

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4740923号

(P4740923)

(45) 発行日 平成23年8月3日 (2011.8.3)

(24) 登録日 平成23年5月13日 (2011.5.13)

(51) Int.Cl.		F I			
HO 4 L	9/32	(2006.01)	HO 4 L	9/00	6 7 5 A
HO 4 N	7/173	(2011.01)	HO 4 N	7/173	6 3 O
GO 6 F	21/00	(2006.01)	GO 6 F	15/00	3 3 O Z
GO 6 F	21/24	(2006.01)	GO 6 F	12/14	5 6 O C

請求項の数 14 (全 19 頁)

(21) 出願番号	特願2007-284414 (P2007-284414)	(73) 特許権者	503447036
(22) 出願日	平成19年10月31日 (2007.10.31)		サムスン エレクトロニクス カンパニー
(62) 分割の表示	特願2005-501356 (P2005-501356)		リミテッド
原出願日	平成15年4月9日 (2003.4.9)		大韓民国キョンギード, スウォン-シ, ヨ
(65) 公開番号	特開2008-118653 (P2008-118653A)	(74) 代理人	100070150
(43) 公開日	平成20年5月22日 (2008.5.22)		弁理士 伊東 忠彦
審査請求日	平成19年11月5日 (2007.11.5)	(74) 代理人	100091214
(31) 優先権主張番号	60/418, 160		弁理士 大貫 進介
(32) 優先日	平成14年10月15日 (2002.10.15)	(74) 代理人	100107766
(33) 優先権主張国	米国 (US)		弁理士 伊東 忠重
(31) 優先権主張番号	60/425, 259		
(32) 優先日	平成14年11月12日 (2002.11.12)		
(33) 優先権主張国	米国 (US)		

最終頁に続く

(54) 【発明の名称】 メタデータの管理方法

(57) 【特許請求の範囲】

【請求項 1】

メタデータを所定の有意のセグメント単位に分割して生成された複数のフラグメントデータと、該複数のフラグメントデータの内の選択されたフラグメントデータのダイジェスト情報と、フラグメントデータのフォーマットのタイプを表すデータフォーマット情報とを受信して、前記メタデータを、クライアント装置で、管理するメタデータ管理方法において、

(a) 前記選択されたフラグメントデータと、該選択されたフラグメントデータのダイジェスト情報と、フラグメントデータのフォーマットのタイプを表すデータフォーマット情報とを受信する段階と、

(b) 受信された前記選択されたフラグメントデータと対応するデータフォーマット情報を使用して、ダイジェスト情報を生成するアルゴリズムにより、選択されたフラグメントデータのダイジェスト情報を生成する段階と、

(c) 前記 (b) 段階で生成されたダイジェスト情報と前記 (a) 段階で受信されたダイジェスト情報とを比較して、前記受信されたメタデータの認証を行う段階とを含むことを特徴とする方法。

【請求項 2】

前記選択されたフラグメントデータと、該選択されたフラグメントデータのダイジェスト情報と、フラグメントデータのフォーマット情報とは、一つのメタデータコンテナに含まれて受信されることを特徴とする請求項 1 に記載の方法。

【請求項 3】

前記データフォーマット情報は、ダイジェスト情報生成のために使用されたフラグメントデータが2進XMLフォーマットであるか、またはテキストXMLフォーマットであるかを表すことを特徴とする請求項1に記載の方法。

【請求項 4】

前記メタデータコンテナには、メタデータ認証レベルを特定する認証レベルフラグが含まれていることを特徴とする請求項2に記載の方法。

【請求項 5】

前記ダイジェスト情報は、前記選択されたフラグメントデータを一方向関数に入力して得られたダイジェスト情報であることを特徴とする請求項1に記載の方法。

10

【請求項 6】

前記一方向関数は、ハッシュ関数であることを特徴とする請求項5に記載の方法。

【請求項 7】

メタデータを所定の有意のセグメント単位に分割して生成された複数のフラグメントデータと、該複数のフラグメントデータの内の選択されたフラグメントデータのダイジェスト情報と、フラグメントデータのフォーマットのタイプを表すデータフォーマット情報と、前記選択されたフラグメントデータのダイジェスト情報とメタデータの認証に用いられる暗号化された第1暗号化キーにより生成されたメタデータを認証するメタデータ認証署名情報と、前記第1暗号化キーとを受信して、前記メタデータを、クライアント装置で、管理するメタデータ管理方法において、

20

(a) 前記選択されたフラグメントデータと、該選択されたフラグメントデータのダイジェスト情報と、フラグメントデータのフォーマットのタイプを表すデータフォーマット情報と、前記メタデータ認証署名情報と、暗号化された第1暗号化キーとを受信する段階と、

(b) 受信された前記選択されたフラグメントデータと対応するデータフォーマット情報を使用して、ダイジェスト情報を生成するアルゴリズムにより、選択されたフラグメントデータのダイジェスト情報を生成する段階と、

(c) 前記受信された暗号化された第1暗号化キーを前記クライアント装置に保存された第2暗号化キーを使用して復号化する段階と、

(d) 前記生成されたダイジェスト情報と前記復号化された第1暗号化キーを使用してメタデータ認証署名情報を生成する段階と、

30

(e) 前記生成されたメタデータ認証署名情報と前記受信されたメタデータ認証署名情報とを比較して、前記受信されたメタデータの認証を行う段階とを含むことを特徴とする方法。

【請求項 8】

前記ダイジェスト情報は、前記選択されたフラグメントデータを一方向関数に入力して得られたダイジェスト情報であることを特徴とする請求項7に記載の方法。

【請求項 9】

前記一方向関数は、ハッシュ関数であることを特徴とする請求項8に記載の方法。

【請求項 10】

前記生成されたメタデータ認証署名情報は、前記生成されたダイジェスト情報と復号化された第1暗号化キーとを一方向関数に入力して得られた結果値であることを特徴とする請求項7に記載の方法。

40

【請求項 11】

前記一方向関数は、ハッシュ関数であることを特徴とする請求項10に記載の方法。

【請求項 12】

前記(a)段階で受信されたフラグメントデータとそれに対応するダイジェスト情報、フラグメントデータのフォーマット情報、メタデータ認証署名情報、及び暗号化された第1暗号化キーは、一つのメタデータコンテナに含まれて受信されることを特徴とする請求項7に記載の管理方法。

50

【請求項 13】

前記データフォーマット情報は、メタデータ関連情報生成のために使用されたフラグメントデータが2進XMLフォーマットであるか、またはテキストXMLフォーマットであるかを表すことを特徴とする請求項7に記載の方法。

【請求項 14】

前記メタデータコンテナには、メタデータ認証レベルを特定する認証レベルフラグが含まれていることを特徴とする請求項12に記載の方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、メタデータ伝送サーバー及びクライアントでのメタデータ管理方法に係り、特に、メタデータが伝送サーバーから伝送された後、クライアント装置により受信されるまでのメタデータのメッセージソース、メッセージ剛健性、及び機密性の認証と関連したメタデータ管理方法に関する。

【背景技術】

【0002】

マルチメディアシステムで、マルチメディアコンテンツ及びそれと関連したメタデータは、サービス提供者からクライアント装置に提供される。そのようなマルチシステムの例は、データがサーバーからクライアントに伝送されるブロードキャスティングシステムまたはサーバー及びクライアントが相互作用するビデオ・オン・デマンド（VOD）のようなサービス方式である。

【0003】

伝送されるメタデータは、クライアント装置により多様な方式で使用される。メタデータの一つの使用例は、クライアント装置が再生、記録、伝送のような動作を行おうとするマルチメディアコンテンツを選択することである。

【0004】

一方、最近放送システムにおける、クライアント装置で使用されるメタデータに含まれる情報は次第に豊かになり、その保安の重要性も高くなっている。したがって、受信されたメタデータの場合、メタデータが生成された後に伝送サーバーから伝送されてクライアントにより受信されるまで、ソース認証と剛健性及び機密性が維持されたかについての認証の必要性が更に高くなっているが、そのようなメタデータの認証を効果的に行うためのメタデータ管理方法が存在していなかった。

【発明の開示】

【発明が解決しようとする課題】

【0005】

本発明は、前記のような必要性を満足するために、伝送されるメタデータの認証を効果的に行わせるためのメタデータ伝送サーバーでのメタデータ管理方法を提供することを目的とする。

【0006】

また、本発明は、受信されたメタデータの認証を効果的に行わせるクライアントでのメタデータ管理方法を提供することを目的とする。

【課題を解決するための手段】

【0007】

前記目的は、メタデータを所定の有意のセグメント単位に分割して生成された複数のフラグメントデータと、該複数のフラグメントデータの内の選択されたフラグメントデータのダイジェスト情報と、フラグメントデータのフォーマットのタイプを表すデータフォーマット情報とを受信して、前記メタデータを、クライアント装置で、管理するメタデータ管理方法において、

(a) 前記選択されたフラグメントデータと、該選択されたフラグメントデータのダイジェスト情報と、フラグメントデータのフォーマットのタイプを表すデータフォーマット

10

20

30

40

50

情報とを受信する段階と、

(b) 受信された前記選択されたフラグメントデータと対応するデータフォーマット情報を使用して、ダイジェスト情報を生成するアルゴリズムにより、選択されたフラグメントデータのダイジェスト情報を生成する段階と、

(c) 前記 (b) 段階で生成されたダイジェスト情報と前記 (a) 段階で受信されたダイジェスト情報とを比較して、前記受信されたメタデータの認証を行う段階とを含むことを特徴とする方法により達成される。

【0008】

また、前記目的は、メタデータを所定の有意のセグメント単位に分割して生成された複数のフラグメントデータと、該複数のフラグメントデータの内の選択されたフラグメントデータのダイジェスト情報と、フラグメントデータのフォーマットのタイプを表すデータフォーマット情報と、前記選択されたフラグメントデータのダイジェスト情報とメタデータの認証に用いられる暗号化された第1暗号化キーにより生成されたメタデータを認証するメタデータ認証署名情報（メタデータ認証署名情報と、前記第1暗号化キーとを受信して、前記メタデータを、クライアント装置で、管理するメタデータ管理方法において、

(a) 前記選択されたフラグメントデータと、該選択されたフラグメントデータのダイジェスト情報と、フラグメントデータのフォーマットのタイプを表すデータフォーマット情報と、前記メタデータ認証署名情報と、暗号化された第1暗号化キーとを受信する段階と、

(b) 受信された前記選択されたフラグメントデータと対応するデータフォーマット情報を使用して、ダイジェスト情報を生成するアルゴリズムにより、選択されたフラグメントデータのダイジェスト情報を生成する段階と、

(c) 前記受信された暗号化された第1暗号化キーを前記クライアント装置に保存された第2暗号化キーを使用して復号化する段階と、

(d) 前記生成されたダイジェスト情報と前記復号化された第1暗号化キーを使用してメタデータ認証署名情報を生成する段階と、

(e) 前記生成されたメタデータ認証署名情報と前記受信されたメタデータ認証署名情報とを比較して、前記受信されたメタデータの認証を行う段階とを含む管理方法により達成される。

【0009】

本発明に係るメタデータ認証方法は、メタデータがサーバーからクライアントに伝送される間に変更如何を確認し、該当メタデータがどのサービス提供者またはメタデータコンデンツ提供者から伝送されるかを効果的に認証するための伝送サーバー及びクライアント装置でのメタデータ管理方法に関する。

【発明を実施するための最良の形態】

【0010】

受信者がメタデータを受信する場合、受信されたメタデータの認証が行われねばならない。ここには、伝送レベルでの認証とソースレベルでの認証とに大別されうる。

【0011】

伝送レベルでの認証は、伝送レベルでのメッセージソース、メッセージ剛健性、及びメッセージ機密性についての認証を行う。

【0012】

伝送レベルでのメッセージソースの認証は、メッセージ、すなわち、メタデータコンデンツを生成したソースについての認証ではなく、メッセージを伝送するソースを認証することである。例えば、図1に示されたように、メタデータコンデンツ提供者120が存在し、SK Telecom Corp. のようなサービス提供者140が別途に存在する場合、クライアント160が受信するメタデータAについての伝送レベルでのメッセージソース認証は、サービス提供者140がメタデータAをクライアント160に伝送したということについての認証を意味する。

【0013】

また、伝送レベルでのメッセージ剛健性についての認証は、サービス提供者140からクライアント160に伝送されたメタデータAが伝送中にいかなる変更もなかったということを認証することである。

【0014】

また、伝送レベルでのメッセージ機密性についての認証は、サービス提供者140からクライアント160に伝送されたメタデータAが伝送中に公開されないということを認証することである。そのような伝送レベルでの認証は、TCP/IPプロトコルでのSSL/TLSアルゴリズム、IEEE 1394プロトコルでのDTPCアルゴリズム、及びDVIプロトコルでのHDCPアルゴリズムにより行われる。

10

【0015】

ソースレベルでの認証は、ソースレベルでのメッセージソース、メッセージ剛健性、及びメッセージ機密性についての認証を意味する。

【0016】

ソースレベルでのメッセージソース認証は、メッセージ、すなわち、メタデータコンデンツを生成したソースを認証することである。例えば、図1に示されたように、メタデータコンデンツ提供者120が存在し、SK telecom corp.のようなサービス提供者140が別途に存在する場合、クライアント160が受信するメタデータAについてのソースレベルでのメッセージソース認証は、メタデータコンデンツ提供者120が該当メタデータAをクライアント160に伝送したということについての認証を意味する。

20

【0017】

また、ソースレベルでのメッセージ剛健性についての認証は、メタデータコンデンツ提供者120からクライアント160に伝送されたメタデータAが伝送中にいかなる変更もなかったということを認証することである。

【0018】

また、ソースレベルでのメッセージ機密性についての認証は、メタデータコンデンツ提供者120からクライアント160に伝送されたメタデータBが伝送中に公開されないということを認証することである。

【0019】

したがって、ソースレベルでのメタデータ認証が行われる場合には、伝送レベルでのメタデータ認証が別途に行われる必要がない。

30

【0020】

図2(a)、(b)、及び(c)は、それぞれのレベルにおける物理階層でのメタデータの伝達方法を説明するための図である。

【0021】

図2(a)は、伝送レベルの認証時に使用される伝送パケットを示している。伝送レベルでの認証は、図2(a)に示されたそれぞれの伝送パケットに対して行われ、それらの伝送パケットは2進XMLタイプである。

【0022】

図2(c)は、ソースレベル認証時に使用されるメタデータを示す。図2(c)のメタデータはテキストXMLタイプである。

40

【0023】

図2(b)は、本発明の一実施例を説明するためのメタデータコンテナレベルの認証時に使用されるメタデータコンテナを示す。それらのメタデータコンテナには、メタデータの意味あるセグメント単位で挿入される。それらのメタデータコンテナの例は、図3及び図4に示されている。

【0024】

図3は、単方向チャンネルでのメタデータコンテナレベルの認証に使用されるメタデータコンテナのフォーマットを示す図である。図3に示されたように、メタデータコンテナ

50

はヘッダー、フラグメントデータ及びメタデータ認証情報などを含み、コンテナヘッダーには、メタデータコンテナレベルでの認証のための制御情報が含まれている。

【0025】

制御情報には、第1制御情報F__1、第2制御情報F__2、第3制御情報F__3、第4制御情報F__4、及び第5制御情報F__5があり、それらの制御情報は、一つの信号またはフラグで形成される。

【0026】

第1制御情報F__1は、該当コンテナが運搬するメタデータのフラグメントデータにメタデータコンテナレベルの認証のための方法が適用されたか否かを表す認証フラグである。メタデータコンテナ認証のための方法としては、メディアアクセス制御(media authentication code:以下、MAC)または電子署名アルゴリズム(digital signature algorithm:以下、DSA)などがある。

10

【0027】

第2制御情報F__2は、メタデータコンテナレベルでの認証情報を生成するために使用される特定アルゴリズムを表すための情報である。第2制御情報F__2は、一つの2進コード集合を使用して表現でき、そのような2進コードと特定アルゴリズムとの関係はあらかじめ定義されており、サービスを提供するサーバー及びメタデータコンテナを受信するクライアントにあらかじめ知られている。

【0028】

20

第3制御情報F__3は、F__2により特定されるアルゴリズムが、メタデータコンテナに共に含まれたフラグメントデータにどのような方式で具体的に使用されるかを表すためのデータフォーマット情報である。それは、メタデータコンテナレベル認証アルゴリズムが適用されたフラグメントデータは、テキスト形態から変換された2進XML形態でもあり、または元来のテキストXMLでもあるためである。

【0029】

第3制御情報F__3が必要である理由は、本発明に係るメタデータの認証情報の生成は、メタデータをハッシュ関数に入力して得た出力値、すなわち、ハッシュ値を利用して行われるため、テキストXMLデータの認証情報は、対応する2進XMLデータの認証署名情報と関連性がないためである。言い換えれば、クライアント装置で受信されたメタデータコンテナに含まれたメタデータとハッシュ値とから認証署名の有効如何を判断するには、ハッシュ値を計算するために使用されたメタデータのフォーマットを知っておらねばならないためである。

30

【0030】

第4制御情報F__4は、メタデータ認証と関連した暗号化キーの情報を意味する。暗号化キーの情報は、メタデータと共にメタデータコンテナに挿入されて伝送サーバーからクライアント装置に伝送されるか、または選択的に別途の保安チャンネルを通じて伝送サーバーからクライアント装置に伝送される。

【0031】

第5制御情報F__5は、適用された認証レベルを表示する認証レベルフラグであって、適用された認証レベルが伝送レベルであるか、またはソースレベルであることを表示する。例えば、F__5が‘0’に設定された場合には、認証レベルが伝送レベルであることを表示し、F__5が‘1’に設定された場合には、認証レベルがソースレベルであることを表示する。そのような認証レベルフラグを使用することによって、クライアント装置のアプリケーションは、伝送されたメタデータの認証レベルが伝送レベルであるか、またはソースレベルであるかによってメタデータの信頼程度を判断して伝送されたメタデータの使用如何を決定できる。

40

【0032】

また、メタデータコンテナはフラグメントデータの保存領域を含み、前記フラグメントデータ保存領域には、少なくとも一つ以上のフラグメントデータが挿入される。本実施例

50

に係るコンテナには、メタデータの各意味のセグメント単位、例えば、一つのプログラムについてのプログラム情報のようなフラグメントデータが挿入される。しかし、選択的に任意の単位のメタデータを運搬する場合にも適用されうる。また、関連されたメタデータは、一連のコンテナによりサービス提供者からクライアント装置に伝送される。また、一つのメタデータコンテナは、一つのメタデータフラグメントまたは複数のフラグメントを含む。例えば、一つのメタデータのフラグメントデータは、XMLツリー構造で表現された全体メタデータのうち、一つの副ツリーである。

【0033】

また、メタデータコンテナレベルの認証情報には、メタデータダイジェスト情報と認証署名情報とがある。

10

【0034】

メタデータダイジェスト情報は、フラグメントデータ保存領域に保存されたフラグメントデータのうち一つを、第2制御情報F__2により特定されたハッシュ関数のような一方向関数に挿入して得られた結果値を意味する。それぞれのメタデータダイジェスト情報は、ポインタを使用してそれぞれの対応フラグメントデータと関連している。例えば、第1メタデータダイジェスト情報は、ポインタにより第1フラグメントデータと関連している。本実施例では、メタデータダイジェスト情報を生成するためにハッシュ関数を使用した。が、選択的に、一方向関数の特性を有する所定の関数を使用してメタデータダイジェスト情報を求めることも可能である。

20

【0035】

認証署名情報は、ダイジェスト情報と暗号化キーKとが結合した値を、第2制御情報F__2により特定されるハッシュ関数のような一方向関数に挿入して得られた結果値を意味する。メタデータダイジェスト情報と同様に、それぞれのメタデータ認証署名情報は、ポインタを使用してそれぞれの対応フラグメントデータと関連している。例えば、第1認証署名情報は、ポインタにより第1フラグメントデータと関連している。本実施例では、認証署名情報を生成するためにハッシュ関数を使用した。が、選択的に、一方向関数の特性を有する所定の関数を使用して認証署名情報を求めることも可能である。

【0036】

図4は、双方向チャンネルでのメタデータコンテナレベルの認証に使用されるSOAPエンベロープのフォーマットを示す図である。図4に示されたように、認証関連情報はSOAPヘッダーに含まれており、メタデータフラグメントデータは本文に含まれている。

30

【0037】

認証関連情報のうち、'Algorithm ID'情報、'Signature Value BaseType'情報、及び'Key Info'情報は、図3の第2制御情報F__2、第3制御情報F__3、及び第4制御情報F__4に対応する。'Digest'情報及び'Signature Value'情報は、図3のメタデータダイジェスト情報及びメタデータ認証署名情報にそれぞれ対応する。'Authenticational Level'情報は、メタデータの認証レベルを特定するための情報であって、図3の第5制御情報F__5認証レベルフラグに対応する。

【0038】

図3及び図4に示されたように、メタデータを意味単位のセグメンテーション単位に分割したフラグメントデータをメタデータコンテナに挿入することにより、効率的な暗号化管理及びメタデータ管理が可能となる。

40

【0039】

例えば、それぞれのフラグメントデータ単位でインデクシング情報を付加することで、図5に示されたように、キャッシュ520に入力されたメタデータのうち、インデックスリスト保存部522に保存されたインデックスリストに基づいて選別されたメタデータのみを保存装置540に保存することが可能である。また、図4に示されたように、メタデータフラグメントデータがプログラム情報、セグメンテーション情報などの意味ある単位に分割されるため、それらそれぞれのフラグメントデータを選択的に暗号化することも可

50

能となる。

【0040】

図6は、図3及び図4に示されたメタデータコンテナを使用したメタデータコンテナレベル認証方法において、図1のメタデータコンデンス提供サーバー120またはサービス提供サーバー140で行われる手続きを説明するフローチャートである。

【0041】

段階610では、伝送されるメタデータを所定のセグメント単位に分割して複数のフラグメントデータを生成する。本実施例で生成されるフラグメントデータは、メタデータで、例えば、一つのプログラムについてのプログラム情報のような意味があるセグメント単位である。

10

【0042】

段階620では、生成されたフラグメントデータのうち、所定のフラグメントデータを選択する。

【0043】

段階630では、選択されたフラグメントデータをハッシュ関数、例えば、SHA-1のような保安ハッシュアルゴリズムに挿入して得られた結果値であるメタデータダイジェスト情報を生成する。本実施例では、メッセージダイジェスト情報を生成するためにハッシュ関数を使用した。しかし、選択的に、ハッシュ関数のような一方向関数の特性を有する他の関数を使用することも可能である。

【0044】

20

段階640では、選択されたフラグメントデータと生成されたメタデータダイジェスト情報及び選択されたフラグメントデータのフォーマットが2進XMLであるか、またはテキストXMLであるかを表すデータフォーマット情報を含むメタデータコンテナを生成した後、それをクライアントに伝送する。

【0045】

選択されたフラグメントデータタイプを表示する理由は、メタデータのフラグメントデータが同じである場合にも、段階620でのメタデータダイジェスト情報生成時に使用されるフラグメントデータのタイプによってメタデータダイジェスト情報が変わるためである。

【0046】

30

段階640で生成されるメタデータコンテナの例は、図3及び図4に示されている。また、選択的に、段階640では、生成されたメタデータコンテナの認証フラグを設定して、該当メタデータコンテナが運搬するメタデータのフラグメントデータにメタデータコンテナレベルの認証方法が適用されたことを表す。

【0047】

選択的に、メタデータコンテナに、メタデータダイジェスト情報生成のために使用されたアルゴリズム情報を挿入する。例えば、段階630でメタデータダイジェスト情報を生成するためにハッシュ関数を使用した場合、ハッシュ関数が認証情報生成アルゴリズムを使用したことを表すアルゴリズム情報を挿入する。しかし、アルゴリズム情報が伝送サーバーとクライアントとの間に知られている場合には、アルゴリズム情報はメタデータコンテナに挿入されない。

40

【0048】

また、フラグメントデータタイプの情報と共にメタデータ認証レベルを特定するフラグを挿入することも可能である。メタデータ認証レベルを特定するためのフラグは、メタデータコンテナを使用したメタデータ認証が伝送レベルで行われるか、またはソースレベルで行われるかを特定する。

【0049】

また、生成されるメタデータコンテナに挿入されるフラグメントデータが複数である場合、メタデータコンテナには、それらのそれぞれについて計算されたメタデータダイジェストが含まれ、それらのそれぞれのフラグメントデータとメタデータダイジェストとの関

50

連関係を表すポインタ情報が共に含まれる。

【0050】

また、生成されるメタデータコンテナに挿入されるフラグメントデータが複数である場合、メタデータコンテナには、それらのそれぞれのフラグメントデータについてのインデクシング情報が共に含まれる。

【0051】

図7は、図3及び図4に示されたメタデータコンテナを使用したメタデータコンテナレベル認証方法において、図1のメタデータクライアントサーバー160で行われる低続きを説明するフローチャートである。

【0052】

段階710では、メタデータコンデンツ提供サーバー120またはサービス提供サーバー140から伝送されたメタデータコンテナを受信する。

【0053】

段階720では、受信されたメタデータコンテナのヘッダーに含まれた第1制御情報F__1、すなわち、認証フラグを読み取る。

【0054】

段階730では、段階730での認証フラグを読み取った結果、該当メタデータコンテナに挿入されたフラグメントデータについて認証方法が適用されたと判断された場合には段階740に進行し、認証方法が適用されていないと判断される場合には段階742に進行する。

【0055】

段階740では、第2制御情報F__2、すなわち、認証情報生成のために使用されたアルゴリズムを認識して、メタデータコンテナに挿入されているメタデータダイジェスト情報を生成するために適用されたアルゴリズムを読み取る。本実施例で使用された認証情報生成アルゴリズムはハッシュ関数である。一方、認証情報生成アルゴリズムが伝送サーバーとクライアントとの間にあらかじめ特定された場合には、前記アルゴリズムの読み取り段階を省略する。

【0056】

また、第3制御情報F__3、すなわち、メタデータフォーマット情報を認識して、メタデータコンテナに挿入されたメタデータダイジェストの計算のために使用されたフラグメントデータのフォーマットを認識する。

【0057】

段階742では、メタデータコンテナレベルの認証手続きを終了する。

【0058】

段階750では、メタデータで、所定のフラグメントデータとそれに対応するメタデータ関連情報とを読み取る。

【0059】

段階760では、段階740で読み取られたフラグメントデータと、データフォーマット情報とに基づいて、段階740で認識されたアルゴリズム、例えば、ハッシュ関数を使用してメタデータダイジェスト情報を生成する。

【0060】

段階770では、段階760で生成されたメタデータダイジェスト情報と、段階750での所定のフラグメントデータに対応するメタデータダイジェスト情報とを比較して、伝送されたメタデータの認証の有効如何を決定する。

【0061】

また、選択的に、受信されたメタデータコンテナには、メタデータ認証レベルフラグが更に含まれており、クライアント装置のアプリケーションは、設定されたメタデータ認証レベルを読み取ることにより、メタデータの認証が伝送レベルに設定されたか、またはソースレベルに設定されたかが分かり、したがって、伝送されたメタデータの信頼程度を判断して、伝送されたメタデータの使用如何を決定することが可能となる。

10

20

30

40

50

【0062】

図8は、図3及び図4に示されたメタデータコンテナを使用したメタデータコンテナレベルの認証方法において、図1のメタデータコンデンス提供サーバー120またはサービス提供サーバー140で行われる手続きを説明するフローチャートである。

【0063】

段階810では、伝送されるメタデータを所定のセグメント単位に分割して複数のフラグメントデータを生成する。本実施例で生成されるフラグメント単位は、メタデータで、例えば、一つのプログラムについてのプログラム情報のような意味を有するセグメント単位である。

【0064】

段階820では、生成されたフラグメントデータのうち、所定のフラグメントデータを選択する。

【0065】

段階830では、選択されたフラグメントデータをハッシュ関数に挿入して得られた結果値であるメタデータダイジェスト情報を生成する。本実施例では、メッセージダイジェスト情報を生成するためにハッシュ関数を使用した。しかし、選択的に、ハッシュ関数のような一方向関数の特性を有する他の関数を使用することも可能である。

【0066】

段階840では、段階830で生成されたメタデータダイジェスト情報と暗号化キーKとをハッシュ関数に入力して、メタデータ認証署名を生成する。使用された暗号化キーKは、サービス提供者に特有のものである。本実施例では、メタデータ認証情報を生成するためにハッシュ関数を使用した。しかし、選択的に、ハッシュ関数のような一方向関数の特性を有する他の関数を使用することも可能である。メタデータ認証署名を生成するために使用された暗号化キーは、更に他の暗号化キーLを使用して暗号化される。ここで、暗号化キーLを使用して暗号化された暗号化キーK値をE(K)という。計算された暗号化キーE(K)は、メタデータコンテナに挿入されて運搬されるか、または別途のチャンネルを通じてクライアント装置に伝送される。また、暗号化キーLは、別途の保安チャンネルを使用してクライアント装置に伝送される。

【0067】

段階850では、選択されたフラグメントデータと対応するメタデータダイジェスト情報、メタデータ認証署名、及び選択されたフラグメントデータのフォーマット情報を含むメタデータコンテナを生成した後、それをクライアントに伝送する。

【0068】

段階850で生成されるメタデータコンテナの例は、図3及び図4に示されている。また、選択的に、段階850では、生成されたメタデータコンテナの認証フラグを設定して、該当メタデータコンテナが運搬するメタデータのフラグメントデータにメタデータコンテナレベルの認証方法が適用されたことを表す。

【0069】

選択的に、生成されたメタデータコンテナにメタデータダイジェスト情報生成のために使用されたアルゴリズム情報を挿入する。

また、選択されたフラグメントデータのフォーマット情報は、メタデータダイジェスト情報及び認証情報を生成するために使用されたフラグメントデータのフォーマットが、例えば、2進XMLであるか、またはテキストXMLであるかを表す。

【0070】

また、生成されるメタデータコンテナに挿入されるフラグメントデータが複数である場合、メタデータコンテナには、それらのそれぞれについて計算されたメタデータダイジェスト情報及び認証署名情報が含まれ、それらのそれぞれのフラグメントデータとメタデータダイジェスト情報及びメタデータ認証署名情報との関連関係を表すポインタ情報が共に含まれる。

【0071】

図9は、図3及び図4に示されたメタデータコンテナを使用したメタデータコンテナレベル認証方法において、図1のメタデータクライアントサーバー160で行われる手続きを説明するフローチャートである。

【0072】

段階910では、メタデータコンテンツ提供サーバー120またはサービス提供サーバー140から伝送されたメタデータコンテナを受信する。

【0073】

段階920では、受信されたメタデータコンテナのヘッダーに含まれた第1制御情報F__1、すなわち、認証フラグを読み取る。

【0074】

段階930では、段階920で認証フラグを読み取った結果、該当メタデータコンテナに挿入されたフラグメントデータについて認証方法が適用されたと判断された場合には段階940に進行し、認証方法が適用されていないと判断される場合には段階942に進行する。

【0075】

段階940では、第2制御情報F__2、すなわち、認証情報生成のために使用されたアルゴリズムを認識して、メタデータコンテナに挿入されているメタデータダイジェスト情報を生成するために適用されたアルゴリズムを読み取る。本実施例で使用された認証情報生成アルゴリズムはハッシュ関数である。認証情報を生成するためのアルゴリズムが伝送サーバーとクライアントとの間に予め特定されている場合には、前記適用アルゴリズムの読み取り段階を省略する。また、第3制御情報F__3、すなわち、フラグメントデータのフォーマット情報を認識して、メタデータコンテナに挿入されたメタデータダイジェスト情報の計算のために使用されたフラグメントデータのフォーマットを認識する。

【0076】

段階942では、メタデータコンテナレベルの認証手続きを終了する。

【0077】

段階950では、メタデータで、所定のフラグメントデータとそれに対応するメタデータ関連情報、メタデータ認証署名、及びデータフォーマット情報を読み取る。

【0078】

段階960では、読み取られたフラグメントデータとデータフォーマット情報とに基づいて、段階940で読み取られたアルゴリズム、例えば、ハッシュ関数を使用してメタデータダイジェスト情報を生成する。

【0079】

段階970では、クライアント装置に保存された暗号化キーLを使用して暗号化されたキーKを復号化する。暗号化キーLは、メタデータ伝送サーバーから別途の保安チャンネルを通じて伝送されたものである。

【0080】

段階980では、段階960で生成されたメタデータダイジェスト情報と復号化されたキーKとからメタデータ認証署名Sを生成する。

【0081】

段階990では、段階980で生成されたメタデータ認証署名情報と段階950で読み取られた認証署名情報とを比較して、伝送されたメタデータ認証署名の有効如何を決定する。

【0082】

また、選択的に、受信されたメタデータコンテナには、メタデータ認証レベルを表示する認証レベルフラグが更に含まれており、クライアント装置のアプリケーションはメタデータ認証レベルを読み取って、認証レベルのタイプによってメタデータ使用如何を決定する。

【0083】

また、別途の利用可能な剛健性の確認方法が存在する。そのような例のうち一つは、公

10

20

30

40

50

開キーを使用した暗号作成法である。その場合、サービス提供者は、秘密キー及び公開キーからなる一対のキー、すなわち、 K_s 、 K_p を保有しており、 K_s を使用してメッセージに署名する。ここで、 K_s は秘密キー、 K_p は公開キーを意味する。

【0084】

クライアント装置は、信頼すべきソースを通じて公開キーを獲得することが可能である。したがって、クライアントが署名を有するメタデータコンテナを受信する場合、クライアント装置は、受信されたメタデータコンテナが伝送されるサービス提供者を確認し、確認されたサービス提供者に対応する公開キー K_p を獲得する。クライアントは、受信された署名が有効であるか否かを確認するために公開キーを使用する。

【0085】

以下では、メタデータの保安を維持するためのメタデータ認証要件及び方法について更に具体的に説明する。

【0086】

メタデータについての適切な保安を維持するには、メタデータアクセス及び利用についての認証と、メタデータの剛健性及びメタデータの機密性の維持と、メタデータの部分集合及び2進フォーマット及びテキストフォーマットに対する効率的な保護が必要である。

【0087】

すなわち、アプリケーションによるメタデータまたはメタデータの一部についてのアクセス認証は、適切な認証規則に従わねばならない。そのような認証手続きは、アプリケーション単位またはアプリケーション及びメタデータ単位で行われる。

【0088】

また、メタデータの全体またはメタデータの一部のアクセスを通じる使用例には、視聴、変形、及び複写などがある。視聴は、アクセスを得ることとほぼ同じである最も簡単な使用例である。メタデータの変形またはローカル複写を制御するには、メタデータファイル管理システムを必要とする。また、遠隔アプリケーションでメタデータを複写すること、例えば、クライアントがメタデータをサービス提供者に伝送することは、メタデータ要請についての認証と、保安認証チャンネルを通じて要請されたデータ及びソース認証情報とを伝送することを必要とする。

【0089】

また、メタデータについての保安を維持するには、メタデータの機密性の維持が必要である。メタデータの引導及び保存中に、メタデータのうち、一部は高い価値またはプライバシーに関連したデータを含んでいるなどの多様な理由で暗号化される必要がある。そのために、伝送レベル、すなわち、伝送中の機密性は、メタデータの伝送ユニットまたはコンテナを暗号化することにより維持されうる。また、ソースレベルでの暗号化は、伝送及び保存レベルでの機密性と関連した問題を何れも解決する。

【0090】

以下では、条件的なアクセスシステムと関連した単方向環境及び双方向チャンネル(TLS)でのメタデータ保安について説明する。

【0091】

条件的なアクセスシステムと関連した単方向環境、すなわち、ブロードキャスト環境の例は、地上波放送(ATSC、DVB)、衛星放送(Direct TV)、ケーブルTV、及びIPマルチキャストなどがある。それらは、トランザクションのような情報の交換のために使用される別途の復帰チャンネルを除いては、単方向チャンネルである。下記の機能は、そのような環境で支援される。

【0092】

ハードウェア装置を備える加入された受信器と送信器との間の認証は自動的に行われる。また、受信器及び送信器は、メインブロードキャストチャンネルとは別途のチャンネルを利用してコモンシークレット共有する。ここで、コモンシークレットは、受信器と送信器のみが共有している別途のコードを意味する。パケットペイロードは、暗号化されて伝送された後、前記コモンシークレットを使用して解読するか、または前記コモンシークレ

10

20

30

40

50

ットを使用して解読されたキーを使用して解読する。

【0093】

双方向チャンネルの環境下では、ハンドシェイクプロトコルを使用し、サーバーまたはクライアントは、第3の証明書認証機関により発行された証明書を交換できるように認証される。コモンシークレットは、クライアントとサーバーとの間に共有され、その後、セッションキーが生成される。パケットペイロードは、セッションキーを使用して暗号化された後に伝送され、その後、同じセッションキーを使用して復号化される。ソース認証は、DSAまたはMACなどのようなアルゴリズムを通じて行われ得る。

【0094】

また、双方向チャンネルの環境下では、クライアント／サーバーの認証は、信頼できる第3の機関による証明書の認証及び交換を通じて行われ、受信されたデータの認証及び伝送間の機密維持は、パケットペイロードの暗号化及びメッセージ認証により行われる。

10

【0095】

そのように、メタデータ伝送と関連した保安を満足するには、送信器と受信器において相互間の認証が行われ、データ認証及びデータを暗号化して伝送することが可能であるように、コモンシークレットが安全に共有されねばならない。

【0096】

以下では、伝送レベル及びソースレベルでのメタデータについての保護方式について記述する。

【0097】

20

伝送中のメタデータ保護と関連して、送信器及び受信器の認証は伝送レベルで行われ、メタデータ認証及び機密維持は放送システムレベルで行われる。

【0098】

例えば、単方向チャンネルの場合には、それぞれのSOAP応答（ヘッダー＋本文）が保護単位として使用されうる。単方向である場合のそのような方法の例は、図10に示されている。また、双方向チャンネルの場合には、データ署名情報は、SOAP応答を使用して伝送されうる。その場合、図11に示されたように、署名情報はSOAPヘッダーに含まれており、データ情報は本文に含まれている。本文のデータ部分は暗号化されうる。

【0099】

また、ソースレベルでのメタデータ保護と関連して、下記では放送装置内でのメタデータの剛健性及び機密維持及びメタデータアクセス及び使用制御について記述する。

30

【0100】

放送装置でのメタデータの剛健性及び機密維持は、認証署名をメタデータに関連付け、それを暗号化することにより可能である。また、メタデータのあらゆる部分が暗号化されるか、または剛健性が維持される必要がないということを考慮して、ポインタを使用して暗号化または認証手続きが行われたメタデータの特定部分を表す必要がある。そのような動作は、権利管理保護（right management protection：RMP）システムによりそのようなポインタが維持されるソースレベルで行われ得る。ソースレベル署名を使用することにより、メタデータソースは実質的に認証されうる。もちろん、メタデータは、そのような情報、すなわち、ソースの認証署名をあらかじめ含んでお

40

【0101】

また、メタデータのアクセス及び使用制御のためには、標準アクセス及び使用権利技術及びそれについての強制が必要である。それと関連した標準技術は、XMLスキーマの一つの形態を有するか、または意味上で明確な意味を有する一つの集合のデータ要素の形態を有しうる。そのような技術の構成に関連しうる既存の道具は、XrML、XACML、及びSAMLなどがある。ライセンス技術及び利用規則はメタデータから分離されうる。

【0102】

また、選択的に、メタデータの使用が記述されうる部分的なメタデータの数が潜在的に多いということを考慮して、アクセス／使用制御を更に単純化するために、一旦、一つの

50

アプリケーションのアクセスが認証される場合、該当アプリケーションの動作は、デフォルトとして使用規則を従うと見なすことも可能である。

【0103】

また、そのような問題と関連したものは、アクセス／使用と関連したRMPシステムでのアプリケーションプログラムインターフェース（API）である。それは、アクセス／使用制御情報がTVA RMPシステムによって管理される場合に必要である。例えば、APIはアクセスを要請及び許可し、変形、複写、及び外部に送る動作を行う。

【0104】

前記で説明したように、構造的なレベルで認証が行われ得る数種の認証がある。

【0105】

第一は、伝送レベルで行われるものであり、第二は、単方向チャンネルでのコンテナまたはSOAPメッセージのような双方向チャンネルで行われるものであり、第三は、ソースレベルで行われるものである。

【0106】

ソースレベルでの認証は、認証が行われるメタデータの具体的な部分についてポインタを使用して認証情報を提供する。SOAPメッセージレベル認証の場合、認証情報は、SOAPメッセージの本文に含まれたメタデータの一部または全部についてのポインタと共にヘッダーに含まれる。

【0107】

また、伝送中に単に剛健性保証のみが要求される場合、伝送レベルでの認証のみで十分である。一方、伝送独立性が必要である場合、コンテナレベルまたはSOAPメッセージレベル認証がそのような要求条件を満たし得る。コンテナまたはSOAPメッセージの本文に含まれるメタデータのサイズは、伝送パケットのサイズよりはるかに大きいため、そのような伝送レベルの認証は、システムにあたえる負荷を減らす。もちろん、保安チャンネルは、そのような場合に維持される必要がない。

【0108】

メタデータソースの認証のためには、コンテナ及びSOAPレベル認証が必要である。ソース認証を可能にするコンテナのシンタックスは、図11に示された通りである。

【0109】

ソースと最終受信地との間の多くの中間ノードにおいても、ソース認証が行われるようにするには、各中間ノードでソース認証が保存されねばならない。

【0110】

更に具体的に、それぞれの中間ノードで受信されたメタデータは、以前のノードから受信された認証情報を使用して認証され、新たな認証情報が生成されて次のノードに伝達されるか、または以前のノードから伝達されたメタデータ及び全ての認証情報が次のノードに伝達される。

【0111】

したがって、幾つかの中間ノードを含むソースレベル認証を使用したメタデータの伝達の場合、一つのノードで以前のノードの認証情報を使用してソースレベルでの認証以後、新たな認証情報が生成されるか否かを表すフラグまたは信号が認証情報に挿入されうる。そのフラグを使用することにより、受信器は、ソース認証情報の有無によって該当メタデータを受信するか否かを決定する。

【0112】

本発明は、前記の実施例に限定されず、本発明の思想の範囲内で当業者による変形が可能である。

【0113】

本発明は、またコンピュータ可読記録媒体にコンピュータ可読コードとして具現することが可能である。コンピュータ可読記録媒体は、コンピュータシステムによって読み取られうるデータが保存されるあらゆる種類の記録装置を含む。コンピュータ可読記録媒体の例としては、ROM、RAM、CD-ROM、磁気テープ、ハードディスク、フロッピー

10

20

30

40

50

(登録商標) ディスク、フラッシュメモリ、光データ保存装置などがあり、また、キャリアウェーブ（例えば、インターネットによる伝送）の形態で具現されるものも含む。また、コンピュータ可読記録媒体は、ネットワークに連結されたコンピュータシステムに分散されて、分散方式でコンピュータ可読コードとして保存されて実行されうる。

【産業上の利用可能性】

【0114】

本発明に係るメタデータ管理方法によれば、メタデータコンテナレベルでメタデータの認証を行わせることで、チャンネル環境に関係なく伝送レベルでの認証が可能であり、また、コンテナに認証のために計算されるメタデータのフォーマットを表す情報を挿入して、伝送レベル及びソースレベルでの認証が何れも選択的に可能であるようにして、伝送レベルでのパケットに比べて、メタデータコンテナレベルのパケットのサイズが比較的に大きい

10

ため、伝送されるパケットの数を減らしてシステム複雑度を減少させることが可能であるという効果がある。

【図面の簡単な説明】

【0115】

【図1】メタデータ認証レベルを説明するためのブロック図である。

【図2】物理階層でのメタデータの伝達方法を説明するための図である。

【図3】単方向チャンネルでのメタデータコンテナレベル認証に使用されるメタデータコンテナのフォーマットを示す図である。

【図4】双方向チャンネルでのメタデータコンテナレベル認証に使用されるSOAPメッセージを示す図である。

20

【図5】メタデータのインデクシング情報を利用したメタデータの分類方法を説明するための図である。

【図6】本発明の一実施例に係るメタデータ伝送サーバーでのメタデータ管理方法を説明するためのフローチャートである。

【図7】本発明の一実施例に係るメタデータクライアントでのメタデータ管理方法を説明するためのフローチャートである。

【図8】本発明の一実施例に係るメタデータ伝送サーバーでのメタデータ管理方法を説明するためのフローチャートである。

【図9】本発明の一実施例に係るメタデータクライアントでのメタデータ管理方法を説明するためのフローチャートである。

30

【図10】単方向チャンネルの場合のデータコンテナフォーマットを示す図である。

【図11】双方向チャンネルの場合のSOAPメッセージを示す図である。

【符号の説明】

【0116】

120 メタデータコンデンツ提供者

140 サービス提供者

160 クライアント

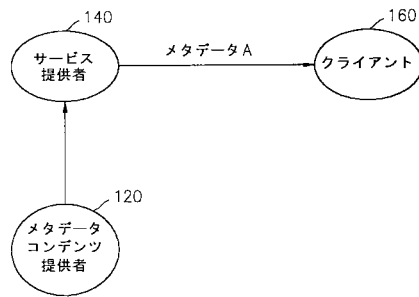
520 キャッシュ

522 インデックスリスト保存部

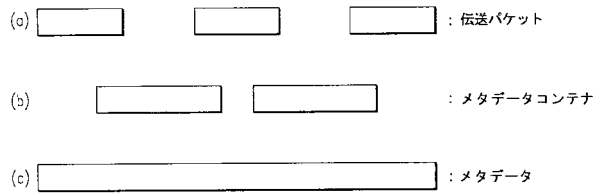
540 保存装置

40

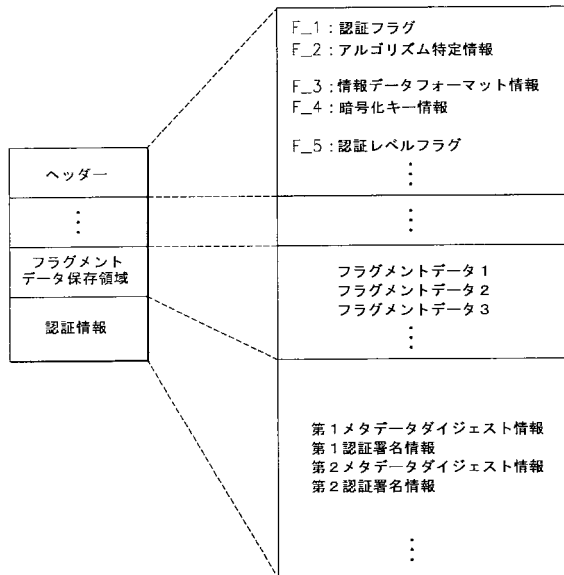
【図 1】



【図 2】



【図 3】

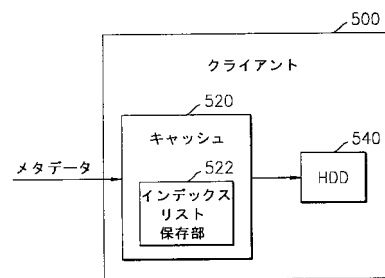


【図 4】

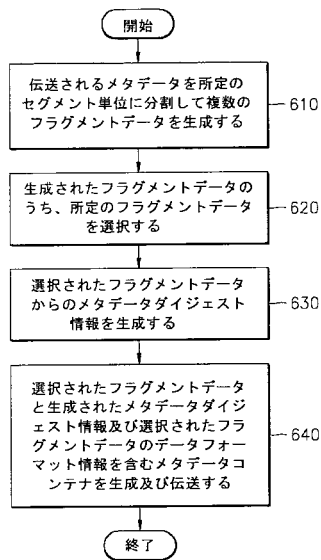
```

<SOAP:Envelope ...>
  <SOAP:Header>
    <Signature fragrefID = 1>
      <Algorithm ID=1>
      <Digest> ... </Digest>
      <SignatureValue> ... </SignatureValue>
      <KeyInfo> ... </KeyInfo>
      <SignatureValueBaseType>Text</SignatureValueBaseType>
      <AuthenticationLevel>Transport</AuthenticationLevel>
    </Signature>
  </SOAP:Header>
  <SOAP:Body>
    <TVAmetadataFragment id=1>
      <ProgramInformation>...</ProgramInformation>
    </TVAmetadataFragment>
    <TVAmetadataFragment id=2>
      <SegmentInformation>...</SegmentInformation>
    </TVAmetadataFragment>
  </SOAP:Body>
</SOAP:Envelope>
    
```

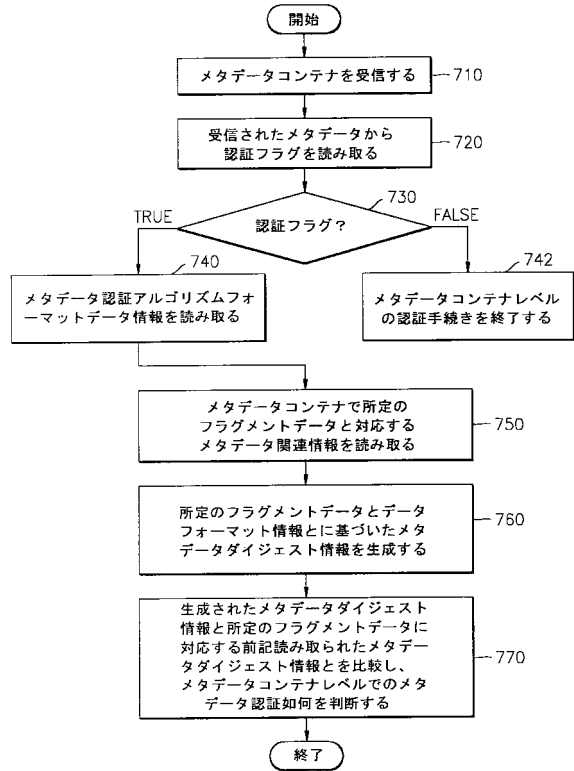
【図 5】



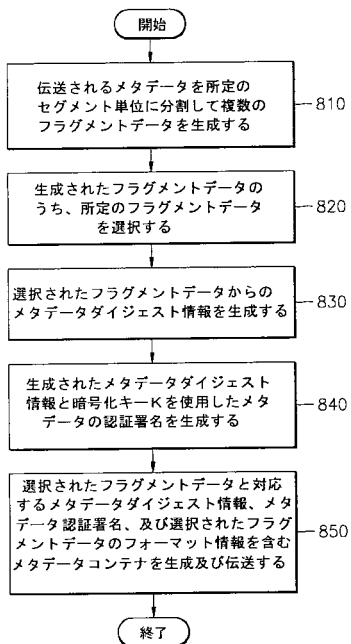
【図 6】



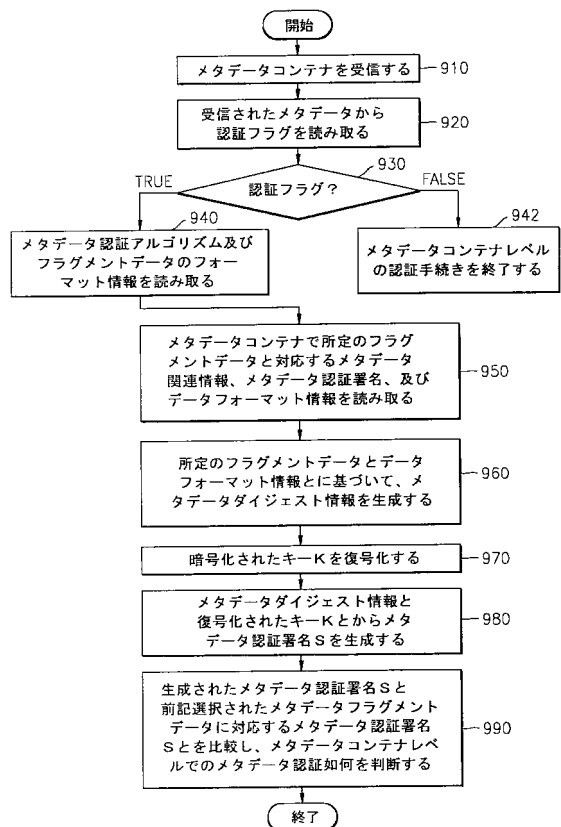
【図 7】



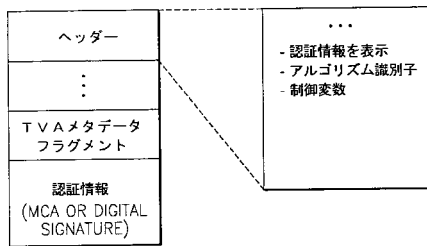
【図 8】



【図 9】



【図 10】



【図 11】

```

<?xml version="1.0" encoding="utf-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP:Header>
    <wssec:credentials xmlns:wssec="http://schemas.xmlsoap.org/ws/2001/10/security">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#" id="SigningCertificate">
        <ds:X509Data>
          <ds:X509Certificate>MIH1zCCBr+gAwIBA...</ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
    </wssec:credentials>
    <wssec:integrity xmlns:wssec="http://schemas.xmlsoap.org/ws/2001/10/security">
      <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
          <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/xml-exc-c14n#" />
          <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
          <ds:Reference URI="/">
            <ds:Transform>
              <ds:Transform Algorithm="http://schemas.xmlsoap.org/2001/10/security
                #RoutingSignatureTransform" />
              <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
            </ds:Transform>
            <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
            <ds:DigestValue>j6lnJnEPDvKIMup4NbeVu8nk=</ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>
        <ds:SignatureValue>ajYECaKqK2PvQaRweWajXup5zwa...</ds:SignatureValue>
      </ds:Signature>
    </wssec:integrity>
  </SOAP:Header>
  <SOAP:Body>
    <TVAmetaDataFragment>
      <ProgramInformation>...</ProgramInformation>
      <enc:EncryptedData>
        <enc:EncryptionMethod Algorithm="xxx_algorithm" />
        <ds:KeyInfo>...</ds:KeyInfo>
        <enc:CipherData>
          <enc:CipherValue>Bazy6TiW2+HcSHtHg...</enc:CipherValue>
        </enc:CipherData>
      </enc:EncryptedData>
      <enc:EncryptedKey>
        <enc:EncryptionMethod Algorithm="yyy_algorithm" />
        <ds:KeyInfo>
          <ds:KeyName>Public/Private Key for TVA metadata</ds:KeyName>
          <enc:CipherData>
            <enc:CipherValue>CCBPowCwYDVRPBA...</enc:CipherValue>
          </enc:CipherData>
        </enc:EncryptedKey>
      </TVAmetaDataFragment>
    </SOAP:Body>
  </SOAP:Envelope>

```

フロントページの続き

(31)優先権主張番号 10-2003-0013002

(32)優先日 平成15年3月3日(2003.3.3)

(33)優先権主張国 韓国(KR)

(72)発明者 チェー, ヤン-リム

大韓民国 463-060 キョンギード ソンナム-シ ブンダン-グ イメードン 124
ハンシン・アパート 210-1509

審査官 石田 信行

(56)参考文献 特開2002-063147 (JP, A)
特開2002-057662 (JP, A)
再公表特許第2002/023903 (JP, A1)
特開2003-051816 (JP, A)
特開2003-234737 (JP, A)
特開2003-248737 (JP, A)
特開2004-072184 (JP, A)
特開2004-102951 (JP, A)
特開2005-285089 (JP, A)

(58)調査した分野(Int.Cl., DB名)

H04L 9/32
G09C 1/00
G06F 21/00
G06F 21/24
H04N 7/173