

(19)日本国特許庁(JP)

## (12)特許公報(B2)

(11)特許番号

特許第7355247号

(P7355247)

(45)発行日 令和5年10月3日(2023.10.3)

(24)登録日 令和5年9月25日(2023.9.25)

(51)国際特許分類

H 0 4 L 9/32 (2006.01)

F I

H 0 4 L 9/32 2 0 0 B

請求項の数 13 (全45頁)

|             |                             |          |                                |
|-------------|-----------------------------|----------|--------------------------------|
| (21)出願番号    | 特願2022-532898(P2022-532898) | (73)特許権者 | 000005223                      |
| (86)(22)出願日 | 令和2年6月30日(2020.6.30)        |          | 富士通株式会社                        |
| (86)国際出願番号  | PCT/JP2020/025724           |          | 神奈川県川崎市中原区上小田中4丁目1番1号          |
| (87)国際公開番号  | WO2022/003841               | (74)代理人  | 100104190                      |
| (87)国際公開日   | 令和4年1月6日(2022.1.6)          |          | 弁理士 酒井 昭徳                      |
| 審査請求日       | 令和4年9月2日(2022.9.2)          | (72)発明者  | 小嶋 陸大                          |
|             |                             |          | 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  | 山本 大                           |
|             |                             |          | 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  | 矢崎 孝一                          |
|             |                             |          | 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】 署名制御方法、署名制御プログラム、および情報処理装置

## (57)【特許請求の範囲】

## 【請求項1】

複数の文書情報と、前記複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得し、

取得した前記複数の文書情報に基づいて、取得した前記複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成し、

生成した前記集約署名情報を、前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報と、前記複数の文書情報とに対応付けて出力する、

処理をコンピュータが実行することを特徴とする署名制御方法。

## 【請求項2】

取得した前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報に基づいて、前記集約公開鍵情報を生成する、

処理を前記コンピュータが実行することを特徴とする請求項1に記載の署名制御方法。

## 【請求項3】

前記集約署名情報を生成する処理は、

取得した前記複数の文書情報のそれぞれの文書情報に対応するハッシュ値に基づいて、前記集約署名情報を生成する、ことを特徴とする請求項1または2に記載の署名制御方法。

## 【請求項4】

生成した前記集約署名情報の正当性は、取得した前記複数の文書情報と、生成した前記集約署名情報と、生成した前記集約公開鍵情報とに基づいて検証可能である、ことを特徴

とする請求項 1～3 のいずれか一つに記載の署名制御方法。

【請求項 5】

複数の文書情報に対応付けられた、前記複数の文書情報に対応する複数の署名情報を集約した集約署名情報と、前記複数の文書情報に対応する複数の公開鍵情報を集約した集約公開鍵情報とを受け付けた場合、前記複数の文書情報と、受け付けた前記集約署名情報と、受け付けた前記集約公開鍵情報とに基づいて、受け付けた前記集約署名情報の正当性を検証する、

処理を前記コンピュータが実行することを特徴とする請求項 1～4 のいずれか一つに記載の署名制御方法。

【請求項 6】

前記複数の文書情報と対応付けて出力した前記集約署名情報が正当ではないと判定された場合、前記集約署名情報の集約元となる署名情報のうち、一部の署名情報を集約した集約署名情報と、前記一部の署名情報以外の署名情報を集約した集約署名情報との 2 つの集約署名情報を取得し、

取得した前記 2 つの集約署名情報のそれぞれの集約署名情報の正当性を検証した結果を出力する、

処理を前記コンピュータが実行することを特徴とする請求項 1～5 のいずれか一つに記載の署名制御方法。

【請求項 7】

取得した前記 2 つの集約署名情報の少なくともいずれかの集約署名情報が正当ではないと判定された場合、前記いずれかの集約署名情報の集約元となる署名情報のうち、一部の署名情報を集約した集約署名情報と、前記一部の署名情報以外の署名情報を集約した集約署名情報との 2 つの集約署名情報を新たに取得し、

新たに取得した前記 2 つの集約署名情報のそれぞれの集約署名情報の正当性を検証した結果を出力する、

処理を前記コンピュータが実行することを特徴とする請求項 6 に記載の署名制御方法。

【請求項 8】

取得した前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報と、取得した前記複数の文書情報のいずれの文書情報にも対応しない公開鍵情報とを集約した集約公開鍵情報を生成し、

前記いずれの文書情報にも対応しない公開鍵情報に基づいて、生成した前記集約署名情報を修正した修正署名情報を生成し、

生成した前記修正署名情報を、前記集約公開鍵情報と、取得した前記複数の文書情報とに対応付けて出力する、

処理を前記コンピュータが実行することを特徴とする請求項 1～7 のいずれか一つに記載の署名制御方法。

【請求項 9】

前記集約公開鍵情報を生成する処理は、

取得した前記複数の文書情報のそれぞれの文書情報に対応する署名情報を生成する際に用いられた秘密鍵情報に対応する公開鍵情報に基づいて、前記集約公開鍵情報を生成する、ことを特徴とする請求項 2 に記載の署名制御方法。

【請求項 10】

前記複数の文書情報は、第 1 の文書の複数の版のそれぞれの版を示す文書情報を含む、ことを特徴とする請求項 1～9 のいずれか一つに記載の署名制御方法。

【請求項 11】

前記出力する処理は、

シードと、前記第 1 の文書のそれぞれの版に対応するハッシュ値とから形成されるハッシュチェーンに基づいて生成されるハッシュ値を、前記集約署名情報に対応付けて出力する、ことを特徴とする請求項 10 に記載の署名制御方法。

【請求項 12】

10

20

30

40

50

複数の文書情報と、前記複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得し、

取得した前記複数の文書情報に基づいて、取得した前記複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成し、

生成した前記集約署名情報を、前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報と、前記複数の文書情報とに対応付けて出力する、

処理をコンピュータに実行させることを特徴とする署名制御プログラム。

#### 【請求項 13】

複数の文書情報と、前記複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得し、

取得した前記複数の文書情報に基づいて、取得した前記複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成し、

生成した前記集約署名情報を、前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報と、前記複数の文書情報とに対応付けて出力する、

制御部を有することを特徴とする情報処理装置。

#### 【発明の詳細な説明】

#### 【技術分野】

#### 【0001】

本発明は、署名制御方法、署名制御プログラム、および情報処理装置に関する。

#### 【背景技術】

#### 【0002】

従来、文書情報の改ざん防止のため、文書情報に対して、電子的な署名情報が付与されることがある。また、複数の秘密鍵を利用して、それぞれ異なる複数の文書情報に対して付与された複数の署名情報を、単一の署名情報に集約するアグリゲートシグネチャと呼ばれる技術がある。

#### 【0003】

先行技術としては、例えば、第1のデータ部分から第1の署名要素を生成し、生成した第1の署名要素と第2のデータ部分とから第1の中間署名要素を生成し、生成した第1の中間署名要素と第1の秘密鍵とから第2の署名要素を生成するものがある。

#### 【先行技術文献】

#### 【特許文献】

#### 【0004】

【文献】特開2013-118706号公報  
特開2019-212241号公報

#### 【発明の概要】

#### 【発明が解決しようとする課題】

#### 【0005】

しかしながら、従来技術では、署名情報の正当性を検証する際にかかるコストの増大化を招くという問題がある。例えば、アグリゲートシグネチャの技術では、生成された単一の署名情報の正当性を検証するため、複数の秘密鍵に対応する複数の公開鍵を管理しておくことになる。このため、公開鍵の数が増加するほど、署名情報の正当性を検証する際の処理コスト、および、公開鍵を管理する際の管理コストの増大化を招くことになる。

#### 【0006】

1つの側面では、本発明は、署名情報の正当性を検証する際にかかるコストの低減化を図ることを目的とする。

#### 【課題を解決するための手段】

#### 【0007】

1つの実施態様によれば、複数の文書情報と、前記複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得し、取得した前記複数の文書情報に基づいて、取得した前記複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生

10

20

30

40

50

成し、生成した前記集約署名情報を、前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報と、前記複数の文書情報とに対応付けて出力する署名制御方法、署名制御プログラム、および情報処理装置が提案される。

【発明の効果】

【0008】

一態様によれば、署名情報の正当性を検証する際にかかるコストの低減化を図ることが可能になる。

【図面の簡単な説明】

【0009】

【図1】図1は、実施の形態にかかる署名制御方法の一実施例を示す説明図である。

10

【図2】図2は、署名制御システム200の一例を示す説明図である。

【図3】図3は、情報処理装置100のハードウェア構成例を示すブロック図である。

【図4】図4は、制御情報管理テーブル400の記憶内容の一例を示す説明図である。

【図5】図5は、検証側端末207のハードウェア構成例を示すブロック図である。

【図6】図6は、署名制御システム200の機能的構成例を示すブロック図である。

【図7】図7は、作成者に対応する署名側端末201の具体的な機能的構成例を示すブロック図である。

【図8】図8は、承認者に対応する署名側端末202の具体的な機能的構成例を示すブロック図である。

【図9】図9は、検証側端末207の具体的な機能的構成例を示すブロック図である。

20

【図10】図10は、プライベートCAサーバ204の具体的な機能的構成例を示すブロック図である。

【図11】図11は、文書管理サーバ205の具体的な機能的構成例を示す説明図である。

【図12】図12は、署名制御システム200の動作の一例を示す説明図である。

【図13】図13は、署名制御システム200の動作の具体例を示す説明図（その1）である。

【図14】図14は、署名制御システム200の動作の具体例を示す説明図（その2）である。

【図15】図15は、署名制御システム200の動作の具体例を示す説明図（その3）である。

30

【図16】図16は、署名を検証する一例を示す説明図である。

【図17】図17は、補助パラメータdを設定する一例を示す説明図（その1）である。

【図18】図18は、補助パラメータdを設定する一例を示す説明図（その2）である。

【図19】図19は、集約署名を修正する一例を示す説明図（その1）である。

【図20】図20は、集約署名を修正する一例を示す説明図（その2）である。

【図21】図21は、署名制御システム200による効果を示す説明図である。

【図22】図22は、鍵生成処理手順の一例を示すフローチャートである。

【図23】図23は、文書作成処理手順の一例を示すフローチャートである。

【図24】図24は、文書承認処理手順の一例を示すフローチャートである。

【図25】図25は、文書決裁処理手順の一例を示すフローチャートである。

40

【図26】図26は、署名集約処理手順の一例を示すフローチャートである。

【図27】図27は、再検証処理手順の一例を示すフローチャートである。

【図28】図28は、別の署名集約処理手順の一例を示すフローチャートである。

【図29】図29は、無効化処理手順の一例を示すフローチャートである。

【発明を実施するための形態】

【0010】

以下に、図面を参照して、本発明にかかる署名制御方法、署名制御プログラム、および情報処理装置の実施の形態を詳細に説明する。

【0011】

（実施の形態にかかる署名制御方法の一実施例）

50

図1は、実施の形態にかかる署名制御方法の一実施例を示す説明図である。情報処理装置100は、文書情報に付与する署名情報を生成するためのコンピュータである。情報処理装置100は、サーバ、または、PC(Personal Computer)などである。

#### 【0012】

署名情報は、文書情報の改ざん防止のため、または、文書情報の内容の証明、確認、承認、あるいは、同意などの意思を表明するために利用されている。例えば、ビジネス上、法人が作成し、法人外に出される最終版の文書情報に、法人を代表する署名情報を付与し、法人外に出すことが考えられる。法人外の検証者は、最終版の文書情報に付与された、法人を代表する署名情報を検証し、最終版の文書情報が改ざんされておらず、正当であるのかを確認しようとする。EU圏では、法人を代表する署名情報として、e-Sealと呼ばれる署名情報が利用される傾向がある。

10

#### 【0013】

具体的には、法人内で、承認フローに従い、複数の承認者が、文書情報を改版しつつ、文書情報に個人の署名情報を付与していき、文書情報が最終版になった後、サーバが、最終版の文書情報に、法人を代表する署名情報を付与するユースケースが考えられる。かかるユースケースにおいて、サーバは、文書情報が最終版になるまでに、それぞれの承認者によって文書情報に付与されてきた署名情報がすべて正当であることが確認された場合のみ、文書情報に法人を代表する署名情報を付与するようにする。

#### 【0014】

しかしながら、かかるユースケースにおいて、不正な文書情報に対して、誤って法人を代表する署名情報が付与されてしまうことがある。不正な文書情報は、例えば、法人に対する攻撃者が作成した文書情報、改ざんされた文書情報、または、承認フローに反し、少なくともいずれかの承認者によって承認されていない文書情報などである。

20

#### 【0015】

例えば、かかるユースケースにおいて、最終版の文書情報に、法人を代表する署名情報を付与するサーバがクラッキングされた結果、不正な文書情報に対して、法人を代表する署名情報が付与されてしまう場合がある。この場合、法人外の検証者は、法人を代表する署名情報に基づいて、不正な文書情報を、誤って正当な文書情報であると判断してしまうことになる。また、この場合、法人外の検証者は、承認フローに従い、複数の承認者が、文書情報を承認し、文書情報に個人の署名情報を付与してきた過程を検証することができず、文書情報が正当であるか否かを検証することが困難である。

30

#### 【0016】

ここで、不正な文書情報に対して、誤って法人を代表する署名情報が付与されてしまうようなリスクについては、具体的には、総務省が開示する下記非特許文献1に記載された内容を参考にすることができる。

#### 【0017】

非特許文献1： 西山 晃、“日本におけるeシールのユースケースと課題”、[online]、2019年3月19日、Japan Trust Service Forum、[2020年6月26日検索]、インターネット<URL: [https://www.soumu.go.jp/main\\_content/000607659.pdf](https://www.soumu.go.jp/main_content/000607659.pdf)>

40

#### 【0018】

このため、承認フローに従い、複数の承認者が、文書情報に個人の署名情報を付与してきた過程を検証可能にし、かつ、正当な文書情報に対してのみ、正しく法人を代表する署名情報を付与可能にすることが望まれる。換言すれば、不正な文書情報に対して、誤って法人を代表する署名情報が付与されてしまうリスクを回避することが望まれる。

#### 【0019】

これに対し、アグリゲートシグネチャと呼ばれる技術が存在する。アグリゲートシグネチャは、例えば、複数の秘密鍵情報を利用して、それぞれ異なる複数の文書情報に対して

50

付与された複数の署名情報を、単一の署名情報に集約する技術である。ここで、アグリゲートシグネチャの実装方式は、例えば、BLS署名ベースの実装方式、または、格子暗号ベースの実装方式などがある。

#### 【0020】

このアグリゲートシグネチャを、上述したユースケースに適用し、複数の承認者が、それぞれ異なる版の文書情報に対して付与してきた署名情報を、単一の署名情報に集約しておき、最終版の文書情報に付与するという手法が考えられる。この手法によれば、法人外の検証者は、複数の秘密鍵情報に対応する複数の公開鍵情報に基づいて、単一の署名情報を検証することにより、承認フローに従い、複数の承認者が、文書情報に個人の署名情報を付与してきた過程を検証可能であると考えられる。数学的には、単一の署名情報を検証することが、集約元となる複数の署名情報を検証することに相当する。

10

#### 【0021】

一方で、この手法では、複数の秘密鍵情報に対応する複数の公開鍵情報を、法人外の検証者が参照可能に、公開用のストレージ装置に登録しておくことになる。ここで、承認者の数が増加すれば、ストレージ装置に登録しておく公開鍵情報の数も増加することになる。このため、公開鍵情報を管理する際にかかる管理コストの増大化を招くことになる。また、単一の署名情報を検証する際にかかる処理コストの増大化を招くことになる。

#### 【0022】

そこで、本実施の形態では、署名情報の正当性を検証する際に用いられる公開鍵情報の数の低減化を図ることにより、署名情報の正当性を検証する際にかかる管理コスト、および、処理コストの低減化を図ることができる署名制御方法について説明する。

20

#### 【0023】

図1において、情報処理装置100は、署名集約アルゴリズムを有する。署名集約アルゴリズムは、複数の署名情報を集約した1つの集約署名情報を生成する方法を規定する。情報処理装置100は、鍵集約アルゴリズムを有する。鍵集約アルゴリズムは、複数の公開鍵情報を集約した1つの集約公開鍵情報を生成する方法を規定する。

#### 【0024】

(1-1) 情報処理装置100は、複数の文書情報と、複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得する。文書情報は、例えば、契約書、または、ソースコードなどといった、複数人によって改版される文書情報である。図1の例では、情報処理装置100は、文書1～3と、文書1～3に対応する署名1～3とを取得する。

30

#### 【0025】

(1-2) 情報処理装置100は、取得した複数の文書情報に基づいて、取得した複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成する。図1の例では、情報処理装置100は、署名集約アルゴリズムに従って、文書1～3に基づいて、文書1～3に対応する署名1～3を集約した集約署名情報110を生成する。

#### 【0026】

(1-3) 情報処理装置100は、取得した複数の文書情報のそれぞれの文書情報に対応する公開鍵情報に基づいて、それぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報を生成する。図1の例では、情報処理装置100は、鍵集約アルゴリズムに従って、公開鍵束120に基づいて、公開鍵束120を集約した集約公開鍵情報121を生成する。

40

#### 【0027】

(1-4) 情報処理装置100は、生成した集約署名情報を、生成した集約公開鍵情報と、複数の文書情報とに対応付けて出力する。情報処理装置100は、集約署名情報を、集約公開鍵情報と、複数の文書情報との対応関係が特定可能に出力する。集約署名情報を、複数の文書情報に対応付けるとは、例えば、集約署名情報を、最終版の文書情報と、最終版以外の文書情報のハッシュ値とに対応付けることである。

#### 【0028】

図1の例では、情報処理装置100は、文書1～3と対応付けて、生成した集約署名情

50

報 1 1 0 と、生成した集約公開鍵情報 1 2 1 とを出力する。情報処理装置 1 0 0 は、具体的には、文書 1 ～ 3 と対応付けて、生成した集約署名情報 1 1 0 と、生成した集約公開鍵情報 1 2 1 とを、他のコンピュータで参照可能に出力する。他のコンピュータは、例えば、検証者に対応するコンピュータである。

#### 【 0 0 2 9 】

これにより、情報処理装置 1 0 0 は、検証者に対応するコンピュータで参照する公開鍵情報の数の低減化を図ることができ、公開鍵情報を管理する際にかかる管理コスト、および、公開鍵情報を利用する際にかかる処理コストの低減化を図ることができる。また、情報処理装置 1 0 0 は、検証者側で、集約署名情報に基づいて、集約署名情報の集約元となる複数の署名情報が生成された経過を検証可能にすることができる。このため、情報処理装置 1 0 0 は、検証者側で、不正な文書情報を、誤って正当な文書情報であると判断してしまう事態を防止することができる。

10

#### 【 0 0 3 0 】

( 1 - 5 ) 他のコンピュータは、1 鍵検証アルゴリズムを有する。他のコンピュータは、複数の文書情報に対応付けられた、集約署名情報と、集約公開鍵情報とを受け付ける。他のコンピュータは、例えば、複数の文書情報に対応付けられた、集約署名情報と、集約公開鍵情報とを、情報処理装置 1 0 0 から受信する。

#### 【 0 0 3 1 】

( 1 - 6 ) 他のコンピュータは、複数の文書情報と、受け付けた集約署名情報と、受け付けた集約公開鍵情報とに基づいて、受け付けた集約署名情報の正当性を検証する。他のコンピュータは、例えば、複数の文書情報のそれぞれの文書情報に対応するハッシュ値を取得する。そして、他のコンピュータは、例えば、1 鍵検証アルゴリズムに従って、取得したハッシュ値と、受け付けた集約署名情報と、受け付けた集約公開鍵情報とに基づいて、受け付けた集約署名情報の正当性を検証する。

20

#### 【 0 0 3 2 】

これにより、他のコンピュータは、集約署名情報に基づいて、集約署名情報の集約元となる複数の署名情報が生成された経過を検証することができる。このため、他のコンピュータは、不正な文書情報を、誤って正当な文書情報であると判断してしまう事態を回避することができる。

#### 【 0 0 3 3 】

ここでは、情報処理装置 1 0 0 が、それぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報を生成する場合について説明したが、これに限らない。例えば、情報処理装置 1 0 0 とは別に、それぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報を生成する生成装置が存在する場合があってもよい。この場合、生成装置が、検証者側に、集約公開鍵情報を提供する。情報処理装置 1 0 0 は、生成した集約署名情報を、集約公開鍵情報と対応することを特定可能に、検証者側に提供する。

30

#### 【 0 0 3 4 】

( 署名制御システム 2 0 0 の一例 )

次に、図 2 を用いて、図 1 に示した情報処理装置 1 0 0 を適用した、署名制御システム 2 0 0 の一例について説明する。

40

#### 【 0 0 3 5 】

図 2 は、署名制御システム 2 0 0 の一例を示す説明図である。図 2 において、署名制御システム 2 0 0 は、作成者に対応する署名側端末 2 0 1 と、承認者に対応する署名側端末 2 0 2 と、決裁者に対応する署名側端末 2 0 3 と、プライベート C A サーバ 2 0 4 と、文書管理サーバ 2 0 5 とを含む。さらに、署名制御システム 2 0 0 は、パブリック C A サーバ 2 0 6 と、検証者に対応する検証側端末 2 0 7 とを含む。

#### 【 0 0 3 6 】

署名制御システム 2 0 0 において、それぞれの署名側端末 2 0 1 ～ 2 0 3 と、プライベート C A サーバ 2 0 4 とは、有線または無線のネットワーク 2 1 0 を介して接続される。ネットワーク 2 1 0 は、例えば、LAN ( Local Area Network )、WA

50

N (Wide Area Network)、インターネットなどである。

【0037】

また、署名制御システム200において、それぞれの署名側端末201～203と、文書管理サーバ205とは、有線または無線のネットワーク210を介して接続される。また、署名制御システム200において、プライベートCAサーバ204と、パブリックCAサーバ206とは、有線または無線のネットワーク210を介して接続される。

【0038】

また、署名制御システム200において、文書管理サーバ205と、検証側端末207とは、有線または無線のネットワーク210を介して接続される。また、署名制御システム200において、パブリックCAサーバ206と、検証側端末207とは、有線または無線のネットワーク210を介して接続される。

10

【0039】

署名側端末201は、文書情報を作成する作成者によって用いられるコンピュータである。署名側端末201は、作成者の操作入力に基づき、新たな文書情報を生成する。署名側端末201は、秘密鍵情報と公開鍵情報との鍵情報ペアを生成する。署名側端末201は、生成した鍵情報ペアの秘密鍵情報に基づいて、生成した新たな文書情報に付与する署名情報を生成する。署名側端末201は、生成した新たな文書情報と、生成した署名情報とを対応付けて、文書管理サーバ205に送信する。署名側端末201は、生成した鍵情報ペアの公開鍵情報を、プライベートCAサーバ204に送信する。署名側端末201は、例えば、サーバ、PC、タブレット端末、または、スマートフォンなどである。

20

【0040】

署名側端末202は、文書情報を承認する承認者によって用いられるコンピュータである。署名側端末202は、承認者の操作入力に基づき、署名情報が付与された文書情報を、文書管理サーバ205から取得する。署名側端末202は、承認者の操作入力に基づき、取得した文書情報を改版し、新たな文書情報を生成する。署名側端末202は、秘密鍵情報と公開鍵情報との鍵情報ペアを生成する。署名側端末202は、生成した鍵情報ペアの秘密鍵情報に基づいて、生成した新たな文書情報に付与する署名情報を生成する。署名側端末202は、生成した新たな文書情報と、生成した署名情報とを対応付けて、文書管理サーバ205に送信する。署名側端末202は、生成した鍵情報ペアの公開鍵情報を、プライベートCAサーバ204に送信する。署名側端末202は、例えば、サーバ、PC、タブレット端末、または、スマートフォンなどである。

30

【0041】

署名側端末203は、文書情報を決裁する決裁者によって用いられるコンピュータである。署名側端末203は、決裁者の操作入力に基づき、署名情報が付与された文書情報を、文書管理サーバ205から取得する。署名側端末203は、決裁者の操作入力に基づき、取得した文書情報を改版し、新たな文書情報を生成する。署名側端末203は、秘密鍵情報と公開鍵情報との鍵情報ペアを生成する。署名側端末203は、生成した鍵情報ペアの秘密鍵情報に基づいて、生成した新たな文書情報に付与する署名情報を生成する。署名側端末203は、生成した新たな文書情報と、生成した署名情報とを対応付けて、文書管理サーバ205に送信する。署名側端末203は、生成した鍵情報ペアの公開鍵情報を、プライベートCAサーバ204に送信する。署名側端末203は、例えば、サーバ、PC、タブレット端末、または、スマートフォンなどである。

40

【0042】

プライベートCAサーバ204は、公開鍵情報を管理するコンピュータである。プライベートCAサーバ204は、図1に示した情報処理装置100としての機能を有する。プライベートCAサーバ204は、複数の公開鍵情報を集約した集約公開鍵情報を生成する。プライベートCAサーバ204は、生成した集約公開鍵情報を、パブリックCAサーバ206に送信する。プライベートCAサーバ204は、例えば、サーバ、または、PCなどである。

【0043】

50

文書管理サーバ２０５は、文書情報と対応付けられた署名情報を管理するコンピュータである。文書管理サーバ２０５は、図１に示した情報処理装置１００としての機能を有する。文書管理サーバ２０５は、複数の署名情報を集約した集約署名情報を生成する。文書管理サーバ２０５は、生成した集約署名情報を、図４に後述する制御情報管理テーブル４００に記憶する。文書管理サーバ２０５は、生成した集約署名情報を、検証側端末２０７に送信する。文書管理サーバ２０５は、例えば、サーバ、または、ＰＣなどである。

【００４４】

パブリックＣＡサーバ２０６は、集約公開鍵情報を管理するコンピュータである。パブリックＣＡサーバ２０６は、集約公開鍵情報を、プライベートＣＡサーバ２０４から受信する。パブリックＣＡサーバ２０６は、受信した集約公開鍵情報を、検証側端末２０７に送信する。パブリックＣＡサーバ２０６は、例えば、サーバ、または、ＰＣなどである。

【００４５】

検証側端末２０７は、検証者によって用いられるコンピュータである。検証側端末２０７は、集約署名情報を、文書管理サーバ２０５から受信する。検証側端末２０７は、集約公開鍵情報を、パブリックＣＡサーバ２０６から受信する。検証側端末２０７は、受信した集約公開鍵情報に基づいて、受信した集約署名情報が正当であるか否かを検証する。検証側端末２０７は、例えば、サーバ、または、ＰＣなどである。

【００４６】

ここでは、署名制御システム２００が、承認者に対応する署名側端末２０２を１つ含む場合について説明したが、これに限らない。例えば、署名制御システム２００が、承認者に対応する署名側端末２０２を複数含む場合があってもよい。ここでは、プライベートＣＡサーバ２０４と、文書管理サーバ２０５とが異なる装置である場合について説明したが、これに限らない。例えば、プライベートＣＡサーバ２０４と、文書管理サーバ２０５とが一体である場合があってもよい。

【００４７】

（情報処理装置１００のハードウェア構成例）

次に、図３を用いて、プライベートＣＡサーバ２０４、および、文書管理サーバ２０５に適用される情報処理装置１００のハードウェア構成例について説明する。

【００４８】

図３は、情報処理装置１００のハードウェア構成例を示すブロック図である。図３において、情報処理装置１００は、ＣＰＵ（Ｃｅｎｔｒａｌ Ｐｒｏｃｅｓｓｉｎｇ Ｕｎｉｔ）３０１と、メモリ３０２と、ネットワークＩ／Ｆ（Ｉｎｔｅｒｆａｃｅ）３０３と、記録媒体Ｉ／Ｆ３０４と、記録媒体３０５とを有する。また、各構成部は、バス３００によってそれぞれ接続される。

【００４９】

ここで、ＣＰＵ３０１は、情報処理装置１００の全体の制御を司る。メモリ３０２は、例えば、ＲＯＭ（Ｒｅａｄ Ｏｎｌｙ Ｍｅｍｏｒｙ）、ＲＡＭ（Ｒａｎｄｏｍ Ａｃｃｅｓｓ Ｍｅｍｏｒｙ）およびフラッシュＲＯＭなどを有する。具体的には、例えば、フラッシュＲＯＭやＲＯＭが各種プログラムを記憶し、ＲＡＭがＣＰＵ３０１のワークエリアとして使用される。メモリ３０２に記憶されるプログラムは、ＣＰＵ３０１にロードされることにより、コーディングされている処理をＣＰＵ３０１に実行させる。

【００５０】

ネットワークＩ／Ｆ３０３は、通信回線を通じてネットワーク２１０に接続され、ネットワーク２１０を介して他のコンピュータに接続される。そして、ネットワークＩ／Ｆ３０３は、ネットワーク２１０と内部のインターフェースを司り、他のコンピュータからのデータの入出力を制御する。ネットワークＩ／Ｆ３０３は、例えば、モデムやＬＡＮアダプタなどである。

【００５１】

記録媒体Ｉ／Ｆ３０４は、ＣＰＵ３０１の制御に従って記録媒体３０５に対するデータのリード／ライトを制御する。記録媒体Ｉ／Ｆ３０４は、例えば、ディスクドライブ、Ｓ

10

20

30

40

50

SD (Solid State Drive)、USB (Universal Serial Bus) ポートなどである。記録媒体 305 は、記録媒体 I/F 304 の制御で書き込まれたデータを記憶する不揮発メモリである。記録媒体 305 は、例えば、ディスク、半導体メモリ、USBメモリなどである。記録媒体 305 は、情報処理装置 100 から着脱可能であってもよい。

#### 【0052】

情報処理装置 100 は、上述した構成部の他、例えば、キーボード、マウス、ディスプレイ、プリンタ、スキャナ、マイク、スピーカーなどを有してもよい。また、情報処理装置 100 は、記録媒体 I/F 304 や記録媒体 305 を複数有していてもよい。また、情報処理装置 100 は、記録媒体 I/F 304 や記録媒体 305 を有していなくてもよい。

10

#### 【0053】

(制御情報管理テーブル 400 の記憶内容)

次に、図 4 を用いて、制御情報管理テーブル 400 の記憶内容の一例について説明する。制御情報管理テーブル 400 は、例えば、図 3 に示した情報処理装置 100 のメモリ 302 や記録媒体 305 などの記憶領域により実現される。

#### 【0054】

図 4 は、制御情報管理テーブル 400 の記憶内容の一例を示す説明図である。図 4 に示すように、制御情報管理テーブル 400 は、署名者と、文書情報と、署名対象と、秘密鍵と、公開鍵と、署名情報とのフィールドを有する。制御情報管理テーブル 400 は、署名者ごとに各フィールドに情報を設定することにより、制御情報がレコード 400-a として記憶される。a は、任意の整数である。

20

#### 【0055】

署名者のフィールドには、文書情報に署名情報を付与する署名者を識別する識別情報が設定される。署名者は、例えば、法人を含む。文書情報のフィールドには、署名情報が付与される文書情報が設定される。文書情報は、文責者、または、決裁者を示す承認フロー情報を含んでもよい。文責者は、例えば、作成者、または、承認者である。承認フロー情報は、文書情報に付与された集約署名情報の正当性が、いずれの公開鍵情報を集約した集約公開鍵情報に基づき検証可能であるのかを特定可能にする。署名対象のフィールドには、文書情報に対応するハッシュ値が設定される。秘密鍵のフィールドには、署名情報を生成する際に用いる秘密鍵情報が設定される。公開鍵のフィールドには、秘密鍵情報に対応する公開鍵情報が設定される。署名情報のフィールドには、文書情報に付与された署名情報が設定される。

30

#### 【0056】

(検証側端末 207 のハードウェア構成例)

次に、図 5 を用いて、検証側端末 207 のハードウェア構成例について説明する。

#### 【0057】

図 5 は、検証側端末 207 のハードウェア構成例を示すブロック図である。図 5 において、検証側端末 207 は、CPU 501 と、メモリ 502 と、ネットワーク I/F 503 と、記録媒体 I/F 504 と、記録媒体 505 とを有する。また、各構成部は、バス 500 によってそれぞれ接続される。

40

#### 【0058】

ここで、CPU 501 は、検証側端末 207 の全体の制御を司る。メモリ 502 は、例えば、ROM、RAM およびフラッシュ ROMなどを有する。具体的には、例えば、フラッシュ ROM や ROM が各種プログラムを記憶し、RAM が CPU 501 のワークエリアとして使用される。メモリ 502 に記憶されるプログラムは、CPU 501 にロードされることにより、コーディングされている処理を CPU 501 に実行させる。

#### 【0059】

ネットワーク I/F 503 は、通信回線を通じてネットワーク 210 に接続され、ネットワーク 210 を介して他のコンピュータに接続される。そして、ネットワーク I/F 503 は、ネットワーク 210 と内部のインターフェースを司り、他のコンピュータからの

50

データの入出力を制御する。ネットワーク I / F 5 0 3 は、例えば、モデムや L A N アダプタなどである。

#### 【 0 0 6 0 】

記録媒体 I / F 5 0 4 は、C P U 5 0 1 の制御に従って記録媒体 5 0 5 に対するデータのリード／ライトを制御する。記録媒体 I / F 5 0 4 は、例えば、ディスクドライブ、S S D、U S B ポートなどである。記録媒体 5 0 5 は、記録媒体 I / F 5 0 4 の制御で書き込まれたデータを記憶する不揮発メモリである。記録媒体 5 0 5 は、例えば、ディスク、半導体メモリ、U S B メモリなどである。記録媒体 5 0 5 は、検証側端末 2 0 7 から着脱可能であってもよい。

#### 【 0 0 6 1 】

検証側端末 2 0 7 は、上述した構成部の他、例えば、キーボード、マウス、ディスプレイ、プリンタ、スキャナ、マイク、スピーカーなどを有してもよい。また、検証側端末 2 0 7 は、記録媒体 I / F 5 0 4 や記録媒体 5 0 5 を複数有していてもよい。また、検証側端末 2 0 7 は、記録媒体 I / F 5 0 4 や記録媒体 5 0 5 を有していなくてもよい。

#### 【 0 0 6 2 】

(署名制御システム 2 0 0 の機能的構成例)

次に、図 6 を用いて、署名制御システム 2 0 0 の機能的構成例について説明する。

#### 【 0 0 6 3 】

図 6 は、署名制御システム 2 0 0 の機能的構成例を示すブロック図である。署名制御システム 2 0 0 において、情報処理装置 1 0 0 は、第 1 記憶部 6 0 0 と、第 1 取得部 6 0 1 と、第 1 生成部 6 0 2 と、修正部 6 0 3 と、解析部 6 0 4 と、第 2 生成部 6 0 5 と、第 1 出力部 6 0 6 とを含む。

#### 【 0 0 6 4 】

第 1 記憶部 6 0 0 は、例えば、図 3 に示したメモリ 3 0 2 や記録媒体 3 0 5 などの記憶領域によって実現される。以下では、第 1 記憶部 6 0 0 が、情報処理装置 1 0 0 に含まれる場合について説明するが、これに限らない。例えば、第 1 記憶部 6 0 0 が、情報処理装置 1 0 0 とは異なる装置に含まれ、第 1 記憶部 6 0 0 の記憶内容が情報処理装置 1 0 0 から参照可能である場合があってもよい。

#### 【 0 0 6 5 】

第 1 取得部 6 0 1 ～第 1 出力部 6 0 6 は、情報処理装置 1 0 0 の制御部の一例として機能する。第 1 取得部 6 0 1 ～第 1 出力部 6 0 6 は、具体的には、例えば、図 3 に示したメモリ 3 0 2 や記録媒体 3 0 5 などの記憶領域に記憶されたプログラムを C P U 3 0 1 に実行させることにより、または、ネットワーク I / F 3 0 3 により、その機能を実現する。各機能部の処理結果は、例えば、図 3 に示したメモリ 3 0 2 や記録媒体 3 0 5 などの記憶領域に記憶される。

#### 【 0 0 6 6 】

第 1 記憶部 6 0 0 は、各機能部の処理において参照され、または更新される各種情報を記憶する。第 1 記憶部 6 0 0 は、複数の文書情報を記憶する。文書情報は、例えば、契約書、または、ソースコードなどといった、複数人によって改版される文書情報である。複数の文書情報は、例えば、第 1 の文書の複数の版のそれぞれの版を示す文書情報を含む。

#### 【 0 0 6 7 】

第 1 記憶部 6 0 0 は、署名者ごとに鍵ペアを記憶する。鍵ペアは、秘密鍵情報と公開鍵情報とを含む。署名者は、例えば、法人を含む。第 1 記憶部 6 0 0 は、所定の承認フローに従って、所定の署名者ごとに鍵ペアを記憶する。第 1 記憶部 6 0 0 が記憶する鍵ペアの公開鍵情報は、第 1 記憶部 6 0 0 が記憶するいずれの文書情報にも対応しない公開鍵情報を含んでもよい。

#### 【 0 0 6 8 】

第 1 記憶部 6 0 0 は、文書情報に対応付けて、文書情報に付与された署名情報を記憶する。第 1 記憶部 6 0 0 は、例えば、予め定められた署名者ごとに、文書情報と、公開鍵情報と、署名情報とを対応付けて記憶する。第 1 記憶部 6 0 0 は、具体的には、図 4 に示し

10

20

30

40

50

た制御情報管理テーブル 400 を記憶する。

【0069】

第1取得部601は、各機能部の処理に用いられる各種情報を取得する。第1取得部601は、取得した各種情報を、第1記憶部600に記憶し、または、各機能部に出力する。また、第1取得部601は、第1記憶部600に記憶しておいた各種情報を、各機能部に出力してもよい。第1取得部601は、例えば、利用者の操作入力に基づき、各種情報を取得する。第1取得部601は、例えば、情報処理装置100とは異なる装置から、各種情報を受信してもよい。

【0070】

第1取得部601は、複数の文書情報と、複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得する。第1取得部601は、例えば、文書情報と、文書情報に対応する署名情報とを、それぞれの署名側端末201～203から収集することにより、複数の文書情報と、複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得する。第1取得部601は、例えば、利用者の操作入力に基づき、複数の文書情報と、複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得してもよい。

10

【0071】

第1取得部601は、いずれかの機能部の処理を開始する開始トリガーを受け付けてもよい。開始トリガーは、例えば、利用者による所定の操作入力があったことである。開始トリガーは、例えば、他のコンピュータから、所定の情報を受信したことであってもよい。開始トリガーは、例えば、いずれかの機能部が所定の情報を出力したことであってもよい。第1取得部601は、例えば、複数の文書情報と、複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得したことを、第1生成部602～第2生成部605の処理を開始する開始トリガーとして受け付けてもよい。

20

【0072】

第1生成部602は、取得した複数の文書情報に基づいて、取得した複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成する。

【0073】

第1生成部602は、取得した複数の文書情報のそれぞれの文書情報に対応するハッシュ値に基づいて、集約署名情報を生成する。第1生成部602は、例えば、シェノア署名アルゴリズムを利用する。第1生成部602が生成した集約署名情報の正当性は、取得した複数の文書情報と、第1生成部602が生成した集約署名情報と、第2生成部605が生成した集約公開鍵情報とに基づいて検証可能になる。これにより、第1生成部602は、検証者側で、文書情報の正当性を検証可能にする集約署名情報を得ることができる。

30

【0074】

第1生成部602は、安全素数と、取得した複数の文書情報のそれぞれの文書情報に対応するハッシュ値とに基づいて、集約署名情報を生成する。第1生成部602は、例えば、安全素数と、取得した複数の文書情報のそれぞれの文書情報に対応するハッシュ値とに基づいて、パラメータを設定する。そして、第1生成部602は、設定したパラメータに基づいて、集約署名情報を生成する。これにより、第1生成部602は、集約署名情報を生成する際にかかる処理量の低減化を図ることができる。

40

【0075】

第1生成部602は、シードと、前記第1の文書のそれぞれの版に対応するハッシュ値とから形成されるハッシュチェーンに基づいて、チェックハッシュ値を生成する。これにより、第1生成部602は、検証者側で、文書情報の正当性をより精度よく検証可能にするチェックハッシュ値を得ることができる。

【0076】

第2生成部605は、取得した複数の文書情報のそれぞれの文書情報に対応する公開鍵情報に基づいて、それぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報を生成する。文書情報に対応する公開鍵情報は、文書情報に対応する署名情報を生成する際に用いられた秘密鍵情報に対応する公開鍵情報である。これにより、第2生成部605は

50

、検証者側で、集約署名情報の正当性を検証する際に用いられる公開鍵情報の数の低減化を図ることができ、集約署名情報の正当性を検証する際にかかるコストの低減化を図ることができる。

【0077】

第2生成部605は、取得した複数の文書情報のそれぞれの文書情報に対応する公開鍵情報と、取得した複数の文書情報のいずれの文書情報にも対応しない公開鍵情報とを集約した集約公開鍵情報を生成する。第2生成部605は、例えば、第1記憶部600が、いずれの文書情報にも対応しない公開鍵情報を記憶していても、第1記憶部600が記憶する公開鍵情報を集約した集約公開鍵情報を生成する。これにより、第2生成部605は、所定の承認フローに従って、集約公開鍵情報を得ることができる。

10

【0078】

修正部603は、いずれの文書情報にも対応しない公開鍵情報に基づいて、生成した集約署名情報を修正した修正署名情報を生成する。これにより、修正部603は、所定の承認フローに反して、いずれかの秘密鍵情報に基づく署名情報が生成されていなくても、集約公開鍵情報に基づいて検証可能な修正署名情報を得ることができる。

【0079】

解析部604は、複数の文書情報と対応付けて出力した集約署名情報が正当ではないと判定された場合、集約元となる複数の署名情報のいずれの署名情報が正当ではないのかを特定する。解析部604は、例えば、集約元となる複数の署名情報のそれぞれの署名情報が正当であるか否かを検証する。

20

【0080】

解析部604は、例えば、集約署名情報が正当ではないと判定された場合、集約署名情報の集約元となる署名情報のうち、一部の署名情報を集約した集約署名情報と、一部の署名情報以外の署名情報を集約した集約署名情報との2つの集約署名情報を取得する。そして、解析部604は、取得した2つの集約署名情報のそれぞれの集約署名情報の正当性を検証する。これにより、解析部604は、集約元となる複数の署名情報のいずれの署名情報が正当ではないのかを特定することができ、検証者側で把握可能にすることができる。

【0081】

さらに、例えば、取得した2つの集約署名情報の少なくともいずれかの集約署名情報が正当ではないと判定された場合が考えられる。この場合、解析部604は、例えば、正当ではないと判定されたいずれかの集約署名情報の集約元となる署名情報のうち、一部の署名情報を集約した集約署名情報と、一部の署名情報以外の署名情報を集約した集約署名情報との2つの集約署名情報を新たに取得する。そして、解析部604は、新たに取得した2つの集約署名情報のそれぞれの集約署名情報の正当性を検証する。これにより、解析部604は、集約元となる複数の署名情報のいずれの署名情報が正当ではないのかを特定ことができ、検証者側で把握可能にすることができる。

30

【0082】

第1出力部606は、少なくともいずれかの機能部の処理結果を出力する。出力形式は、例えば、ディスプレイへの表示、プリンタへの印刷出力、ネットワークI/F303による外部装置への送信、または、メモリ302や記録媒体305などの記憶領域への記憶である。これにより、第1出力部606は、少なくともいずれかの機能部の処理結果を利用者に通知可能にすることができる。

40

【0083】

第1出力部606は、生成した集約署名情報を出力する。第1出力部606は、例えば、承認フローを示し、集約公開鍵情報を特定可能にする情報と共に、生成した集約署名情報を、複数の文書情報に対応付けて、検証者宛てに出力する。これにより、第1出力部606は、検証者側で、文書情報の正当性を検証可能にすることができる。

【0084】

第1出力部606は、生成した集約公開鍵情報を出力する。第1出力部606は、例えば、生成した集約公開鍵情報を、検証者宛てに出力する。第1出力部606は、具体的に

50

は、生成した集約公開鍵情報を、検証者が参照可能にパブリックCAサーバ206に登録する。これにより、第1出力部606は、検証者側で、文書情報の正当性を検証可能にすることができる。

【0085】

第1出力部606は、生成した集約署名情報を、生成した集約公開鍵情報と、複数の文書情報とに対応付けて出力する。第1出力部606は、例えば、生成した集約署名情報を、生成した集約公開鍵情報と、取得した複数の文書情報とに対応付けて、取得した複数の文書情報についての検証者宛てに出力する。集約署名情報は、例えば、複数の文書情報の最終版の文書情報と、最終版以外の文書情報に対応するハッシュ値と対応付けられることにより、複数の文書情報に対応付けられる。集約署名情報は、例えば、集約署名情報と対応付けられた最終版の文書情報に含まれる文責者および決裁者を識別する情報により、集約公開鍵情報と対応付けられる。これにより、第1出力部606は、検証者側で、文書情報の正当性を検証可能にすることができる。

10

【0086】

第1出力部606は、取得した2つの集約署名情報のそれぞれの集約署名情報の正当性を検証した結果を出力する。第1出力部606は、例えば、取得した2つの集約署名情報のそれぞれの集約署名情報の正当性を検証した結果を、利用者が参照可能に出力する。これにより、第1出力部606は、利用者が、いずれの署名情報が正当ではなく、いずれの承認者による承認行為に問題があるのかを把握可能にすることができる。

【0087】

20

第1出力部606は、生成した修正署名情報を出力する。第1出力部606は、例えば、承認フローを示し、集約公開鍵情報を特定可能にする情報と共に、生成した修正署名情報を、複数の文書情報に対応付けて、検証者宛てに出力する。これにより、第1出力部606は、検証者側で、文書情報の正当性を検証可能にすることができる。

【0088】

第1出力部606は、生成した修正署名情報を、生成した集約公開鍵情報と、複数の文書情報とに対応付けて出力する。第1出力部606は、例えば、生成した修正署名情報を、生成した集約公開鍵情報と、複数の文書情報とに対応付けて、検証者宛てに出力する。これにより、修正部603は、所定の承認フローに反して、いずれかの秘密鍵情報に基づく署名情報が生成されていなくても、検証者側で、文書情報の正当性を検証可能にすることができる。

30

【0089】

第1出力部606は、例えば、チェックハッシュ値を、集約署名情報に対応付けて出力する。これにより、第1出力部606は、検証者側で、文書情報の正当性をより精度よく検証可能にすることができる。

【0090】

署名制御システム200において、検証側端末207は、第2記憶部610と、第2取得部611と、検証部612と、第2出力部613とを含む。

【0091】

第2記憶部610は、例えば、図5に示したメモリ502や記録媒体505などの記憶領域によって実現される。以下では、第2記憶部610が、検証側端末207に含まれる場合について説明するが、これに限らない。例えば、第2記憶部610が、検証側端末207とは異なる装置に含まれ、第2記憶部610の記憶内容が検証側端末207から参照可能である場合があってもよい。

40

【0092】

第2取得部611～第2出力部613は、検証側端末207の制御部の一例として機能する。第2取得部611～第2出力部613は、具体的には、例えば、図5に示したメモリ502や記録媒体505などの記憶領域に記憶されたプログラムをCPU501に実行させることにより、または、ネットワークE/F503により、その機能を実現する。各機能部の処理結果は、例えば、図5に示したメモリ502や記録媒体505などの記憶領

50

域に記憶される。

【0093】

第2記憶部610は、各機能部の処理において参照され、または更新される各種情報を記憶する。第2記憶部610は、複数の文書情報に対応付けられた、複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報と、複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報とを記憶する。第2記憶部610は、例えば、複数の文書情報のうちの最終版の文書情報と、複数の文書情報のうちの最終版以外の文書情報に対応するハッシュ値との組み合わせに対応付けられた、集約署名情報と、集約公開鍵情報とを記憶する。

【0094】

第2取得部611は、各機能部の処理に用いられる各種情報を取得する。第2取得部611は、取得した各種情報を、記憶部に記憶し、または、各機能部に出力する。また、第2取得部611は、記憶部に記憶しておいた各種情報を、各機能部に出力してもよい。第2取得部611は、例えば、利用者の操作入力に基づき、各種情報を取得する。第2取得部611は、例えば、検証側端末207とは異なる装置から、各種情報を受信してもよい。

【0095】

第2取得部611は、複数の文書情報に対応付けられた、集約署名情報と、集約公開鍵情報とを取得する。第2取得部611は、例えば、集約署名情報を生成する装置から、複数の文書情報に対応付けられた集約署名情報を取得する。第2取得部611は、例えば、集約公開鍵情報を生成する装置から、複数の文書情報に対応付けられた集約公開鍵情報を取得する。

【0096】

第2記憶部610は、例えば、最終版の文書情報と、最終版以外の文書情報に対応するハッシュ値との組み合わせに対応付けられた、集約署名情報と、集約公開鍵情報とを、情報処理装置100から受信することにより取得する。第2記憶部610は、例えば、検証者の操作入力に基づき、複数の文書情報に対応付けられた、集約署名情報と、集約公開鍵情報とを受け付けてもよい。

【0097】

第2取得部611は、いずれかの機能部の処理を開始する開始トリガーを受け付けてもよい。開始トリガーは、例えば、利用者による所定の操作入力があったことである。開始トリガーは、例えば、他のコンピュータから、所定の情報を受信したことであってもよい。開始トリガーは、例えば、いずれかの機能部が所定の情報を出力したことであってもよい。第2取得部611は、例えば、複数の文書情報に対応付けられた、集約署名情報と、集約公開鍵情報とを取得したことを、検証部612の処理を開始する開始トリガーとして受け付ける。

【0098】

検証部612は、複数の文書情報と、受け付けた集約署名情報と、受け付けた集約公開鍵情報とに基づいて、受け付けた集約署名情報の正当性を検証する。検証部612は、例えば、複数の文書情報のそれぞれの文書情報に対応するハッシュ値と、受け付けた集約署名情報と、受け付けた集約公開鍵情報とに基づいて、受け付けた集約署名情報の正当性を検証する。これにより、検証部612は、集約署名情報の正当性を検証することができ、文書情報の正当性を検証することができる。

【0099】

第2出力部613は、少なくともいずれかの機能部の処理結果を出力する。出力形式は、例えば、ディスプレイへの表示、プリンタへの印刷出力、ネットワークI/F503による外部装置への送信、または、メモリ502や記録媒体505などの記憶領域への記憶である。これにより、第2出力部613は、少なくともいずれかの機能部の処理結果を利用者に通知可能にすることができる。第2出力部613は、例えば、検証部612が検証した結果を出力する。

【0100】

ここでは、情報処理装置 100 が、検証部 612 を有さない場合について説明したが、これに限らない。例えば、情報処理装置 100 が、検証部 612 を有する場合があってもよい。また、ここでは、情報処理装置 100 が、第 2 生成部 605 を有する場合について説明したが、これに限らない。例えば、情報処理装置 100 が、第 2 生成部 605 を有さず、第 2 生成部 605 を有する他のコンピュータが存在する場合があってもよい。

#### 【0101】

(署名制御システム 200 の具体的な機能的構成例)

次に、図 7～図 11 を用いて、署名制御システム 200 の具体的な機能的構成例について説明する。例えば、図 7～図 11 を用いて、署名制御システム 200 に含まれる各種装置の具体的な機能的構成例について説明する。

#### 【0102】

まず、図 7 を用いて、作成者に対応する署名側端末 201 の具体的な機能的構成例について説明する。

#### 【0103】

図 7 は、作成者に対応する署名側端末 201 の具体的な機能的構成例を示すブロック図である。署名制御システム 200 において、作成者に対応する署名側端末 201 は、鍵ペア生成部 700 と、データ受信部 701 と、シード値生成部 702 と、文書作成部 703 と、ハッシュ値計算部 704 と、署名生成部 705 と、データ送信部 706 とを含む。

#### 【0104】

また、作成者に対応する署名側端末 201 は、秘密鍵記憶部 710 と、公開鍵記憶部 720 とを含む。秘密鍵記憶部 710 は、署名側端末 201 で生成された秘密鍵を記憶する。公開鍵記憶部 720 は、署名側端末 201 で生成された公開鍵を記憶する。

#### 【0105】

鍵ペア生成部 700 は、秘密鍵と公開鍵との鍵ペアを生成し、秘密鍵を秘密鍵記憶部 710 に記憶し、公開鍵を公開鍵記憶部 720 に記憶する。データ受信部 701 は、メッセージを生成するためのデータを受信する。シード値生成部 702 は、シードを生成する。文書作成部 703 は、受信したデータに基づいて、メッセージを生成する。ハッシュ値計算部 704 は、生成したメッセージに対応するハッシュ値を計算する。署名生成部 705 は、計算したハッシュ値に基づいて、メッセージに付与する署名を生成する。

#### 【0106】

データ送信部 706 は、生成した署名を付与した署名付きメッセージを、文書管理サーバ 205 に送信する。文書管理サーバ 205 では、図 11 に後述する署名付き文書記憶部 1110 に、生成した署名を付与した署名付きメッセージが記憶される。データ送信部 706 は、公開鍵を、プライベート CA サーバ 204 に送信する。プライベート CA サーバ 204 では、図 10 に後述する個人公開鍵記憶部 1010 に、公開鍵が記憶される。

#### 【0107】

次に、図 8 を用いて、承認者に対応する署名側端末 202 の具体的な機能的構成例について説明する。

#### 【0108】

図 8 は、承認者に対応する署名側端末 202 の具体的な機能的構成例を示すブロック図である。署名制御システム 200 において、承認者に対応する署名側端末 202 は、鍵ペア生成部 800 と、データ受信部 801 と、文書作成部 802 と、ハッシュ値計算部 803 と、署名生成部 804 と、データ送信部 805 とを含む。

#### 【0109】

また、承認者に対応する署名側端末 202 は、秘密鍵記憶部 810 と、公開鍵記憶部 820 とを含む。秘密鍵記憶部 810 は、署名側端末 202 で生成された秘密鍵を記憶する。公開鍵記憶部 820 は、署名側端末 202 で生成された公開鍵を記憶する。

#### 【0110】

鍵ペア生成部 800 は、秘密鍵と公開鍵との鍵ペアを生成し、秘密鍵を秘密鍵記憶部 810 に記憶し、公開鍵を公開鍵記憶部 820 に記憶する。データ受信部 801 は、署名付

10

20

30

40

50

きメッセージを文書管理サーバ205から受信する。データ受信部801は、受信した署名付きメッセージを改版し、新たなメッセージを生成するためのデータを受信する。文書作成部802は、受信した署名付きメッセージと、受信したデータとに基づいて、新たなメッセージを生成する。ハッシュ値計算部803は、生成したメッセージに対応するハッシュ値を計算する。署名生成部804は、計算したハッシュ値に基づいて、メッセージに付与する署名を生成する。

#### 【0111】

データ送信部805は、生成した署名を付与した署名付きメッセージを、文書管理サーバ205に送信する。文書管理サーバ205では、図11に後述する署名付き文書記憶部1110に、生成した署名を付与した署名付きメッセージが記憶される。データ送信部805は、公開鍵を、プライベートCAサーバ204に送信する。プライベートCAサーバ204では、図10に後述する個人公開鍵記憶部1010に、公開鍵が記憶される。

#### 【0112】

また、決裁者に対応する署名側端末203の具体的な機能的構成例は、例えば、承認者に対応する署名側端末202の具体的な機能的構成例と同様である。

#### 【0113】

次に、図9を用いて、検証側端末207の具体的な機能的構成例について説明する。

#### 【0114】

図9は、検証側端末207の具体的な機能的構成例を示すブロック図である。署名制御システム200において、検証側端末207は、データ受信部901と、ハッシュチェーン検証部902と、集約署名検証部903とを含む。また、検証側端末207は、集約署名付き文書記憶部910を含む。一方で、パブリックCAサーバ206は、組織公開鍵記憶部900を含む。組織公開鍵記憶部900は、集約公開鍵を記憶する。

#### 【0115】

集約署名付き文書記憶部910は、集約署名付きメッセージを記憶する。集約署名付きメッセージは、例えば、文書管理サーバ205から送信される。データ受信部901は、集約署名の正当性を検証する際に用いられる集約公開鍵を、パブリックCAサーバ206から受信する。ハッシュチェーン検証部902は、集約署名付きメッセージに付与されたチェックハッシュ値に基づいて、集約署名付きメッセージに対応するハッシュチェーンの正当性を検証する。ハッシュチェーン検証部902は、集約署名付きメッセージに対応するハッシュチェーンが正当であれば、集約署名付きメッセージを集約署名検証部903に渡す。集約署名検証部903は、集約公開鍵に基づいて、集約署名付きメッセージに付与された集約署名の正当性を検証する。

#### 【0116】

次に、図10を用いて、プライベートCAサーバ204の具体的な機能的構成例について説明する。

#### 【0117】

図10は、プライベートCAサーバ204の具体的な機能的構成例を示すブロック図である。署名制御システム200において、プライベートCAサーバ204は、集約公開鍵生成部1001と、公開鍵登録申請部1002とを含む。また、プライベートCAサーバ204は、個人公開鍵記憶部1010を含む。個人公開鍵記憶部1010は、それぞれの署名側端末201～203で生成された公開鍵を記憶する。

#### 【0118】

集約公開鍵生成部1001は、個人公開鍵記憶部1010に記憶された、それぞれの署名側端末201～203で生成された公開鍵を集約し、集約公開鍵を生成する。公開鍵登録申請部1002は、生成した集約公開鍵を、パブリックCAサーバ206に送信する。パブリックCAサーバ206では、図9に示した組織公開鍵記憶部900に、生成した集約公開鍵が記憶される。

#### 【0119】

次に、図11を用いて、文書管理サーバ205の具体的な機能的構成例について説明す

10

20

30

40

50

る。

#### 【0120】

図11は、文書管理サーバ205の具体的な機能的構成例を示す説明図である。署名制御システム200において、文書管理サーバ205は、集約署名生成部1101と、データ送信部1102とを含む。また、文書管理サーバ205は、署名付き文書記憶部1110を含む。署名付き文書記憶部1110は、それぞれの署名側端末201～203で生成された署名付きメッセージを記憶する。

#### 【0121】

集約署名生成部1101は、署名付き文書記憶部1110に記憶された、署名付きメッセージに付与された署名を集約し、集約署名を生成する。集約署名生成部1101は、生成した集約署名を、最終版のメッセージに付与し、集約署名付きメッセージを生成する。データ送信部1102は、生成した集約署名付きメッセージを、検証側端末207に送信する。検証側端末207では、図9に示した集約署名付き文書記憶部910に、生成した集約署名付きメッセージが記憶される。

#### 【0122】

(署名制御システム200の動作の一例)

次に、図12を用いて、署名制御システム200の動作の一例について説明する。

#### 【0123】

図12は、署名制御システム200の動作の一例を示す説明図である。図12において、位数は、 $p$ とする。秘密鍵を生成する際、または、署名を生成する際などに用いられる剰余群は、 $Z_p = Z / Z_p$ とする。公開鍵を生成する際、または、署名を生成する際などに用いられる値は、 $g$ とする。暗号的ハッシュ関数は、 $H(x)$ とする。 $H(x)$ は、例えば、SHA256、または、SHA-3などのSecure Hash Algorithmを利用する。

#### 【0124】

署名側端末201～203は、剰余群 $Z_p$ から一様ランダムに $x$ を選択し、秘密鍵に設定する。署名側端末201～203は、 $X := g^x$ を、秘密鍵 $x$ に対応する公開鍵に設定する。署名側端末201～203は、メッセージ $m$ に対して、秘密鍵 $x$ に基づいて署名を生成する。署名側端末201～203は、例えば、剰余群 $Z_p$ から一様ランダムに $r$ を選択し、署名要素 $R := g^r$ を設定する。

#### 【0125】

署名側端末201～203は、例えば、メッセージハッシュ $c := H(R || m)$ を設定する。署名側端末201～203は、署名要素 $s := r + xc \pmod{p-1}$ を設定する。署名側端末201～203は、署名要素 $s$ 、 $R$ を、メッセージ $m$ に対する署名 $(s, R)$ に設定し、文書管理サーバ205に送信する。

#### 【0126】

図12の例では、メッセージ $m_1, m_2, m_3$ が存在するとする。文書管理サーバ205は、メッセージ $m_1, m_2, m_3$ と、それぞれメッセージ $m_1, m_2, m_3$ に対応する3つの署名 $(s_1, R_1), (s_2, R_2), (s_3, R_3)$ を有する。

#### 【0127】

文書管理サーバ205は、3つの署名 $(s_1, R_1), (s_2, R_2), (s_3, R_3)$ を単一の集約署名 $(s', R')$ に集約する。文書管理サーバ205は、例えば、補助パラメータ $d = \{d_i = (\prod_{j \in [1,2,3] - \{i\}} c_j)\} = \{d_1 = c_2 \times c_3, d_2 = c_3 \times c_1, d_3 = c_1 \times c_2\} \pmod{p-1}$ を設定する。文書管理サーバ205は、例えば、署名要素 $s' := s_1 \times d_1 + s_2 \times d_2 + s_3 \times d_3 \pmod{p-1}$ を設定する。文書管理サーバ205は、例えば、署名要素 $R' := R_1^{(d_1)} \times R_2^{(d_2)} \times R_3^{(d_3)}$ を設定する。文書管理サーバ205は、単一の集約署名 $(s', R')$ を設定する。

#### 【0128】

文書管理サーバ205は、3つの公開鍵 $X_1, X_2, X_3$ を単一の集約公開鍵 $X'$ に集約する。文書管理サーバ205は、例えば、単一の集約公開鍵 $X' = X_1 \times X_2 \times X_3$ を設定する。

。これにより、文書管理サーバ205は、単一の集約署名( $s'$ ,  $R'$ )の正当性を検証する際に、3つの公開鍵 $X_1$ ,  $X_2$ ,  $X_3$ を用いずに済ませることができ、単一の集約公開鍵 $X'$ を用いればよくすることができる。このため、文書管理サーバ205は、単一の集約署名( $s'$ ,  $R'$ )の正当性を検証する際に用いられる公開鍵の数の低減化を図り、単一の集約署名( $s'$ ,  $R'$ )の正当性を検証する際にかかるコストの低減化を図ることができる。

【0129】

検証側端末207は、集約公開鍵 $X'$ に基づいて、集約署名( $s'$ ,  $R'$ )の正当性を検証する。検証側端末207は、メッセージハッシュ $c_1 := H(R_1 || m_1)$ ,  $c_2 := H(R_2 || m_2)$ ,  $c_3 := H(R_3 || m_3)$ を設定する。検証側端末207は、 $c' := q \times c_2 \times c_3$ を算出し、 $g^{(s)}$ と $R' \times (X'^{(s)})$ が一致するか否かを判定する。検証側端末207は、一致すると判定した場合、単一の集約署名( $s'$ ,  $R'$ )が正当であると判定する。一方で、検証側端末207は、一致しないと判定した場合、単一の集約署名( $s'$ ,  $R'$ )が正当ではないと判定する。

10

【0130】

これにより、検証側端末207は、単一の集約署名( $s'$ ,  $R'$ )の正当性を検証し、メッセージ $m_1$ ,  $m_2$ ,  $m_3$ が正当であるか否かを判断することができる。また、検証側端末207は、単一の集約署名( $s'$ ,  $R'$ )の正当性を検証し、3つの署名( $s_1$ ,  $R_1$ ), ( $s_2$ ,  $R_2$ ), ( $s_3$ ,  $R_3$ )が正当であるか否かを判断することができる。このため、検証側端末207は、セキュリティの向上を図ることができる。検証側端末207は、署名の数に依らず、単一の集約公開鍵 $X'$ に基づいて、単一の集約署名( $s'$ ,  $R'$ )の正当性を検証することができ、検証にかかる処理量の低減化を図ることができる。

20

【0131】

(署名制御システム200の動作の具体例)

次に、図13～図15を用いて、署名制御システム200の動作の具体例について説明する。

【0132】

図13～図15は、署名制御システム200の動作の具体例を示す説明図である。具体例では、署名制御システム200は、複数版の文書のうち、最終版以外の文書を検証者から秘匿しつつ、最終版の文書の内容に基づいて、複数版の文書が生成された過程を検証可能とする。

30

【0133】

図13において、作成者が、文書第1版を作成する。承認者が、文書第1版に基づく文書第2版を作成する。決裁者が、文書第2版に基づく文書第3版を作成する。文書第3版が、文書最終版として確定される。このように、文書が段階的に修正される状況である。

【0134】

文書管理サーバ205は、作成者により署名側端末201で生成された、文書第1版と、文書第1版に対応する署名 $1 := (s_1, R_1)$ とを取得する。文書管理サーバ205は、承認者により署名側端末202で生成された、文書第1版に基づく文書第2版と、文書第2版に対応する署名 $2 := (s_2, R_2)$ とを取得する。文書管理サーバ205は、決裁者により署名側端末203で生成された、文書第2版に基づく文書第3版と、文書第3版に対応する署名 $3 := (s_3, R_3)$ とを取得する。文書管理サーバ205は、文書第3版に基づく文書最終版を取得する。

40

【0135】

文書管理サーバ205は、署名 $1 := (s_1, R_1)$ と、署名 $2 := (s_2, R_2)$ と、署名 $3 := (s_3, R_3)$ とを、単一の集約署名( $s'$ ,  $R'$ )に集約する。単一の集約署名( $s'$ ,  $R'$ )は、法人を代表するe-Sealとなる。

【0136】

文書管理サーバ205は、例えば、補助パラメータ $d = \{d_i = (\prod_{j \in [1,2,3]-[i]} c_j)\} = \{d_1 = c_2 \times c_3, d_2 = c_3 \times c_1, d_3 = c_1 \times c_2\} \pmod{(p-1)}$ を設定する。補助パラメータ $d$ を設定する一例については、図17および図18を用

50

いて後述する。

【0137】

文書管理サーバ205は、例えば、署名要素  $s' := s_1 \times d_1 + s_2 \times d_2 + s_3 \times d_3 \pmod{(p-1)}$  を設定する。文書管理サーバ205は、例えば、署名要素  $R' := R^{(d_1)} \times R_2^{(d_2)} \times R_3^{(d_3)}$  を設定する。文書管理サーバ205は、単一の集約署名 ( $s'$ ,  $R'$ ) を生成する。

【0138】

プライベートCAサーバ204は、公開鍵  $X_1$ ,  $X_2$ ,  $X_3$  を集約し、集約公開鍵  $X' = X_1 \cdot X_2 \cdot X_3$  を生成する。プライベートCAサーバ204は、集約公開鍵  $X'$  を、社外に存在するパブリックCAサーバ206に送信する。ここで、一旦、図14の説明に移行する。

10

【0139】

図14において、文書管理サーバ205は、ハッシュチェーン1400を生成し、チェックハッシュ値  $chk_{hash}$  を生成する。文書管理サーバ205は、例えば、シード値と、文書各版のメッセージハッシュ  $hs_i$  とから形成されるハッシュチェーン1400に基づいて、チェックハッシュ値  $chk_{hash}$  を生成する。

【0140】

文書管理サーバ205は、具体的には、ハッシュ関数を利用して、シード値と、文書第1版のメッセージハッシュ  $hs_1$  とに基づいて、第1ハッシュ値を算出する。文書管理サーバ205は、具体的には、ハッシュ関数を利用して、算出した第1ハッシュ値と、文書第2版のメッセージハッシュ  $hs_2$  とに基づいて、第2ハッシュ値を算出する。文書管理サーバ205は、具体的には、ハッシュ関数を利用して、算出した第2ハッシュ値と、文書第3版のメッセージハッシュ  $hs_3$  とに基づいて、第3ハッシュ値を算出する。

20

【0141】

文書管理サーバ205は、第3ハッシュ値を、チェックハッシュ値  $chk_{hash}$  に設定する。文書管理サーバ205は、シード値と、文書第1版のメッセージハッシュ  $hs_1$  と、文書第2版のメッセージハッシュ  $hs_2$  と、文書最終版=文書第3版と、チェックハッシュ値  $chk_{hash}$  との組み合わせを、検証用情報として纏めておく。

【0142】

図13の説明に戻り、文書管理サーバ205は、纏められた検証用情報と、生成された単一の集約署名 ( $s'$ ,  $R'$ ) とを対応付けて、検証側端末207に送信する。検証側端末207は、纏められた検証用情報と、生成された単一の集約署名 ( $s'$ ,  $R'$ ) とを受信する。検証側端末207は、生成された集約公開鍵  $X'$  を、パブリックCAサーバ206から取得する。ここで、一旦、図15の説明に移行する。

30

【0143】

図15において、検証側端末207は、ハッシュチェーン1500を生成し、チェックハッシュ値  $chk_{hash}$  を生成する。検証側端末207は、例えば、シード値と、文書第1版のメッセージハッシュ  $hs_1$  と、文書第2版のメッセージハッシュ  $hs_2$  と、文書最終版とから形成されるハッシュチェーン1500に基づいて、チェックハッシュ値  $chk_{hash}$  を生成する。

【0144】

検証側端末207は、具体的には、ハッシュ関数を利用して、シード値と、文書第1版のメッセージハッシュ  $hs_1$  とに基づいて、第1ハッシュ値を算出する。検証側端末207は、具体的には、ハッシュ関数を利用して、算出した第1ハッシュ値と、文書第2版のメッセージハッシュ  $hs_2$  とに基づいて、第2ハッシュ値を算出する。検証側端末207は、具体的には、ハッシュ関数を利用して、文書最終版=文書第3版のメッセージハッシュ  $hs_3$  を算出する。検証側端末207は、具体的には、ハッシュ関数を利用して、算出した第2ハッシュ値と、文書第3版のメッセージハッシュ  $hs_3$  とに基づいて、第3ハッシュ値を算出する。

40

【0145】

検証側端末207は、第3ハッシュ値を、チェックハッシュ値  $chk_{hash}$  に設定する。

50

検証側端末207は、設定したチェックハッシュ値  $chk_{hash}$  が、受信した検証用情報に含まれるチェックハッシュ値  $chk_{hash}$  と一致するか否かを判定する。ここで、検証側端末207は、一致すると判定した場合、ハッシュチェーン1500が正当であると判定する。一方で、検証側端末207は、一致しないと判定した場合、ハッシュチェーン1500が正当ではなく、文書最終版が正当ではないと判定する。

【0146】

図13の説明に戻り、検証側端末207は、ハッシュチェーン1500が正当であると判定した場合、受信した集約公開鍵  $X'$  に基づいて、受信した単一の集約署名  $(s', R')$  の正当性を検証する。検証側端末207は、例えば、メッセージハッシュ  $c_1 = H(R_1 || m_1)$  ,  $c_2 = H(R_2 || m_2)$  ,  $c_3 = H(R_3 || m_3)$  を取得し、 $c' = qc_2c_3$  を算出する。ここで、検証側端末207は、 $g^{(s')} \equiv R' (X'^c)'$  であれば、単一の集約署名  $(s', R')$  が正当であると判定する。一方で、検証側端末207は、 $g^{(s')} \equiv R' (X'^c)'$  でなければ、単一の集約署名  $(s', R')$  が正当ではなく、文書最終版が正当ではないと判定する。

【0147】

検証側端末207は、ハッシュチェーン1500が正当、かつ、単一の集約署名  $(s', R')$  が正当であれば、文書最終版が正当であると判定する。これにより、検証側端末207は、検証者が、不正な文書を、誤って正当な文書と認識してしまうことを防止することができる。検証側端末207は、複数の公開鍵を取得せずとも、単一の集約公開鍵  $X'$  を取得すれば、単一の集約署名  $(s', R')$  の正当性を検証することができる。このため、検証側端末207は、単一の集約署名  $(s', R')$  の正当性を検証する際にかかるコストの低減化を図ることができる。検証側端末207は、ハッシュチェーン1500の正当性を検証するため、文書第1版、および、文書第2版が、改ざん、または、置き換えされていないことを確認することができる。このため、検証側端末207は、セキュリティの向上を図ることができる。

【0148】

(署名を検証する一例)

次に、図16を用いて、単一の集約署名  $(s', R')$  が正当ではないと判定された場合に、文書管理サーバ205が、署名を検証する一例について説明する。

【0149】

図16は、署名を検証する一例を示す説明図である。図16において、文書管理サーバ205は、単一の集約署名  $(s', R')$  が正当ではないと判定され、検証失敗したことを検出する。文書管理サーバ205は、例えば、単一の集約署名  $(s', R')$  が検証失敗したことの通知を、検証側端末207から受信する。この場合、単一の集約署名  $(s', R')$  の集約元となる少なくともいずれかの署名に、検証失敗した原因が存在すると考えられる。

【0150】

このため、文書管理サーバ205が、検証失敗した原因となる署名を発見し、署名側端末201～203に、検証失敗した原因となる署名を生成し直させるようにすることが望まれる。文書管理サーバ205は、例えば、集約元となるN個の署名のそれぞれの署名  $j$  ( $j = 1 \sim N$ ) の正当性を検証することにより、検証失敗した原因となる署名を発見する。これにより、文書管理サーバ205は、署名側端末201～203に、検証失敗した原因となる署名を生成し直させることができる。

【0151】

ここでは、文書管理サーバ205が、集約元となるN個の署名のそれぞれの署名  $j$  ( $j = 1 \sim N$ ) の正当性を検証し、検証失敗した原因となる署名を発見する場合について説明したが、これに限らない。例えば、文書管理サーバ205が、さらに、検証失敗した原因となる署名を発見する際にかかる処理量の低減化を図る場合があってもよい。

【0152】

この場合、文書管理サーバ205は、予め、単一の集約署名  $s_{1,1} = (s', R')$  を生成

するにあたり、集約元となる  $j + 1$  個の署名のうち、一部の署名を集約した集約署名  $s_{x,y}$  を生成し、リストに登録しておく。文書管理サーバ 205 は、例えば、番号順に並べた  $j + 1$  個の署名のうち、隣り合う署名同士を集約し、段  $x = k$  に属する集約署名  $s_{x=k,y}$  を生成し、リストに登録しておく。例えば、 $y = 1, 2, 3, \dots$  である。

【0153】

文書管理サーバ 205 は、下段  $x = i + 1$  に属する集約署名  $s_{x=i+1,y}$  のうち、隣り合う署名同士を集約し、上段  $x = i$  に属する集約署名  $s_{x=i,y}$  を生成し、集約元と集約先との関係を特定可能に、リストに登録しておくことを、再帰的に繰り返す。最終的に、文書管理サーバ 205 は、単一の集約署名  $s_{1,1} = (s', R')$  を生成する。

【0154】

文書管理サーバ 205 は、単一の集約署名  $(s', R')$  が検証失敗したことを検出すると、リストを読み出す。文書管理サーバ 205 は、リストのうち、段  $x = 2$  に属する集約署名  $s_{x=2,y}$  の正当性を検証する。文書管理サーバ 205 は、上段  $x = i$  に属する集約署名  $s_{x=i,y}$  のうち、正当ではないと判定された集約署名  $s_{x=i,y}$  の集約元となった、下段  $x = i + 1$  に属する集約署名  $s_{x=i+1,y}$  の正当性を検証することを繰り返す。これにより、文書管理サーバ 205 は、正当ではない署名  $j$  を発見することができる。また、文書管理サーバ 205 は、検証失敗した原因となる署名を発見する際にかかる処理量を、 $O(\log(N))$  に抑制することができる。

【0155】

(補助パラメータ  $d$  を設定する一例)

次に、図 17 および図 18 を用いて、文書管理サーバ 205 が、補助パラメータ  $d$  を設定する一例について説明する。

【0156】

図 17 および図 18 は、補助パラメータ  $d$  を設定する一例を示す説明図である。文書管理サーバ 205 は、剰余の性質を利用して、補助パラメータ  $d$  を効率的に算出する。文書管理サーバ 205 は、例えば、位数  $p$  を安全素数に設定する。安全素数は、 $p = 2q + 1$  の  $q$  を素数にする  $p$  である。

【0157】

(17-1) 文書管理サーバ 205 は、 $2^k$  除算アルゴリズムにより、それぞれのメッセージハッシュ  $c_i$  を 2 進数展開し、数式 1711 ~ 171n に示す「 $c_i' \leftarrow 2^{-(t)} \cdot c_i$ 」の形式に変形する。

【0158】

(17-2) 文書管理サーバ 205 は、 $t_i$  ソートアルゴリズムにより、数式 1711 ~ 171n に示す「 $c_i' \leftarrow 2^{-(t)} \cdot c_i$ 」の形式を、 $t_1 \leq t_2 \leq \dots \leq t_N$  の順にソートする。

【0159】

(17-3) 文書管理サーバ 205 は、C 計算アルゴリズムにより、数式 1720 に示す  $C$  を算出する。

【0160】

(17-4) 文書管理サーバ 205 は、逆数計算アルゴリズムにより、それぞれの  $c_i'$  について、数式 1731 ~ 173n に示す逆数  $y_i$  を算出する。

【0161】

(17-5) 文書管理サーバ 205 は、 $d$  計算アルゴリズムにより、算出した  $C$  に基づいて、それぞれの逆数  $y_i$  に基づいて、数式 1741 ~ 174n に示す  $d_i$  を算出する。

【0162】

(17-6) 文書管理サーバ 205 は、算出した  $d_i$  を纏め、数式 1750 に示す補助パラメータ  $d$  を設定する。次に、図 18 の説明に移行する。

【0163】

図 18 の表 1800 は、補助パラメータ  $d$  を設定する際にかかる処理時間を示す。表 1800 は、例えば、図 17 に示した手法で、補助パラメータ  $d$  を設定する際にかかる処理

10

20

30

40

50

時間を示す。また、表 1800 は、例えば、比較例として、 $d = \{d_i = \prod_{j \in [N] - [i]} c_j\} \pmod{(p-1)}$  を算出する手法で、補助パラメータ  $d$  を設定する際にかかる処理時間を示す。

【0164】

表 1800 に示すように、文書管理サーバ 205 は、図 17 に示した手法によれば、比較例とする手法よりも、補助パラメータ  $d$  を設定する際にかかる処理時間を低減することができる。文書管理サーバ 205 は、例えば、処理時間を、 $O(N^2)$  ではなく  $O(N)$  にすることができる。文書管理サーバ 205 は、図 17 に示した手法によれば、署名の数が増加しても、比較的短い処理時間で、補助パラメータ  $d$  を設定することができる。

【0165】

(集約署名を修正する一例)

次に、図 19 および図 20 を用いて、集約署名を修正する一例について説明する。

【0166】

図 19 および図 20 は、集約署名を修正する一例を示す説明図である。例えば、承認フローに定められた、いずれかの承認者が、文書を承認することができず、署名側端末 201 ~ 203 を操作することができず、文書に署名を付与することができない場合がある。具体的には、いずれかの承認者が、休暇中、または、出張中などの状態で、不在であり、署名側端末 201 ~ 203 を操作することができず、文書に署名を付与することができない場合がある。

【0167】

しかしながら、いずれかの承認者が、休暇中、または、出張中などの状態にある場合、署名制御システム 200 において、集約公開鍵により検証可能な集約署名を生成することができないとすれば、署名制御システム 200 の利便性の低下を招くことになる。従って、署名制御システム 200 において、一時的に公開鍵を無効化し、いずれかの承認者により、文書に署名が付与されていなくても、集約公開鍵により検証可能な集約署名を生成可能にすることが望まれる。また、承認者が、休暇中、または、出張中などの状態にある場合、承認フローを変更し、集約公開鍵を配布し直すことは、作業コストの観点などから好ましくない。

【0168】

図 19 において、文書管理サーバ 205 は、メッセージ  $m_i$  と、いずれかの承認者に対応する無効化する公開鍵  $p_{k_{rev}}$  とに基づいて、いずれかの承認者に対応する署名が集約されなかった集約署名を修正し、修正署名を生成する。修正署名は、いずれかの承認者に対応する公開鍵  $p_{k_{rev}}$  が集約された集約公開鍵  $X'$  により検証可能な署名である。例えば、 $i = 1, 2, 3$  である。ここで、図 20 の説明に移行し、文書管理サーバ 205 が、集約署名を修正する具体例について説明する。

【0169】

図 20 において、文書管理サーバ 205 は、無効化する公開鍵  $X_2$  の指定を受け付ける。ここで、生成済みの集約署名 ( $S'' = s_1 c_3 + s_3 c_1$ ,  $R'' = R_1^{(c_3)} R_3^{(c_2)}$ ) であり、公開鍵  $X_2$  によって検証可能な署名 ( $s_2$ ,  $R_2$ ) が集約されていない情報である。文書管理サーバ 205 は、集約署名 ( $S''$ ,  $R''$ ) を修正し、集約公開鍵  $X' = X_1 X_2 X_3$  により検証可能な修正署名を生成する。修正署名は、集約公開鍵  $X' = X_1 X_2 X_3$  により検証可能にするため、下記式 (1) が成立するように生成されればよい。

【0170】

$$\begin{aligned} g^{(s)} &\equiv R' \cdot (X')^{(t)} \pmod{(p)} \\ \therefore g^{(s)} &\equiv R'' \cdot (X_1 X_3)^{(c_1 c_3)} \\ &\equiv X_2^{(-c_1 c_3)} R'' \cdot (X_1 X_2 X_3)^{(c_1 c_3)} \\ &\equiv R' \cdot (X')^{(t)} \end{aligned} \quad \dots (1)$$

【0171】

文書管理サーバ 205 は、具体的には、上記式 (1) が成立するよう、集約署名 ( $S''$ ,  $R''$ ) を修正する。文書管理サーバ 205 は、より具体的には、集約署名 ( $S''$ ,  $R''$ ) と、

10

20

30

40

50

メッセージ $m_1$ ,  $m_3$ と、無効化する公開鍵 $p k_2 = X_2$ とに基づいて、集約公開鍵 $X' = X_1 X_2 X_3$ により検証可能な修正署名 $s' = (s'', R', \frac{1}{2}(X_1^{1c3})R'')$ を生成する。これにより、文書管理サーバ205は、承認者が不在であっても、集約公開鍵 $X$ により検証可能な修正署名 $s'$ を生成することができる。このため、文書管理サーバ205は、一時的に公開鍵を無効化することができ、署名制御システム200の利便性の向上を図ることができる。

【0172】

(署名制御システム200による効果)

次に、図21を用いて、署名制御システム200による効果について説明する。

【0173】

図21は、署名制御システム200による効果を示す説明図である。図21の表2100は、プライベートCAサーバ204で管理する公開鍵の数、および、パブリックCAサーバ206で管理する公開鍵の数を示す。表2100は、例えば、署名制御システム200において、プライベートCAサーバ204で管理する公開鍵の数、および、パブリックCAサーバ206で管理する公開鍵の数を示す。また、表2100は、例えば、従来のプライベートCAサーバ204で管理する公開鍵の数、および、従来のパブリックCAサーバ206で管理する公開鍵の数を示す。

10

【0174】

表2100に示すように、署名制御システム200は、パブリックCAサーバ206で管理する公開鍵の数を、従来に比べて、 $N$ 個から1個に低減することができる。このため、署名制御システム200は、公開鍵を管理する際にかかるコスト、および、公開鍵を利用する際にかかるコストの低減化を図ることができる。

20

【0175】

具体的には、公開鍵のデータ長が、2048 [bit]であれば、従来のパブリックCAサーバ206には、承認者が100人存在する場合、 $2048 \times 100$  [bit]の情報が記憶されることになる。このため、従来では、集約署名の正当性を検証する際にかかる処理コストは、 $O(N)$ となる。

【0176】

一方で、署名制御システム200は、集約署名の正当性を検証する際、単一の集約公開鍵を利用すればよい。集約署名の正当性を検証する際にかかる処理コストを、 $O(1)$ に低減することができる。署名制御システム200は、承認者の数によらず、パブリックCAサーバ206で管理する情報を、2048 [bit]に固定することができる。

30

【0177】

(鍵生成処理手順)

次に、図22を用いて、署名制御システム200が実行する、鍵生成処理手順の一例について説明する。鍵生成処理は、例えば、図3に示したCPU301と、メモリ302や記録媒体305などの記憶領域と、ネットワークI/F303となどによって実現される。

【0178】

図22は、鍵生成処理手順の一例を示すフローチャートである。図22において、プライベートCAサーバ204は、鍵生成要求を、作成者、承認者、および、決裁者に対応する署名側端末201~203に送信する(ステップS2201)。

40

【0179】

一方で、署名側端末201~203は、鍵生成要求を受信すると、秘密鍵 $x_i$ と、公開鍵 $X_i$ との鍵ペアを生成する(ステップS2202)。次に、署名側端末201~203は、生成した秘密鍵 $x_i$ を記憶する(ステップS2203)。そして、署名側端末201~203は、生成した公開鍵 $X_i$ を記憶する(ステップS2204)。そして、署名側端末201~203は、生成した公開鍵 $X_i$ を、プライベートCAサーバ204に送信する(ステップS2205)。

【0180】

一方で、プライベートCAサーバ204は、公開鍵 $X_i$ を、それぞれの署名側端末201~203から受信する(ステップS2206)。 $i = 1, 2, 3$ である。次に、プライベ

50

ートCAサーバ204は、公開鍵 $X_i$ を集約した集約公開鍵 $X'$ を生成して記憶する（ステップS2207）。そして、プライベートCAサーバ204は、集約公開鍵 $X'$ を含めた、集約公開鍵 $X'$ のデジタル証明書の作成依頼を、パブリックCAサーバ206に送信する（ステップS2208）。

【0181】

一方で、パブリックCAサーバ206は、集約公開鍵 $X'$ のデジタル証明書の作成依頼を受け付けると、集約公開鍵 $X'$ のデジタル証明書を作成し、プライベートCAサーバ204に送信する（ステップS2209）。

【0182】

一方で、プライベートCAサーバ204は、受信した集約公開鍵 $X'$ のデジタル証明書を記憶する（ステップS2210）。次に、プライベートCAサーバ204は、公開鍵 $X_i$ のデジタル証明書を作成して記憶する（ステップS2211）。その後、署名制御システム200は、鍵生成処理を終了する。

【0183】

（文書作成処理手順）

次に、図23を用いて、署名制御システム200が実行する、文書作成処理手順の一例について説明する。文書作成処理は、例えば、図3に示したCPU301と、メモリ302や記録媒体305などの記憶領域と、ネットワークI/F303となどによって実現される。

【0184】

図23は、文書作成処理手順の一例を示すフローチャートである。図23において、署名側端末201は、シードを生成し、文書管理サーバ205に送信する（ステップS2301）。

【0185】

一方で、文書管理サーバ205は、シードを受信する（ステップS2302）。

【0186】

一方で、署名側端末201は、文書 $m_1$ を生成する（ステップS2303）。次に、署名側端末201は、シードと文書 $m_1$ とに基づいて、ハッシュ値 $h_{s1}$ を生成する（ステップS2304）。そして、署名側端末201は、秘密鍵 $x_1$ を取得する（ステップS2305）。

【0187】

次に、署名側端末201は、秘密鍵 $x_1$ に基づいて、文書 $m_1$ に対する署名 $\sigma_1$ を生成する（ステップS2306）。そして、署名側端末201は、文書 $m_1$ と、ハッシュ値 $h_{s1}$ と、署名 $\sigma_1$ とを、文書管理サーバ205に送信する（ステップS2307）。

【0188】

一方で、文書管理サーバ205は、文書 $m_1$ と、ハッシュ値 $h_{s1}$ と、署名 $\sigma_1$ とを、署名側端末201から受信する（ステップS2308）。次に、文書管理サーバ205は、受信したハッシュ値 $h_{s1}$ が、受信したシードと、受信した文書 $m_1$ とに対応するか否かを判定する（ステップS2309）。

【0189】

ここで、受信したハッシュ値 $h_{s1}$ が、受信したシードと、受信した文書 $m_1$ とに対応しない場合（ステップS2309：検証失敗）、文書管理サーバ205は、検証失敗の通知を、署名側端末201に送信する。そして、署名側端末201は、ステップS2312の処理に移行する。一方で、受信したハッシュ値 $h_{s1}$ が、受信したシードと、受信した文書 $m_1$ とに対応する場合（ステップS2309：検証成功）、文書管理サーバ205は、検証成功と判断し、ステップS2310、S2311の処理に移行する。

【0190】

文書管理サーバ205は、公開鍵 $X_1$ を取得する（ステップS2310）。そして、文書管理サーバ205は、公開鍵 $X_1$ に基づいて、受信した署名 $\sigma_1$ が正当であるか否かを判定する（ステップS2311）。

10

20

30

40

50

## 【0191】

ここで、受信した署名 $\sigma_1$ が正当ではない場合（ステップS2311：検証失敗）、文書管理サーバ205は、検証失敗の通知を、署名側端末201に送信する。そして、署名側端末201は、ステップS2312の処理に移行する。一方で、受信した署名 $\sigma_1$ が正当である場合（ステップS2311：検証成功）、文書管理サーバ205は、検証成功と判断し、ステップS2313の処理に移行する。

## 【0192】

ステップS2312では、署名側端末201は、検証失敗の通知を受け付けると、署名 $\sigma_1$ を生成し直す（ステップS2312）。そして、署名側端末201は、ステップS2301の処理に戻る。

## 【0193】

ステップS2313では、文書管理サーバ205は、受信した文書 $m_1$ と、受信したハッシュ値 $h_{s1}$ と、受信した署名 $\sigma_1$ とを、自装置が有する制御情報管理テーブル400に記憶する（ステップS2313）。次に、文書管理サーバ205は、文書 $m_1$ を登録したことの通知を、署名側端末201に送信する（ステップS2314）。

## 【0194】

一方で、署名側端末201は、文書 $m_1$ を登録したことの通知を受信する（ステップS2315）。その後、署名制御システム200は、文書作成処理を終了する。

## 【0195】

（文書承認処理手順）

次に、図24を用いて、署名制御システム200が実行する、文書承認処理手順の一例について説明する。文書承認処理は、例えば、図3に示したCPU301と、メモリ302や記録媒体305などの記憶領域と、ネットワークI/F303となどによって実現される。

## 【0196】

図24は、文書承認処理手順の一例を示すフローチャートである。図24において、文書管理サーバ205は、記憶した文書 $m_1$ と、ハッシュ値 $h_{s1}$ と、署名 $\sigma_1$ とを、承認者に対応する署名側端末202に送信する（ステップS2401）。

## 【0197】

一方で、署名側端末201は、文書 $m_1$ と、ハッシュ値 $h_{s1}$ と、署名 $\sigma_1$ とを受信する（ステップS2402）。次に、署名側端末202は、文書 $m_2$ を生成する（ステップS2403）。そして、署名側端末202は、ハッシュ値 $h_{s1}$ と文書 $m_2$ とに基づいて、ハッシュ値 $h_{s2}$ を生成する（ステップS2404）。

## 【0198】

次に、署名側端末202は、秘密鍵 $x_2$ を取得する（ステップS2405）。そして、署名側端末202は、秘密鍵 $x_2$ に基づいて、文書 $m_2$ に対する署名 $\sigma_2$ を生成する（ステップS2406）。そして、署名側端末202は、文書 $m_2$ と、ハッシュ値 $h_{s2}$ と、署名 $\sigma_2$ とを、文書管理サーバ205に送信する（ステップS2407）。

## 【0199】

一方で、文書管理サーバ205は、文書 $m_2$ と、ハッシュ値 $h_{s2}$ と、署名 $\sigma_2$ とを、署名側端末202から受信する（ステップS2408）。そして、文書管理サーバ205は、受信したハッシュ値 $h_{s2}$ が、ハッシュ値 $h_{s1}$ と、受信した文書 $m_2$ とに対応するか否かを判定する（ステップS2409）。

## 【0200】

ここで、受信したハッシュ値 $h_{s2}$ が、ハッシュ値 $h_{s1}$ と、受信した文書 $m_2$ とに対応しない場合（ステップS2409：検証失敗）、文書管理サーバ205は、検証失敗の通知を、署名側端末202に送信する。そして、署名側端末202は、ステップS2412の処理に移行する。一方で、受信したハッシュ値 $h_{s2}$ が、ハッシュ値 $h_{s1}$ と、受信した文書 $m_2$ とに対応する場合（ステップS2409：検証成功）、文書管理サーバ205は、検証成功と判断し、ステップS2410、S2411の処理に移行する。

10

20

30

40

50

## 【0201】

文書管理サーバ205は、公開鍵 $X_2$ を取得する（ステップS2410）。そして、文書管理サーバ205は、公開鍵 $X_2$ に基づいて、受信した署名 $\sigma_2$ が正当であるか否かを判定する（ステップS2411）。

## 【0202】

ここで、受信した署名 $\sigma_2$ が正当ではない場合（ステップS2411：検証失敗）、文書管理サーバ205は、検証失敗の通知を、署名側端末202に送信する。そして、署名側端末202は、ステップS2412の処理に移行する。一方で、受信した署名 $\sigma_2$ が正当である場合（ステップS2411：検証成功）、文書管理サーバ205は、検証成功と判断し、ステップS2413の処理に移行する。

10

## 【0203】

ステップS2412では、署名側端末202は、検証失敗の通知を受け付けると、署名 $\sigma_2$ を生成し直す（ステップS2412）。そして、署名側端末202は、ステップS2402の処理に戻る。

## 【0204】

ステップS2413では、文書管理サーバ205は、受信した文書 $m_2$ と、受信したハッシュ値 $h_{s_2}$ と、受信した署名 $\sigma_2$ とを、自装置が有する制御情報管理テーブル400に記憶する（ステップS2413）。次に、文書管理サーバ205は、文書 $m_2$ を登録したことの通知を、署名側端末202に送信する（ステップS2414）。

## 【0205】

一方で、署名側端末202は、文書 $m_2$ を登録したことの通知を受信する（ステップS2415）。その後、署名制御システム200は、文書承認処理を終了する。

20

## 【0206】

## （文書決裁処理手順）

次に、図25を用いて、署名制御システム200が実行する、文書決裁処理手順の一例について説明する。文書決裁処理は、例えば、図3に示したCPU301と、メモリ302や記録媒体305などの記憶領域と、ネットワークI/F303となどによって実現される。

## 【0207】

図25は、文書決裁処理手順の一例を示すフローチャートである。図25において、文書管理サーバ205は、記憶した文書 $m_2$ と、ハッシュ値 $h_{s_2}$ と、署名 $\sigma_2$ とを、決裁者に対応する署名側端末203に送信する（ステップS2501）。

30

## 【0208】

次に、署名側端末203は、文書 $m_2$ と、ハッシュ値 $h_{s_2}$ と、署名 $\sigma_2$ とを受信する（ステップS2502）。そして、署名側端末203は、文書 $m_3$ を生成する（ステップS2503）。そして、署名側端末203は、ハッシュ値 $h_{s_2}$ と文書 $m_3$ とに基づいて、ハッシュ値 $h_{s_3}$ を生成する（ステップS2504）。

## 【0209】

次に、署名側端末203は、秘密鍵 $x_3$ を取得する（ステップS2505）。そして、署名側端末203は、秘密鍵 $x_3$ に基づいて、文書 $m_3$ に対する署名 $\sigma_3$ を生成する（ステップS2506）。そして、署名側端末203は、文書 $m_3$ と、ハッシュ値 $h_{s_3}$ と、署名 $\sigma_3$ とを、文書管理サーバ205に送信する（ステップS2507）。

40

## 【0210】

一方で、文書管理サーバ205は、文書 $m_3$ と、ハッシュ値 $h_{s_3}$ と、署名 $\sigma_3$ とを、署名側端末203から受信する（ステップS2508）。そして、文書管理サーバ205は、受信したハッシュ値 $h_{s_3}$ が、ハッシュ値 $h_{s_2}$ と、受信した文書 $m_3$ とに対応するか否かを判定する（ステップS2509）。

## 【0211】

ここで、受信したハッシュ値 $h_{s_3}$ が、ハッシュ値 $h_{s_2}$ と、受信した文書 $m_3$ とに対応しない場合（ステップS2509：検証失敗）、文書管理サーバ205は、検証失敗の通

50

知を、署名側端末203に送信する。そして、署名側端末203は、ステップS2512の処理に移行する。一方で、受信したハッシュ値 $h_{s3}$ が、ハッシュ値 $h_{s2}$ と、受信した文書 $m_3$ とに対応する場合（ステップS2509：検証成功）、文書管理サーバ205は、検証成功と判断し、ステップS2510、S2511の処理に移行する。

【0212】

文書管理サーバ205は、公開鍵 $X_3$ を取得する（ステップS2510）。そして、文書管理サーバ205は、公開鍵 $X_3$ に基づいて、受信した署名 $\sigma_3$ が正当であるか否かを判定する（ステップS2511）。

【0213】

ここで、受信した署名 $\sigma_3$ が正当ではない場合（ステップS2511：検証失敗）、文書管理サーバ205は、検証失敗の通知を、署名側端末203に送信する。そして、署名側端末203は、ステップS2512の処理に移行する。一方で、受信した署名 $\sigma_3$ が正当である場合（ステップS2511：検証成功）、文書管理サーバ205は、検証成功と判断し、ステップS2513の処理に移行する。

【0214】

ステップS2512では、署名側端末203は、検証失敗の通知を受け付けると、署名 $\sigma_3$ を生成し直す（ステップS2512）。そして、署名側端末203は、ステップS2502の処理に戻る。

【0215】

ステップS2513では、文書管理サーバ205は、受信した文書 $m_3$ と、受信したハッシュ値 $h_{s3}$ と、受信した署名 $\sigma_3$ とを、自装置が有する制御情報管理テーブル400に記憶する（ステップS2513）。次に、文書管理サーバ205は、受信した文書 $m_3$ に基づいて、最終版の文書 $m_f$ を生成し、自装置が有する制御情報管理テーブル400に記憶する（ステップS2514）。そして、文書管理サーバ205は、文書 $m_3$ を登録したことの通知を、署名側端末203に送信する（ステップS2515）。

【0216】

一方で、署名側端末203は、文書 $m_3$ を登録したことの通知を受信する（ステップS2516）。そして、署名制御システム200は、文書決裁処理を終了する。

【0217】

（署名集約処理手順）

次に、図26を用いて、署名制御システム200が実行する、署名集約処理手順の一例について説明する。署名集約処理は、例えば、図3に示したCPU301と、メモリ302や記録媒体305などの記憶領域と、ネットワークI/F303となどによって実現される。

【0218】

図26は、署名集約処理手順の一例を示すフローチャートである。図26において、文書管理サーバ205は、文書を送信することの通知を、検証側端末207に送信する（ステップS2601）。一方で、検証側端末207は、通知を受け付けると、組織の公開鍵証明書の要求を、文書管理サーバ205に送信する（ステップS2602）。

【0219】

次に、文書管理サーバ205は、登録済みの公開鍵 $X'$ のデジタル証明書を、検証側端末207に送信する（ステップS2603）。一方で、検証側端末207は、受信した公開鍵 $X'$ のデジタル証明書を検証する（ステップS2604）。検証側端末207は、検証失敗の場合、検証失敗の通知を、文書管理サーバ205に送信する。そして、検証側端末207は、ステップS2609の処理に移行する。

【0220】

一方で、文書管理サーバ205は、検証失敗の通知を受け付けた場合には、公開鍵 $X'$ が有効であるか否かを確認する（ステップS2605）。文書管理サーバ205は、公開鍵 $X'$ が有効でなければ、公開鍵 $X'$ を生成し直してもよい。

【0221】

10

20

30

40

50

次に、文書管理サーバ205は、ハッシュ値 $h_{s_i}$ と、署名情報 $\sigma_i$ とを取得する（ステップS2606）。そして、文書管理サーバ205は、ハッシュ値 $h_{s_i}$ と、署名情報 $\sigma_i$ とに基づいて、集約署名情報 $\sigma'$ を生成する（ステップS2607）。そして、文書管理サーバ205は、最終版の文書 $m_f$ と、生成した集約署名情報 $\sigma'$ とを対応付けて、検証側端末207に送信する（ステップS2608）。

【0222】

一方で、検証側端末207は、ハッシュチェーンを検証する（ステップS2609）。検証側端末207は、検証成功であれば、ステップS2610の処理に移行する。検証側端末207は、検証失敗であれば、検証失敗の通知を、文書管理サーバ205に送信する。

【0223】

ステップS2610では、検証側端末207は、集約署名情報 $\sigma'$ を検証する（ステップS2610）。検証側端末207は、検証成功であれば、ステップS2611の処理に移行する。検証側端末207は、検証失敗であれば、検証失敗の通知を、文書管理サーバ205に送信する。

【0224】

ステップS2611では、検証側端末207は、文書 $m_n$ について検証成功したと認識する（ステップS2611）。

【0225】

一方で、文書管理サーバ205は、検証失敗の通知を受け付けると、Re-Verifyを利用して、図27に後述する再検証処理を実行することにより、原因調査を実施する（ステップS2612）。そして、署名制御システム200は、署名集約処理を終了する。

【0226】

（再検証処理手順）

次に、図27を用いて、文書管理サーバ205が実行する、再検証処理手順の一例について説明する。再検証処理は、例えば、図3に示したCPU301と、メモリ302や記録媒体305などの記憶領域と、ネットワークI/F303とによって実現される。

【0227】

図27は、再検証処理手順の一例を示すフローチャートである。図27において、文書管理サーバ205は、集約署名情報 $s_{1,1}$ を検証し、検証失敗したことを確認する（ステップS2701）。

【0228】

次に、文書管理サーバ205は、集約署名情報 $s_{2,1}$ を検証する（ステップS2702）。文書管理サーバ205は、検証成功であれば、ステップS2703の処理に移行する。一方で、文書管理サーバ205は、検証失敗であれば、ステップS2704の処理に移行する。

【0229】

ステップS2703では、文書管理サーバ205は、集約署名情報 $s_{2,2}$ を検証する（ステップS2703）。文書管理サーバ205は、検証成功であれば、不正な署名が存在しないと判断し、再検証処理を終了する。一方で、文書管理サーバ205は、検証失敗であれば、ステップS2704以降の処理と同様の処理を実行する。

【0230】

ステップS2704では、文書管理サーバ205は、集約署名情報 $s_{3,1}$ を検証する（ステップS2704）。文書管理サーバ205は、検証成功であれば、ステップS2705の処理に移行する。一方で、文書管理サーバ205は、検証失敗であれば、ステップS2706の処理に移行する。

【0231】

ステップS2705では、文書管理サーバ205は、集約署名情報 $s_{3,2}$ を検証する（ステップS2705）。文書管理サーバ205は、検証成功であれば、不正な署名が存在しないと判断し、再検証処理を終了する。一方で、文書管理サーバ205は、検証失敗であれば、ステップS2706以降の処理と同様の処理を実行する。

10

20

30

40

50

## 【0232】

ステップS2706では、文書管理サーバ205は、集約署名情報 $s_{4,1}$ を検証する（ステップS2706）。文書管理サーバ205は、検証成功、および、検証失敗に応じて、集約署名情報 $s_{x \rangle 4,y}$ を検証する処理を、再帰的に実行することになる。その後、文書管理サーバ205は、ステップS2707の処理に移行する。

## 【0233】

ステップS2707では、文書管理サーバ205は、検証失敗した署名 $\sigma_i$ を特定し、検証失敗した署名 $\sigma_i$ を再生成し、集約署名情報 $\sigma'$ を再生成する（ステップS2707）。次に、文書管理サーバ205は、再生成した集約署名情報 $\sigma'$ を出力する（ステップS2708）。そして、文書管理サーバ205は、再検証処理を終了する。

10

## 【0234】

（別の署名集約処理手順）

次に、図28を用いて、署名制御システム200が実行する、別の署名集約処理手順の一例について説明する。別の署名集約処理は、例えば、図3に示したCPU301と、メモリ302や記録媒体305などの記憶領域と、ネットワークI/F303となどによって実現される。

## 【0235】

図28は、別の署名集約処理手順の一例を示すフローチャートである。図28において、文書管理サーバ205は、文書を送信することの通知を、検証側端末207に送信する（ステップS2801）。一方で、検証側端末207は、通知を受け付けると、組織の公開鍵証明書の要求を、文書管理サーバ205に送信する（ステップS2802）。

20

## 【0236】

次に、文書管理サーバ205は、登録済みの公開鍵 $X'$ のデジタル証明書を、検証側端末207に送信する（ステップS2803）。一方で、検証側端末207は、受信した公開鍵 $X'$ のデジタル証明書を検証する（ステップS2804）。検証側端末207は、検証失敗の場合、検証失敗の通知を、文書管理サーバ205に送信する。そして、検証側端末207は、ステップS2810の処理に移行する。

## 【0237】

一方で、文書管理サーバ205は、検証失敗の通知を受け付けた場合には、公開鍵 $X'$ が有効であるか否かを確認する（ステップS2805）。文書管理サーバ205は、公開鍵 $X'$ が有効でなければ、公開鍵 $X'$ を生成し直してもよい。

30

## 【0238】

次に、文書管理サーバ205は、ハッシュ値 $h_{s_i}$ と、署名情報 $\sigma_i$ とを取得する（ステップS2806）。そして、文書管理サーバ205は、ハッシュ値 $h_{s_i}$ と、署名情報 $\sigma_i$ とに基づいて、集約署名情報 $\sigma'$ を生成する（ステップS2807）。

## 【0239】

次に、文書管理サーバ205は、図29に後述する無効化処理を実行する（ステップS2808）。そして、文書管理サーバ205は、最終版の文書 $m_n$ と、生成した集約署名情報 $\sigma'$ とを対応付けて、検証側端末207に送信する（ステップS2809）。

## 【0240】

一方で、検証側端末207は、ハッシュチェーンを検証する（ステップS2810）。検証側端末207は、検証成功であれば、ステップS2811の処理に移行する。検証側端末207は、検証失敗であれば、検証失敗の通知を、文書管理サーバ205に送信する。

40

## 【0241】

ステップS2811では、検証側端末207は、集約署名情報 $\sigma'$ を検証する（ステップS2811）。検証側端末207は、検証成功であれば、ステップS2812の処理に移行する。検証側端末207は、検証失敗であれば、検証失敗の通知を、文書管理サーバ205に送信する。

## 【0242】

ステップS2812では、検証側端末207は、文書 $m_n$ について検証成功したと認識す

50

る（ステップS2812）。

【0243】

一方で、文書管理サーバ205は、検証失敗の通知を受け付けると、Re-Verifyを利用して、図27に示した再検証処理を実行することにより、原因調査を実施する（ステップS2813）。そして、署名制御システム200は、署名集約処理を終了する。

【0244】

（無効化処理手順）

次に、図29を用いて、文書管理サーバ205が実行する、無効化処理手順の一例について説明する。無効化処理は、例えば、図3に示したCPU301と、メモリ302や記録媒体305などの記憶領域と、ネットワークI/F303とによって実現される。

【0245】

図29は、無効化処理手順の一例を示すフローチャートである。図29において、文書管理サーバ205は、一時的に無効化したい公開鍵が存在するか否かを判定する（ステップS2901）。ここで、一時的に無効化したい公開鍵が存在しない場合（ステップS2901：No）、文書管理サーバ205は、無効化処理を終了する。一時的に無効化したい公開鍵が存在する場合（ステップS2901：Yes）、文書管理サーバ205は、ステップS2902の処理に移行する。

【0246】

ステップS2902では、文書管理サーバ205は、無効化する対象となる公開鍵を、プライベートCAサーバ204に問い合わせる（ステップS2902）。一方で、プライベートCAサーバ204は、問い合わせに応じて、無効化する対象となる公開鍵を、文書管理サーバ205に送信する（ステップS2903）。

【0247】

文書管理サーバ205は、無効化する対象となる公開鍵に基づいて、集約署名情報σ'を修正した修正署名情報σ''を生成する（ステップS2904）。次に、文書管理サーバ205は、修正署名情報σ''を出力する（ステップS2905）。そして、文書管理サーバ205は、無効化処理を終了する。

【0248】

以上説明したように、情報処理装置100によれば、複数の文書情報と、複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得することができる。情報処理装置100によれば、取得した複数の文書情報に基づいて、取得した複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成することができる。情報処理装置100によれば、生成した集約署名情報を、複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報と、複数の文書情報とに対応付けて出力することができる。これにより、情報処理装置100は、集約署名情報を検証する際に用いる鍵情報が1つであっても、集約署名情報を検証可能にすることができ、集約署名情報を検証する際にかかるコストの低減化を図ることができる。

【0249】

情報処理装置100によれば、取得した複数の文書情報のそれぞれの文書情報に対応する公開鍵情報に基づいて、集約公開鍵情報を生成することができる。これにより、情報処理装置100は、集約署名情報を検証する際に用いる鍵情報の数の低減化を図ることができる。集約署名情報を検証する際にかかるコストの低減化を図ることができる。

【0250】

情報処理装置100によれば、取得した複数の文書情報のそれぞれの文書情報に対応するハッシュ値に基づいて、集約署名情報を生成することができる。これにより、情報処理装置100は、ハッシュ値に基づく署名情報を集約した集約署名情報を生成することができる。

【0251】

情報処理装置100によれば、生成した集約署名情報の正当性を、取得した複数の文書情報と、生成した集約署名情報と、生成した集約公開鍵情報とに基づいて検証可能にする

10

20

30

40

50

ことができる。これにより、情報処理装置１００は、集約署名情報を検証する際に用いる鍵情報が１つであっても、集約署名情報を検証可能にすることができ、集約署名情報を検証する際にかかるコストの低減化を図ることができる。

【０２５２】

情報処理装置１００によれば、複数の文書情報に対応付けられた、集約署名情報と、集約公開鍵情報とを受け付けることができる。情報処理装置１００によれば、複数の文書情報と、受け付けた集約署名情報と、受け付けた集約公開鍵情報とに基づいて、受け付けた集約署名情報の正当性を検証することができる。これにより、情報処理装置１００は、集約署名情報の正当性を検証し、複数の文書情報が改ざんされているか否かを確認することができ、セキュリティの向上を図ることができる。

10

【０２５３】

情報処理装置１００によれば、集約署名情報が正当ではないことを検出することができる。情報処理装置１００によれば、集約署名情報の集約元となる署名情報のうち、一部の署名情報を集約した集約署名情報と、一部の署名情報以外の署名情報を集約した集約署名情報との２つの集約署名情報を取得することができる。情報処理装置１００によれば、取得した２つの集約署名情報のそれぞれの集約署名情報の正当性を検証した結果を出力することができる。これにより、情報処理装置１００は、集約署名情報が正当ではない原因となる、集約元となるいずれかの署名情報を特定することができる。

【０２５４】

情報処理装置１００によれば、取得した２つの集約署名情報の少なくともいずれかの集約署名情報が正当ではないことを検出することができる。情報処理装置１００によれば、いずれかの集約署名情報の集約元となる署名情報のうち、一部の署名情報を集約した集約署名情報と、一部の署名情報以外の署名情報を集約した集約署名情報との２つの集約署名情報を新たに取得することができる。情報処理装置１００によれば、新たに取得した２つの集約署名情報のそれぞれの集約署名情報の正当性を検証した結果を出力することができる。これにより、情報処理装置１００は、集約署名情報が正当ではない原因となる、集約元となるいずれかの署名情報を特定することができる。

20

【０２５５】

情報処理装置１００によれば、取得した複数の文書情報のそれぞれの文書情報に対応する公開鍵情報と、取得した複数の文書情報のいずれの文書情報にも対応しない公開鍵情報とを集約した集約公開鍵情報を生成することができる。情報処理装置１００によれば、いずれの文書情報にも対応しない公開鍵情報に基づいて、生成した集約署名情報を修正した修正署名情報を生成することができる。情報処理装置１００によれば、生成した修正署名情報を、集約公開鍵情報と、取得した複数の文書情報とに対応付けて出力することができる。これにより、情報処理装置１００は、いずれかの公開鍵情報に基づく署名情報が存在しなくても、集約公開鍵情報に基づき検証可能な修正署名情報を生成することができる。

30

【０２５６】

情報処理装置１００によれば、取得した複数の文書情報のそれぞれの文書情報に対応する署名情報を生成する際に用いられた秘密鍵情報に対応する公開鍵情報に基づいて、集約公開鍵情報を生成することができる。これにより、情報処理装置１００は、署名情報を検証可能な公開鍵情報を集約し、集約公開鍵情報を生成することができる。

40

【０２５７】

情報処理装置１００によれば、複数の文書情報として、第１の文書の複数の版のそれぞれの版を示す文書情報を採用することができる。これにより、情報処理装置１００は、文書が改版されていく状況に適用することができる。

【０２５８】

情報処理装置１００によれば、シードと、第１の文書のそれぞれの版に対応するハッシュ値とから形成されるハッシュチェーンに基づいて生成されるハッシュ値を、集約署名情報に対応付けて出力することができる。これにより、情報処理装置１００は、複数の文書情報が改ざんされているか否かを確認可能にすることができ、セキュリティの向上を図る

50

ことができる。

【0259】

情報処理装置100によれば、安全素数と、取得した複数の文書情報のそれぞれの文書情報に対応するハッシュ値とに基づいて、集約署名情報を生成することができる。これにより、情報処理装置100は、集約署名情報を生成する際にかかる処理時間の低減化を図ることができる。

【0260】

情報処理装置100によれば、シェノア署名アルゴリズムを利用して、集約署名情報を生成することができる。これにより、情報処理装置100は、シェノア署名アルゴリズムに基づく署名情報を集約した集約署名情報を生成することができる。

10

【0261】

情報処理装置100によれば、生成した集約署名情報を、集約公開鍵情報と、取得した複数の文書情報とに対応付けて、取得した複数の文書情報についての検証者宛てに出力することができる。これにより、情報処理装置100は、検証者が、集約署名情報の正当性を検証可能にし、複数の文書情報が改ざんされているか否かを確認可能にし、セキュリティを向上させることができる。

【0262】

情報処理装置100によれば、予め定められた署名者ごとに、文書情報と、公開鍵情報と、署名情報とを対応付けておくことができる。これにより、情報処理装置100は、集約署名情報と、集約公開鍵情報とを生成し易くすることができる。

20

【0263】

情報処理装置100によれば、複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成する装置から、複数の文書情報に対応付けられた、集約署名情報を受け付けることができる。情報処理装置100によれば、複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報を生成する装置から、複数の文書情報に対応付けられた、集約公開鍵情報を受け付けることができる。情報処理装置100によれば、複数の文書情報と、受け付けた集約署名情報と、受け付けた集約公開鍵情報とに基づいて、受け付けた集約署名情報の正当性を検証することができる。これにより、情報処理装置100は、集約署名情報の正当性を検証し、複数の文書情報が改ざんされているか否かを確認することができ、セキュリティの向上を図ることができる。

30

【0264】

なお、本実施の形態で説明した署名制御方法は、予め用意されたプログラムをPCやワークステーションなど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。本実施の形態で説明した署名制御プログラムは、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。記録媒体は、ハードディスク、フレキシブルディスク、CD (Compact Disc) - ROM、MO (Magnet Optical disc)、DVD (Digital Versatile Disc) などである。また、本実施の形態で説明した署名制御プログラムは、インターネットなどのネットワークを介して配布してもよい。

【0265】

上述した実施の形態に関し、さらに以下の付記を開示する。

40

【0266】

(付記1) 複数の文書情報と、前記複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得し、

取得した前記複数の文書情報に基づいて、取得した前記複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成し、

生成した前記集約署名情報を、前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報と、前記複数の文書情報とに対応付けて出力する、

処理をコンピュータが実行することを特徴とする署名制御方法。

【0267】

50

(付記 2) 取得した前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報に基づいて、前記集約公開鍵情報を生成する、

処理を前記コンピュータが実行することを特徴とする付記 1 に記載の署名制御方法。

【0268】

(付記 3) 前記集約署名情報を生成する処理は、

取得した前記複数の文書情報のそれぞれの文書情報に対応するハッシュ値に基づいて、前記集約署名情報を生成する、ことを特徴とする付記 1 または 2 に記載の署名制御方法。

【0269】

(付記 4) 生成した前記集約署名情報の正当性は、取得した前記複数の文書情報と、生成した前記集約署名情報と、生成した前記集約公開鍵情報とに基づいて検証可能である、ことを特徴とする付記 1～3 のいずれか一つに記載の署名制御方法。

【0270】

(付記 5) 複数の文書情報に対応付けられた、前記複数の文書情報に対応する複数の署名情報を集約した集約署名情報と、前記複数の文書情報に対応する複数の公開鍵情報を集約した集約公開鍵情報とを受け付けた場合、前記複数の文書情報と、受け付けた前記集約署名情報と、受け付けた前記集約公開鍵情報とに基づいて、受け付けた前記集約署名情報の正当性を検証する、

処理を前記コンピュータが実行することを特徴とする付記 1～4 のいずれか一つに記載の署名制御方法。

【0271】

(付記 6) 前記複数の文書情報と対応付けて出力した前記集約署名情報が正当ではないと判定された場合、前記集約署名情報の集約元となる署名情報のうち、一部の署名情報を集約した集約署名情報と、前記一部の署名情報以外の署名情報を集約した集約署名情報との 2 つの集約署名情報を取得し、

取得した前記 2 つの集約署名情報のそれぞれの集約署名情報の正当性を検証した結果を出力する、

処理を前記コンピュータが実行することを特徴とする付記 1～5 のいずれか一つに記載の署名制御方法。

【0272】

(付記 7) 取得した前記 2 つの集約署名情報の少なくともいずれかの集約署名情報が正当ではないと判定された場合、前記いずれかの集約署名情報の集約元となる署名情報のうち、一部の署名情報を集約した集約署名情報と、前記一部の署名情報以外の署名情報を集約した集約署名情報との 2 つの集約署名情報を新たに取得し、

新たに取得した前記 2 つの集約署名情報のそれぞれの集約署名情報の正当性を検証した結果を出力する、

処理を前記コンピュータが実行することを特徴とする付記 6 に記載の署名制御方法。

【0273】

(付記 8) 取得した前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報と、取得した前記複数の文書情報のいずれの文書情報にも対応しない公開鍵情報とを集約した集約公開鍵情報を生成し、

前記いずれの文書情報にも対応しない公開鍵情報に基づいて、生成した前記集約署名情報を修正した修正署名情報を生成し、

生成した前記修正署名情報を、前記集約公開鍵情報と、取得した前記複数の文書情報とに対応付けて出力する、

処理を前記コンピュータが実行することを特徴とする付記 1～7 のいずれか一つに記載の署名制御方法。

【0274】

(付記 9) 前記集約公開鍵情報を生成する処理は、

取得した前記複数の文書情報のそれぞれの文書情報に対応する署名情報を生成する際に用いられた秘密鍵情報に対応する公開鍵情報に基づいて、前記集約公開鍵情報を生成する

10

20

30

40

50

、ことを特徴とする付記 2 に記載の署名制御方法。

【0275】

(付記 10) 前記複数の文書情報は、第 1 の文書の複数の版のそれぞれの版を示す文書情報を含む、ことを特徴とする付記 1 ～ 9 のいずれか一つに記載の署名制御方法。

【0276】

(付記 11) 前記出力する処理は、

シードと、前記第 1 の文書のそれぞれの版に対応するハッシュ値とから形成されるハッシュチェーンに基づいて生成されるハッシュ値を、前記集約署名情報に対応付けて出力する、ことを特徴とする付記 10 に記載の署名制御方法。

【0277】

(付記 12) 前記集約署名情報を生成する処理は、

安全素数と、取得した前記複数の文書情報のそれぞれの文書情報に対応するハッシュ値とに基づいて、前記集約署名情報を生成する、ことを特徴とする付記 1 ～ 11 のいずれか一つに記載の署名制御方法。

【0278】

(付記 13) 前記集約署名情報を生成する処理は、

シェノア署名アルゴリズムを利用して、前記集約署名情報を生成する、ことを特徴とする付記 1 ～ 12 のいずれか一つに記載の署名制御方法。

【0279】

(付記 14) 前記出力する処理は、

生成した前記集約署名情報を、前記集約公開鍵情報と、取得した前記複数の文書情報とに対応付けて、取得した前記複数の文書情報についての検証者宛てに出力する、ことを特徴とする付記 1 ～ 13 のいずれか一つに記載の署名制御方法。

【0280】

(付記 15) 予め定められた署名者ごとに、文書情報と、公開鍵情報と、署名情報とが対応付けられる、ことを特徴とする付記 1 ～ 14 のいずれか一つに記載の署名制御方法。

【0281】

(付記 16) 複数の文書情報と、前記複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得し、

取得した前記複数の文書情報に基づいて、取得した前記複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成し、

生成した前記集約署名情報を、前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報と、前記複数の文書情報とに対応付けて出力する、

処理をコンピュータに実行させることを特徴とする署名制御プログラム。

【0282】

(付記 17) 複数の文書情報と、前記複数の文書情報のそれぞれの文書情報に対応する署名情報とを取得し、

取得した前記複数の文書情報に基づいて、取得した前記複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成し、

生成した前記集約署名情報を、前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報と、前記複数の文書情報とに対応付けて出力する、

制御部を有することを特徴とする情報処理装置。

【0283】

(付記 18) 複数の文書情報のそれぞれの文書情報に対応する署名情報を集約した集約署名情報を生成する装置から、前記複数の文書情報に対応付けられた、前記集約署名情報を受け付け、

前記複数の文書情報のそれぞれの文書情報に対応する公開鍵情報を集約した集約公開鍵情報を生成する装置から、前記複数の文書情報に対応付けられた、前記集約公開鍵情報を受け付け、

前記複数の文書情報と、受け付けた前記集約署名情報と、受け付けた前記集約公開鍵情

10

20

30

40

50

報とに基づいて、受け付けた前記集約署名情報の正当性を検証する、  
処理をコンピュータが実行することを特徴とする署名制御方法。

【符号の説明】

【0284】

|                |             |    |
|----------------|-------------|----|
| 100            | 情報処理装置      |    |
| 110            | 集約署名情報      |    |
| 120            | 公開鍵束        |    |
| 121            | 集約公開鍵情報     |    |
| 200            | 署名制御システム    |    |
| 201～203        | 署名側端末       | 10 |
| 204            | プライベートCAサーバ |    |
| 205            | 文書管理サーバ     |    |
| 206            | パブリックCAサーバ  |    |
| 207            | 検証側端末       |    |
| 210            | ネットワーク      |    |
| 300, 500       | バス          |    |
| 301, 501       | CPU         |    |
| 302, 502       | メモリ         |    |
| 303, 503       | ネットワークI/F   |    |
| 304, 504       | 記録媒体I/F     | 20 |
| 305, 505       | 記録媒体        |    |
| 400            | 制御情報管理テーブル  |    |
| 600            | 第1記憶部       |    |
| 601            | 第1取得部       |    |
| 602            | 第1生成部       |    |
| 603            | 修正部         |    |
| 604            | 解析部         |    |
| 605            | 第2生成部       |    |
| 606            | 第1出力部       |    |
| 610            | 第2記憶部       | 30 |
| 611            | 第2取得部       |    |
| 612            | 検証部         |    |
| 613            | 第2出力部       |    |
| 700, 800       | 鍵ペア生成部      |    |
| 701, 801, 901  | データ受信部      |    |
| 702            | シード値生成部     |    |
| 703, 802       | 文書作成部       |    |
| 704, 803       | ハッシュ値計算部    |    |
| 705, 804       | 署名生成部       |    |
| 706, 805, 1102 | データ送信部      | 40 |
| 710, 810       | 秘密鍵記憶部      |    |
| 720, 820       | 公開鍵記憶部      |    |
| 900            | 組織公開鍵記憶部    |    |
| 902            | ハッシュチェーン検証部 |    |
| 903            | 集約署名検証部     |    |
| 910            | 集約署名付き文書記憶部 |    |
| 1001           | 集約公開鍵生成部    |    |
| 1002           | 公開鍵登録申請部    |    |
| 1010           | 個人公開鍵記憶部    |    |
| 1101           | 集約署名生成部     | 50 |

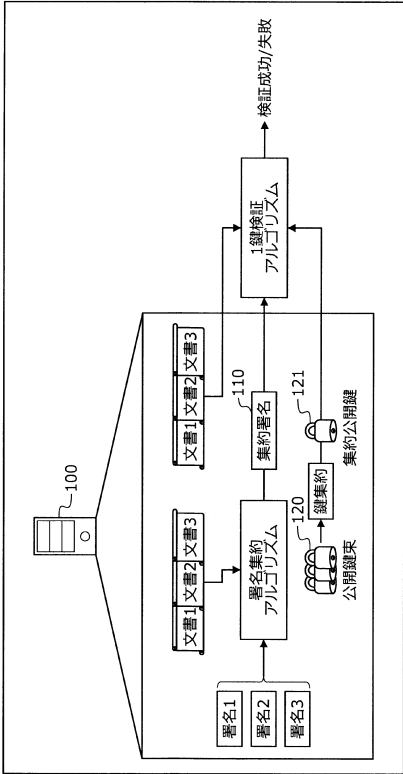
1 1 1 0 署名付き文書記憶部  
1 4 0 0, 1 5 0 0 ハッシュチェーン  
1 7 1 1 ~ 1 7 1 n, 1 7 2 0, 1 7 3 1 ~ 1 7 3 n, 1 7 4 1 ~ 1 7 4 n, 1 7 5 0

数式

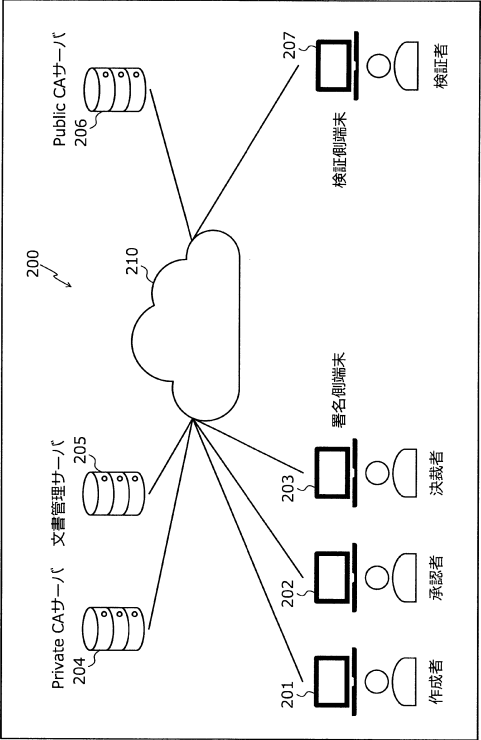
1 8 0 0, 2 1 0 0 表

【図面】

【図 1】



【図 2】



10

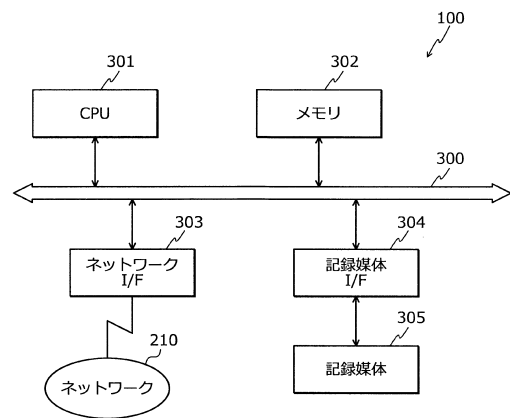
20

30

40

50

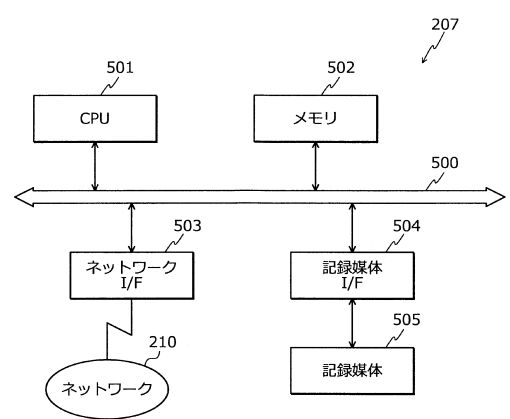
【図 3】



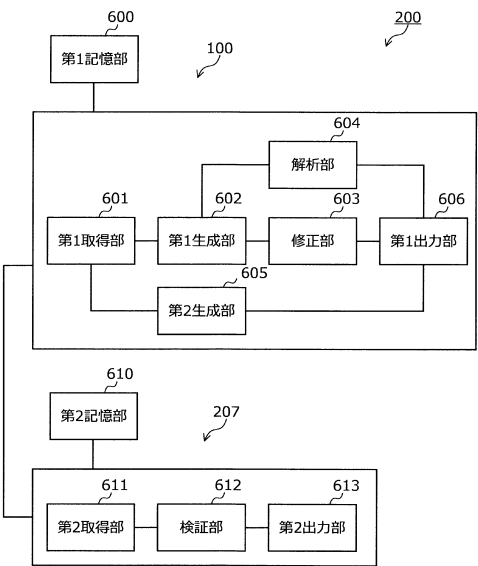
【図 4】

| 署名者 | 文書情報                                                                                                        | 署名対象<br>(文書のハッシュ値) | 秘密鍵   | 公開鍵    | 署名情報       |
|-----|-------------------------------------------------------------------------------------------------------------|--------------------|-------|--------|------------|
|     | $m_1$ : 上記の通り契約を結びます。<br>文書: Alice                                                                          | $hs_1$             | $x_1$ | $x'_1$ | $\sigma_1$ |
|     | $m_2$ : 上記の通り契約を結びます。<br>文書: Alice, Carol                                                                   | $hs_2$             | $x_2$ | $x'_2$ | $\sigma_2$ |
|     | $m_3$ : 上記の通り契約を結びます。<br>文書: Alice, Carol<br>決裁: Charlie                                                    | $hs_3$             | $x_3$ | $x'_3$ | $\sigma_3$ |
|     | $m_f$ : 上記の通り契約を結びます。<br>文書: Alice, Carol<br>決裁: Charlie<br>+ ハッシュチェーン情報:<br>$seed, hs_1, hs_2, chk_{hash}$ | $hs_f$             | ※     | $x'$   | $\sigma'$  |

【図 5】



【図 6】



10

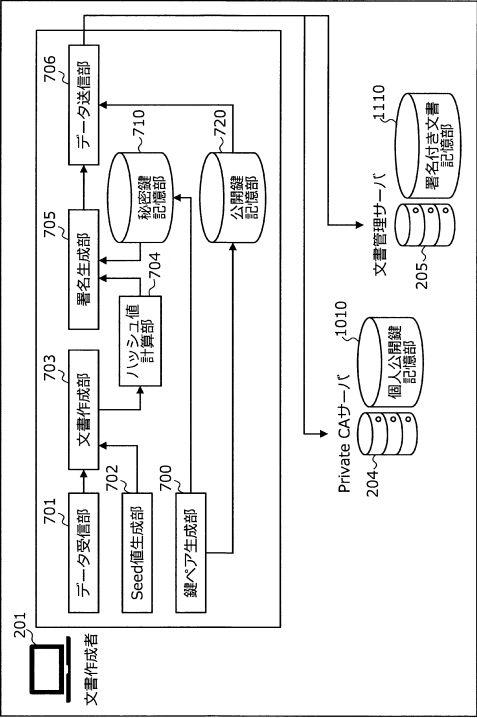
20

30

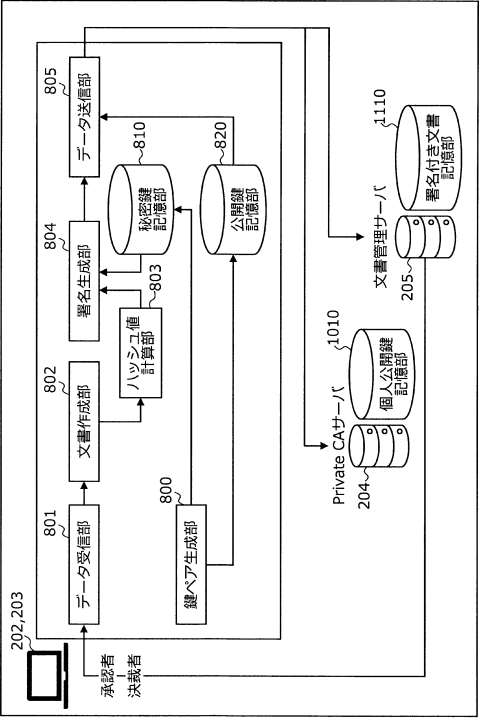
40

50

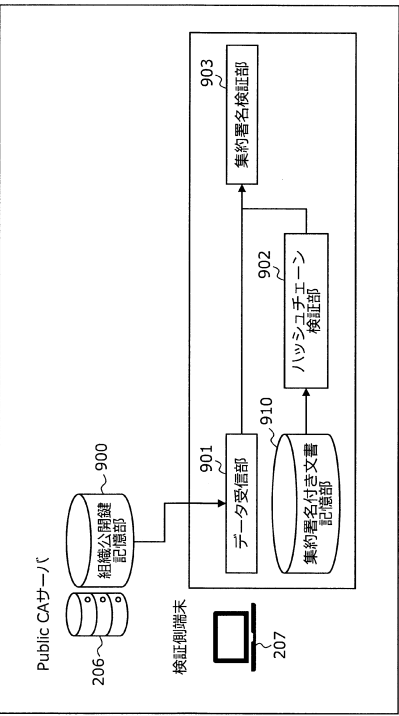
【図 7】



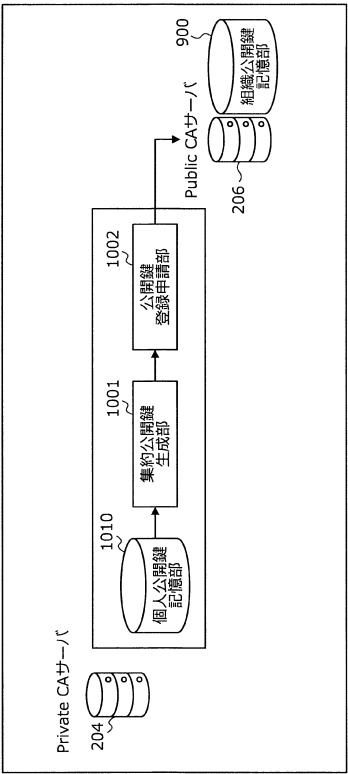
【図 8】



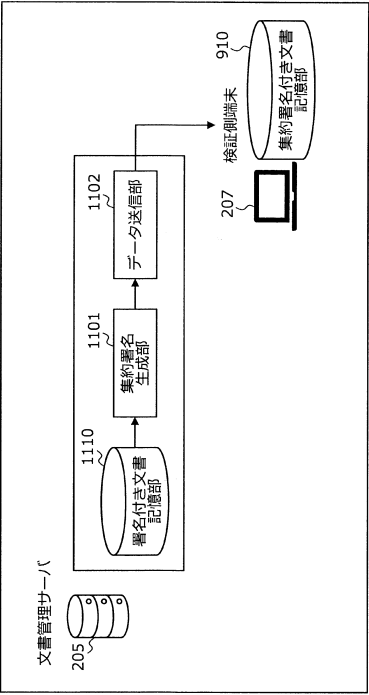
【図 9】



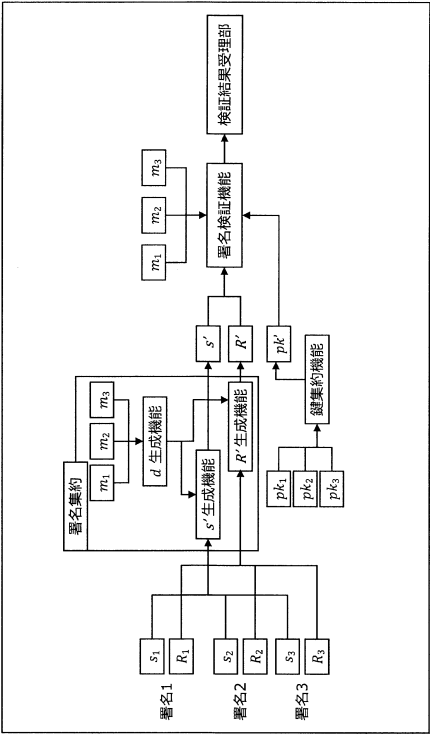
【図 10】



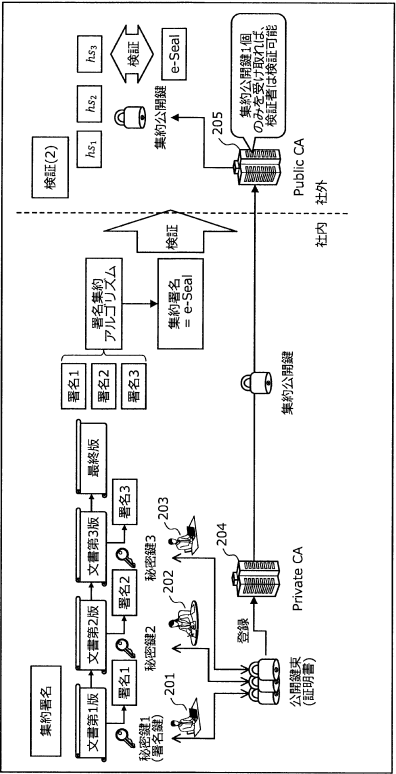
【図 1 1】



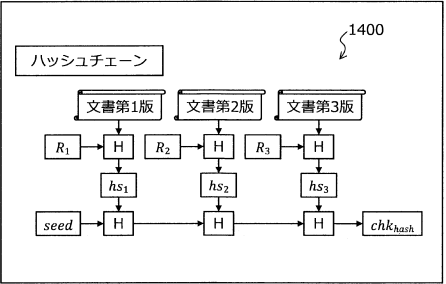
【図 1 2】



【図 1 3】



【図 1 4】



10

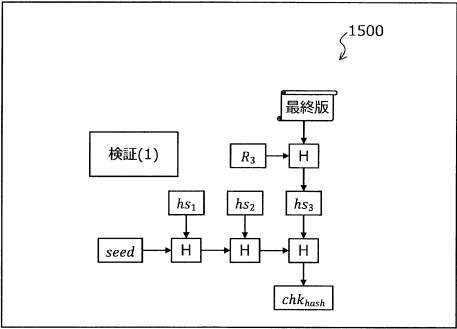
20

30

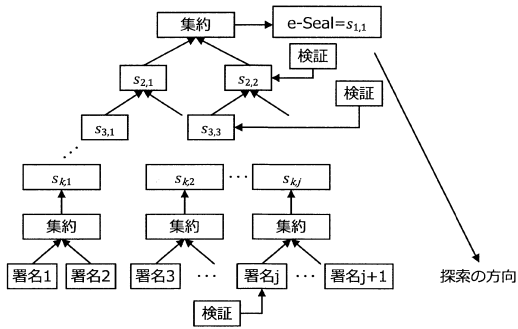
40

50

【図 15】

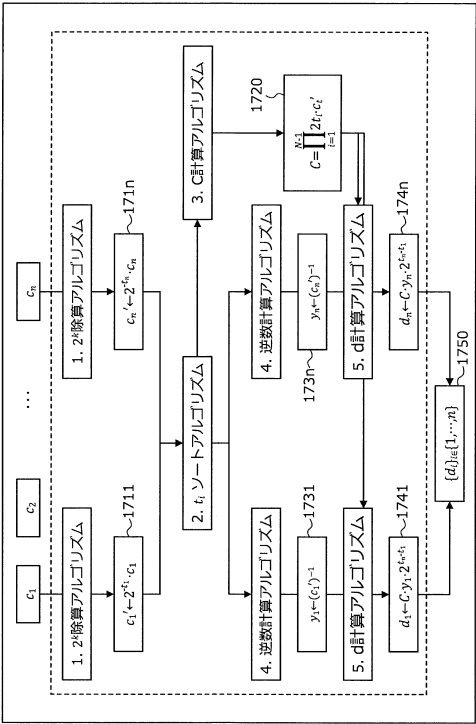


【図 16】



10

【図 17】



【図 18】

|         |          |         |         |         |
|---------|----------|---------|---------|---------|
| 署名数     | 500      | 1000    | 1500    | 2000    |
| 手法(比較例) | 2.31[s]  | 9.38[s] | 20.3[s] | 37.1[s] |
| 手法(図17) | 0.785[s] | 1.71[s] | 2.69[s] | 3.45[s] |

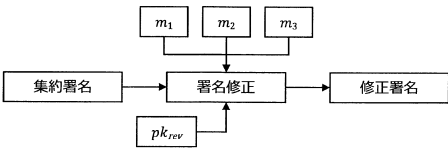
20

30

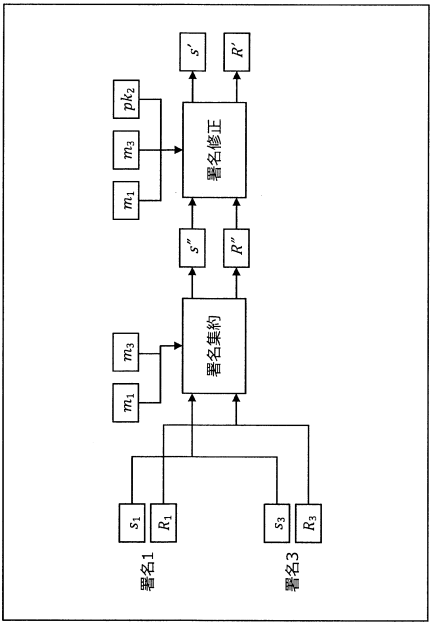
40

50

【図 19】



【図 20】



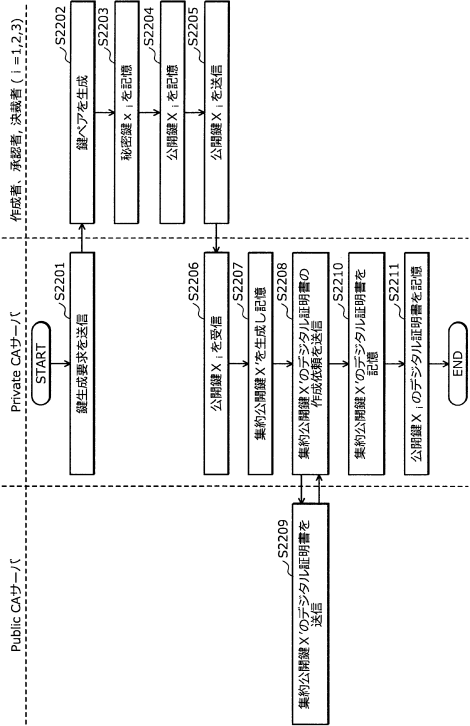
10

20

【図 21】

| 2100       |    |          |
|------------|----|----------|
| 登録公開鍵数     | 従来 | 署名制御システム |
| Private CA | N  | N        |
| Public CA  | N  | 1        |

【図 22】

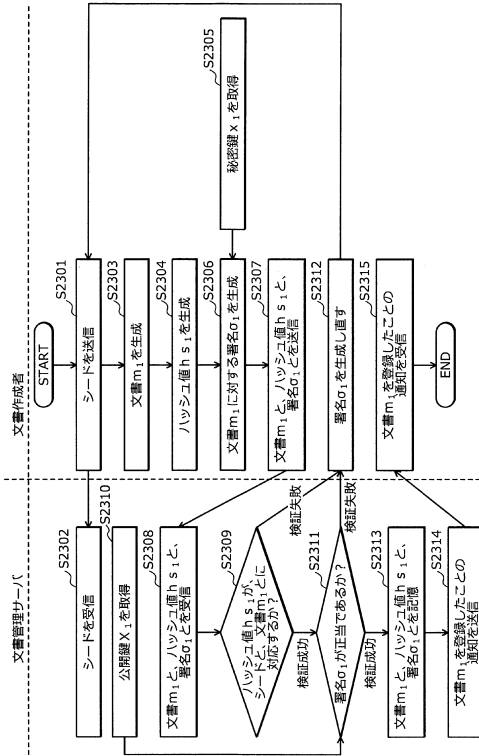


30

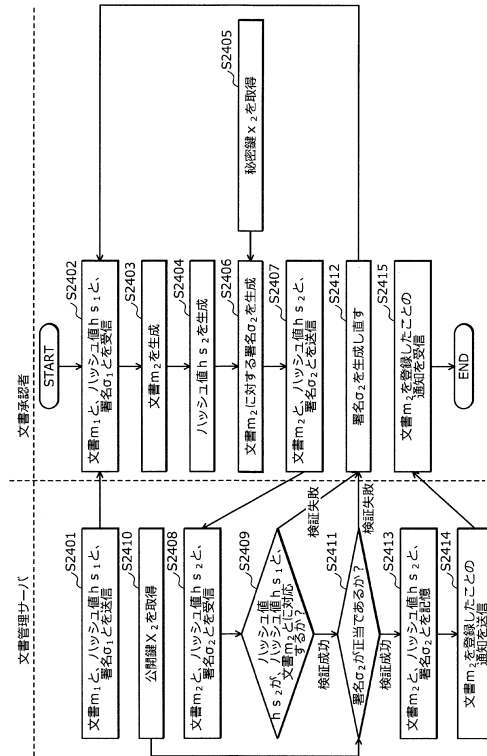
40

50

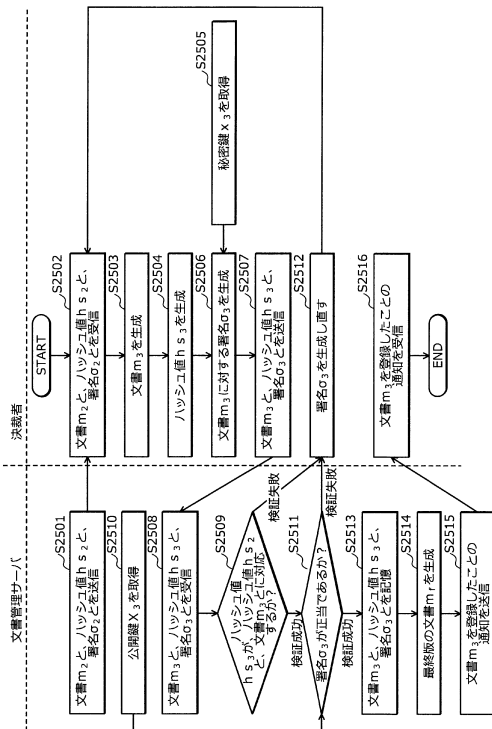
【図 2 3】



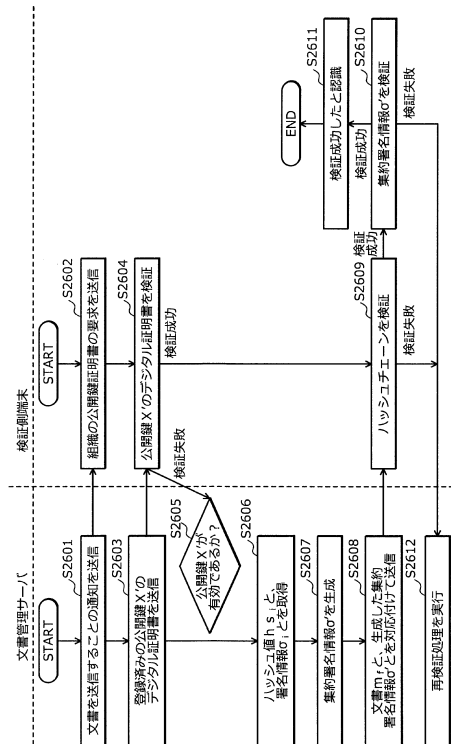
【図 2 4】



【図 2 5】



【図 2 6】



10

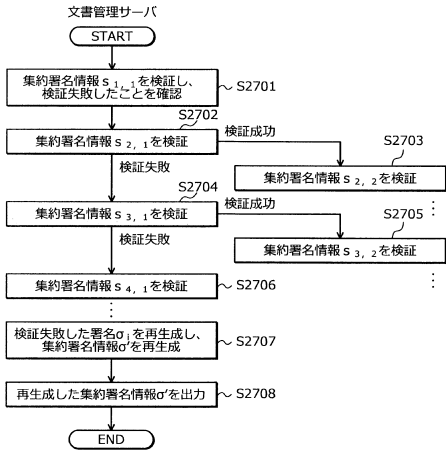
20

30

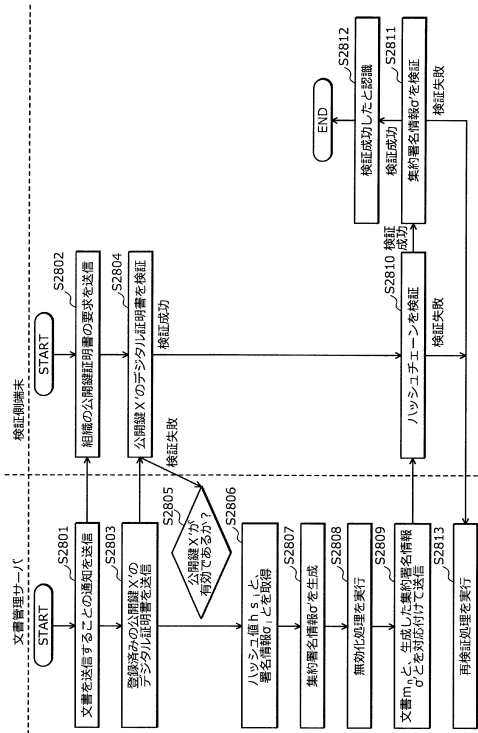
40

50

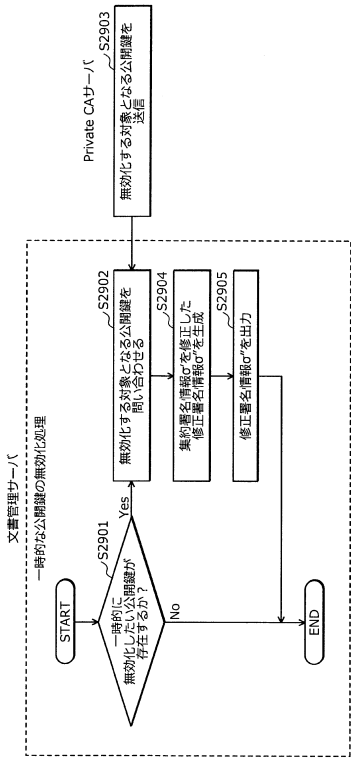
【図 27】



【図 28】



【図 29】



10

20

30

40

50

## フロントページの続き

(72)発明者 山口 純平

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 清水 俊也

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

審査官 中里 裕正

(56)参考文献 特表2005-522968(JP, A)

国際公開第2008/075420(WO, A1)

米国特許出願公開第2020/0076586(US, A1)

AMBROSIN, M. et al., SANA: Secure and Scalable Aggregate Network Attestation, Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 2016年10月, pp.731-742

MAXWELL, G. et al., Simple Schnorr Multi-Signatures with Applications to Bitcoin, Cryptology ePrint Archive, Report 2018/068 ver:20180520:191909, [online], 2018年05月20日, pp.1-34, &lt;URL:https://eprint.iacr.org/2018/068/20180520:191909&gt;

ZHAO, Y., Aggregation of Gamma-Signatures and Applications to Bitcoin, Cryptology ePrint Archive, Report 2018/414 ver:20181205:162615, [online], 2018年12月05日, pp.1-22, &lt;URL:https://eprint.iacr.org/2018/414/20181205:162615&gt;

(58)調査した分野 (Int.Cl., DB名)

H04L 9/32

JSTPlus/JMEDPlus/JST7580(JDreamII)

IEEE Explore