(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4710132号
(P4710132)
(45)発行日　平成23年6月29日(2011.6.29)　(24)登録日　平成23年4月1日(2011.4.1)

(51)Int.Cl.　　F I
H04L 9/08 (2006.01)　　H04L 9/00 601B
G06F 21/24 (2006.01)　　G06F 12/14 540A

請求項の数 10　(全 76 頁)

(21)出願番号　特願2000-396098(P2000-396098)
(22)出願日　平成12年12月26日(2000.12.26)
(65)公開番号　特開2002-198952(P2002-198952A)
(43)公開日　平成14年7月12日(2002.7.12)
審査請求日　平成19年12月21日(2007.12.21)

(73)特許権者　000002185
ソニー株式会社
東京都港区港南１丁目７番１号
(74)代理人　100101801
弁理士　山田　英治
(74)代理人　100093241
弁理士　宮田　正昭
(74)代理人　100086531
弁理士　澤田　俊夫
(72)発明者　浅野　智之
東京都品川区北品川６丁目７番３５号　ソニー株式会社内
(72)発明者　大澤　義知
東京都品川区北品川６丁目７番３５号　ソニー株式会社内

最終頁に続く

(54)【発明の名称】情報処理システム、および情報処理方法、並びにプログラム記録媒体

(57)【特許請求の範囲】
【請求項１】
複数のデバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上の下位キーによる上位キーの暗号化処理データを有し、前記選択パスに対応するノードキーセットを利用可能なデバイスにおいてのみ復号可能とした有効化キーブロック（ＥＫＢ）をデバイスに提供する構成を持つ情報処理システムであり、
前記キーツリーは、カテゴリに基づいて区分され、カテゴリ管理装置によって管理されるサブツリーとしてのカテゴリツリーを複数有する構成であり、
カテゴリツリーにおいて共通に復号処理可能なＥＫＢを生成して発行するキー発行センター（ＫＤＣ）の情報処理装置は、
ＥＫＢタイプ識別子と、該ＥＫＢタイプ識別子によって特定されるＥＫＢの復号処理が可能なデバイス群の設定されたカテゴリツリーの識別データとの対応データを登録したＥＫＢタイプ定義リストを有し、
前記ＥＫＢタイプ定義リストに登録された登録カテゴリツリーの状態変化として、
（ａ）登録カテゴリツリーにおけるリボーク（機器排除）発生に伴う状態変化、または、
（ｂ）登録カテゴリツリーに属するデバイスのデバイス格納キーの変更に伴う状態変化、

上記（ａ），（ｂ）の少なくともいずれかの登録カテゴリツリーの状態変化に関する通知処理を、
前記ＥＫＢタイプ定義リストに前記登録カテゴリツリーの識別データの対応データとして登録されたＥＫＢタイプ識別子によって特定されるＥＫＢを利用するＥＫＢ利用装置に対して実行する構成を有することを特徴とする情報処理システム。
【請求項２】
前記ＥＫＢ利用装置は、前記キー発行センター（ＫＤＣ）の情報処理装置に対するＥＫＢ生成要求装置としてのＥＫＢリクエスタを含むことを特徴とする請求項１に記載の情報処理システム。
【請求項３】
前記ＥＫＢ利用装置は、前記ＥＫＢタイプ定義リスト中に定義されたＥＫＢタイプ識別子によって特定されるＥＫＢの復号処理が可能なデバイス群の設定されたカテゴリツリーの管理処理を実行するカテゴリ管理装置を含むことを特徴とする請求項１に記載の情報処理システム。
【請求項４】
前記キー発行センター（ＫＤＣ）の情報処理装置は、
前記状態変化発生に関する通知処理を、前記ＥＫＢタイプ定義リストの利用装置である、
前記キー発行センター（ＫＤＣ）の情報処理装置に対するＥＫＢ生成要求装置としてのＥＫＢリクエスタ、および、
カテゴリツリーの管理処理を実行するカテゴリ管理装置、
に対して実行することを特徴とする請求項１に記載の情報処理システム。
【請求項５】
前記キー発行センター（ＫＤＣ）の情報処理装置は、
カテゴリツリーにおける状態変化発生情報を、該カテゴリツリーの管理装置であるカテゴリ管理装置から受信し、
該カテゴリ管理装置からの状態変化発生情報受信に基づいて、状態変化発生に関する通知処理を実行する構成を有することを特徴とする請求項１に記載の情報処理システム。
【請求項６】
カテゴリに基づいて区分され、カテゴリ管理装置によって管理されるサブツリーとしてのカテゴリツリーを複数有し、デバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上の下位キーによる上位キーの暗号化処理データを有し、前記選択パスに対応するノードキーセットを利用可能なデバイスにおいてのみ復号可能とした有効化キーブロック（ＥＫＢ）をデバイスに提供する構成を持つシステムにおける情報処理方法であり、
カテゴリツリーにおいて共通に復号処理可能なＥＫＢを生成して発行するキー発行センター（ＫＤＣ）の情報処理装置は、
ＥＫＢタイプ識別子と、該ＥＫＢタイプ識別子によって特定されるＥＫＢの復号処理が可能なデバイス群の設定されたカテゴリツリーの識別データとの対応データを登録したＥＫＢタイプ定義リストに登録された登録カテゴリツリーの状態変化として、
（ａ）登録カテゴリツリーにおけるリボーク（機器排除）発生に伴う状態変化、または、
（ｂ）登録カテゴリツリーに属するデバイスのデバイス格納キーの変更に伴う状態変化、
上記（ａ），（ｂ）の少なくともいずれかの登録カテゴリツリーの状態変化に関する通知処理を、
前記ＥＫＢタイプ定義リストに前記登録カテゴリツリーの識別データの対応データとして登録されたＥＫＢタイプ識別子によって特定されるＥＫＢを利用するＥＫＢ利用装置に対して実行することを特徴とする情報処理方法。

【請求項７】
前記ＥＫＢ利用装置は、前記キー発行センター（ＫＤＣ）の情報処理装置に対するＥＫＢ生成要求装置としてのＥＫＢリクエスタを含むことを特徴とする請求項６に記載の情報処理方法。
【請求項８】
前記ＥＫＢ利用装置は、前記ＥＫＢタイプ定義リスト中に定義されたＥＫＢタイプ識別子によって特定されるＥＫＢの復号処理が可能なデバイス群の設定されたカテゴリツリーの管理処理を実行するカテゴリ管理装置を含むことを特徴とする請求項６に記載の情報処理方法。
【請求項９】
前記キー発行センター（ＫＤＣ）の情報処理装置は、
前記状態変化発生に関する通知処理を、前記ＥＫＢタイプ定義リストの利用装置である、
前記キー発行センター（ＫＤＣ）の情報処理装置に対するＥＫＢ生成要求装置としてのＥＫＢリクエスタ、および、
カテゴリツリーの管理処理を実行するカテゴリ管理装置、
に対して実行することを特徴とする請求項６に記載の情報処理方法。
【請求項１０】
前記キー発行センター（ＫＤＣ）の情報処理装置は、
カテゴリツリーにおける状態変化発生情報を、該カテゴリツリーの管理装置であるカテゴリ管理装置から受信し、
該カテゴリ管理装置からの状態変化発生情報受信に基づいて、状態変化発生に関する通知処理を実行する構成を有することを特徴とする請求項６に記載の情報処理方法。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】
本発明は、情報処理システム、および情報処理方法、並びにプログラム記録媒体に関し、特に、コンテンツなど各種データを特定の正当なユーザに提供する暗号処理を伴う配信システムおよび方法に関する。特に、木（ツリー）構造の階層的鍵配信方式を用い、配信デバイスに応じて生成したキーブロックを用いて、例えばコンテンツの暗号化キーとしてのコンテンツキー配信、あるいはその他各種の安全性を保持することを可能とする情報処理システム、および情報処理方法、並びにプログラム記録媒体に関する。
【０００２】
【従来の技術】
昨今、ゲームプログラム、音声データ、画像データ等、様々なソフトウエアデータ（以下、これらをコンテンツ（Content）と呼ぶ）を、インターネット等のネットワーク、あるいはＤＶＤ、ＣＤ等の流通可能な記憶媒体を介しての流通が盛んになってきている。これらの流通コンテンツは、ユーザの所有するＰＣ（Personal Computer）、ゲーム機器によってデータ受信、あるいは記憶媒体の装着がなされて再生されたり、あるいはＰＣ等に付属する記録再生機器内の記録デバイス、例えばメモリカード、ハードディスク等に格納されて、格納媒体からの新たな再生により利用される。
【０００３】
ビデオゲーム機器、ＰＣ等の情報機器には、流通コンテンツをネットワークから受信するため、あるいはＤＶＤ、ＣＤ等にアクセスするためのインタフェースを有し、さらにコンテンツの再生に必要となる制御手段、プログラム、データのメモリ領域として使用されるＲＡＭ、ＲＯＭ等を有する。
【０００４】
音楽データ、画像データ、あるいはプログラム等の様々なコンテンツは、再生機器として利用されるゲーム機器、ＰＣ等の情報機器本体からのユーザ指示、あるいは接続された入力手段を介したユーザの指示により記憶媒体から呼び出され、情報機器本体、あるいは接

続されたディスプレイ、スピーカ等を通じて再生される。
【０００５】
ゲームプログラム、音楽データ、画像データ等、多くのソフトウエア・コンテンツは、一般的にその作成者、販売者に頒布権等が保有されている。従って、これらのコンテンツの配布に際しては、一定の利用制限、すなわち、正規なユーザに対してのみ、ソフトウエアの使用を許諾し、許可のない複製等が行われないようにする、すなわちセキュリティを考慮した構成をとるのが一般的となっている。
【０００６】
ユーザに対する利用制限を実現する１つの手法が、配布コンテンツの暗号化処理である。すなわち、例えばインターネット等を介して暗号化された音声データ、画像データ、ゲームプログラム等の各種コンテンツを配布するとともに、正規ユーザであると確認された者に対してのみ、配布された暗号化コンテンツを復号する手段、すなわち復号鍵を付与する構成である。
【０００７】
暗号化データは、所定の手続きによる復号化処理によって利用可能な復号データ（平文）に戻すことができる。このような情報の暗号化処理に暗号化鍵を用い、復号化処理に復号化鍵を用いるデータ暗号化、復号化方法は従来からよく知られている。
【０００８】
暗号化鍵と復号化鍵を用いるデータ暗号化・復号化方法の態様には様々な種類あるが、その１つの例としていわゆる共通鍵暗号化方式と呼ばれている方式がある。共通鍵暗号化方式は、データの暗号化処理に用いる暗号化鍵とデータの復号化に用いる復号化鍵を共通のものとして、正規のユーザにこれら暗号化処理、復号化に用いる共通鍵を付与して、鍵を持たない不正ユーザによるデータアクセスを排除するものである。この方式の代表的な方式にＤＥＳ（データ暗号標準：Deta encryption standard）がある。
【０００９】
上述の暗号化処理、復号化に用いられる暗号化鍵、復号化鍵は、例えばあるパスワード等に基づいてハッシュ関数等の一方向性関数を適用して得ることができる。一方向性関数とは、その出力から逆に入力を求めるのは非常に困難となる関数である。例えばユーザが決めたパスワードを入力として一方向性関数を適用して、その出力に基づいて暗号化鍵、復号化鍵を生成するものである。このようにして得られた暗号化鍵、復号化鍵から、逆にそのオリジナルのデータであるパスワードを求めることは実質上不可能となる。
【００１０】
また、暗号化するときに使用する暗号化鍵による処理と、復号するときに使用する復号化鍵の処理とを異なるアルゴリズムとした方式がいわゆる公開鍵暗号化方式と呼ばれる方式である。公開鍵暗号化方式は、不特定のユーザが使用可能な公開鍵を使用する方法であり、特定個人に対する暗号化文書を、その特定個人が発行した公開鍵を用いて暗号化処理を行なう。公開鍵によって暗号化された文書は、その暗号化処理に使用された公開鍵に対応する秘密鍵によってのみ復号処理が可能となる。秘密鍵は、公開鍵を発行した個人のみが所有するので、その公開鍵によって暗号化された文書は秘密鍵を持つ個人のみが復号することができる。公開鍵暗号化方式の代表的なものにはＲＳＡ（Rivest-Shamir-Adleman）暗号がある。このような暗号化方式を利用することにより、暗号化コンテンツを正規ユーザに対してのみ復号可能とするシステムが可能となる。
【００１１】
【発明が解決しようとする課題】
上記のようなコンテンツ配信システムでは、コンテンツを暗号化してユーザにネットワーク、あるいはＤＶＤ、ＣＤ等の記録媒体に格納して提供し、暗号化コンテンツを復号するコンテンツキーを正当なユーザにのみ提供する構成が多く採用されている。コンテンツキー自体の不正なコピー等を防ぐため、コンテンツキーを暗号化して正当なユーザに提供し、正当なユーザのみが有する復号キーを用いて暗号化コンテンツキーを復号してコンテンツキーを使用可能とする構成が提案されている。

【００１２】
正当なユーザであるか否かの判定は、一般には、例えばコンテンツの送信者であるコンテンツプロバイダとユーザデバイス間において、コンテンツ、あるいはコンテンツキーの配信前に認証処理を実行することによって可能となる。一般的な認証処理においては、相手の確認を行なうとともに、その通信でのみ有効なセッションキーを生成して、認証が成立した場合に、生成したセッションキーを用いてデータ、例えばコンテンツあるいはコンテンツキーを暗号化して通信を行なう。認証方式には、共通鍵暗号方式を用いた相互認証と、公開鍵方式を使用した認証方式があるが、共通鍵を使った認証においては、システムワイドで共通な鍵が必要になり、更新処理等の際に不便である。また、公開鍵方式においては、計算負荷が大きくまた必要なメモリ量も大きくなり、各デバイスにこのような処理手段を設けることは望ましい構成とはいえない。
【００１３】
本発明では、上述のようなデータの送信者、受信者間の相互認証処理に頼ることなく、正当なユーザに対してのみ、安全にデータを送信することを可能とするとともに、階層的鍵配信ツリーをカテゴリ単位としたサブツリー、すなわちカテゴリツリーを形成し、複数のカテゴリツリー内で適用（復号処理）可能な暗号化キーブロックを使用する構成を提案する。
【００１４】
さらに、１以上の選択されたカテゴリツリーにおいて復号可能な暗号化鍵データブロックである有効化キーブロック（ＥＫＢ）を生成して各カテゴリツリーに属するデバイスにおいて共通に使用可能とするとともに、どのカテゴリツリーで処理可能、すなわち復号可能であるかを示すＥＫＢタイプ定義リストを使用することにより、ＥＫＢ生成、管理処理の効率化を可能とした情報処理システム、および情報処理方法、並びにプログラム記録媒体を提供することを目的とする。
【００１５】
さらに、ＥＫＢタイプ定義リストに定義されたＥＫＢの処理可能なカテゴリツリーにおける状態変化発生に関する通知処理を、ＥＫＢの利用エンティティに対して実行することにより、最新のＥＫＢタイプ定義情報に基づく処理を可能とした情報処理システム、および情報処理方法、並びにプログラム記録媒体を提供することを目的とする。
【００１６】
【課題を解決するための手段】
本発明の第１の側面は、
複数のデバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上の下位キーによる上位キーの暗号化処理データを有し、前記選択パスに対応するノードキーセットを利用可能なデバイスにおいてのみ復号可能とした有効化キーブロック（ＥＫＢ）をデバイスに提供する構成を持つ情報処理システムであり、
前記キーツリーは、カテゴリに基づいて区分され、カテゴリ・エンティティによって管理されるサブツリーとしてのカテゴリツリーを複数有する構成であり、カテゴリツリーにおいて共通に復号処理可能なＥＫＢを生成して発行するキー発行センター（ＫＤＣ）は、
ＥＫＢタイプ識別子と、ＥＫＢ処理可能なカテゴリツリーの識別データとを対応付けたＥＫＢタイプ定義リストを有し、
前記ＥＫＢタイプ定義リストに定義されたＥＫＢの処理可能なカテゴリツリーにおける状態変化発生に関する通知処理を、少なくとも前記状態変化発生カテゴリツリーを処理可能カテゴリツリーとして設定したＥＫＢの利用エンティティに対して実行する構成を有することを特徴とする情報処理システムにある。
【００１７】
さらに、本発明の情報処理システムの一実施態様において、前記カテゴリツリーにおける状態変化は、該カテゴリツリーにおけるリボーク（機器排除）発生に伴う状態変化である

ことを特徴とする。
【００１８】
さらに、本発明の情報処理システムの一実施態様において、前記カテゴリツリーにおける状態変化は、該カテゴリツリーに属するデバイスのデバイス格納キーの変更に伴う状態変化であることを特徴とする。
【００１９】
さらに、本発明の情報処理システムの一実施態様において、前記ＥＫＢの利用エンティテイは、前記キー発行センター（ＫＤＣ）に対するＥＫＢ生成要求エンティテイとしてのＥＫＢリクエスタを含むことを特徴とする。
【００２０】
さらに、本発明の情報処理システムの一実施態様において、前記ＥＫＢの利用エンティテイは、前記ＥＫＢタイプ定義リスト中に定義されたＥＫＢ処理可能なカテゴリツリーの管理エンティテイとしてのカテゴリ・エンティテイを含むことを特徴とする。
【００２１】
さらに、本発明の情報処理システムの一実施態様において、前記キー発行センター（ＫＤＣ）は、前記状態変化発生に関する通知処理を、前記ＥＫＢタイプ定義リストの利用エンティテイである前記キー発行センター（ＫＤＣ）に対するＥＫＢ生成要求エンティテイとしてのＥＫＢリクエスタ、およびカテゴリツリーの管理エンティテイとしてのカテゴリ・エンティテイすべてに対して実行することを特徴とする。
【００２２】
さらに、本発明の情報処理システムの一実施態様において、前記キー発行センター（ＫＤＣ）は、カテゴリツリーにおける状態変化発生情報を、該カテゴリツリーの管理エンティテイであるカテゴリ・エンティテイから受信し、該カテゴリ・エンティテイからの状態変化発生情報受信に基づいて、状態変化発生に関する通知処理を実行する構成を有することを特徴とする。
【００２３】
さらに、本発明の第２の側面は、
カテゴリに基づいて区分され、カテゴリ・エンティテイによって管理されるサブツリーとしてのカテゴリツリーを複数有し、デバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上の下位キーによる上位キーの暗号化処理データを有し、前記選択パスに対応するノードキーセットを利用可能なデバイスにおいてのみ復号可能とした有効化キーブロック（ＥＫＢ）をデバイスに提供する構成を持つシステムにおける情報処理方法であり、
カテゴリツリーにおいて共通に復号処理可能なＥＫＢを生成して発行するキー発行センター（ＫＤＣ）は、
ＥＫＢタイプ識別子と、ＥＫＢ処理可能なカテゴリツリーの識別データとを対応付けたＥＫＢタイプ定義リストに定義されたＥＫＢの処理可能なカテゴリツリーにおける状態変化発生に関する通知処理を、少なくとも前記状態変化発生カテゴリツリーを処理可能カテゴリツリーとして設定したＥＫＢの利用エンティテイに対して実行することを特徴とする情報処理方法にある。
【００２４】
さらに、本発明の情報処理方法の一実施態様において、前記カテゴリツリーにおける状態変化は、該カテゴリツリーにおけるリボーク（機器排除）発生に伴う状態変化であることを特徴とする。
【００２５】
さらに、本発明の情報処理方法の一実施態様において、前記カテゴリツリーにおける状態変化は、該カテゴリツリーに属するデバイスのデバイス格納キーの変更に伴う状態変化であることを特徴とする。
【００２６】

さらに、本発明の情報処理方法の一実施態様において、前記ＥＫＢの利用エンティテイは、前記キー発行センター（ＫＤＣ）に対するＥＫＢ生成要求エンティテイとしてのＥＫＢリクエスタを含むことを特徴とする。

【００２７】

さらに、本発明の情報処理方法の一実施態様において、前記ＥＫＢの利用エンティテイは、前記ＥＫＢタイプ定義リスト中に定義されたＥＫＢ処理可能なカテゴリツリーの管理エンティテイとしてのカテゴリ・エンティテイを含むことを特徴とする。

【００２８】

さらに、本発明の情報処理方法の一実施態様において、前記キー発行センター（ＫＤＣ）は、前記状態変化発生に関する通知処理を、前記ＥＫＢタイプ定義リストの利用エンティテイである前記キー発行センター（ＫＤＣ）に対するＥＫＢ生成要求エンティテイとしてのＥＫＢリクエスタ、およびカテゴリツリーの管理エンティテイとしてのカテゴリ・エンティテイすべてに対して実行することを特徴とする。

【００２９】

さらに、本発明の情報処理方法の一実施態様において、前記キー発行センター（ＫＤＣ）は、カテゴリツリーにおける状態変化発生情報を、該カテゴリツリーの管理エンティテイであるカテゴリ・エンティテイから受信し、該カテゴリ・エンティテイからの状態変化発生情報受信に基づいて、状態変化発生に関する通知処理を実行する構成を有することを特徴とする。

【００３０】

さらに、本発明の第３の側面は、

カテゴリに基づいて区分され、カテゴリ・エンティテイによって管理されるサブツリーとしてのカテゴリツリーを複数有し、デバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上の下位キーによる上位キーの暗号化処理データを有し、前記選択パスに対応するノードキーセットを利用可能なデバイスにおいてのみ復号可能とした有効化キーブロック（ＥＫＢ）をデバイスに提供する構成を持つシステムにおける情報処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを記録したプログラム記録媒体であって、前記コンピュータ・プログラムは、カテゴリツリーにおける状態変化発生情報を、該カテゴリツリーの管理エンティテイであるカテゴリ・エンティテイから受信するステップと、

該カテゴリ・エンティテイからの状態変化発生情報受信に基づいて、状態変化発生に関する通知処理を少なくとも前記状態変化発生カテゴリツリーを処理可能カテゴリツリーとして設定したＥＫＢの利用エンティテイに対して実行するステップと、

を有することを特徴とするプログラム記録媒体にある。

【００３１】

【作用】

本発明の構成においては、ツリー（木）構造の階層的構造の暗号化鍵配信構成を用い、各機器をｎ分木の各葉（リーフ）に配置した構成の鍵配信方法を用い、記録媒体もしくは通信回線を介して、例えばコンテンツデータの暗号鍵であるコンテンツキーもしくは認証処理に用いる認証キー、あるいはプログラムコード等を有効化キーブロックとともに配信する構成としている。

【００３２】

さらに、有効化キーブロックを暗号化キーデータ部と、暗号化キーの位置を示すタグ部てによって構成し、データ量を少なくし、デバイスにおける復号処理を用意かつ迅速に実行することを可能としている。本構成により、正当なデバイスのみが復号可能なデータを安全に配信することが可能となる。

【００３３】

さらに、１以上の選択されたカテゴリツリーにおいて復号可能な暗号化鍵データブロックである有効化キーブロック（ＥＫＢ）を生成して各カテゴリツリーに属するデバイスに共

通に使用可能とするとともに、どのカテゴリツリーで処理可能、すなわち復号可能であるかを示すＥＫＢタイプ定義リストを使用することにより、ＥＫＢの生成管理処理の効率化を可能としている。

【００３４】

なお、本発明のプログラム記録媒体は、例えば、様々なプログラム・コードを実行可能な汎用コンピュータ・システムに対して、コンピュータ・プログラムをコンピュータ可読な形式で提供する媒体である。媒体は、ＣＤやＦＤ、ＭＯなどの記録媒体、あるいは、ネットワークなどの伝送媒体など、その形態は特に限定されない。

【００３５】

このようなプログラム記録媒体は、コンピュータ・システム上で所定のコンピュータ・プログラムの機能を実現するための、コンピュータ・プログラムと記録媒体との構造上又は機能上の協働的関係を定義したものである。換言すれば、該記録媒体を介してコンピュータ・プログラムをコンピュータ・システムにインストールすることによって、コンピュータ・システム上では協働的作用が発揮され、本発明の他の側面と同様の作用効果を得ることができるのである。

【００３６】

なお、本発明の説明中におけるシステムとは、複数の装置の論理的集合構成であり、各構成の装置が同一筐体内にあるものには限らない。

【００３７】

本発明のさらに他の目的、特徴や利点は、後述する本発明の実施例や添付する図面に基づくより詳細な説明によって明らかになるであろう。

【００３８】

【発明の実施の形態】

［システム概要］

図１に本発明の情報処理システムが適用可能なコンテンツ配信システム例を示す。コンテンツの配信側１０は、コンテンツ受信側２０の有する様々なコンテンツ再生可能な機器に対してコンテンツ、あるいはコンテンツキーを暗号化して送信する。受信側２０における機器では、受信した暗号化コンテンツ、あるいは暗号化コンテンツキー等を復号してコンテンツあるいはコンテンツキーを取得して、画像データ、音声データの再生、あるいは各種プログラムの実行等を行なう。コンテンツの配信側１０とコンテンツ受信側２０との間のデータ交換は、インターネット等のネットワークを介して、あるいはＤＶＤ、ＣＤ等の流通可能な記憶媒体を介して実行される。

【００３９】

コンテンツ配信側１０のデータ配信手段としては、インターネット１１、衛星放送１２、電話回線１３、ＤＶＤ、ＣＤ等のメディア１４等があり、一方、コンテンツ受信側２０のデバイスとしては、パーソナルコンピュータ（ＰＣ）２１、ポータブルデバイス（ＰＤ）２２、携帯電話、ＰＤＡ（Personal Digital Assistants）等の携帯機器２３、ＤＶＤ、ＣＤプレーヤ等の記録再生器２４、ゲーム端末等の再生専用器２５等がある。これらコンテンツ受信側２０の各デバイスは、コンテンツ配信側１０から提供されたコンテンツをネットワーク等の通信手段あるいは、あるいはメディア３０から取得する。

【００４０】

［デバイス構成］

図２に、図１に示すコンテンツ受信側２０のデバイスの一例として、記録再生装置１００の構成ブロック図を示す。記録再生装置１００は、入出力Ｉ／Ｆ(Interface)１２０、ＭＰＥＧ(Moving Picture Experts Group)コーデック１３０、Ａ／Ｄ，Ｄ／Ａコンバータ１４１を備えた入出力Ｉ／Ｆ(Interface)１４０、暗号処理手段１５０、ＲＯＭ（Read Only Memory）１６０、ＣＰＵ(Central Processing Unit)１７０、メモリ１８０、記録媒体１９５のドライブ１９０を有し、これらはバス１１０によって相互に接続されている。

【００４１】

入出力Ｉ／Ｆ１２０は、外部から供給される画像、音声、プログラム等の各種コンテンツ

を構成するディジタル信号を受信し、バス１１０上に出力するとともに、バス１１０上のディジタル信号を受信し、外部に出力する。ＭＰＥＧコーデック１３０は、バス１１０を介して供給されるＭＰＥＧ符号化されたデータを、ＭＰＥＧデコードし、入出力Ｉ／Ｆ１４０に出力するとともに、入出力Ｉ／Ｆ１４０から供給されるディジタル信号をＭＰＥＧエンコードしてバス１１０上に出力する。入出力Ｉ／Ｆ１４０は、Ａ／Ｄ，Ｄ／Ａコンバータ１４１を内蔵している。入出力Ｉ／Ｆ１４０は、外部から供給されるコンテンツとしてのアナログ信号を受信し、Ａ／Ｄ，Ｄ／Ａコンバータ１４１でＡ／Ｄ(Analog Digital)変換することで、ディジタル信号として、ＭＰＥＧコーデック１３０に出力するとともに、ＭＰＥＧコーデック１３０からのディジタル信号を、Ａ／Ｄ，Ｄ／Ａコンバータ１４１でＤ／Ａ(Digital Analog)変換することで、アナログ信号として、外部に出力する。

【００４２】

暗号処理手段１５０は、例えば、１チップのＬＳＩ(Large Scale Integrated Curcuit)で構成され、バス１１０を介して供給されるコンテンツとしてのディジタル信号の暗号化、復号処理、あるいは認証処理を実行し、暗号データ、復号データ等をバス１１０上に出力する構成を持つ。なお、暗号処理手段１５０は１チップＬＳＩに限らず、各種のソフトウェアまたはハードウェアを組み合わせた構成によって実現することも可能である。ソフトウェア構成による処理手段としての構成については後段で説明する。

【００４３】

ＲＯＭ１６０は、記録再生装置によって処理されるプログラムデータを格納する。ＣＰＵ１７０は、ＲＯＭ１６０、メモリ１８０に記憶されたプログラムを実行することで、ＭＰＥＧコーデック１３０や暗号処理手段１５０等を制御する。メモリ１８０は、例えば、不揮発性メモリで、ＣＰＵ１７０が実行するプログラムや、ＣＰＵ１７０の動作上必要なデータ、さらにデバイスによって実行される暗号処理に使用されるキーセットを記憶する。キーセットについては後段で説明する。ドライブ１９０は、デジタルデータを記録再生可能な記録媒体１９５を駆動することにより、記録媒体１９５からディジタルデータを読み出し（再生し）、バス１１０上に出力するとともに、バス１１０を介して供給されるディジタルデータを、記録媒体１９５に供給して記録させる。

【００４４】

記録媒体１９５は、例えば、ＤＶＤ、ＣＤ等の光ディスク、光磁気ディスク、磁気ディスク、磁気テープ、あるいはＲＡＭ等の半導体メモリ等のディジタルデータの記憶可能な媒体であり、本実施の形態では、ドライブ１９０に対して着脱可能な構成であるとする。但し、記録媒体１９５は、記録再生装置１００に内蔵する構成としてもよい。

【００４５】

なお、図２に示す暗号処理手段１５０は、１つのワンチップＬＳＩとして構成してもよく、また、ソフトウェア、ハードウェアを組み合わせた構成によって実現する構成としてもよい。

【００４６】

［キー配信構成としてのツリー（木）構造について］

次に、図１に示すコンテンツ配信側１０からコンテンツ受信側２０の各デバイスに暗号データを配信する場合における各デバイスにおける暗号処理鍵の保有構成およびデータ配信構成を図３を用いて説明する。

【００４７】

図３の最下段に示すナンバ０～１５がコンテンツ受信側２０の個々のデバイスである。すなわち図３に示す階層ツリー（木）構造の各葉(リーフ：leaf)がそれぞれのデバイスに相当する。

【００４８】

各デバイス０～１５は、製造時あるいは出荷時、あるいはその後において、図３に示す階層ツリー（木）構造における、自分のリーフからルートに至るまでのノードに割り当てられた鍵（ノードキー）および各リーフのリーフキーからなるキーセットをメモリに格納する。図３の最下段に示すＫ００００～Ｋ１１１１が各デバイス０～１５にそれぞれ割り当

てられたリーフキーであり、最上段のＫＲ（ルートキー）から、最下段から２番目の節（ノード）に記載されたキー：ＫＲ～Ｋ１１１をノードキーとする。

【００４９】

図３に示すツリー構成において、例えばデバイス０はリーフキーＫ００００と、ノードキー：Ｋ０００、Ｋ００、Ｋ０、ＫＲを所有する。デバイス５はＫ０１０１、Ｋ０１０、Ｋ０１、Ｋ０、ＫＲを所有する。デバイス１５は、Ｋ１１１１、Ｋ１１１、Ｋ１１、Ｋ１、ＫＲを所有する。なお、図３のツリーにはデバイスが０～１５の１６個のみ記載され、ツリー構造も４段構成の均衡のとれた左右対称構成として示しているが、さらに多くのデバイスがツリー中に構成され、また、ツリーの各部において異なる段数構成を持つことが可能である。

【００５０】

また、図３のツリー構造に含まれる各デバイスには、様々な記録媒体、例えば、デバイス埋め込み型あるいはデバイスに着脱自在に構成されたＤＶＤ、ＣＤ、ＭＤ、フラッシュメモリ等を使用する様々なタイプのデバイスが含まれている。さらに、様々なアプリケーションサービスが共存可能である。このような異なるデバイス、異なるアプリケーションの共存構成の上に図３に示すコンテンツあるいは鍵配布構成である階層ツリー構造が適用される。

【００５１】

これらの様々なデバイス、アプリケーションが共存するシステムにおいて、例えば図３の点線で囲んだ部分、すなわちデバイス０，１，２，３を同一の記録媒体を用いる１つのグループとして設定する。例えば、この点線で囲んだグループ内に含まれるデバイスに対しては、まとめて、共通のコンテンツを暗号化してプロバイダから送付したり、各デバイス共通に使用するコンテンツキーを送付したり、あるいは各デバイスからプロバイダあるいは決済機関等にコンテンツ料金の支払データをやはり暗号化して出力するといった処理が実行される。コンテンツプロバイダ、あるいは決済処理機関等、各デバイスとのデータ送受信を行なう機関は、図３の点線で囲んだ部分、すなわちデバイス０，１，２，３を１つのグループとして一括してデータを送付する処理を実行する。このようなグループは、図３のツリー中に複数存在する。コンテンツプロバイダ、あるいは決済処理機関等、各デバイスとのデータ送受信を行なう機関は、メッセージデータ配信手段として機能する。

【００５２】

なお、ノードキー、リーフキーは、ある１つの鍵管理センタによって統括して管理してもよいし、各グループに対する様々なデータ送受信を行なうプロバイダ、決済機関等のメッセージデータ配信手段によってグループごとに管理する構成としてもよい。これらのノードキー、リーフキーは例えばキーの漏洩等の場合に更新処理が実行され、この更新処理は鍵管理センタ、プロバイダ、決済機関等が実行する。

【００５３】

このツリー構造において、図３から明らかなように、１つのグループに含まれる３つのデバイス０，１，２，３はノードキーとして共通のキーＫ００、Ｋ０、ＫＲを保有する。このノードキー共有構成を利用することにより、例えば共通のコンテンツキーをデバイス０，１，２，３のみに提供することが可能となる。たとえば、共通に保有するノードキーＫ００自体をコンテンツキーとして設定すれば、新たな鍵送付を実行することなくデバイス０，１，２，３のみが共通のコンテンツキーの設定が可能である。また、新たなコンテンツキーＫｃｏｎをノードキーＫ００で暗号化した値Ｅｎｃ（Ｋ００，Ｋｃｏｎ）を、ネットワークを介してあるいは記録媒体に格納してデバイス０，１，２，３に配布すれば、デバイス０，１，２，３のみが、それぞれのデバイスにおいて保有する共有ノードキーＫ００を用いて暗号Ｅｎｃ（Ｋ００，Ｋｃｏｎ）を解いてコンテンツキー：Ｋｃｏｎを得ることが可能となる。なお、Ｅｎｃ（Ｋａ，Ｋｂ）はＫｂをＫａによって暗号化したデータであることを示す。

【００５４】

また、ある時点ｔにおいて、デバイス３の所有する鍵：Ｋ００１１,Ｋ００１,Ｋ００,Ｋ

０,ＫＲが攻撃者（ハッカー）により解析されて露呈したことが発覚した場合、それ以降、システム（デバイス０，１，２，３のグループ）で送受信されるデータを守るために、デバイス３をシステムから切り離す必要がある。そのためには、ノードキー：Ｋ００１，Ｋ００,Ｋ０,ＫＲをそれぞれ新たな鍵Ｋ（ｔ）００１,Ｋ（ｔ）００,Ｋ（ｔ）０,Ｋ（ｔ）Ｒに更新し、デバイス０，１，２にその更新キーを伝える必要がある。ここで、Ｋ（ｔ）ａａａは、鍵Ｋａａａの世代（Generation）：ｔの更新キーであることを示す。

【００５５】

更新キーの配布処理ついて説明する。キーの更新は、例えば、図４（Ａ）に示す有効化キーブロック（ＥＫＢ：Enabling Key Block）と呼ばれるブロックデータによって構成されるテーブルをたとえばネットワーク、あるいは記録媒体に格納してデバイス０，１，２に供給することによって実行される。なお、有効化キーブロック（ＥＫＢ）は、図３に示すようなツリー構造を構成する各リーフに対応するデバイスに新たに更新されたキーを配布するための暗号化キーによって構成される。有効化キーブロック（ＥＫＢ）は、キー更新ブロック（ＫＲＢ：Key Renewal Block）と呼ばれることもある。

【００５６】

図４（Ａ）に示す有効化キーブロック（ＥＫＢ）には、ノードキーの更新の必要なデバイスのみが更新可能なデータ構成を持つブロックデータとして構成される。図４の例は、図３に示すツリー構造中のデバイス０，１，２において、世代ｔの更新ノードキーを配布することを目的として形成されたブロックデータである。図３から明らかなように、デバイス０，デバイス１は、更新ノードキーとしてＫ（ｔ）００、Ｋ（ｔ）０、Ｋ（ｔ）Ｒが必要であり、デバイス２は、更新ノードキーとしてＫ（ｔ）００１、Ｋ（ｔ）００、Ｋ（ｔ）０、Ｋ（ｔ）Ｒが必要である。

【００５７】

図４（Ａ）のＥＫＢに示されるようにＥＫＢには複数の暗号化キーが含まれる。最下段の暗号化キーは、Ｅｎｃ（Ｋ００１０，Ｋ（ｔ）００１）である。これはデバイス２の持つリーフキーＫ００１０によって暗号化された更新ノードキーＫ（ｔ）００１であり、デバイス２は、自身の持つリーフキーによってこの暗号化キーを復号し、Ｋ（ｔ）００１を得ることができる。また、復号により得たＫ（ｔ）００１を用いて、図４（Ａ）の下から２段目の暗号化キーＥｎｃ（Ｋ（ｔ）００１，Ｋ（ｔ）００）を復号可能となり、更新ノードキーＫ（ｔ）００を得ることができる。以下順次、図４（Ａ）の上から２段目の暗号化キーＥｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）０）を復号し、更新ノードキーＫ（ｔ）０、図４（Ａ）の上から１段目の暗号化キーＥｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）Ｒ）を復号しＫ（ｔ）Ｒを得る。一方、デバイスＫ００００．Ｋ０００１は、ノードキーＫ０００は更新する対象に含まれておらず、更新ノードキーとして必要なのは、Ｋ（ｔ）００、Ｋ（ｔ）０、Ｋ（ｔ）Ｒである。デバイスＫ００００．Ｋ０００１は、図４（Ａ）の上から３段目の暗号化キーＥｎｃ（Ｋ０００，Ｋ（ｔ）００）を復号しＫ（ｔ）００、を取得し、以下、図４（Ａ）の上から２段目の暗号化キーＥｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）０）を復号し、更新ノードキーＫ（ｔ）０、図４（Ａ）の上から１段目の暗号化キーＥｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）Ｒ）を復号しＫ（ｔ）Ｒを得る。このようにして、デバイス０，１，２は更新した鍵Ｋ（ｔ）Ｒを得ることができる。なお、図４（Ａ）のインデックスは、復号キーとして使用するノードキー、リーフキーの絶対番地を示す。

【００５８】

図３に示すツリー構造の上位段のノードキー：Ｋ（ｔ）０,Ｋ（ｔ）Ｒの更新が不要であり、ノードキーＫ００のみの更新処理が必要である場合には、図４（Ｂ）の有効化キーブロック（ＥＫＢ）を用いることで、更新ノードキーＫ（ｔ）００をデバイス０，１，２に配布することができる。

【００５９】

図４（Ｂ）に示すＥＫＢは、例えば特定のグループにおいて共有する新たなコンテンツキーを配布する場合に利用可能である。具体例として、図３に点線で示すグループ内のデバイス０，１，２，３がある記録媒体を用いており、新たな共通のコンテンツキーＫ（ｔ）

ｃｏｎが必要であるとする。このとき、デバイス０，１，２，３の共通のノードキーＫ００を更新したＫ（ｔ）００を用いて新たな共通の更新コンテンツキー：Ｋ（ｔ）ｃｏｎを暗号化したデータＥｎｃ（Ｋ（ｔ），Ｋ（ｔ）ｃｏｎ）を図４（Ｂ）に示すＥＫＢとともに配布する。この配布により、デバイス４など、その他のグループの機器においては復号されないデータとしての配布が可能となる。

【００６０】

すなわち、デバイス０，１，２はＥＫＢを処理して得たＫ（ｔ）００を用いて上記暗号文を復号すれば、ｔ時点でのコンテンツキーＫ（ｔ）ｃｏｎを得ることが可能になる。

【００６１】

［ＥＫＢを使用したコンテンツキーの配布］

図５に、ｔ時点でのコンテンツキーＫ（ｔ）ｃｏｎを得る処理例として、Ｋ（ｔ）００を用いて新たな共通のコンテンツキーＫ（ｔ）ｃｏｎを暗号化したデータＥｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）ｃｏｎ）と図４（Ｂ）に示すＥＫＢとを記録媒体を介して受領したデバイス０の処理を示す。すなわちＥＫＢによる暗号化メッセージデータをコンテンツキーＫ（ｔ）ｃｏｎとした例である。

【００６２】

図５に示すように、デバイス０は、記録媒体に格納されている世代：t時点のＥＫＢと自分があらかじめ格納しているノードキーＫ０００を用いて上述したと同様のＥＫＢ処理により、ノードキーＫ（ｔ）００を生成する。さらに、復号した更新ノードキーＫ（ｔ）００を用いて更新コンテンツキーＫ（ｔ）ｃｏｎを復号して、後にそれを使用するために自分だけが持つリーフキーＫ０００００で暗号化して格納する。

【００６３】

［ＥＫＢのフォーマット］

図６に有効化キーブロック（ＥＫＢ）のフォーマット例を示す。バージョン６０１は、有効化キーブロック（ＥＫＢ）のバージョンを示す識別子である。なお、バージョンは最新のＥＫＢを識別する機能とコンテンツとの対応関係を示す機能を持つ。デプスは、有効化キーブロック（ＥＫＢ）の配布先のデバイスに対する階層ツリーの階層数を示す。データポインタ６０３は、有効化キーブロック（ＥＫＢ）中のデータ部の位置を示すポインタであり、タグポインタ６０４はタグ部の位置、署名ポインタ６０５は署名の位置を示すポインタである。

【００６４】

データ部６０６は、例えば更新するノードキーを暗号化したデータを格納する。例えば図５に示すような更新されたノードキーに関する各暗号化キー等を格納する。

【００６５】

タグ部６０７は、データ部に格納された暗号化されたノードキー、リーフキーの位置関係を示すタグである。このタグの付与ルールを図７を用いて説明する。図７では、データとして先に図４（Ａ）で説明した有効化キーブロック（ＥＫＢ）を送付する例を示している。この時のデータは、図７の表（ｂ）に示すようになる。このときの暗号化キーに含まれるトップノードのアドレスをトップノードアドレスとする。この場合は、ルートキーの更新キーＫ（ｔ）Ｒが含まれているので、トップノードアドレスはＫＲとなる。このとき、例えば最上段のデータＥｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）Ｒ）は、図７の（ａ）に示す階層ツリーに示す位置にある。ここで、次のデータは、Ｅｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）０）であり、ツリー上では前のデータの左下の位置にある。データがある場合は、タグが０、ない場合は１が設定される。タグは｛左（Ｌ）タグ，右（Ｒ）タグ｝として設定される。最上段のデータＥｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）Ｒ）の左にはデータがあるので、Ｌタグ＝０、右にはデータがないので、Ｒタグ＝１となる。以下、すべてのデータにタグが設定され、図７（ｃ）に示すデータ列、およびタグ列が構成される。

【００６６】

タグは、データＥｎｃ（Ｋｘｘｘ，Ｋｙｙｙ）がツリー構造のどこに位置しているのかを示すために設定されるものである。データ部に格納されるキーデータＥｎｃ（Ｋｘｘｘ，

Ｋｙｙｙ）．．．は、単純に暗号化されたキーの羅列データに過ぎないので、上述したタグによってデータとして格納された暗号化キーのツリー上の位置を判別可能としたものである。上述したタグを用いずに、先の図４で説明した構成のように暗号化データに対応させたノード・インデックスを用いて、例えば、
０：Ｅｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）ｒｏｏｔ）
００：Ｅｎｃ（Ｋ（ｔ）００，Ｋ（ｔ）０）
０００：Ｅｎｃ（Ｋ（（ｔ）０００，Ｋ（Ｔ）００）
．．．のようなデータ構成とすることも可能であるが、このようなインデックスを用いた構成とすると冗長なデータとなりデータ量が増大し、ネットワークを介する配信等においては好ましくない。これに対し、上述したタグをキー位置を示す索引データとして用いることにより、少ないデータ量でキー位置の判別が可能となる。

【００６７】
図６に戻って、ＥＫＢフォーマットについてさらに説明する。署名（Signature）は、有効化キーブロック（ＥＫＢ）を発行した例えば鍵管理センタ、コンテンツロバイダ、決済機関等が実行する電子署名である。ＥＫＢを受領したデバイスは署名検証によって正当な有効化キーブロック（ＥＫＢ）発行者が発行した有効化キーブロック（ＥＫＢ）であることを確認する。

【００６８】
［ＥＫＢを使用したコンテンツキーおよびコンテンツの配信］
上述の例では、コンテンツキーのみをＥＫＢとともに送付する例について説明したが、コンテンツキーで暗号化したコンテンツと、コンテンツキー暗号キーで暗号化したコンテンツキーと、ＥＫＢによって暗号化したコンテンツキー暗号鍵を併せて送付する構成について以下説明する。

【００６９】
図８にこのデータ構成を示す。図８（ａ）に示す構成において、Ｅｎｃ（Ｋｃｏｎ，ｃｏｎｔｅｎｔ）８０１は、コンテンツ（Content）をコンテンツキー（Ｋｃｏｎ）で暗号化したデータであり、Ｅｎｃ（ＫＥＫ，Ｋｃｏｎ）８０２は、コンテンツキー（Ｋｃｏｎ）をコンテンツキー暗号キー（ＫＥＫ：Key Encryption Key）で暗号化したデータであり、Ｅｎｃ（ＥＫＢ，ＫＥＫ）８０３は、コンテンツキー暗号キーＫＥＫを有効化キーブロック（ＥＫＢ）によって暗号化したデータであることを示す。

【００７０】
ここで、コンテンツキー暗号キーＫＥＫは、図３で示すノードキー（Ｋ０００，Ｋ００…）、あるいはルートキー（ＫＲ）自体であってもよく、またノードキー（Ｋ０００，Ｋ００…）、あるいはルートキー（ＫＲ）によって暗号化されたキーであってもよい。

【００７１】
図８（ｂ）は、複数のコンテンツがメディアに記録され、それぞれが同じＥｎｃ（ＥＫＢ，ＫＥＫ）８０５を利用している場合の構成例を示す、このような構成においては、各データに同じＥｎｃ（ＥＫＢ，ＫＥＫ）を付加することなく、Ｅｎｃ（ＥＫＢ，ＫＥＫ）にリンクするリンク先を示すデータを各データに付加する構成とすることができる。

【００７２】
図９にコンテンツキー暗号キーＫＥＫを、図３に示すノードキーＫ００を更新した更新ノードキーＫ（ｔ）００として構成した場合の例を示す。この場合、図３の点線枠で囲んだグループにおいてデバイス３が、例えば鍵の漏洩によりリボーク（排除）されているとして、他のグループのメンバ、すなわち、デバイス０，１，２に対して図９に示す（ａ）有効化キーブロック（ＥＫＢ）と、（ｂ）コンテンツキー（Ｋｃｏｎ）をコンテンツキー暗号キー（ＫＥＫ＝Ｋ（ｔ）００）で暗号化したデータと、（ｃ）コンテンツ（content）をコンテンツキー（Ｋｃｏｎ）で暗号化したデータとを配信することにより、デバイス０，１，２はコンテンツを得ることができる。

【００７３】
図９の右側には、デバイス０における復号手順を示してある。デバイス０は、まず、受領

した有効化キーブロックから自身の保有するリーフキーＫ０００を用いた復号処理により、コンテンツキー暗号キー（ＫＥＫ＝Ｋ（ｔ）００）を取得する。次に、Ｋ（ｔ）００による復号によりコンテンツキーＫｃｏｎを取得し、さらにコンテンツキーＫｃｏｎによりコンテンツの復号を行なう。これらの処理により、デバイス０はコンテンツを利用可能となる。デバイス１，２においても各々異なる処理手順でＥＫＢを処理することにより、コンテンツキー暗号キー（ＫＥＫ＝Ｋ（ｔ）００）を取得することが可能となり、同様にコンテンツを利用することが可能となる。

【００７４】

図３に示す他のグループのデバイス４，５，６…は、この同様のデータ（ＥＫＢ）を受信したとしても、自身の保有するリーフキー、ノードキーを用いてコンテンツキー暗号キー（ＫＥＫ＝Ｋ（ｔ）００）を取得することができない。同様にリボークされたデバイス３においても、自身の保有するリーフキー、ノードキーでは、コンテンツキー暗号キー（ＫＥＫ＝Ｋ（ｔ）００）を取得することができず、正当な権利を有するデバイスのみがコンテンツを復号して利用することが可能となる。

【００７５】

このように、ＥＫＢを利用したコンテンツキーの配送を用いれば、データ量を少なくして、かつ安全に正当権利者のみが復号可能とした暗号化コンテンツを配信することが可能となる。

【００７６】

なお、有効化キーブロック（ＥＫＢ）、コンテンツキー、暗号化コンテンツ等は、ネットワークを介して安全に配信することが可能な構成であるが、有効化キーブロック（ＥＫＢ）、コンテンツキー、暗号化コンテンツをＤＶＤ、ＣＤ等の記録媒体に格納してユーザに提供することも可能である。この場合、記録媒体に格納された暗号化コンテンツの復号には、同一の記録媒体に格納された有効化キーブロック（ＥＫＢ）の復号により得られるコンテンツキーを使用するように構成すれば、予め正当権利者のみが保有するリーフキー、ノードキーによってのみ利用可能な暗号化コンテンツの配布処理、すなわち利用可能なユーザデバイスを限定したコンテンツ配布が簡易な構成で実現可能となる。

【００７７】

図１０に記録媒体に暗号化コンテンツとともに有効化キーブロック（ＥＫＢ）を格納した構成例を示す。図１０に示す例においては、記録媒体にコンテンツＣ１～Ｃ４が格納され、さらに各格納コンテンツに対応するの有効化キーブロック（ＥＫＢ）を対応付けたデータが格納され、さらにバージョンＭの有効化キーブロック（ＥＫＢ＿Ｍ）が格納されている。例えばＥＫＢ＿１はコンテンツＣ１を暗号化したコンテンツキーＫｃｏｎ１を生成するのに使用され、例えばＥＫＢ＿２はコンテンツＣ２を暗号化したコンテンツキーＫｃｏｎ２を生成するのに使用される。この例では、バージョンＭの有効化キーブロック（ＥＫＢ＿Ｍ）が記録媒体に格納されており、コンテンツＣ３，Ｃ４は有効化キーブロック（ＥＫＢ＿Ｍ）に対応付けられているので、有効化キーブロック（ＥＫＢ＿Ｍ）の復号によりコンテンツＣ３，Ｃ４のコンテンツキーを取得することができる。ＥＫＢ＿１、ＥＫＢ＿２はディスクに格納されていないので、新たな提供手段、例えばネットワーク配信、あるいは記録媒体による配信によってそれぞれのコンテンツキーを復号するために必要なＥＫＢ＿１，ＥＫＢ＿２を取得することが必要となる。

【００７８】

図１１に、複数のデバイス間でコンテンツキーが流通する場合のＥＫＢを利用したコンテンツキーの配信と、従来のコンテンツキー配信処理の比較例を示す。上段（ａ）が従来構成であり、下段（ｂ）が本発明の有効化キーブロック（ＥＫＢ）を利用した例である。なお、図１１においてＫａ（Ｋｂ）は、ＫｂをＫａで暗号化したデータであることを示す。

【００７９】

（ａ）に示すように、従来は、データ送受信者の正当性を確認し、またデータ送信の暗号化処理に使用するセッションキーＫｓｅｓを共有するために各デバイス間において、認証処理および鍵交換処理（ＡＫＥ：Authentication and Key Exchange）を実行し、認証が

成立したことを条件としてセッションキーＫｓｅｓでコンテンツキーＫｃｏｎを暗号化して送信する処理を行なっていた。

【００８０】

例えば図１１（ａ）のＰＣにおいては、受信したセッションキーで暗号化したコンテンツキーＫｓｅｓ（Ｋｃｏｎ）をセッションキーで復号してＫｃｏｎを得ることが可能であり、さらに取得したＫｃｏｎをＰＣ自体の保有する保存キーＫｓｔｒで暗号化して自身のメモリに保存することが可能となる。

【００８１】

図１１（ａ）において、コンテンツプロバイダは、図１１（ａ）の記録デバイス１１０１にのみデータを利用可能な形で配信したい場合でも、間にＰＣ、再生装置が存在する場合は、図１１（ａ）に示すように認証処理を実行し、それぞれのセッションキーでコンテンツキーを暗号化して配信するといった処理が必要となる。また、間に介在するＰＣ、再生装置においても認証処理において生成し共有することになったセッションキーを用いることで暗号化コンテンツキーを復号してコンテンツキーを取得可能となる。

【００８２】

一方、図１１（ｂ）の下段に示す有効化キーブロック（ＥＫＢ）を利用した例においては、コンテンツプロバイダから有効化キーブロック（ＥＫＢ）と、有効化キーブロック（ＥＫＢ）の処理によって得られるノードキー、またはルートキーによってコンテンツキーＫｃｏｎを暗号化したデータ（図の例ではＫｒｏｏｔ（Ｋｃｏｎ））を配信することにより、配信したＥＫＢの処理が可能な機器においてのみコンテンツキーＫｃｏｎを復号して取得することが可能になる。

【００８３】

従って、例えば図１１（ｂ）の右端にのみ利用可能な有効化キーブロック（ＥＫＢ）を生成して、その有効化キーブロック（ＥＫＢ）と、そのＥＫＢ処理によって得られるノードキー、またはルートキーによってコンテンツキーＫｃｏｎを暗号化したデータを併せて送ることにより、間に存在するＰＣ、再生機器等は、自身の有するリーフキー、ノードキーによっては、ＥＫＢの処理を実行することができない。従って、データ送受信デバイス間での認証処理、セッションキーの生成、セッションキーによるコンテンツキーＫｃｏｎの暗号化処理といった処理を実行することなく、安全に正当なデバイスに対してのみ利用可能なコンテンツキーを配信することが可能となる。

【００８４】

ＰＣ、記録再生器にも利用可能なコンテンツキーを配信したい場合は、それぞれにおいて処理可能な有効化キーブロック（ＥＫＢ）を生成して、配信することにより、共通のコンテンツキーを取得することが可能となる。

【００８５】

［有効化キーブロック（ＥＫＢ）を使用した認証キーの配信（共通鍵方式）］

上述の有効化キーブロック（ＥＫＢ）を使用したデータあるいはキーの配信において、デバイス間で転送される有効化キーブロック（ＥＫＢ）およびコンテンツあるいはコンテンツキーは常に同じ暗号化形態を維持しているため、データ伝走路を盗み出して記録し、再度、後で転送する、いわゆるリプレイアタックにより、不正コピーが生成される可能性がある。これを防ぐ構成としては、データ転送デバイス間において、従来と同様の認証処理および鍵交換処理を実行することが有効な手段である。ここでは、この認証処理および鍵交換処理を実行する際に使用する認証キーＫａｋｅを上述の有効化キーブロック（ＥＫＢ）を使用してデバイスに配信することにより、安全な秘密鍵として共有する認証キーを持ち、共通鍵方式に従った認証処理を実行する構成について説明する。すなわちＥＫＢによる暗号化メッセージデータを認証キーとした例である。

【００８６】

図１２に、共通鍵暗号方式を用いた相互認証方法（ISO/IEC 9798-2）を示す。図１２においては、共通鍵暗号方式としてＤＥＳを用いているが、共通鍵暗号方式であれば他の方式も可能である。図１２において、まず、Ｂが６４ビットの乱数Ｒｂを生成し、Ｒｂおよび

自己のＩＤであるＩＤ（ｂ）をＡに送信する。これを受信したＡは、新たに６４ビットの乱数Ｒａを生成し、Ｒａ、Ｒｂ、ＩＤ（ｂ）の順に、ＤＥＳのＣＢＣモードで鍵Ｋａｂを用いてデータを暗号化し、Ｂに返送する。なお、鍵Ｋａｂは、ＡおよびＢに共通の秘密鍵としてそれぞれの記録素子内に格納する鍵である。ＤＥＳのＣＢＣモードを用いた鍵Ｋａｂによる暗号化処理は、例えばＤＥＳを用いた処理においては、初期値とＲａとを排他的論理和し、ＤＥＳ暗号化部において、鍵Ｋａｂを用いて暗号化し、暗号文Ｅ１を生成し、続けて暗号文Ｅ１とＲｂとを排他的論理和し、ＤＥＳ暗号化部において、鍵Ｋａｂを用いて暗号化し、暗号文Ｅ２を生成し、さらに、暗号文Ｅ２とＩＤ（ｂ）とを排他的論理和し、ＤＥＳ暗号化部において、鍵Ｋａｂを用いて暗号化して生成した暗号文Ｅ３とによって送信データ（Token-AB）を生成する。

【００８７】

これを受信したＢは、受信データを、やはり共通の秘密鍵としてそれぞれの記録素子内に格納する鍵Ｋａｂ（認証キー）で復号化する。受信データの復号化方法は、まず、暗号文Ｅ１を認証キーＫａｂで復号化し、乱数Ｒａを得る。次に、暗号文Ｅ２を認証キーＫａｂで復号化し、その結果とＥ１を排他的論理和し、Ｒｂを得る。最後に、暗号文Ｅ３を認証キーＫａｂで復号化し、その結果とＥ２を排他的論理和し、ＩＤ(ｂ)を得る。こうして得られたＲａ、Ｒｂ、ＩＤ(ｂ)のうち、ＲｂおよびＩＤ(ｂ)が、Ｂが送信したものと一致するか検証する。この検証に通った場合、ＢはＡを正当なものとして認証する。

【００８８】

次にＢは、認証後に使用するセッションキー（Ｋｓｅｓ）を生成する（生成方法は、乱数を用いる）。そして、Ｒｂ、Ｒａ、Ｋｓｅｓの順に、ＤＥＳのＣＢＣモードで認証キーＫａｂを用いて暗号化し、Ａに返送する。

【００８９】

これを受信したＡは、受信データを認証キーＫａｂで復号化する。受信データの復号化方法は、Ｂの復号化処理と同様であるので、ここでは詳細を省略する。こうして得られたＲｂ、Ｒａ、Ｋｓｅｓの内、ＲｂおよびＲａが、Ａが送信したものと一致するか検証する。この検証に通った場合、ＡはＢを正当なものとして認証する。互いに相手を認証した後には、セッションキーＫｓｅｓは、認証後の秘密通信のための共通鍵として利用される。

【００９０】

なお、受信データの検証の際に、不正、不一致が見つかった場合には、相互認証が失敗したものとして処理を中断する。

【００９１】

上述の認証処理においては、Ａ，Ｂは共通の認証キーＫａｂを共有する。この共通鍵Ｋａｂを上述の有効化キーブロック（ＥＫＢ）を使用してデバイスに配信する。

【００９２】

例えば、図１２の例では、Ａ，またはＢのいずれかが他方が復号可能な有効化キーブロック（ＥＫＢ）を生成して生成した有効化キーブロック（ＥＫＢ）によって認証キーＫａｂを暗号化して、他方に送信する構成としてもよいし、あるいは第３者がデバイスＡ，Ｂに対して双方が利用可能な有効化キーブロック（ＥＫＢ）を生成してデバイスＡ，Ｂに対して生成した有効化キーブロック（ＥＫＢ）によって認証キーＫａｂを暗号化して配信する構成としてもよい。

【００９３】

図１３および図１４に複数のデバイスに共通の認証キーＫａｋｅを有効化キーブロック（ＥＫＢ）によって配信する構成例を示す。図１３はデバイス０，１，２，３に対して復号可能な認証キーＫａｋｅを配信する例、図１４はデバイス０，１，２，３中のデバイス３をリボーク（排除）してデバイス０，１，２に対してのみ復号可能な認証キーを配信する例を示す。

【００９４】

図１３の例では、更新ノードキーＫ（ｔ）００によって、認証キーＫａｋｅを暗号化したデータ（ｂ）とともに、デバイス０，１，２，３においてそれぞれの有するノードキー、

リーフキーを用いて更新されたノードキーＫ（ｔ）００を復号可能な有効化キーブロック（ＥＫＢ）を生成して配信する。それぞれのデバイスは、図１３の右側に示すようにまず、ＥＫＢを処理（復号）することにより、更新されたノードキーＫ（ｔ）００を取得し、次に、取得したノードキーＫ（ｔ）００を用いて暗号化された認証キー：Ｅｎｃ（Ｋ（ｔ）００，Ｋａｋｅ）を復号して認証キーＫａｋｅを得ることが可能となる。

【００９５】

その他のデバイス４，５，６，７…は同一の有効化キーブロック（ＥＫＢ）を受信しても自身の保有するノードキー、リーフキーでは、ＥＫＢを処理して更新されたノードキーＫ（ｔ）００を取得することができないので、安全に正当なデバイスに対してのみ認証キーを送付することができる。

【００９６】

一方、図１４の例は、図３の点線枠で囲んだグループにおいてデバイス３が、例えば鍵の漏洩によりリボーク（排除）されているとして、他のグループのメンバ、すなわち、デバイス０，１，２，に対してのみ復号可能な有効化キーブロック（ＥＫＢ）を生成して配信した例である。図１４に示す（ａ）有効化キーブロック（ＥＫＢ）と、（ｂ）認証キー（Ｋａｋｅ）をノードキー（Ｋ（ｔ）００）で暗号化したデータを配信する。

【００９７】

図１４の右側には、復号手順を示してある。デバイス０，１，２は、まず、受領した有効化キーブロックから自身の保有するリーフキーまたはノードキーを用いた復号処理により、更新ノードキー（Ｋ（ｔ）００）を取得する。次に、Ｋ（ｔ）００による復号により認証キーＫａｋｅを取得する。

【００９８】

図３に示す他のグループのデバイス４，５，６…は、この同様のデータ（ＥＫＢ）を受信したとしても、自身の保有するリーフキー、ノードキーを用いて更新ノードキー（Ｋ（ｔ）００）を取得することができない。同様にリボークされたデバイス３においても、自身の保有するリーフキー、ノードキーでは、更新ノードキー（Ｋ（ｔ）００）を取得することができず、正当な権利を有するデバイスのみが認証キーを復号して利用することが可能となる。

【００９９】

このように、ＥＫＢを利用した認証キーの配送を用いれば、データ量を少なくして、かつ安全に正当権利者のみが復号可能とした認証キーを配信することが可能となる。

【０１００】

［公開鍵認証と有効化キーブロック（ＥＫＢ）を使用したコンテンツキーの配信］

次に、公開鍵認証と有効化キーブロック（ＥＫＢ）を使用したコンテンツキーの配信処理について説明する。まず、公開鍵暗号方式である１６０ビット長の楕円曲線暗号を用いた相互認証方法を、図１５を用いて説明する。図１５において、公開鍵暗号方式としてＥＣＣを用いているが、同様な公開鍵暗号方式であればいずれでもよい。また、鍵サイズも１６０ビットでなくてもよい。図１５において、まずＢが、６４ビットの乱数Ｒｂを生成し、Ａに送信する。これを受信したＡは、新たに６４ビットの乱数Ｒａおよび素数ｐより小さい乱数Ａｋを生成する。そして、ベースポイントＧをＡｋ倍した点Ａｖ＝Ａｋ×Ｇを求め、Ｒａ、Ｒｂ、Ａｖ（X座標とY座標）に対する電子署名Ａ．Ｓｉｇを生成し、Ａの公開鍵証明書とともにＢに返送する。ここで、ＲａおよびＲｂはそれぞれ６４ビット、ＡｖのＸ座標とＹ座標がそれぞれ１６０ビットであるので、合計４４８ビットに対する電子署名を生成する。

【０１０１】

Ａの公開鍵証明書、Ｒａ、Ｒｂ、Ａｖ、電子署名Ａ．Ｓｉｇを受信したＢは、Ａが送信してきたＲｂが、Ｂが生成したものと一致するか検証する。その結果、一致していた場合には、Ａの公開鍵証明書内の電子署名を認証局の公開鍵で検証し、Ａの公開鍵を取り出す。そして、取り出したＡの公開鍵を用い電子署名Ａ．Ｓｉｇを検証する。

【０１０２】

次に、Ｂは、素数ｐより小さい乱数Ｂｋを生成する。そして、ベースポイントＧをＢｋ倍した点Ｂｖ＝Ｂｋ×Ｇを求め、Ｒｂ、Ｒａ、Ｂｖ（Ｘ座標とＹ座標）に対する電子署名Ｂ．Ｓｉｇを生成し、Ｂの公開鍵証明書とともにＡに返送する。

【０１０３】

Ｂの公開鍵証明書、Ｒｂ、Ｒａ、Ａｖ、電子署名Ｂ．Ｓｉｇを受信したＡは、Ｂが送信してきたＲａが、Ａが生成したものと一致するか検証する。その結果、一致していた場合には、Ｂの公開鍵証明書内の電子署名を認証局の公開鍵で検証し、Ｂの公開鍵を取り出す。そして、取り出したＢの公開鍵を用い電子署名Ｂ．Ｓｉｇを検証する。電子署名の検証に成功した後、ＡはＢを正当なものとして認証する。

【０１０４】

両者が認証に成功した場合には、ＢはＢｋ×Ａｖ（Ｂｋは乱数だが、Ａｖは楕円曲線上の点であるため、楕円曲線上の点のスカラー倍計算が必要）を計算し、ＡはＡｋ×Ｂｖを計算し、これら点のＸ座標の下位６４ビットをセッションキーとして以降の通信に使用する（共通鍵暗号を６４ビット鍵長の共通鍵暗号とした場合）。もちろん、Ｙ座標からセッション鍵を生成してもよいし、下位６４ビットでなくてもよい。なお、相互認証後の秘密通信においては、送信データはセッションキーで暗号化されるだけでなく、電子署名も付されることがある。

【０１０５】

電子署名の検証や受信データの検証の際に、不正、不一致が見つかった場合には、相互認証が失敗したものとして処理を中断する。

【０１０６】

図１６に公開鍵認証と有効化キーブロック（ＥＫＢ）を使用したコンテンツキーの配信処理例を示す。まずコンテンツプロバイダとＰＣ間において図１５で説明した公開鍵方式による認証処理が実行される。コンテンツプロバイダは、コンテンツキー配信先である再生装置、記録媒体の有するノードキー、リーフキーによって復号可能なＥＫＢを生成して、更新ノードキーによる暗号化を実行したコンテンツキーＥ（Ｋｃｏｎ）と、有効化キーブロック（ＥＫＢ）とをＰＣ間の認証処理において生成したセッションキーＫｓｅｓで暗号化してＰＣに送信する。

【０１０７】

ＰＣはセッションキーで暗号化された［更新ノードキーによる暗号化を実行したコンテンツキーＥ（Ｋｃｏｎ）と、有効化キーブロック（ＥＫＢ）］をセッションキーで復号した後、再生装置、記録媒体に送信する。

【０１０８】

再生装置、記録媒体は、自身の保有するノードキーまたはリーフキーによって［更新ノードキーによる暗号化を実行したコンテンツキーＥ（Ｋｃｏｎ）と、有効化キーブロック（ＥＫＢ）］を復号することによってコンテンツキーＫｃｏｎを取得する。

【０１０９】

この構成によれば、コンテンツプロバイダとＰＣ間での認証を条件として［更新ノードキーによる暗号化を実行したコンテンツキーＥ（Ｋｃｏｎ）と、有効化キーブロック（ＥＫＢ）］が送信されるので、例えば、ノードキーの漏洩があった場合でも、確実な相手に対するデータ送信が可能となる。

【０１１０】

［プログラムコードの有効化キーブロック（ＥＫＢ）を使用した配信］

上述した例では、コンテンツキー、認証キー等を有効化キーブロック（ＥＫＢ）を用いて暗号化して配信する方法を説明したが、様々なプログラムコードを有効化キーブロック（ＥＫＢ）を用いて配信する構成も可能である。すなわちＥＫＢによる暗号化メッセージデータをプログラムコードとした例である。以下、この構成について説明する。

【０１１１】

図１７にプログラムコードを有効化キーブロック（ＥＫＢ）の例えば更新ノードキーによって暗号化してデバイス間で送信する例を示す。デバイス１７０１は、デバイス１７０２

の有するノードキー、リーフキーによって復号可能な有効化キーブロック（ＥＫＢ）と、有効化キーブロック（ＥＫＢ）に含まれる更新ノードキーで暗号処理したプログラムコードをデバイス１７０２に送信する。デバイス１７０２は受信したＥＫＢを処理して更新ノードキーを取得して、さらに取得した更新ノードキーによってプログラムコードの復号を実行して、プログラムコードを得る。

【０１１２】

図１７に示す例では、さらに、デバイス１７０２において取得したプログラムコードによる処理を実行して、その結果をデバイス１７０１に返して、デバイス１７０１がその結果に基づいて、さらに処理を続行する例を示している。

【０１１３】

このように有効化キーブロック（ＥＫＢ）と、有効化キーブロック（ＥＫＢ）に含まれる更新ノードキーで暗号処理したプログラムコードを配信することにより、特定のデバイスにおいて解読可能なプログラムコードを前述の図３で示した特定のデバイス、あるいはグループに対して配信することが可能となる。

【０１１４】

［送信コンテンツに対するチェック値（ＩＣＶ:Integrity Check Value）を対応させる構成］

次に、コンテンツの改竄を防止するためにコンテンツのインテグリティ・チェック値（ＩＣＶ）を生成して、コンテンツに対応付けて、ＩＣＶの計算により、コンテンツ改竄の有無を判定する処理構成について説明する。

【０１１５】

コンテンツのインテグリティ・チェック値（ＩＣＶ）は、例えばコンテンツに対するハッシュ関数を用いて計算され、ＩＣＶ＝ｈａｓｈ（Ｋｉｃｖ，Ｃ１，Ｃ２，…）によって計算される。ＫｉｃｖはＩＣＶ生成キーである。Ｃ１，Ｃ２はコンテンツの情報であり、コンテンツの重要情報のメッセージ認証符号（ＭＡＣ：Message authentication Code）が使用される。

【０１１６】

ＤＥＳ暗号処理構成を用いたＭＡＣ値生成例を図１８に示す。図１８の構成に示すように対象となるメッセージを８バイト単位に分割し、（以下、分割されたメッセージをＭ１、Ｍ２、・・・、ＭＮとする）、まず、初期値（Initial Value（以下、ＩＶとする））とＭ１を排他的論理和する（その結果をＩ１とする）。次に、Ｉ１をＤＥＳ暗号化部に入れ、鍵（以下、Ｋ１とする）を用いて暗号化する（出力をＥ１とする）。続けて、Ｅ１およびＭ２を排他的論理和し、その出力Ｉ２をＤＥＳ暗号化部へ入れ、鍵Ｋ１を用いて暗号化する（出力Ｅ２）。以下、これを繰り返し、全てのメッセージに対して暗号化処理を施す。最後に出てきたＥＮがメッセージ認証符号（ＭＡＣ（Message Authentication Code））となる。

【０１１７】

このようなコンテンツのＭＡＣ値とＩＣＶ生成キーにハッシュ関数を適用して用いてコンテンツのインテグリティ・チェック値（ＩＣＶ）が生成される。改竄のないことが保証された例えばコンテンツ生成時に生成したＩＣＶと、新たにコンテンツに基づいて生成したＩＣＶとを比較して同一のＩＣＶが得られればコンテンツに改竄のないことが保証され、ＩＣＶが異なれば、改竄があったと判定される。

【０１１８】

［チェック値（ＩＣＶ）の生成キーＫｉｃｖをＥＫＢによって配布する構成］

次に、コンテンツのインテグリティ・チェック値（ＩＣＶ）生成キーであるＫｉｃｖを上述の有効化キーブロックによって送付する構成について説明する。すなわちＥＫＢによる暗号化メッセージデータをコンテンツのインテグリティ・チェック値（ＩＣＶ）生成キーとした例である。

【０１１９】

図１９および図２０に複数のデバイスに共通のコンテンツを送付した場合、それらのコン

テンツの改竄の有無を検証するためのインテグリティ・チェック値生成キーＫｉｃｖを有効化キーブロック（ＥＫＢ）によって配信する構成例を示す。図１９はデバイス０，１，２，３に対して復号可能なチェック値生成キーＫｉｃｖを配信する例、図２０はデバイス０，１，２，３中のデバイス３をリボーク（排除）してデバイス０，１，２に対してのみ復号可能なチェック値生成キーＫｉｃｖを配信する例を示す。

【０１２０】

図１９の例では、更新ノードキーＫ（ｔ）００によって、チェック値生成キーＫｉｃｖを暗号化したデータ（ｂ）とともに、デバイス０，１，２，３においてそれぞれの有するノードキー、リーフキーを用いて更新されたノードキーＫ（ｔ）００を復号可能な有効化キーブロック（ＥＫＢ）を生成して配信する。それぞれのデバイスは、図１９の右側に示すようにまず、ＥＫＢを処理（復号）することにより、更新されたノードキーＫ（ｔ）００を取得し、次に、取得したノードキーＫ（ｔ）００を用いて暗号化されたチェック値生成キー：Ｅｎｃ（Ｋ（ｔ）００，Ｋｉｃｖ）を復号してチェック値生成キーＫｉｃｖを得ることが可能となる。

【０１２１】

その他のデバイス４，５，６，７…は同一の有効化キーブロック（ＥＫＢ）を受信しても自身の保有するノードキー、リーフキーでは、ＥＫＢを処理して更新されたノードキーＫ（ｔ）００を取得することができないので、安全に正当なデバイスに対してのみチェック値生成キーを送付することができる。

【０１２２】

一方、図２０の例は、図３の点線枠で囲んだグループにおいてデバイス３が、例えば鍵の漏洩によりリボーク（排除）されているとして、他のグループのメンバ、すなわち、デバイス０，１，２，に対してのみ復号可能な有効化キーブロック（ＥＫＢ）を生成して配信した例である。図２０に示す（ａ）有効化キーブロック（ＥＫＢ）と、（ｂ）チェック値生成キー（Ｋｉｃｖ）をノードキー（Ｋ（ｔ）００）で暗号化したデータを配信する。

【０１２３】

図２０の右側には、復号手順を示してある。デバイス０，１，２は、まず、受領した有効化キーブロックから自身の保有するリーフキーまたはノードキーを用いた復号処理により、更新ノードキー（Ｋ（ｔ）００）を取得する。次に、Ｋ（ｔ）００による復号によりチェック値生成キーＫｉｃｖを取得する。

【０１２４】

図３に示す他のグループのデバイス４，５，６…は、この同様のデータ（ＥＫＢ）を受信したとしても、自身の保有するリーフキー、ノードキーを用いて更新ノードキー（Ｋ（ｔ）００）を取得することができない。同様にリボークされたデバイス３においても、自身の保有するリーフキー、ノードキーでは、更新ノードキー（Ｋ（ｔ）００）を取得することができず、正当な権利を有するデバイスのみがチェック値生成キーを復号して利用することが可能となる。

【０１２５】

このように、ＥＫＢを利用したチェック値生成キーの配送を用いれば、データ量を少なくして、かつ安全に正当権利者のみが復号可能としたチェック値生成キーを配信することが可能となる。

【０１２６】

このようなコンテンツのインテグリティ・チェック値（ＩＣＶ）を用いることにより、ＥＫＢと暗号化コンテンツの不正コピーを排除することができる。例えば図２１に示すように、コンテンツＣ１とコンテンツＣ２とをそれぞれのコンテンツキーを取得可能な有効化キーブロック（ＥＫＢ）とともに格納したメディア１があり、これをそのままメディア２にコピーした場合を想定する。ＥＫＢと暗号化コンテンツのコピーは可能であり、これをＥＫＢを復号可能なデバイスでは利用できることになる。

【０１２７】

図２１の（ｂ）に示すように各メディアに正当に格納されたコンテンツに対応付けてイン

テグリティ・チェック値（ＩＣＶ（Ｃ１，Ｃ２））を格納する構成とする。なお、（ＩＣＶ（Ｃ１，Ｃ２））は、コンテンツＣ１とコンテンツＣ２にハッシュ関数を用いて計算されるコンテンツのインテグリティ・チェック値であるＩＣＶ＝ｈａｓｈ（Ｋｉｃｖ，Ｃ１，Ｃ２）を示している。図２１の（ｂ）の構成において、メディア１には正当にコンテンツ１とコンテンツ２が格納され、コンテンツＣ１とコンテンツＣ２に基づいて生成されたインテグリティ・チェック値（ＩＣＶ（Ｃ１，Ｃ２））が格納される。また、メディア２には正当にコンテンツ１が格納され、コンテンツＣ１に基づいて生成されたインテグリティ・チェック値（ＩＣＶ（Ｃ１））が格納される。この構成において、メディア１に格納された｛ＥＫＢ，コンテンツ２｝をメディア２にコピーしたとすると、メディア２で、コンテンツチェック値を新たに生成するとＩＣＶ（Ｃ１，Ｃ２）が生成されることになり、メディアに格納されているＫｉｃｖ（Ｃ１）と異なり、コンテンツの改竄あるいは不正なコピーによる新たなコンテンツの格納が実行されたことが明らかになる。メディアを再生するデバイスにおいて、再生ステップの前ステップにＩＣＶチェックを実行して、生成ＩＣＶと格納ＩＣＶの一致を判別し、一致しない場合は、再生を実行しない構成とすることにより、不正コピーのコンテンツの再生を防止することが可能となる。

【０１２８】

また、さらに、安全性を高めるため、コンテンツのインテグリティ・チェック値（ＩＣＶ）を書き換えカウンタを含めたデータに基づいて生成する構成としてもよい。すなわちＩＣＶ＝ｈａｓｈ（Ｋｉｃｖ，ｃｏｕｎｔｅｒ＋１，Ｃ１，Ｃ２，…）によって計算する構成とする。ここで、カウンタ（ｃｏｕｎｔｅｒ＋１）は、ＩＣＶの書き換えごとに１つインクリメントされる値として設定する。なお、カウンタ値はセキュアなメモリに格納する構成とすることが必要である。

【０１２９】

さらに、コンテンツのインテグリティ・チェック値（ＩＣＶ）をコンテンツと同一メディアに格納することができない構成においては、コンテンツのインテグリティ・チェック値（ＩＣＶ）をコンテンツとは別のメディア上に格納する構成としてもよい。

【０１３０】

例えば、読み込み専用メディアや通常のＭＯ等のコピー防止策のとられていないメディアにコンテンツを格納する場合、同一メディアにインテグリティ・チェック値（ＩＣＶ）を格納するとＩＣＶの書き換えが不正なユーザによりなされる可能性があり、ＩＣＶの安全性が保てないおそれがある。この様な場合、ホストマシン上の安全なメディアにＩＣＶを格納して、コンテンツのコピーコントロール（例えばcheck-in/check-out、move）にＩＣＶを使用する構成とすることにより、ＩＣＶの安全な管理およびコンテンツの改竄チェックが可能となる。

【０１３１】

この構成例を図２２に示す。図２２では読み込み専用メディアや通常のＭＯ等のコピー防止策のとられていないメディア２２０１にコンテンツが格納され、これらのコンテンツに関するインテグリティ・チェック値（ＩＣＶ）を、ユーザが自由にアクセスすることの許可されないホストマシン上の安全なメディア２２０２に格納し、ユーザによる不正なインテグリティ・チェック値（ＩＣＶ）の書き換えを防止した例である。このような構成として、例えばメディア２２０１を装着したデバイスがメディア２２０１の再生を実行する際にホストマシンであるＰＣ、サーバにおいてＩＣＶのチェックを実行して再生の可否を判定する構成とすれば、不正なコピーコンテンツあるいは改竄コンテンツの再生を防止できる。

【０１３２】

［階層ツリー構造のカテゴリ分類］

暗号鍵をルートキー、ノードキー、リーフキー等、図３の階層ツリー構造として構成し、コンテンツキー、認証キー、ＩＣＶ生成キー、あるいはプログラムコード、データ等を有効化キーブロック（ＥＫＢ）とともに暗号化して配信する構成について説明してきたが、ノードキー等を定義している階層ツリー構造を各デバイスのカテゴリ毎に分類して効率的

なキー更新処理、暗号化キー配信、データ配信を実行する構成について、以下説明する。

【０１３３】

図２３に階層ツリー構造のカテゴリの分類の一例を示す。図２３において、階層ツリー構造の最上段には、ルートキーＫｒｏｏｔ２３０１が設定され、以下の中間段にはノードキー２３０２が設定され、最下段には、リーフキー２３０３が設定される。各デバイスは個々のリーフキーと、リーフキーからルートキーに至る一連のノードキー、ルートキーを保有する。

【０１３４】

ここで、一例として最上段から第M段目のあるノードをカテゴリノード２３０４として設定する。すなわち第M段目のノードの各々を特定カテゴリのデバイス設定ノードとする。第M段の１つのノードを頂点として以下、M＋１段以下のノード、リーフは、そのカテゴリに含まれるデバイスに関するノードおよびリーフとする。

【０１３５】

例えば図２３の第M段目の１つのノード２３０５にはカテゴリ［メモリステッイク（商標）］が設定され、このノード以下に連なるノード、リーフはメモリステッイクを使用した様々なデバイスを含むカテゴリ専用のノードまたはリーフとして設定される。すなわち、ノード２３０５以下を、メモリスティックのカテゴリに定義されるデバイスの関連ノード、およびリーフの集合として定義する。

【０１３６】

さらに、M段から数段分下位の段をサブカテゴリノード２３０６として設定することができる。例えば図に示すようにカテゴリ［メモリスティック］ノード２３０５の２段下のノードに、メモリスティックを使用したデバイスのカテゴリに含まれるサブカテゴリノードとして、［再生専用器］のノードを設定する。さらに、サブカテゴリノードである再生専用器のノード２３０６以下に、再生専用器のカテゴリに含まれる音楽再生機能付き電話のノード２３０７が設定され、さらにその下位に、音楽再生機能付き電話のカテゴリに含まれる［ＰＨＳ］ノード２３０８と［携帯電話］ノード２３０９を設定することができる。

【０１３７】

さらに、カテゴリ、サブカテゴリは、デバイスの種類のみならず、例えばあるメーカー、コンテンツプロバイダ、決済機関等が独自に管理するノード、すなわち処理単位、管轄単位、あるいは提供サービス単位等、任意の単位（これらを総称して以下、エンティティと呼ぶ）で設定することが可能である。例えば１つのカテゴリノードをゲーム機器メーカーの販売するゲーム機器ＸＹＺ専用の頂点ノードとして設定すれば、メーカーの販売するゲーム機器ＸＹＺにその頂点ノード以下の下段のノードキー、リーフキーを格納して販売することが可能となり、その後、暗号化コンテンツの配信、あるいは各種キーの配信、更新処理を、その頂点ノードキー以下のノードキー、リーフキーによって構成される有効化キーブロック（ＥＫＢ）を生成して配信し、頂点ノード以下のデバイスに対してのみ利用可能なデータが配信可能となる。

【０１３８】

このように、１つのノードを頂点としして、以下のノードをその頂点ノードに定義されたカテゴリ、あるいはサブカテゴリの関連ノードとして設定する構成とすることにより、カテゴリ段、あるいはサブカテゴリ段の１つの頂点ノードを管理するメーカー、コンテンツプロバイダ等がそのノードを頂点とする有効化キーブロック（ＥＫＢ）を独自に生成して、頂点ノード以下に属するデバイスに配信する構成が可能となり、頂点ノードに属さない他のカテゴリのノードに属するデバイスには全く影響を及ぼさずにキー更新を実行することができる。

【０１３９】

［簡略ＥＫＢによるキー配信構成（１）］

先に説明した例えば図３のツリー構成において、キー、例えばコンテンツキーを所定デバイス（リーフ）宛に送付する場合、キー配布先デバイスの所有しているリーフキー、ノードキーを用いて復号可能な有効化キーブロック（ＥＫＢ）を生成して提供する。例えば図

２４（ａ）に示すツリー構成において、リーフを構成するデバイスａ，ｇ，ｊに対してキー、例えばコンテンツキーを送信する場合、ａ，ｇ，ｊの各ノードにおいて復号可能な有効化キーブロック（ＥＫＢ）を生成して配信する。

【０１４０】

例えば更新ルートキーＫ（ｔ）ｒｏｏｔでコンテンツキーＫ（ｔ）ｃｏｎを暗号化処理し、ＥＫＢとともに配信する場合を考える。この場合、デバイスａ，ｇ，ｊは、それぞれが図２４（ｂ）に示すリーフおよびノードキーを用いて、ＥＫＢの処理を実行してＫ（ｔ）ｒｏｏｔを取得し、取得した更新ルートキーＫ（ｔ）ｒｏｏｔによってコンテンツキーＫ（ｔ）ｃｏｎの復号処理を実行してコンテンツキーを得る。

【０１４１】

この場合に提供される有効化キーブロック（ＥＫＢ）の構成は、図２５に示すようになる。図２５に示す有効化キーブロック（ＥＫＢ）は、先の図６で説明した有効化キーブロック（ＥＫＢ）のフォーマットにしたがって構成されたものであり、データ（暗号化キー）と対応するタグとを持つ。タグは、先に図７を用いて説明したように左（Ｌ）、右（Ｒ）、それぞれの方向にデータがあれば０、無ければ１を示している。

【０１４２】

有効化キーブロック（ＥＫＢ）を受領したデバイスは、有効化キーブロック（ＥＫＢ）の暗号化キーとタグに基づいて、順次暗号化キーの復号処理を実行して上位ノードの更新キーを取得していく。図２５に示すように、有効化キーブロック（ＥＫＢ）は、ルートからリーフまでの段数（デプス）が多いほど、そのデータ量は増加していく。段数（デプス）は、デバイス（リーフ）数に応じて増大するものであり、キーの配信先となるデバイス数が多い場合は、ＥＫＢのデータ量がさらに増大することになる。

【０１４３】

このような有効化キーブロック（ＥＫＢ）のデータ量の削減を可能とした構成について説明する。図２６は、有効化キーブロック（ＥＫＢ）をキー配信デバイスに応じて簡略化して構成した例を示すものである。

【０１４４】

図２５と同様、リーフを構成するデバイスａ，ｇ，ｊに対してキー、例えばコンテンツキーを送信する場合を想定する。図２６の（ａ）に示すように、キー配信デバイスによってのみ構成されるツリーを構築する。この場合、図２４（ｂ）に示す構成に基づいて新たなツリー構成として図２６（ｂ）のツリー構成が構築される。ＫｒｏｏｔからＫｊまでは全く分岐がなく１つの枝のみが存在すればよく、ＫｒｏｏｔからＫａおよびＫｇに至るためには、Ｋ０に分岐点を構成するのみで、２分岐構成の図２６（ａ）のツリーが構築される。

【０１４５】

図２６（ａ）に示すように、ノードとしてＫ０のみを持つ簡略化したツリーが生成される。更新キー配信のための有効化キーブロック（ＥＫＢ）は、これらの簡略ツリーに基づいて生成する。図２６（ａ）に示すツリーは、有効化キーブロック（ＥＫＢ）を復号可能な末端ノードまたはリーフを最下段とした２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーである。更新キー配信のための有効化キーブロック（ＥＫＢ）は、この再構築階層ツリーのノードまたはリーフに対応するキーのみに基づいて構成される。

【０１４６】

先の図２５で説明した有効化キーブロック（ＥＫＢ）は、各リーフａ，ｇ，ｊからＫｒｏｏｔに至るまでのすべてのキーを暗号化したデータを格納していたが、簡略化ＥＫＢは、簡略化したツリーを構成するノードについてのみの暗号化データを格納する。図２６（ｂ）に示すようにタグは３ビット構成を有する。第２および第３ビットは、図２５の例と、同様の意味を持ち、左（Ｌ）、右（Ｒ）、それぞれの方向にデータがあれば０、無ければ１を示す。第１番目のビットは、ＥＫＢ内に暗号化キーが格納されているか否かを示すためのビットであり、データが格納されている場合は１、データが無い場合は、０として設

定される。

【０１４７】

データ通信網、あるいは記憶媒体に格納されてデバイス（リーフ）に提供される有効化キーブロック（ＥＫＢ）は、図２６（ｂ）に示すように、図２５に示す構成に比較すると、データ量が大幅に削減されたものとなる。図２６に示す有効化キーブロック（ＥＫＢ）を受領した各デバイスは、タグの第１ビットに１が格納された部分のデータのみを順次復号することにより、所定の暗号化キーの復号を実現することができる。例えばデバイスａは、暗号化データＥｎｃ（Ｋａ，Ｋ（ｔ）０）をリーフキーＫａで復号して、ノードキーＫ（ｔ）０を取得して、ノードキーＫ（ｔ）０によって暗号化データＥｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）ｒｏｏｔ）を復号してＫ（ｔ）ｒｏｏｔを取得する。デバイスｊは、暗号化データＥｎｃ（Ｋｊ，Ｋ（ｔ）ｒｏｏｔ）をリーフキーＫｊで復号して、Ｋ（ｔ）ｒｏｏｔを取得する。

【０１４８】

このように、配信先のデバイスによってのみ構成される簡略化した新たなツリー構成を構築して、構築されたツリーを構成するリーフおよびノードのキーのみを用いて有効化キーブロック（ＥＫＢ）を生成することにより、少ないデータ量の有効化キーブロック（ＥＫＢ）を生成することが可能となり、有効化キーブロック（ＥＫＢ）のデータ配信が効率的に実行可能となる。

【０１４９】

［簡略ＥＫＢによるキー配信構成（２）］

図２６で示した簡略化したツリーに基づいて生成される有効化キーブロック（ＥＫＢ）をさらに、簡略化してデータ量を削減し、効率的な処理を可能とした構成について説明する。

【０１５０】

図２６を用いて説明した構成は、有効化キーブロック（ＥＫＢ）を復号可能な末端ノードまたはリーフを最下段とした２分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーであった。更新キー配信のための有効化キーブロック（ＥＫＢ）は、この再構築階層ツリーのノードまたはリーフに対応するキーのみに基づいて構成される。

【０１５１】

図２６（ａ）に示す再構築階層ツリーは、リーフａ，ｇ，ｊにおいて更新ルートキーＫ（ｔ）ｒｏｏｔを取得可能とするため、図２６（ｂ）に示す有効化キーブロック（ＥＫＢ）を配信する。図２６（ｂ）の有効化キーブロック（ＥＫＢ）の処理において、リーフｊは、Ｅｎｃ（Ｋｊ，Ｋ（ｔ）ｒｏｏｔ）の１回の復号処理によりルートキー：Ｋ（ｔ）ｒｏｏｔを取得できる。しかし、リーフａ，ｇは、Ｅｎｃ（Ｋａ，Ｋ（ｔ）０）または、Ｅｎｃ（Ｋｇ，Ｋ（ｔ）０）の復号処理によりＫ（ｔ）０を得た後、さらに、Ｅｎｃ（Ｋ（ｔ）０，Ｋ（ｔ）ｒｏｏｔ）の復号処理を実行してルートキー：Ｋ（ｔ）ｒｏｏｔを取得する。すなわち、リーフａ，ｇは、２回の復号処理を実行することが必要となる。

【０１５２】

図２６の簡略化した再構築階層ツリーは、ノードＫ０がその下位リーフａ，ｇの管理ノードとして独自の管理を実行している場合、例えば後述するサブルート・ノードとして、下位リーフの管理を実行している場合には、リーフａ，ｇが更新キーを取得したことを確認する意味で有効であるが、ノードＫ０が下位リーフの管理を行なっていない場合、あるいは行なっていたとしても、上位ノードからの更新キー配信を許容している場合には、図２６（ａ）に示す再構築階層ツリーをさらに簡略化して、ノードＫ０のキーを省略して有効化キーブロック（ＥＫＢ）を生成して配信してもよい。

【０１５３】

図２７に、このような有効化キーブロック（ＥＫＢ）の構成を示す。図２６と同様、リーフを構成するデバイスａ，ｇ，ｊに対してキー、例えばコンテンツキーを送信する場合を想定する。図２７の（ａ）に示すように、ルートＫｒｏｏｔと各リーフａ，ｇ，ｊを直接

接続したツリーを構築する。

【０１５４】

図２７（ａ）に示すように、図２６（ａ）に示す再構築階層ツリーからノードＫ０が省かれた簡略化したツリーが生成される。更新キー配信のための有効化キーブロック（ＥＫＢ）は、これらの簡略ツリーに基づいて生成する。図２７（ａ）に示すツリーは、有効化キーブロック（ＥＫＢ）を復号可能なリーフをとルートとを直接結ぶパスのみによって再構築される再構築階層ツリーである。更新キー配信のための有効化キーブロック（ＥＫＢ）は、この再構築階層ツリーのリーフに対応するキーのみに基づいて構成される。

【０１５５】

なお、図２７（ａ）の例は、末端をリーフとした構成例であるが、例えば最上位ノードか複数の中位、下位ノードに対してキーを配信する場合も、最上位ノードと中位、下位ノードとを直接接続した簡略化ツリーに基づいて有効化キーブロック（ＥＫＢ）を生成してキー配信を実行することが可能である。このように、再構築階層ツリーは、簡略化したツリーを構成する頂点ノードと、簡略化したツリーを構成する末端ノードまたはリーフとを直接、接続した構成を持つ。この簡略化ツリーでは、頂点ノードからの分岐は２に限らず、配信ノードまたはリーフ数に応じて３以上の多分岐を持つツリーとして構成することが可能である。

【０１５６】

先の図２５で説明した有効化キーブロック（ＥＫＢ）は、各リーフａ，ｇ，ｊからＫｒｏｏｔに至るまでのすべてのキーを暗号化したデータを格納し、図２６で説明した有効化キーブロック（ＥＫＢ）は、リーフａ，ｇ，ｊのリーフキー、ａ，ｇの共通ノードとしてのＫ０、さらに、ルートキーを格納した構成であったが、図２７（ａ）に示す簡略化階層ツリーに基づく有効化キーブロック（ＥＫＢ）は、ノードＫ０のキーを省略したので、図２７（ｂ）に示すように、さらにデータ量の少ない有効化キーブロック（ＥＫＢ）となる。

【０１５７】

図２７（ｂ）の有効化キーブロック（ＥＫＢ）は、図２６（ｂ）の有効化キーブロック（ＥＫＢ）と同様、３ビット構成のタグを有する。第２および第３ビットは、図２６で説明したと同様、左（Ｌ）、右（Ｒ）、それぞれの方向にデータがあれば０、無ければ１を示す。第１番目のビットは、ＥＫＢ内に暗号化キーが格納されているか否かを示すためのビットであり、データが格納されている場合は１、データが無い場合は、０として設定される。

【０１５８】

図２７（ｂ）の有効化キーブロック（ＥＫＢ）において、各リーフａ，ｇ，ｊは、Ｅｎｃ（Ｋａ，Ｋ（ｔ）ｒｏｏｔ）、またはＥｎｃ（Ｋｇ，Ｋ（ｔ）ｒｏｏｔ）Ｅｎｃ（Ｋｊ，Ｋ（ｔ）ｒｏｏｔ）の１回の復号処理によりルートキー：Ｋ（ｔ）ｒｏｏｔを取得できる。

【０１５９】

このように簡略化された再構築ツリーの最上位ノードと、ツリーを構成する末端ノードまたはリーフとを直接、接続した構成を持つツリーに基づいて生成される有効化キーブロック（ＥＫＢ）は、図２７（ｂ）に示すように、再構築階層ツリーの頂点ノードおよび末端ノードまたはリーフに対応するキーのみに基づいて構成される。

【０１６０】

図２６または図２７で説明した有効化キーブロック（ＥＫＢ）のように、配信先のデバイスによってのみ構成される簡略化した新たなツリー構成を構築して、構築されたツリーを構成するリーフのみ、あるいはリーフと共通ノードのキーのみを用いて有効化キーブロック（ＥＫＢ）を生成することにより、少ないデータ量の有効化キーブロック（ＥＫＢ）を生成することが可能となり、有効化キーブロック（ＥＫＢ）のデータ配信が効率的に実行可能となる。

【０１６１】

なお、簡略化した階層ツリー構成は、後段で説明するサブツリーとして設定されるカテゴ

リツリー単位のＥＫＢ管理構成において特に有効に活用可能である。カテゴリツリーは、キー配信構成としてのツリー構成を構成するノードあるいはリーフから選択した複数のノードあるいはリーフの集合体ブロックである。カテゴリツリーは、デバイスの種類に応じて設定される集合であったり、あるいはデバイス提供メーカー、コンテンツプロバイダ、決済機関等の管理単位等、ある共通点を持った処理単位、管轄単位、あるいは提供サービス単位等、様々な態様の集合として設定される。１つのカテゴリツリーには、ある共通のカテゴリに分類されるデバイスが集まっており、例えば複数のカテゴリツリーの頂点ノード（サブルート）によって上述したと同様の簡略化したツリーを再構築してＥＫＢを生成することにより、選択されたカテゴリツリーに属するデバイスにおいて復号可能な簡略化された有効化キーブロック（ＥＫＢ）の生成、配信が可能となる。カテゴリツリー単位の管理構成については後段で詳細に説明する。

【０１６２】

なお、このような有効化キーブロック（ＥＫＢ）は、光ディスク、ＤＶＤ等の情報記録媒体に格納した構成とすることが可能である。例えば、上述の暗号化キーデータによって構成されるデータ部と、暗号化キーデータの階層ツリー構造における位置識別データとしてのタグ部とを含む有効化キーブロック（ＥＫＢ）にさらに、更新ノードキーによって暗号化したコンテンツ等のメッセージデータとを格納した情報記録媒体を各デバイスに提供する構成が可能である。デバイスは有効化キーブロック（ＥＫＢ）に含まれる暗号化キーデータをタグ部の識別データにしたがって順次抽出して復号し、コンテンツの復号に必要なキーを取得してコンテンツの利用を行なうことが可能となる。もちろん、有効化キーブロック（ＥＫＢ）をインターネット等のネットワークを介して配信する構成としてもよい。

【０１６３】

［カテゴリツリー単位のＥＫＢ管理構成］

次に、キー配信構成としてのツリー構成を構成するノードあるいはリーフを、複数のノードあるいはリーフの集合としてのブロックで管理する構成について説明する。なお、複数のノードあるいはリーフの集合としてのブロックを以下カテゴリツリーと呼ぶ。カテゴリツリーは、デバイスの種類に応じて設定される集合であったり、あるいはデバイス提供メーカー、コンテンツプロバイダ、決済機関等の管理単位等、ある共通点を持った処理単位、管轄単位、あるいは提供サービス単位等、様々な態様の集合として設定される。

【０１６４】

カテゴリツリーについて、図２８を用いて説明する。図２８（ａ）はツリーのカテゴリツリー単位での管理構成を説明する図である。１つのカテゴリツリーは図では、三角形として示し、例えば１カテゴリツリー２７０１内には、複数のノードが含まれる。１カテゴリツリー内のノード構成を示すのが（ｂ）である。１つのカテゴリツリーは、１つのノードを頂点とした複数段の２分岐形ツリーによって構成される。以下、カテゴリツリーの頂点ノード２７０２をサブルートと呼ぶ。

【０１６５】

ツリーの末端は、（ｃ）に示すようにリーフ、すなわちデバイスによって構成される。デバイスは、複数デバイスをリーフとし、サブルートである頂点ノード２７０２を持つツリーによって構成されるいずれかのカテゴリツリーに属する。

【０１６６】

図２８（ａ）から理解されるように、カテゴリツリーは、階層構造を持つ。この階層構造について、図２９を用いて説明する。

【０１６７】

図２９（ａ）は、階層構造を簡略化して説明するための図であり、Ｋｒｏｏｔから数段下の段にカテゴリツリーＡ０１～Ａｎｎが構成され、カテゴリツリーＡ１～Ａｎの下位には、さらに、カテゴリツリーＢ０１～Ｂｎｋ、さらに、その下位にカテゴリツリーＣ１～Ｃｎｑが設定されている。各カテゴリツリーは、図２９（ｂ），（ｃ）に示す如く、複数段のノード、リーフによって構成されるツリー形状を持つ。

【０１６８】

例えばカテゴリツリーＢｎｋの構成は、（ｂ）に示すように、サブルート２８１１を頂点ノードとして、末端ノード２８１２に至るまでの複数ノードを有する。このカテゴリツリーは識別子Ｂｎｋを持ち、カテゴリツリーＢｎｋ内のノードに対応するノードキー管理をカテゴリツリーＢｎｋ独自に実行することにより、末端ノード２８１２を頂点として設定される下位（子）カテゴリツリーの管理を実行する。また、一方、カテゴリツリーＢｎｋは、サブルート２８１１を末端ノードとして持つ上位（親）カテゴリツリーＡｎｎの管理下にある。

【０１６９】

カテゴリツリーＣｎ３の構成は、（ｃ）に示すように、サブルート２８５１を頂点ノードとして、各デバイスである末端ノード２８５２、この場合はリーフに至るまで複数ノード、リーフを有する。このカテゴリツリーは識別子Ｃｎ３を持ち、カテゴリツリーＣｎ３内のノード、リーフに対応するノードキー、リーフキー管理をカテゴリツリーＣｎ３独自に実行することにより、末端ノード２８５２に対応するリーフ（デバイス）の管理を実行する。また、一方、カテゴリツリーＣｎ３は、サブルート２８５１を末端ノードとして持つ上位（親）カテゴリツリーＢｎ２の管理下にある。各カテゴリツリーにおけるキー管理とは、例えばキー更新処理、リボーク処理等であるが、これらは後段で詳細に説明する。

【０１７０】

最下段カテゴリツリーのリーフであるデバイスには、デバイスの属するカテゴリツリーのリーフキーから、自己の属するカテゴリツリーの頂点ノードであるサブルートノードに至るパスに位置する各ノードのノードキーおよびリーフキーが格納される。例えば末端ノード２８５２のデバイスは、末端ノード（リーフ）２８５２から、サブルートノード２８５１までの各キーを格納する。

【０１７１】

図３０を用いて、さらにカテゴリツリーの構成について説明する。カテゴリツリーは様々な段数によって構成されるツリー構造を持つことが可能である。段数、すなわちデプス（depth）は、カテゴリツリーで管理する末端ノードに対応する下位（子）カテゴリツリーの数、あるいはリーフとしてのデバイス数に応じて設定可能である。

【０１７２】

図３０の（ａ）に示すような上下カテゴリツリー構成を具体化すると、（ｂ）に示す態様となる。ルートツリーは、ルートキーを持つ最上段のツリーである。ルートツリーの末端ノードに複数の下位カテゴリツリーとしてカテゴリツリーＡ，Ｂ，Ｃが設定され、さらに、カテゴリツリーＣの下位カテゴリツリーとしてカテゴリツリーＤが設定される。カテゴリツリーＣ２９０１は、その末端ノードの１つ以上のノードをリザーブノード２９５０として保持し、自己の管理するカテゴリツリーを増加させる場合、さらに複数段のツリー構成を持つカテゴリツリーＣ’２９０２をリザーブノード２９５０を頂点ノードとして新設することにより、管理末端ノード２９７０を増加させて、管理末端ノードに増加した下位カテゴリツリーを追加することができる。

【０１７３】

リザーブノードについて、さらに図３１を用いて説明する。カテゴリツリーＡ，３０１１は、管理する下位カテゴリツリーＢ，Ｃ，Ｄ…を持ち、１つのリザーブノード３０２１を持つ。カテゴリツリーは管理対象の下位カテゴリツリーをさらに増加させたい場合、リザーブノード３０２１に、自己管理の下位カテゴリツリーＡ’，３０１２を設定し、下位カテゴリツリーＡ’，３０１２の末端ノードにさらに管理対象の下位カテゴリツリーＦ，Ｇを設定することができる。自己管理の下位カテゴリツリーＡ’，３０１２も、その末端ノードの少なくとも１つをリザーブノード３０２２として設定することにより、さらに下位カテゴリツリーＡ’’３０１３を設定して、さらに管理カテゴリツリーを増加させることができる。下位カテゴリツリーＡ’’３０１３の末端ノードにも１以上のリザーブノードを確保する。このようなリザーブノード保有構成をとることにより、あるカテゴリツリーの管理する下位カテゴリツリーは、際限なく増加させることが可能となる。なお、リザーブカテゴリツリーは、末端ノードの１つのみではなく、複数個設定する構成としてもよい

。
【０１７４】
それぞれのカテゴリツリーでは、カテゴリツリー単位で有効化キーブロック（ＥＫＢ）が構成され、カテゴリツリー単位でのキー更新、リボーク処理を実行することになる。図３１のように複数のカテゴリツリーＡ，Ａ’，Ａ’’には各カテゴリツリー個々の有効化キーブロック（ＥＫＢ）が設定されることになるが、これらは、カテゴリツリーＡ，Ａ’，Ａ’’を共通に管理する例えばあるデバイスメーカーが一括して管理することが可能である。
【０１７５】
［新規カテゴリツリーの登録処理］
次に、新規カテゴリツリーの登録処理について説明する。登録処理シーケンスを図３２に示す。図３２のシーケンスにしたがって説明する。新たにツリー構成中に追加される新規（子）カテゴリツリー（Ｎ－Ｅｎ）は、上位（親）カテゴリツリー（Ｐ－Ｅｎ）に対して新規登録要求を実行する。なお、各カテゴリツリーは、公開鍵暗号方式に従った公開鍵を保有し、新規カテゴリツリーは自己の公開鍵を登録要求に際して上位カテゴリツリー（Ｐ－Ｅｎ）に送付する。
【０１７６】
登録要求を受領した上位カテゴリツリー（Ｐ－Ｅｎ）は、受領した新規（子）カテゴリツリー（Ｎ－Ｅｎ）の公開鍵を証明書発行局（ＣＡ：Certificate Authority）に転送し、ＣＡの署名を付加した新規（子）カテゴリツリー（Ｎ－Ｅｎ）の公開鍵を受領する。これらの手続きは、上位カテゴリツリー（Ｐ－Ｅｎ）と新規（子）カテゴリツリー（Ｎ－Ｅｎ）との相互認証の手続きとして行われる。
【０１７７】
これらの処理により、新規登録要求カテゴリツリーの認証が終了すると、上位カテゴリツリー（Ｐ－Ｅｎ）は、新規（子）カテゴリツリー（Ｎ－Ｅｎ）の登録を許可し、新規（子）カテゴリツリー（Ｎ－Ｅｎ）のノードキーを新規（子）カテゴリツリー（Ｎ－Ｅｎ）に送信する。このノードキーは、上位カテゴリツリー（Ｐ－Ｅｎ）の末端ノードの１つのノードキーであり、かつ、新規（子）カテゴリツリー（Ｎ－Ｅｎ）の頂点ノード、すなわちサブルートキーに対応する。
【０１７８】
このノードキー送信が終了すると、新規（子）カテゴリツリー（Ｎ－Ｅｎ）は、新規（子）カテゴリツリー（Ｎ－Ｅｎ）のツリー構成を構築し、構築したツリーの頂点に受信した頂点ノードのサブルートキーを設定し、各ノード、リーフのキーを設定して、カテゴリツリー内の有効化キーブロック（ＥＫＢ）を生成する。１つのカテゴリツリー内の有効化キーブロック（ＥＫＢ）をサブＥＫＢと呼ぶ。
【０１７９】
一方、上位カテゴリツリー（Ｐ－Ｅｎ）は、新規（子）カテゴリツリー（Ｎ－Ｅｎ）の追加により、有効化する末端ノードを追加した上位カテゴリツリー（Ｐ－Ｅｎ）内のサブＥＫＢを生成する。
【０１８０】
新規（子）カテゴリツリー（Ｎ－Ｅｎ）は、新規（子）カテゴリツリー（Ｎ－Ｅｎ）内のノードキー、リーフキーによって構成されるサブＥＫＢを生成すると、これを上位カテゴリツリー（Ｐ－Ｅｎ）に送信する。
【０１８１】
新規（子）カテゴリツリー（Ｎ－Ｅｎ）からサブＥＫＢを受信した上位カテゴリツリー（Ｐ－Ｅｎ）は、受信したサブＥＫＢと、上位カテゴリツリー（Ｐ－Ｅｎ）の更新したサブＥＫＢとをキー発行センター（ＫＤＣ：Key Distribute Center）に送信する。
【０１８２】
キー発行センター（ＫＤＣ）は、すべてのカテゴリツリーのサブＥＫＢに基づいて、様々な態様のＥＫＢ、すなわち特定のカテゴリツリーあるいはデバイスのみが復号可能なＥＫ

Ｂを生成することが可能となる。このように復号可能なカテゴリツリーあるいはデバイスを設定したＥＫＢを例えばコンテンツプロバイダに提供し、コンテンツプロバイダがＥＫＢに基づいてコンテンツキーを暗号化して、ネットワークを介して、あるいは記録媒体に格納して提供することにより、特定のデバイスでのみ利用可能なコンテンツを提供することが可能となる。

【０１８３】

なお、新規カテゴリツリーのサブＥＫＢのキー発行センター（ＫＤＣ）に対する登録処理は、サブＥＫＢを上位カテゴリツリーを介してを順次転送して実行する方法に限るものではなく、上位カテゴリツリーを介さずに、新規登録カテゴリツリーから直接、キー発行センター（ＫＤＣ）に登録する処理を実行する構成としてもよい。

【０１８４】

上位カテゴリツリーと、上位カテゴリツリーに新規追加する下位カテゴリツリーとの対応について図３３を用いて説明する。上位カテゴリツリーの末端ノードの１つ３２０１を新規追加カテゴリツリーの頂点ノードとして、下位カテゴリツリーに提供することによって下位カテゴリツリーは、上位カテゴリツリーの管理下のカテゴリツリーとして追加される。上位カテゴリツリーの管理下のカテゴリツリーとは、後段で詳細に説明するが、下位カテゴリツリーのリボーク（排除）処理を上位カテゴリツリーが実行できる構成であるという意味を含むものである。

【０１８５】

図３３に示すように、上位カテゴリツリーに新規カテゴリツリーが設定されると、上位カテゴリツリーのリーフである末端ノードの１つのノード３２０１と新規追加カテゴリツリーの頂点ノード３２０２とが等しいノードとして設定される。すなわち上位ノードの１つのリーフである１つの末端ノードが、新規追加カテゴリツリーのサブルートとして設定される。このように設定されることにより、新規追加カテゴリツリーが全体ツリー構成の下で有効化される。

【０１８６】

図３４に新規追加カテゴリツリーを設定した際に上位カテゴリツリーが生成する更新ＥＫＢの例を示す。図３４は、（ａ）に示す構成、すなわち既に有効に存在する末端ノード（ｎｏｄｅ０００）３３０１と末端ノード（ｎｏｄｅ００１）３３０２があり、ここに新規追加カテゴリツリーに新規カテゴリツリー追加末端ノード（ｎｏｄｅ１００）３３０３を付与した際に上位カテゴリツリーが生成するサブＥＫＢの例を示したものである。

【０１８７】

サブＥＫＢは、図３４の（ｂ）に示すようにな構成を持つ。それぞれ有効に存在する末端ノードキーにより暗号化された上位ノードキー、上位ノードキーで暗号化されたさらなる上位ノードキー、…さらに上位に進行してサブルートキーに至る構成となっている。この構成によりサブＥＫＢが生成される。各カテゴリツリーは図３４（ｂ）に示すと同様、有効な末端ノード、あるいはリーフキーにより暗号化された上位ノードキー、上位ノードキーでさらに上位のノードキーを暗号化し、順次上位に深厚してサブルートに至る暗号化データによって構成されるＥＫＢを有し、これを管理する。

【０１８８】

［カテゴリツリー管理下におけるリボーク処理］

次に、キー配信ツリー構成をカテゴリツリー単位として管理する構成におけるデバイスあるいはカテゴリツリーのリボーク（排除）処理について説明する。先の図３，４では、ツリー構成全体の中から特定のデバイスのみ復号可能で、リボークされたデバイスは復号不可能な有効化キーブロック（ＥＫＢ）を配信する処理について説明した。図３，４で説明したリボーク処理は、ツリー全体の中から特定のリーフであるデバイスをリボークする処理であったが、ツリーのカテゴリツリー管理による構成では、カテゴリツリー毎にリボーク処理が実行可能となる。

【０１８９】

図３５以下の図を用いてカテゴリツリー管理下のツリー構成におけるリボーク処理につい

て説明する。図３５は、ツリーを構成するカテゴリツリーのうち、最下段のカテゴリツリー、すなわち個々のデバイスを管理しているカテゴリツリーによるデバイスのリボーク処理を説明する図である。

【０１９０】

図３５（ａ）は、カテゴリツリー管理によるキー配信ツリー構造を示している。ツリー最上位にはルートノードが設定され、その数段下にカテゴリツリーＡ０１～Ａｎｎ、さらにその下位段にＢ０１～Ｂｎｋのカテゴリツリー、さらにその下位段にＣ１～ｃｎのカテゴリツリーが構成されている。最も下のカテゴリツリーは、末端ノード（リーフ）が個々のデバイス、例えば記録再生器、再生専用器等であるとする。

【０１９１】

ここで、リボーク処理は、各カテゴリツリーにおいて独自に実行される。例えば、最下段のカテゴリツリーＣ１～Ｃｎでは、リーフのデバイスのリボーク処理が実行される。図３５（ｂ）には、最下段のカテゴリツリーの１つであるカテゴリツリーＣｎ，３４３０のツリー構成を示している。カテゴリツリーＣｎ，３４３０は、頂点ノード３４３１を持ち、末端ノードであるリーフに複数のデバイスを持つ構成である。

【０１９２】

この末端ノードであるリーフ中に、リボーク対象となるデバイス、例えばデバイス３４３２があったとすると、カテゴリツリーＣｎ，３４３０は、独自に更新したカテゴリツリーＣｎ内のノードキー、リーフキーによって構成される有効化キーブロック（サブＥＫＢ）を生成する。この有効化キーブロックは、リボークデバイス３４３２においては復号できず、他のリーフを構成するデバイスにおいてのみ復号可能な暗号化キーにより構成されるキーブロックである。カテゴリツリーＣｎの管理者は、これを更新されたサブＥＫＢとして生成する。具体的には、サブルートからリボークデバイス３４３２に連なるパスを構成する各ノード３４３１，３４３４，３４３５のノードキーを更新して、この更新ノードキーをリボークデバイス３４３２以外のリーフデバイスにおいてのみ復号可能な暗号化キーとして構成したブロックを更新サブＥＫＢとする。この処理は、先の図３，４において説明したリボーク処理構成において、ルートキーを、カテゴリツリーの頂点キーであるサブルートキーに置き換えた処理に対応する。

【０１９３】

このようにカテゴリツリーＣｎ，３４３０がリボーク処理によって更新した有効化キーブロック（サブＥＫＢ）は、上位カテゴリツリーに送信される。この場合、上位カテゴリツリーはカテゴリツリーＢｎｋ，３４２０であり、カテゴリツリーＣｎ，３４３０の頂点ノード３４３１を末端ノードとして有するカテゴリツリーである。

【０１９４】

カテゴリツリーＢｎｋ，３４２０は、下位カテゴリツリーＣｎ，３４３０から有効化キーブロック（サブＥＫＢ）を受領すると、そのキーブロックに含まれるカテゴリツリーＣｎｋ，３４３０の頂点ノード３４３１に対応するカテゴリツリーＢｎｋ，３４２０の末端ノード３４３１を、下位カテゴリツリーＣｎ，３４３０において更新されたキーに設定して、自身のカテゴリツリーＢｎｋ，３４２０のサブＥＫＢの更新処理を実行する。図３５（ｃ）にカテゴリツリーＢｎｋ，３４２０のツリー構成を示す。カテゴリツリーＢｎｋ，３４２０において、更新対象となるノードキーは、図３５（ｃ）のサブルート３４２１からリボークデバイスを含むカテゴリツリーを構成する末端ノード３４３１に至るパス上のノードキーである。すなわち、更新サブＥＫＢを送信してきたカテゴリツリーのノード３４３１に連なるパスを構成する各ノード３４２１，３４２４，３４２５のノードキーが更新対象となる。これら各ノードのノードキーを更新してカテゴリツリーＢｎｋ，３４２０の新たな更新サブＥＫＢを生成する。

【０１９５】

さらに、カテゴリツリーＢｎｋ，３４２０が更新した有効化キーブロック（サブＥＫＢ）は、上位カテゴリツリーに送信される。この場合、上位カテゴリツリーはカテゴリツリーＡｎｎ，３４１０であり、カテゴリツリーＢｎｋ，３４２０の頂点ノード３４２１を末端

ノードとして有するカテゴリツリーである。

【０１９６】

カテゴリツリーＡｎｎ，３４１０は、下位カテゴリツリーＢｎｋ，３４２０から有効化キーブロック（サブＥＫＢ）を受領すると、そのキーブロックに含まれるカテゴリツリーＢｎｋ，３４２０の頂点ノード３４２１に対応するカテゴリツリーＡｎｎ，３４１０の末端ノード３４２１を、下位カテゴリツリーＢｎｋ，３４２０において更新されたキーに設定して、自身のカテゴリツリーＡｎｎ，３４１０のサブＥＫＢの更新処理を実行する。図３５（ｄ）にカテゴリツリーＡｎｎ，３４１０のツリー構成を示す。カテゴリツリーＡｎｎ，３４１０において、更新対象となるノードキーは、図３５（ｄ）のサブルート３４１１から更新サブＥＫＢを送信してきたカテゴリツリーのノード３４２１に連なるパスを構成する各ノード３４１１，３４１４，３４１５のノードキーである。これら各ノードのノードキーを更新してカテゴリツリーＡｎｎ，３４１０の新たな更新サブＥＫＢを生成する。

【０１９７】

これらの処理を順次、上位のカテゴリツリーにおいて実行し、図３０（ｂ）で説明したルートカテゴリツリーまで実行する。この一連の処理により、デバイスのリボーク処理が完結する。なお、それぞれのカテゴリツリーにおいて更新されたサブＥＫＢは、最終的にキー発行センター（ＫＤＣ）に送信され、保管される。キー発行センター（ＫＤＣ）は、すべてのカテゴリツリーの更新サブＥＫＢに基づいて、様々なＥＫＢを生成する。更新ＥＫＢは、リボークされたデバイスでの復号が不可能な暗号化キーブロックとなる。

【０１９８】

デバイスのリボーク処理のシーケンス図を図３６に示す。処理手順を図３６のシーケンス図に従って説明する。まず、ツリー構成の最下段にあるデバイス管理カテゴリツリー（Ｄ－Ｅｎ）は、デバイス管理カテゴリツリー（Ｄ－Ｅｎ）内のリボーク対象のリーフを排除するために必要なキー更新を行ない、デバイス管理カテゴリツリー（Ｄ－Ｅｎ）の新たなサブＥＫＢ（Ｄ）を生成する。更新サブＥＫＢ（Ｄ）は、上位カテゴリツリーに送付される。更新サブＥＫＢ（Ｄ）を受領した上位（親）カテゴリツリー（Ｐ１－Ｅｎ）は、更新サブＥＫＢ（Ｄ）の更新頂点ノードに対応した末端ノードキーの更新および、その末端ノードからサブルートに至るパス上のノードキーを更新した更新サブＥＫＢ（Ｐ１）を生成する。これらの処理を順次、上位カテゴリツリーにおいて実行して、最終的に更新されたすべてのサブＥＫＢがキー発行センター（ＫＤＣ）に格納され管理される。

【０１９９】

図３７にデバイスのリボーク処理によって上位カテゴリツリーが更新処理を行なって生成する有効化キーブロック（ＥＫＢ）の例を示す。

【０２００】

図３７は、（ａ）に示す構成において、リボークデバイスを含む下位カテゴリツリーから更新サブＥＫＢを受信した上位カテゴリツリーにおいて生成するＥＫＢの例を説明する図である。リボークデバイスを含む下位カテゴリツリーの頂点ノードは、上位カテゴリツリーの末端ノード（ｎｏｄｅ１００）３６０１に対応する。

【０２０１】

上位カテゴリツリーは、上位カテゴリツリーのサブルートから末端ノード（ｎｏｄｅ１００）３６０１までのパスに存在するノードキーを更新して新たな更新サブＥＫＢを生成する。更新サブＥＫＢは、図３７（ｂ）のようになる。更新されたキーは、下線および［’］を付して示してある。このように更新された末端ノードからサブルートまでのパス上のノードキーを更新してそのカテゴリツリーにおける更新サブＥＫＢとする。

【０２０２】

次に、リボークする対象をカテゴリツリーとした場合の処理、すなわちカテゴリツリーのリボーク処理について説明する。

【０２０３】

図３８（ａ）は、カテゴリツリー管理によるキー配信ツリー構造を示している。ツリー最上位にはルートノードが設定され、その数段下にカテゴリツリーＡ０１～Ａｎｎ、さらに

その下位段にＢ０１～Ｂｎｋのカテゴリツリー、さらにその下位段にＣ１～ｃｎのカテゴリツリーが構成されている。最も下のカテゴリツリーは、末端ノード（リーフ）が個々のデバイス、例えば記録再生器、再生専用器等であるとする。

【０２０４】

ここで、リボーク処理を、カテゴリツリーＣｎ，３７３０に対して実行する場合について説明する。最下段のカテゴリツリーＣｎ，３７３０は、図３８（ｂ）に示すように頂点ノード３４３１を持ち、末端ノードであるリーフに複数のデバイスを持つ構成である。

【０２０５】

カテゴリツリーＣｎ，３７３０をリボークすることにより、カテゴリツリーＣｎ，３７３０に属するすべてのデバイスのツリー構造からの一括排除が可能となる。カテゴリツリーＣｎ，３７３０のリボーク処理は、カテゴリツリーＣｎ，３７３０の上位カテゴリツリーであるカテゴリツリーＢｎｋ，３７２０において実行される。カテゴリツリーＢｎｋ，３７２０は、カテゴリツリーＣｎ，３７３０の頂点ノード３７３１を末端ノードとして有するカテゴリツリーである。

【０２０６】

カテゴリツリーＢｎｋ，３７２０は、下位カテゴリツリーＣｎ，３７３０のリボークを実行する場合、カテゴリツリーＣｎｋ，３７３０の頂点ノード３７３１に対応するカテゴリツリーＢｎｋ，３７２０の末端ノード３７３１を更新し、さらに、そのリボークカテゴリツリー３７３０からカテゴリツリーＢｎｋ，３７２０のサブルートまでのパス上のノードキーの更新を行ない有効化キーブロックを生成して更新サブＥＫＢを生成する。更新対象となるノードキーは、図３８（ｃ）のサブルート３７２１からリボークカテゴリツリーの頂点ノードを構成する末端ノード３７３１に至るパス上のノードキーである。すなわち、ノード３７２１，３７２４，３７２５，３７３１のノードキーが更新対象となる。これら各ノードのノードキーを更新してカテゴリツリーＢｎｋ，３７２０の新たな更新サブＥＫＢを生成する。

【０２０７】

あるいは、カテゴリツリーＢｎｋ，３７２０は、下位カテゴリツリーＣｎ，３７３０のリボークを実行する場合、カテゴリツリーＣｎｋ，３７３０の頂点ノード３７３１に対応するカテゴリツリーＢｎｋ，３７２０の末端ノード３７３１は更新せず、そのリボークカテゴリツリー３７３０からカテゴリツリーＢｎｋ，３７２０のサブルートまでのパス上の末端ノード３７３１を除くノードキーの更新を行ない有効化キーブロックを生成して更新サブＥＫＢを生成してもよい。

【０２０８】

さらに、カテゴリツリーＢｎｋ，３７２０が更新した有効化キーブロック（サブＥＫＢ）は、上位カテゴリツリーに送信される。この場合、上位カテゴリツリーはカテゴリツリーＡｎｎ，３７１０であり、カテゴリツリーＢｎｋ，３７２０の頂点ノード３７２１を末端ノードとして有するカテゴリツリーである。

【０２０９】

カテゴリツリーＡｎｎ，３７１０は、下位カテゴリツリーＢｎｋ，３７２０から有効化キーブロック（サブＥＫＢ）を受領すると、そのキーブロックに含まれるカテゴリツリーＢｎｋ，３７２０の頂点ノード３７２１に対応するカテゴリツリーＡｎｎ，３７１０の末端ノード３７２１を、下位カテゴリツリーＢｎｋ，３７２０において更新されたキーに設定して、自身のカテゴリツリーＡｎｎ，３７１０のサブＥＫＢの更新処理を実行する。図３８（ｄ）にカテゴリツリーＡｎｎ，３７１０のツリー構成を示す。カテゴリツリーＡｎｎ，３７１０において、更新対象となるノードキーは、図３８（ｄ）のサブルート３７１１から更新サブＥＫＢを送信してきたカテゴリツリーのノード３７２１に連なるパスを構成する各ノード３７１１，３７１４，３７１５のノードキーである。これら各ノードのノードキーを更新してカテゴリツリーＡｎｎ，３７１０の新たな更新サブＥＫＢを生成する。

【０２１０】

これらの処理を順次、上位のカテゴリツリーにおいて実行し、図３０（ｂ）で説明したル

ートカテゴリツリーまで実行する。この一連の処理により、カテゴリツリーのリボーク処理が完結する。なお、それぞれのカテゴリツリーにおいて更新されたサブＥＫＢは、最終的にキー発行センター（ＫＤＣ）に送信され、保管される。キー発行センター（ＫＤＣ）は、すべてのカテゴリツリーの更新サブＥＫＢに基づいて、様々なＥＫＢを生成する。更新ＥＫＢは、リボークされたカテゴリツリーに属するデバイスでの復号が不可能な暗号化キーブロックとなる。

【０２１１】

カテゴリツリーのリボーク処理のシーケンス図を図３９に示す。処理手順を図３９のシーケンス図に従って説明する。まず、カテゴリツリーをリボークしようとするカテゴリツリー管理カテゴリツリー（Ｅ－Ｅｎ）は、カテゴリツリー管理カテゴリツリー（Ｅ－Ｅｎ）内のリボーク対象の末端ノードを排除するために必要なキー更新を行ない、カテゴリツリー管理カテゴリツリー（Ｅ－Ｅｎ）の新たなサブＥＫＢ（Ｅ）を生成する。更新サブＥＫＢ（Ｅ）は、上位カテゴリツリーに送付される。更新サブＥＫＢ（Ｅ）を受領した上位（親）カテゴリツリー（Ｐ１－Ｅｎ）は、更新サブＥＫＢ（Ｅ）の更新頂点ノードに対応した末端ノードキーの更新および、その末端ノードからサブルートに至るパス上のノードキーを更新した更新サブＥＫＢ（Ｐ１）を生成する。これらの処理を順次、上位カテゴリツリーにおいて実行して、最終的に更新されたすべてのサブＥＫＢがキー発行センター（ＫＤＣ）に格納され管理される。キー発行センター（ＫＤＣ）は、すべてのカテゴリツリーの更新サブＥＫＢに基づいて、様々なＥＫＢを生成する。更新ＥＫＢは、リボークされたカテゴリツリーに属するデバイスでの復号が不可能な暗号化キーブロックとなる。

【０２１２】

図４０にリボークされた下位カテゴリツリーと、リボークを行なった上位カテゴリツリーの対応を説明する図を示す。上位カテゴリツリーの末端ノード３９０１は、カテゴリツリーのリボークにより更新され、上位カテゴリツリーのツリーにおける末端ノード３９０１からサブルートまでのパスに存在するノードキーの更新により、新たなサブＥＫＢが生成される。その結果、リボークされた下位カテゴリツリーの頂点ノード３９０２のノードキーと、上位カテゴリツリーの末端ノード３９０１のノードキーは不一致となる。カテゴリツリーのリボーク後にキー発行センター（ＫＤＣ）によって生成されるＥＫＢは、上位カテゴリツリーにおいて更新された末端ノード３９０１のキーに基づいて生成されることになるので、その更新キーを保有しない下位カテゴリツリーのリーフに対応するデバイスは、キー発行センター（ＫＤＣ）によって生成されるＥＫＢの復号ガ不可能になる。

【０２１３】

なお、上述の説明では、デバイスを管理する最下段のカテゴリツリーのリボーク処理について説明したが、ツリーの中段にあるカテゴリツリー管理カテゴリツリーをその上位カテゴリツリーがリボークする処理も上記と同様のプロセスによって可能である。中段のエンティティ管理カテゴリツリーをリボークすることにより、リボークされたカテゴリツリー管理カテゴリツリーの下位に属するすべての複数カテゴリツリーおよびデバイスを一括してリボーク可能となる。

【０２１４】

このように、カテゴリツリー単位でのリボークを実行することにより、１つ１つのデバイス単位で実行するリボーク処理に比較して簡易なプロセスでのリボーク処理が可能となる。

【０２１５】

［カテゴリツリーのケイパビリティ管理］

次に、カテゴリツリー単位でのキー配信ツリー構成において、各エンティティの許容するケイパビリティ(Capability)を管理して、ケイパビリティに応じたコンテンツ配信を行なう処理構成について説明する。ここでケイパビリティとは、例えば特定の圧縮音声データの復号が可能であるとか、特定の音声再生方式を許容するとか、あるいは特定の画像処理プログラムを処理できる等、デバイスがどのようなコンテンツ、あるいはプログラム等を処理できるデバイスであるか、すなわちデバイスのデータ処理能力の定義情報である。

【０２１６】
図４１にケイパビリティを定義したカテゴリツリー構成例を示す。キー配信ツリー構成の最頂点ににルートノードが位置し、下層に複数のカテゴリツリーが接続されて各ノードが２分岐を持つツリー構成である。ここで、例えばカテゴリツリー４００１は、音声再生方式Ａ，Ｂ，Ｃのいずれかを許容するケイパビリティを持つカテゴリツリーとして定義される。具体的には、例えばある音声圧縮プログラム－Ａ、Ｂ、またはＣ方式で圧縮した音楽データを配信した場合に、カテゴリツリー４００１以下に構成されたカテゴリツリーに属するデバイスは圧縮データを伸長する処理が可能である。
【０２１７】
同様にカテゴリツリー４００２は音声再生方式ＢまたはＣ、カテゴリツリー４００３は音声再生方式ＡまたはＢ、カテゴリツリー４００４は音声再生方式Ｂ、カテゴリツリー４００５は音声再生方式Ｃを処理することが可能なケイパビリティを持つカテゴリツリーとして定義される。
【０２１８】
一方、カテゴリツリー４０２１は、画像再生方式ｐ，ｑ，ｒを許容するカテゴリツリーとして定義され、カテゴリツリー４０２２は方式ｐ，ｑの画像再生方式、カテゴリツリー４０２３は方式ｐの画像再生が可能なケイパビリティを持つカテゴリツリーとして定義される。
【０２１９】
このような各カテゴリツリーのケイパビリティ情報は、キー発行センター（ＫＤＣ）において管理される。キー発行センター（ＫＤＣ）は、例えばあるコンテンツプロバイダが特定の圧縮プログラムで圧縮した音楽データを様々なデバイスに配信したい場合、その特定の圧縮プログラムを再生可能なデバイスに対してのみ復号可能な有効化キーブロック（ＥＫＢ）を各カテゴリツリーのケイパビリティ情報に基づいて生成することができる。コンテンツを提供するコンテンツプロバイダは、ケイパビリティ情報に基づいて生成した有効化キーブロック（ＥＫＢ）によって暗号化したコンテンツキーを配信し、そのコンテンツキーで暗号化した圧縮音声データを各デバイスに提供する。この構成により、データの処理が可能なデバイスに対してのみ特定の処理プログラムを確実に提供することが可能となる。
【０２２０】
なお、図４１では全てのカテゴリツリーについてケイパビリティ情報を定義している構成であるが、図４１の構成ようにすべてのカテゴリツリーにケイパビリティ情報を定義することは必ずしも必要ではなく、例えば図４２に示すようにデバイスが属する最下段のカテゴリツリーについてのみケイパビリティを定義して、最下段のカテゴリツリーに属するデバイスのケイパビリティをキー発行センター（ＫＤＣ）において管理して、コンテンツプロバイダが望む処理の可能なデバイスにのみ復号可能な有効化キーブロック（ＥＫＢ）を最下段のカテゴリツリーに定義されたケイパビリティ情報に基づいて生成する構成としてもよい。図４２では、末端ノードにデバイスが定義されたカテゴリツリー４１０１＝４１０５におけるケイパビリテイが定義され、これらのカテゴリツリーについてのケイパビリティをキー発行センター（ＫＤＣ）において管理する構成である。例えばカテゴリツリー４１０１には音声再生については方式Ｂ、画像再生については方式ｒの処理が可能なデバイスが属している。カテゴリツリー４１０２には音声再生については方式Ａ、画像再生については方式ｑの処理が可能なデバイスが属している等である。
【０２２１】
図４３にキー発行センター（ＫＤＣ）において管理するケイパビリティ管理テーブルの構成例を示す。ケイパビリティ管理テーブルは、図４３（ａ）のようなデータ構成を持つ。すなわち、各カテゴリツリーを識別する識別子としてのカテゴリツリーＩＤ、そのカテゴリツリーに定義されたケイパビリティを示すケイパビリティリスト、このケイパビリティリストは図４３（ｂ）に示すように、例えば音声データ再生処理方式（Ａ）が処理可能であれば［１］、処理不可能であれは［０］、音声データ再生処理方式（Ｂ）が処理可能で

あれば［１］、処理不可能であれは［０］…等、様々な態様のデータ処理についての可否を１ビットづつ［１］または［０］を設定して構成されている。なお、このケイパビリティ情報の設定方法はこのような形式に限らず、カテゴリツリーの管理デバイスについてのケイパビリティを識別可能であれば他の構成でもよい。

【０２２２】

ケイパビリティ管理テーブルには、さらに、各カテゴリツリーのサブＥＫＢ、あるいはサブＥＫＢが別のデータベースに格納されている場合は、サブＥＫＢの識別情報が格納され、さらに、各カテゴリツリーのサブルートノード識別データが格納される。

【０２２３】

キー発行センター（ＫＤＣ）は、ケイパビリティ管理テーブルに基づいて、例えば特定のコンテンツの再生可能なデバイスのみが復号可能な有効化キーブロック（ＥＫＢ）を生成する。図４４を用いて、ケイパビリティ情報に基づく有効化キーブロックの生成処理について説明する。

【０２２４】

まず、ステップＳ４３０１において、キー発行センター（ＫＤＣ）は、ケイパビリティ管理テーブルから、指定されたケイパビリティを持つカテゴリツリーを選択する。具体的には、例えばコンテンツプロバイダが音声データ再生処理方式Ａに基づく再生可能なデータを配信したい場合は、図４３（ａ）のケイパビリティリストから、例えば音声データ再生処理（方式Ａ）の項目が［１］に設定されたカテゴリツリーを選択する。

【０２２５】

次に、ステップＳ４３０２において、選択されたカテゴリツリーによって構成される選択カテゴリツリーＩＤのリストを生成する。次に、ステップＳ４３０３で、選択カテゴリツリーＩＤによって構成されるツリーに必要なパス（キー配信ツリー構成のパス）を選択する。ステップＳ４３０４では、選択カテゴリツリーＩＤのリストに含まれる全てのパス選択が完了したか否かを判定し、完了するまで、ステップＳ４３０３においてパスを生成する。これは、複数のカテゴリツリーが選択された場合に、それぞれのパスを順次選択する処理を意味している。

【０２２６】

選択カテゴリツリーＩＤのリストに含まれる全てのパス選択が完了すると、ステップＳ４３０５に進み、選択したパスと、選択カテゴリツリーによってのみ構成されるキー配信ツリー構造を構築する。

【０２２７】

次に、ステップＳ４３０６において、ステップＳ４３０５で生成したツリー構造のノードキーの更新処理を行ない、更新ノードキーを生成する。さらに、ツリーを構成する選択カテゴリツリーのサブＥＫＢをケイパビリティ管理テーブルから取り出し、サブＥＫＢと、ステップＳ４３０６で生成した更新ノードキーとに基づいて選択カテゴリツリーのデバイスにおいてのみ復号可能な有効化キーブロック（ＥＫＢ）を生成する。このようにして生成した有効化キーブロック（ＥＫＢ）は、特定のケイパビリティを持つデバイスにおいてのみ利用、すなわち復号可能な有効化キーブロック（ＥＫＢ）となる。この有効化キーブロック（ＥＫＢ）で例えばコンテンツキーを暗号化して、そのコンテンツキーで特定プログラムに基づいて圧縮したコンテンツを暗号化してデバイスに提供することで、キー発行センター（ＫＤＣ）によって選択された特定の処理可能なデバイスにおいてのみコンテンツが利用される。

【０２２８】

このようにキー発行センター（ＫＤＣ）は、ケイパビリティ管理テーブルに基づいて、例えば特定のコンテンツの再生可能なデバイスのみが復号可能な有効化キーブロック（ＥＫＢ）を生成する。従って、新たなカテゴリツリーが登録される場合には、その新規登録カテゴリツリーのケイパビリティを予め取得することが必要となる。このカテゴリツリー新規登録に伴うケイパビリティ通知処理について図４５を用いて説明する。

【０２２９】

図４５は、新規カテゴリツリーがキー配信ツリー構成に参加する場合のケイパビリティ通知処理シーケンスを示した図である。

【０２３０】

新たにツリー構成中に追加される新規（子）カテゴリツリー（Ｎ－Ｅｎ）は、上位（親）カテゴリツリー（Ｐ－Ｅｎ）に対して新規登録要求を実行する。なお、各カテゴリツリーは、公開鍵暗号方式に従った公開鍵を保有し、新規カテゴリツリーは自己の公開鍵を登録要求に際して上位カテゴリツリー（Ｐ－Ｅｎ）に送付する。

【０２３１】

登録要求を受領した上位カテゴリツリー（Ｐ－Ｅｎ）は、受領した新規（子）カテゴリツリー（Ｎ－Ｅｎ）の公開鍵を証明書発行局（ＣＡ：Certificate Authority）に転送し、ＣＡの署名を付加した新規（子）カテゴリツリー（Ｎ－Ｅｎ）の公開鍵を受領する。これらの手続きは、上位カテゴリツリー（Ｐ－Ｅｎ）と新規（子）カテゴリツリー（Ｎ－Ｅｎ）との相互認証の手続きとして行われる。

【０２３２】

これらの処理により、新規登録要求カテゴリツリーの認証が終了すると、上位カテゴリツリー（Ｐ－Ｅｎ）は、新規（子）カテゴリツリー（Ｎ－Ｅｎ）の登録を許可し、新規（子）カテゴリツリー（Ｎ－Ｅｎ）のノードキーを新規（子）カテゴリツリー（Ｎ－Ｅｎ）に送信する。このノードキーは、上位カテゴリツリー（Ｐ－Ｅｎ）の末端ノードの１つのノードキーであり、かつ、新規（子）カテゴリツリー（Ｎ－Ｅｎ）の頂点ノード、すなわちサブルートキーに対応する。

【０２３３】

このノードキー送信が終了すると、新規（子）カテゴリツリー（Ｎ－Ｅｎ）は、新規（子）カテゴリツリー（Ｎ－Ｅｎ）のツリー構成を構築し、構築したツリーの頂点に受信した頂点ノードのサブルートキーを設定し、各ノード、リーフのキーを設定して、カテゴリツリー内の有効化キーブロック（サブＥＫＢ）を生成する。一方、上位カテゴリツリー（Ｐ－Ｅｎ）も、新規（子）カテゴリツリー（Ｎ－Ｅｎ）の追加により、有効化する末端ノードを追加した上位カテゴリツリー（Ｐ－Ｅｎ）内のサブＥＫＢを生成する。

【０２３４】

新規（子）カテゴリツリー（Ｎ－Ｅｎ）は、新規（子）カテゴリツリー（Ｎ－Ｅｎ）内のノードキー、リーフキーによって構成されるサブＥＫＢを生成すると、これを上位カテゴリツリー（Ｐ－Ｅｎ）に送信し、さらに、自己のカテゴリツリーで管理するデバイスについてのケイパビリティ情報を上位カテゴリツリーに通知する。

【０２３５】

新規（子）カテゴリツリー（Ｎ－Ｅｎ）からサブＥＫＢおよびケイパビリティ情報を受信した上位カテゴリツリー（Ｐ－Ｅｎ）は、受信したサブＥＫＢとケイパビリティ情報と、上位カテゴリツリー（Ｐ－Ｅｎ）の更新したサブＥＫＢとをキー発行センター（ＫＤＣ：Key Distribute Center）に送信する。

【０２３６】

キー発行センター（ＫＤＣ）は、受領したカテゴリツリーのサブＥＫＢおよびケイパビリティ情報とを図４３で説明したケイパビリティ管理テーブルに登録し、ケイパビリティ管理テーブルを更新する。キー発行センター（ＫＤＣ）は、更新したケイパビリティ管理テーブルに基づいて、様々な態様のＥＫＢ、すなわち特定のケイパビリティを持つカテゴリツリーあるいはデバイスのみが復号可能なＥＫＢを生成することが可能となる。

【０２３７】

［ＥＫＢタイプ定義リストを使用したＥＫＢ管理構成］

次に、１以上の選択されたカテゴリツリーにおいて復号可能なＥＫＢを生成して各カテゴリツリーに属するデバイスに共通に使用可能なＥＫＢを提供する構成において、どのカテゴリツリーで処理可能、すなわち復号可能であるかを示すＥＫＢタイプ定義リストを使用した構成について説明する。

【０２３８】

本構成においては、キー発行センター（ＫＤＣ）は、コンテンツプロバイダなどのＥＫＢの使用、発行処理を望むＥＫＢリクエスタからＥＫＢ発行要求を受領する。ＥＫＢ発行要求にはＥＫＢタイプ定義リストに定義されたＥＫＢタイプを示すＥＫＢタイプ識別ナンバーが含まれ、キー発行センター（ＫＤＣ）は、ＥＫＢタイプ識別ナンバーに従って１または複数のカテゴリツリーにおいて処理（復号）可能なＥＫＢを生成する。

【０２３９】

ＥＫＢの生成に際しては、キー発行センター（ＫＤＣ）は、ＥＫＢタイプ定義リストのＥＫＢタイプ識別ナンバーに対応して設定された各カテゴリツリーのトップノード識別子に基づき、カテゴリツリー管理者としてのトップレベル・カテゴリ・エンティテイ（ＴＬＣＥ：Top Level Category Entity）にサブＥＫＢの生成を要求し、各ＴＬＣＥの生成したサブＥＫＢを受領し、複数のサブＥＫＢの合成処理を実行して複数のカテゴリツリーにおいて処理可能なＥＫＢを生成する。

【０２４０】

本構成においては、コンテンツプロバイダ（ＣＰ）などのＥＫＢの発行要求者は、ＥＫＢタイプ定義リストに基づいて特定のカテゴリツリーの選択を実行することが可能となる。コンテンツプロバイダ（ＣＰ）などのＥＫＢの発行要求者は、ＥＫＢタイプ定義リストを参照して特定カテゴリツリーにおいて処理可能なＥＫＢの発行をキー発行センター（ＫＤＣ）に依頼する。キー発行センター（ＫＤＣ）は、ＥＫＢ発行要求に基づいて、選択されたカテゴリツリーの管理エンティテイに対してサブＥＫＢ発行要求を行ない、各選択されたカテゴリツリーの管理エンティテイは、管理エンティテイのリボークされていない正当なデバイスにおいてのみ処理可能なサブＥＫＢを生成してキー発行センター（ＫＤＣ）に送信する。キー発行センター（ＫＤＣ）は、1以上のサブＥＫＢを組み合わせてＥＫＢの発行要求者の要求した選択カテゴリツリーにのみ処理可能なＥＫＢを生成してＥＫＢ発行要求者に提供する。ＥＫＢ発行要求者は、キー発行センター（ＫＤＣ）からＥＫＢを受領し、ＥＫＢの処理によって取得可能なキーでのみ復号可能な暗号化キー、または暗号化コンテンツの配信を実行する。

【０２４１】

まず、以下の説明における構成エンティテイについて簡単に説明する。

キー発行センター（ＫＤＣ：Key Distribution Center）

有効化キーブロック（ＥＫＢ）を発行し、発行したＥＫＢに関するＥＫＢタイプ定義リストを管理する。

【０２４２】

トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ：Top Level Category Entity）

あるカテゴリツリーを管理するエンティテイ。たとえば記録デバイスのフォーマットホルダー。カテゴリツリーを管理し、管理下のカテゴリツリー内のデバイスにおいて処理（復号）可能なＥＫＢであるサブＥＫＢを生成し、キー発行センター（ＫＤＣ）に提出する。

【０２４３】

ＥＫＢ・リクエスタ（ＥＫＢ requester）

たとえば、電子コンテンツ提供（ＥＣＤ：Electronic Content Distribution）サービスを実行するコンテンツプロバイダ（ＣＰ）など、画像、音声、プログラムなど様々のコンテンツをユーザデバイスに対して提供するエンティテイ、あるいは記録メディアのフォーマットホルダーであり、提供コンテンツの暗号化キーなどにＥＫＢ処理によって取得可能なキーを用いる設定としてコンテンツ、メディアを提供する。この際に用いるＥＫＢの発行要求をキー発行センター（ＫＤＣ）に対して要求する。

【０２４４】

例えば、コンテンツプロバイダ（ＣＰ）は、キー発行センター（ＫＤＣ）の生成したＥＫＢのルートキー（Root Key）を用いて自分のコンテンツを暗号化して配信する。記録メディアのフォーマットホルダーは、ＥＫＢを記録メディアの製造時に書きこんで配布し、記録されるコンテンツがそのＥＫＢのルートキー（Root Key）を用いて暗号化されるようにする。

【０２４５】
（ＴＬＣＥとカテゴリベースのツリー管理）
カテゴリベースのツリー管理については、前述したが、トップレベル・カテゴリ・エンティティ（ＴＬＣＥ）とカテゴリツリーの関係について、図４６を用いて説明する。
【０２４６】
まず、カテゴリは、前述したように同じ性質を持ったデバイスの集合であり、具体的には、同じメーカー製のデバイス、あるいは同じエンコードフォーマットを扱えるデバイス等である。図４６において、Ａ、Ｂ、Ｃ、Ｄはそれぞれカテゴリツリーを示す。
【０２４７】
図４６において、最上段のルートツリーは、例えば８段構成（ノード段数）であり、ルートツリーの最下段にカテゴリツリーのトップノードが設定される。カテゴリツリーは、複数が上位、下位の関係になることが可能であり、図４６において、カテゴリツリーＣは、カテゴリツリーＤの上位に対応する。
【０２４８】
最上段のルートツリーに直接連なるカテゴリツリーをトップレベルカテゴリツリーと呼び、トップレベルカテゴリツリーを管理するエンティティをトップレベル・カテゴリ・エンティティ（ＴＬＣＥ）と呼ぶ。図４６では、Ａ、Ｂ、Ｃがトップレベルカテゴリであり、これらを管理するエンティティがトップレベル・カテゴリ・エンティティ（ＴＬＣＥ）である。トップレベル・カテゴリ・エンティティ（ＴＬＣＥ）は、基本的に、自己のツリー以下全体を管理する責任がある。つまり、図４６のツリーＣを管理するＴＬＣＥは、ツリーＣと同様ツリーＤについての管理も実行する。Ｄ以下にさらに下層のカテゴリツリーが存在すればその下層カテゴリツリー管理も行なう。ただし、例えば下層のカテゴリツリーＤを管理するカテゴリエンティティ(Sub Category Entity)を置いて、その責任と権利を委譲することも可能である。
【０２４９】
コンテンツの利用を行なう記録再生装置などの各デバイスは、トップレベル・カテゴリ・エンティティ（ＴＬＣＥ）により、あるツリーのリーフにアサインされ、そのリーフからルートに至るパスの間のいくつかのノードの鍵を所有する。１つのデバイスが持つノードキーの組をデバイス・ノード・キー（ＤＮＫ：Device Node Key）と呼ぶ。各デバイスが、何個の鍵を持つか（ＤＮＫに何個の鍵が含まれるか）はトップレベル・カテゴリ・エンティティ（ＴＬＣＥ）が決定する。
【０２５０】
図４７にキー発行センター（ＫＤＣ）、トップレベル・カテゴリ・エンティティ（ＴＬＣＥ）、ＥＫＢ・リクエスタ各エンティティの対応、処理の概要を説明する図を示す。
【０２５１】
キー発行センター（ＫＤＣ）４５１１は、ツリー構成を用いたＥＫＢ配信システムの管理エンティティ４５１０として位置づけられる。管理エンティティ４５１０には、さらに、ＥＫＢに対する署名処理を実行する認証局（ＣＡ）４５１２がある。
【０２５２】
キー発行センター（ＫＤＣ）４５１１は、トップレベルカテゴリツリーなどのサブツリーのキー管理を行ない、後述するＥＫＢタイプ定義リストの管理、ＥＫＢの生成を実行する。認証局（ＣＡ）４５１２は、キー発行センター（ＫＤＣ）の生成したＥＫＢに署名を実行するとともに、署名を施した秘密鍵に対応する公開鍵を署名検証用の鍵として発行する。
【０２５３】
キー発行センター（ＫＤＣ）４５１１に対してＥＫＢの発行要求を行なうのがＥＫＢリクエスタ４５２０である。ＥＫＢリクエスタは、例えばコンテンツを格納したＣＤ、ＤＶＤなどのメディアを提供するコンテンツ格納メディアに関するコンテンツプロバイダ（ＣＰ）、電子コンテンツの配信を実行するコンテンツプロバイダ（ＣＰ）、フラッシュメモリなどのストレージシステムのフォーマットについてのライセンスを提供するストレージシステムライセンサなどである。

【０２５４】
これらのＥＫＢリクエスタ４５２０は、それぞれの提供するメデイア、コンテンツ、ライセンスの使用に際し、必要となるキーをＥＫＢ処理によって得られるキーとして設定したＥＫＢをコンテンツ、メディア、ライセンスフォーマットなどに対応付けて提供する。ＥＫＢは、ＥＫＢリクエスタ４５２０からキー発行センター（ＫＤＣ）４５１１に対するＥＫＢ発行要求に従ってキー発行センター（ＫＤＣ）４５１１が生成する。
【０２５５】
ＥＫＢリクエスタ４５２０は、キー発行センター（ＫＤＣ）４５１１に対する発行要求の結果として受領したＥＫＢをメディア製造者４５４０、デバイス製造者４５５０に対して提供し、ＥＫＢを格納したメディア、またはデバイスをユーザに対して供給する処理が可能である。これらのＥＫＢは、例えば１つまたは複数のカテゴリツリーにおいて処理可能なＥＫＢとして生成される。
【０２５６】
本システムでは、複数、例えば２つあるいは３以上のカテゴリツリーにおいて共通に処理可能なＥＫＢや、唯一のカテゴリツリーにおいてのみ処理可能なＥＫＢなど、様々なタイプのＥＫＢが生成され、使用される状況になる。このような様々なタイプのＥＫＢについてリスト化したのがＥＫＢタイプ定義リストである。ＥＫＢタイプ定義リストはキー発行センター（ＫＤＣ）が管理する。ＥＫＢタイプ定義リストについては、後段で詳細に説明する。ＥＫＢリクエスタ４５２０は、ＥＫＢタイプ定義リストの要求をキー発行センター（ＫＤＣ）４５１１に要求してリストを取得可能であり、また、リストのデータ変更があった場合は、キー発行センター（ＫＤＣ）４５１１からＥＫＢリクエスタ４５２０に対して通知される。
【０２５７】
トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）４５３０は、前述したようにルートツリーに連なるカテゴリツリーの管理エンティテイであり、サブツリーのキー管理、管理デバイスＩＤと各デバイスに格納されるＥＫＢ処理のためのノードキー・セットであるデバイス・ノード・キー（ＤＮＫ）との対応リストを管理する。さらに管理下のデバイスに対応するデバイスを製造するデバイス製造者４５５０に対して、デバイス格納用のデバイス・ノード・キー（ＤＮＫ）の生成、提供処理を実行する。
【０２５８】
キー発行センター（ＫＤＣ）４５１１がＥＫＢリクエスタ４５２０からＥＫＢ発行要求を受信すると、キー発行センター（ＫＤＣ）４５１１は、発行要求に従ったＥＫＢを生成する。生成するＥＫＢが例えば２つのトップレベル・カテゴリツリーにおいて処理可能なＥＫＢであった場合は、その２つのトップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）４５３０に対してサブＥＫＢの発行要求を送信し、サブＥＫＢの発行要求を受信したトップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）４５３０はそれぞれのカテゴリツリー内の正当デバイスがルートキー取得可能なサブＥＫＢを生成してキー発行センター（ＫＤＣ）４５１１に送信する。キー発行センター（ＫＤＣ）４５１１は、ＴＬＣＥから受信した1つまたは複数のサブＥＫＢに基づいてＥＫＢを生成する。サブＥＫＢに基づくＥＫＢ生成処理については後段でさらに説明する。
【０２５９】
トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）４５３０は、ＥＫＢリクエスタ４５２０と同様、ＥＫＢタイプ定義リストの要求をキー発行センター（ＫＤＣ）４５１１に要求してリストを取得可能である。
【０２６０】
トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）４５３０は、さらに、ＥＫＢタイプ定義リストの自己のツリーに関して定義されたタイプの削除要求をキー発行センター（ＫＤＣ）４５１１に要求することができる。例えば他のカテゴリツリーと共有のＥＫＢとして定義されたＥＫＢタイプをリストから削除する要求である。トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）４５３０は、さらに、自己の管理するツリーに関する変更があ

った場合には、変更情報をキー発行センター（ＫＤＣ）４５１１に通知する。これらの処理についてはフローを用いて後段で説明する。

【０２６１】

デバイス製造者４５５０は、２つの種類のデバイス製造者に区分される。１つは、製造するデバイスにデバイスノードキー（ＤＮＫ）と、ＥＫＢの両データを格納したデバイスを製造するＤＮＫＥデバイス製造者４５５１であり、他方は、デバイスにデバイスノードキー（ＤＮＫ）のみを格納したデバイスを製造するＤＮＫデバイス製造者４５５２である。

【０２６２】

図４８に図４７に示すキー発行センター（ＫＤＣ）、ＥＫＢリクエスタ、トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）それぞれの構成例をブロック図として示す。キー発行センター（ＫＤＣ）はＥＫＢ発行情報処理装置、ＥＫＢリクエスタはＥＫＢ要求情報処理装置、トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）はカテゴリツリー管理情報処理装置として、基本的に暗号通信可能なデータ処理装置として構成される。

【０２６３】

各エンティテイを構成する情報処理装置は、それぞれ他エンティテイとの相互認証、データ通信時における暗号処理全般を司る暗号処理部を有する。暗号処理部内の制御部は、認証処理、暗号化／復号化処理等の暗号処理全般に関する制御を実行する制御部である。内部メモリは、相互認証処理、暗号化、復号化処理等、各種処理において必要となる鍵データ、識別データ等を格納する。識別データは、例えば他エンティテイとの相互認証処理等において用いられる。

【０２６４】

暗号／復号化部は、内部メモリに格納された鍵データ等を使用したデータ転送時の認証処理、暗号化処理、復号化処理、データの検証、乱数の発生などの処理を実行する。

【０２６５】

ただし、ＥＫＢリクエスタとしての情報処理装置においては、鍵の生成処理を自装置内では実行しない構成も可能である。この場合には、鍵の生成に必要な構成要素、例えば乱数発生装置などを省略可能となる。具体的には、ＥＫＢに含ませるルートキーを自ら生成して生成したルートキーを含むＥＫＢの生成をキー発行センターに要求するＥＫＢリクエスタとしての情報処理装置はルートキーを生成するための手段が必要となるが、ＥＫＢに含ませるルートキーを自ら生成せず、キー発行センターにルートキーの生成処理を要求し、キー発行センター（ＫＤＣ）において生成したルートキーを含むＥＫＢ生成をキー発行センターに要求するＥＫＢリクエスタとしての情報処理装置は乱数発生装置などの鍵生成処理に伴う構成要素が省略可能である。

【０２６６】

暗号処理部の内部メモリは、暗号鍵などの重要な情報を保持しているため、外部から不正に読み出しにくい構造にしておく必要がある。従って、暗号処理部は、外部からアクセスしにくい構造を持った例えば半導体チップで構成された耐タンパメモリとして構成される。

【０２６７】

各エンティテイは、これらの暗号処理機能の他に、中央演算処理装置（ＣＰＵ：Central Processing Unit）、ＲＡＭ（Random Access Memory）、ＲＯＭ(Read Only Memory)、入力部、表示部、データベースＩ／Ｆ、データベースを備えている。

【０２６８】

中央演算処理装置（ＣＰＵ：Central Processing Unit）、ＲＡＭ（Random Access Memory）、ＲＯＭ(Read Only Memory)は、各エンティテイ本体の制御系として機能する構成部である。ＲＡＭは、ＣＰＵにおける各種処理用の主記憶メモリとして使用され、ＣＰＵによる処理のための作業領域として使用される。ＲＯＭは、ＣＰＵでの起動プログラム等が格納される。

【０２６９】

各エンティテイを構成する情報処理装置のデータベースまたはその他の記憶手段には、そ

れぞれ各エンティテイの管理するデータ、例えばキー発行センター（ＫＤＣ）であれば、発行したＥＫＢに関する管理データ、さらにＥＫＢタイプ定義リストなどが格納され、また、トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）のデータベースには、管理デバイスとデバイスノードキー（ＤＮＫ）の対応など、カテゴリツリーに属するデバイスの管理データが格納され、ＥＫＢリクエスタのデータベースには、提供コンテンツとコンテンツに対して使用されているＥＫＢとの関係を対応付けた管理データ、コンテンツの提供先デバイスについての管理データなどが格納される。なお、ＥＫＢタイプ定義リストは、ＥＫＢリクエスタ、トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）を構成する情報処理装置中にも格納し参照可能な状態とする構成が好ましい。あるいは、ＥＫＢリクエスタ、トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）のアクセス可能なキー発行センター（ＫＤＣ）の管理するウェブ（Web）サイトに置く構成としてもよい。

【０２７０】

前述したように、デバイスは、ＥＫＢ処理（復号）のためにデバイス・ノード・キー（ＤＮＫ：Device Node Key）を使用する。１つのデバイスが持つデバイス・ノード・キー（ＤＮＫ）について図４９を用いて説明する。図４９に示すツリーは１つのカテゴリツリーを示し、最下段がデバイスが対応付けられたリーフであり、例えばトップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）の管理ツリーに相当する。さらに上段にはルートツリー（ｅｘ．8段構成）が連なっている。ここでデバイスは、図４９に示すようにデバイスから上段に至るパス上のノードキーを有する。これらのキーセットをデバイス・ノード・キー（ＤＮＫ）として保有し、デバイス・ノード・キー（ＤＮＫ）を用いてＥＫＢの復号を行なう。

【０２７１】

基本的に、１つのデバイスが１つのリーフに重ならないようにアサインされる。例外として、たとえばＰＣソフトなどのソフトウェアをリーフに対応付ける場合は、１つのバージョンのソフトウェア・パッケージがすべて１つのリーフにアサインされる場合もある。これもＴＬＣＥが決める。つまり、デバイスをどのようにリーフにアサインし、どのノードキーを持たせるかはＴＬＣＥが決める。

【０２７２】

トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）は、デバイス自体の提供業者（メーカー）である場合もあり、製造デバイスに対して予めデバイス・ノード・キー（ＤＮＫ）を格納してユーザに提供（販売）することが可能である。すなわち、記録再生装置などのデバイスに対して予め、ある特定のカテゴリツリーのノードキーのセットをデバイス・ノード・キー（ＤＮＫ）としてメモリに格納してユーザに提供（販売）することが可能である。

【０２７３】

（ＥＫＢタイプ定義リスト）

カテゴリ単位でのＥＫＢ配信については、すでに説明した通りであるが、複数のカテゴリに共通のＥＫＢ、すなわち異なるカテゴリツリーに属するデバイスにおいて処理可能なＥＫＢを生成して発行した場合には、いくつかの問題点が発生する場合がある。

【０２７４】

例えば、ある再書き込み可能なメディア（記録媒体）例えば、携帯型フラッシュメモリのフォーマットのライセンシー（ライセンス受領者）として、Ａ社とＢ社の異なる２社が存在し、メディア（携帯型フラッシュメモリ）のライセンサー（ライセンス許諾者）であるメーカーがトップレベルカテゴリとして存在し、その下にＡ社の管理するカテゴリツリーとＢ社の管理するカテゴリツリーがある構成において、Ａ社とＢ社は相互のデバイスに互換性を持たせ、様々な配布コンテンツを共通に利用することを可能とするため、Ａ社のカテゴリツリーとＢ社のカテゴリツリーの２つのカテゴリツリーの所属デバイスにおいて処理（復号）可能なＥＫＢをキー発行センター（ＫＤＣ）において生成し発行する。

【０２７５】

このような状況でＡ社の管理するカテゴリツリーに属する１つのデバイスのデバイス・ノ

ード・キー（ＤＮＫ）が漏洩してしまうと、そのデバイスノードキー（ＤＮＫ）を利用してＡ社、Ｂ社の相互のデバイスにおいて利用可能とした配布コンテンツがすべて不正に利用されてしまう可能性が発生することになる。利用を排除するためには、リボーク処理としてのＥＫＢ更新処理が必要となるが、この場合、Ａ社のカテゴリツリーに関するリボーク処理ではなく、Ａ社およびＢ社の２つのカテゴリツリーに共通のＥＫＢが存在するため、ＥＫＢ更新処理はＡ社およびＢ社の２つのカテゴリツリーに関して実行することが必要となる。

【０２７６】

このように、複数のカテゴリツリーに共通のＥＫＢを生成して提供した場合、１つのカテゴリツリー内でのリボーク処理、ＥＫＢ更新処理のみではなく、共通するＥＫＢを使用するすべての他のカテゴリツリーにおいてリボークに伴うＥＫＢ更新処理を実行することが必要となる。これはＢ社にとっては、自己の管理するデバイスと異なる他の管理カテゴリツリーの影響を受けることになり、処理負荷が高まることになってしまう。

【０２７７】

このような状況を解決するため、複数のカテゴリにおいて共通に使用可能なＥＫＢの発行の許可権限を、それぞれのカテゴリを管理するカテゴリ・エンティティが有する構成とする。つまり、互換性をとるために、相手のカテゴリに属するデバイスの不具合によって引き起こされる自分のカテゴリ内デバイスへのリスクを許容できる場合にのみ、互換性をとるＥＫＢの発行を認め、リスクが許容できない場合は、共通に使用可能なＥＫＢの発行、または使用を認めないものとする。

【０２７８】

このような処理を行なおうとすると、複数、例えば２つあるいは３以上のカテゴリツリーにおいて共通に処理可能なＥＫＢや、唯一のカテゴリツリーにおいてのみ処理可能なＥＫＢなど、様々なタイプのＥＫＢが生成され、使用される状況になる。このような様々なタイプのＥＫＢについてリスト化したのがＥＫＢタイプ定義リストである。図５０にＥＫＢタイプ定義リストの例を示す。ＥＫＢタイプ定義リストはキー発行センター（ＫＤＣ）が記録媒体に記録して管理する。また、ＥＫＢリクエスタ、ＴＬＣＥに対しても必要に応じて提供または閲覧可能な状態におかれる。

【０２７９】

図５０に示すように、ＥＫＢタイプ定義リストは、「ＥＫＢタイプ識別ナンバー」、「ノード」、「説明」の各フィールドを有し、「ＥＫＢタイプ識別ナンバー」は、ＥＫＢタイプ定義リストにリストアップされた様々な態様のＥＫＢを識別するナンバーであり、識別ナンバーが異なれば、そのＥＫＢを処理可能なカテゴリツリーまたはその組み合わせが異なる構成となっている。

【０２８０】

「ノード」フィールドは、ＥＫＢを適用可能なカテゴリツリーのトップノードＩＤを記録するフィールドである。例えばＥＫＢタイプ識別ナンバー：１のＥＫＢは、ＭＳ（MemoryStick：メモリスティック）のカテゴリツリーのトップノードＩＤが記録される。また、ＥＫＢタイプ識別ナンバー：３のＥＫＢは、ＭＳ（MemoryStick：メモリスティック）のカテゴリツリーのトップノードＩＤとＰＨＳのカテゴリツリーのトップノードＩＤが記録される。

【０２８１】

「説明」フィールドは、ＥＫＢタイプ定義リストにリストアップされた様々な態様のＥＫＢの説明を記録するフィールドであり、例えばＥＫＢタイプ識別ナンバー：１のＥＫＢは、ＭＳ（MemoryStick：メモリスティック）用のＥＫＢであることを示している。また、ＥＫＢタイプ識別ナンバー：３のＥＫＢは、ＭＳ（MemoryStick：メモリスティック）とＰＨＳのカテゴリツリーのデバイスに共通に使用可能なＥＫＢであることを示している。

【０２８２】

図５０に示すＥＫＢタイプ定義リストはキー発行センター（ＫＤＣ）が管理する。また、ＥＫＢの処理により取得可能なキーによって暗号化した暗号化キーまたは暗号化コンテン

ツ等の暗号化データ配信を行なおうとするエンティテイ、例えばコンテンツプロバイダは、図５０に示すＥＫＢタイプ定義リストを参照して、コンテンツの提供対象となるデバイスを含むカテゴリツリーによって処理可能なＥＫＢタイプを選択し、そのＥＫＢタイプ識別ナンバーを指定して、キー発行センター（ＫＤＣ）にＥＫＢ生成処理を依頼する。

【０２８３】

ただし、ＥＫＢタイプ定義リストに対する様々なタイプのＥＫＢ登録処理においては、登録対象となるカテゴリツリーのトップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）の承認が必要となる。例えばカテゴリツリーＡのＴＬＣＥ－Ａが他のカテゴリと共有するＥＫＢの発行を拒否すれば、カテゴリツリーＡと他のカテゴリツリーの共有となるＥＫＢのタイプはＥＫＢタイプ定義リストに登録されない。

【０２８４】

例えばカテゴリツリーＡのＴＬＣＥ－Ａ、カテゴリツリーＢのＴＬＣＥ－Ｂ、カテゴリツリーＣのＴＬＣＥ－Ｃの各々が共有のＥＫＢの発行を承認すれば、これら３つのカテゴリツリーにおいて処理可能な共通のＥＫＢのタイプがＥＫＢタイプ定義リストに登録され、例えばコンテンツプロバイダがその登録タイプを示すＥＫＢタイプ識別ナンバーを指定してキー発行センター（ＫＤＣ）にＥＫＢ生成処理を依頼することが可能となる。

【０２８５】

つまり、ＥＫＢタイプ定義リストに新たなＥＫＢタイプを登録し、そのＥＫＢタイプに対応するＥＫＢタイプ識別ナンバーを定義するためには、下記の処理が必要となる。

（１）定義しようとするＥＫＢタイプ識別ナンバーに対応するＥＫＢの適用対象となるカテゴリを管理するすべてのＴＬＣＥがＥＫＢタイプ登録リクエストをキー発行センター（ＫＤＣ）に送る。

（２）キー発行センター（ＫＤＣ）は要求にある登録対象となるＥＫＢを処理可能な１以上のカテゴリツリーのトップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）のすべてが上記のＥＫＢタイプ登録リクエストを送ってきたことを確認した後に、新たなＥＫＢタイプ識別ナンバーを定義し、ＥＫＢタイプ定義リストに加える。

（３）キー発行センター（ＫＤＣ）はＥＫＢタイプ定義リストに変更があったことを知らせるため、ＥＫＢタイプ定義リスト変更通知を全ＴＬＣＥおよびＥＫＢリクエスタに送る。

【０２８６】

なお、ＥＫＢタイプ定義リストは、全ＴＬＣＥおよびＥＫＢリクエスタに送られ、またウェブ（Web）サイトに置かれるなどして全ＴＬＣＥおよびＥＫＢリクエスタに公開される。従って、ＴＬＣＥおよびＥＫＢリクエスタは、常に最新のＥＫＢタイプ定義リストに登録されたＥＫＢタイプ情報を取得することが可能となる。

【０２８７】

（ＥＫＢタイプ登録処理）

ＥＫＢタイプ定義リストに新たなＥＫＢタイプを登録する際に、キー発行センター（ＫＤＣ）の実行する処理を説明する処理フローを図５１に示す。まず、キー発行センター（ＫＤＣ）は、新たなＥＫＢタイプの登録要求を行なうＴＬＣＥからのＥＫＢタイプ登録リクエストを受信（Ｓ１０１）する。ＴＬＣＥからのＥＫＢタイプ登録リクエストには、登録要求ＥＫＢが共通に使用可能とするカテゴリ数が含まれる。キー発行センター（ＫＤＣ）は、要求内のカテゴリ数に一致する数のカテゴリに対応するＴＬＣＥから同様のＥＫＢタイプ登録リクエストを受領したか否かを判定（Ｓ１０２）し、要求内のカテゴリ数に一致する数のカテゴリに対応するＴＬＣＥからの要求を受理したことを条件として、ＥＫＢタイプ定義リストに対して要求に従った新たなＥＫＢタイプを登録し、リストの更新処理、リストの更新通知処理（Ｓ１０３）を行なう。更新通知処理は、ＴＬＣＥおよびＥＫＢリクエスタに対して行われる。

【０２８８】

このように、キー発行センター（ＫＤＣ）は、ＥＫＢタイプ定義リストに対するＥＫＢタイプ識別子の新規登録処理において、登録予定のＥＫＢタイプの処理可能なカテゴリツリ

ーとして選択された１以上のカテゴリツリーを管理するすべてのカテゴリ・エンティティの承認を条件として登録を行なう。

【０２８９】

なお、これらの処理において、キー発行センター（ＫＤＣ）とＴＬＣＥ、ＥＫＢリクエスタ間の通信においては必要に応じて相互認証処理、送信データの暗号化処理が行われる。また、その他のメッセージ暗号化処理、デジタル署名生成、検証処理を行なう構成としてもよい。なお、公開鍵暗号方式に基づく認証あるいは暗号通信を実行する場合は、各エンティティ間において予め公開鍵を保有し合う手続きを行なっておく。

【０２９０】

（ＥＫＢタイプ無効化処理）

たとえば、あるカテゴリに属するすべての機器をリボークしなければならないときには、トップレベル・カテゴリ・エンティティ（ＴＬＣＥ）はそのカテゴリが要素となっているＥＫＢタイプを無効化する要求をキー配信センター（ＫＤＣ）に出す必要がある。また、トップレベル・カテゴリ・エンティティ（ＴＬＣＥ）は、たとえばあるサービスを停止するなどの理由で、現在登録されているＥＫＢタイプを無効化する要求をＫＤＣに出すことができる。

【０２９１】

このＥＫＢタイプ無効化処理の流れを図５２の処理フローに従って説明する。キー発行センター（ＫＤＣ）は、ＥＫＢタイプの無効化要求を行なうＴＬＣＥからのＥＫＢタイプ無効化リクエストを受信（Ｓ２０１）する。ＴＬＣＥからのＥＫＢタイプ無効化リクエストを受信すると、キー発行センター（ＫＤＣ）は、そのリクエストにより無効化されるＥＫＢタイプの要素となっているカテゴリを管理するＴＬＣＥがそのリクエストの発信者であることを確認した上で、ＥＫＢタイプ定義リスト内の無効化リクエストにおいて指定されたタイプに対応するＥＫＢタイプ識別ナンバーを無効化してＥＫＢタイプ定義リストを更新し、リストの更新通知処理（Ｓ２０２）を行なう。更新通知処理は、ＴＬＣＥおよびＥＫＢリクエスタに対して行われる。

【０２９２】

このように、キー発行センター（ＫＤＣ）は、ＥＫＢタイプ定義リストに登録されたＥＫＢタイプ識別子の無効化処理において、無効化予定のＥＫＢタイプの処理可能なカテゴリツリーとして選択された１以上のカテゴリツリーを管理する少なくとも１つのカテゴリ・エンティティの無効化要求を条件として無効化処理を行なう。この場合、他のカテゴリ・エンティティの承認は必要としない。

【０２９３】

なお、これらの処理において、キー発行センター（ＫＤＣ）とＴＬＣＥ、ＥＫＢリクエスタ間の通信においては必要に応じて相互認証処理、送信データの暗号化処理が行われる。また、その他のメッセージ暗号化処理、デジタル署名生成、検証処理を行なう構成としてもよい。なお、公開鍵暗号方式に基づく認証あるいは暗号通信を実行する場合は、各エンティティ間において予め公開鍵を保有し合う手続きを行なっておく。

【０２９４】

（ＥＫＢタイプ定義リスト変更通知処理）

たとえばあるカテゴリツリー内において、デバイスリボケーション（デバイス排除）や、あるデバイスが格納したＤＮＫを新しいものに交換するデバイスノードキー（ＤＮＫ）の更新などのツリー内の状態を変化させる処理をそのカテゴリツリーを管理するＴＬＣＥが行った場合、それらのデバイスを対象とするＥＫＢを使用しているＥＫＢリクエスタまたは関連ＴＬＣＥに対して、これらの処理が起こったことを知らせる必要がある。

【０２９５】

なぜならば、デバイスリボケーションが起こったのにそれを知らせず、コンテンツプロバイダ（ＣＰ）が古いＥＫＢを使いつづけてコンテンツを暗号化して配信したとすると、古いＥＫＢでは、リボークされたデバイスにおいてもＥＫＢ処理（復号）が可能であり、コンテンツの不正利用が続けられる可能性があるからである。また、デバイスノードキー（

ＤＮＫ）の更新を行なった場合、通常は置きかえられた古いＤＮＫは捨てられ、デバイスは新しいＤＮＫを持つことになるが、この新しいＤＮＫに対応したＥＫＢをコンテンツプロバイダが使用しなければ、新しいＤＮＫを持つデバイスはＥＫＢを処理（復号）することができなくなり、コンテンツにアクセスできなくなってしまうからである。

【０２９６】

このような弊害を避けるため、

*デバイスリボケーションなどの結果として、ＥＫＢのタグパートに変更が生じた場合、

*デバイスノードキー（ＤＮＫ）の更新などの結果として少なくともひとつの機器が持つＤＮＫの値に変更が生じた場合、

これらの場合には、ＴＬＣＥは、ツリー変更通知（Tree Change Notification）をキー発行センター（ＫＤＣ）に送る必要がある。ツリー変更通知（Tree Change Notification）には、変更を要するＥＫＢタイプ定義リストに登録済みのＥＫＢタイプ識別ナンバー、ＥＫＢタイプ識別ナンバーに対応して登録されているどのカテゴリで起こったかを示す情報と、リボケーション、ＤＮＫ更新の何が起こったかという情報が含まれる。

【０２９７】

ＥＫＢタイプ定義リスト変更通知処理の流れを図５３の処理フローに従って説明する。キー発行センター（ＫＤＣ）は、ＴＬＣＥからツリー変更通知を受信（Ｓ３０１）する。ＴＬＣＥからのツリー変更通知を受信すると、キー発行センター（ＫＤＣ）は、ＥＫＢタイプ定義リストから、そのカテゴリを要素に持つＥＫＢタイプ識別ナンバーを抽出し、どのＥＫＢタイプ識別ナンバーに、どのような変化（ｅｘ．リボケーションか、ＤＮＫ更新（リプレイスメント）か）が起こったかの情報を持つＥＫＢタイプ定義リスト変更通知をすべてのＴＬＣＥおよびＥＫＢリクエスタに対して行なう。なお、これらの処理において、キー発行センター（ＫＤＣ）とＴＬＣＥ、ＥＫＢリクエスタ間の通信においては必要に応じて相互認証処理、送信データの暗号化処理が行われる。また、その他のメッセージ暗号化処理、デジタル署名生成、検証処理を行なう構成としてもよい。なお、公開鍵暗号方式に基づく認証あるいは暗号通信を実行する場合は、各エンティティ間において予め公開鍵を保有し合う手続きを行なっておく。

【０２９８】

（ＥＫＢタイプ定義リスト要求）

トップレベル・カテゴリ・エンティティ（ＴＬＣＥ）やＴＬＣＥ以外のサブカテゴリ・エンティティ（ＳＣＥ）、あるいはコンテンツプロバイダ等のＥＫＢリクエスタは、最新版のＥＫＢタイプ定義リストを知るために、ＥＫＢタイプ定義リストの送付をキー発行センター（ＫＤＣ）に要求することができる。キー発行センター（ＫＤＣ）はこの要求に対して、最新版のＥＫＢタイプ定義リストを要求者に送り返す。

【０２９９】

ＥＫＢタイプ定義リスト要求処理の流れを図５４の処理フローに従って説明する。キー発行センター（ＫＤＣ）は、ＴＬＣＥ、サブカテゴリ・エンティティ、またはＥＫＢリクエスタのいずれかからＥＫＢタイプ定義リスト要求を受信（Ｓ４０１）する。ＥＫＢタイプ定義リスト要求を受信すると、キー発行センター（ＫＤＣ）は、最新のＥＫＢタイプ定義リストを抽出し、要求処理を行なったエンティティに対して最新のＥＫＢタイプ定義リストを送信（Ｓ４０２）する。なお、これらの処理において、キー発行センター（ＫＤＣ）とＴＬＣＥ、サブカテゴリ・エンティティ、ＥＫＢリクエスタ間の通信においては必要に応じて相互認証処理、送信データの暗号化処理が行われる。また、その他のメッセージ暗号化処理、デジタル署名生成、検証処理を行なう構成としてもよい。なお、公開鍵暗号方式に基づく認証あるいは暗号通信を実行する場合は、各エンティティ間において予め公開鍵を保有し合う手続きを行なっておく。

【０３００】

（ＥＫＢ発行処理）

ＥＫＢの発行処理は、ＥＫＢリクエスタによるＥＫＢ発行要求に基づいて行われる。ＥＫＢリクエスタは、

［ａ］ＣＤ、ＤＶＤなどの、コンテンツ格納メディアを提供するコンテンツプロバイダ（ＣＰ）。
［ｂ］電子情報配信（ＥＣＤ：Electronic Content Distribution）サービスを提供するコンテンツプロバイダ。
［ｃ］記録システムのフォーマットホルダー。
など、ＥＫＢの復号によって取得されるキーを用いてコンテンツの利用、フォーマットの使用を可能とするサービス、メディア、デバイスを提供するエンティティである。
【０３０１】
上記の［ｃ］記録システムのフォーマットホルダーには、
［ｃ１]たとえば製造時に記録媒体にＥＫＢを格納するようなフォーマットにおいて、記録媒体の製造業社に取得したＥＫＢを与えるフォーマットホルダー。
［ｃ２］たとえば製造時に記録デバイスにＥＫＢを格納するようなフォーマットにおいて、記録デバイスの製造業社に取得したＥＫＢを与えるフォーマットホルダー。
の２種類のフォーマットホルダーがある。
【０３０２】
ＥＫＢ発行処理の手順について、以下説明する。
【０３０３】
（１）コンテンツキーの作成
まず、コンテンツプロバイダなどのＥＫＢリクエスタは、自己の提供するコンテンツ、デバイス、メディアに対応して使用されるコンテンツキーを生成する。
例えばＥＫＢリクエスタが、
［ａ］ＣＤ、ＤＶＤなどの、コンテンツ格納メディアを提供するコンテンツプロバイダ（ＣＰ）。
［ｂ］電子情報配信（ＥＣＤ：Electronic Content Distribution）サービスを提供するコンテンツプロバイダ。
である場合、
生成するコンテンツキーは、メディア上や電子情報配信（ＥＣＤ）サービスにおいて、コンテンツを守る（暗号化する）鍵として使用される。
【０３０４】
また、ＥＫＢリクエスタが、
［ｃ１]製造時に記録媒体にＥＫＢを格納するようなフォーマットにおいて、記録媒体の製造業社に取得したＥＫＢを与えるフォーマットホルダー。
である場合、コンテンツキーは、その記録媒体上に記録されるコンテンツを守る（暗号化する）鍵として使用される。
さらに、ＥＫＢリクエスタが、
［ｃ２］製造時に記録デバイスにＥＫＢを格納するようなフォーマットにおいて、記録デバイスの製造業社に取得したＥＫＢを与えるフォーマットホルダー。
である場合、コンテンツキーは、その記録デバイスを用いて記録されるコンテンツを守る（暗号化する）鍵として使用される。
【０３０５】
なお、コンテンツキーを用いてコンテンツを保護するための暗号アルゴリズムなどのメカニズムは、各フォーマットごとに任意に決めることができる。
【０３０６】
（２）ルートキーの生成
ＥＫＢリクエスタはＥＫＢの復号処理によって取得可能としたルートキーを生成する。なお、ＥＫＢリクエスタは自らはルートキーを生成せず、キー発行センター（ＫＤＣ）に生成を依頼してもよい。ルートキーはコンテンツキーを保護する（暗号化する）ために使用される。なおルートキーを用いてコンテンツキーを保護するための暗号アルゴリズムなどのメカニズムは、各フォーマットごとに任意に決めることができる。
【０３０７】

（３）ＥＫＢ発行要求
ＥＫＢリクエスタはＥＫＢの発行要求をキー発行センター（ＫＤＣ）に送る。
このリクエストには上記のルートキーおよび、ＥＫＢによりルートキーをどのカテゴリの機器に送るかという、ＥＫＢタイプ定義リストに登録されているＥＫＢタイプ識別ナンバーのひとつが含まれる。ＥＫＢリクエスタは、自装置の記憶手段に格納したＥＫＢタイプ定義リスト、あるいはネットワーク上の閲覧可能サイトから取得したＥＫＢタイプ定義リストに基づいてコンテンツ提供などのサービスを提供する対象となるデバイスを含むカテゴリからなるＥＫＢタイプを選択して、選択したＥＫＢタイプを示すＥＫＢタイプ識別ナンバーをＥＫＢ発行要求中に含ませてキー発行センター（ＫＤＣ）に送信する。

【０３０８】
（４）ＥＫＢ発行処理
キー発行センター（ＫＤＣ）はＥＫＢリクエスタからのＥＫＢ発行要求に基づき、ＥＫＢ発行要求中にルートキーが含まれてい場合は、そのルートキーを含むＥＫＢの生成を行ない、ＥＫＢ発行要求中にルートキーが含まれず、ルートキー生成処理依頼がなされた場合は、ＫＤＣがルートキーを生成し、生成ルートキーを含むＥＫＢを生成してＥＫＢリクエスタに送信する。

【０３０９】
キー発行センターの生成するＥＫＢは、単一のカテゴリツリーにおいて処理可能なＥＫＢである場合と、複数のカテゴリツリーにおいて共通に処理可能なＥＫＢである場合がある。キー発行センター（ＫＤＣ）はＥＫＢ発行要求に含まれるＥＫＢタイプ識別ナンバーに基づいて、そのＥＫＢタイプ識別ナンバーの構成要素となっているカテゴリ、すなわちＥＫＢタイプ定義リストにおいて、指定されたＥＫＢタイプ識別ナンバーのノードフィールドに記録されたノードを抽出する。ノードフィールドにはカテゴリツリーのトップノードＩＤが記録されている。これは、そのカテゴリツリーの管理エンティテイに対応するノードＩＤである。このノードＩＤに基づいて、カテゴリツリーの管理エンティテイであるトップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）に対し、サブＥＫＢの発行要求を出す。サブＥＫＢの発行要求にはルートキーと、各カテゴリを表す情報が含まれる。

【０３１０】
キー発行センター（ＫＤＣ）からサブＥＫＢ発行要求を受け取ったＴＬＣＥは、指定された１つ以上のカテゴリ内の（リボークされていない）各機器から、最終的にルートキーを得られる構成を持つサブＥＫＢを生成してキー発行センター（ＫＤＣ）に送信する。

【０３１１】
トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）の生成するサブＥＫＢは、バージョン番号やその検証用の情報（Version Check Value）を持たないほかは、通常のＥＫＢ（図６参照）と同様の構造を持つ情報の組である。ここで、サブＥＫＢにおけるリーフキーやノードキーを用いて上位のノードキーやルートキーを暗号化するアルゴリズムや鍵長、モードは、サブＥＫＢを生成する各ＴＬＣＥ（フォーマットホルダー）ごとに任意に決めることができる。これにより、他のフォーマットとは別個に独自のセキュリティ方式を用いることができる。また、デフォルトとしてたとえば 暗号アルゴリズムを FIPS46-2 のトリプルＤＥＳ（Triple-DES）と決めておき、これに異存のないＴＬＣＥはトリプルＤＥＳアルゴリズムを適用する構成としてもよい。ＴＬＣＥが任意に暗号アルゴリズムや鍵長を決める場合でも、別のＴＬＣＥが作ったサブＥＫＢと合成されたＥＫＢを、他のＴＬＣＥの支配下にある機器でも処理できるように、ひとつひとつの（暗号化された）鍵は、所定長、たとえば１６バイト（16Byte）のデータで表すと決める。このように複数のカテゴリツリーで共通のＥＫＢを生成する場合に所定のルールに従って、データを設定することにより、異なるカテゴリツリーの各機器は、ＥＫＢのタグを辿って、自分が何番目の鍵データが必要か判断可能となる。すなわちＥＫＢ内に含まれる鍵データの各々が１６バイトであれば、自デバイスで処理可能な鍵データを順次抽出して処理することが可能となり、最終的にルートキーを取得することが可能となる。

【０３１２】

すなわち、サブＥＫＢに基づいて生成される合成ＥＫＢは、複数のキー・データの各々が固定長のデータフィールド内に格納された構成を有する。従って、各々独自のアルゴリズム、独自のキー・データ長を持つサブ有効化キーブロック（サブＥＫＢ）に基づいて生成される合成ＥＫＢは、サブＥＫＢ内の複数の暗号化キーデータを、キーツリーにおけるノードまたはリーフ位置に応じて再配列して生成されても、ＥＫＢのタグを辿って必要なキー・データを順次取得可能となる。このような合成ＥＫＢは、ネットワークを介してあるいは様々な記録媒体に格納して、ユーザ（デバイス）に対して配信または提供される。

【０３１３】

キー発行センター（ＫＤＣ）は、ＴＬＣＥから送られてきたサブＥＫＢを、必要に応じて組替え、合成し、バージョン番号と、その検証用の情報を付加して合成した合成ＥＫＢを完成させてＥＫＢリクエスタに送信する。ただし公開鍵暗号技術を用いたデジタル署名は、キー発行センター（ＫＤＣ）とは別の認証局（ＣＡ：Certificate Authority）に依頼する場合もある。

【０３１４】

サブＥＫＢの生成、サブＥＫＢから合成ＥＫＢの生成について、図を参照して説明する。図５５は、カテゴリツリーＡ，５１００とカテゴリツリーＢ，５２００に共通の合成ＥＫＢを生成する処理において、カテゴリツリーＡ，５１００のＴＬＣＥの生成するサブＥＫＢ－（Ａ）の構成を説明する図である。サブＥＫＢ－（Ａ）は、カテゴリツリーＡ，５１００の各デバイスがルートキーを取得可能なＥＫＢとして生成される。なお、図においてルートツリー領域５３００は上述の説明では８段構成として説明してきたが、ここでは説明を簡略化するため２段構成としてある。

【０３１５】

図５５において、ツリー構成内に記載されたアンダーラインを付加した３桁の数値［ＸＸＸ］はＥＫＢ内のタグ（ｅ，ｌ，ｒ）を示し、前述（図２６，図２７参照）したように、ｅ＝１はデータあり、ｅ＝０はデータなしを示し、ｌ＝１は左に枝なし、ｌ＝０は左に枝ありを示し、ｒ＝１は右に枝なし、ｒ＝０は右に枝ありを示している。

【０３１６】

図５５のカテゴリツリーＡ，５１００の各デバイス（リーフ）がルートキーを取得するためには、各リーフが共通に格納しているノードキーによってルートキーを暗号化したデータを格納したＥＫＢを生成すればよい。各デバイスは、図５５のカテゴリツリーＡ，５１００のデバイスノードキー（ＤＮＫ）領域５１２０のツリーの各パスのノードキーを保有しているので、ＤＮＫ領域５１２０の最上段のノードキーでルートキーを暗号化したＥＫＢを生成すればよい。

【０３１７】

従って、カテゴリツリーＡ，５１００のＴＬＣＥの生成するサブＥＫＢ－（Ａ）は、タグパート：１０１，０１０，０００，１１１，１１１、キーパート：Ｅｎｃ（Ｋ０１０，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ０１１，Ｋｒｏｏｔ）となるサブＥＫＢ－（Ａ）となる。カテゴリツリーＡ，５１００のＴＬＣＥは、このサブＥＫＢ－（Ａ）をキー発行センター（ＫＤＣ）に送信する。

【０３１８】

次に、カテゴリツリーＢ，５２００の生成するサブＥＫＢ－（Ｂ）について図５６を用いて説明する。カテゴリツリーＢ，５２００の各デバイス（リーフ）がルートキーを取得するためには、各リーフが共通に格納しているノードキーによってルートキーを暗号化したデータを格納したＥＫＢを生成すればよい。各デバイスは、図５６のカテゴリツリーＢ５２００のデバイスノードキー（ＤＮＫ）領域５２２０のツリーの各パスのノードキーを保有しているので、ＤＮＫ領域５２２０の最上段のノードキーでルートキーを暗号化したＥＫＢを生成すればよい。

【０３１９】

従って、カテゴリツリーＢ，５２００のＴＬＣＥの生成するサブＥＫＢ－（Ｂ）は、タグパート：１１０，０１０，０００，１１１，１１１、キーパート：Ｅｎｃ（Ｋ１１０，Ｋ

ｒｏｏｔ），Ｅｎｃ（Ｋ１１１，Ｋｒｏｏｔ）となるサブＥＫＢ－（Ｂ）となる。カテゴリツリーＢ，５２００のＴＬＣＥは、このサブＥＫＢ－（Ｂ）をキー発行センター（ＫＤＣ）に送信する。

【０３２０】

キー発行センターは、各ＴＬＣＥの生成したサブＥＫＢ－（Ａ）とサブＥＫＢ－（Ｂ）とから合成ＥＫＢを生成する。合成ＥＫＢの生成について図５７を用いて説明する。合成ＥＫＢは、カテゴリツリーＡ，５１００、およびカテゴリツリーＢ，５２００の各ツリーに属するデバイスがルートキーを取得可能としたＥＫＢとして構成される。基本的には、受領した複数のサブＥＫＢの鍵データ配列を混合してツリー上段から揃える作業によって合成ＥＫＢが生成される。なお、同一段では左側を先とするデータ配列を行なう。

【０３２１】

この結果、合成ＥＫＢは、タグパート：１００，０１０，０１０，０００，０００，１１１，１１１，１１１，１１１、キーパート：Ｅｎｃ（Ｋ０１０，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ０１１，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ１１０，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ１１１，Ｋｒｏｏｔ）を持つＥＫＢとして生成される。各キーパートの鍵データは前述したように各々例えば１６バイトとして設定することにより、各カテゴリツリー内のデバイスは自デバイスで処理可能な鍵データ位置を検出可能であるので、合成ＥＫＢからルートキーを取得することが可能となる。

【０３２２】

以上は、いずれのカテゴリツリーにもリボークされたデバイスがない場合のサブＥＫＢの生成および合成ＥＫＢの生成処理構成であるが、次に、リボークデバイスがある場合のサブＥＫＢの生成および合成ＥＫＢの生成について説明する。

【０３２３】

図５８は、カテゴリツリーＡ，５１００にリボークデバイス（０１１０１）５１５０が存在する場合のサブＥＫＢの生成について説明する図である。この場合のサブＥＫＢは、リボークデバイス（０１１０１）５１５０のみが処理できないサブＥＫＢ－（Ａ’）として生成される。

【０３２４】

この場合、図の太線で示すパスを接続した鍵データ構成を持つサブＥＫＢを生成することになる。従って、カテゴリツリーＡ，５１００のＴＬＣＥの生成するサブＥＫＢ－（Ａ’）は、タグパート：１０１，０１０，０００，１１１，０００，００１，１１１，１１１、キーパート：Ｅｎｃ（Ｋ０１０，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ０１１１，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ０１１００，Ｋｒｏｏｔ）となるサブＥＫＢ－（Ａ’）となる。カテゴリツリーＡ，５１００のＴＬＣＥは、このサブＥＫＢ－（Ａ’）をキー発行センター（ＫＤＣ）に送信する。

【０３２５】

キー発行センターは、各ＴＬＣＥの生成したサブＥＫＢ－（Ａ’）と、リボークデバイスのないカテゴリツリーＢ，５２００のＴＬＣＥから受領したサブＥＫＢ－（Ｂ）（図５６参照）とから合成ＥＫＢを生成する。合成ＥＫＢの生成について図５９を用いて説明する。合成ＥＫＢは、カテゴリツリーＡ，５１００のリボークデバイス（０１１０１）５１５０を除くデバイス、およびカテゴリツリーＢ，５２００のツリーに属するデバイスがルートキーを取得可能としたＥＫＢとして構成される。基本的には、受領した複数のサブＥＫＢの鍵データ配列を混合してツリー上段から揃える作業によって合成ＥＫＢが生成される。なお、同一段では左側を先とするデータ配列を行なう。

【０３２６】

この結果、合成ＥＫＢは、タグパート：１００，０１０，０１０，０００，０００，１１１，０００，１１１，１１１，００１，１１１，１１１、キーパート：Ｅｎｃ（Ｋ０１０，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ１１０，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ１１１，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ０１１１，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ０１１００，Ｋｒｏｏｔ）を持つＥＫＢとして生成される。この合成ＥＫＢは、カテゴリツリーＡ，５１００のリボークデバイス

（０１１０１）５１５０を除くデバイス、およびカテゴリツリーＢ，５２００のツリーに属するデバイスがルートキーを取得可能なＥＫＢである。

【０３２７】

（５）ＥＫＢの利用

上述のような処理によってキー発行センター（ＫＤＣ）の生成したＥＫＢは、ＥＫＢリクエスタに送信される。

例えばＥＫＢリクエスタが、

［ａ］ＣＤ、ＤＶＤなどの、コンテンツ格納メディアを提供するコンテンツプロバイダ（ＣＰ）。

［ｂ］電子情報配信（ＥＣＤ：Electronic Content Distribution）サービスを提供するコンテンツプロバイダ。

である場合、

ＥＫＢによって取得可能なルートキーでコンテンツキーを暗号化し、コンテンツキーでユーザデバイスに提供するコンテンツを暗号化してコンテンツを流通させることになる。この構成により、ＥＫＢの処理可能な特定のカテゴリツリーに属するデバイスのみがコンテンツの利用が可能となる。

【０３２８】

また、ＥＫＢリクエスタが、

［ｃ１]製造時に記録媒体にＥＫＢを格納するようなフォーマットにおいて、記録媒体の製造業社に取得したＥＫＢを与えるフォーマットホルダー。

である場合、生成したＥＫＢ、ルートキーで暗号化したコンテンツキーを記録媒体製造業者に提供して、ＥＫＢおよびルートキーで暗号化したコンテンツキーを格納した記録媒体を製造、あるいは自ら記録媒体を製造して流通させる。この構成により、ＥＫＢの処理可能な特定のカテゴリツリーに属するデバイスのみが記録媒体のＥＫＢを利用したコンテンツ記録再生時の暗号化処理、復号処理が可能となる。

【０３２９】

さらに、ＥＫＢリクエスタが、

［ｃ２］製造時に記録デバイスにＥＫＢを格納するようなフォーマットにおいて、記録デバイスの製造業社に取得したＥＫＢを与えるフォーマットホルダー。である場合、生成したＥＫＢ、ルートキーで暗号化したコンテンツキーを記録デバイス製造業者に提供して、ＥＫＢおよびルートキーで暗号化したコンテンツキーを格納した記録デバイスを製造、あるいは自ら記録デバイスを製造して流通させる。この構成により、ＥＫＢの処理可能な特定のカテゴリツリーに属するデバイスのみがＥＫＢを利用したコンテンツ記録再生時の暗号化処理、復号処理が可能となる。

【０３３０】

以上のような処理によってＥＫＢが発行されることになる。なお、ＥＫＢ発行処理プロセスにおける各エンティテイ、ＥＫＢリクエスタ、キー発行センター（ＫＤＣ）、ＴＬＣＥ間の通信においては必要に応じて相互認証処理、送信データの暗号化処理が行われる。また、その他のメッセージ暗号化処理、デジタル署名生成、検証処理を行なう構成としてもよい。なお、公開鍵暗号方式に基づく認証あるいは暗号通信を実行する場合は、各エンティテイ間において予め公開鍵を保有し合う手続きを行なっておく。

【０３３１】

（サブＥＫＢの単純集合を合成ＥＫＢとする構成例）

上述したサブＥＫＢから合成ＥＫＢを生成する処理においては、個々のサブＥＫＢに含まれる暗号化鍵データの配列を全体ツリーの上段から下段に至るように並び替える処理を行なっていた。次に、このような並び替え処理を実行することなく、各カテゴリツリーのＴＬＣＥの生成したサブＥＫＢをそのまま合成ＥＫＢに順次格納して合成ＥＫＢを生成する構成について説明する。

【０３３２】

図６０は、複数のカテゴリツリーのＴＬＣＥの生成したサブＥＫＢをそのままの形で複数

格納した合成ＥＫＢ６０００の例を示した図である。

【０３３３】

ＥＫＢの発行処理において、キー発行センター（ＫＤＣ）は、ＥＫＢリクエスタによって指定されたＥＫＢタイプ識別ナンバーに対応してＥＫＢタイプ定義リストに記録されたカテゴリツリーの管理エンティテイであるＴＬＣＥに対してサブＥＫＢの生成要求を発行し、各ＴＬＣＥから提出されたサブＥＫＢ６１１０，６１２０…を単に集めて合成ＥＫＢ内に格納する。ただし各カテゴリに属する機器がその合成ＥＫＢ中からその機器が処理可能な自デバイスの属するカテゴリに対応するサブＥＫＢを選択できるように、各サブＥＫＢ部分の大きさ（ｅｘ．データレングス）６１１１、そのサブＥＫＢがどのカテゴリ用のものかを表すデータ（ｅｘ．ノードＩＤ）６１１２を付加する。

【０３３４】

すなわち、格納対象として選択されたサブＥＫＢの各々には、サブＥＫＢ格納領域のデータ長を示すレングス、およびサブＥＫＢ識別データとしての各サブＥＫＢの対応カテゴリツリーのノード識別子としてのノードＩＤが対応付けられて格納される。また合成ＥＫＢに含まれるサブＥＫＢの数がヘッダ情報６２００として付加される。合成ＥＫＢの全データに基づいて署名（ｅｘ．認証局（ＣＡ）の署名）６３００が生成されて付加される。

【０３３５】

本方式に従って、前述の図５７を用いた説明に対応する合成ＥＫＢを生成すると、図６１に示すような合成ＥＫＢが生成されることになる。サブＥＫＢ６１１０の格納ＥＫＢは、図５５で説明したカテゴリツリーＡのＴＬＣＥの生成したサブＥＫＢ－（Ａ）そのものであり、タグパート：１０１，０１０，０００，１１１，１１１、キーパート：Ｅｎｃ（Ｋ０１０，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ０１１，Ｋｒｏｏｔ）となる。また、サブＥＫＢ６１２０の格納ＥＫＢは、図５６で説明したカテゴリツリーＢのＴＬＣＥの生成したサブＥＫＢ－（Ｂ）そのものであり、タグパート：１１０，０１０，０００，１１１，１１１、キーパート：Ｅｎｃ（Ｋ１１０，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ１１１，Ｋｒｏｏｔ）となる。

【０３３６】

また、前述の図５８、図５９を用いて説明したリボークデバイスがある場合の合成ＥＫＢは、図６２に示すデータ構成となる。サブＥＫＢ６１１０の格納ＥＫＢは、図５８で説明したカテゴリツリーＡのＴＬＣＥの生成したサブＥＫＢ－（Ａ’）そのものであり、サブＥＫＢ－（Ａ’）は、タグパート：１０１，０１０，０００，１１１，０００，００１，１１１，１１１、キーパート：Ｅｎｃ（Ｋ０１０，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ０１１１，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ０１１００，Ｋｒｏｏｔ）となる。また、リボークデバイスの発生していないサブＥＫＢ６１２０の格納ＥＫＢは、図５６で説明したカテゴリツリーＢのＴＬＣＥの生成したサブＥＫＢ－（Ｂ）そのものであり、タグパート：１１０，０１０，０００，１１１，１１１、キーパート：Ｅｎｃ（Ｋ１１０，Ｋｒｏｏｔ），Ｅｎｃ（Ｋ１１１，Ｋｒｏｏｔ）となる。

【０３３７】

このような構成をとることにより、各カテゴリに属するデバイスは自己のデバイスが属するカテゴリに対応するサブＥＫＢを選択して処理（復号）することが可能となる。従って、各カテゴリ（ＴＬＣＥ）ごとに、完全に任意の暗号アルゴリズムや鍵長を用いて、サブＥＫＢを生成することができる。すなわち、他のカテゴリに左右されず、ＴＬＣＥが暗号アルゴリズムや鍵長を決めることができる。

【０３３８】

キー発行センター（ＫＤＣ）にとっては、各ＴＬＣＥから集めたサブＥＫＢのタグ、および鍵データ部分を分解、組み直ししなくてよくなり、負荷が軽くなる。

【０３３９】

この方式に従ったＥＫＢを入手した機器は、自分が属するカテゴリのサブＥＫＢを見つけ、それを、自デバイスを管理するＴＬＣＥが定める独自の方法で処理することによりルートキーを得ることができる。他のサブＥＫＢを処理するための、他のカテゴリのＴＬＣＥが定めた方法は知る必要がなく、またサブＥＫＢにおいて個々の鍵を固定長で表すなどの

工夫が不要なため、理論的にはどの大きさの鍵でも用いることができるようになる。
【０３４０】
（リボケーション処理－（１））
複数カテゴリにおいて共通に使用可能なＥＫＢを利用した処理におけるリボークの発生に際して実行される処理について、以下説明する。暗号化コンテンツをネットワークまたはメデイアによって外部から受領してＥＫＢによって取得したキーを用いてコンテンツキーを取得してコンテンツ利用を行なう場合のリボーク処理についてまず説明する。
【０３４１】
図６３を参照しながら説明する。カテゴリツリーＡ，７１００とカテゴリツリーＢ，７２００において共通に使用されるＥＫＢ７０００が利用されている状況を想定する。また、カテゴリツリーＡ，７１００とカテゴリツリーＢ，７２００において共通に使用されるＥＫＢ７０００はＥＫＢタイプ定義リストでは、ＥＫＢタイプ識別ナンバーが＃１に定義されているものとする。
【０３４２】
このような状況において、コンテンツプロバイダは、ネットワークまたはメデイアによってコンテンツキーで暗号化したコンテンツを提供し、カテゴリツリーＡ，７１００とカテゴリツリーＢ，７２００に属するデバイスは、ＥＫＢ７０００を用いてルートキーの取得、ルートキーによる復号処理によるコンテンツキーの取得、コンテンツキーによる暗号化コンテンツの取得を実行してコンテンツを利用している。
【０３４３】
この状況でカテゴリツリーＡ，７１００に属するデバイスＡ１，７１２０の鍵データの漏洩など不正処理可能な状況が発覚し、デバイスＡ１，７１２０のリボークを行なうとする。
【０３４４】
この場合、カテゴリツリーＡ，７１００のＴＬＣＥは、キー発行センター（ＫＤＣ）に対してツリー変更通知（図５３参照）を実行し、キー発行センター（ＫＤＣ）は、受信したツリー変更通知に基づいて管理下の各ＴＬＣＥ、ＥＫＢリクエスタに対して通知する。この時点の通知は、ツリー変更通知を受領したことを知らせるのみであり、ＥＫＢタイプ定義リストの更新処理は実行されない。
【０３４５】
なお、リボーク発生に基づくツリー変更通知は、リボークの発生したカテゴリツリーにおいて処理可能なＥＫＢを利用しているエンティテイとしてのＥＫＢリクエスタに対してのみ、あるいはさらに、リボークの発生したカテゴリツリーと共有のＥＫＢが適用されている他のカテゴリツリーを管理するカテゴリ・エンティテイに対してのみ実行する構成としてもよい。この処理を行なうため、キー発行センター（ＫＤＣ）は、発行済みＥＫＢの利用者リストとして、ＥＫＢタイプ識別ナンバーとそのＥＫＢタイプを利用しているＥＫＢリクエスタとを対応付けたリストを保有する。
【０３４６】
リボーク処理を実行したカテゴリツリーのデバイスを対象としてコンテンツの配信を実行しているＥＫＢリクエスタとしてのコンテンツプロバイダは、リボーク処理対象以外のデバイスにおいてのみ処理可能な更新されたＥＫＢを生成するようにキー発行センター（ＫＤＣ）に対してＥＫＢ発行要求を行なう。この場合、ＥＫＢリクエスタとしてのコンテンツプロバイダは、カテゴリツリーＡ，７１００とカテゴリツリーＢ，７２００において共通に使用されるＥＫＢのタイプとして定義されているＥＫＢタイプ識別ナンバー＃１を指定する。また、新たなルートキーをＥＫＢリクエスタ自ら生成してＫＤＣに送付するか、あるいは新たなルートキーの生成をＫＤＣに依頼する。
【０３４７】
キー発行センター（ＫＤＣ）は、指定されたＥＫＢタイプ識別ナンバー＃１に基づいて、ＥＫＢタイプ定義リストを参照し、対応するカテゴリツリーのノードに基づいてカテゴリツリーＡ，７１００とカテゴリツリーＢ，７２００のＴＬＣＥに対して新たなルートキー

を正当なデバイスにおいて取得可能なサブＥＫＢの生成を依頼する。
【０３４８】
カテゴリツリーＡ，７１００とカテゴリツリーＢ，７２００のＴＬＣＥの各々は、依頼に基づいてサブＥＫＢを生成する。この場合、カテゴリツリーＡ，７１００においてはリボークされたデバイスＡ１，７１２０を排除した他のデバイスにおいてのみ新規のルートキーを取得可能なサブＥＫＢ－（Ａ）が生成される。カテゴリツリーＢ，７２００ではリボークされたデバイスが存在しなければ、カテゴリに属するすべてのデバイスにおいて新規のルートキーを取得可能なサブＥＫＢ－（Ｂ）を生成してキー発行センター（ＫＤＣ）に対して送信する。
【０３４９】
キー発行センター（ＫＤＣ）は各ＴＬＣＥから受信したサブＥＫＢに基づいて合成ＥＫＢを前述した方法に従って生成し、生成したＥＫＢをＥＫＢリクエスタ（ｅｘ．コンテンツプロバイダ）に送信する。
【０３５０】
ＥＫＢリクエスタ（ｅｘ．コンテンツプロバイダ）はキー発行センター（ＫＤＣ）から受領した新たなＥＫＢを適用してコンテンツ配信を行なう。具体的にはコンテンツキーで暗号化したコンテンツを提供し、ＥＫＢの復号によって得られるルートキーでコンテンツキーを暗号化して提供する。カテゴリツリーＡ，７１００とカテゴリツリーＢ，７２００に属するデバイスは、ＥＫＢを用いてルートキーの取得、ルートキーによる復号処理によるコンテンツキーの取得、コンテンツキーによる暗号化コンテンツの取得を実行してコンテンツが利用可能である。ただし、カテゴリツリーＡ，７１００のリボークデバイスＡ１，７１２０は更新されたＥＫＢを処理できないのでコンテンツの利用ができなくなる。
【０３５１】
なお、上述の説明においては、キー発行センター（ＫＤＣ）はＴＬＣＥからのツリー変更通知を受領した場合、その時点では、ＥＫＢタイプ定義リストの更新処理を実行しない例を説明したが、ＫＤＣがツリー変更通知を受領した時点で、キー発行センター（ＫＤＣ）がツリー変更情報に基づいて、ＥＫＢタイプ定義リストの更新処理、ＥＫＢ更新処理を実行し、各ＥＫＢリクエスタ、ＴＬＣＥに更新されたＥＫＢタイプ定義リストを送付する構成としてもよい。
【０３５２】
（リボケーション処理－（２））
次に、例えば記録デバイスあるいは記録媒体にＥＫＢを格納した構成で、記録媒体に対してユーザが様々なコンテンツを暗号化して記録し暗号化処理、復号処理に必要となるキーを記録デバイスあるいは記録媒体に格納したＥＫＢから取得されるルートキーを用いたものとするいわゆる自己記録型の形態におけるリボーク処理に伴う処理について説明する。
【０３５３】
図６４を参照しながら説明する。カテゴリツリーＡ，８１００とカテゴリツリーＢ，８２００において共通に使用されるＥＫＢ８０００が利用されている状況を想定する。すなわちカテゴリツリーＡ，８１００とカテゴリツリーＢ，８２００において共通に使用される記録デバイスあるいは記録媒体には、共通のＥＫＢが格納され、ユーザは、ＥＫＢを利用したコンテンツ暗号化、復号処理によるコンテンツ記録再生を行なっているものとする。なお、カテゴリツリーＡ，８１００とカテゴリツリーＢ，８２００において共通に使用されるＥＫＢ８０００はＥＫＢタイプ定義リストでは、ＥＫＢタイプ識別ナンバーが＃１に定義されているものとする。
【０３５４】
この状況でカテゴリツリーＡ，８１００に属するデバイスＡ１，８１２０の鍵データの漏洩など不正処理可能な状況が発覚し、デバイスＡ１，８１２０のリボークを行なうとする。
【０３５５】
この場合、カテゴリツリーＡ，８１００のＴＬＣＥは、キー発行センター（ＫＤＣ）に対

してツリー変更通知（図５３参照）を実行し、キー発行センター（ＫＤＣ）は、受信したツリー変更通知に基づいて管理下の各ＴＬＣＥ、関連ＥＫＢリクエスタに対して通知する。この時点の通知は、ツリー変更通知を受領したことを知らせるのみであり、ＥＫＢタイプ定義リストの更新処理は実行されない。

【０３５６】

リボーク処理を実行したカテゴリツリーのＴＬＣＥは、リボークデバイスＡ１，８１２０における将来におけるＥＫＢを利用した新たなコンテンツ処理を停止させるため、自らＥＫＢリクエスタとして、リボーク処理対象以外のデバイスにおいてのみ処理可能な更新されたＥＫＢを生成するようにキー発行センター（ＫＤＣ）に対してＥＫＢ発行要求を行なう。この場合、ＥＫＢリクエスタとしてのＴＬＣＥは、カテゴリツリーＡ，８１００とカテゴリツリーＢ，８２００において共通に使用されるＥＫＢのタイプとして定義されているＥＫＢタイプ識別ナンバー＃１を指定する。また、新たなルートキーをＥＫＢリクエスタ自ら生成してＫＤＣに送付するか、あるいは新たなルートキーの生成をＫＤＣに依頼する。

【０３５７】

キー発行センター（ＫＤＣ）は、指定されたＥＫＢタイプ識別ナンバー＃１に基づいて、ＥＫＢタイプ定義リストを参照し、対応するカテゴリツリーのノードに基づいてカテゴリツリーＡ，８１００とカテゴリツリーＢ，８２００のＴＬＣＥに対して新たなルートキーを正当なデバイスにおいて取得可能なサブＥＫＢの生成を依頼する。

【０３５８】

カテゴリツリーＡ，８１００とカテゴリツリーＢ，８２００のＴＬＣＥの各々は、依頼に基づいてサブＥＫＢを生成する。この場合、カテゴリツリーＡ，８１００においてはリボークされたデバイスＡ１，８１２０を排除した他のデバイスにおいてのみ新規のルートキーを取得可能なサブＥＫＢ－（Ａ）が生成される。カテゴリツリーＢ，８２００ではリボークされたデバイスが存在しなければ、カテゴリに属するすべてのデバイスにおいて新規のルートキーを取得可能なサブＥＫＢ－（Ｂ）を生成してキー発行センター（ＫＤＣ）に対して送信する。

【０３５９】

キー発行センター（ＫＤＣ）は各ＴＬＣＥから受信したサブＥＫＢに基づいて合成ＥＫＢを前述した方法に従って生成し、生成したＥＫＢを各ＴＬＣＥ（ｅｘ．フォーマットホルダー）に送信する。

【０３６０】

各ＴＬＣＥ（ｅｘ．フォーマットホルダー）はキー発行センター（ＫＤＣ）から受領した新たなＥＫＢを各デバイスに配信して、ＥＫＢの更新を実行させる。カテゴリツリーＡ，８１００とカテゴリツリーＢ，８２００に属するデバイスは、新たなコンテンツの記録デバイスに対する記録を更新したＥＫＢを用いて取得したルートキーを適用した暗号化処理として実行する。新たなＥＫＢを用いて暗号化記録されたコンテンツは対応するＥＫＢを適用した場合にのみ復号可能となるので、リボークされたデバイスにおいては利用不可能となる。

【０３６１】

以上、特定の実施例を参照しながら、本発明について詳解してきた。しかしながら、本発明の要旨を逸脱しない範囲で当業者が該実施例の修正や代用を成し得ることは自明である。すなわち、例示という形態で本発明を開示してきたのであり、限定的に解釈されるべきではない。本発明の要旨を判断するためには、冒頭に記載した特許請求の範囲の欄を参酌すべきである。

【０３６２】

【発明の効果】

以上、説明したように、本発明の情報処理システムおよび方法によれば、カテゴリに基づいて区分され、カテゴリ・エンティティによって管理されるサブツリーを複数有するキーツリーを構成し、キーツリーを構成するパスを選択して選択パス上の下位キーによる上位

キーの暗号化処理データからなるＥＫＢを生成してデバイスに提供する構成において、ＥＫＢタイプ識別子と、ＥＫＢ処理可能な1以上のカテゴリツリーの識別データとを対応付けたＥＫＢタイプ定義リストに基づいてＥＫＢの発行管理を実行するように構成したので、ＥＫＢ生成要求者としてのＥＫＢリクエスタが容易に適用対象となるカテゴリを選択できる。

【０３６３】

また、本発明の情報処理システムおよび方法によれば、ＥＫＢタイプ定義リストに定義されたＥＫＢの処理可能なカテゴリツリーにおけるリボーク等による状態変化発生に関する通知処理を、ＥＫＢの利用エンティテイに対して実行する構成としたので、ＥＫＢリクエスタ他のエンティテイは、常に最新のＥＫＢタイプ定義情報に基づく処理が可能となる。

【図面の簡単な説明】

【図１】本発明の情報処理システムの構成例を説明する図である。

【図２】本発明の情報処理システムにおいて適用可能な記録再生装置の構成例を示すブロック図である。

【図３】本発明の情報処理システムにおける各種キー、データの暗号化処理について説明するツリー構成図である。

【図４】本発明の情報処理システムにおける各種キー、データの配布に使用される有効化キーブロック（ＥＫＢ）の例を示す図である。

【図５】本発明の情報処理システムにおけるコンテンツキーの有効化キーブロック（ＥＫＢ）を使用した配布例と復号処理例を示す図である。

【図６】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）のフォーマット例を示す図である。

【図７】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）のタグの構成を説明する図である。

【図８】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、コンテンツキー、コンテンツを併せて配信するデータ構成例を示す図である。

【図９】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、コンテンツキー、コンテンツを併せて配信した場合のデバイスでの処理例を示す図である。

【図１０】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）とコンテンツを記録媒体に格納した場合の対応について説明する図である。

【図１１】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、コンテンツキーを送付する処理を従来の送付処理と比較した図である。

【図１２】本発明の情報処理システムにおいて適用可能な共通鍵暗号方式による認証処理シーケンスを示す図である。

【図１３】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、認証キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その１）である。

【図１４】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、認証キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その２）である。

【図１５】本発明の情報処理システムにおいて適用可能な公開鍵暗号方式による認証処理シーケンスを示す図である。

【図１６】本発明の情報処理システムにおいて公開鍵暗号方式による認証処理を用いて有効化キーブロック（ＥＫＢ）と、コンテンツキーを併せて配信する処理を示す図である。

【図１７】本発明の情報処理システムにおいて有効化キーブロック（ＥＫＢ）と、暗号化プログラムデータを併せて配信する処理を示す図である。

【図１８】本発明の情報処理システムにおいて適用可能なコンテンツ・インテグリティ・チェック値（ＩＣＶ）の生成に使用するＭＡＣ値生成例を示す図である。

【図１９】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、ＩＣＶ生成キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その１）である。

【図２０】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）と、ＩＣＶ

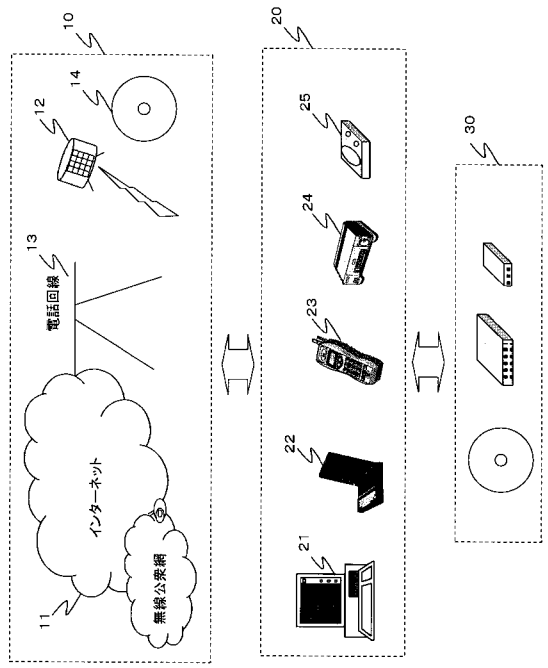
生成キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その２）である。
【図２１】本発明の情報処理システムにおいて適用可能なコンテンツ・インテグリティ・チェック値（ＩＣＶ）をメディアに格納した場合のコピー防止機能を説明する図である。
【図２２】本発明の情報処理システムにおいて適用可能なコンテンツ・インテグリティ・チェック値（ＩＣＶ）をコンテンツ格納媒体と別に管理する構成を説明する図である。
【図２３】本発明の情報処理システムにおける階層ツリー構造のカテゴリ分類の例を説明する図である。
【図２４】本発明の情報処理システムにおける簡略化有効化キーブロック（ＥＫＢ）の生成過程を説明する図である。
【図２５】本発明の情報処理システムにおける有効化キーブロック（ＥＫＢ）の生成過程を説明する図である。
【図２６】本発明の情報処理システムにおける簡略化有効化キーブロック（ＥＫＢ）（例１）を説明する図である。
【図２７】本発明の情報処理システムにおける簡略化有効化キーブロック（ＥＫＢ）（例２）を説明する図である。
【図２８】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成について説明する図である。
【図２９】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成の詳細について説明する図である。
【図３０】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成について説明する図である。
【図３１】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成でのリザーブノードについて説明する図である。
【図３２】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成での新規カテゴリツリー登録処理シーケンスについて説明する図である。
【図３３】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成での新規カテゴリツリーと上位カテゴリツリーの関係について説明する図である。
【図３４】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成で用いるサブＥＫＢについて説明する図である。
【図３５】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成でのデバイスリボーク処理について説明する図である。
【図３６】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成でのデバイスリボーク処理シーケンスについて説明する図である。
【図３７】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成でのデバイスリボーク時の更新サブＥＫＢについて説明する図である。
【図３８】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成でのカテゴリツリーリボーク処理について説明する図である。
【図３９】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成でのカテゴリツリーリボーク処理シーケンスについて説明する図である。
【図４０】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成でのリボークカテゴリツリーと上位カテゴリツリーの関係について説明する図である。
【図４１】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成でのケイパビリテイ設定について説明する図である。
【図４２】本発明の情報処理システムにおける階層ツリー構造のカテゴリツリー管理構成でのケイパビリテイ設定について説明する図である。
【図４３】本発明の情報処理システムにおけるキー発行センター（ＫＤＣ）の管理するケイパビリティ管理テーブル構成を説明する図である。
【図４４】本発明の情報処理システムにおけるキー発行センター（ＫＤＣ）の管理するケイパビリティ管理テーブルに基づくＥＫＢ生成処理フロー図である。

【図４５】本発明の情報処理システムにおける新規カテゴリツリー登録時のケイパビリティ通知処理を説明する図である。
【図４６】本発明の情報処理システムにおけるカテゴリツリーの構成を説明する図である。
【図４７】本発明の情報処理システムにおけるＥＫＢリクエスタ、キー発行センター、トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）との関係、処理例を説明する図である。
【図４８】本発明の情報処理システムにおけるＥＫＢリクエスタ、キー発行センター、トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）のハード構成例を説明する図である。
【図４９】本発明の情報処理システムにおけるデバイスの保有するデバイスノードキー（ＤＮＫ）について説明する図である。
【図５０】本発明の情報処理システムにおけるＥＫＢタイプ定義リストのデータ構成を説明する図である。
【図５１】本発明の情報処理システムにおけるＥＫＢタイプ登録処理フローを示す図である。
【図５２】本発明の情報処理システムにおけるＥＫＢタイプ無効化処理フローを示す図である。
【図５３】本発明の情報処理システムにおけるツリー変更通知処理フローを示す図である。
【図５４】本発明の情報処理システムにおけるＥＫＢタイプリスト要求処理フローを示す図である。
【図５５】本発明の情報処理システムにおけるサブＥＫＢの生成処理を説明する図である。
【図５６】本発明の情報処理システムにおけるサブＥＫＢの生成処理を説明する図である。
【図５７】本発明の情報処理システムにおけるサブＥＫＢから合成したＥＫＢを生成する処理を説明する図である。
【図５８】本発明の情報処理システムにおけるリボークデバイスがある場合のサブＥＫＢの生成処理を説明する図である。
【図５９】本発明の情報処理システムにおけるリボークデバイスがある場合のサブＥＫＢから合成したＥＫＢを生成する処理を説明する図である。
【図６０】本発明の情報処理システムにおけるサブＥＫＢから合成したＥＫＢのデータ構成を説明する図である。
【図６１】本発明の情報処理システムにおけるサブＥＫＢから合成したＥＫＢのデータ構成を説明する図である。
【図６２】本発明の情報処理システムにおけるリボークデバイスがある場合のサブＥＫＢから合成したＥＫＢのデータ構成を説明する図である。
【図６３】本発明の情報処理システムにおけるデータ配信型のシステムにおけるリボーク処理を説明する図である。
【図６４】本発明の情報処理システムにおける自己記録型のシステムにおけるリボーク処理を説明する図である。
【符号の説明】
１０　コンテンツ配信側
１１　インターネット
１２　衛星放送
１３　電話回線
１４　メディア
２０　コンテンツ受信側
２１　パーソナルコンピュータ（ＰＣ）
２２　ポータブルデバイス（ＰＤ）

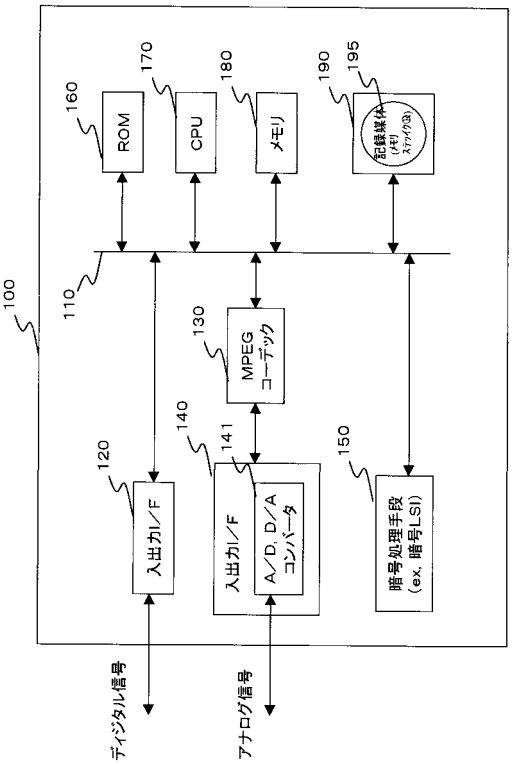
２３　携帯電話、ＰＤＡ
２４　記録再生器
２５　再生専用器
３０　メディア
１００　記録再生装置
１１０　バス
１２０　入出力Ｉ／Ｆ
１３０　ＭＰＥＧコーデック
１４０　入出力Ｉ／Ｆ
１４１　Ａ／Ｄ，Ｄ／Ａコンバータ
１５０　暗号処理手段
１６０　ＲＯＭ
１７０　ＣＰＵ
１８０　メモリ
１９０　ドライブ
１９５　記録媒体
６０１　バージョン
６０２　デプス
６０３　データポインタ
６０４　タグポインタ
６０５　署名ポインタ
６０６　データ部
６０７　タグ部
６０８　署名
１１０１　記録デバイス
２３０１　ルートキー
２３０２　ノードキー
２３０３　リーフキー
２３０４　カテゴリノード
２３０６　サブカテゴリノード
２７０１　カテゴリツリー
２７０２　サブルート
２８１１，２８５１　サブルート
２８１２，２８５２　カテゴリツリー末端ノード
２９０１，２９０２　カテゴリツリー
２９５０　リザーブノード
２９７０　管理末端ノード
３０１１，３０１２，３０１３　カテゴリツリー
３０２１，３０２２，３０２３　リザーブノード
３２０１　末端ノード
３２０２　頂点ノード
３３０１，３３０２　末端ノード
３３０３　新規カテゴリツリー追加末端ノード
３４１０，３４２０，３４３０　カテゴリツリー
３４１１，３４２１，３４３１　サブルート
３４３２　リボークデバイスノード
３６０１　末端ノード
３７１０，３７２０，３７３０　カテゴリツリー
３７１１，３７２１，３７３１　サブルート
３９０１　末端ノード

３９０２　頂点ノード（サブルートノード）
４００１～４００５　カテゴリツリー
４０２１，４０２２，４０２３　カテゴリツリー
４１０１～４１０５　カテゴリツリー
４５１０　管理エンティテイ
４５１１　キー発行センター（ＫＤＣ）
４５２０　ＥＫＢリクエスタ
４５３０　トップレベル・カテゴリ・エンティテイ（ＴＬＣＥ）
４５４０　メディア製造者
４５５０　デバイス製造者
４５５１　ＤＮＫＥデバイス製造者
４５５２　ＤＮＫデバイス製造者
５１００　カテゴリツリーＡ
５１２０　デバイスノードキー領域
５１５０　リボークデバイス
５２００　カテゴリツリーＢ
５２２０　デバイスノードキー領域
５３００　ルートツリー
６０００　ＥＫＢ
６１１０，６１２０　サブＥＫＢ
６１１１，６１２１　レングスデータ
６１１２，６１２２　カテゴリ識別データ
６２００　ヘッダ情報
６３００　署名データ
７０００　ＥＫＢ
７１００　カテゴリツリーＡ
７１２０　リボークデバイス
７２００　カテゴリツリーＢ
８０００　ＥＫＢ
８１００　カテゴリツリーＡ
８１２０　リボークデバイス
８２００　カテゴリツリーＢ

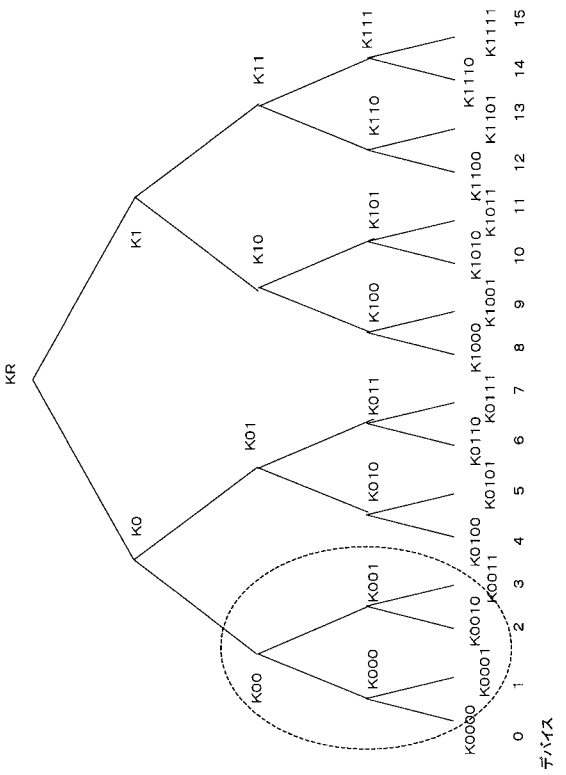
【図１】



【図２】



【図３】



【図４】

（A） 有効化キーブロック(EKB:Enabling Key Block) 例1

デバイス0, 1, 2にバージョン:tのノードキーを送付

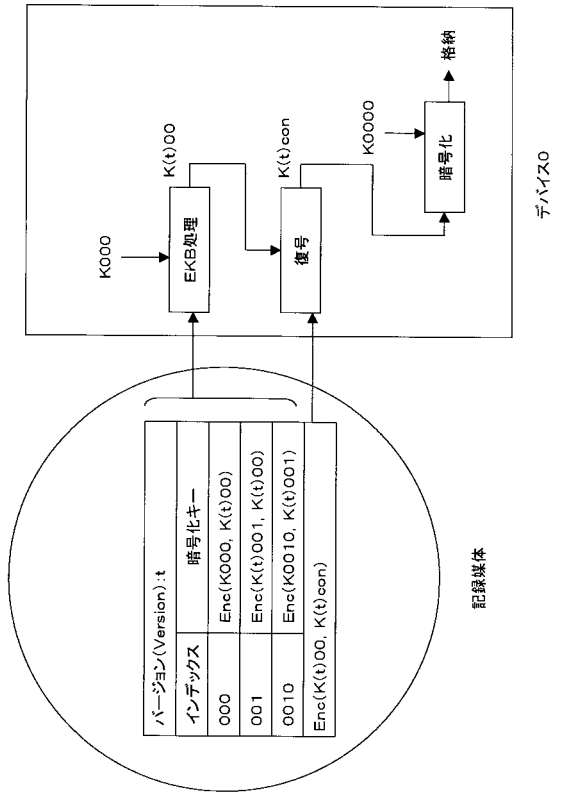
バージョン(Version):t	
インデックス	暗号化キー
0	Enc(K(t)0, K(t)R)
00	Enc(K(t)00, K(t)0)
000	Enc(K000, K(t)00)
001	Enc(K(t)001, K(t)00)
0010	Enc(K0010, K(t)001)

（B） 有効化キーブロック(EKB:Enabling Key Block) 例2

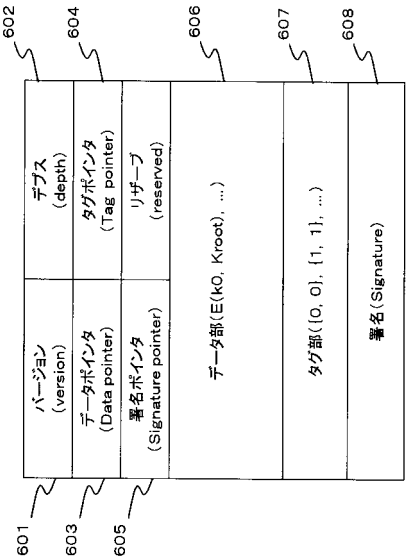
デバイス0, 1, 2にバージョン:tのノードキーを送付

バージョン(Version):t	
インデックス	暗号化キー
000	Enc(K000, K(t)00)
001	Enc(K(t)001, K(t)00)
0010	Enc(K0010, K(t)001)

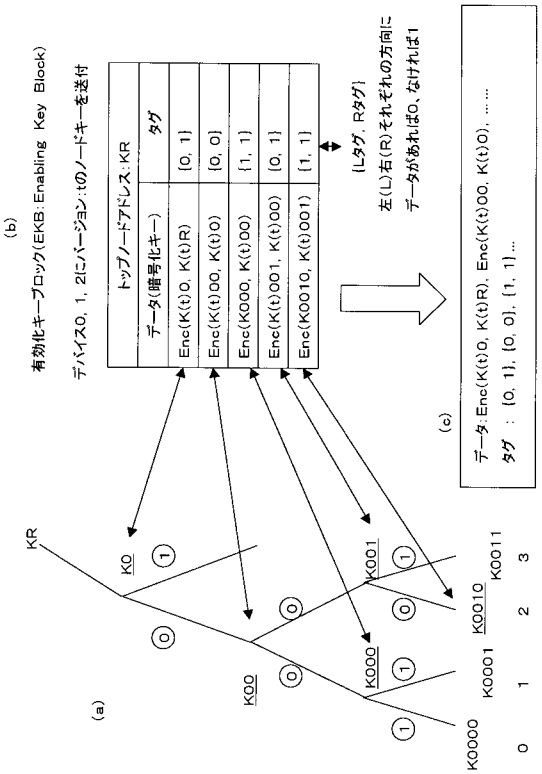
【図５】


K000
EKB処理
K(t)00
復号
K(t)con
K0000
暗号化
格納
デバイス0
記録媒体
バージョン(Version):t
インデックス
暗号化キー
000
Enc(K000, K(t)00)
001
Enc(K(t)001, K(t)00)
0010
Enc(K0010, K(t)001)
Enc(K(t)00, K(t)con)

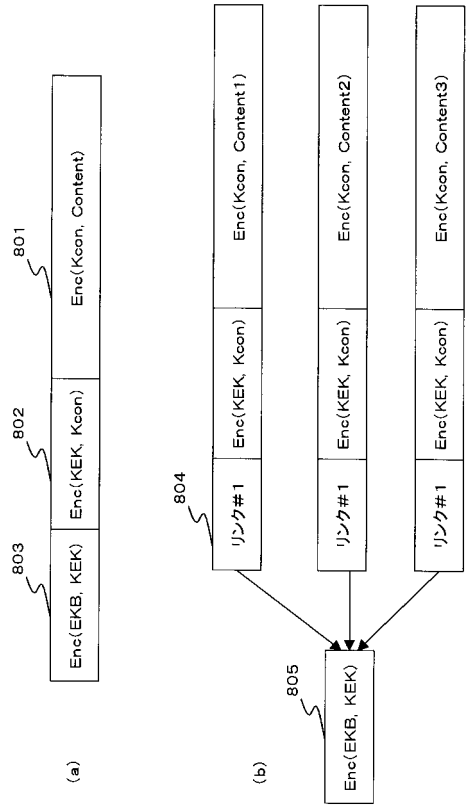
【図６】


602
604
606
607
608
バージョン (version)
デプス (depth)
データポインタ (Data pointer)
タグポインタ (Tag pointer)
署名ポインタ (Signature pointer)
リザーブ (reserved)
データ部(E(k0, Kroot), ...)
タグ部({0, 0}, {1, 1}, ...)
署名(Signature)
601
603
605

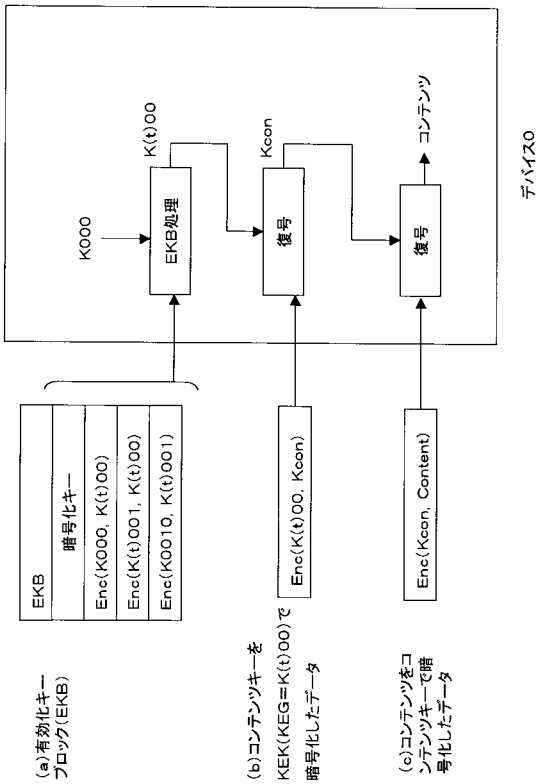
【図７】


(b)
有効化キーブロック(EKB: Enabling Key Block)
デバイス0, 1, 2にバージョン:tのノードキーを送付
トップノードアドレス:KR
データ(暗号化キー)
タグ
Enc(K(t)0, K(t)R)
{0, 1}
Enc(K(t)00, K(t)0)
{0, 0}
Enc(K000, K(t)00)
{1, 1}
Enc(K(t)001, K(t)00)
{0, 1}
Enc(K0010, K(t)001)
{1, 1}
{Lタグ, Rタグ}
左(L)右(R)それぞれの方向に
データがあれば0, なければ1
(c)
データ:Enc(K(t)0, K(t)R), Enc(K(t)00, K(t)0), ...
タグ : {0, 1}, {0, 0}, {1, 1} ...
KR
K0
K00
K000
K001
K0010
K0011
K0000
K0001
0
1
2
3
(a)

【図８】

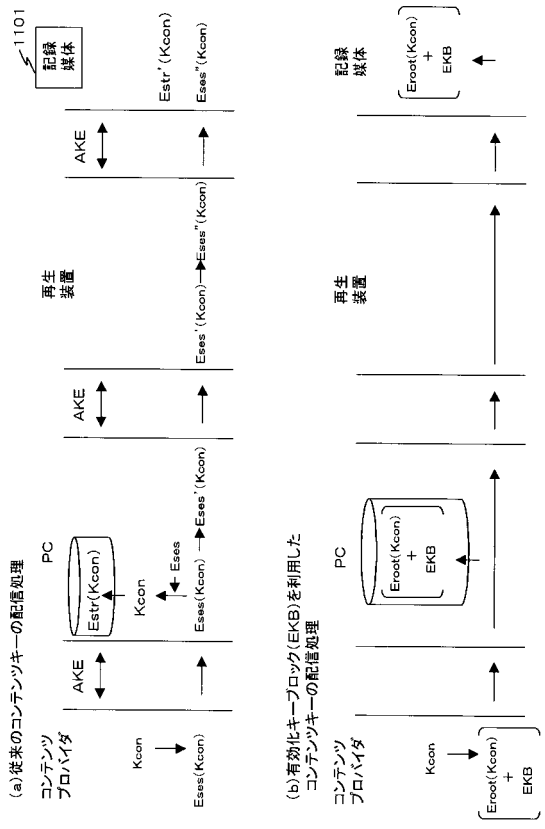

(a)
801
802
803
Enc(EKB, KEK)
Enc(KEK, Kcon)
Enc(Kcon, Content)
(b)
804
805
リンク#1
Enc(KEK, Kcon)
Enc(Kcon, Content1)
Enc(Kcon, Content2)
Enc(Kcon, Content3)
Enc(EKB, KEK)

【図９】

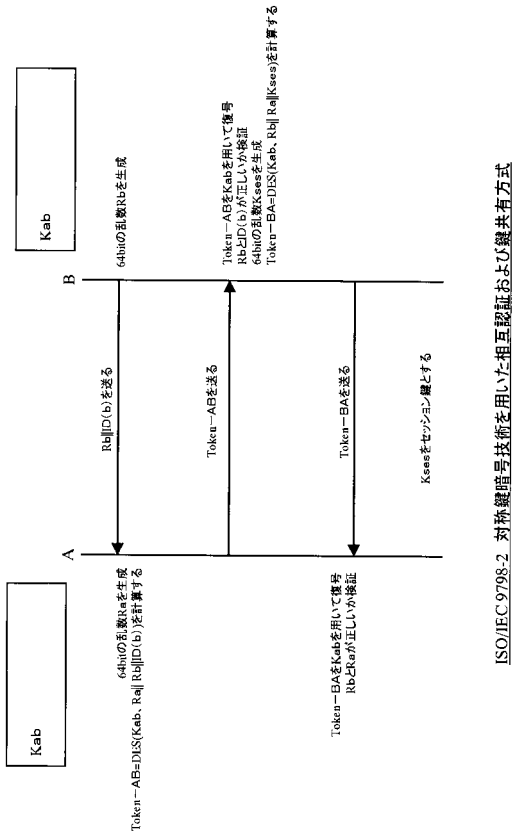


【図１０】

【図１１】

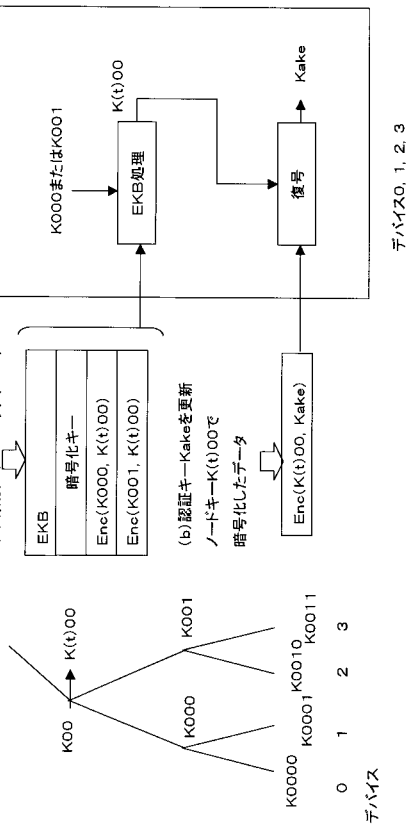


【図１２】

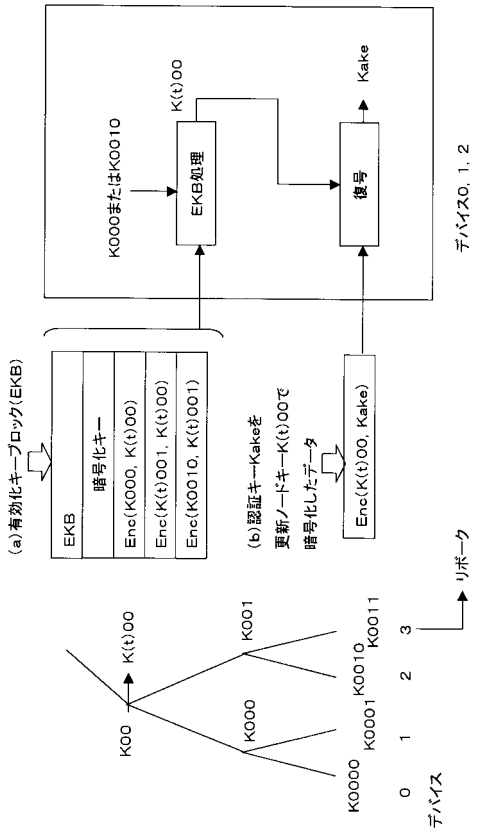


ISO/IEC 9798-2 対称鍵暗号技術を用いた相互認証および鍵共有方式

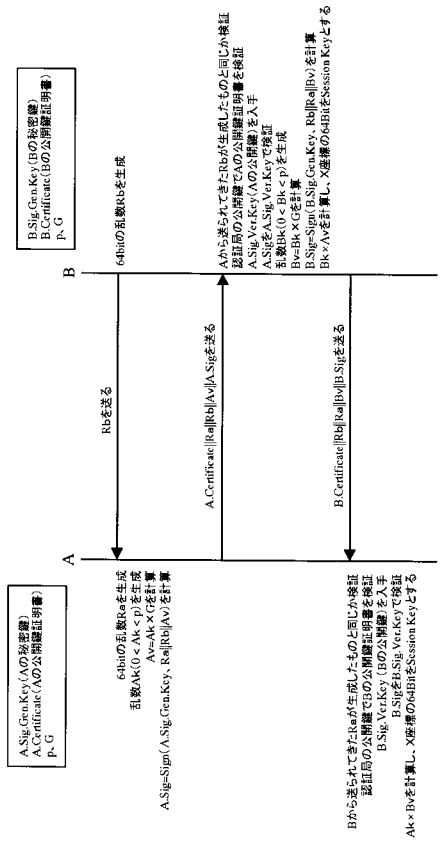
【図１３】



【図１４】

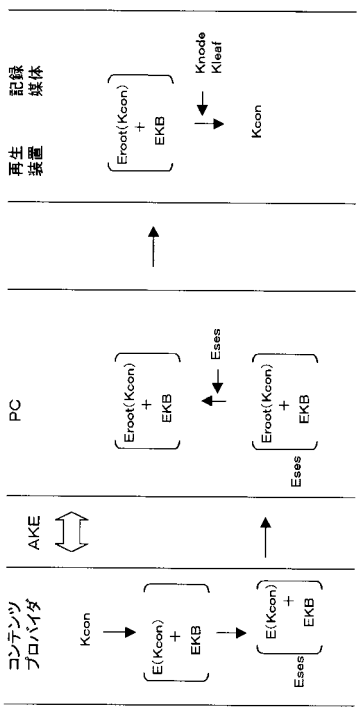


【図１５】

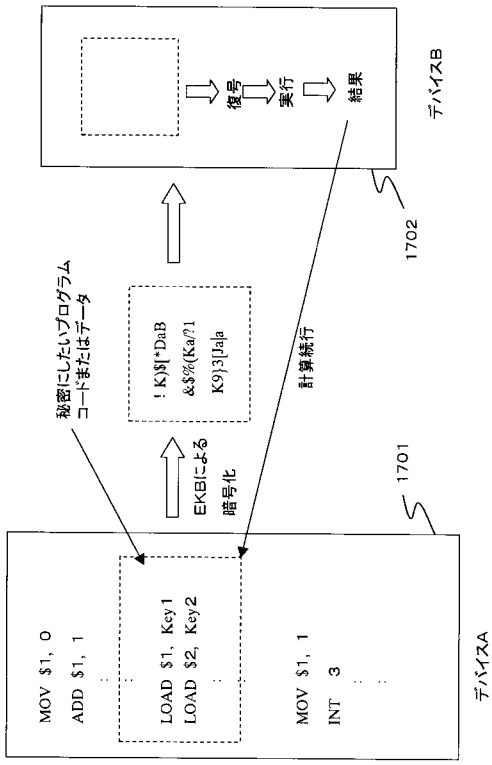


ISO/IEC 9798-3 非対称鍵暗号技術を用いた相互認証および鍵共有方式

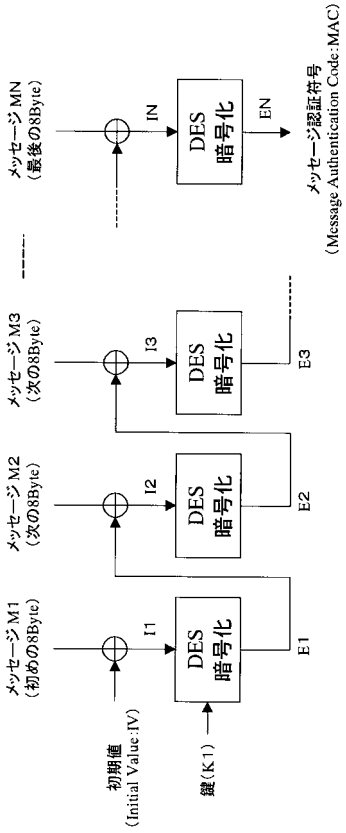
【図１６】



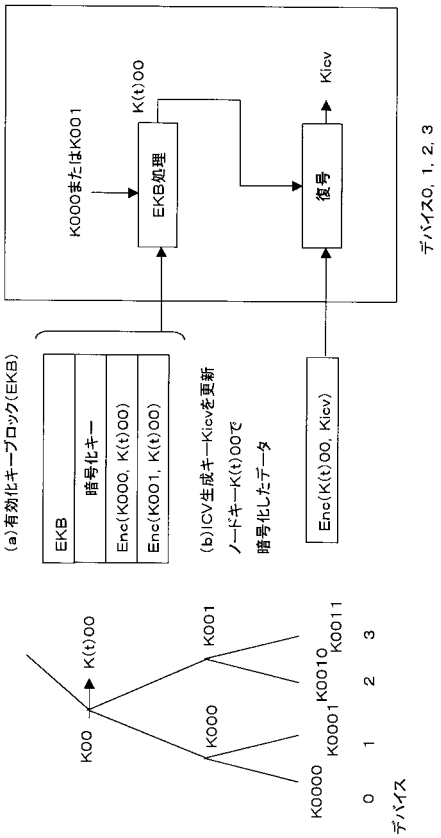
【図１７】



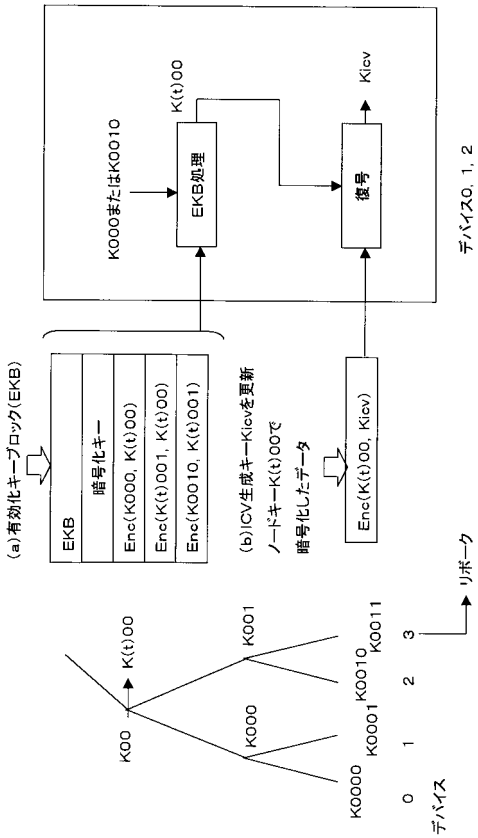
【図１８】



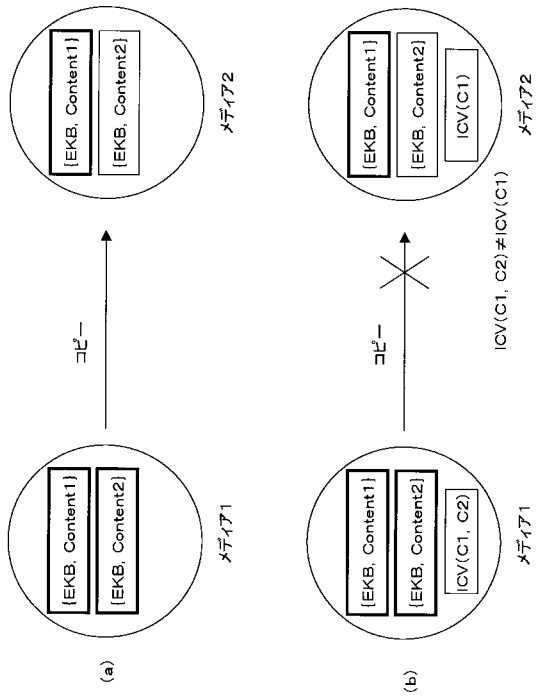
【図１９】



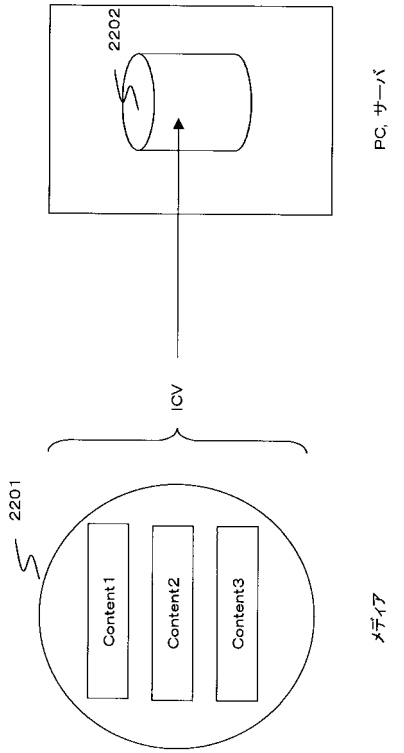
【図２０】



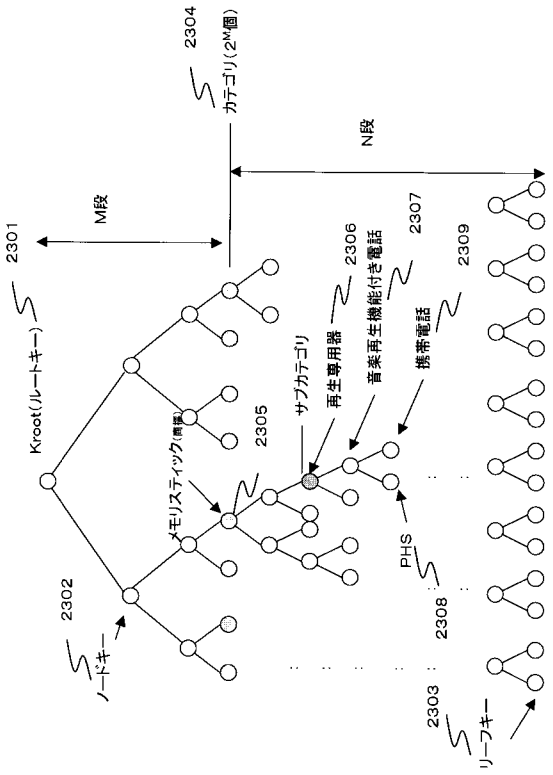
【図２１】



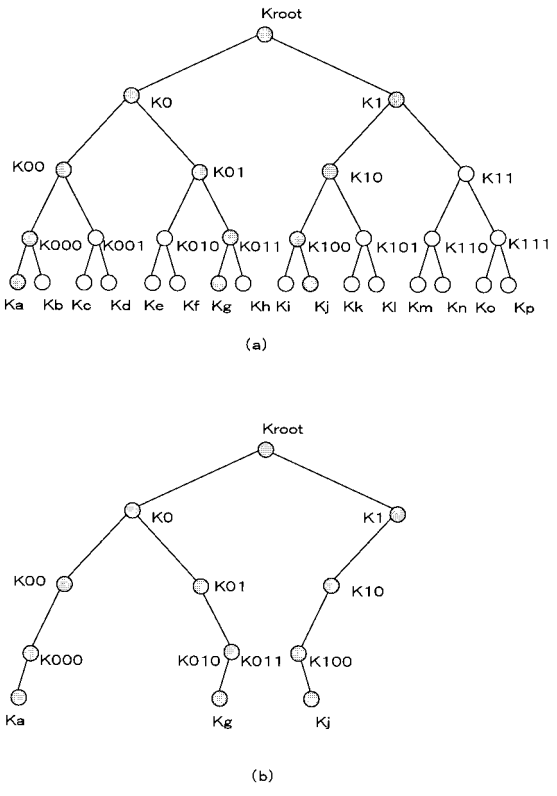
【図２２】



【図２３】



【図２４】

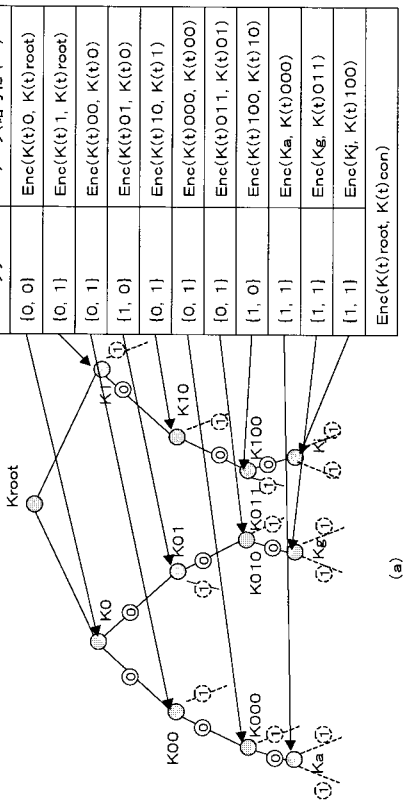


【図２５】

有効化キーブロック(EKB: Enabling Key Block)を用いた
デバイスKa, Kg, Kjへのバージョンｔのコンテンツキー送付処理

タグ	データ(暗号化キー)
{0, 0}	Enc(K(t)0, K(t)root)
{0, 1}	Enc(K(t)1, K(t)root)
{0, 1}	Enc(K(t)00, K(t)0)
{1, 0}	Enc(K(t)01, K(t)0)
{0, 1}	Enc(K(t)10, K(t)1)
{0, 1}	Enc(K(t)000, K(t)00)
{0, 1}	Enc(K(t)011, K(t)01)
{1, 0}	Enc(K(t)100, K(t)10)
{1, 1}	Enc(Ka, K(t)000)
{1, 1}	Enc(Kg, K(t)011)
{1, 1}	Enc(Kj, K(t)100)
	Enc(K(t)root, K(t)con)

(b)

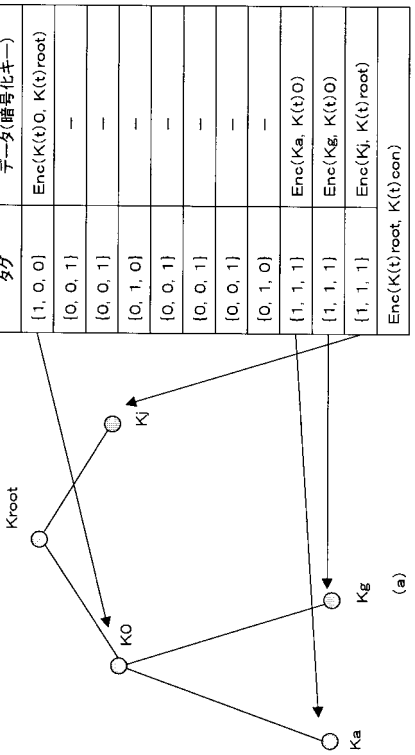


【図２６】

簡略化した有効化キーブロック(EKB: Enabling Key Block)を用いた
デバイスKa, Kg, Kjへのバージョンｔのコンテンツキー送付処理

タグ	データ(暗号化キー)
{1, 0, 0}	Enc(K(t)0, K(t)root)
{0, 0, 1}	—
{0, 0, 1}	—
{0, 1, 0}	—
{0, 0, 1}	—
{0, 0, 1}	—
{0, 1, 0}	—
{1, 1, 1}	Enc(Ka, K(t)0)
{1, 1, 1}	Enc(Kg, K(t)0)
{1, 1, 1}	Enc(Kj, K(t)root)
	Enc(K(t)root, K(t)con)

(b)

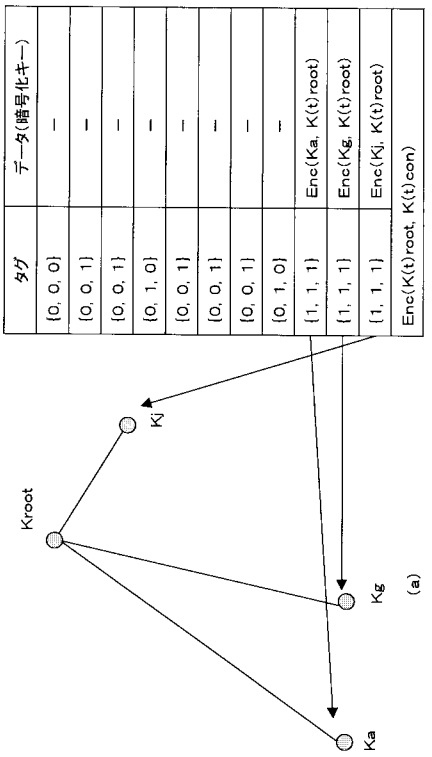


【図２７】

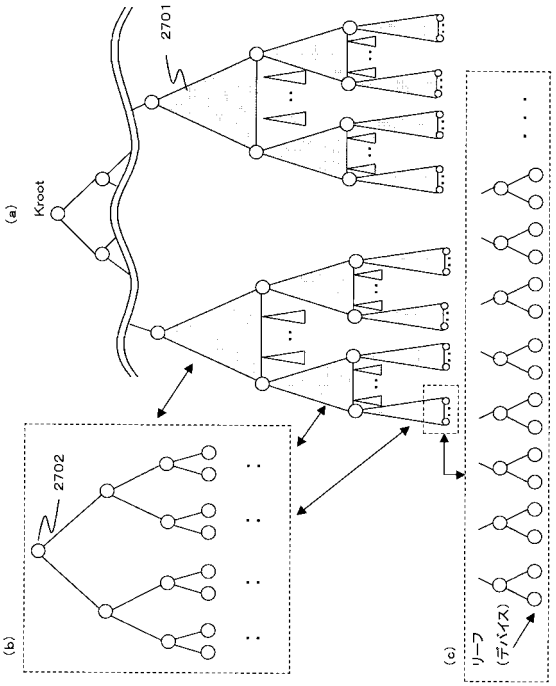
簡略化した有効化キーブロック(EKB: Enabling Key Block)を用いた
デバイスKa, Kg, Kjへのバージョンｔのコンテンツキー送付処理

タグ	データ(暗号化キー)
{0, 0, 0}	—
{0, 0, 1}	—
{0, 0, 1}	—
{0, 1, 0}	—
{0, 0, 1}	—
{0, 0, 1}	—
{0, 1, 0}	—
{1, 1, 1}	Enc(Ka, K(t)root)
{1, 1, 1}	Enc(Kg, K(t)root)
{1, 1, 1}	Enc(Kj, K(t)root)
	Enc(K(t)root, K(t)con)

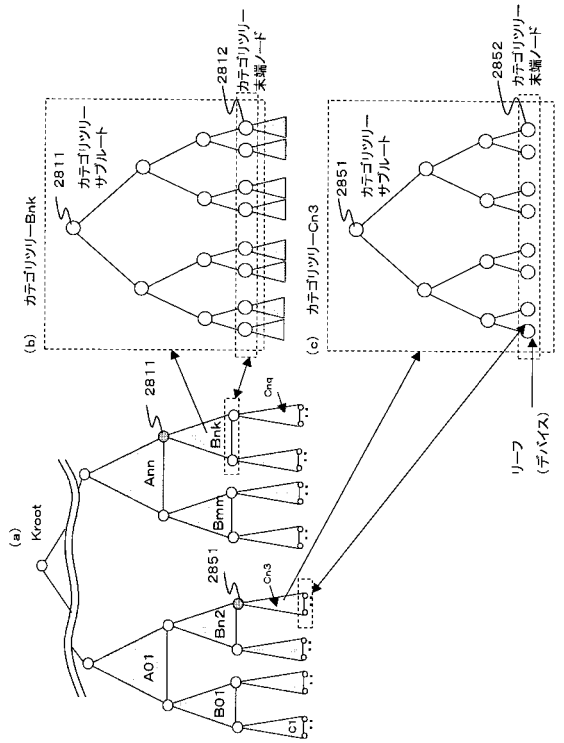
(b)



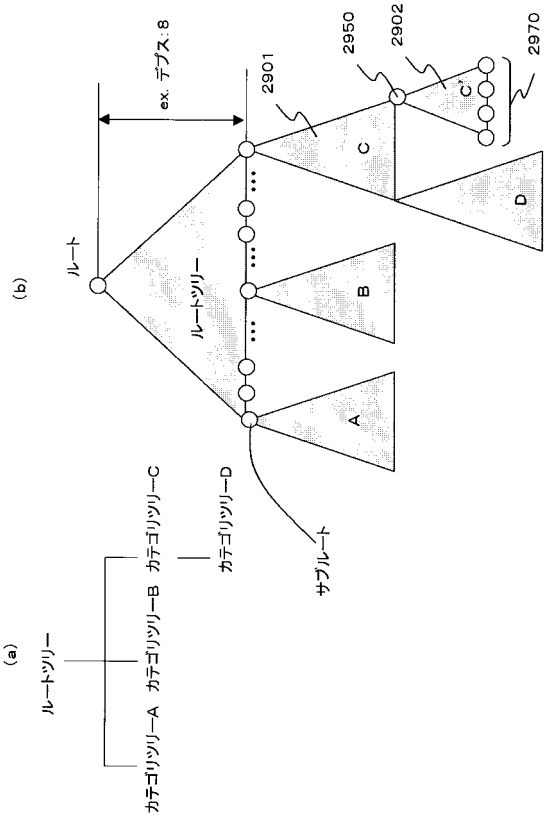
【図２８】



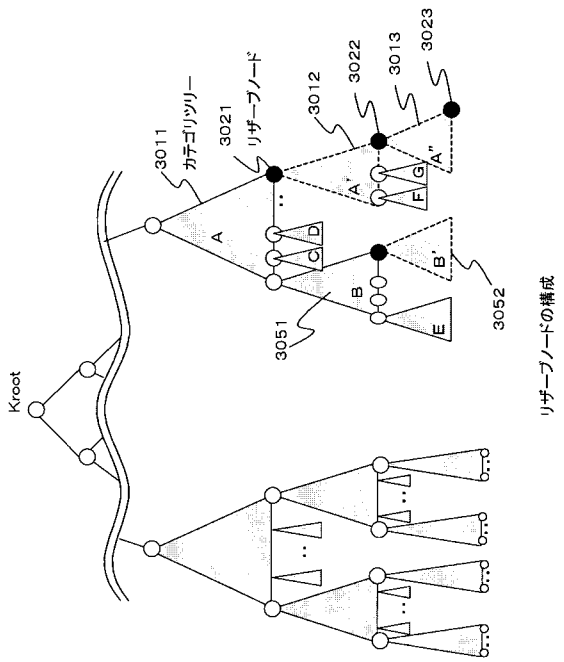
【図２９】



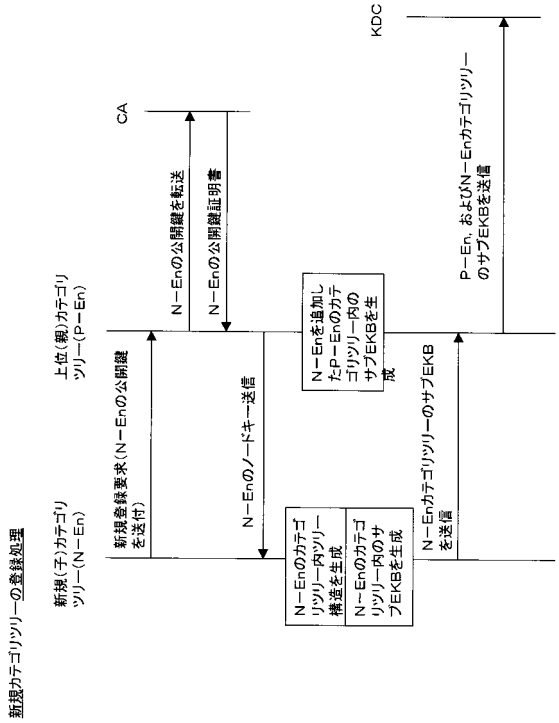
【図３０】



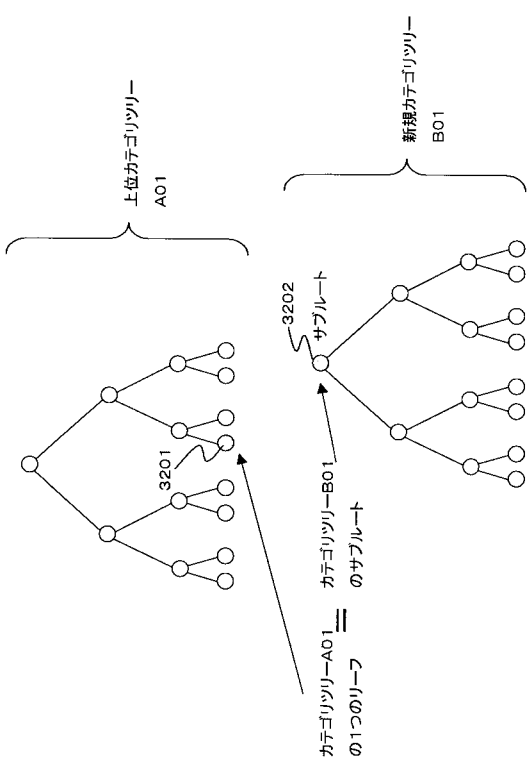
【図３１】



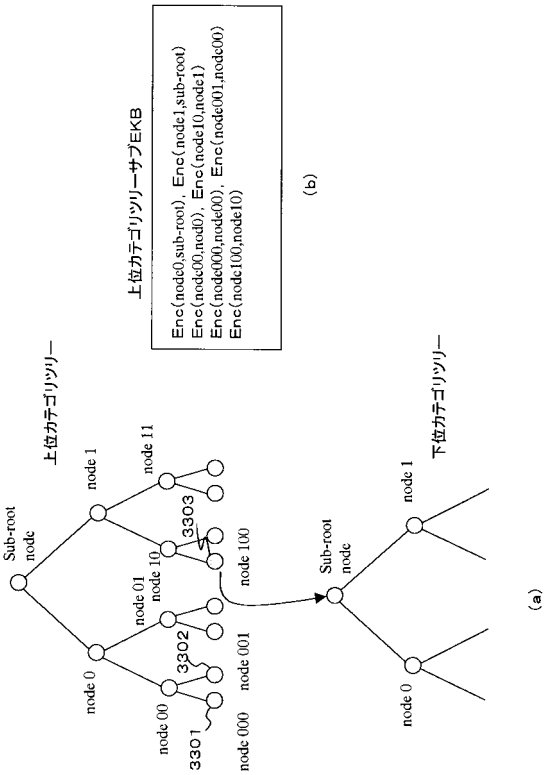
【図３２】



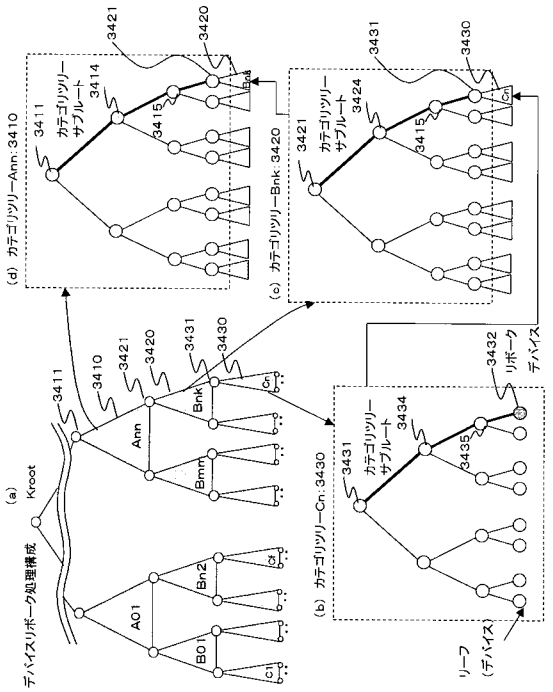
【図３３】



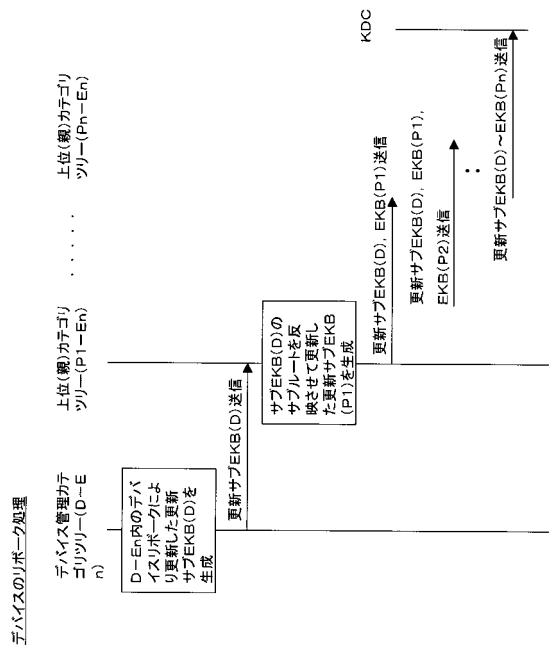
【図３４】



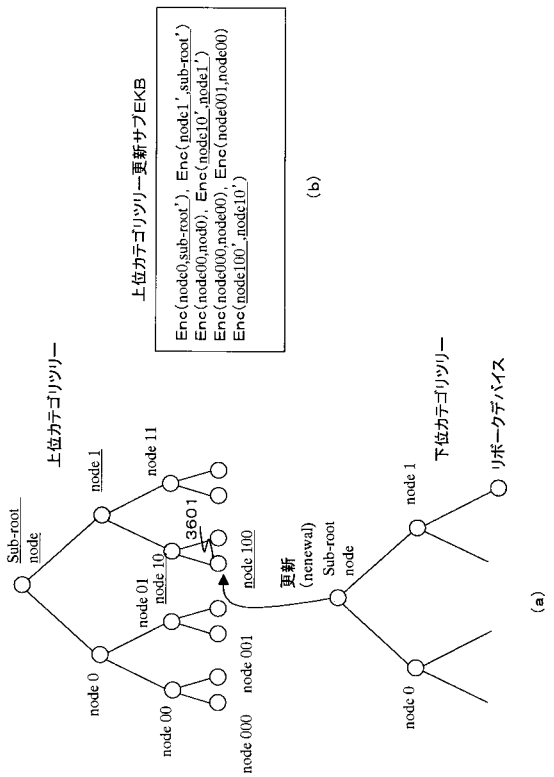
【図３５】



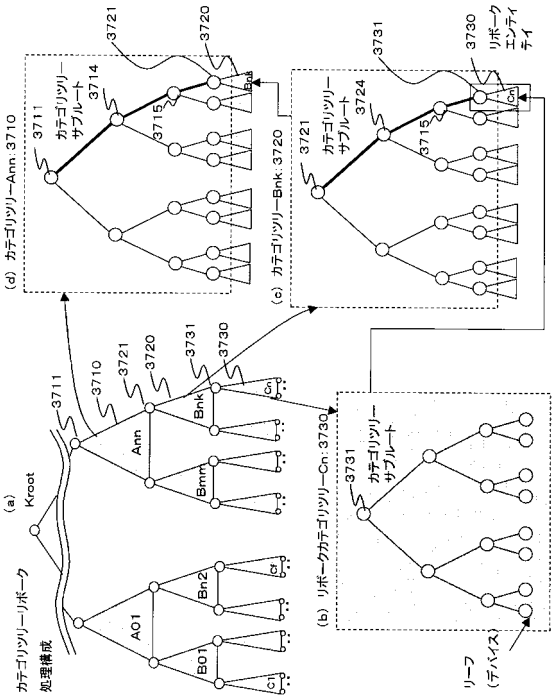
【図３６】



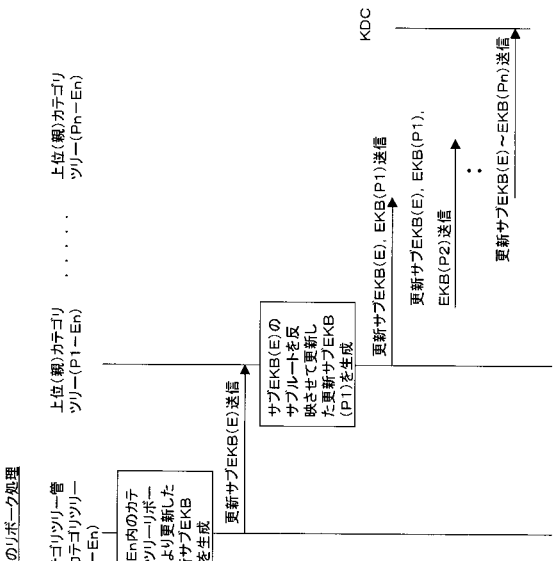
【図３７】



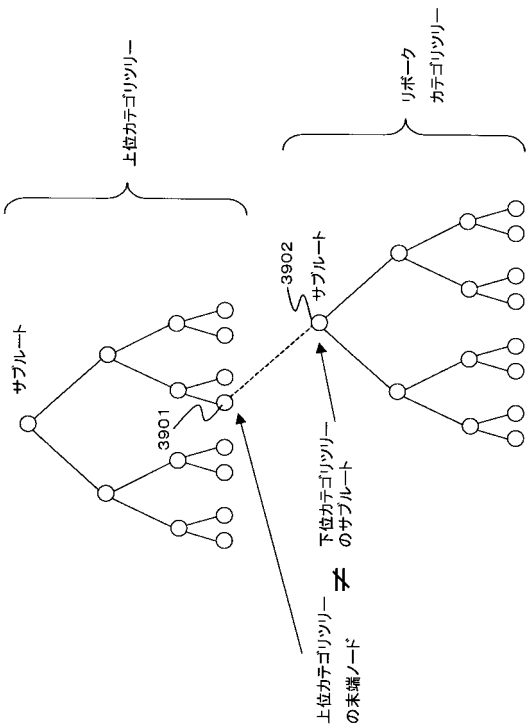
【図３８】



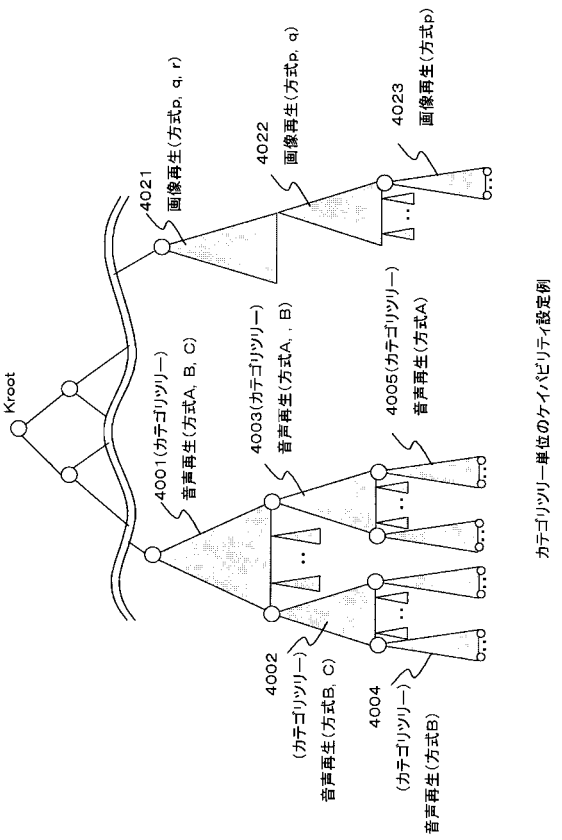
【図３９】



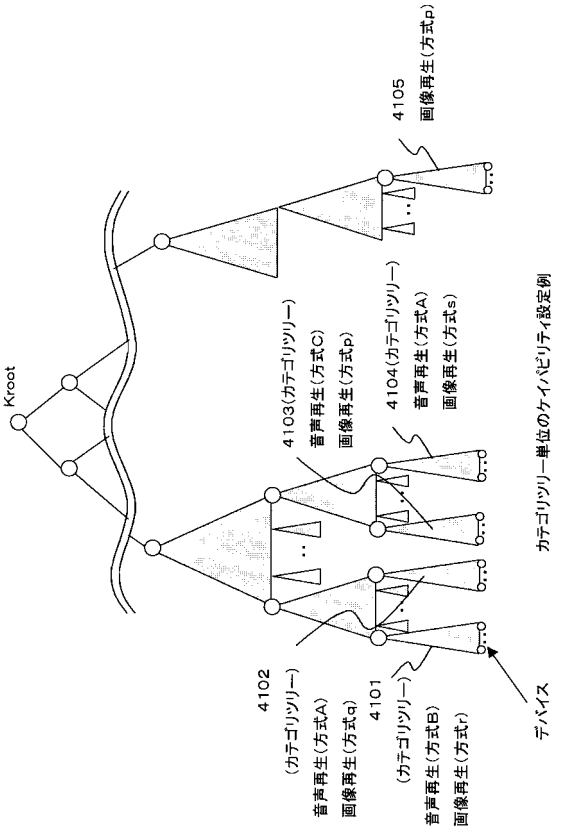
【図４０】



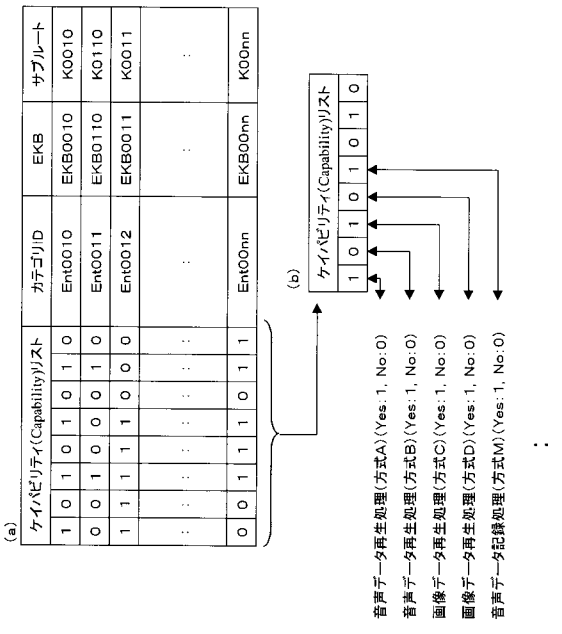
【図４１】



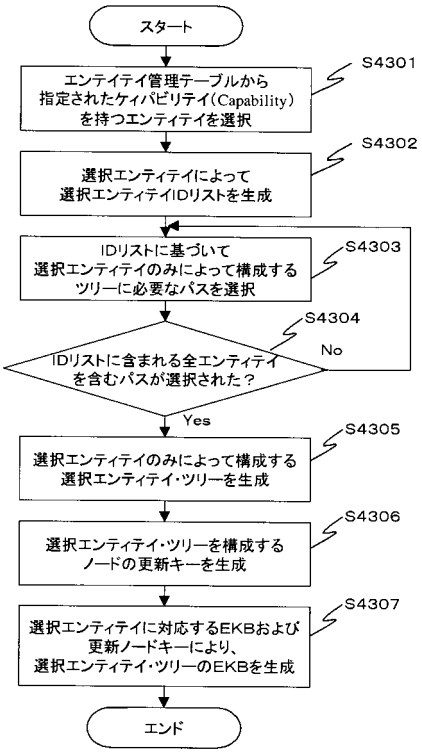
【図４２】



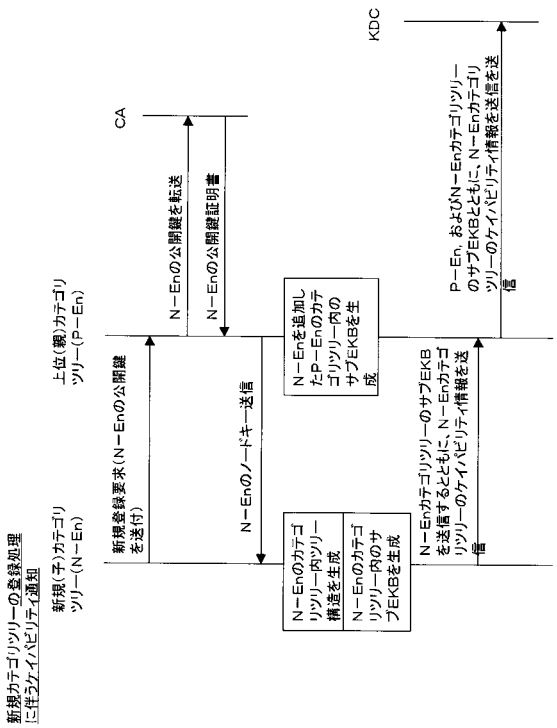
【図４３】



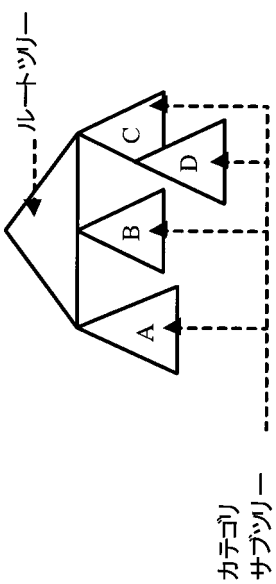
【図４４】



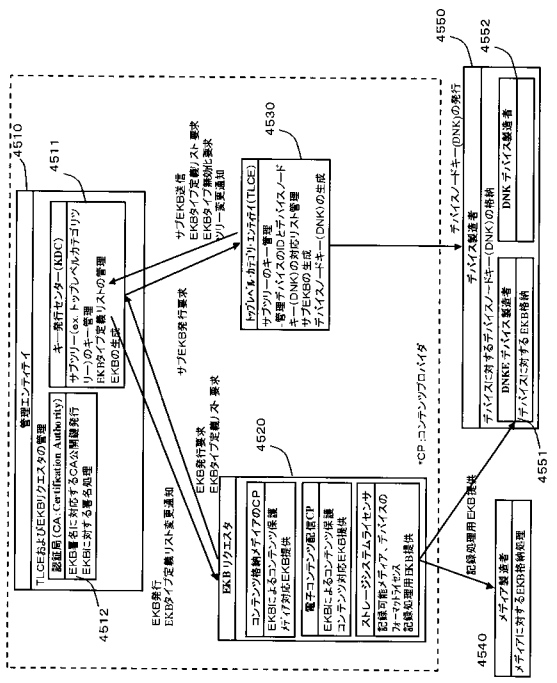
【図４５】



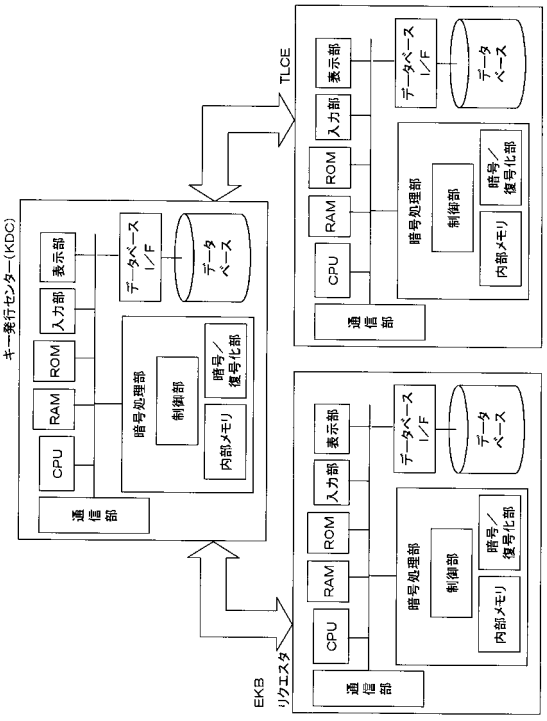
【図４６】



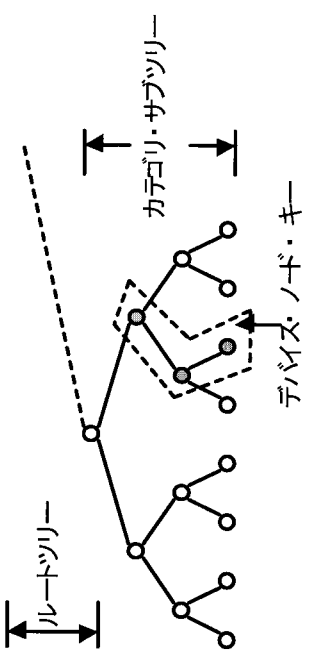
【図４７】



【図４８】



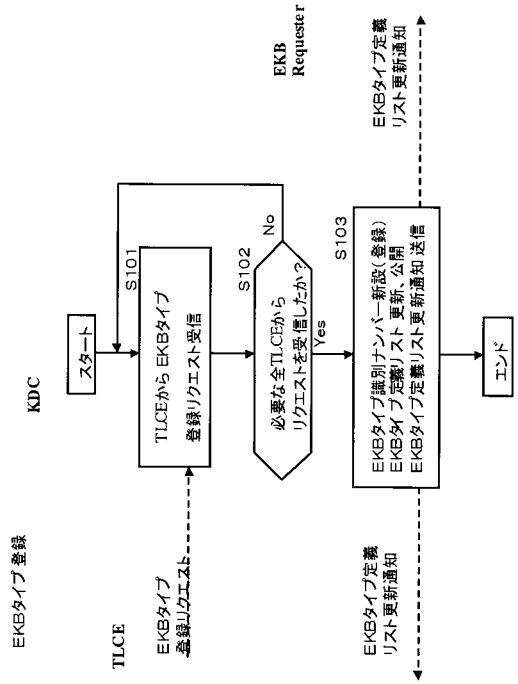
【図４９】



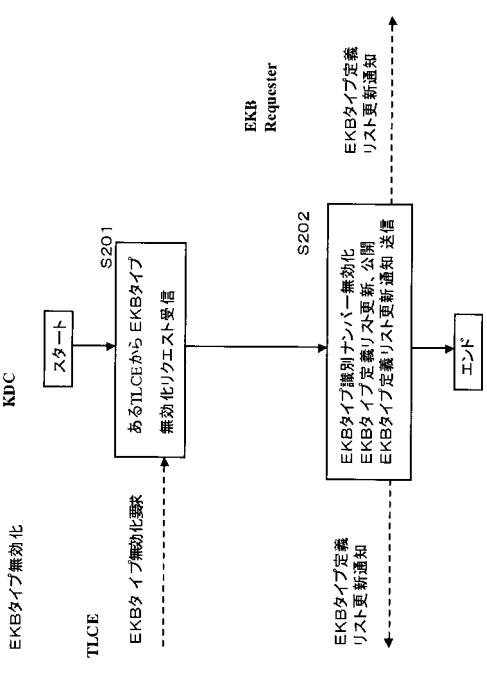
【図５０】

EKBタイプ識別ナンバー	ノード	説明
1	MS の Node ID	これは MemoryStick 用
2	PHS の Node ID	これは PHS 用
3	MS の Node ID と PHS の Node ID	これは MemoryStick + PHS 用
4	MD	これは MD 用
5	...	...

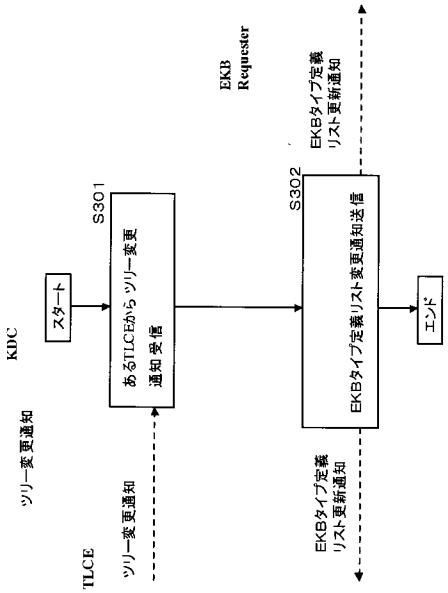
【図５１】



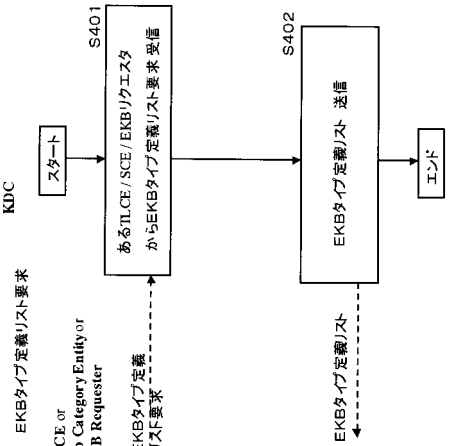
【図５２】



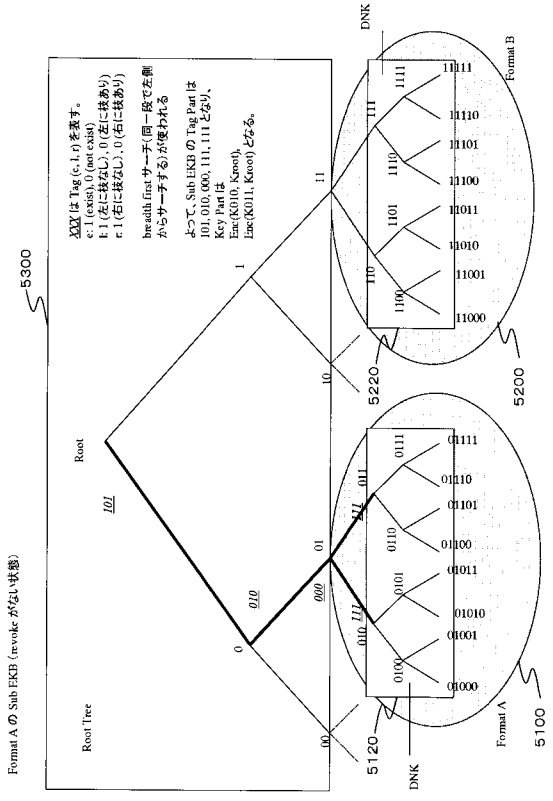
【図５３】



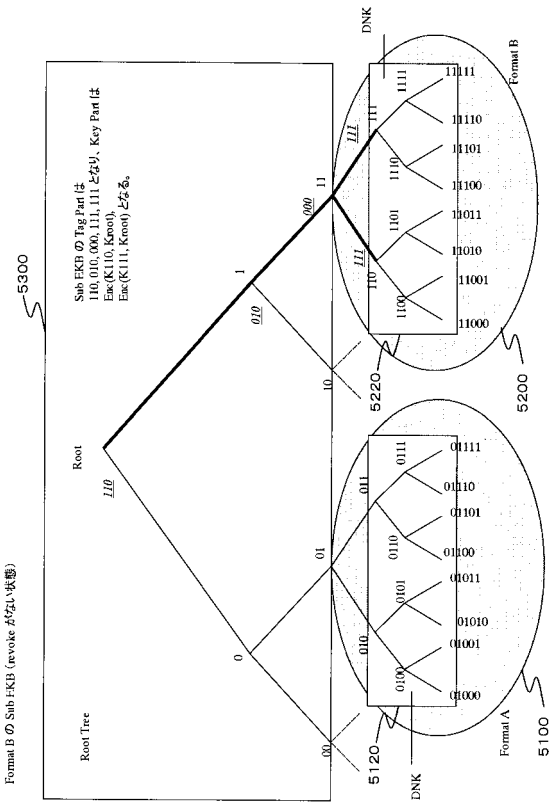
【図５４】



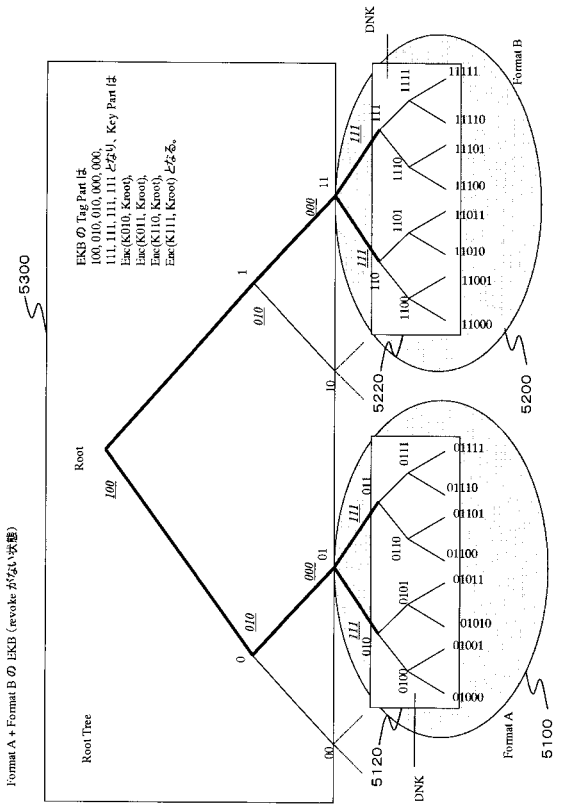
【図５５】



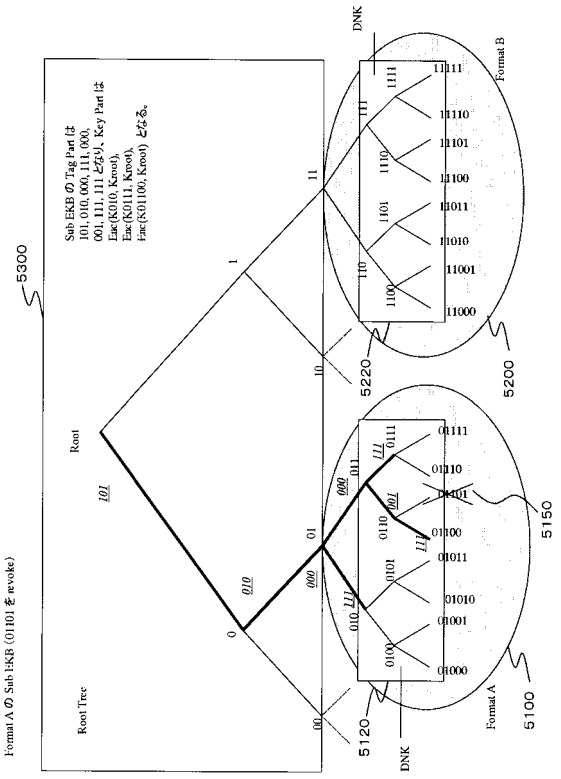
【図５６】



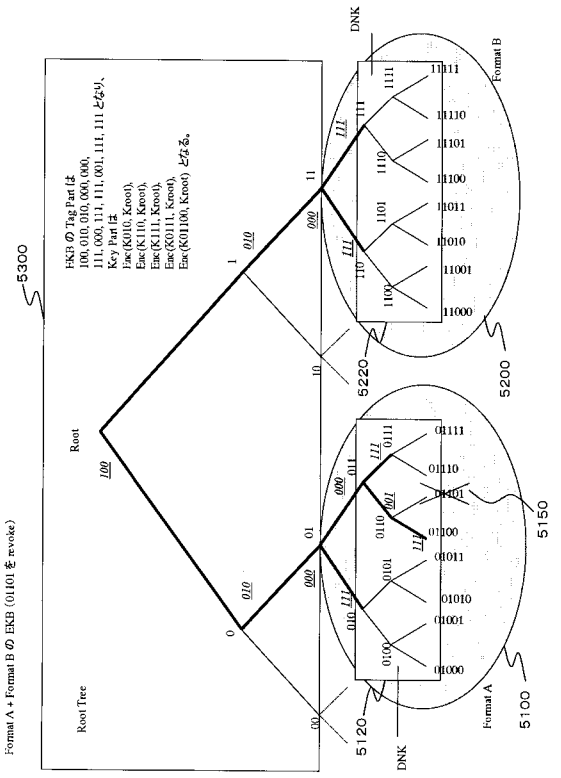
【図５７】



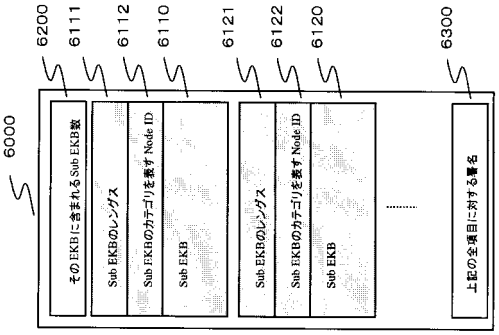
【図５８】



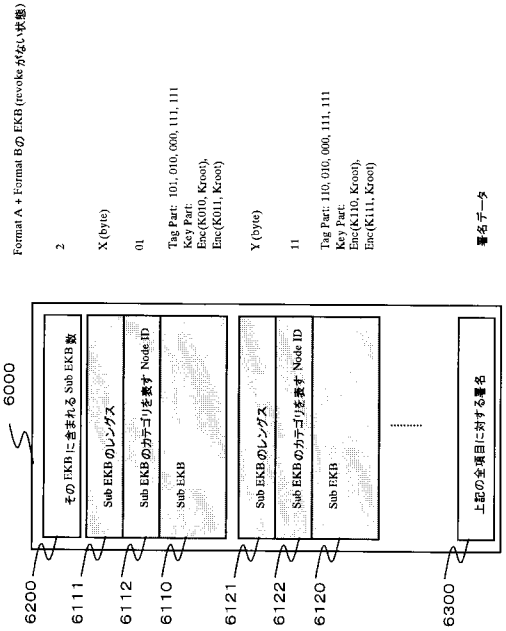
【図５９】



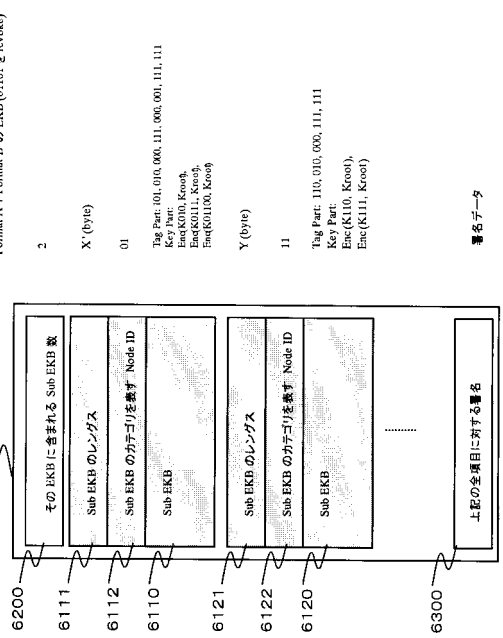
【図６０】



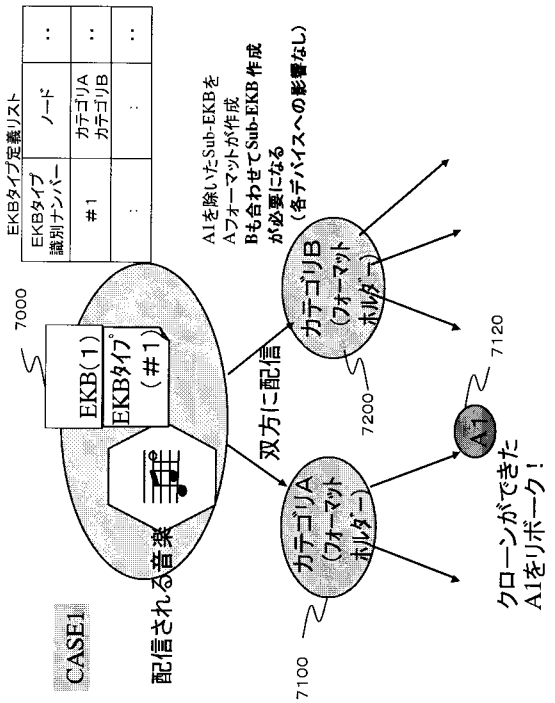
【図６１】



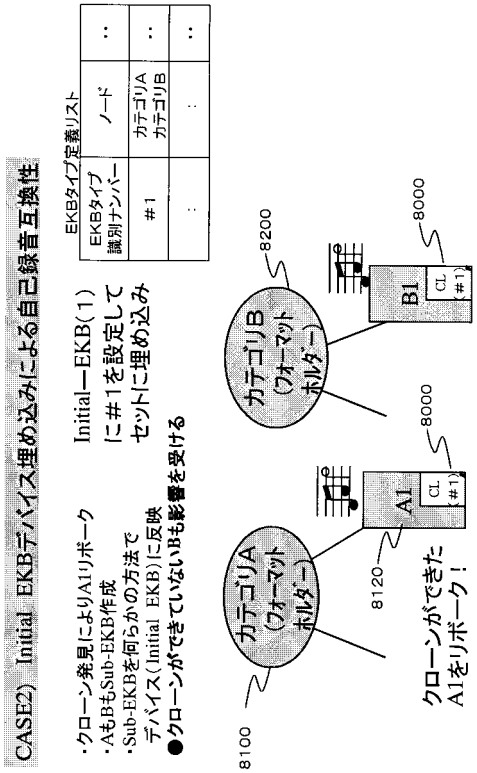
【図６２】



【図６３】



【図６４】



フロントページの続き

(72)発明者　大石　丈於
東京都品川区北品川６丁目７番３５号　ソニー株式会社内
(72)発明者　石黒　隆二
東京都品川区北品川６丁目７番３５号　ソニー株式会社内
(72)発明者　瀧　隆太
東京都品川区北品川６丁目７番３５号　ソニー株式会社内

審査官　中里　裕正

(56)参考文献　特開平１１－１８７０１３（ＪＰ，Ａ）
米国特許第０６０４９８７８（ＵＳ，Ａ）
Waldvogel, M. et al., The VersaKey Framework:Versatile Group Key Management, IEEE Journal on Selected Areas in Communications, １９９９年　９月, Vol.17 No.9, p.1614-1631

(58)調査した分野(Int.Cl., ＤＢ名)
H04L　9/08
G06F　21/24
JSTPlus/JMEDPlus/JST7580(JDreamII)