(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第4748762号
(P4748762)

(45)発行日　平成23年8月17日(2011.8.17)　(24)登録日　平成23年5月27日(2011.5.27)

(51)Int.Cl.		F I		
G06Q 50/00	(2006.01)	G06F	17/60	142
G06Q 30/00	(2006.01)	G06F	17/60	302E
G06Q 10/00	(2006.01)	G06F	17/60	512
H04L 9/32	(2006.01)	H04L	9/00	675B

請求項の数 5　(全 14 頁)

(21)出願番号　特願2004-244132(P2004-244132)
(22)出願日　平成16年8月24日(2004.8.24)
(65)公開番号　特開2006-65408(P2006-65408A)
(43)公開日　平成18年3月9日(2006.3.9)
審査請求日　平成19年8月22日(2007.8.22)

(73)特許権者　000001007
キヤノン株式会社
東京都大田区下丸子３丁目３０番２号
(74)代理人　100076428
弁理士　大塚　康徳
(74)代理人　100112508
弁理士　高柳　司郎
(74)代理人　100115071
弁理士　大塚　康弘
(74)代理人　100116894
弁理士　木村　秀二
(74)代理人　100130409
弁理士　下山　治
(74)代理人　100134175
弁理士　永川　行光

最終頁に続く

(54)【発明の名称】署名生成方法及び情報処理装置

(57)【特許請求の範囲】

【請求項１】

情報処理装置が行う署名生成方法であって、

前記情報処理装置が有する入力手段が、コンテンツと、該コンテンツごとに動的に変化する数値化データを含む利用制御情報とを入力する入力工程と、

前記情報処理装置が有する抽出手段が、前記入力工程において入力された利用制御情報に含まれる数値化データの上限値及び下限値を抽出する抽出工程と、

前記情報処理装置が有する被署名データ生成手段が、前記利用制御情報に含まれる数値化データを前記抽出工程において抽出された上限値及び下限値に設定して被署名データを生成する被署名データ生成工程と、

前記情報処理装置が有する署名生成手段が、前記被署名データ生成工程において生成された前記被署名データと前記コンテンツとを連結し、署名を施して署名データを生成する署名生成工程と、

前記情報処理装置が有する出力手段が、前記入力工程において入力されたコンテンツと、前記入力工程において入力された利用制御情報と、前記署名生成工程において生成された署名データとを出力する出力工程と、

を有することを特徴とする署名生成方法。

【請求項２】

前記利用制御情報は、拡張可能マークアップ言語（ＸＭＬ）で記述されていることを特徴とする請求項１に記載の署名生成方法。

【請求項３】
コンテンツと、該コンテンツごとに動的に変化する数値化データを含む利用制御情報とを入力する入力手段と、
前記入力手段により入力された利用制御情報に含まれる数値化データの上限値及び下限値を抽出する抽出手段と、
前記利用制御情報に含まれる数値化データを前記抽出工程において抽出された上限値及び下限値に設定して被署名データを生成する被署名データ生成手段と、
前記被署名データ生成手段により生成された前記被署名データと前記コンテンツとを連結し、署名を施して署名データを生成する署名生成手段と、
前記入力手段により入力されたコンテンツと、前記入力手段により入力された利用制御情報と、前記署名生成手段により生成された署名データとを出力する出力手段と、
を有することを特徴とする情報処理装置。
【請求項４】
コンピュータを、請求項３に記載の情報処理装置が有する各手段として機能させるためのコンピュータプログラム。
【請求項５】
請求項４に記載のコンピュータプログラムを記録したコンピュータ読み取り可能な記録媒体。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、情報処理装置が行う署名生成方法及び情報処理装置に関するものである。
【背景技術】
【０００２】
近年、高速な通信回線やＤＶＤなどの大容量記録メディアを通じて、文書や画像データなどのデジタルコンテンツを流通する機会が増加している。中でも、デジタルコンテンツ配信サービスは、特定のユーザに対してコンテンツを流布するサービスであるが、ユーザ以外にはコンテンツが漏洩しない仕組みが要求されている。また、大容量メディアによるコンテンツ配信においても、同様のユーザによるアクセス制御の機構が検討されている。その際には、コンテンツデータに対する暗号化やスクランブル処理などが行われており、正当な鍵情報を持つ、もしくはデスクランブル処理を知っている正当なユーザのみが復号処理を行い、正規の文書や画像データなどのコンテンツを享受できる仕組みが提供されている。
【０００３】
上述したようなコンテンツ配信サービスにおいては、コンテンツを配信するコンテンツプロバイダが存在する。このコンテンツプロバイダでは、複数のコンテンツのそれぞれに対して異なるアクセス制御情報の設定を行う必要があり、コンテンツごと、ユーザごと、更にはユーザのアクション（例えば、閲覧、コピーなどのアクション）ごとに異なる鍵による暗号化処理を行うことが想定される。このコンテンツに対する操作処理においては、コンテンツホルダーがコンテンツ利用ユーザに対して処理に対する制限を行う場合が想定される。特に、同じ操作に対しても利用期間（例えば、2000年1月1日から1月31日まで）や利用回数（例えば、５回まで印刷可）などに対する制限情報は数値化された情報として扱われる。この場合、許可情報と同時に、処理許可最大回数や現在の許可回数の残り回数などのカウンタに関する情報も管理する必要がある。
【０００４】
尚、このカウンタはコンテンツに対して印刷、閲覧などの操作（アクションとも呼ぶ）を行うたびに動的に変化するものであるが、カウンタに対する制限（上限値及び下限値）はコンテンツそのものやコンテンツ利用制御情報に含まれる場合がある。
【０００５】
一方、デジタルコンテンツはコンテンツの完全性を保証するために、即ち、送信されて

きたデータが改ざんされたか否かを受信者が検出するために、改ざん防止用の付加データを検証するデジタル署名が施されている。このデジタル署名技術は、データ改ざんだけではなく、インターネット上でのなりすまし、否認などを防止する効果も持ち合わせている。以下、デジタル署名の概要について説明する。

【０００６】

[デジタル署名]

デジタル署名データの生成には、ハッシュ関数と公開鍵暗号とが用いられる。これは、秘密鍵をＫｓ、公開鍵をＫｐとすれば、送信者は、入力データＭにハッシュ処理を施して固定長データＨ（Ｍ）を算出した後、秘密鍵Ｋｓでその固定長データＨ（Ｍ）を変換してデジタル署名データＳを作成し、その後、デジタル署名データＳと入力データＭとを受信者に送信する。

【０００７】

一方、受信者は、そのデジタル署名データＳを公開鍵Ｋｐで変換（復号）したデータと、入力データＭにハッシュ処理を施したデータとが一致するか否かを検証する。ここで、検証結果が一致していなければ、入力データＭに改ざんが行われたと判定する。

【０００８】

デジタル署名にはＲＳＡ，ＤＳＡなど公開鍵暗号方式が用いられおり、署名の安全性は秘密鍵を所有者以外のエンティティが署名を偽造或いは秘密鍵を解読することが計算的に困難であるという事実に基づいている。

【０００９】

図１は、署名作成処理（Sign process）と署名検証処理（Verify process）を表す模式図である。図１に示すように、上述したデジタル署名データを作成する署名作成処理と、デジタル署名データを用いて入力データを検証する署名検証処理が行われる。

【００１０】

[ハッシュ関数]

次に、デジタル署名データの生成を高速化するために用いられるハッシュ関数について説明する。このハッシュ関数は、任意の長さのデータＭに処理を行い、一定の長さの出力データを生成する機能を有する。ここで、出力Ｈ（Ｍ）を平文データＭのハッシュデータと呼ぶ。特に、一方向性ハッシュ関数は、データＭを与えたとき、Ｈ（Ｍ’）＝Ｈ（Ｍ）となる平文データＭ’の算出が計算量的に困難であるという性質を持っている。ここで、一方向性ハッシュ関数としてはＭＤ２、ＭＤ５、ＳＨＡ－１などの標準的なアルゴリズムが存在しており、これらのアルゴリズムは公開されている。

【００１１】

[公開鍵暗号]

次に、公開鍵暗号について説明する。公開鍵暗号は、２つの異なる鍵を利用し、片方の鍵で暗号処理したデータは、もう片方の鍵でしか復号処理できないという性質を持っている。２つの鍵のうち、一方の鍵は公開鍵と呼ばれ、広く公開するようにしている。また、もう片方の鍵は秘密鍵と呼ばれ、本人のみが持つ鍵である。

【００１２】

この公開鍵暗号方式を用いたデジタル署名としては、例えばＤＳＡ署名，ＲＳＡ署名，Schnorr署名などが挙げられる。ここでは一例としてＤＳＡ署名を紹介する。

【００１３】

[ＤＳＡ署名]

非特許文献１に記載の方式を説明する。ｐ，ｑを素数とし，ｐ－１はｑを割り切るものする。ｇをＺ＿ｐ＊（位数ｐの巡回群Ｚ＿ｐから０を省いた乗法群）から任意に選択した、位数ｑの元（生成元）とする。Ｚ＿ｐ＊から任意に選択したｘを秘密鍵とし、それに対する公開鍵ｙをｙ：＝ｇ^{x} ｍｏｄ ｐとおく。Ｈ（）はハッシュ関数とする。

【００１４】

[ＤＳＡ署名作成]

文書Ｍに対する署名の作成手順

１）αをＺ＿ｑから任意に選択し、Ｔ：＝（ｇ$^{\alpha}$ ｍｏｄ ｐ） ｍｏｄ ｑとおく。
２）ｃ：＝Ｈ（Ｍ）とおく。
３）ｓ：＝α^{-1}（ｃ＋ｘＴ） ｍｏｄ ｑとおき，（ｓ，Ｔ）を署名データとする。
【００１５】
[ＤＳＡ署名検証]
文書Ｍに対する署名データ（ｓ，Ｔ）の検証手順
Ｔ＝（ｇ$^{Ｈ（Ｍ）／ｓ}$ｙ$^{Ｔ／ｓ}$ mod ｐ） mod ｑか否かを検証する。
【００１６】
以上のようにデジタル署名技術は、インターネット上でのなりすまし、データ改ざん、否認などを防止する効果がある。
【非特許文献１】Federal Information Processing Standards (FIPS) 186-2, Digital Signature Standard (DSS) , January 2000
【発明の開示】
【発明が解決しようとする課題】
【００１７】
上述したように、デジタルコンテンツに対する制御方法を考えた場合、署名対象データ（被署名データとも呼ぶ）にカウンタの情報が含まれる場合がある。その場合、カウンタを書き換えることで署名検証に失敗してしまう。そのため、カウンタなどの処理を行うごとにダイナミックに変化する情報に対しては署名範囲から除外する必要があり、カウンタ部分が動的に変化しても署名検証を可能にするために、署名付加及び署名検証処理が複雑化するという問題がある。
【００１８】
即ち、署名対象データを集約するようにコンテンツ構造を規定し直すコンテンツデータ構造変更方式においては、カウンタの制限情報はコンテンツ操作と共に付随するデータであり、コンテンツ操作が複数規定されリスト構造で表現されている場合、リスト構造内に含まれる「操作」要素間にカウンタの制限情報が挿入されるために、被署名データが分散され、署名付加及び署名検証処理が複雑化するという問題が生ずる。
【００１９】
この問題を回避するための単純な方法としては、カウンタが取り得る全ての値に対して、一つ一つ署名を施しておき、検証時にそのうちの一つの署名データを選択して検証処理を行う方法が考えられる。しかし、署名データが膨大になるため、コンテンツの情報量が肥大化するという問題や、署名付与時の計算量が増大するという問題が生じる。
【００２０】
このように、従来のデジタル署名方式は被署名データの１ビットでも変更されると署名検証が失敗するものであった。
【００２１】
本発明は、上記課題を解決するためになされたもので、コンテンツと、そのコンテンツごとに動的に変化する数値化データを含む利用制御情報とを用いて署名データを生成可能にすることを目的とする。
【課題を解決するための手段】
【００２２】
本発明は、情報処理装置が行う署名生成方法であって、前記情報処理装置が有する入力手段が、コンテンツと、該コンテンツごとに動的に変化する数値化データを含む利用制御情報とを入力する入力工程と、前記情報処理装置が有する抽出手段が、前記入力工程において入力された利用制御情報に含まれる数値化データの上限値及び下限値を抽出する抽出工程と、前記情報処理装置が有する被署名データ生成手段が、前記利用制御情報に含まれる数値化データを前記抽出工程において抽出された上限値及び下限値に設定して被署名データを生成する被署名データ生成工程と、前記情報処理装置が有する署名生成手段が、前記被署名データ生成工程において生成された前記被署名データと前記コンテンツとを連結し、署名を施して署名データを生成する署名生成工程と、前記情報処理装置が有する出力

手段が、前記入力工程において入力されたコンテンツと、前記入力工程において入力された利用制御情報と、前記署名生成工程において生成された署名データとを出力する出力工程と、を有することを特徴とする。

【発明の効果】

【００２４】

本発明によれば、コンテンツと、そのコンテンツごとに動的に変化する数値化データを含む利用制御情報とを用いて署名データを生成することができる。

【発明を実施するための最良の形態】

【００２５】

以下、図面を参照しながら発明を実施するための最良の形態について詳細に説明する。

【実施例１】

【００２６】

図２は、実施例１における情報処理装置の構成の一例を示す概略図である。尚、本発明を実現する際に、図２に示される全ての機能を使用することが必須でないことは言うまでもない。

【００２７】

情報処理装置２００は、図２に示すように、モデム２１８、モニタ２０２、ＣＰＵ２０３、ＲＯＭ２０４、ＲＡＭ２０５、ＨＤ（ハードディスク）２０６、ネットワーク接続部２０７、ＣＤ２０８、ＦＤ（フレキシブル・ディスク）２０９、ＤＶＤ（デジタル・ビデオ・ディスク又はDigital Versatile Disk）２１０、プリンタ２１５とのインターフェース（Ｉ／Ｆ）２１７、操作部としてのマウス２１２やキーボード２１３などとのインターフェース（Ｉ／Ｆ）２１１で構成され、バス２１６を介して互いに通信可能に接続されている。以下、情報処理装置２００を構成する各ユニットについて順に説明する。

【００２８】

まずマウス２１２及びキーボード２１３は、情報処理装置２００に対する各種指示等をユーザが入力するための操作部である。この操作部を介して入力された情報（操作情報）は、インターフェース２１１を介して情報処理装置２００内に取り込まれる。

【００２９】

情報処理装置２００での各種情報（文字情報や画像情報等）は、プリンタ２１５により印刷出力できるように構成されている。

【００３０】

モニタ２０２は、ユーザへの各種指示情報や、文字情報或いは画像情報等の各種情報の表示を行う。

【００３１】

ＣＰＵ２０３は、情報処理装置２００全体の動作制御を司るものであり、実施例１では制御手段として機能している。即ち、ＣＰＵ２０３は、ＨＤ（ハードディスク）２０６等から処理プログラム（ソフトウェアプログラム）を読み出して実行することで、情報処理装置２００全体を制御する。

【００３２】

特に、実施例１では、ＣＰＵ２０３はＨＤ２０６等から署名作成機能及び署名検証機能を実現する処理プログラムを読み出して実行することで、詳細は後述する情報変換処理を実施する。

【００３３】

ＲＯＭ２０４は、システムのブートプログラムや各種処理プログラム或いは制御データが格納されている。

【００３４】

ＲＡＭ２０５は、ＣＰＵ２０３での各種処理のために、一時的に処理プログラムや処理対象の情報を格納するための作業用エリア等として使用される。

【００３５】

ＨＤ２０６は、大容量記憶装置の一例としての構成要素であり、各種データ或いは各種

処理の実行時にＲＡＭ２０５等へ転送される情報変換処理等のための処理プログラム等を保存する。

【００３６】

ＣＤ（ＣＤドライブ）２０８は、外部記憶媒体の一例としてのＣＤ（ＣＤ－Ｒ）に記憶されたデータを読み込み、また当該ＣＤへデータを書き出す機能を有する。

【００３７】

ＦＤ（ＦＤドライブ）２０９は、上述のＣＤ２０８と同様に、外部記憶媒体の一例としてのＦＤに記憶されたデータを読み出す。また種々のデータを当該ＦＤへ書き込む機能を有している。

【００３８】

ＤＶＤ（ＤＶＤドライブ）２１０は、上述のＣＤ２０８やＦＤ２０９と同様に、外部記憶媒体の一例としてのＤＶＤに記憶されたデータを読み出し、また当該ＤＶＤへデータを書き込む機能を有している。

【００３９】

尚、ＣＤ２０８、ＦＤ２０９、ＤＶＤ２１０等の外部記憶媒体に対して、例えば編集用のプログラム或いはプリンタドライバが記憶されている場合には、これらのプログラムをＨＤ２０６へインストールしておき、必要に応じて、ＲＡＭ２０５へ転送するように構成しても良い。

【００４０】

インターフェース（Ｉ／Ｆ）２１１は、マウス２１２やキーボード２１３によるユーザからの入力を受け付けるためのものである。

【００４１】

モデム２１８は、インターフェース（Ｉ／Ｆ）２１９を介して、例えば公衆回線等を通じて外部の通信ネットワークに接続された通信機器との間で通信を行うための通信モデムである。

【００４２】

ネットワーク接続部２０７は、インターフェース（Ｉ／Ｆ）２１４を介してＬＡＮなどのネットワークへの接続を制御する。

【００４３】

ここで、上述した情報処理装置においてコンテンツ及びコンテンツ利用制御情報の署名を作成する署名作成処理及びその署名を検証する署名検証処理について説明する。

【００４４】

[署名生成]

図３は、実施例１における署名作成処理を示すフローチャートである。尚、この処理は図２に示す情報処理装置２００、特にマウス２１２やキーボード２１３からの入力指示によりＨＤ２０６等に格納されている所定のプログラムをＣＰＵ２０３などが実行することによって実現される処理である。

【００４５】

まず、ステップＳ３０１において、コンテンツ保護対象となるコンテンツＣを入力する。次に、ステップＳ３０２において、ステップＳ３０１で入力したコンテンツＣに対する利用制御情報ＤＲ＿Ｃを入力する。

【００４６】

尚、この利用制御情報ＤＲ＿Ｃの拡張可能マークアップ言語（ＸＭＬ）による記載例は後述するが、利用制御情報ＤＲ＿Ｃには、利用期間（例えば、2000年1月1日から1月31日まで）や利用回数（例えば、５回まで印刷可）などのユーザの操作ごとに動的に変化する数値化データが含まれる。

【００４７】

次に、ステップＳ３０３において、ステップＳ３０２で入力した利用制御情報ＤＲ＿Ｃから数値化データの下限値及び上限値を抽出する。つまり、この利用制御情報ＤＲ＿Ｃに含まれる数値化データはある範囲内に限定されるため、全ての数値化データに対する下限

値及び上限値をそれぞれ抽出する。

【００４８】

次に、ステップＳ３０４において、全ての数値化データを上限値に設定した場合の利用制御情報ＤＡＴＡ＿Ｕと下限値に設定した場合の利用制御情報ＤＡＴＡ＿Ｌという２つの被署名データを作成する。そして、ステップＳ３０５において、ステップＳ３０４で作成した被署名データとコンテンツＣとを以下のように連結し、公開鍵暗号方式などの公知のアルゴリズムを用いて署名データＳを生成する。

【００４９】

Ｃ||ＤＡＴＡ＿Ｕ||ＤＡＴＡ＿Ｌ

ここで、記号「||」はデータの連結を意味するが、どのような構造化データであってもかまわない。

【００５０】

最後に、ステップＳ３０６において、上述のステップＳ３０１で入力したコンテンツＣとステップＳ３０２で入力した利用制御情報ＤＲ＿Ｃとステップ３０５で生成した署名データＳとを一つに纏め、新しくフォーマッティングしたコンテンツデータＰとして出力する。

【００５１】

尚、上述した署名作成処理を図１に示す模式図で考えると、メッセージＭそのものではなく、メッセージＭに変換処理（被署名データの作成及びコンテンツとの連結）を施してハッシュ関数に適用させていることがわかる。また、この変換処理はステップＳ３０４に相当し、署名データの作成はステップＳ３０５に相当する。

【００５２】

以下は、ＸＭＬで記載したコンテンツＣと利用制御情報ＤＲ＿Ｃの一例である。

```
<mdf>
  <contents>
    <binary_embeded type="base64" id="image1">
      deadbeef…
    </binary_embeded>
  </contents>
  <contents_condition>
    <target ref="#image1" >
      <conditions>
        <print>
          <amount upper="5">0</amount>
        </print>
      </conditions>
    </target>
  </contents_condition>
</mdf>。
```

【００５３】

上記のmdf要素には、contents要素（コンテンツＣに対応）とcontents_condition要素（利用制御情報ＤＲ＿Ｃに対応）が含まれている。また、contents要素にはidがimage1という画像データがbase64エンコーディングされて埋め込まれている。そして、画像image1に対する利用制御情報としてはprint操作に対する数量制限が記載されており、upper属性に記載されているように印刷回数は５回までに制限されていることを示している。また、Amount要素に包含される数値は、現在の印刷回数が記載されており、ここでは、初期値の０が入っているが、印刷処理が行われるごとに１ずつ増加されるカウンタであり、動的に変化する。そのため、mdf要素全体に署名を施した場合、カウンタの値を変化させると、署名検証に失敗してしまう。

【００５４】

そこで、実施例１では被署名データＤＡＴＡ＿ＵとＤＡＴＡ＿Ｌを次のようにする。
ＤＡＴＡ＿Ｕ：
<contents_condition>
<target ref="#image1">
<conditions>
<print>
<amount>5</amount>
</print>
</conditions>
</target>
</contents_condition>
ＤＡＴＡ＿Ｌ：
<contents_condition>
<target ref="#image1">
<conditions>
<print>
<amount>0</amount>
</print>
</conditions>
</target>
</contents_condition>。

【００５５】

それぞれamount要素に包含される数値は上限値と下限値である５と０に置き換えられている。ここで上限値と下限値を示すデータを一括して次のＤＡＴＡ＿（Ｕ＋Ｌ）のように表現してもよい。
ＤＡＴＡ＿（Ｕ＋Ｌ）：
<contents_condition>
<target ref="#image1">
<conditions>
<print>
<amount>lower="0" upper="5"</amount>
</print>
</conditions>
</target>
</contents_condition>。

【００５６】

[署名検証]

次に、図４を用いて上述した署名作成処理で作成された署名データを検証する署名検証処理について説明する。

【００５７】

図４は、実施例１における署名検証処理を示すフローチャートである。尚、この処理は図２に示す情報処理装置２００、特にマウス２１２やキーボード２１３からの入力指示によりＨＤ２０６等に格納されている所定のプログラムをＣＰＵ２０３などが実行することによって実現される処理である。

【００５８】

まず、ステップＳ４０１において、利用制御情報が含まれるコンテンツデータＰを入力する。次に、ステップＳ４０２において、上述した署名作成処理のステップＳ３０３での処理と同様に、ステップＳ４０１１で入力した利用制御情報から下限値及び上限値を抽出する。そして、ステップＳ４０３において、上述したステップＳ３０４での処理と同様に、全ての数値化データを上限値にセットした場合の利用制御情報ＤＡＴＡ＿Ｕと下限値に

セットした場合の利用制御情報ＤＡＴＡ＿Ｌという２つの被署名データを作成する。次に、ステップＳ４０４において、ステップＳ４０３で作成した被署名データとコンテンツとを以下のように連結し、署名データＳを用いてコンテンツの（暗号的な）検証を行う。

【００５９】

Ｃ||ＤＡＴＡ＿Ｕ||ＤＡＴＡ＿Ｌ

最後に、ステップＳ４０５において、制御対象となっている数値化データが上限値及び下限値の範囲内に含まれているか検証を行う。

【００６０】

例えば、次のデータの検証を行うことを考える。

```
＜mdf＞
  ＜contents＞
    ＜binary_embeded type="base64" id="image1"＞
      deadbeef…
    ＜/binary_embeded＞
  ＜/contents＞
  ＜contents_condition＞
    ＜target ref="#image1"＞
      ＜conditions＞
        ＜print＞
          ＜amount upper="5"＞2＜/amount＞
        ＜/print＞
      ＜/conditions＞
    ＜/target＞
  ＜/contents_condition＞
  ＜signature＞…＜/signature＞
＜/mdf＞。
```

【００６１】

上記のamount要素には印刷回数を示す２が記載されているが、上限５と下限０の範囲に含まれているため、ステップＳ４０５での範囲検証でvalidの結果を得る。signature要素にはmdf要素全体の署名が記載されているが、Ｗ３Ｃ国際標準規格ＸＭＬSignatureにおけるEnveloped signature に準拠する。また、実施例１における署名検証処理をsignature要素内のtransform要素に記述することでインターオペラビリティを確保することが可能である。

【００６２】

実施例１によれば、動的に変化する数値化データを含む利用制御情報から署名データを生成し、その署名データを用いてコンテンツの検証を行うことができる。

【実施例２】

【００６３】

次に、本発明に係る実施例２について詳細に説明する。尚、実施例２における情報処理装置の構成は、図２を用いて説明した実施例１の構成と同様であり、その説明は省略する。

【００６４】

実施例１では一つの数値化データに対する署名を取り上げたが、複数の数値化データでも処理可能であることを次の例で示す。

```
＜mdf＞
  ＜contents＞
    ＜binary_embeded type="base64" id="image1"＞
      deadbeef…
    ＜/binary_embeded＞
  ＜/contents＞
```

```
  <contents_condition>
    <target ref="#image1">
      <conditions>
        <print>
          <amount upper="5">0</amount>
        </print>
        <display>
          <time lower="2000-01-01-0900" upper="2000-01-31-2100">
            #include-time
          </time>
        </display>
      </conditions>
    </target>
  </contents_condition>
  <signature>…</signature>
</mdf>。
```

【００６５】

ここではprint要素内のamount要素（即ち、print操作における数量制限）と、display要素内のtime要素（即ち、display操作における時間制限）が記載されている。この場合のＤＡＴＡ＿Ｕ、ＤＡＴＡ＿Ｌは次の通りである。

ＤＡＴＡ＿Ｕ：

```
  <contents_condition>
    <target ref="#image1">
      <conditions>
        <print>
          <amount>5</amount>
        </print>
        <display>
          <time>2000-01-01-0900</time>
        </display>
      </conditions>
    </target>
  </contents_condition>
```

ＤＡＴＡ＿Ｌ：

```
  <contents_condition>
    <target ref="#image1">
      <conditions>
        <print>
          <amount>0</amount>
        </print>
        <display>
          <time>2000-01-31-2100</time>
        </display>
      </conditions>
    </target>
  </contents_condition>。
```

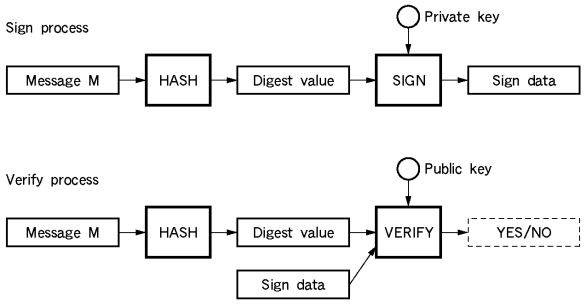
【００６６】

尚、実施例２における署名検証処理は実施例１と同様であるが、コンテンツＰにおけるdisplay要素内のtime要素に内包されるデータには#include-timeとあり、これはシステムにおける現在の時刻と置き換えて検証することを意味する。

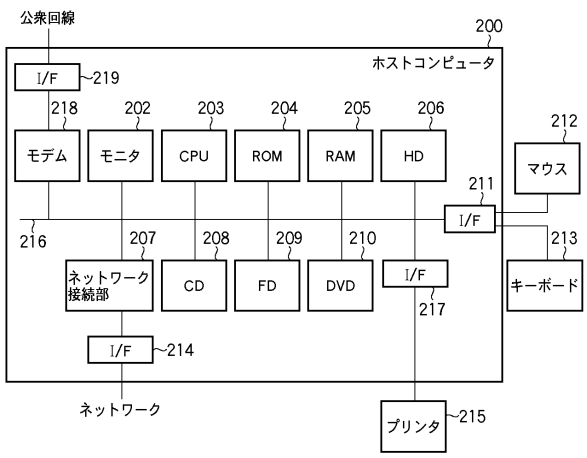
【００６７】
このように、コンテンツ内部にカウンタの情報を含まず、システムやその他のリソースからダウンロードしてくる場合も考えられる。特に、コンテンツ管理サーバから利用チケットをダウンロードして使う場合、コンテンツ利用時に、この数値化データをインタラクティブにダウンロードして検証時にチェックする場合も含まれる。
【００６８】
また、コンテンツそのものに、例えば電子透かし技術を用いて不可分に挿入されている場合も含まれる。この場合、カウンタではなく利用制御情報が電子透かしでコンテンツに挿入される場合も想定される。
【００６９】
[他の実施例]
上記実施例では、上限と下限がある場合について説明したが、上限だけ、もしくは、下限しか持たない場合でも適用可能であることは言うまでもない。
【００７０】
本発明は、複数の機器（例えばホストコンピュータ、インターフェース機器、リーダ、プリンタ等）から構成されるシステムの一部として適用しても、ひとつの機器（例えば、複写機、ファクシミリ装置）からなるものの一部に適用してもよい。
【００７１】
また、本発明は上述した実施例を実現するための装置、方法及び実施例で説明した方法を組み合わせて行う方法のみに限定されるものではなく、上述したシステム又は装置内のコンピュータ（ＣＰＵ又はＭＰＵ）に、上述した実施例を実現するためのソフトウェアのプログラムコードを供給し、このプログラムコードに従って上述したシステム或いは装置のコンピュータが上述の各種デバイスを動作させることにより上述した実施例を実現する場合も本発明の範疇に含まれる。
【００７２】
また、この場合、ソフトウェアのプログラムコード自体が上述した実施例の機能を実現することになり、そのプログラムコード自体、及びそのプログラムコードをコンピュータに供給するための手段、具体的には上記プログラムコードを記録した記録媒体は本発明の範疇に含まれる。
【００７３】
このようなプログラムコードを記録する記録媒体としては、例えばフロッピー（登録商標）ディスク、ハードディスク、光ディスク、光磁気ディスク、ＣＤ－ＲＯＭ、磁気テープ、不揮発性のメモリカード、ＲＯＭ等を用いることができる。
【００７４】
また、上記コンピュータが、供給されたプログラムコードのみに従って各種デバイスを制御することにより、上述した実施例の機能が実現される場合だけではなく、上記プログラムコードがコンピュータ上で稼働しているＯＳ（オペレーティングシステム）、或いは他のアプリケーションソフト等と共同して上述した実施例が実現される場合にも、かかるプログラムコードは本発明の範疇に含まれる。
【００７５】
更に、この供給されたプログラムコードが、コンピュータの機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに格納された後、そのプログラムコードの指示に基づいてその機能拡張ボードや機能格納ユニットに備わるＣＰＵ等が実際の処理の一部又は全部を行い、その処理によって上述した実施例が実現される場合も本発明の範疇に含まれる。
【図面の簡単な説明】
【００７６】
【図１】署名作成処理（Sign process）と署名検証処理（Verify process）を表す模式図である。
【図２】実施例１における情報処理装置の構成の一例を示す概略図である。

【図３】実施例１における署名作成処理を示すフローチャートである。
【図４】実施例１における署名検証処理を示すフローチャートである。

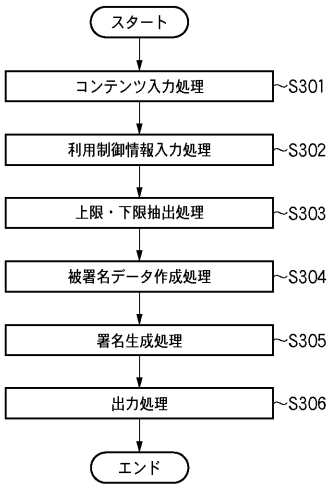
【図１】



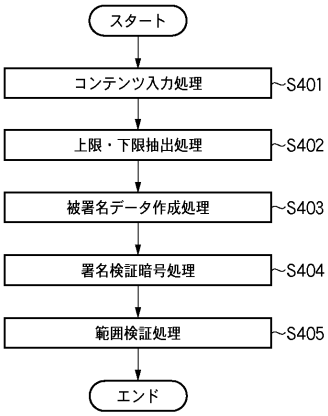
【図２】



【図３】



【図４】


スタート
コンテンツ入力処理
S401
上限・下限抽出処理
S402
被署名データ作成処理
S403
署名検証暗号処理
S404
範囲検証処理
S405
エンド

フロントページの続き

(72)発明者　須賀　祐治
東京都大田区下丸子３丁目３０番２号　キヤノン株式会社内

審査官　唐橋　拓史

(56)参考文献　特開２００３－３０８４４０（ＪＰ，Ａ）
特開２００４－０６２８９０（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｑ　１０／００－５０／００
Ｈ０４Ｌ　９／３２
ＪＳＴＰｌｕｓ（ＪＤｒｅａｍＩＩ）
ＪＳＴ７５８０（ＪＤｒｅａｍＩＩ）