(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4359622号
(P4359622)
(45)発行日　平成21年11月4日(2009.11.4)　(24)登録日　平成21年8月14日(2009.8.14)

(51)Int.Cl.　F I
H04L 9/32 (2006.01)　H04L 9/00 675B

請求項の数 4　(全 35 頁)

(21)出願番号　特願2007-12048(P2007-12048)
(22)出願日　平成19年1月22日(2007.1.22)
(65)公開番号　特開2008-178048(P2008-178048A)
(43)公開日　平成20年7月31日(2008.7.31)
審査請求日　平成20年12月5日(2008.12.5)

(73)特許権者　000005223
富士通株式会社
神奈川県川崎市中原区上小田中４丁目１番１号
(74)代理人　100104190
弁理士　酒井　昭徳
(72)発明者　武仲　正彦
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
(72)発明者　伊豆　哲也
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
(72)発明者　吉岡　孝司
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
最終頁に続く

(54)【発明の名称】電子署名プログラム、および電子署名装置

(57)【特許請求の範囲】
【請求項１】
コンピュータを、
ストリーミングデータを所定単位の部分データに分割する分割手段、
乱数を取得する取得手段、
前記取得手段によって取得された乱数がルート乱数であってリーフ列に対応する乱数列が前記部分データの個数である乱数二分木の生成を、分岐元乱数と第１の数値とに基づくデータを一方向性関数に与えることにより第１の分岐先乱数を生成するとともに、前記分岐元乱数と前記第１の数値とは異なる第２の数値とに基づくデータを前記一方向性関数に与えることにより第２の分岐先乱数を生成することによって行う乱数二分木生成手段、
前記乱数二分木生成手段によって生成された乱数二分木のリーフ列に対応する乱数列と、前記部分データとに基づくデータを一方向性関数に入力することにより、該部分データに対応する出力値列を生成する出力値生成手段、
前記出力値生成手段によって生成された出力値列のみがリーフ列に対応している二分木である出力値二分木の生成を、隣り合うリーフ又は内部ノードに対応する出力値に基づくデータを前記一方向性関数に入力することにより内部ノードに対応する出力値を生成することによって行う出力値二分木生成手段、
前記出力値二分木生成手段によって生成された出力値二分木のルートに対応する出力値であるルート出力値に基づき、前記ストリーミングデータに対する署名時の電子署名である署名時署名を生成する署名時署名生成手段、

前記部分データ、前記ルート乱数、前記ルート出力値、および、前記署名時署名を出力する署名時出力手段、
として機能させることを特徴とする電子署名プログラム。
【請求項２】
前記コンピュータを、
前記ストリーミングデータの抽出範囲を受け付ける受付手段、
前記出力値二分木に基づき、前記受け付けた抽出範囲に該当しない部分データである削除部分データに対応する出力値のみをリーフとする１つ以上の二分木である削除部分出力値二分木を抽出し、該削除部分出力値二分木それぞれのルートに対応する出力値である削除部分ルート出力値列を特定する削除部分ルート出力値特定手段、
前記乱数二分木に基づき、前記受け付けた抽出範囲に該当する部分データである抽出部分データに対応する乱数列のみをリーフとする抽出部分乱数二分木を抽出し、該抽出部分乱数二分木のルートに対応する乱数である抽出部分ルート乱数を特定する抽出部分ルート乱数特定手段、
前記削除部分ルート出力値列および前記抽出部分ルート乱数に基づき、抽出時の電子署名である抽出時署名を生成する抽出時署名生成手段、
前記抽出部分データ、前記削除部分ルート出力値列、前記抽出部分ルート乱数、および、前記抽出時署名を出力する抽出時出力手段、
として機能させることを特徴とする請求項１に記載の電子署名プログラム。
【請求項３】
ストリーミングデータを所定単位の部分データに分割する分割手段と、
乱数を取得する取得手段と、
前記取得手段によって取得された乱数がルート乱数であってリーフ列に対応する乱数列が前記部分データの個数である乱数二分木の生成を、分岐元乱数と第１の数値とに基づくデータを一方向性関数に与えることにより第１の分岐先乱数を生成するとともに、前記分岐元乱数と前記第１の数値とは異なる第２の数値とに基づくデータを前記一方向性関数に与えることにより第２の分岐先乱数を生成することによって行う乱数二分木生成手段と、
前記乱数二分木生成手段によって生成された乱数二分木のリーフ列に対応する乱数列と、前記部分データとに基づくデータを一方向性関数に入力することにより、該部分データに対応する出力値列を生成する出力値生成手段と、
前記出力値生成手段によって生成された出力値列のみがリーフ列に対応している二分木である出力値二分木の生成を、隣り合うリーフ又は内部ノードに対応する出力値に基づくデータを前記一方向性関数に入力することにより内部ノードに対応する出力値を生成することによって行う出力値二分木生成手段と、
前記出力値二分木生成手段によって生成された出力値二分木のルートに対応する出力値であるルート出力値に基づき、前記ストリーミングデータに対する署名時の電子署名である署名時署名を生成する署名時署名生成手段と、
前記部分データ、前記ルート乱数、前記ルート出力値、および、前記署名時署名を出力する署名時出力手段と、
を備えたことを特徴とする電子署名装置。
【請求項４】
前記ストリーミングデータの抽出範囲を受け付ける受付手段と、
前記出力値二分木に基づき、前記受け付けた抽出範囲に該当しない部分データである削除部分データに対応する出力値のみをリーフとする１つ以上の二分木である削除部分出力値二分木を抽出し、該削除部分出力値二分木それぞれのルートに対応する出力値である削除部分ルート出力値列を特定する削除部分ルート出力値特定手段と、
前記乱数二分木に基づき、前記受け付けた抽出範囲に該当する部分データである抽出部分データに対応する乱数列のみをリーフとする抽出部分乱数二分木を抽出し、該抽出部分乱数二分木のルートに対応する乱数である抽出部分ルート乱数を特定する抽出部分ルート乱数特定手段と、

前記削除部分ルート出力値列および前記抽出部分ルート乱数に基づき、抽出時の電子署名である抽出時署名を生成する抽出時署名生成手段と、
前記抽出部分データ、前記削除部分ルート出力値列、前記抽出部分ルート乱数、および、前記抽出時署名を出力する抽出時出力手段と、
をさらに備えたことを特徴とする請求項３に記載の電子署名装置。

【発明の詳細な説明】

【技術分野】

【０００１】

この発明は、動画像や音声といった署名対象のストリーミングデータからの部分的な抽出（具体的には変更・抽出・墨塗りなどを含む）、署名対象のストリーミングデータからの抽出箇所の特定、ならびに、抽出されたストリーミングデータの正当性を担保し、第三者証明を可能にする電子署名プログラム、および電子署名装置に関する。

【背景技術】

【０００２】

店舗や繁華街、集合住宅などでの監視カメラ設置や、業務車両へのドライブレコーダ設置等が一般化し、動画像を証拠物件として取り扱う事例が増加している。また、電話による取引やサポート業務のトラブル対策として、顧客とオペレータとの会話を録音し証拠として保持することも常識となりつつある。

【０００３】

現在、動画像や音声を証拠とする場合、ビデオテープや画像・音声ファイルをそのまま提供している。しかし、画像・音声保存のデジタル化が進めば、それらの改ざんや編集は容易になり、証拠として取り扱う場合は署名やタイムスタンプといった第三者証明が必要となる。現に電話オペレータの音声をタイムスタンプ付で録音・記録するサービスや製品が販売されており、今後このような技術のニーズが高まることが予想される。

【０００４】

一方で、増加する監視カメラ等に対して、撮影された映像の利用に対するプライバシーの保護が問題となり、議論が行われている。また、個人情報保護法の施行などにより、個人のプライバシー情報の利用が厳しく制限され、本人の要求があれば、開示や部分的な削除などが必要である。

【０００５】

証拠性とプライバシー保護の両立という課題に対して、電子文書の一部に対する部分的な原本性（完全性）の保証や秘匿（墨塗り）する墨塗り署名技術の研究が進んでいる。特に、下記非特許文献１および下記特許文献１は、電子署名を施した文書の原本性を保証しながら、一部を追加や変更、墨塗り（秘匿）、削除が可能な署名技術である。

【０００６】

これらの署名技術を適用することにより、署名付き電子文書に対して墨塗りを施した状態でも署名検証が可能、かつ、墨塗り（非特許文献１の場合「変更」や「追加」も可能）箇所以外は改変がないことを第三者証明することが可能となる。

【０００７】

ここで、非特許文献１の署名技術について説明する。この署名技術はＰＩＡＴ技術と呼ばれる部分完全性保証署名技術であり、署名者と抽出者と検証者とからなる三者モデルにより署名対象となる電子データの部分完全性保証をおこなう。

【０００８】

署名者とは、署名対象となる電子データに対して署名を施す者またはその者が扱うコンピュータ装置であり、署名することで署名対象の電子データの内容を保証する。署名対象のうちどの部分が抽出されるかわからないという条件下で署名を施す必要がある。

【０００９】

抽出者とは、署名対象となる電子データから一部の電子データを抽出（墨塗り・変更）する者またはその者が扱うコンピュータ装置であり、署名者が署名した電子データから、部分的に電子データを抽出して、検証者に開示する。抽出された電子データを抽出データ

と称す。抽出方法には、顕名抽出と匿名抽出の２種類の方法がある。抽出者が抽出処理を匿名でおこなう方法を匿名抽出、抽出者の情報を同時に開示し、誰がその抽出処理をおこなったかを明示する方法を顕名抽出と呼ぶ。

【００１０】

検証者とは、抽出者により開示された抽出データが署名者によって保証されているかどうかを検証する者またはその者が扱うコンピュータ装置である。匿名抽出の場合、開示された抽出データは署名者が署名をした電子データの一部であることを検証する。顕名抽出の場合には、開示された抽出データは、署名者が署名をした電子データの一部であることに加え、その抽出処理が抽出者によっておこなわれたことを検証する。つぎに、ＰＩＡＴ技術のアルゴリズムについて説明する。

【００１１】

図２２は、ＰＩＡＴ技術の署名者用のアルゴリズムの概要を示す説明図である。図２２において、署名者は、署名対象となる電子文書（全文字列）２２００を部分データ（各行の文字列）に分割し、各部分データのハッシュ値を計算して、ハッシュ値集合２２０１を作成する。その後、作成したハッシュ値集合２２０１に対して署名者の電子署名をおこない、ハッシュ値集合２２０１と署名者の電子署名２２０２をあわせて署名者のＰＩＡＴ署名２２０３とする。

【００１２】

図２３は、ＰＩＡＴ技術の抽出者用のアルゴリズムの概要を示す説明図である。図２３において、抽出者は、署名者がＰＩＡＴ署名を施したデータから、部分データ（抽出データ２３００）を抽出する。その後、署名者と同様の操作を施す。そして、そのハッシュ値集合２３０１と抽出者の電子署名２３０２をあわせて抽出者のＰＩＡＴ署名２３０３とする。ここで、抽出ではなく削除部分を明確に示したい場合は、墨塗り処理となり、削除後の部分データを“ＸＸＸＸＸＸＸ”など墨塗りであることが区別可能な部分データに変更する。

【００１３】

図２４は、ＰＩＡＴ技術の検証者用のアルゴリズムの概要を示す説明図である。図２４において、検証者は、まず、署名者と抽出者のＰＩＡＴ署名から、ハッシュ値集合２２０１，２３０１の完全性を検証する。つぎ、開示された抽出データ２３００からハッシュ値集合２３０１を作成し、そのハッシュ値集合２３０１が抽出者のＰＩＡＴ署名２３０３に含まれるハッシュ値集合２３０１と同一であることを検証する。

【００１４】

最後に、署名者のハッシュ値集合２２０１と抽出者のハッシュ値集合２３０１を比較することにより、ハッシュ値が同一であるデータの位置が電子文書２２００における抽出位置であることがわかる。もし、抽出データ２３００のハッシュ値集合２３０１が署名者のＰＩＡＴ署名２２０３のハッシュ値集合２２０１に含まれていない場合は、その抽出データ２３００は改ざんされていることになる。

【００１５】

ＰＩＡＴ技術では、基本的に顕名抽出を前提としており、誰がどの部分を抽出したかまで検証可能である。しかし、匿名抽出の場合は、抽出者がおこなうＰＩＡＴ署名処理は省略可能で、その場合、検証者は、署名者のＰＩＡＴ署名内のハッシュ値集合とその電子署名の検証処理、開示された抽出データからのハッシュ値集合の生成処理、および署名者のＰＩＡＴ署名内のハッシュ値集合との比較処理をおこなう。

【００１６】

また、上記非特許文献１の署名技術（ＰＩＡＴ技術）を、電子文書のみならず、そのままＭＰＥＧ－１のような動画像ファイルや音声ファイルといったストリーミングデータに適用することもできる。

【００１７】

この場合、署名対象となるストリーミングデータを部分データに分割する。ＭＰＥＧ－１のストリーミングデータを抽出可能なように部分データに分割する場合、画像単位とＧ

ＯＰ（Ｇｒｏｕｐ　ｏｆ　Ｐｉｃｔｕｒｅ）単位が考えられる。

【００１８】

しかし、ＭＰＥＧ－１ではフレーム間予測技術が用いられているため、ＰフレームおよびＢフレームでは画像単位の独立性がなく、抽出が制限されることが考えられる。一方、ＧＯＰは、何枚かの画像をまとめた動画像の最小単位であり、その単位で独立した再生が可能であり、動画像を途中から再生したり編集したりするための構造である。ここでは、単純化のために、部分データへの分割をＧＯＰ単位とする。

【００１９】

署名者は、署名対象の動画像ファイル（たとえば、ＭＰＥＧ－１ファイル）をＧＯＰ単位で分割し、各ＧＯＰのハッシュ値を計算して、ハッシュ値集合を作成する。その後、作成したハッシュ値集合に対して署名者の電子署名を施し、ハッシュ値集合と電子署名をあわせてＰＩＡＴ署名とする。

【００２０】

抽出者は、署名者がＰＩＡＴ署名を施した署名対象の動画像ファイルから、必要な部分の動画像を抽出する。匿名抽出の場合はそのまま抽出した動画像と署名者のＰＩＡＴ署名を開示する。顕名抽出の場合はその後、署名者と同様の操作を行って、抽出者のＰＩＡＴ署名を作成し、抽出動画像、署名者のＰＩＡＴ署名および抽出者のＰＩＡＴ署名を開示する。

【００２１】

検証者は、顕名抽出の場合、まず、署名者のＰＩＡＴ署名および抽出者のＰＩＡＴ署名から、ハッシュ値集合の完全性を検証する。つぎに、開示された抽出動画像からハッシュ値集合を作成し、抽出者のＰＩＡＴ署名に含まれるハッシュ値集合と同一であることを検証する。

【００２２】

最後に、署名者のハッシュ値集合と抽出者のハッシュ値集合を比較することにより、ハッシュ値が同一であるデータの位置が署名対象の動画像ファイルの抽出位置であることがわかる。匿名抽出の場合は、署名者のＰＩＡＴ署名内のハッシュ値集合とその電子署名の検証処理、開示された抽出動画像からのハッシュ値集合の生成処理、および署名者のＰＩＡＴ署名内のハッシュ値集合との比較処理をおこない、部分情報であるかどうかを確認する。

【００２３】

つぎに、特許文献１の署名技術について説明する。特許文献１の署名技術は、署名関連データの一部であるハッシュ値を二分木で管理することで署名関連データ量を削減することができ、その署名対象となるデータは、上述したＰＩＡＴ技術と同様、電子文書のみならず、動画像や音声のようなストリーミングデータも署名対象となる。ここでは、署名対象を動画像のストリーミングデータとした場合を例にあげて説明する。

【００２４】

図２５は、動画像のストリーミングデータを署名対象とした特許文献１の署名技術における署名者による署名処理を示す説明図である。図２５において、特許文献１の署名技術では、署名対象となる動画像ファイルＦをＧＯＰ単位で分割した部分動画像f_1～f_5のハッシュ値h_1～h_5を完全二分木で管理する。そのためにはダミーデータＤの追加が必要となる。

【００２５】

署名者は、まず、署名対象の動画像ファイルＦの末尾にダミーデータＤを付加し、１段目の処理として、ＧＯＰ単位でハッシュ値h_1～h_8を算出する。つぎに、２段目の処理として、隣接するハッシュ値h_x，h_yにおいて、ハッシュ値h_xの末尾とハッシュ値h_yの先頭を連結して、あらたなハッシュ値$h_{x,y}$を算出する。

【００２６】

３段目以降もハッシュ値が単一になるまで繰り返して、ハッシュ値二分木Ｔを生成する。単一になったハッシュ値をルートＲａと称す。図２５では、４段目のハッシュ値でルー

トＲａが生成される。署名者は、ルートＲａに自己の電子署名Ｓａを施す。
【００２７】
そして、このとき保存される開示情報Ａは、元の動画像ファイルＦと、ダミーデータＤ（またはダミーデータＤの部分動画像f_6～f_8のハッシュ値をリーフとするルートハッシュ値列｛h_6，$h_{7,8}$｝）と、署名者の電子署名Ｓａが施されたルートＲａである。
【００２８】
図２６は、動画像ファイルを署名対象とした特許文献１の署名技術における抽出者による抽出処理を示す説明図である。抽出者は、署名対象の動画像ファイルＦから部分動画像f_3，f_4を抽出し、１段目の処理として、動画像ファイルＦおよびダミーデータＤの各部分動画像f_1～f_8のハッシュ値h_1～h_8を算出する。このあと、抽出された部分動画像f_3，f_4以外のすべての部分動画像f_1，f_2，f_5～f_8を消去する。
【００２９】
そして、２段目以降、隣接するハッシュ値h_xの末尾とハッシュ値h_yの先頭を連結して、あらたなハッシュ値$h_{x,y}$を算出し、ハッシュ値が単一になるまで繰り返してハッシュ値二分木Ｔを生成する。図２６では、４段目のハッシュ値でルートＲｂが生成される。抽出者は、ルートＲｂに自己の電子署名Ｓｂを施す。
【００３０】
そして、このとき開示される署名関連データＢは、抽出された動画像f_3，f_4と、消去された部分動画像f_1，f_2，f_5～f_8のみのハッシュ値から得られるルートハッシュ値$h_{1,2}$，$h_{5,8}$と、抽出者の電子署名Ｓｂが施されたルートＲｂである。
【００３１】
【非特許文献１】吉岡　孝司、武仲　正彦著　「電子文書の訂正・流通を考慮した部分完全性保証技術の提案」　第３回情報科学技術フォーラム　ＦＩＴ　２００４，Ｍ－０６６，２００４年
【特許文献１】特開２００６－６０７２２号公報
【発明の開示】
【発明が解決しようとする課題】
【００３２】
しかしながら、上述した署名技術では、動画像ファイルや音声ファイルのように大きなストリーミングデータの一部を抽出するような場合、抽出された動画像以外の動画像のハッシュ値をすべて保持しなければならないため、署名関連データのデータ量が膨大になるという問題があった。
【００３３】
また、上記特許文献１の署名技術では、署名関連データの一部であるハッシュ値を二分木により管理することで署名関連データのデータ量を削減することができるが、ハッシュ値を完全二分木で管理するためにはダミーデータの追加が必要となる。
【００３４】
したがって、ダミーデータＤが付加された分、保持しなければならない情報量（ダミーデータＤまたはダミーデータＤのみからなるハッシュツリーＴのハッシュ値群）が増加する場合があるという問題があった。
【００３５】
また、ダミーデータＤが付加された分、そのハッシュ値計算が必要となり、処理時間も増加するという問題があった。さらに、署名対象の動画像ファイルの末尾のデータをダミーデータに置き換えることで、署名対象の動画像ファイルの偽造（短縮）ができてしまうという問題があった。
【００３６】
また、抽出された部分動画像f_3，f_4の唯一性を保証するために必要な署名関連データの一部である乱数もそのまま保持しなければならならない。図２７は、部分動画像f_1～f_5に付加された乱数を示す説明図である。図２７において、ＧＯＰ単位の部分動画像f_1～f_5ごとのハッシュ値h_1～h_5を単一のハッシュ値ｈにまとめることができるが、部分

動画像f_1～f_5の数と同数の乱数r_1～r_5が必要となり、署名関連データのデータ量が膨大になるという問題があった。

【００３７】

また、上述したように、動画像ファイルなどのストリーミングデータに対し単純にＰＩＡＴ技術を適用すれば、動画像の抽出をおこなっても署名対象の動画像ファイルの一部であることを署名技術で保証することができる。

【００３８】

しかしながら、上述したように、署名対象の動画像ファイルが記録時間の長い動画像ファイルやフレームレートの高い動画像ファイルである場合、フレーム数やＧＯＰ数が大きくなりＰＩＡＴ署名に含まれるハッシュ値集合のデータ量が増加してしまうという問題がある。

【００３９】

また、署名対象の動画像ファイルの内容がアニメーションである場合、同じフレームやＧＯＰが繰り返し出現する。ＰＩＡＴ技術では、同じ画像が部分データに出現すると、ハッシュ値集合により消去部分の動画像が復元できる可能性がある。部分データの唯一性を保証するために、同じ画像が部分データに出現する場合、元の部分データ毎に乱数を付加する必要がある。

【００４０】

図２８は、アニメーションの動画像ファイルを示す説明図である。図２８において、画像E_1，E_2を消去しても、画像E_1～E_3のハッシュ値h_1～h_3が同一であれば、消去された画像E_1，E_2は画像E_3と同一画像であるため、消去された画像E_1，E_2は画像E_3のハッシュ値から復元されてしまう。この復元を防止するには、各画像E_1～E_4に乱数を付加しなければならない。

【００４１】

しかしながら、乱数を付加する場合、さらにＧＯＰ数×乱数長のデータを追加的に保持する必要があるため、動画像以外の保存データがさらに増加してしまうという問題がある。たとえば、テレビジョン程度のフレームレートの一時間の動画像にＰＩＡＴ署名を施す場合、署名データは数百キロ［Ｂｙｔｅ］～数メガ［Ｂｙｔｅ］必要となり、画像データからすればデータ量は小さいものの、署名関連データの増大が問題となることが考えられる。

【００４２】

この発明は、上述した従来技術による問題点を解消するため、動画像や音声などの署名対象となるストリーミングデータの原本性の保証、署名対象からのプライバシー保護可能なデータ抽出、および署名関連データの大幅なデータ量削減を実現することができる電子署名プログラム、および電子署名装置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】

【００４３】

上述した課題を解決し、目的を達成するため、本発明にかかる電子署名プログラムは、コンピュータを、ストリーミングデータを所定単位の部分データに分割する分割手段、乱数を取得する取得手段、前記取得手段によって取得された乱数がルート乱数であってリーフ列に対応する乱数列が前記部分データの個数である乱数二分木の生成を、分岐元乱数と第１の数値とに基づくデータを一方向性関数に与えることにより第１の分岐先乱数を生成するとともに、前記分岐元乱数と前記第１の数値とは異なる第２の数値とに基づくデータを前記一方向性関数に与えることにより第２の分岐先乱数を生成することによって行う乱数二分木生成手段、前記乱数二分木生成手段によって生成された乱数二分木のリーフ列に対応する乱数列と、前記部分データとに基づくデータを一方向性関数に入力することにより、該部分データに対応する出力値列を生成する出力値生成手段、前記出力値生成手段によって生成された出力値列のみがリーフ列に対応している二分木である出力値二分木の生成を、隣り合うリーフ又は内部ノードに対応する出力値に基づくデータを前記一方向性関数に入力することにより内部ノードに対応する出力値を生成することによって行う出力値

二分木生成手段、前記出力値二分木生成手段によって生成された出力値二分木のルートに対応する出力値であるルート出力値に基づき、前記ストリーミングデータに対する署名時の電子署名である署名時署名を生成する署名時署名生成手段、前記部分データ、前記ルート乱数、前記ルート出力値、および、前記署名時署名を出力する署名時出力手段、として機能させることを特徴とする。

【００４４】

また、上記発明において、前記コンピュータを、前記ストリーミングデータの抽出範囲を受け付ける受付手段、前記出力値二分木に基づき、前記受け付けた抽出範囲に該当しない部分データである削除部分データに対応する出力値のみをリーフとする１つ以上の二分木である削除部分出力値二分木を抽出し、該削除部分出力値二分木それぞれのルートに対応する出力値である削除部分ルート出力値列を特定する削除部分ルート出力値特定手段、前記乱数二分木に基づき、前記受け付けた抽出範囲に該当する部分データである抽出部分データに対応する乱数列のみをリーフとする抽出部分乱数二分木を抽出し、該抽出部分乱数二分木のルートに対応する乱数である抽出部分ルート乱数を特定する抽出部分ルート乱数特定手段、前記削除部分ルート出力値列および前記抽出部分ルート乱数に基づき、抽出時の電子署名である抽出時署名を生成する抽出時署名生成手段、前記抽出部分データ、前記削除部分ルート出力値列、前記抽出部分ルート乱数、および、前記抽出時署名を出力する抽出時出力手段、として機能させることを特徴とする。

【００４５】

また、この発明にかかる電子署名装置は、ストリーミングデータを所定単位の部分データに分割する分割手段と、乱数を取得する取得手段と、前記取得手段によって取得された乱数がルート乱数であってリーフ列に対応する乱数列が前記部分データの個数である乱数二分木の生成を、分岐元乱数と第１の数値とに基づくデータを一方向性関数に与えることにより第１の分岐先乱数を生成するとともに、前記分岐元乱数と前記第１の数値とは異なる第２の数値とに基づくデータを前記一方向性関数に与えることにより第２の分岐先乱数を生成することによって行う乱数二分木生成手段と、前記乱数二分木生成手段によって生成された乱数二分木のリーフ列に対応する乱数列と、前記部分データとに基づくデータを一方向性関数に入力することにより、該部分データに対応する出力値列を生成する出力値生成手段と、前記出力値生成手段によって生成された出力値列のみがリーフ列に対応している二分木である出力値二分木の生成を、隣り合うリーフ又は内部ノードに対応する出力値に基づくデータを前記一方向性関数に入力することにより内部ノードに対応する出力値を生成することによって行う出力値二分木生成手段と、前記出力値二分木生成手段によって生成された出力値二分木のルートに対応する出力値であるルート出力値に基づき、前記ストリーミングデータに対する署名時の電子署名である署名時署名を生成する署名時署名生成手段と、前記部分データ、前記ルート乱数、前記ルート出力値、および、前記署名時署名を出力する署名時出力手段と、を備えたことを特徴とする。

【００４６】

また、上記発明において、前記ストリーミングデータの抽出範囲を受け付ける受付手段と、前記出力値二分木に基づき、前記受け付けた抽出範囲に該当しない部分データである削除部分データに対応する出力値のみをリーフとする１つ以上の二分木である削除部分出力値二分木を抽出し、該削除部分出力値二分木それぞれのルートに対応する出力値である削除部分ルート出力値列を特定する削除部分ルート出力値特定手段と、前記乱数二分木に基づき、前記受け付けた抽出範囲に該当する部分データである抽出部分データに対応する乱数列のみをリーフとする抽出部分乱数二分木を抽出し、該抽出部分乱数二分木のルートに対応する乱数である抽出部分ルート乱数を特定する抽出部分ルート乱数特定手段と、前記削除部分ルート出力値列および前記抽出部分ルート乱数に基づき、抽出時の電子署名である抽出時署名を生成する抽出時署名生成手段と、前記抽出部分データ、前記削除部分ルート出力値列、前記抽出部分ルート乱数、および、前記抽出時署名を出力する抽出時出力手段と、をさらに備えたことを特徴とする。

【発明の効果】

【００５１】
本発明によれば、動画像や音声などの署名対象となるストリーミングデータの原本性の保証、署名対象からのプライバシー保護可能なデータ抽出、および署名関連データの大幅なデータ量削減を実現することができるという効果を奏する。

【発明を実施するための最良の形態】
【００５２】
以下に添付図面を参照して、この発明にかかる電子署名プログラム、該プログラムを記録した記録媒体、電子署名装置、および電子署名方法の好適な実施の形態を詳細に説明する。
【００５３】
（電子署名装置のハードウェア構成）
まず、この発明の実施の形態にかかる電子署名装置のハードウェア構成について説明する。図１は、この発明の実施の形態にかかる電子署名装置のハードウェア構成を示すブロック図である。
【００５４】
図１において、電子署名装置１００は、コンピュータ本体１１０と、入力装置１２０と、出力装置１３０と、から構成されており、不図示のルータやモデムを介してＬＡＮ，ＷＡＮやインターネットなどのネットワーク１４０に接続可能である。
【００５５】
コンピュータ本体１１０は、ＣＰＵ，メモリ，インターフェースを有する。ＣＰＵは、電子署名装置１００の全体の制御を司る。メモリは、ＲＯＭ，ＲＡＭ，ＨＤ，光ディスク１１１，フラッシュメモリから構成される。メモリはＣＰＵのワークエリアとして使用される。
【００５６】
また、メモリには各種プログラムが格納されており、ＣＰＵからの命令に応じてロードされる。ＨＤおよび光ディスク１１１はディスクドライブによりデータのリード／ライトが制御される。また、光ディスク１１１およびフラッシュメモリはコンピュータ本体１１０に対し着脱自在である。インターフェースは、入力装置１２０からの入力、出力装置１３０への出力、ネットワーク１４０に対する送受信の制御をおこなう。
【００５７】
また、入力装置１２０としては、キーボード１２１、マウス１２２、スキャナ１２３などがある。キーボード１２１は、文字、数字、各種指示などの入力のためのキーを備え、データの入力をおこなう。また、タッチパネル式であってもよい。マウス１２２は、カーソルの移動や範囲選択、あるいはウィンドウの移動やサイズの変更などをおこなう。スキャナ１２３は、画像を光学的に読み取る。読み取られた画像は画像データとして取り込まれ、コンピュータ本体１１０内のメモリに格納される。なお、スキャナ１２３にＯＣＲ機能を持たせてもよい。
【００５８】
また、出力装置１３０としては、ディスプレイ１３１、プリンタ１３２、スピーカ１３３などがある。ディスプレイ１３１は、カーソル、アイコンあるいはツールボックスをはじめ、文書、画像、機能情報などのデータを表示する。また、プリンタ１３２は、画像データや文書データを印刷する。またスピーカ１３３は、効果音や読み上げ音などの音声を出力する。
【００５９】
（電子署名装置１００の機能的構成）
つぎに、この発明の実施の形態にかかる電子署名装置１００の機能的構成について説明する。電子署名装置１００は、署名処理、抽出処理、および検証処理を実行することができる。ここでは、上記処理ごとの機能的構成について説明する。
【００６０】

図２は、署名時における電子署名装置１００の機能的構成を示す説明図である。図２において、電子署名装置１００は、二分木生成部２０１と、電子署名処理部２０２と、から構成されている。二分木生成部２０１には、開示情報である動画像や音声などのストリーミングデータＳＴが入力される。また、乱数使用モードの場合、ルート乱数r_Rの生成依頼を受け付ける。乱数使用モードとは、乱数を使用してハッシュ値を算出するモードである。

【００６１】

また、二分木生成部２０１は、ストリーミングデータＳＴが与えられると、ハッシュ値をノードとしてツリー構造化されたハッシュ値二分木を生成し、そのルートとなるルートハッシュ値h_Rを電子署名処理部２０２に出力する。また、ルート乱数r_Rの生成依頼を受け付けると、乱数をノードとしてツリー構造化された乱数二分木を生成し、そのルートとなるルート乱数r_Rを開示情報として出力する。

【００６２】

また、電子署名処理部２０２は、二分木生成部２０１から出力されたルートハッシュ値を用いて、署名者の電子署名S_Aを生成する。電子署名処理部２０２による電子署名の生成は、既存の電子署名技術により実行される。

【００６３】

図３は、抽出時における電子署名装置１００の機能的構成を示す説明図である。図３において、電子署名装置１００は、抽出部３０１と、二分木生成部２０１と、電子署名処理部２０２と、から構成されている。抽出部３０１は、抽出情報を受け付ける。抽出情報とは、抽出者が抽出しようとするストリーミングデータＳＴの範囲である。

【００６４】

抽出部３０１は、抽出情報を受け付けると、ストリーミングデータＳＴの中からその抽出情報に該当する範囲のデータ列（以下、「抽出データ列L_{de}」と称す。）を抽出する。抽出データ列L_{de}は一または複数の連続する部分データ（以下、「抽出データ」と称す。）からなり、開示情報として出力される。匿名抽出である場合、この抽出データ列L_{de}と署名者の電子署名S_Aとを開示情報として出力することとなる。

【００６５】

また、二分木生成部２０１は、抽出情報を受け付けると、抽出データ列L_{de}を検証するためのルートハッシュ値列L_{hRD}を開示情報として出力する。また、乱数使用モードの場合、署名時に得られたルート乱数r_Rを受け付けると、抽出データ列L_{de}に応じたルート乱数列L_{rRE}を開示情報として出力する。

【００６６】

また、電子署名処理部２０２は、顕名抽出である場合、二分木生成部２０１によって生成されたハッシュ値二分木を用いて、抽出者の電子署名S_Bを生成する。また、乱数使用モードの場合、さらに、乱数二分木も用いて抽出者の電子署名S_Bを生成する。顕名抽出である場合、生成された抽出者の電子署名S_Bは、抽出データ列L_{de}および署名者の電子署名S_Aとともに開示情報として出力される。

【００６７】

図４は、検証時における電子署名装置１００の機能的構成を示す説明図である。図４において、電子署名装置１００は、二分木生成部２０１と、電子署名処理部２０２と、から構成されている。二分木生成部２０１は、抽出データ列L_{de}およびルートハッシュ値列L_{hRD}を受け付けて、ハッシュ値二分木を生成し、ハッシュ値二分木のルートハッシュ値列L_{hRD}を電子署名処理部２０２に出力する。乱数使用モードの場合、ルート乱数列L_{rRE}を受け付けて、乱数二分木を生成する。そして、抽出データ列L_{de}の各抽出データに応じた乱数となるリーフを用いて、ハッシュ値二分木を生成（復元）する。

【００６８】

電子署名処理部２０２は、匿名抽出の場合、署名者の電子署名S_Aを受け付けて、復元されたハッシュ値二分木のルートハッシュ値から、署名者の電子署名S_Aの正当性を検証する。顕名抽出の場合、それに加えて、抽出者の電子署名S_Bを受け付けて、ルート乱数

列L_{rRE}およびルートハッシュ値列L_{hRD}から、抽出者の電子署名S_Bの正当性を検証する。電子署名処理部２０２による電子署名の検証（確認）は、既存の電子署名技術により実行される。

【００６９】

（二分木生成部２０１の機能的構成）

つぎに、上述した二分木生成部２０１の機能的構成について説明する。二分木生成部２０１の機能は署名・抽出時と検証時とで構成が異なるため、署名・抽出時と検証時とで分けて説明する。

【００７０】

＜署名・抽出時＞

図５は、署名・抽出時における二分木生成部２０１の機能的構成を示すブロック図である。二分木生成部２０１は、ストリーミングデータＳＴに対するハッシュ値二分木Ｔｈや乱数二分木Ｔｒなどの各種二分木を生成する機能である。

【００７１】

図５において、二分木生成部２０１は、分割部５０１と、ハッシュ値計算部５０２と、ハッシュ値列生成部５０３と、ハッシュ値二分木生成部５０４と、削除ハッシュ値二分木生成部５０５と、乱数生成部５０６と、スイッチ５０７と、乱数二分木生成部５０８と、抽出乱数二分木生成部５０９と、を備えている。

【００７２】

分割部５０１は、ストリーミングデータＳＴが入力されると、所定単位の部分データに分割する。部分データの集合が部分データ列となる。分割単位は、任意に設定することができる。たとえば、署名対象となるストリーミングデータＳＴがＭＰＥＧ－１である場合、画像単位とＧＯＰ（Ｇｒｏｕｐ　ｏｆ　Ｐｉｃｔｕｒｅ）単位で分割することができる。

【００７３】

また、ハッシュ値計算部５０２は、数値が入力されるとその数値をハッシュ関数に与えることによりハッシュ値を計算し、数値の入力元に返す。なお、本実施の形態では、一方向性関数の一例としてハッシュ関数を用い、その出力値をハッシュ値としているが、一方向性関数であれば、ハッシュ関数に限らず、他の方式や他の関数であってもよい。本実施の形態では、代表例としてハッシュ関数を用いる。

【００７４】

ハッシュ値列生成部５０３は、ハッシュ値計算部５０２と連携して分割単位となる各部分データのハッシュ値からなるハッシュ値列を生成する。具体的には、たとえば、部分データを既存の電子署名技術により数値化し、ハッシュ値計算部５０２に与える。そして、ハッシュ値計算部５０２により算出されたハッシュ値をハッシュ値計算部５０２から受け取ることにより、ハッシュ値列を生成する。なお、生成されるハッシュ値列のハッシュ値数をｎとする。

【００７５】

ここで、ある部分データd_iのハッシュ値をh_iとし、既存の電子署名技術により生成される部分データd_iのハッシュパラメータをe_iとすると、部分データd_iのハッシュ値h_iは下記式（１）によりあらわされる。

【００７６】

$$h_i = H(e_i) \cdots (1)$$

【００７７】

ここで、Ｈ（）はハッシュ関数である。また、部分データd_iについて乱数r_iが与えられたとすると、部分データd_iのハッシュ値h_iは下記式（２）によりあらわされる。

【００７８】

$$h_i = H(e_i \| r_i) \cdots (2)$$

【００７９】

ここで、ｘ‖ｙは、前の数値ｘの末尾と後の数値ｙの先頭との結合を示す。具体的には

、たとえば、ｘ＝１２３、ｙ＝４５６である場合、ｘ‖ｙ＝１２３４５６となる。

【００８０】

ハッシュ値二分木生成部５０４は、ハッシュ値計算部５０２と連携して、ハッシュ値列生成部５０３によって生成されたハッシュ値列をリーフ群とするハッシュ値二分木Ｔｈを生成する。リーフの数はハッシュ値の数ｎと同数である。

【００８１】

ハッシュ値二分木Ｔｈでは、ノード（リーフやルート含む）が階層構造化される。具体的には、下位の分岐先のノードを示すハッシュ値をh_a，h_bとした場合、分岐元のノードとなるハッシュ値$h_{a,b}$は、下記式（３）によりあらわされる。

【００８２】

$h_{a,b}=H(h_a \| h_b)$・・・（３）

【００８３】

ハッシュ値二分木生成部５０４は、下位の分岐先のノードを示すハッシュ値h_a，h_bをハッシュ値計算部５０２に与え、ハッシュ値計算部５０２は$h_{a,b}$を計算して、二分木生成部２０１に返す。なお、ハッシュ値h_a，h_bは同一階層の隣り合うノードを示すハッシュ値である。なお、本実施の形態において、ハッシュ値の符号の添え字は、そのハッシュ値の二分木上の位置を特定する位置情報となる。抽出したり復元する際には、この位置情報を手掛かりとする。

【００８４】

ハッシュ値二分木生成部５０４では、部分データ数ｎ、すなわちリーフ数ｎが奇数か偶数かによって生成手法が異なる。

【００８５】

図６は、部分データ数ｎが奇数（ｎ＝５）である場合に生成されるハッシュ値二分木Ｔｈを示す説明図である。

【００８６】

図６において、各部分データd_1～d_5は、分割部５０１によりストリーミングデータＳＴをＧＯＰ単位で分割されたデータである。各ハッシュ値h_1～h_5は、各部分データd_1～d_5に対応するハッシュ値であり、ハッシュ値列｛h_1，h_2，h_3，h_4，h_5｝としてハッシュ値列生成部５０３によって生成される。このハッシュ値列｛h_1，h_2，h_3，h_4，h_5｝がハッシュ値二分木Ｔｈのリーフ群となる。

【００８７】

ハッシュ値二分木生成部５０４では、先頭のハッシュ値から順に、隣り合うハッシュ値とペアを組む。他のペアに属するハッシュ値をペアにすることはできない。図６では、先頭のハッシュ値から順に、ハッシュ値h_1，h_2がペアとなり、ハッシュ値h_3，h_4がペアとなる。ハッシュ値h_2，h_3はペアを構成しない。また、部分データ数ｎが奇数であるため、ハッシュ値h_5は孤立する。

【００８８】

リーフを１段目のハッシュ値とした場合、その分岐元となる上位のハッシュ値を２段目のハッシュ値として、上記式（３）を用いて生成する。上記式（３）により、ハッシュ値h_1，h_2の分岐元ノードとなるハッシュ値$h_{1,2}$が生成され、ハッシュ値h_3，h_4の分岐元ノードとなるハッシュ値$h_{3,4}$が生成される。３段目のハッシュ値$h_{1,4}$も同様に、２段目のハッシュ値$h_{1,2}$，$h_{3,4}$を用いて、上記式（３）により生成される。

【００８９】

分岐先ノードを示すハッシュ値$h_{1,4}$が単一となったため、そのハッシュ値$h_{1,4}$とリーフで孤立したハッシュ値h_5を用いて、上記式（３）により、分岐元ノードを示すハッシュ値$h_{1,5}$を生成する。ここで生成されたハッシュ値$h_{1,5}$がルート（ハッシュ値）h_Rとなる。そして、このハッシュ値二分木Ｔｈからルート（ハッシュ値）h_Rを抽出して電子署名処理部２０２に出力することで、署名者の電子署名S_Aが生成される。

【００９０】

なお、部分データ数ｎが偶数（ｎ＝４）である場合については、図６において部分デー

タd_5がないとした場合に相当する。すなわち、リーフ群となるハッシュ値列{h_1，h_2，h_3，h_4}となるため、３段目の分岐元ノードを示すハッシュ値$h_{1,4}$がルートハッシュ値h_Rとなる。そして、このハッシュ値二分木ＴｈからルートＴｈ（ハッシュ値）h_Rを抽出して電子署名処理部２０２に出力することで、署名者の電子署名S_Aが生成される。

【００９１】

図７は、部分データ数ｎが奇数（ｎ＝７）である場合に生成されるハッシュ値二分木Ｔｈを示す説明図である。

【００９２】

図７では、リーフ以外の階層となる２段目においてハッシュ値の個数が奇数（３個）となる。この場合、先頭のハッシュ値から順に、ハッシュ値$h_{1,2}$とハッシュ値$h_{3,4}$とがペアとなり、１段目のハッシュ値h_7とともにハッシュ値$h_{5,6}$も孤立する。したがって、３段目においては、孤立したハッシュ値$h_{5,6}$，h_7から、上記式（３）を用いて分岐元ノードとなるハッシュ値$h_{5,7}$を生成する。

【００９３】

最後に、４段目において、ハッシュ値$h_{1,4}$，$h_{5,7}$から、上記式（３）を用いて分岐元ノードとなるハッシュ値$h_{1,7}$を生成する。ここで生成されたハッシュ値$h_{1,7}$がルート（ハッシュ値）h_Rとなる。そして、このハッシュ値二分木ＴｈからルートＴｈ（ハッシュ値）h_Rを抽出して電子署名処理部２０２に出力することで、署名者の電子署名S_Aが生成される。

【００９４】

なお、部分データ数ｎが偶数（ｎ＝６）である場合については、図７において部分データd_7がないとした場合に相当する。すなわち、リーフ群となるハッシュ値列{h_1，h_2，h_3，h_4，h_5，h_6}となるため、リーフのハッシュ値h_7は存在せず、３段目のハッシュ値$h_{5,7}$も生成されない。

【００９５】

したがって、２段目の孤立したハッシュ値$h_{5,6}$は、３段目のハッシュ値$h_{1,4}$と結合することで、上記式（３）を用いて分岐元ノードとなるハッシュ値$h_{1,6}$を生成する。このハッシュ値$h_{1,6}$がルートハッシュ値h_Rとなる。そして、このハッシュ値二分木ＴｈからルートＴｈ（ハッシュ値）h_Rを抽出して電子署名処理部２０２に出力することで、署名者の電子署名S_Aが生成される。

【００９６】

これにより，署名時における生成データ量は、従来のようにハッシュ値集合を使用しないため、ストリーミングデータＳＴの再生時間に依存せず、署名者の電子署名S_Aのデータ量にまで削減できる。

【００９７】

また、図５において、削除ハッシュ値二分木生成部５０５は、抽出時に実行される機能であり、ストリーミングデータＳＴのうち削除データのハッシュ値のみをリーフとする二分木（以下、「削除ハッシュ値二分木Ｔｄ」と称す。）を生成する。削除データとは、ストリーミングデータＳＴのうち抽出データ以外の部分データである。

【００９８】

そして、削除ハッシュ値二分木Ｔｄのルートとなるルートハッシュ値列L_{hRD}を抽出して、開示情報として出力する。なお、抽出データの個数をｃとすると、ルートハッシュ値列L_{hRD}に含まれるルートハッシュ値h_Rの個数の最大値の上限は$2 \cdot \log_2\{(n/c)/2\}$となる。

【００９９】

図８は、削除ハッシュ値二分木Ｔｄの生成例を示す説明図である。図８では、図５に示したストリーミングデータＳＴを用いて説明する。図８において、削除ハッシュ値二分木Ｔｄを生成する場合、抽出データのリーフは使用せず、削除データのリーフのみを使用する。まず、１段目において、リーフを示すハッシュ値h_1，h_2，h_5が選ばれる。

【０１００】

つぎに、２段目において、ハッシュ値二分木生成部５０４と同様、ハッシュ値計算部５０２と連携して二分木処理を実行する。すなわち、上記式（３）により、ハッシュ値h_1，h_2の分岐元ノードとなるハッシュ値$h_{1,2}$を生成する。

【０１０１】

２段目のハッシュ値$h_{1,2}$はもともと結合すべきハッシュ値が存在しないため孤立しており、これ以上上位階層のハッシュ値に変化しない。これにより、ハッシュ値h_1，h_2をリーフとし、ハッシュ値$h_{1,2}$をルートハッシュ値h_{RD1}とするルートハッシュ値二分木Ｔｄ１が生成される。

【０１０２】

また、ハッシュ値h_5も同様孤立しているため、これ以上上位階層のハッシュ値に変化しない。これにより、ハッシュ値h_5のみからなるルートハッシュ値二分木ＴｈＴｄ２が生成される。すなわち、ルートハッシュ値二分木Ｔｄ２では、ハッシュ値h_5がルートハッシュ値h_{RD2}となる。そして、ルートハッシュ値$h_{1,2}$，h_5からなるルートハッシュ値列L_{hRD}を、開示情報として出力する。

【０１０３】

また、図５において、乱数生成部５０６は、既存の乱数発生器により構成され、任意の乱数を生成する。スイッチ５０７は、既存のルート乱数と乱数生成部５０６によって生成された新規の乱数とを選択する。そして、選択された乱数を今回用いるルート乱数r_Rとして乱数二分木生成部５０８および抽出乱数二分木生成部５０９に出力するとともに、ルート乱数r_R自身も開示情報として出力する。ルート乱数r_Rとは、後述する乱数二分木Ｔｒのルートに位置する乱数である。この乱数生成部５０６およびスイッチ５０７によりルート乱数r_Rを取得することができる。

【０１０４】

また、乱数二分木生成部５０８は、ハッシュ値計算部５０２と連携して、乱数二分木Ｔｒを生成する。使用する乱数に必要な性質は，使用する乱数の部分集合から，それ以外の乱数が予測できないことである。そのため、ハッシュ値のデータ量削減とは逆に、ルート乱数r_Rからリーフ乱数へハッシュ関数を使用して乱数二分木Ｔｒを生成する。

【０１０５】

また、ハッシュ値二分木Ｔｈは完全二分木ではなかったが、乱数生成では、必要な乱数の量を超える完全二分木とする。なお、本実施の形態において、乱数の符号の添え字は、その乱数の二分木上の位置を特定する位置情報となる。抽出したり復元する際には、この位置情報を手掛かりとする。

【０１０６】

乱数二分木生成部５０８はスイッチ５０７から出力されてくるルート乱数r_Rを基点として、リーフとなる乱数（リーフ乱数）まで算出して、階層構造化された乱数二分木Ｔｒを生成する。リーフ乱数の個数は、部分データ数ｎと一致するように生成する。ここで、乱数二分木Ｔｒの生成例について説明する。

【０１０７】

分岐先の２つのノードを示すハッシュ値の分岐元となる乱数を$r_{i,j}$とする。分岐先のノードを示す２つの乱数は、下記式（４），（５）によってあらわされる。

【０１０８】

【数１】

$$r_{i,\lfloor (i+j)/2 \rfloor} = H(r_{i,j} \| L) \quad \cdots(4)$$

上記式(4)中、$\lfloor a \rfloor$は、aのうち小数点以下の端数を切り下げることを意味する記号。

(例)$\lfloor 3.6 \rfloor = 3$

Lは一方の分岐先乱数$r_{i,\lfloor (i+j)/2 \rfloor}$の生成元となる数字列。

【０１０９】

【数２】

$$r_{\lceil (i+j)/2 \rceil, j} = H(r_{i,j} \parallel R) \quad \cdots(5)$$

上記式(5)中、$\lceil a \rceil$は、aのうち小数点以下の端数を切り上げることを意味する記号。

(例)$\lceil 3.6 \rceil = 4$

Rは他方の分岐先乱数$r_{\lceil (i+j)/2 \rceil, j}$の生成元となる数字列で、$L \neq R$。

【０１１０】

また、乱数二分木Ｔｒを生成する際、まず、ルート乱数ｒ$_{R}$がｒ$_{i,j}$となる。ここで、ｒ$_{i,j}$＝ｒ$_{R}$のとき、ｉは先頭の部分データの昇順番号と一致（ｉ＝１）する。

【０１１１】

一方、ｊは、部分データ数ｎから計算される。具体的には、下記式（６）によってあらわされる。

【０１１２】

【数３】

$$j = 2^{\lceil Log_2(n) \rceil} \quad \cdots(6)$$

【０１１３】

また、上記において、以下の条件Ａを満たした場合、上記式（４）で示した一方の分岐先乱数はリーフ乱数ｒ$_{j}$となる。

【０１１４】

【数４】

＜条件Ａ＞

上記式(4)において、$\lfloor (i+j)/2 \rfloor = i$である場合、

一方の分岐先乱数$r_{i,\lfloor (i+j)/2 \rfloor}$＝リーフ乱数$r_i$となる。

【０１１５】

また、上記において、以下の条件Ｂを満たした場合、上記式（５）で示した他方の分岐先乱数はリーフ乱数ｒ$_{j}$となる。

【０１１６】

【数５】

＜条件Ｂ＞

上記式(5)において、$\lceil (i+j)/2 \rceil = j$である場合、

他方の分岐先乱数$r_{\lceil (i+j)/2 \rceil, j}$はリーフ乱数$r_j$となる。

また、上記式(5)において、分岐先乱数$r_{\lceil (i+j)/2 \rceil, j}$が$\lceil (i+j)/2 \rceil > n$である場合、

分岐先乱数$r_{\lceil (i+j)/2 \rceil, j}$は生成しない。

【０１１７】

ここで、部分データ数ｎ＝５である場合について説明する。図９は、部分データ数ｎ＝５である場合に生成される乱数二分木Ｔｒを示す説明図である。図９では、図５に示したストリーミングデータＳＴを用いて説明する。図９において、ｊは下記式（７）のようになるので、１段目にルート乱数ｒ$_{R}$＝ｒ$_{1,8}$を設定する。

【０１１８】

【数６】

$$j = 2^{\lceil Log_2(5) \rceil} = 2^{\lceil 2.32193\cdots \rceil} = 2^3 = 8 \quad \cdots(7)$$

【０１１９】

つぎに、２段目において、ルート乱数$r_R = r_{1,8}$を分岐元乱数とする２つの分岐先乱数$r_{1,4}$，$r_{5,8}$を生成する。そして、３段目において、乱数$r_{1,4}$を分岐元乱数とする２つの分岐先乱数$r_{1,2}$，$r_{3,4}$を生成する。また、乱数$r_{5,8}$については、$r_{i,j} = r_{n,n+3}$に該当するため、一方の分岐先乱数となる乱数$r_{5,6}$のみ生成する。

【０１２０】

このあと、４段目において、乱数$r_{1,2}$を分岐元乱数とする２つの分岐先乱数$r_{1,1} = r_1$，$r_{2,2} = r_2$を生成する。乱数$r_{3,4}$についても同様に、乱数$r_{3,4}$を分岐元乱数とする２つの分岐先乱数$r_{3,3} = r_3$，$r_{4,4} = r_4$を生成する。

【０１２１】

また、乱数$r_{5,6}$については、$r_{i,j} = r_{n,n+1}$に該当するため、一方の分岐先乱数$r_{i,(i+j)/2}$となる乱数$r_{5,5} = r_5$のみ生成する。これにより、ルート乱数r_Rのみを保持しておくだけで、他のすべての乱数を生成することができる。このあと、上記式（２）を用いることで、ハッシュ値列生成部５０３により、乱数r_iを考慮したハッシュ値h_iを部分データd_iごとに生成することができる。

【０１２２】

なお、部分データ数ｎ＝４である場合については、図９において部分データd_5がないとした場合に相当する。すなわち、ルート乱数r_Rは乱数$r_{1,4}$となり、乱数$r_{1,8}$，乱数$r_{5,6}$，乱数r_5は生成されない。

【０１２３】

つぎに、部分データ数ｎ＝７である場合について説明する。図１０は、部分データ数ｎ＝７である場合に生成される乱数二分木Ｔｒを示す説明図である。図１０では、図７に示したストリーミングデータＳＴを用いて説明する。なお、図９と異なる箇所のみ説明する。

【０１２４】

３段目において、乱数$r_{5,8}$を分岐元乱数とする２つの分岐先乱数$r_{5,6}$，$r_{7,8}$を生成する。つぎに、４段目において、乱数$r_{5,6}$を分岐元乱数とする２つの分岐先乱数$r_{5,5} = r_5$，$r_{6,6} = r_6$を生成する。乱数$r_{7,8}$については、一方の分岐先乱数は下記式（８）のようになるため、もう一方の分岐先乱数は下記式（９）のように、乱数$r_{7,7} = r_7$のみ生成する。

【０１２５】

【数７】

$$\text{一方の分岐先乱数 } r_{\lceil (i+j)/2 \rceil, j} \text{において、} \lceil (i+j)/2 \rceil = 8 > 7 \quad \cdots(8)$$

$$\text{もう一方の分岐先乱数} r_{i, \lfloor (i+j)/2 \rfloor} \text{となる乱数} r_{7,7} = r_7 \quad \cdots(9)$$

【０１２６】

これにより、ルート乱数r_Rのみを保持しておくだけで、他のすべての乱数を生成することができる。このあと、上記式（２）を用いることで、ハッシュ値列生成部５０３により、乱数r_iを考慮したハッシュ値h_iを部分データd_iごとに生成することができる。

【０１２７】

なお、部分データ数ｎ＝６である場合については、図１０において部分データd_7がないとした場合も同様である。すなわち、乱数$r_{5,8}$から乱数$r_{7,8}$が生成されないため、乱数r_7は生成されない。

【０１２８】

また、図５において、抽出乱数二分木生成部５０９は、抽出時に実行される機能であり

、乱数二分木Ｔｒのうち、抽出データに対応する乱数のみをリーフとする乱数二分木Ｔｒ（以下、「抽出乱数二分木Ｔｒｅ」と称す。）を生成する。そして、抽出乱数二分木Ｔｒｅのルート乱数$ｒ_{RE}$のみからなるルート乱数列$Ｌ_{rRE}$を開示情報として出力する。

【０１２９】

図１１は、抽出乱数二分木Ｔｒｅの生成例を示す説明図である。図１１では、図５に示したストリーミングデータＳＴを用いて説明する。図１１において、抽出データ$ｄ_3$，$ｄ_4$に対応するリーフ乱数は、乱数$ｒ_3$，$ｒ_4$である。したがって、乱数二分木Ｔｒのうち乱数$ｒ_3$，$ｒ_4$のみをリーフとし、$ｒ_{3,4}$をルート乱数$ｒ_{RE}$とする乱数二分木Ｔｒを抽出する。このルート乱数$ｒ_{RE}$さえあれば、それより大きい乱数二分木Ｔｒがなくても抽出データ$ｄ_3$，$ｄ_4$に必要なリーフ乱数$ｒ_3$，$ｒ_4$を生成することができる。

【０１３０】

＜検証時＞

図１２は、検証時における二分木生成部２０１の機能的構成を示すブロック図である。分割部５０１は、図５に示した署名・抽出時と同じ機能である。すなわち、抽出データ列$Ｌ_{de}$が入力されると、所定単位の部分データに分割する。分割単位は、署名・抽出時と同一の単位とする。また、ハッシュ値計算部５０２は、数値が入力されるとその数値をハッシュ関数に与えることによりハッシュ値を計算し、数値の入力元に返す。

【０１３１】

ハッシュ値列生成部５０３は、ハッシュ値計算部５０２と連携して分割単位となる各抽出データのハッシュ値からなるハッシュ値列を生成する。具体的には、たとえば、抽出データを既存の電子署名技術により数値化し、ハッシュ値計算部５０２に与える。そして、ハッシュ値計算部５０２により算出されたハッシュ値をハッシュ値計算部５０２から受け取ることにより、ハッシュ値列を生成する。なお、ハッシュ値の具体的な生成方法は、図５に示した内容と同一である。

【０１３２】

また、乱数二分木生成部５０８は、図５に示した機能と同様な手法により、開示情報である抽出乱数二分木Ｔｒｅのルート乱数列$Ｌ_{rRE}$を取得して、乱数二分木Ｔｒを生成する。通常は、ルート乱数列$Ｌ_{rRE}$を構成するルート乱数は一つであるため、このルート乱数を基点として乱数二分木Ｔｒを生成する。そして、生成された乱数二分木Ｔｒのリーフ乱数列をハッシュ値列生成部５０３に出力する。

【０１３３】

ハッシュ値二分木生成部５０４は、ハッシュ値計算部５０２と連携して、ストリーミングデータＳＴのハッシュ値二分木Ｔｈを復元する。具体的には、ハッシュ値列生成部５０３によって生成された抽出データ列のハッシュ値列と、開示情報である削除ハッシュ値二分木Ｔｄのルートハッシュ値列$Ｌ_{RD}$とを用いて、元のハッシュ値二分木Ｔｈを復元する。ここで、ハッシュ値二分木Ｔｈの復元例について説明する。

【０１３４】

図１３は、ハッシュ値二分木Ｔｈの復元例（その１）を示す説明図である。図１３では、図５に示したストリーミングデータＳＴを用いて説明する。図１３において、ハッシュ値二分木Ｔｈを復元する場合、抽出データ列$Ｌ_{de}$を構成する各抽出データ$ｄ_3$，$ｄ_4$のハッシュ値$ｈ_3$，$ｈ_4$を生成する。ハッシュ値$ｈ_3$，$ｈ_4$を生成することにより、ハッシュ値$ｈ_3$，$ｈ_4$からハッシュ値$ｈ_{3,4}$を生成することができる。

【０１３５】

また、抽出時の開示情報である削除ハッシュ値二分木Ｔｄのルートハッシュ値列$Ｌ_{hRD}$を構成するルートハッシュ値$ｈ_{1,2}$，$ｈ_5$を取得する。ルートハッシュ値$ｈ_{1,2}$はストリーミングデータＳＴから削除された部分データ$ｄ_1$，$ｄ_2$のハッシュ値$ｈ_1$，$ｈ_2$を合成したハッシュ値である。

【０１３６】

したがって、計算によって得られたハッシュ値$ｈ_3$，$ｈ_4$と合成することができ、ハッシュ値$ｈ_{1,4}$を生成することができる。そして、孤立しているルートハッシュ値$ｈ_5$とハッシ

ュ値$h_{1,4}$とを合成することで、ハッシュ値$h_{1,5}$を生成することができる。このハッシュ値$h_{1,5}$がルートハッシュ値h_Rとなる。これにより、元のハッシュ値二分木Ｔｈを復元することができる。

【０１３７】

図１４は、ハッシュ値二分木Ｔｈの復元例（その２）を示す説明図である。図１４では、図７に示したストリーミングデータＳＴを用いて説明する。図１４において、ハッシュ値二分木Ｔｈを復元する場合、抽出データ列L_{de}を構成する各抽出データのハッシュ値h_3，h_4を生成する。ハッシュ値h_3，h_4を生成することにより、ハッシュ値h_3，h_4からハッシュ値$h_{3,4}$を生成することができる。

【０１３８】

また、抽出時の開示情報である削除ハッシュ値二分木Ｔｄのルートハッシュ値列L_{hRD}を構成するルートハッシュ値$h_{1,2}$，$h_{5,6}$，h_7を取得する。ルートハッシュ値$h_{1,2}$はストリーミングデータＳＴから削除された部分データd_1，d_2のハッシュ値h_1，h_2を合成したハッシュ値である。したがって、計算によって得られたハッシュ値h_3，h_4と合成することにより、ハッシュ値$h_{1,4}$を生成することができる。

【０１３９】

また、ルートハッシュ値$h_{5,6}$はストリーミングデータＳＴから削除された部分データd_5，d_6のハッシュ値h_5，h_6を合成したハッシュ値である。したがって、孤立しているルートハッシュ値h_7とルートハッシュ値$h_{5,6}$とを合成することにより、ハッシュ値$h_{5,7}$を生成することができる。

【０１４０】

このあと、同階層のハッシュ値$h_{1,4}$とハッシュ値$h_{5,7}$とを合成することにより、ハッシュ値$h_{1,7}$を生成することができる。このハッシュ値$h_{1,7}$がルートハッシュ値h_Rとなる。これにより、元のハッシュ値二分木Ｔｈを復元することができる。

【０１４１】

（署名処理手順）

つぎに、この発明の実施の形態にかかる電子署名装置１００による署名処理手順について説明する。図１５は、この発明の実施の形態にかかる電子署名装置１００による署名処理手順を示すフローチャートである。図１５において、まず、分割部５０１により、ストリーミングデータＳＴを読み込んで、所定の分割単位（たとえば、ＧＯＰ）に分割する（ステップＳ１５０１）。

【０１４２】

つぎに、乱数使用モードか否かを判断する（ステップＳ１５０２）。乱数使用モードである場合（ステップＳ１５０２：Ｙｅｓ）、乱数二分木生成部５０８による乱数二分木生成処理を実行する（ステップＳ１５０３）。一方、乱数使用モードでない場合（ステップＳ１５０２：Ｎｏ）、ステップＳ１５０４に移行する。

【０１４３】

つぎに、ステップＳ１５０４において、ハッシュ値列生成部５０３によりハッシュ値列生成処理を実行する（ステップＳ１５０４）。そして、ハッシュ値二分木生成部５０４により、ハッシュ値二分木Ｔｈを生成する（ステップＳ１５０５）。

【０１４４】

このあと、電子署名処理部２０２により、ハッシュ値二分木Ｔｈのルートハッシュ値h_Rを用いて署名者の電子署名S_Aを生成する（ステップＳ１５０６）。このあと、署名者の電子署名S_Aや必要に応じて生成されたルート乱数r_Rなどの開示情報を保存する（ステップＳ１５０７）。

【０１４５】

（乱数二分木生成処理手順）

つぎに、図１５に示した乱数二分木生成処理（ステップＳ１５０３）による乱数二分木生成処理手順について説明する。図１６は、図１５に示した乱数二分木生成処理（ステップＳ１５０３）による乱数二分木生成処理手順を示すフローチャートである。

【０１４６】
図１６において、インデックスｉ＝１、部分データ数ｎとし（ステップＳ１６０１）、部分データ数ｎにより、インデックスｊの値を設定する（ステップＳ１６０２）。ｊの値は、上述した５）から８）を基準に設定される。
【０１４７】
そして、乱数生成部５０６により乱数$r_{i,j}$を生成して、ルート乱数r_Rに設定する（ステップＳ１６０３）。このあと、このルート乱数r_Rを乱数集合Ｓｒに入れる（ステップＳ１６０４）。
【０１４８】
つぎに、乱数集合Ｓｒが空集合（Ｓｒ＝Φ）であるか否かを判断し（ステップＳ１６０５）、Ｓｒ≠Φである場合（ステップＳ１６０５：Ｎｏ）、乱数集合Ｓｒの中から任意の乱数$r_{i,j}$を抽出し、乱数集合Ｓｒから削除する（ステップＳ１６０６）。そして、分岐先乱数算出処理を実行し（ステップＳ１６０７）、ステップＳ１６０５に戻る。一方、ステップＳ１６０５において、Ｓｒ＝Φである場合（ステップＳ１６０５：Ｙｅｓ）、ハッシュ値列生成処理（ステップＳ１５０４）に移行する。
【０１４９】
（分岐先乱数算出処理手順）
つぎに、図１６に示した分岐先乱数算出処理（ステップＳ１６０７）による分岐先乱数算出処理手順について説明する。図１７は、図１６に示した分岐先乱数算出処理（ステップＳ１６０７）による分岐先乱数算出処理手順を示すフローチャートである。
【０１５０】
図１７において、上記式（４）に示したように、乱数$r_{i,j}$に対する乱数（下記式（１０）を参照）を算出する（ステップＳ１７０１）。
【０１５１】
【数８】

$$r_{i,\lfloor (i+j)/2 \rfloor} \qquad \cdots(10)$$

【０１５２】
そして、以下の式（１１）を満たしているか否かを判断する（ステップＳ１７０２）。
【０１５３】
【数９】

$$\lfloor (i+j)/2 \rfloor = i \qquad \cdots(11)$$

【０１５４】
式（１１）を満たしている場合（ステップＳ１７０２：Ｙｅｓ）、上記式（１０）の乱数をリーフ乱数r_iに設定して（ステップＳ１７０３）、ステップＳ１７０６に移行する。
【０１５５】
一方、ステップＳ１７０２において、上記式（１１）を満たしていない場合（ステップＳ１７０２：Ｎｏ）、上記式（１０）の乱数を乱数$r_{i,j}$の第１の分岐先乱数に設定する（ステップＳ１７０４）。そして、この第１の分岐先乱数を乱数集合Ｓｒに入れて（ステップＳ１７０５）、ステップＳ１７０６に移行する。
【０１５６】
ステップＳ１７０６において、ｉ＝ｎであるか否かを判断する（ステップＳ１７０６）。ｉ＝ｎである場合（ステップＳ１７０６：Ｙｅｓ）、上記式（５）に示したように、乱数$r_{i,j}$に対する乱数（下記式（１２）を参照）を算出する（ステップＳ１７０７）。一方、ｉ＝ｎでない場合（ステップＳ１７０６：Ｎｏ）、ｉ＝ｎ＋３であるか否かを判断する（ステップＳ１７０８）。
【０１５７】

【数１０】

$$r_{\lceil (i+j)/2 \rceil, j} \quad \cdots(12)$$

【０１５８】
ｉ＝ｎ＋３である場合（ステップＳ１７０８：Ｙｅｓ）、ステップＳ１６０５に移行する。たとえば、乱数$ｒ_{ｉ,ｊ}$が図９に示した乱数$ｒ_{5,8}$である場合、この処理のように、乱数$ｒ_{7,8}$は生成されない。一方、ｉ＝ｎ＋３でない場合（ステップＳ１７０８：Ｎｏ）、ｊ＝ｉ＋１であるか否かを判断する（ステップＳ１７０９）。

【０１５９】
ｊ＝ｉ＋１である場合（ステップＳ１７０９：Ｙｅｓ）、ステップＳ１６０５に移行する。たとえば、乱数$ｒ_{ｉ,ｊ}$が図９に示した乱数$ｒ_{5,6}$である場合、この処理のように、乱数$ｒ_{6,6}$は生成されない。一方、ｊ＝ｉ＋１でない場合（ステップＳ１７０９：Ｎｏ）、ステップＳ１７０７に移行する。

【０１６０】
また、ステップＳ１７０７のあと、以下の式（１３）を満たしているか否かを判断する（ステップＳ１７１０）。

【０１６１】
【数１１】

$$\lceil (i+j)/2 \rceil = j \quad \cdots(13)$$

【０１６２】
上記式（１３）を満たしている場合（ステップＳ１７１０：Ｙｅｓ）、上記式（１２）に示した乱数をリーフ乱数$ｒ_{ｊ}$に設定する（ステップＳ１７１１）。

【０１６３】
一方、上記式（１３）を満たしていない場合（ステップＳ１７１０：Ｎｏ）、上記式（１２）に示した乱数を乱数$ｒ_{ｉ,ｊ}$の第２の分岐先乱数に設定する（ステップＳ１７１２）。そして、この第２の分岐先乱数を乱数集合Ｓｒに入れて（ステップＳ１７１３）、ステップＳ１６０５に移行する。

【０１６４】
（ハッシュ値列生成処理手順）
つぎに、図１５に示したハッシュ値列生成処理（ステップＳ１５０４）によるハッシュ値列生成処理手順について説明する。図１８は、図１５に示したハッシュ値列生成処理（ステップＳ１５０４）によるハッシュ値列生成処理手順を示すフローチャートである。

【０１６５】
図１８において、インデックスｉ＝１、部分データ数ｎとし（ステップＳ１８０１）、乱数使用モードか否かを判断する（ステップＳ１８０２）。乱数使用モードでない場合（ステップＳ１８０２：Ｎｏ）、上記式（１）により、部分データ$ｄ_{ｉ}$のハッシュパラメータ$ｅ_{ｉ}$を算出する（ステップＳ１８０３）。一方、乱数使用モードである場合（ステップＳ１８０２：Ｙｅｓ）、上記式（２）により、部分データ$ｄ_{ｉ}$と乱数$ｒ_{ｉ}$とにより、ハッシュパラメータ$ｅ_{ｉ}$を算出する（ステップＳ１８０４）。

【０１６６】
そして、ハッシュパラメータ$ｅ_{ｉ}$をハッシュ関数Ｈ（）に与えることにより、ハッシュ値$ｈ_{ｉ}$を算出する（ステップＳ１８０５）。そして、ハッシュ値$ｈ_{ｉ}$をハッシュ値列$Ｌｈ_{1}$に追加する（ステップＳ１８０６）。

【０１６７】
このあと、ｉ＞ｎであるか否かを判断し（ステップＳ１８０７）、ｉ＞ｎでない場合（ステップＳ１８０７：Ｎｏ）、ｉを１つインクリメントして（ステップＳ１８０８）、ステップＳ１８０２に戻る。一方、ｉ＞ｎである場合（ステップＳ１８０７：Ｙｅｓ）、ステップＳ１５０５に移行する。

【０１６８】

（ハッシュ値二分木生成処理手順）

つぎに、図１５に示したハッシュ値二分木生成処理（ステップＳ１５０５）によるハッシュ値二分木生成処理手順について説明する。図１９は、図１５に示したハッシュ値二分木生成処理（ステップＳ１５０５）によるハッシュ値列生成処理手順を示すフローチャートである。

【０１６９】

図１９において、生成されるハッシュ値二分木Ｔｈの階層を示す段数ｍ＝１とし、ハッシュ値列$Ｌｈ_{m}$に含まれているハッシュ値の総数をＮとする（ステップＳ１９０１）。段数ｍ＝１のハッシュ値列$Ｌｈ_{m}$は、ハッシュ値列生成処理で生成されたハッシュ値列である。

【０１７０】

つぎに、ハッシュ値列$Ｌｈ_{m}$を読み込み（ステップＳ１９０２）、Ｎ＝１であるか否かを判断する（ステップＳ１９０３）。Ｎ＝１でない場合（ステップＳ１９０３：Ｎｏ）、ハッシュ値列$Ｌｈ_{m}$の先頭からハッシュ値を２つ抽出して、ハッシュ値列$Ｌｈ_{m}$から削除する（ステップＳ１９０４）。

【０１７１】

そして、上述した式（３）で示したように、抽出された２つのハッシュ値を結合してハッシュパラメータとしてハッシュ関数Ｈ（）に与えることにより、あらたなハッシュ値を算出する（ステップＳ１９０５）。そして、算出ハッシュ値を次の段のハッシュ値列$Ｌｈ_{m+1}$に追加する（ステップＳ１９０６）。

【０１７２】

そして、ハッシュ値列$Ｌｈ_{m}$内のハッシュ値の残存数ＮｈがＮｈ≧２であるか否かを判断する（ステップＳ１９０７）。Ｎｈ≧２である場合（ステップＳ１９０７：Ｙｅｓ）、ステップＳ１９０４に戻る。一方、Ｎｈ≧２でない場合（ステップＳ１９０７：Ｎｏ）、Ｎｈ＝１であるか否かを判断する（ステップＳ１９０８）。

【０１７３】

Ｎｈ＝１である場合（ステップＳ１９０８：Ｙｅｓ）、ハッシュ値列$Ｌｈ_{m}$内の残存ハッシュ値を孤立ハッシュ値$ｈ_{k}$としてメモリに保存して（ステップＳ１９０９）、ステップＳ１９１０に移行する。

【０１７４】

一方、Ｎｈ＝１でない場合（ステップＳ１９０８：Ｎｏ）、段数ｍを１つインクリメントし、つぎの段のハッシュ値列$Ｌｈ_{m}$のハッシュ値総数Ｎを半分（Ｎ／２）にして（ステップＳ１９１０）、ステップＳ１９０２に戻る。

【０１７５】

また、ステップＳ１９０３において、Ｎ＝１である場合（ステップＳ１９０３：Ｙｅｓ）、孤立ハッシュ値$ｈ_{k}$があるか否かを判断する（ステップＳ１９１１）。孤立ハッシュ値$ｈ_{k}$がある場合（ステップＳ１９１１：Ｙｅｓ）、ハッシュ値列$Ｌｈ_{m}$内の単一のハッシュ値と孤立ハッシュ値とを用いて、図６および図７に示したようにルートハッシュ値$ｈ_{R}$を算出する（ステップＳ１９１２）。一方、孤立ハッシュ値$ｈ_{k}$がない場合（ステップＳ１９１１：Ｎｏ）、ハッシュ値二分木Ｔｈがすでに完成されているため、ステップＳ１９１３に移行する。

【０１７６】

これにより、ハッシュ値二分木Ｔｈが生成される。このあと、ハッシュ値二分木Ｔｈをメモリに保存し（ステップＳ１９１３）、ルートハッシュ値$ｈ_{R}$を電子署名処理部２０２に出力して（ステップＳ１９１４）、ステップＳ１５０６に移行する。

【０１７７】

（データ抽出処理手順）

つぎに、この発明の実施の形態にかかる電子署名装置１００によるデータ抽出処理手順について説明する。図２０は、この発明の実施の形態にかかる電子署名装置１００によるデータ抽出処理手順を示すフローチャートである。図２０において、まず、抽出部３０１

により、ストリーミングデータＳＴの抽出範囲を示す抽出情報を待ち受ける（ステップＳ２００１：Ｎｏ）。

【０１７８】

抽出情報が入力された場合（ステップＳ２００１：Ｙｅｓ）、該当する部分データを抽出して、他の部分データを削除する（ステップＳ２００２）。そして、削除ハッシュ値二分木生成部５０５により、削除ハッシュ値二分木Ｔｄを生成し（ステップＳ２００３）、その中からルートハッシュ値列L_{hRD}を抽出する（ステップＳ２００４）。

【０１７９】

つぎに、乱数使用モードであるか否かを判断する（ステップＳ２００５）。乱数使用モードである場合（ステップＳ２００５：Ｙｅｓ）、抽出乱数二分木生成部５０９により抽出乱数二分木Ｔｒｅを生成して（ステップＳ２００６）、そのルート乱数列L_{rRE}を抽出する（ステップＳ２００７）。このあと、ステップＳ２００８に移行する。一方、乱数使用モードでない場合（ステップＳ２００５：Ｎｏ）、ステップＳ２００８に移行する。

【０１８０】

ステップＳ２００８において、電子署名処理部２０２により、抽出者の電子署名S_Bを生成する（ステップＳ２００８）。通常モードの場合は、ルートハッシュ値列L_{hRD}を用いて抽出者の電子署名S_Bを生成する。

【０１８１】

一方、乱数使用モードでは、ルートハッシュ値列L_{hRD}およびルート乱数列L_{rRE}を用いて抽出者の電子署名S_Bを生成する。いずれにしても、既存の署名技術により生成することができる。このあと、抽出データ列L_{de}、抽出者の電子署名S_B、ルートハッシュ値列L_{hRD}、ルート乱数列L_{rRE}、署名者の電子署名S_Aなどの開示情報を保存する（ステップＳ２００９）。

【０１８２】

（検証処理手順）

つぎに、この発明の実施の形態にかかる電子署名装置１００による検証処理手順について説明する。図２１は、この発明の実施の形態にかかる電子署名装置１００によるデータ検証処理手順を示すフローチャートである。

【０１８３】

図２１において、まず、抽出時の開示情報を取得して（ステップＳ２１０１）、分割部５０１により、抽出データ列L_{de}を署名時と同一の分割単位で分割する（ステップＳ２１０２）。そして、ハッシュ値列生成部５０３により、各抽出データのハッシュ値列を生成する（ステップＳ２１０３）。

【０１８４】

そして、削除ハッシュ値二分木Ｔｄのルートハッシュ値列L_{hRD}と、抽出データのハッシュ値とを用いて、ハッシュ値二分木生成部５０４により、ハッシュ値二分木Ｔｈを復元する（ステップＳ２１０４）。そして、復元されたハッシュ値二分木Ｔｈのルートハッシュ値h_Rを抽出し（ステップＳ２１０５）、抽出されたルートハッシュ値h_Rを用いて署名者／抽出者の電子署名S_A，S_Bの正当性を確認する検証処理を実行する（ステップＳ２１０６）。

【０１８５】

以上説明したように、本実施の形態によれば、署名・抽出時におけるデータ量を大幅に削減することができる。また、検証時では，開示されたストリーミングデータＳＴと消去部分のルートハッシュ値列L_{hRD}から原本のルートハッシュ値h_Rを計算し署名検証をおこなうことで、原本のストリーミングデータＳＴの一部が抽出されたことを確認できる。また、顕名署名の場合、ルートハッシュ値列L_{hRD}とその署名を検証することで，誰が消去したのかも確認可能である。

【０１８６】

このように、この発明の実施の形態によれば、ＰＩＡＴ技術による抽出されたストリーミングデータ（抽出データ列L_{de}）の原本性保証機能はそのままで、データ量を大幅に削

減することができる。

【０１８７】

また、本実施の形態では、完全二分木ではなく、一般形の二分木を使用している。そのため上述した特許文献１のようなダミーデータＤを付加する必要がなく、原本情報のダミーデータＤへの偽造が不可能である。さらに、ダミーデータＤを保持したり、ダミーデータＤのハッシュ値を計算したりする必要もないため、計算量も削減することができる。

【０１８８】

また、本実施の形態では、ルート乱数r_Rさえ保持しておくだけで、乱数二分木Ｔｒを復元することができるため、乱数二分木Ｔｒを構成するすべての乱数を保持しなくてもよい。したがって、データ量の削減を図ることができる。

【０１８９】

また、検証処理では、開示されたルート乱数列L_{rRE}から、乱数二分木構成のアルゴリズムを使用して、抽出データ列L_{de}を復元することができる。この抽出データ列L_{de}と抽出データ列L_{de}，およびルートハッシュ値列L_{hRD}から原本のルートハッシュ値r_Rを計算し署名検証をおこなうことで，原本のストリーミングデータＳＴの一部が抽出されたことを確認できる。また、顕名抽出の場合、ルートハッシュ値列L_{hRD}とその署名者の電子署名S_Aを検証することで、誰が消去したのかも確認することができる。

【０１９０】

このように、本実施の形態では、長時間の動画像や音声データなどのストリーミングデータＳＴに対してＰＩＡＴ署名を行っても、必要な署名データ量をデータ長のｌｏｇオーダで削減可能であり、実用性が高い電子署名をおこなうことができる。図２９は、動画像のストリーミングデータにＰＩＡＴ署名を適用する場合の削減効果を示す図表である。図２９において、計算量のＨはハッシュ値の計算回数、Ｒは乱数生成回数を示す。

【０１９１】

以上説明したように、本実施の形態によれば、動画像や音声などの署名対象となるストリーミングデータの原本性の保証、署名対象からのプライバシー保護可能なデータ抽出、および署名関連データの大幅なデータ量削減を実現することができるという効果を奏する。

【０１９２】

なお、本実施の形態で説明した電子署名方法は、予め用意されたプログラムをパーソナル・コンピュータやワークステーション等のコンピュータで実行することにより実現することができる。このプログラムは、ハードディスク、フレキシブルディスク、ＣＤ－ＲＯＭ、ＭＯ、ＤＶＤ等のコンピュータで読み取り可能な記録媒体に記録され、コンピュータによって記録媒体から読み出されることによって実行される。またこのプログラムは、インターネット等のネットワークを介して配布することが可能な伝送媒体であってもよい。

【０１９３】

また、本実施の形態で説明した電子署名装置１００は、スタンダードセルやストラクチャードＡＳＩＣ（Ａｐｐｌｉｃａｔｉｏｎ Ｓｐｅｃｉｆｉｃ Ｉｎｔｅｇｒａｔｅｄ Ｃｉｒｃｕｉｔ）などの特定用途向けＩＣ（以下、単に「ＡＳＩＣ」と称す。）やＦＰＧＡなどのＰＬＤ（Ｐｒｏｇｒａｍｍａｂｌｅ Ｌｏｇｉｃ Ｄｅｖｉｃｅ)によっても実現することができる。具体的には、たとえば、上述した電子署名装置１００の機能的構成をＨＤＬ記述によって機能定義し、そのＨＤＬ記述を論理合成してＡＳＩＣやＰＬＤに与えることにより、電子署名装置１００を製造することができる。

【０１９４】

（付記１）コンピュータを、

動画像または音声に関するストリーミングデータを所定単位の複数の部分データに分割する分割手段、

前記分割手段によって得られた各部分データに関する数値を一方向性関数に与えることにより、前記複数の部分データに関する前記一方向性関数の出力値列を生成する出力値列生成手段、

前記出力値列生成手段によって生成された出力値列のみをリーフとする前記ストリーミングデータに関する二分木を生成する二分木生成手段、
前記二分木生成手段によって生成された二分木のルートを示す出力値を用いて、前記ストリーミングデータに対する署名者の電子署名を生成する電子署名生成手段、
として機能させることを特徴とする電子署名プログラム。

【０１９５】

（付記２）前記コンピュータを、
前記ストリーミングデータの中から任意の部分データ列を抽出し、残余の部分データを削除する抽出手段として機能させ、
前記二分木生成手段は、
前記ストリーミングデータに関する二分木の中から、前記抽出手段によって削除された部分データ（以下、「削除部分データ」という）に関する出力値のみをリーフとする前記削除部分データに関する二分木を抽出し、当該二分木のルートを示す出力値列を前記抽出手段によって抽出された部分データ列とともに出力することを特徴とする付記１に記載の電子署名プログラム。

【０１９６】

（付記３）前記電子署名生成手段は、
前記削除部分データに関する二分木のルートを示す出力値列を用いて、前記抽出手段によって抽出された部分データ列を抽出した抽出者の電子署名を生成することを特徴とする付記２に記載の電子署名プログラム。

【０１９７】

（付記４）前記分割手段は、
前記部分データ列を前記所定単位の複数の部分データに分割し、
前記出力値列生成手段は、
前記分割手段によって前記部分データ列から分割された各部分データに関する数値を前記一方向性関数に与えることにより、前記複数の部分データに関する前記一方向性関数の出力値列を生成し、
前記二分木生成手段は、
前記出力値列生成手段によって生成された前記複数の部分データに関する出力値列と、前記削除部分データに関する二分木と、に基づいて、前記ストリーミングデータに関する二分木を生成することを特徴とする付記２または３に記載の電子署名プログラム。

【０１９８】

（付記５）コンピュータを、
動画像または音声に関するストリーミングデータを所定単位の複数の部分データに分割する分割手段、
乱数を取得する取得手段、
前記取得手段によって取得された乱数をルート乱数とする二分木を、当該二分木のリーフ列を構成する乱数列が前記分割手段によって分割された部分データの個数と同数となるように生成する二分木生成手段、
前記二分木生成手段によって生成された二分木のリーフ乱数列と、前記複数の部分データとに基づいて、前記複数の部分データに関する出力値列を生成する出力値列生成手段、
として機能させることを特徴とする電子署名プログラム。

【０１９９】

（付記６）前記二分木生成手段は、
分岐元乱数と第１の数値とを用いた数値を一方向性関数に与えることにより、その出力値を第１の分岐先乱数として出力するとともに、前記分岐元乱数と前記第１の数値とは異なる第２の数値とを用いた数値を前記一方向性関数に与えることにより、その出力値を前記第１の分岐先乱数とは異なる第２の分岐先乱数として出力することにより、前記二分木を生成することを特徴とする付記５に記載の電子署名プログラム。

【０２００】

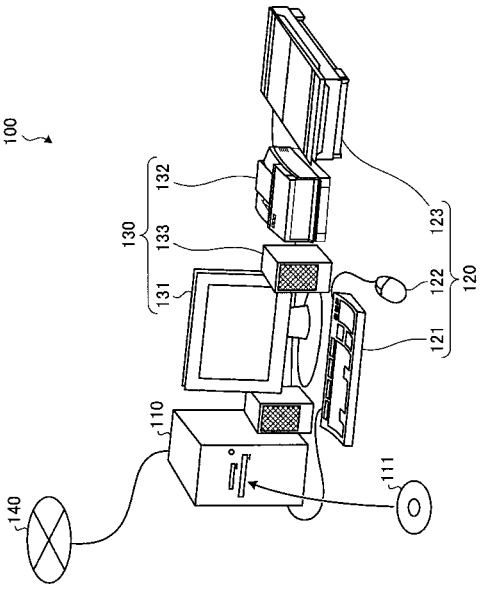
（付記７）前記二分木生成手段は、
前記分岐元乱数の位置情報と前記第２の分岐先乱数の位置情報とに基づいて、前記第２の分岐先乱数を生成しないことにより、前記二分木を生成することを特徴とする付記６に記載の電子署名プログラム。
【０２０１】
（付記８）前記コンピュータを、
前記ストリーミングデータの中から任意の部分データ列を抽出し、残余の部分データを削除する抽出手段として機能させ、
前記分割手段は、
前記部分データ列を前記所定単位の複数の部分データに分割し、
前記二分木生成手段は、
前記二分木の中から、前記分割手段によって分割された複数の部分データに対応する乱数列のみをリーフ乱数列とする二分木を抽出して、前記部分データ列とともに出力することを特徴とする付記５～７のいずれか一つに記載の電子署名プログラム。
【０２０２】
（付記９）付記１～８のいずれか一つに記載の電子署名プログラムを記録した前記コンピュータに読み取り可能な記録媒体。
【０２０３】
（付記１０）動画像または音声に関するストリーミングデータを所定単位の複数の部分データに分割する分割手段と、
前記分割手段によって得られた各部分データに関する数値を一方向性関数に与えることにより、前記複数の部分データに関する前記一方向性関数の出力値列を生成する出力値列生成手段と、
前記出力値列生成手段によって生成された出力値列のみをリーフとする前記ストリーミングデータに関する二分木を生成する二分木生成手段と、
前記二分木生成手段によって生成された二分木のルートを示す出力値を用いて、前記ストリーミングデータに対する署名者の電子署名を生成する電子署名生成手段と、
を備えることを特徴とする電子署名装置。
【０２０４】
（付記１１）動画像または音声に関するストリーミングデータを所定単位の複数の部分データに分割する分割手段と、
乱数を取得する取得手段と、
前記取得手段によって取得された乱数をルート乱数とする二分木を、当該二分木のリーフ列を構成する乱数列が前記分割手段によって分割された部分データの個数と同数となるように生成する二分木生成手段と、
前記二分木生成手段によって生成された二分木のリーフ乱数列と、前記複数の部分データとに基づいて、前記複数の部分データに関する出力値列を生成する出力値列生成手段と、
を備えることを特徴とする電子署名装置。
【０２０５】
（付記１２）動画像または音声に関するストリーミングデータを所定単位の複数の部分データに分割する分割工程と、
前記分割工程によって得られた各部分データに関する数値を一方向性関数に与えることにより、前記複数の部分データに関する前記一方向性関数の出力値列を生成する出力値列生成工程と、
前記出力値列生成工程によって生成された出力値列のみをリーフとする前記ストリーミングデータに関する二分木を生成する二分木生成工程と、
前記二分木生成工程によって生成された二分木のルートを示す出力値を用いて、前記ストリーミングデータに対する署名者の電子署名を生成する電子署名生成工程と、
を含んだことを特徴とする電子署名方法。

【０２０６】
（付記１３）動画像または音声に関するストリーミングデータを所定単位の複数の部分データに分割する分割工程と、
乱数を取得する取得工程と、
前記取得工程によって取得された乱数をルート乱数とする二分木を、当該二分木のリーフ列を構成する乱数列が前記分割工程によって分割された部分データの個数と同数となるように生成する二分木生成工程と、
前記二分木生成工程によって生成された二分木のリーフ乱数列と、前記複数の部分データとに基づいて、前記複数の部分データに関する出力値列を生成する出力値列生成工程と、
を含んだことを特徴とする電子署名方法。
【産業上の利用可能性】
【０２０７】
以上のように、本発明にかかる電子署名プログラム、該プログラムを記録した記録媒体、電子署名装置、および電子署名方法は、動画像や音声などのストリーミングデータに対する電子署名に適している。
【図面の簡単な説明】
【０２０８】
【図１】この発明の実施の形態にかかる電子署名装置のハードウェア構成を示すブロック図である。
【図２】署名時における電子署名装置の機能的構成を示す説明図である。
【図３】抽出時における電子署名装置の機能的構成を示す説明図である。
【図４】検証時における電子署名装置の機能的構成を示す説明図である。
【図５】署名・抽出時における二分木生成部の機能的構成を示すブロック図である。
【図６】部分データ数ｎが奇数である場合に生成されるハッシュ値二分木を示す説明図である。
【図７】部分データ数ｎが奇数である場合に生成されるハッシュ値二分木を示す説明図である。
【図８】削除ハッシュ値二分木の生成例を示す説明図である。
【図９】部分データ数ｎ＝５である場合に生成される乱数二分木Ｔｒを示す説明図である。
【図１０】部分データ数ｎ＝７である場合に生成される乱数二分木Ｔｒを示す説明図である。
【図１１】抽出乱数二分木の生成例を示す説明図である。
【図１２】検証時における二分木生成部の機能的構成を示すブロック図である。
【図１３】ハッシュ値二分木Ｔｈの復元例（その１）を示す説明図である。
【図１４】ハッシュ値二分木Ｔｈの復元例（その２）を示す説明図である。
【図１５】この発明の実施の形態にかかる電子署名装置による署名処理手順を示すフローチャートである。
【図１６】図１５に示した乱数二分木生成処理（ステップＳ１５０３）による乱数二分木生成処理手順を示すフローチャートである。
【図１７】図１６に示した分岐先乱数算出処理（ステップＳ１６０７）による分岐先乱数算出処理手順を示すフローチャートである。
【図１８】図１５に示したハッシュ値列生成処理（ステップＳ１５０４）によるハッシュ値列生成処理手順を示すフローチャートである。
【図１９】図１５に示したハッシュ値二分木生成処理（ステップＳ１５０５）によるハッシュ値列生成処理手順を示すフローチャートである。
【図２０】この発明の実施の形態にかかる電子署名装置によるデータ抽出処理手順を示すフローチャートである。
【図２１】この発明の実施の形態にかかる電子署名装置によるデータ検証処理手順を示す

フローチャートである。
【図２２】ＰＩＡＴ技術の署名者用のアルゴリズムの概要を示す説明図である。
【図２３】ＰＩＡＴ技術の抽出者用のアルゴリズムの概要を示す説明図である。
【図２４】ＰＩＡＴ技術の検証者用のアルゴリズムの概要を示す説明図である。
【図２５】動画像のストリーミングデータを署名対象とした特許文献１の署名技術における署名者による署名処理を示す説明図である。
【図２６】動画像ファイルを署名対象とした特許文献１の署名技術における抽出者による抽出処理を示す説明図である。
【図２７】部分動画像に付加された乱数を示す説明図である。
【図２８】アニメーションの動画像ファイルを示す説明図である。
【図２９】動画像のストリーミングデータにＰＩＡＴ署名を適用する場合の削減効果を示す図表である。
【符号の説明】
【０２０９】
１００　電子署名装置
２０１　二分木生成部
２０２　電子署名処理部
３０１　抽出部
５０１　分割部
５０２　ハッシュ値計算部
５０３　ハッシュ値列生成部
５０４　ハッシュ値二分木生成部
５０５　削除ハッシュ値二分木生成部
５０６　乱数生成部
５０７　スイッチ
５０８　乱数二分木生成部
５０９　抽出乱数二分木生成部

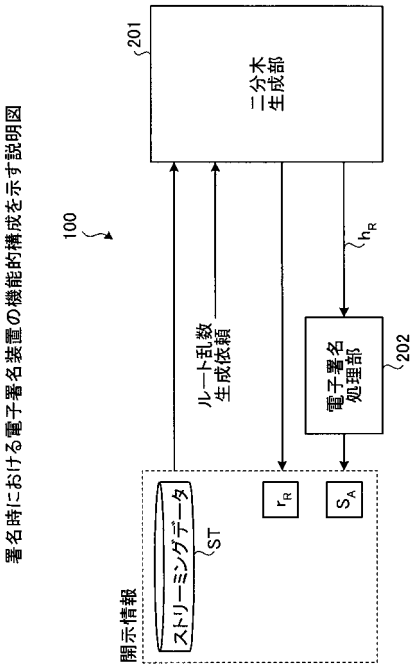
【図１】

この発明の実施の形態にかかる電子署名装置のハードウェア構成を示すブロック図



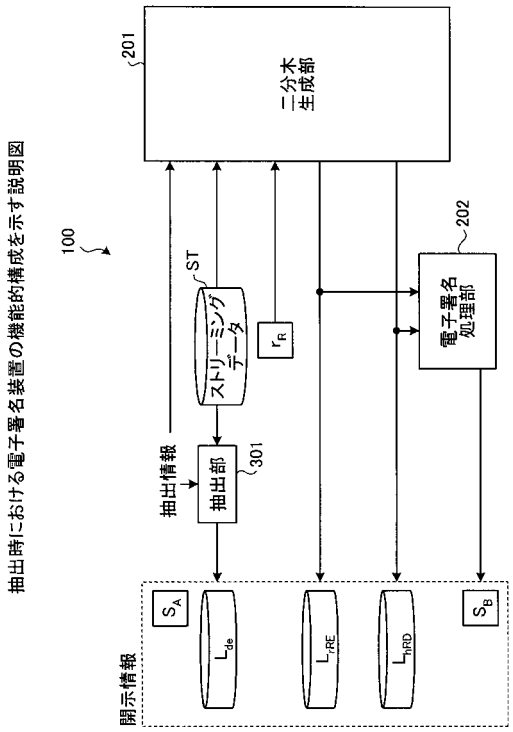
【図２】

署名時における電子署名装置の機能的構成を示す説明図



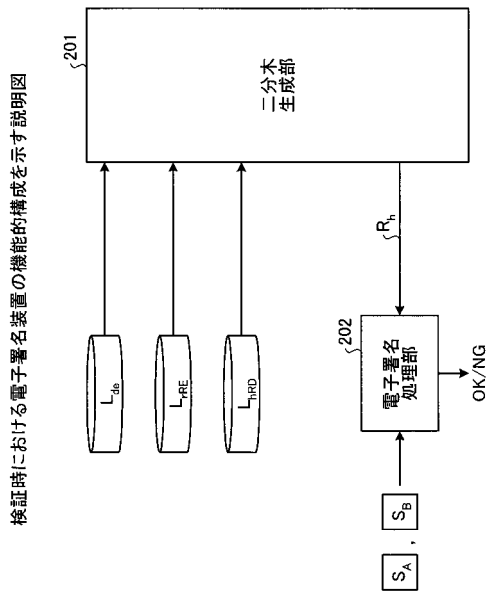
【図３】

抽出時における電子署名装置の機能的構成を示す説明図

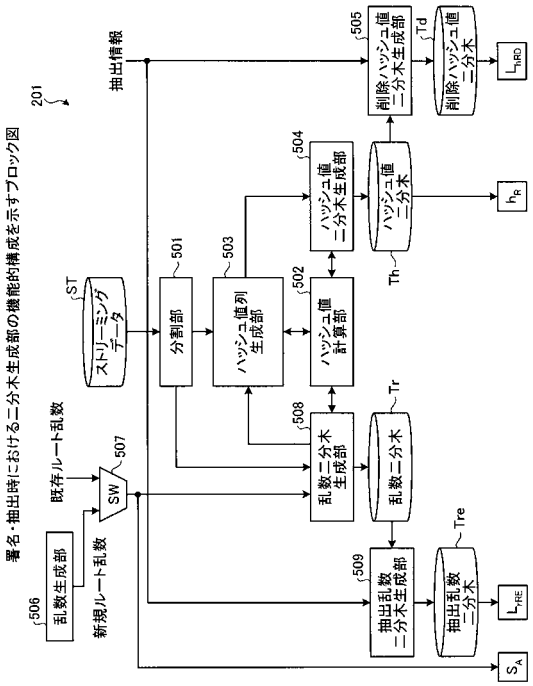


【図４】

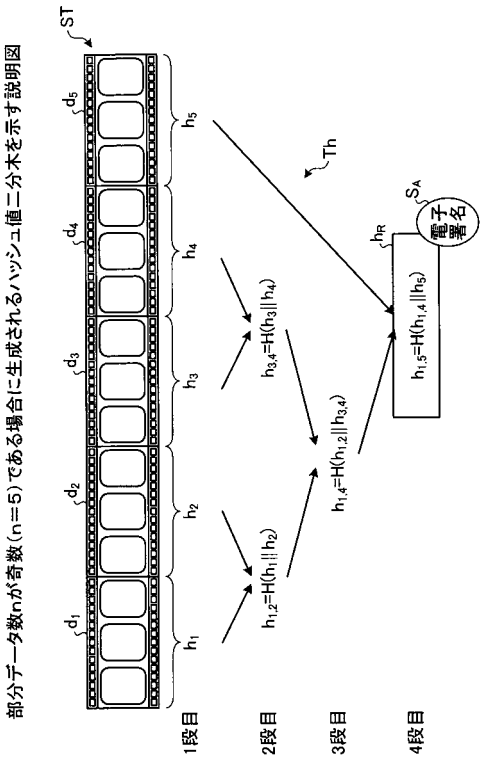
検証時における電子署名装置の機能的構成を示す説明図



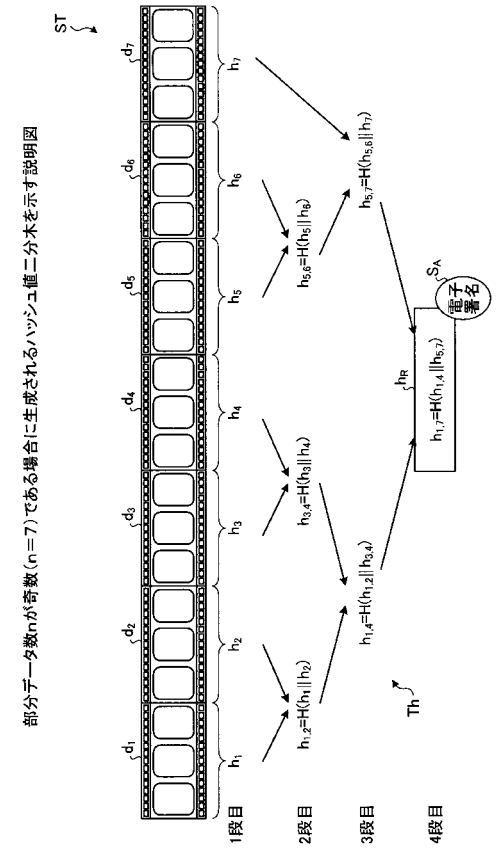
【図５】


署名・抽出時における二分木生成部の機能的構成を示すブロック図
201
抽出情報
ST
ストリーミングデータ
既存ルート乱数
新規ルート乱数
SW
507
506
乱数生成部
分割部
501
ハッシュ値列生成部
503
ハッシュ値計算部
502
ハッシュ値二分木生成部
504
削除ハッシュ値二分木生成部
505
乱数二分木生成部
508
乱数二分木
Tr
抽出乱数二分木生成部
509
抽出乱数二分木
Tre
ハッシュ値二分木
Th
削除ハッシュ値二分木
Td
h_R
L_{hRE}
L_{hRD}
S_A

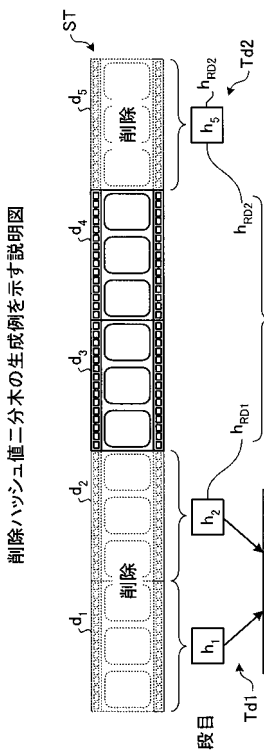
【図６】


部分データ数nが奇数(n＝5)である場合に生成されるハッシュ値二分木を示す説明図
ST
d_1
d_2
d_3
d_4
d_5
h_1
h_2
h_3
h_4
h_5
$h_{1,2}=H(h_1||h_2)$
$h_{3,4}=H(h_3||h_4)$
$h_{1,4}=H(h_{1,2}||h_{3,4})$
$h_{1,5}=H(h_{1,4}||h_5)$
Th
h_R
S_A
電子署名
1段目
2段目
3段目
4段目

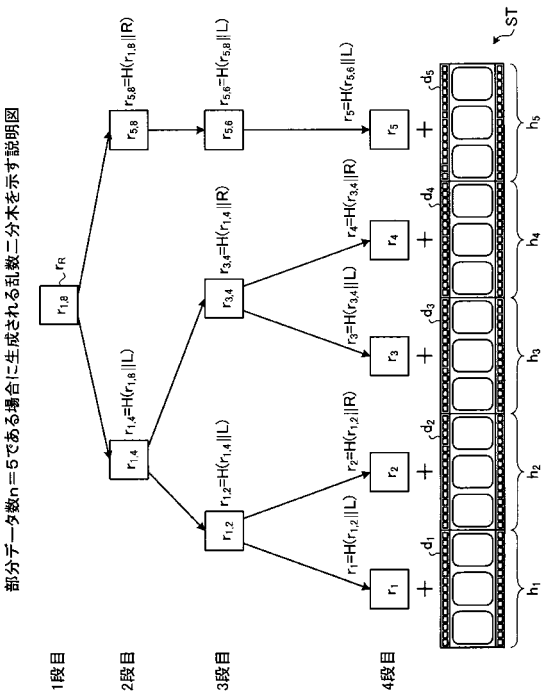
【図７】


部分データ数nが奇数(n＝7)である場合に生成されるハッシュ値二分木を示す説明図
ST
d_1
d_2
d_3
d_4
d_5
d_6
d_7
h_1
h_2
h_3
h_4
h_5
h_6
h_7
$h_{1,2}=H(h_1||h_2)$
$h_{3,4}=H(h_3||h_4)$
$h_{5,6}=H(h_5||h_6)$
$h_{1,4}=H(h_{1,2}||h_{3,4})$
$h_{5,7}=H(h_{5,6}||h_7)$
$h_{1,7}=H(h_{1,4}||h_{5,7})$
Th
h_R
S_A
電子署名
1段目
2段目
3段目
4段目

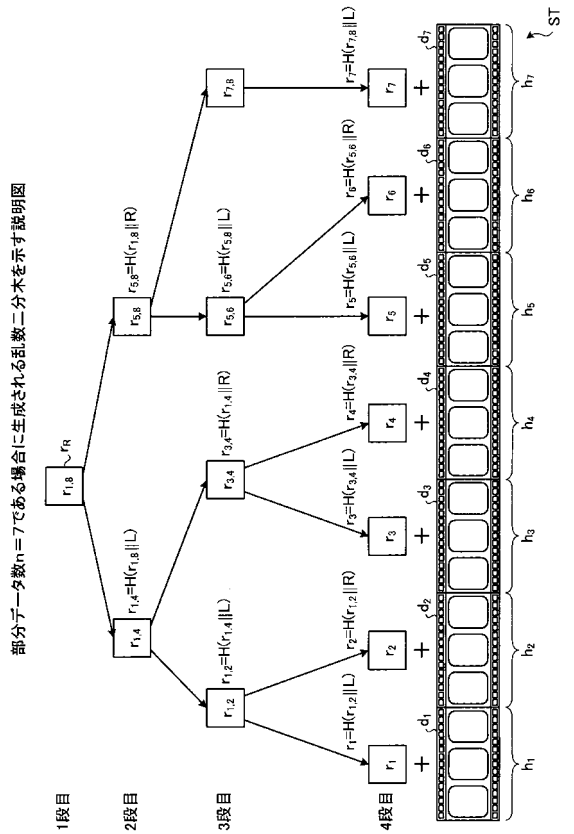
【図８】


削除ハッシュ値二分木の生成例を示す説明図
ST
d_1
d_2
d_3
d_4
d_5
削除
h_1
h_2
h_5
$h_{1,2}=H(h_1||h_2)$
h_{RD1}
h_{RD2}
h_{RD}
L_{hRD}
Td1
Td2
1段目
2段目

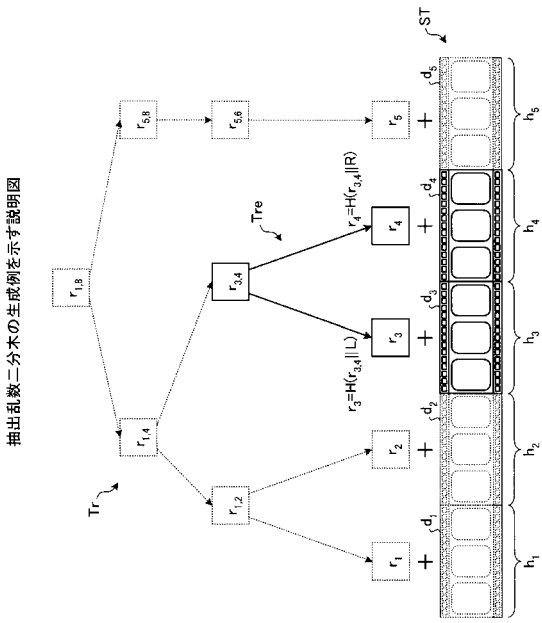
【図９】



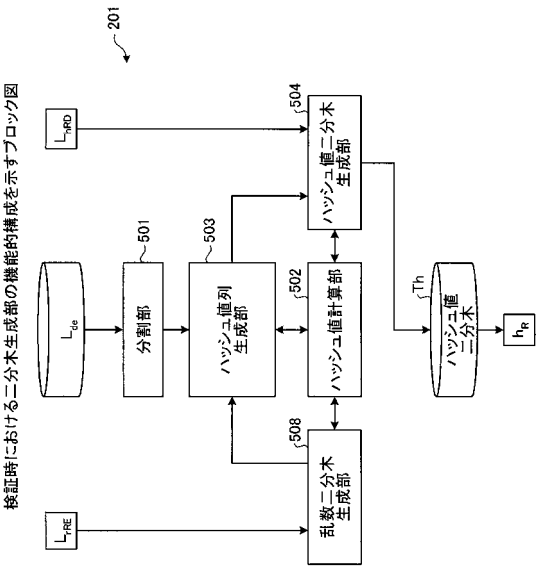
【図１０】



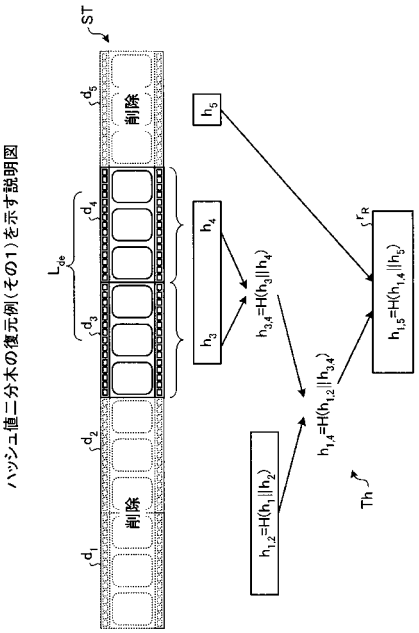
【図１１】



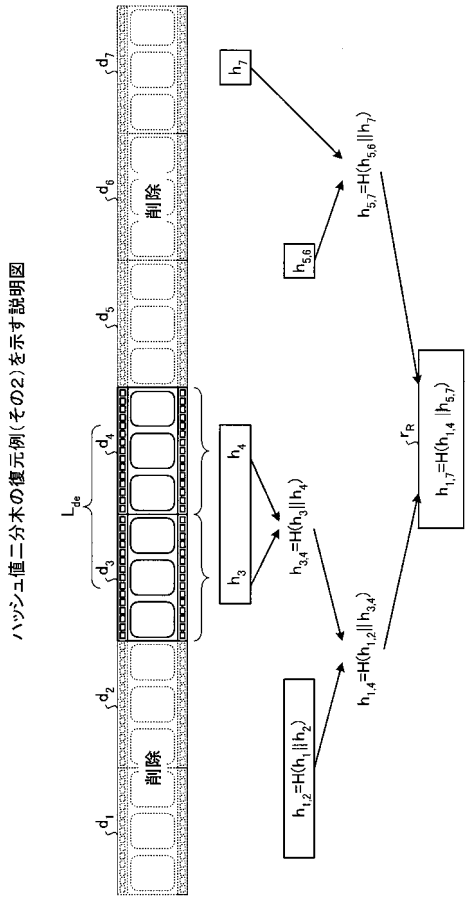
【図１２】



【図１３】

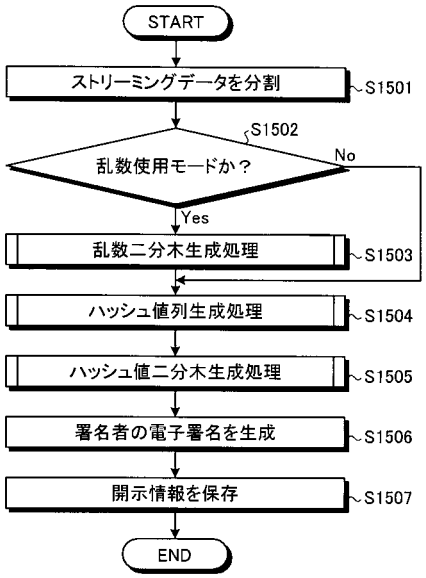


【図１４】



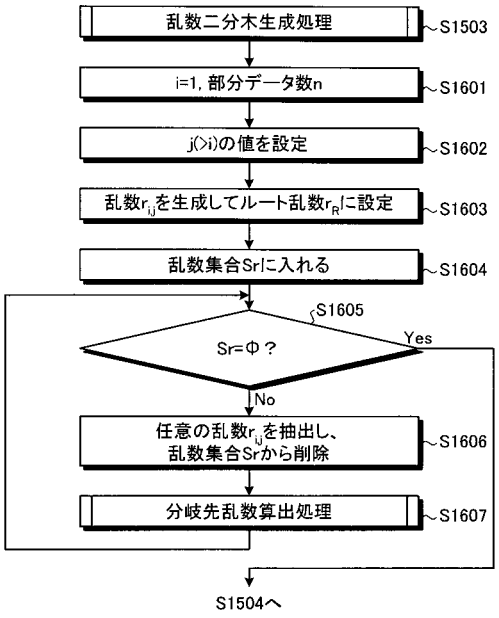
【図１５】

この発明の実施の形態にかかる電子署名装置による
署名処理手順を示すフローチャート



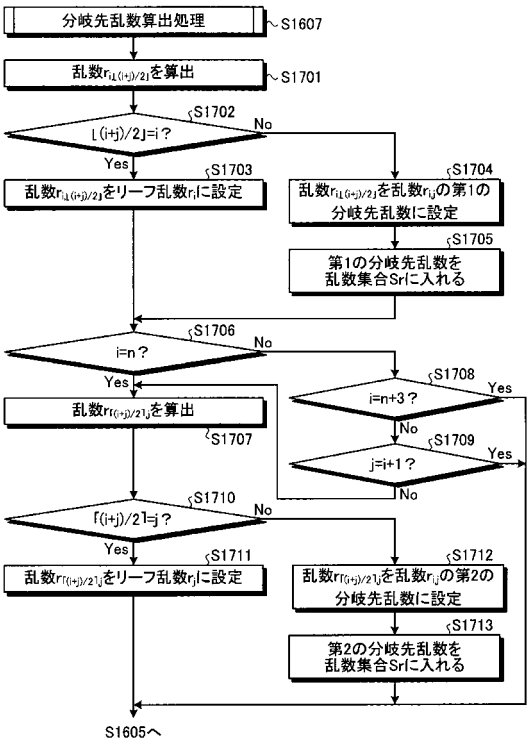
【図１６】

図15に示した乱数二分木生成処理(ステップS1503)による
乱数二分木生成処理手順を示すフローチャート



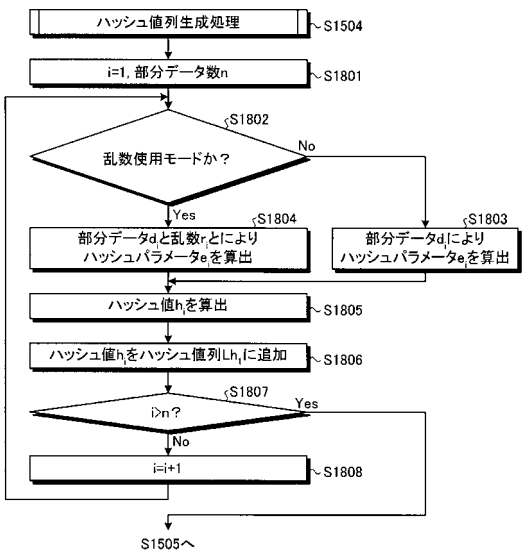
【図１７】

図16に示した分岐先乱数算出処理（ステップS1607）による分岐先乱数算出処理手順を示すフローチャート



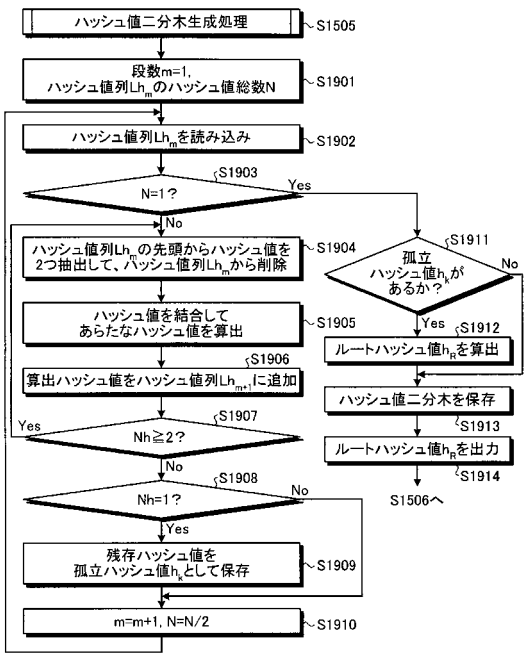
【図１８】

図15に示したハッシュ値列生成処理（ステップS1504）によるハッシュ値列生成処理手順を示すフローチャート



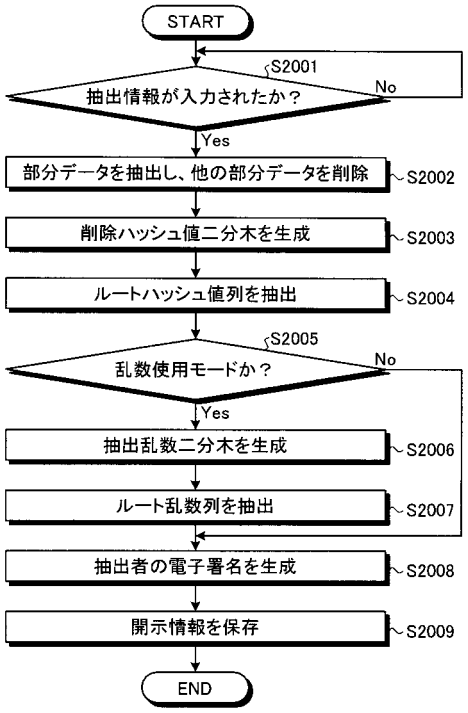
【図１９】

図15に示したハッシュ値二分木生成処理（ステップS1505）によるハッシュ値列生成処理手順を示すフローチャート



【図２０】

この発明の実施の形態にかかる電子署名装置によるデータ抽出処理手順を示すフローチャート

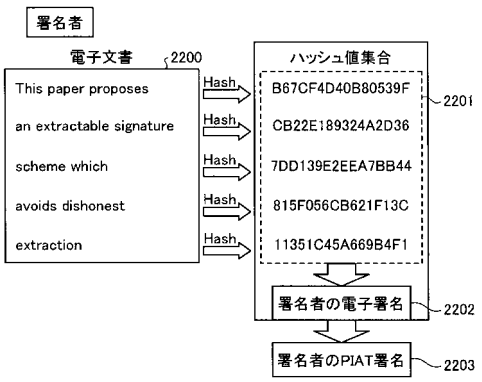


【図２１】

この発明の実施の形態にかかる電子署名装置による
データ検証処理手順を示すフローチャート

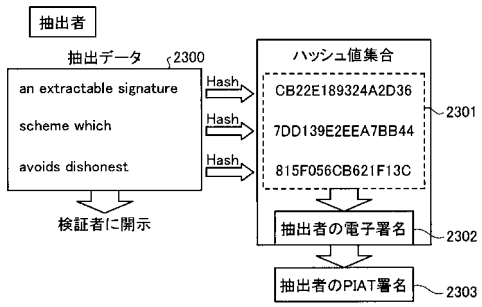
【図２２】

PIAT技術の署名者用のアルゴリズムの概要を示す説明図



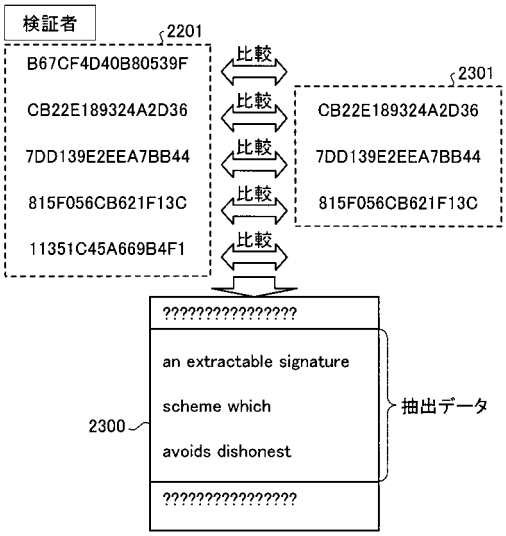
【図２３】

PIAT技術の抽出者用のアルゴリズムの概要を示す説明図

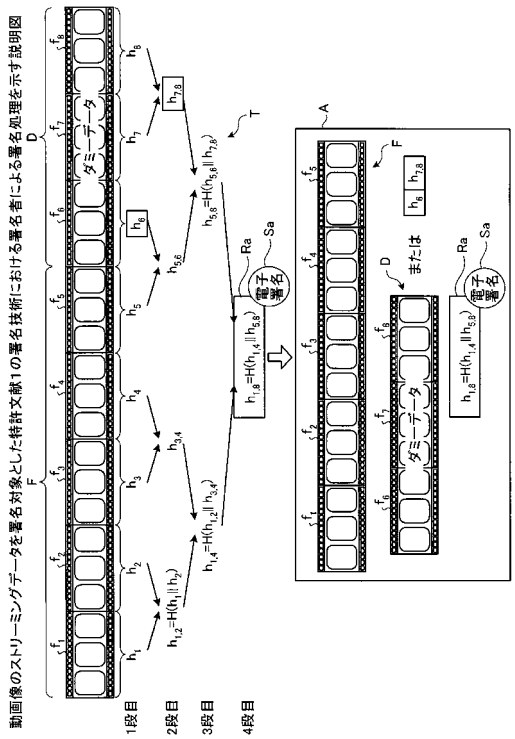


【図２４】

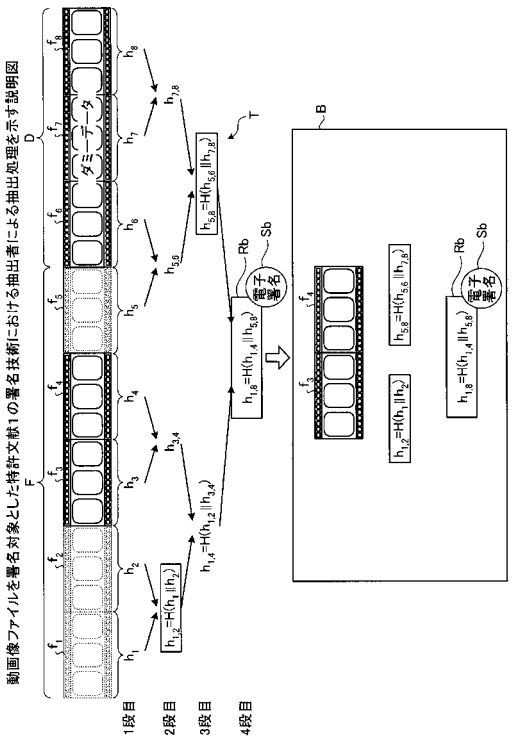
PIAT技術の検証者用のアルゴリズムの概要を示す説明図



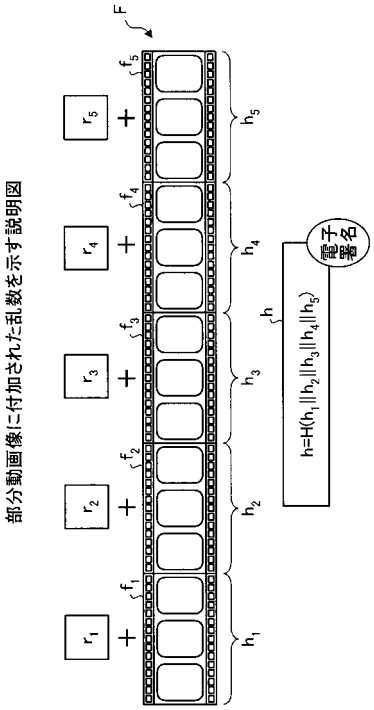
【図２５】



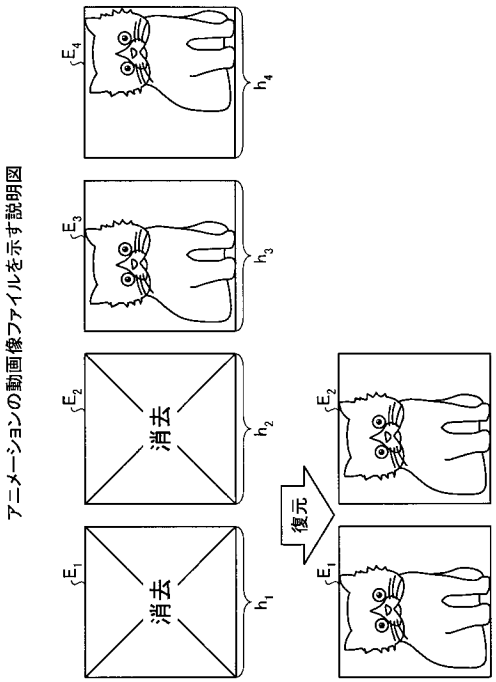
【図２６】



【図２７】



【図２８】



【図２９】

動画像のストリーミングデータにPIAT署名を適用する場合の削減効果を示す図表

データ	処理	PIAT		本発明	
		データ量	計算量	データ量	計算量
ハッシュ値	署名時	n個	nH	0個	≦2nH
	抽出時	m個	0	≦$2\log_2((m-n)/2)$個	≦2(n-m)H
乱数値	署名時	n個	nR	1個	≦R+2(n-1)H
	抽出時	m個	0	≦$2\log_2(m+2)-2$個	≦2mH

フロントページの続き

審査官　青木　重徳

(56)参考文献　特開２０００－２８６８３６（ＪＰ，Ａ）
特表２００１－５１９９３０（ＪＰ，Ａ）
特表２００５－５２６４５１（ＪＰ，Ａ）
特開２００６－２５３８２２（ＪＰ，Ａ）
Tetsuya Izu, Nobuyuki Kanaya, Masahiko Takenaka, and Takashi Yoshioka, “PIATS: A Partially Sanitizable Signature Scheme”, LNCS, ２００５年, Vol.3783, p.72-83
吉岡孝司, 武仲正彦, 伊豆哲也, “部分完全性保証技術ＰＩＡＴ：動画像・音声への適用”, ２００７年暗号と情報セキュリティシンポジウム, 日本, 社団法人電子情報通信学会, ２００７年　1月２３日, １Ｂ２　コンテンツ保護（１）, １Ｂ２－２, ｐ．１－６
吉岡孝司, 武仲正彦, “動画像・音声データに対する部分完全性保証技術ＰＩＡＴの実現（II）”, 情報処理学会研究報告（２００８－ＣＳＥＣ－４２）, 日本, 社団法人情報処理学会, ２００８年　7月17日, Ｖｏｌ．２００８, Ｎｏ．７１, ｐ．３７－４４
武仲正彦, 吉岡孝司, 金谷延幸, “検証者が署名者と墨塗り者を識別可能な電子文書の墨塗り方式”, コンピュータセキュリティシンポジウム２００４論文集, 日本, 社団法人情報処理学会, ２００４年１０月２０日, Ｖｏｌｕｍｅ　ＩＩ　ｏｆ　II, ｐ．４７５－４８０, 情報処理学会シンポジウムシリーズ, Ｖｏｌ．２００８, Ｎｏ．１１
吉岡孝司, 武仲正彦, “電子文書の訂正・流通を考慮した部分完全性保証方式の改良”, ２００５年暗号と情報セキュリティシンポジウム　ＳＣＩＳ２００５　予稿集付録ＣＤ－ＲＯＭ, 日本, ２００５年　1月２５日, ２Ｃ１　コンテンツセキュリティ－１, ２Ｃ１－１

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