それぞれのテーブルＴ’$_i$は、それぞれ対応するテーブルＴ$_i$を表し、少なくとも１つのテーブルＴ’$_{o1}$（０≦ｏ$_1$≦ｎ）は、Ｔ$_{o1}$及びＯのアベールコンポジションを通して形成され、少なくとも１つのテーブルＴ’$_{c1}$（０≦ｃ$_1$≦ｍ，ｃ$_1$≠ｏ$_1$）は、Ｃを含むアベールコンポジションを通して形成される。テーブルＴ’$_i$を実行装置に提供する手段が使用される。実行装置は、テーブルを受ける手段、テーブルＴ’$_i$のアベールコンポジションにより暗号化関数Ｆに機能的に等価な関数Ｆ’を形成するプロセッサを含む。

【先行技術文献】

【特許文献】

【００１４】

【特許文献１】国際特許出願WO2006/046187公報

【特許文献２】国際特許出願WO2006/046187公報

【非特許文献】

【００１５】

【非特許文献１】“White-Box Cryptography and an AES Implementation”, by Stanley Chow, Philip Eisen, Harold Johnson, and Paul C. Van Oorschot, in Selected Areas in Cryptography: 9th Annual International Workshop, SAC2002, St. John’s, Newfoundland, Canada, August 15-16, 2002

【非特許文献２】“A White-Box DES Implementation for DRM Applications”, by Stanley Chow, Phil Eisen, Harold Johnson, and Paul C. Van Oorshot, in Digital Rights Management: ACM CCS-9 Workshop, DRM 2002, Washington, DC, USA, November 18, 2002

【発明の概要】

【発明が解決しようとする課題】

【００１６】

デジタルデータ処理装置の改竄に対する抵抗を増加する改善されたシステムを有することが有利である。

【課題を解決するための手段】

【００１７】

この課題に良好に対処するため、本発明の第一の態様では、以下を有するシステムが提供される。第一のユニット９０１は、デジタルデータを受信する入力（ＩＮ）９０４、受信されたデジタルデータにおける値に基づいた少なくとも１つのルックアップテーブル（ＬＵＴ）９１６で発見される値に依存して受信されたデジタルデータを処理する処理手段（ＰＲＯＣ）９０６を有しており、第二のユニット９０２は、デジタルデータを処理するとき、ルックアップテーブルにおいて少なくとも１つの予め決定された値を第一のユニットに発見させる、デジタルデータに含まれる少なくとも１つの値を計算する手段（ＣＯＭＰ）９１２、デジタルデータに少なくとも１つの値を含む挿入手段（ＩＮＳ）９１０、デジタルデータを第一のユニットに送信する出力（ＯＵＴ）９０８を有する。

【００１８】

第一のユニットにより実行される処理は、データのレンダリングを有する。データは、エンコードされたオーディオ及び／又はビデオコンテンツを有する。第一のユニットは、１以上のルックアップテーブル９１６を有し、デジタルデータの処理は、多数のテーブルのルックアップを実行することで少なくとも部分的に実行される。多数のシステムでは、データ及び／又は擬似ランダムプロセスは、どのルックアップテーブルのエントリがデコーディングのために使用されたかを判定する。かかるシステムに関し、所定のルックアップテーブルのエントリは使用されないか、又はデータの比較的大きな部分を既にデコードした後にのみ使用されることが生じる。本発明のこの態様により、第二のユニットは、データを処理するときにアクセスされるルックアップテーブルにおいて少なくとも１つの予め決定された値を規定することができる。第二のユニットは、少なくとも１つの値をデータに含ませることで、これを規定することができ、少なくとも１つの値は、予め決定され

た値がアクセスされ処理で使用されるように選択される。ルックアップテーブルにおける予め決定された値がアタッカーにより変更された場合、処理は失敗する。これは、成功する処理は変更されない値の使用を必要とするためである。

【００１９】

第一のユニットは、たとえば、ユーザ端末、コンピュータ、セットトップボックス又はテレビジョンである。第二のユニットは、たとえばコンテンツサーバ又はサーバである。第一及び第二のユニットは、単一の物理的な装置の一部である。

【００２０】

本発明の態様によれば、第二のユニットは、暗号化されたコンテンツをデジタルデータに含む暗号化手段を有し、挿入手段は、少なくとも１つの値を暗号化されたコンテンツに含むために構成され、処理手段は、発見する動作に基づいて暗号化されたコンテンツを復号するために構成される。

【００２１】

復号及び暗号化は、ルックアップテーブルを使用して効果的に実現することができる。値を暗号化されたコンテンツに挿入することで、デコーダは、係るように挿入された値を暗号化されたデータに現れる他の値から区別するのを可能にすることなしに、予め決定されたルックアップテーブルのエントリにアクセスする。デコードの後、デコードされたデータは、挿入された値のデコードの結果を識別するラベルを含み、これにより、処理手段は、挿入された値のデコードの結果を廃棄することができる。

【００２２】

本発明の態様によれば、本システムは、処理手段の出力を参照値と比較する照合手段を有する。

【００２３】

照合手段は、第一のユニット、第二のユニット又は第三のユニットに含まれ、処理の結果を受ける。照合手段は、処理の結果を正しい結果であると考えられる値と比較する。ルックアップテーブルにおける予め決定された値がそれが含むべき値を含まない場合、比較は失敗し、照合ユニットは改竄を検出する。

【００２４】

本発明の態様によれば、処理手段は、予め決定された順序でデジタルデータを処理するために構成され、発見された値は、発見する動作の後にその処理が開始するデータの処理結果に影響を及ぼす。

【００２５】

この態様は、ルックアップテーブルの値が改竄された場合、データの比較的大きな部分が正しく処理されないという利点を有する。挿入された値の後の多数のデータは、係る改竄の場合に誤って処理されるであろう。

【００２６】

本発明の態様によれば、挿入手段は、デジタルデータにおいて少なくとも１つの値を位置決めするために構成され、デジタルデータの予め決定されたブロックを処理する前にルックアップテーブルにおける少なくとも１つの予め決定された値をデコードする手段に調べさせる。

【００２７】

これは、デジタルコンテンツの予め決定されたブロックが改竄されているシステムでの処理に対して良好にプロテクトされることを確認することである。

【００２８】

本発明の態様によれば、第一のユニットは、第一のユニットにより実行されるべきソフトウェアのビット表現を記憶するメモリを有し、ソフトウェアのビット表現の少なくとも１部は、処理手段によりルックアップテーブルの少なくとも１部として使用され、ルックアップテーブルにおける少なくとも１つの予め決定された値は、ソフトウェアのビット表現の少なくとも１部で生じる。

【００２９】

メモリ位置が２つの独立した目的のために使用される事実により、そのメモリ位置で改竄することが更に困難になる。これは、メモリ位置の最初の使用に関連される目的を達成するために変更が行われた場合、この変更は克服することが困難であるやり方でメモリ位置の第二の使用にも影響を及ぼすためである。

【００３０】

同時係属の特許出願EP06116693.0（代理人ドケットPH005600）は、ソフトウェアシステムの改竄の抵抗を増加する方法を開示しており、この方法は、複数のパラメータに基づいてデジタルデータを処理する複数のコンピュータ実行可能な命令を構成するステップ、処理の間に読取り可能なコンピュータ実行可能なコードのビット表現に等しい部分を、パラメータのビット表現において識別するステップ、命令の実行の間、参照によりパラメータのビット表現の等しい部分を読取るコードのビット表現を保持する少なくとも１つのメモリアドレスを使用する命令を構成するステップを含む。

【００３１】

本発明の態様によれば、ソフトウェアのビット表現の少なくとも１部は、処理手段の動作の一部として実行される命令を含む。

【００３２】

ルックアップテーブルと同様に処理手段の一部として実行される命令は、それらを１つの纏まりとし、メモリに単一のコピーを記憶することでプロテクトされる。さらに、処理手段は、復号手段、暗号化手段、又は圧縮（伸張）手段を有する。

【００３３】

本発明のこれらの態様及び他の態様は、添付図面を参照して以下に明らかにされるであろう。

【図面の簡単な説明】

【００３４】

図１は、ＡＥＳのラウンドにおける動作を説明する図である。

図２は、難読化されたテーブルの例を示す図である。

図３は、実施の形態を説明する図である。

図４は、処理ステップを説明するフローチャートである。

図５は、実施の形態を説明する図である。

【発明を実施するための形態】

【００３５】

図３は、本発明の実施の形態を説明する。この図は、第一のユニット９０１及び第二のユニット９０２を示す。第一のユニット９０１は、出力９０８を介して第二のユニット９０２からデータを受信するために入力９０４を使用する。入力９０４は、たとえばＴＣＰ／ＩＰサポート又は取り外し可能な記憶媒体（たとえばＤＶＤ，ＣＤ，テープ）からデータを読取るユニットをもつネットワーク入力を有する。出力９０８は、これに応じて、ネットワーク出力又はたとえばディスクマスタリング機器を有する。矢印で接続される出力９０８と入力９０４は、出力９０８から入力９０４にデータを送信するディストリビューションチャネルを表す。第二のユニット９０２は、データストレージ又はデータジェネレータ９１４を更に有する。データ９１４は、ＺＩＰ、ＭＰＥＧ、ＭＰ３のような特定のフォーマットでエンコードされ、また暗号化される。また、第二のユニットは、特定の値をデータストリームに挿入する挿入手段９１０を有する。挿入手段９１０は、挿入された値がオリジナルのデータ９１４のフォーマットに準拠するのを確かめるため、幾つかの情報を組み込む。データがオンザフライで生成された場合、挿入手段は、データフォーマットで挿入された値を適切に収容するためにデータジェネレータと協働する。また、挿入された値は、それらが係るように認識することができるようにラベル付けされる。これは、第一のユニットが挿入された値を誤って扱うのを防止するためである。しかし、ラベリングは、幾つかの処理が行なわれた後に現れるだけである。

【００３６】

第一のユニット９０１は、処理手段９０６及び１以上のルックアップテーブル９１６を

更に有する。処理手段は、到来するデータを処理する。処理手段は、データが提供されるフォーマット（ＺＩＰ、ＭＰＥＧ３、ＭＰ３、暗号化等）を解釈する。たとえば、ＺＩＰフォーマットは、処理が伸張を含むことを意味する。また、処理手段は、ＭＰＥＧ，ＭＰ３又は他のフォーマットをデコードするために構成される。最終的な例として、処理手段は、到来するデータを復号又は暗号化するために構成される。処理手段９０６は、ルックアップテーブル９１６の１つにおけるデータから導出される値を調べることで少なくとも部分的に処理を実行する。調べられた値は、たとえば次のテーブルルックアップを定義するためといった、更なる処理のために使用されるか、又は、調べられた値は、出力データを表す。ルックアップテーブルは、データの処理において重要な役割を担う。ルックアップテーブルのエントリがアタッカーにより変更された場合、そのルックアップテーブルのエントリはデータに関連して使用され、処理手段は、誤って処理されたデータを生成する。これにより、アタッカーが成功する変更を行なうことが更に困難になる。

【００３７】

あるパーティが第一のユニット９０１のインテグリティに関心があるとする。そのパーティは、第一のユニット９０１のインテグリティを確認するために第二のユニット９０２を使用する。勿論、インテグリティは、入力データから導出されるルックアップにより連続的にチェックされる。しかし、第二のユニット９０２は、特定の予め定義された値の存在をチェックするために使用することができる。このため、挿入手段９１０は、特に計算された値をデータストリームに挿入する。第二のユニット９０２は、デジタルデータに含むための値を計算する手段９１２を更に有する。値は、第一のユニット９０１の一部である処理手段９０６がデータを処理しているときに特定の予め定義された値を調べる方法で計算される。そのため、値を計算する手段９１２は、処理手段９０６及び／又はルックアップテーブル９１６に関する情報を有する。

【００３８】

幾つかのケースでは、挿入された値は、ルックアップテーブルのエントリをチェックするためにもっぱら含まれているので、処理されたデータで使用されるべきではない。このため、挿入手段９１０は、挿入された値を識別するためにデータにマーカを含むために構成される。好ましくは、さもなければアタッカーは挿入された値をスキップするために処理手段９０６に手を加えるため、処理手段９０６によりデータを処理した後にマーカが目に見えるだけである。さらに、符号化は、通常は、挿入された値が挿入された値の前後のデータの幾つかにも影響を及ぼすものである。これにより、ルックアップテーブルのエントリが変更されない場合に、挿入された値の周りのデータを使用不可能にされ、これがまさに望まれるものである。また、処理されたデータ（の一部）を第二のユニット９０２にリターンするのを第二のユニット９０２が第一のユニット９０１に要求することが可能である。第二のユニット９０２は、処理されたデータを確認し、これにより予め定義されたルックアップテーブルのエントリが変更されたか否かを発見する。

【００３９】

以下では、アルゴリズムの実現が、難読化されたルックアップテーブルを使用することで、どのように更に改竄に対する抵抗を高くして行なわれるかを示す。また、テーブル値によるコードの一体化が説明される。ＡＥＳ及びＤＥＳの例が与えられる。しかし、本方法は、多くの異なる種類のアルゴリズム、特に１以上のルックアップテーブルを使用して実現することができるアルゴリズムに適用することができる。

【００４０】

［難読化したルックアップテーブル］

ソフトウェアアプリケーションの制御及びデータパスにおけるランダム性と複雑度のベールを追加するアプローチは、ソフトウェアが改竄を受けるのを妨げないが、改竄を行なう者の目的を達成するためにどのような変更が行われる必要があるかを判定することが更に困難になるだけである。耐改竄性の背後にある一般的な原理は、以下のように概説することができる。プログラムＰは、アクセス制御及び／又は許可Ｘ及び機能Ｙのコンポジットとして表現することができる。アタッカーは、アクセス制御又は許可が機能に影響を及

ぼすことなしに除かれるように、プログラムを改竄するのを望む場合がある。改竄されたプログラムは、次いで、アクセス制御又は許可が全くなしに実行されるか、又はこれらの制御が無視されるように少なくとも実行される。本発明は、Ｙが処理機能を含むケースについて主に説明される。この機能は、暗号化、復号、圧縮、伸張、レンダリング、評価、認証を含む。本発明は、任意の種類の機能Ｙに適用される。

【００４１】

理想的に、改竄防止プログラムについて、どんなに改竄が小さかろうと、Ｘを改竄することにより即座にＹの損失となる。言い換えれば、Ｘ及びＹは分離することが不可能であり、又は少なくとも大きな困難によってのみ分離可能である。分離不可能なことを実現する１つの方法は、Ｘへの意図される変更によりＹへの意図されない変更となり、これによって機能がＹから除かれるように、ＸとＹの間の関係を形成することである。Ｙの機能を回復するため、Ｙへの更なる変更が必要とされる。プログラムの機能及び制御エレメントは分離不可能にされているとき、アタックは非常に困難となる。係る分離不可能なことがプログラムのコードにわたり形成された場合、プログラムは、プログラムコードがベールで必然的に隠される必要なしに耐改竄性にされる。耐改竄性のソフトウェアは、目標指向の改竄を実行するために複雑にされるソフトウェアである。

【００４２】

ＡＥＳは、１２８バイト又は１６バイトのブロックサイズをもつブロック暗号である。プレインテキストは、符号化アルゴリズムの初期状態を形成する１６バイトのブロックに分割され、符号化アルゴリズムの最後の状態は、暗号文である。ＡＥＳを概念的に説明するため、状態のバイトは、４×４バイトのマトリクスとして編成される。ＡＥＳは多数のラウンドから構成される。それぞれのラウンドは、状態マトリクスのバイト、行又は列について動作する類似の処理ステップから構成され、それぞれのラウンドは、これらの処理ステップにおいて異なるラウンドキーを使用する。

【００４３】

図１は、ＡＥＳのベーシックラウンドの幾つかの主要な処理ステップを示す。処理ステップは、以下を含む。

AddRoundKey２；状態のそれぞれのバイトはラウンドキーのバイトでＸＯＲされる。 SubBytes４；ルックアップテーブルを使用したバイト対バイトの置換。

ShiftRows６；状態のそれぞれの行は固定されたバイト数で回転される。

MixColumns８；それぞれの列はＧＦ（2^8）においてモジュロ乗算を使用して処理される。

【００４４】

ステップSubBytes４、ShiftRows６及びMixColumns８は、使用される特定のキーに独立である。キーは、ステップAddRoundKey２で使用される。ステップShiftRows６を除いて、処理ステップは、他の列の情報なしに４×４状態マトリクスのそれぞれの列で実行することができる。したがって、これらは、それぞれの列が４つの８ビット値から構成されるとき、３２ビット演算としてみなすことができる。破線１０は、必要とされるラウンド数が実行されるまでプロセスが繰り返されることを示す。

【００４５】

これらのステップのそれぞれ又はステップの組み合わせは、ルックアップテーブルにより表現されるか、ルックアップテーブルのネットワーク（Ｓボックス）により表現される。また、フルラウンドのＡＥＳをルックアップテーブルのネットワークで置き換えることも可能である。たとえば、AddRoundKeyステップは、ラウンドキーと単にＸＯＲすることで実現することができ、SubBytes、ShiftRows、MixColumnsステップは、テーブルルックアップを使用して実現される。しかし、これは、ホワイトボックスのアタックのコンテクストでアタッカーにとってキーがなお目に見えることを意味する。AddRoundKeyステップは、ルックアップテーブルに埋め込まれ、これにより、キーを発見するために明らかさを低くする。図示されるステップ２，４，６及び８の順序は、暗号化のために通常使用される。復号について、ステップは、逆の順序で実行される。しかし、図示されるようにステ

ップ２，４，６及び８の順序を使用するように、復号プロセスを再記述することができる。

【００４６】

図２は、キーを抽出するのを更に困難にする方法を示す。Ｘ及びＹを２つの関数とする。ダイアグラム１２として例示される演算

【００４７】

【数２】

$$Y \circ X(c) = Y(X(c))$$

を考え、ｃは、たとえば４バイトの状態の列といった入力値である。しかし、このアプローチは、任意のタイプの入力値ｃに適用される。マッピングＸ及びＹは、メモリに記憶することができるルックアップテーブルとして実現されるが、これらがメモリに記憶されるとき、値がアタッカーにより読取ることができる。図１４は、入力符号化Ｆと出力符号化Ｈとを使用して、ルックアップテーブルのコンテンツがどのように難読化することができるかを示す。

【００４８】

【外２】

【００４９】

$$X \circ F^{-1} \text{ 及び } H \circ Y$$

に対応するルックアップテーブルは、Ｘ及びＹの代わりに例示されるように記憶され、Ｘ及びＹを抽出することが更に困難になる。ダイアグラム１６は、更なる、２つのテーブルの中間の結果も符号化されるように、たとえばランダムな全単射の関数（bijective function）をどのように加えるべきかを示す。このケースでは、２つのテーブル

【００５０】

【数３】

$$X' = G \circ X \circ F^{-1} \text{ . 及び } Y' = H \circ Y \circ G^{-1}$$

がメモリに記憶される。これは、ダイアグラム１８において再び示される。

【００５１】

【数４】

$$Y' \circ X' = (H \circ Y \circ G^{-1}) \circ (G \circ X \circ F^{-1}) = H \circ (Y \circ X) \circ F^{-1}$$

ここで

【００５２】

【外３】

【００５３】

$\circ$

は関数のコンポジションを示し（すなわち２つの関数ｆ（ｘ）及びｇ（ｘ）について定義により

【００５４】

【数５】

$$f \circ g\,(x) = f(g(x))$$

）、Ｘ及びＹは、ルックアップテーブルによる実現に適した関数である。同様に、２を超える関数から構成されるネットワークが符号化される。Ｘ及びＹを符号化する実際のテーブルは、単一のルックアップテーブルに

【００５５】

【外４】

【００５６】

$$H \circ Y \circ G^{-1}$$

を組み込み、単一のルックアップテーブルに

【００５７】

【外５】

【００５８】

$$G \circ X \circ F^{-1}$$

を組む込むことで難読化される。Ｆ，Ｇ及び／又はＨが未知のままである限り、アタッカーは、ルックアップテーブルからＸ及び／又はＹに関する情報を抽出することができず、したがって、アタッカーは、Ｘ及び／又はＹに基づいてキーを抽出することができない。ＤＥＳ及びＲｉｊｎｄａｅｌを含む他の暗号化アルゴリズムは（そのうちＡＥＳが特定のインスタンス化である）、先に類似したやり方で難読化されるルックアップテーブル（のカスケード又はネットワーク）として符号化される場合もある。本発明は、上述された例示的な暗号化アルゴリズムに限定されない。

【００５９】

Chow1は、個々のステップではなくコンポジションを表現するランダムな全単射によりそのテーブルを符号化することでキーを隠すことを意図する方法を開示する。秘密鍵の抽出を防止することは、ソフトウェアプロテクションの目的が他のコンピュータでバイパスされるキーイングマテリアルをアタッカーが抽出するのを防止されるか、又はインストールされたソフトウェアの大きなユーザベースにわたり安全対策を打破する「グローバルクラック“global crack”」を効果的に形成するキーイングマテリアルを公開するのを防止されるという利点を有する。これは、ソフトウェアのみのソリューションの制約及び敵対的なホストのリアリティを前提として、増加されたプロテクションの程度を提供する。Chow1のアプローチでは、（１）個々のステップではなくコンポジションのためのテーブルを使用し、（２）これらのテーブルをランダムな全単射でエンコードし、（３）暗号化の境界を暗号化アルゴリズム自身を超えて閉鎖したアプリケーション（containing application）にまで拡張し、アタッカー（リバースエンジニア）にそれらの目的を達成するために著しく大きなコードセグメントを理解させることで、キーが隠される。Chow1は、固定されたキーアプローチを説明している。キーは、キー入力が不要であるように、キーに関して部分的な評価による実現において埋め込まれる。部分的な評価は、キーを含む表現ができるだけ合理的に評価されることを意味し、結果は、完全な表現ではなくコードに配置される。アタッカーは、キーに特化した実現を抽出し、それをキーの代わりに使用するが、暗号は、典型的に、コンポーネントが設計され、敵対者が取り除くのが困難であることがわかる、処理された又は符号化された形式で暗号コンポーネントに入力を提供することができる大きな閉鎖したシステム（containing system）のコンポーネントである。符号化テーブルのステップを参照して、符号化は任意であるので、１つのステップの出力の符号化が継ぎの入力符号化に一致する場合にのみ結果に意味がある。たとえば、ステップＸにステップＹが続く場合（

【００６０】

【外６】

【００６１】

$$Y \circ X$$

の計算となる）、計算は

【００６２】

【数６】

$$Y' \circ X' = (H \circ Y \circ G^{-1}) \circ (G \circ X \circ F^{-1}) = H \circ (Y \circ X) \circ F^{-1}$$

として符号化される。このように、
【００６３】
【外７】
【００６４】

$$Y \circ X$$

は、入力がＦで符号化される必要があり、出力がH^{-1}で復号化される必要があるにもかかわらず、適切に計算される。このステップは、Ｙ’及びＸ’に対応するテーブルとして個別に表され、Ｆ，Ｇ及びＨはＸ及びＹと同様に隠される。かかる曖昧なステップから離れて、Chow1は、基礎をなす動作を更に隠すため、線形（全単射）変換による拡散ステップを使用する。用語「混合の全単射“mixing bijection”」は係る線形の変換を記述するために使用される。Chow1の実現は、処理された形式で入力を受け、異なって処理された形式で出力を生成し、これにより、ホワイトボックスのアタックコンテクスト（WBAC: white-box attack context）の抵抗力のあるＡＥＳがその閉鎖したアプリケーションから分離するのを困難にする。
【００６５】
Chow2は、プログラムからの秘密鍵の抽出を防止する目的による、ホワイトボックスアタックコンテクストに抵抗するために設計されるＤＥＳの暗号化の実現を示す。難読化するルックアップテーブルのネットワークに関するこの文献で議論される技術は、ＡＥＳ及び他を含む他の暗号化アルゴリズムにも大部分について適用される。実行環境を制御するアタッカーがキーを明示的に抽出することなしにソフトウェア自身を（たとえば復号のため）明らかに使用することができる一方で、インストールされたインスタンスを使用するのをアタッカーに強制することは、デジタル著作権の管理（ＤＲＭ）システムプロバイダにとって価値がある。一般に、Chow2におけるアプローチは、置換ボックスを全体的に構成する実現に向かって機能し、アフィン変換を実現するものではない。Chow2では、一般的なアプローチをサポートするために必要とされる多数の技術が開示されている。これらの技術の幾つかは、Ｉ／Ｏブロック符号化、結合機能符号化、バイパス符号化、スピリットパス符号化、及び出力スプリッティングである。
【００６６】
部分的な評価は、実現（implementation）時に（部分的に）知られている値に基づく表現が事前に評価されることを意味する。簡略化された例では、キーが“５”であって、オリジナルの実現が表現“２*key”を含むとき、“２*５”を実現に組み込むよりはむしろ、事前に評価された表現“１０”が実現に組み込まれる。このように、キー“５”はコードにおいて直接に存在しない。固定されたキーをもつＤＥＳのケースで、これは、（ランタイムでキーから計算される）標準的なＳボックスを（コンパイル時間で又はコンパイル時間前にキーから計算される）キーに特化した事前に評価されたＳボックスで置き換えることを含む。Chow2に係る混合の全単射は、それぞれの出力ビットが多数の入力ビットに依存するように設計された全単射のアフィン変換である。Ｉ／Ｏブロック符号化は、多数の入力及び出力ビットを扱う符号化方法である。このケースでは、符号化／復号化は、符号化の連結として計算することができ、この場合、それぞれの符号化は、入力／出力ビットのサブセットを処理する。結合された機能の符号化は、２以上の動作が並列に処理することができる場合に、単一の符号化機能が並列の動作の入力（それぞれ出力）の連結に適用されることを意味する。これは、多かれ少なかれＩ／Ｏブロック符号化の反対である。バイパス符号化は、符号化の変換が多数のエントロピーの余分なビットを難読化されるべき変換の入力及び／又は出力に加え、手順の最終的な出力に影響を及ぼさないように、余分なビットをバイパスするために難読化されるべき変換を設計しなおすことを意味する。スピリットパス符号化は、更なる出力ビットを提供して本質的な情報ビットを難読化するために、ある機能が変更されることを意味する。出力スピリッティングは、ある機能の出力が幾つかの部分的な機能を通して分散されることを意味し、この場合、全ての部分的な機能の出力は、その機能のオリジナルの出力を得るために明白でないやり方で結合され

る必要がある。

【００６７】

Chow2は、３２ビット又は更には９６ビットのワイドな入力をもつＳボックスを構築するためにビルディングエンコードネットワークを提案する。アフィン変換を表す係るワイド入力のＳボックスは、Ｓボックスのネットワークに分割され、それぞれ更に狭い入力及び出力を有し、Ｓボックスのそれぞれは、Ｓボックスにおける符号化機能を組み込むことで符号化される。符号化機能の逆は、Ｓボックスの出力を処理するＳボックスに組み込まれる。

【００６８】

［ルックアップテーブル値によるコードの一体化］

本発明の態様では、デジタルデータの改竄に対する抵抗の分散を可能にする方法が提供される。データは、デジタルデータの受信機にとって利用可能な命令を含むコンピュータコードにより処理される必要がある。処理の目的は、データにより表現されるオーディオ／ビデオ信号のレンダリングである。処理は、暗号化、復号、圧縮、伸張又は他の処理を有する場合がある。本方法は、デジタルデータを処理する処理アルゴリズムの実現を有する複数の命令を構成することを含む。複数の命令は、たとえばディストリビュートされたコンテンツの成功した再生のためにユーザ端末で必要とされるプラグイン又はメディアプレーヤといった、コンピュータプログラムを形成する。処理アルゴリズムは、パラメータに基づく。復号のケースでは、パラメータは暗号化鍵を表す。ディストリビュートされたデータは、対応する暗号化鍵を使用して（部分的に）暗号化される場合がある。

【００６９】

プロセッサの命令の一部に等しいパラメータの一部が識別される。より詳細には、パラメータのビット表現の一部は、命令のビット表現の一部に等しい。パラメータの残りの一部は、プロセッサの命令とは異なる。識別されたパラメータは、処理アルゴリズムの実現に含まれるプロセッサ命令に等しい。しかし、識別されたパラメータは、システムの何処かに現れるプロセッサ命令のビット表現に等しい場合もある。たとえば、識別されたパラメータは、オペレーティングシステムのカーネルで現れる幾つかの特定のビットストリング、又はＴＣＰ／ＩＰ通信スタックのようなシステムの幾つかのドライバで現れるビットに等しい。

【００７０】

プロセッサ命令は、実行の間、パラメータの等しい部分がプロセッサ命令のビット表現の一部のメモリ位置を参照することで（たとえば復号プロセスにおける使用のために）読取られるように構成される。一致するプロセッサ命令を保持するメモリアドレスに必要とされるビット表現が既に存在するとき、パラメータの等しい部分は、メモリに個別に記憶されない。効果的に、同じメモリアドレスは、復号アルゴリズムへのパラメータの記憶位置、及び同時に実行されるべきプロセッサ命令の記憶位置といった２つの方法で使用される。典型的に、パラメータとしてメモリ位置を読取る命令は、異なるメモリ位置に記憶され、参照によりパラメータを保持するメモリ位置にアクセスする。この命令は、プログラムコードを形成する。この命令は、プログラムコードが実行される実行環境に準拠する。たとえば、これらの命令は、プロセッサ命令であるか又は仮想マシン命令（たとえばjava bytecode）のような擬似コード命令である。

【００７１】

本発明の別の態様では、比較的大きな等しい部分を含むようにパラメータが選択される。単一のメモリアドレスは、プロセッサ命令として及びパラメータ値として、２つの一見したところ関連しない方法で使用することができるデータを保持する。これは、アタッカーがプロセッサ命令を変更する場合、パラメータが無効になり、逆に、アタッカーがパラメータを変更する場合、プロセッサ命令が無効になるという効果を有する。目的に指示された改竄を実行することは、アタッカーにとって更に困難になる。

【００７２】

図４は、実現を構成するステップ（ＣＯＭＰ　ＩＮＳＴＲ）６０３、パラメータの一致

する部分を識別するステップ（ＩＤＥＮＴ　ＥＱ　ＰＡＲＳ）６１３、及び実現をアレンジするステップ（ＡＲＲ　ＩＮＳＴＲ）６１５を示す。パラメータは、１以上のルックアップテーブルを含み、たとえばルックアップテーブルのネットワークを形成する。ルックアップテーブルの係るネットワークは、たとえば暗号化鍵から計算される（ステップ６０４（ＣＯＭＰ　ＬＵＴ））。プロセッサ命令のビット表現のワードは、ネットワークにおけるルックアップテーブルの少なくとも１つに含まれるために選択される（ステップ６０６（ＳＥＬ　ＷＯＲＤ））。ワードの包含は、ルックアップテーブルのエレメントに変換を適用することで実現される（ステップ６０８（ＴＲＡＮＳＦ　ＬＵＴ））。この変換は、変換の作用とは逆の作用である相補的な変換を他のルックアップテーブルの少なくとも１つのエレメントに適用することで補償される（ステップ６１０（ＣＯＭＰＥＮＳ　ＴＲＡＮＳ））。通常、少なくとも２つの変換されたルックアップテーブルは、ルックアップテーブルのネットワークを介して接続される。ルックアップテーブルの変換されたネットワークは、ルックアップテーブルのオリジナルネットワークではなく、暗号化アルゴリズムのパラメータとして使用される。

【００７３】

実施の形態は、命令のビット表現の複数のワードを選択すること、「コードを含んでいる」ルックアップテーブルを形成するため、この命令のビット表現を含むルックアップテーブルを作成することを含む。コードを含んでいるルックアップテーブルは、データ処理プログラムへのパラメータを形成するルックアップテーブルのネットワークに含まれる。通常、コードを含むルックアップテーブルの作用は、ルックアップテーブルのネットワークに適切に選択されたルックアップテーブルを含むことで補償される。プログラムコードを形成する命令は、コードを含んでいるルックアップテーブルで現れる命令を保持するメモリアドレスがコードを含んでいるルックアップテーブルの値を読取るために使用されるようにアレンジされる。

【００７４】

［強制されたテーブルルックアップ］

実施の形態では、図３を参照して、第二のユニット９０２は、デジタルデータ９１４に暗号化されたコンテンツを含む暗号化手段を含む。挿入手段９１０は、処理手段９０６による復号のために暗号化されたデータストリームに計算された値を含める。処理手段９０６は、挿入された値を含む暗号化されたコンテンツを復号する復号手段を有する。復号手段は、上述された方法のうちの１つでルックアップテーブル９１６を使用する。好ましくは、復号アルゴリズムのホワイトボックスの実現が使用される。これは、Chow1及びChow2に類似したＡＥＳ又はＤＥＳ実現であるが、他の実現又は復号スキームが同様に使用可能である。

【００７５】

実施の形態では、処理手段は、予め決定された順序でデジタルデータを復号化するために構成される。先に処理されたデータに依存してデータは処理される。これを実現するための可能性のうちの１つは、暗号化方法である。これは、たとえば暗号をストリーミングすることに適用される。暗号文の代わりに（又は暗号文に加えて）平文による依存性が形成される暗号ブロックチェイン（ＣＢＣ）モードを使用する復号スキームにも適用される。

【００７６】

挿入手段は、正のテスト結果が、データの重要なブロックの適切な処理のために必要とされるように、データの重要なブロックの前に予め定義されたルックアップテーブルのエントリをテストする値を配置するために構成される。たとえば、それぞれが異なる予め決定されたルックアップテーブルのエントリをテストする一連の値は、データストリームの開始で挿入される。このように、テストされたエントリの何れかへの変更は、完全なデータストリームを使用不可能にする。

【００７７】

メモリにおけるキー及びコードの一体化は、改竄に対する抵抗を増加するために行なわ

れる。一般に処理されるべきデータは、どのルックアップテーブルのエントリが処理において使用されるかを判定する。したがって、コードを改竄した後でさえ、改竄されたメモリ値がデータを処理するために必要とされないため、処理の比較的大きな部分が上手く実行されることが生じる。たとえば、１バイトが８ビットにおいてｋビット（ｋ＞０）のルックアップテーブルに変更され、唯一の８ビット値がコードのブロックを処理するために必要とされ、２５６の８ビット値のそれぞれが等しい発生確率を有する場合、変更されたバイトが処理で使用される確率は１／２５６又は０．４％である。この確率を増加すること、すなわち処理が失敗する確率を増加すること、及び／又はコードの１以上のバイトの変更後にデータの大きな部分について処理が失敗することを保証することが望まれる。また、コードで一体化されてないが、他の特別の意味を有する予め定義されたルックアップテーブルエントリにアクセスすることが望まれることが理解されるであろう。また、所定の値が変更されていることが疑われる場合、その値をチェックすることが望まれる。

【００７８】

Ｘをホワイトボックスの実現により処理されるべき（たとえば暗号化又は復号されるべき）データブロックの（おそらく空の）ストリームであるとする。ホワイトボックスの実現において特定のルックアップテーブルのエントリ（すなわち特定のキーの部分）にアクセスするやり方で選択された多数のブロックB_1，B_2，．．．，B_mをＸに挿入することが可能である。また、ブロックB_1，B_2，．．．，B_mによりＸを先行することも可能である。

【００７９】

コードの１以上のバイトの変更後にデータの大きな部分について処理が失敗する確率を増加するアプローチが以下に示される。誤った復号が更なる復号を通して伝播するブロック暗号モードを選択することができる。相対的に、それぞれ暗号ブロックを他の暗号ブロックとは独立に復号することができる「エレクトロニッククックブック（ＥＣＢ）」では、それぞれのブロックは、改善されたルックアップテーブルのエントリにアクセスする独立の確率（先の例では０．４％）を有する。たとえば、前に復号されたブロックの復号結果に依存してブロックの復号プロセスを行うことで、誤った復号化が行われ、次の復号に更に伝播する。復号結果が誤りであるデータの最初のブロックの後、全ての後続のブロックも誤って復号される。これにより、コードのテストされた部分が改竄されている場合に、データの非常に大きな部分を正しく処理することができない。

【００８０】

実施の形態では、（上述された理由のため）エラーを伝播する幾つかの種類のモードにおいてホワイトボックスの実現が使用される。たとえば、暗号化文の代わりに平文による依存性が形成される暗号ブロックチェイン（ＣＢＣ）モードの変形が使用される。たとえば、「ノーマル」ＣＢＣモードでは、データブロックｉは、それを暗号化されたデータブロックｉ－１とＸＯＲした後に暗号化される。「提案される」変形では、データブロックｉは、それを暗号化されていない（平文）データブロックｉ－１とＸＯＲした後に暗号化される。「ノーマル」ＣＢＣモードは、データストリームを更にランダムにする。「提案される」変形例により、単一の復号エラーが全ての後続するデータブロックに伝播される。暗号化されるべきコンテンツをもつストリームＳは、特定のルックアップテーブルのエントリ、特にコードを含むエントリにアクセスするのを目的とする多数のブロックにより先行される。一体化されたコードにおける１以上のビットが変更された場合、Ｓのブロックの何れもが正しく復号されない。また、前のブロックの平文と暗号化文の両者による依存性を形成する（たとえばＸＯＲする）ことで、「ノーマル」ＣＢＣモードを「提案される」ＣＢＣモードと結合することも可能である。このように、両方のモード（ランダム性とエラー伝搬性）の可能性のある利点が結合される。

【００８１】

データブロックは、特定のルックアップテーブルのエントリが復号プロセスの間（又は暗号化プロセスの間、規定通りに）にアクセスされるように計算される。ルックアップテーブルを使用してＡＥＳ復号（又は暗号化）アルゴリズムの標準的な（ホワイトボックス

ではない）実現を考える。さらに、この実現について以下の問題を考える。ラウンドｒ及びこのラウンドに対する入力I_rが与えられる。ラウンドｒへの入力がI_rであるように、復号アルゴリズムの最初のラウンドへの入力I_1を発見する。この問題のソリューションにより、当業者は、特定のルックアップテーブルのエントリが復号アルゴリズムの予め定義されたホワイトボックスではない実現によりアクセスされるようにデータブロックを設計することができることは明らかである。f_iを標準（ホワイトボックスではない）ＡＥＳのラウンドｉで計算された関数であるとし、すなわちf_i（I_i）は、その入力がI_iにより与えられた場合にラウンドｉの出力である。ラウンドf_iの逆関数f_i^{-1}を計算することは容易である。これは、f_iの計算における全てのステップ（AddRoundKey, SubBytes, ShiftRows, MixColumns）を逆にすることが容易であるためである。結果として、I_iは、以下のように計算される。

【００８２】

【数７】

$$I_1 = f_1^{-1} \circ f_2^{-1} \circ \ldots \circ f_{r-1}^{-1}(I_r)$$

このアルゴリズムは、たとえばタイプＩＩのテーブルＴにおける特定の行ｌにアクセスするためにホワイトボックスの復号アルゴリズムを実施するデータブロックを導出するために変更される。変更されたアルゴリズムは、行ｌがアクセスされるようにラウンドを含むテーブルＴへの（符号化された）入力を導出することで開始する。符号化は、ＡＥＳの（ホワイトボックスではない）実現のラウンドへの入力を得るため、この入力から除かれる。前のパラグラフで概説されたアルゴリズムは、ホワイトボックスの実現が行ｌにアクセスするデータブロックを導出するために使用される。一般に、符号化は、符号化された入力から、特定のルックアップテーブルの行ｌにアクセスさせるルックアップテーブルのネットワークにおける特定のルックアップテーブルに移動され、処理は、符号化なしに処理アルゴリズムのバージョンを使用して反転される場合がある。符号化のない処理ステップは、ホワイトボックスの実現の符号化されたルックアップテーブルよりも逆の処理を行なうことが容易である。符号化は、符号化の情報を有さないアタッカーによってではなく、符号化の情報を有する人物又はシステムによってのみ取り除くことができる。

【００８３】

なお、ネットワークにおける１つの特定のテーブルの１つの特定の行ｌにアクセスするデータブロックを導出する代わりに、ルックアップテーブルのネットワークにおける複数のそれぞれのルックアップテーブルのそれぞれにおける予め決定されたエントリにプログラムにアクセスさせるデータブロックを導出することができる。この理由は、ブロックの入力ビットが複数のテーブルを通して分散されるためであり、したがって、それぞれ予め決定されたルックアップテーブルのエントリにアクセスするため、それぞれのテーブルに分散される複数のビットを選択することができる。

【００８４】

また、ラウンドｒにおける特定の行にアクセスするために定義されるデータブロックは他のラウンドにおける行にもアクセスする。これは、そのラウンドは、ルックアップテーブルのネットワークの一部であるためである。これらの更にアクセスされた行の幾つかもコードにより一体化される。結果として、コードで一体化された所定数のバイトがアルゴリズムによりアクセスされることを達成するため、ブロックが含まれる必要がない。

【００８５】

説明されたＣＢＣモードに加えて、カウンタ（ＣＴＲ）モードも使用される。係るモードでは、データではなく、カウンタ（たとえば擬似ランダム系列の値）が暗号化される。データは、暗号化されたカウンタのストリームとＸＯＲされる。このモードについて、カウンタストリームに（予め決定されたテーブルルックアップに対応する）所望の値を含めるため、カウンタ値を選択する自由度が使用される。たとえば、カウンタ値のセットは、データストリームの開始近くで生じるようにされ、これにより、一体化されたテーブルエントリの大きな部分がアクセスされる。カウンタストリームの暗号化が「提案される」Ｃ

ＢＣモードで再び実行された場合、一体化されたコードにおける変化は、データストリームの大きな部分の処理にとって致命的である。ここでのＣＴＲモードを使用する利点は、選択されたカウンタ値の暗号化の結果が有効な平文を得るために暗号化されたデータとＸＯＲされるとき、余分の復号／暗号化が存在しないことである。

【００８６】

また、本方法は、２値画像の検証のために使用される。ルックアップテーブルにおける全てのエントリに迅速にアクセスすることで、ルックアップテーブルにおける値が正しいかが迅速に検証される。加えて、リアルデータを処理する必要がない。はじめに、全てのＳボックスがアクセスされることを実施するデータブロックのセットが導出される。これは、記載されるアルゴリズムで行なうことができる。つぎに、このセットにおける全てのデータブロックについて、ルックアップテーブルが正しい回答を与えるかがテストされる。正しい回答を与える場合、二値画像は恐らく正しい。上述された暗号ブロックチェインモードを使用した場合、全ての前の動作が正しかった場合にのみ正しいため、最後の結果のみが検証される必要がある。

【００８７】

図５は、本発明の実施の形態を例示する。この図は、デジタルコンテンツのプロバイダと接続するため、インターネットへの接続のような通信ポート（ＣＯＭ）９５を示す。また、コンテンツは、ＤＶＤ又はＣＤのようなメディア９６から取得される。ＰＣのデジタルコンテンツは、メモリ（ＭＥＭ）９１を使用してプロセッサ（ＰＲＯＣ）９２により実行されるメディアプレーヤを使用して典型的にレンダリングされる。係るプレーヤは、特定のコンテンツフォーマットについて、通信ポート９５及び／又は媒体９６を介して得られるコンテンツに対応するフォーマットに特化した復号化を実行するためにそれぞれのプラグインを実行する。それらのコンテンツフォーマットは、ＡＶＩ，ＤＶ，Ｍｏｔｉｏｎ　ＪＰＥＧ，ＭＰＥＧ－１，ＭＰＥＧ－２，ＭＰＥＧ－４，ＷＭＶ，Ａｕｄｉｏ　ＣＤ，ＭＰ３，ＷＭＡ，ＷＡＶ，ＡＩＦＦ／ＡＩＦＣ，ＡＵ等を含む。デジタル著作権の管理の目的について、コンテンツをデコードするだけでなく、コンテンツを復号するセキュアなプラグインが使用される。このプラグインは、メモリ９１に記憶されるプロセッサ命令及び（難読化されたルックアップテーブルのような）パラメータを含む。プロセッサ命令及びパラメータは、上述されたようにオーバラップしており、そのケースでは、メモリ９１における幾つかのメモリ位置は、プラグインの実行の間のプロセッサ命令とパラメータ値の両者を表す値を含む。コンテンツでは、幾つかの予め決定されたメモリ位置がそれらのデコード／復号の間にルックアップテーブルのエントリとして使用されるのを確かめるため、データブロックが挿入される。たとえば、プロセッサ命令及びパラメータ値の両者を表すメモリ位置がアドレス指定される場合がある。ユーザ入力（ＩＮＰ）９４は、コンテンツがレンダリングされるべきことを示すためにユーザからのコマンドを取得するために提供され、ディスプレイ（ＤＩＳＰ）９３及び／又はスピーカは、デコードされたコンテンツ及び／又は復号されたコンテンツをレンダリングするために提供される。

【００８８】

本発明は、コンピュータプログラム、本発明を実施するために適合される、特にキャリア上のコンピュータプログラム又はキャリアにおけるコンピュータプログラムにも拡張することができる。プログラムは、ソースコード、オブジェクトコード、コードインターミディエートソース及び部分的にコンパイルされた形式のようなオブジェクトコードの形式、或いは、本発明に係る方法の実現における使用向けに適した他の形式である。キャリアは、プログラムを伝送可能なエンティティ又は装置である場合がある。たとえば、キャリアは、たとえばＣＤ　ＲＯＭといったＲＯＭのような記憶媒体、又は半導体メモリ、或いは、たとえばフロプティカルディスク又はハードディスクといった磁気記録媒体を含む。さらに、キャリアは、電気又は光ケーブルを介して、或いは無線又は他の手段により伝達される電気又は光信号のような伝送キャリアである。プログラムが係る信号で実施されるとき、キャリアは、係るケーブル或いは他の装置又は手段により構成される。代替的に、キャリアは、プログラムが埋め込まれる集積回路であり、集積回路は、関連する方法を実

行するか、関連する方法のパフォーマンスにおける使用のため適合される。

【００８９】

上述された実施の形態は本発明を限定するのではなく例示するものであり、当業者であれば、特許請求の範囲から逸脱することなしに多くの代替的な実施の形態を設計することができる。請求項では、括弧間に配置される参照符号は請求項を限定するものとして解釈されるべきではない。動詞「有する」及びその派生語の使用は、請求項で述べた以外のエレメント又はステップの存在を排除するものではない。エレメントに先行する冠詞“ａ”又は“ａｎ”は、複数の係るエレメントの存在を排除するものではない。本発明は、幾つかの固有のエレメントを有するハードウェアにより、適切にプログラムされたコンピュータにより実現される。幾つかの手段を列挙する装置の請求項では、これらの手段の幾つかは、同一アイテムのハードウェアにより実施される。所定の手段が相互に異なる従属の請求項で引用される事実は、これらの手段の組み合わせを利用することができないことを示すものではない。

【図１】

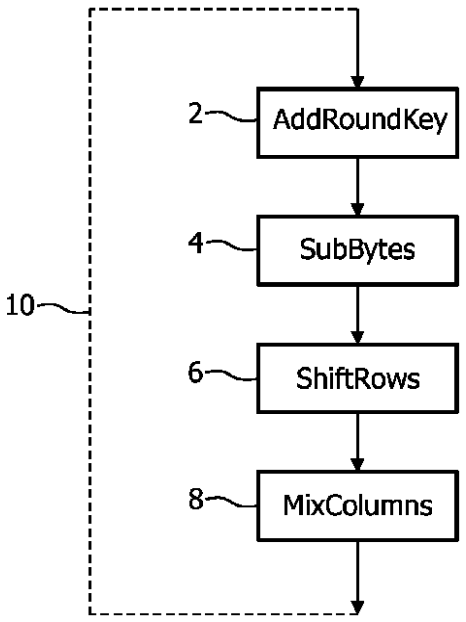

FIG. 1

【図２】

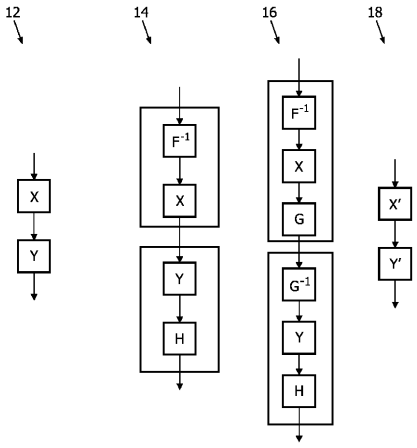

FIG. 2

【図３】

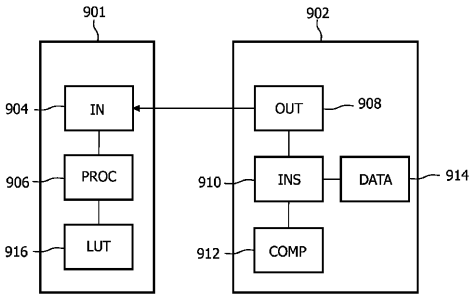

FIG. 3

【図４】

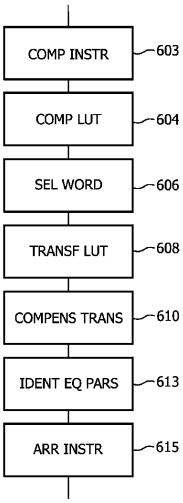

FIG. 4

【図５】

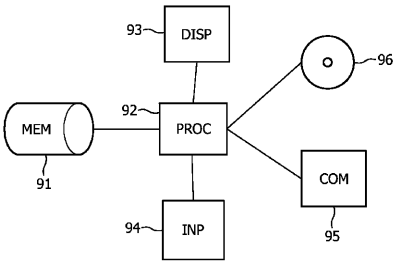

FIG. 5

フロントページの続き

(72)発明者 ミヒールス，ウィルヘルムス，ペー．，アー．，イェー．
オランダ国，５６５６ アーエー アインドーフェン，ハイ・テク・キャンパス ４４
(72)発明者 ホリッセン，パウルス，エム．，ハー．，エム．，アー．
オランダ国，５６５６ アーエー アインドーフェン，ハイ・テク・キャンパス ４４

審査官 青木 重徳

(56)参考文献 特表２００９－５２９８１９（ＪＰ，Ａ）
特開２００６－０７９３４７（ＪＰ，Ａ）
特開２００５－３３１６５６（ＪＰ，Ａ）
特開平１０－１５４９７６（ＪＰ，Ａ）
特開平０７－３１２５９３（ＪＰ，Ａ）
特表２００２－５１４３３３（ＪＰ，Ａ）
米国特許出願公開第２００６／０１４０４０１（ＵＳ，Ａ１）
Hamilton E. Link and William D. Neumann, “Clarifying Obfuscation: Improving the Security of White-Box DES”, International Conference on Information Technology: Coding and Computing (ITCC 2005), [online], ２００５年 ４月 ４日, Vol.1, p.679-684, [retrieved on 2012-09-11]. Retrieved from the Internet, ＵＲＬ, <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1428542>
Julien BRINGER, Herve CHABANNE, Emmanuelle DOTTAX, “Perturbing and Protecting a Traceable Block Cipher”, Cryptology ePrint Archive: Report 2006/064, [online], ２００６年 ２月２０日, Version: 20060223:223232, p.1-12, [retrieved on 2012-09-19]. Retrieved from the Internet, ＵＲＬ, <http://eprint.iacr.org/2006/064.pdf>
Olivier Billet, Henri Gilbert, and Charaf Ech-Chatbi, “Cryptanalysis of a White Box AES Implementation”, LNCS, Selected Areas in Cryptography, ２００４年 ８月, Vol.3357, pp.227-240

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｈ０４Ｌ ９／３２
Ｇ０９Ｃ １／００
ＪＳＴＰｌｕｓ／ＪＭＥＤＰｌｕｓ／ＪＳＴ７５８０（ＪＤｒｅａｍＩＩＩ）