(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6595822号
(P6595822)
(45)発行日　令和1年10月23日(2019.10.23)　(24)登録日　令和1年10月4日(2019.10.4)

(51)Int.Cl.　F I
G06F 21/57 (2013.01)　G06F 21/57 320
G06F 21/12 (2013.01)　G06F 21/12 310

請求項の数 7　(全 16 頁)

(21)出願番号　特願2015-136367(P2015-136367)
(22)出願日　平成27年7月7日(2015.7.7)
(65)公開番号　特開2017-21434(P2017-21434A)
(43)公開日　平成29年1月26日(2017.1.26)
審査請求日　平成30年7月3日(2018.7.3)

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

最終頁に続く

(54)【発明の名称】情報処理装置及びその制御方法

(57)【特許請求の範囲】
【請求項１】
セキュリティチップを具備する情報処理装置であって、
単調に増加するカウンタ値を保持するカウンタ部と、
前記カウンタ部が保持するカウンタ値で前記情報処理装置内のソフトウェアの現在のバージョン番号を管理するバージョン管理部と、
前記ソフトウェアのアップデート用ソフトウェア、及び前記アップデート用ソフトウェアのバージョン番号の正当性を検証する第１の検証部と、
前記アップデート用ソフトウェアのバージョン番号と前記カウンタ部が保持する前記ソフトウェアの現在のバージョン番号を比較することで、前記アップデート用ソフトウェアのバージョンが現在のソフトウェアのバージョンより新しいバージョンか否かを検知するロールバック検知部と、
前記ロールバック検知部で前記アップデート用ソフトウェアのバージョンの方が現在のソフトウェアのバージョンより新しいと判断された場合に前記アップデート用ソフトウェアを用いて前記ソフトウェアを更新し、前記ロールバック検知部で前記アップデート用ソフトウェアのバージョンの方が現在のソフトウェアのバージョンより新しいと判断されなかった場合に前記ソフトウェアの更新を中断するアップデート部と、
前記アップデート部が前記ソフトウェアの更新に成功したか否かを検証する第２の検証部と、
アクセス制御可能な不揮発性メモリである保存部と、を有し、

前記バージョン管理部は、前記第２の検証部が前記ソフトウェアの更新に成功したと判断した場合に、前記カウンタ部が保持するバージョン番号を、前記アップデート用ソフトウェアのバージョン番号と一致するまで増加させ、

前記カウンタ部は、前記バージョン管理部からバージョン番号の増加の要求があった場合、パスワードである認可シークレットを要求し、前記認可シークレットが正しい場合にのみ前記カウンタ部が保持するバージョン番号の増加を実行し、

前記認可シークレットは、前記情報処理装置で起動するソフトウェアに改竄がない場合にのみアクセスできるようにアクセス制御された前記保存部に保存されていることを特徴とする情報処理装置。

【請求項２】

前記認可シークレットは、

前記情報処理装置のOSが起動していない場合にのみアクセスできるようにアクセス制御された前記保存部に保存されていることを特徴とする請求項１に記載の情報処理装置。

【請求項３】

前記第１の検証部は、当該ソフトウェアのアップデート用ソフトウェア、及び前記アップデート用ソフトウェアのバージョン番号の正当性をデジタル署名で検証するために公開鍵証明書であるルート証明書を用いることを特徴とし、

前記ルート証明書は、前記情報処理装置で起動するソフトウェアに改竄がない場合にのみアクセスできるようにアクセス制御された前記保存部に保存されている

ことを特徴とする請求項１又は２のいずれか１項に記載の情報処理装置。

【請求項４】

前記認可シークレットは、前記情報処理装置で起動するソフトウェアに改竄がない場合にのみ復号できるように前記セキュリティチップを用い暗号化されていることを特徴とする請求項１又は２に記載の情報処理装置。

【請求項５】

アクセス制御可能な不揮発性メモリである保存部を有し、セキュリティチップを用いる情報処理装置の制御方法であって、

単調に増加するカウンタ値を保持するカウンタ工程と、

前記カウンタ工程で保持するカウンタ値でソフトウェアの現在のバージョン番号を管理するバージョン管理工程と、

前記ソフトウェアのアップデート用ソフトウェア、及び前記アップデート用ソフトウェアのバージョン番号の正当性を検証する第１の検証工程と、

前記アップデート用ソフトウェアのバージョン番号と前記カウンタ工程で保持する前記ソフトウェアの現在のバージョン番号を比較することで、前記アップデート用ソフトウェアのバージョンが現在のソフトウェアのバージョンより新しいバージョンか否かを検知するロールバック検知工程と、

前記ロールバック検知工程で前記アップデート用ソフトウェアのバージョンの方が現在のソフトウェアのバージョンより新しいと判断された場合に前記アップデート用ソフトウェアを用いて前記ソフトウェアを更新し、前記ロールバック検知工程で前記アップデート用ソフトウェアのバージョンの方が現在のソフトウェアのバージョンより新しいと判断されなかった場合に前記ソフトウェアの更新を中断するアップデート工程と、

前記アップデート工程で前記ソフトウェアの更新に成功したか否かを検証する第２の検証工程と、を有し、

前記バージョン管理工程は、前記第２の検証工程が前記ソフトウェアの更新に成功したと判断した場合に、前記カウンタ工程で保持するバージョン番号を、前記アップデート用ソフトウェアのバージョン番号と一致するまで増加させ、

前記カウンタ工程は、前記バージョン管理工程からバージョン番号の増加の要求があった場合、パスワードである認可シークレットを要求し、前記認可シークレットが正しい場合にのみ前記カウンタ工程が保持するバージョン番号の増加を実行し、

前記認可シークレットは、前記情報処理装置で起動するソフトウェアに改竄がない場合

にのみアクセスできるようにアクセス制御された前記保存部に保存されていることを特徴とする情報処理装置の制御方法。
【請求項６】
コンピュータに読み込ませ実行させることで、前記コンピュータに、請求項５に記載の方法の各工程を実行させるためのプログラム。
【請求項７】
請求項６に記載のプログラムを格納したことを特徴とするコンピュータが読み取り可能な記憶媒体。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は情報処理装置及びその方法に関し、特に情報処理装置の基本ソフトウェアであるファームウェアを古いバージョンに戻すこと（以下、ロールバック）の防止に用いて好適なものである。
【背景技術】
【０００２】
パーソナルコンピュータ（PC）等の情報処理装置において、耐タンパーなセキュリティチップであるTPM（Trusted Platform Module）を用いて、情報処理装置のファームウェアのロールバックを防止する従来技術がある（特許文献１）。この特許文献１は、TPMチップが具備するNVRAM（不揮発性メモリ）に保存した現在のファームウェアのバージョン番号と、更新（アップデート）する配信ファームウェアのバージョン番号とを比較することで、ロールバックを検知・防止している。例えば、配信ファームウェアのバージョン番号が「5」で、TPM内のNVRAMに保存した現在のファームウェアのバージョン番号が「10」の場合、ロールバックとして検知し、ファームウェアのアップデートを中止する。また、別の従来技術（非特許文献１）では、TPMが具備する単調にしか増加しないカウンタ（単調増加カウンタ）で現在のファームウェアのバージョン番号を管理し、配信ファームウェアのバージョン番号と比較することで、ロールバックを検知・防止している。ロールバックを防止することで、古いファームウェアの脆弱性を突いた攻撃から情報処理装置を保護することができる。
【先行技術文献】
【特許文献】
【０００３】
【特許文献１】米国特許第８７４５６１２号公報
【非特許文献】
【０００４】
【非特許文献１】TCG Mobile Trusted Module Specification(Version 1.0, Revision 7.02)
【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、特許文献１には、TPMのNVRAMに現在のファームウェアのバージョン番号を保存しているため、NVRAMにアクセスし、古いバージョン番号でNVRAM内のバージョン番号を書き換えれば、ロールバックが可能となる。例えば、NVRAM内のバージョン番号を「10」から「2」に書き換えることで、現在のバージョン番号である「10」よりも古い、「3」から「9」のバージョンのファームウェアに戻すことができる。一方、非特許文献１には、バージョン番号を単調増加カウンタで管理しているため、バージョン番号を古い値に書き換えることは物理的に不可能である。しかし、非特許文献１には、単調増加カウンタを増加させるタイミングについては言及していないため、次の課題が想定される。つまり、ファームウェアのアップデートに失敗し、単調増加カウンタのバージョン番号を元には戻そうとしても２度と元には戻せないため、単調増加カウンタの増加タイミングが適切で

なければ、今後のアップデートができなくなる課題がある。例えば、バージョン番号が「2」のファームウェアから、バージョン番号が「11」のファームウェアにアップデートする場合を考える。このとき、アップデートにより、単調増加カウンタが管理するバージョン番号が「2」から「11」に増加するが、増加した後でファームウェアの書き換えなどに失敗すると、単調増加カウンタだけが「11」に更新され、実際のファームウェアは更新されない。従って、もう一度バージョン番号が「11」のファームウェアへのアップデートをやり直そうとしても、単調増加カウンタで管理するバージョン番号よりも値が大きくないためアップデートができなくなってしまう可能性がある。

【課題を解決するための手段】

【０００６】

この課題を解決するため、例えば本発明の情報処理装置は以下の構成を備える。すなわち、

セキュリティチップを具備する情報処理装置であって、

単調に増加するカウンタ値を保持するカウンタ部と、

前記カウンタ部が保持するカウンタ値で前記情報処理装置内のソフトウェアの現在のバージョン番号を管理するバージョン管理部と、

前記ソフトウェアのアップデート用ソフトウェア、及び前記アップデート用ソフトウェアのバージョン番号の正当性を検証する第１の検証部と、

前記アップデート用ソフトウェアのバージョン番号と前記カウンタ部が保持する前記ソフトウェアの現在のバージョン番号を比較することで、前記アップデート用ソフトウェアのバージョンが現在のソフトウェアのバージョンより新しいバージョンか否かを検知するロールバック検知部と、

前記ロールバック検知部で前記アップデート用ソフトウェアのバージョンの方が現在のソフトウェアのバージョンより新しいと判断された場合に前記アップデート用ソフトウェアを用いて前記ソフトウェアを更新し、前記ロールバック検知部で前記アップデート用ソフトウェアのバージョンの方が現在のソフトウェアのバージョンより新しいと判断されなかった場合に前記ソフトウェアの更新を中断するアップデート部と、

前記アップデート部が前記ソフトウェアの更新に成功したか否かを検証する第２の検証部と、

アクセス制御可能な不揮発性メモリである保存部と、を有し、

前記バージョン管理部は、前記第２の検証部が前記ソフトウェアの更新に成功したと判断した場合に、前記カウンタ部が保持するバージョン番号を、前記アップデート用ソフトウェアのバージョン番号と一致するまで増加させ、

前記カウンタ部は、前記バージョン管理部からバージョン番号の増加の要求があった場合、パスワードである認可シークレットを要求し、前記認可シークレットが正しい場合にのみ前記カウンタ部が保持するバージョン番号の増加を実行し、

前記認可シークレットは、前記情報処理装置で起動するソフトウェアに改竄がない場合にのみアクセスできるようにアクセス制御された前記保存部に保存されていることを特徴とする。

【発明の効果】

【０００７】

本発明によれば、情報処理装置のファームウェアアップデートにおいて、ファームウェアのロールバックを防止することができる。

【図面の簡単な説明】

【０００８】

【図１】情報処理装置の構成例を示すブロック図。

【図２】情報処理装置の機能構成例を説明するブロック図。

【図３】第１の実施形態のファームウェアアップデートにおけるロールバック検知処理を示すフローチャート。

【図４】変形例１の単調増加カウンタの不正増加防止処理を説明するフローチャート。

【図５】変形例１及び２の認可シークレットのアクセス制御を説明する図。
【図６】変形例３のルート証明書のアクセス制御を説明する図。
【図７】配信ファームウェア情報を説明する図。
【発明を実施するための形態】
【０００９】
以下、添付図面に従って本発明にかかる実施形態を詳細に説明する。
【００１０】
［実施形態］
［装置構成］
図１は、本実施形態が適応可能な情報処理装置100のブロック構成図である。情報処理装置100は、例えば一般に普及しているパーソナルコンピュータ・情報携帯端末、或いは画像データのコピー、スキャン、プリント等を実行可能な画像処理装置、或いはデジタル写真を撮影可能な撮像装置である。なお、例えば撮像装置の場合、当然、撮像装置に固有のハードウェア構成要素である操作部、撮像部等を持つことになる。実施形態でが、説明を単純化するため装置種類に固有のハードウェア構成は省略し、図１ではファームウェアのアップデートに関連する構成のみを示した。
【００１１】
図１に示すように、本実施形態における情報処理装置100は、ROM101、HDD102、TPM(Trusted Platform Module)103、RAM104、及びCPU105から構成される。
【００１２】
ROM101は、物理的、或いは論理的な書き換えが不可能な不揮発性メモリであり、BIOS110や各種ソフトウェア、及びデータを記憶可能な記憶装置である。BIOS110は情報処理装置100全体を制御するソフトウェアである。また、BIOS110は、情報処理装置100に電源が投入された際、情報処理装置内部で最初に起動されるソフトウェアである。
【００１３】
HDD102は、ブートローダ111、OS(Operation System)112、ソフトウェアA 113、ソフトウェアB 114、アップデータ（アップデート用ソフトウェア）123をはじめ様々なファイルが記憶（格納）可能な記憶装置である。ここで、ブートローダ111はOS112やアップデータ123の起動を制御をするソフトウェアである。OS112は、各種ソフトウェア（後述のソフトウェアA113、ソフトウェアB114）のロード、RAM104のメモリ管理、及び不図示の画面出力などの入出力機能を制御するソフトウェアである。ソフトウェアA 113、及びソフトウェアB 114 は、メール、ワードプロセッサ、表計算、データベース管理、ネットワークブラウジング、映像・音声再生、印刷、通信など、情報処理装置100が実現する各種機能を提供するソフトウェアである。本実施形態ではHDD102内の各種機能を提供するソフトウェアをソフトウェアA 113、及びソフトウェアB 114 から構成されているものとして説明するが、これに限定されることなく、より多くのソフトウェアから構成されていても良い。アップデータ123は、情報処理装置100を制御する基本ソフトウェアであるファームウェア（例えば、OS112など）を、アップデート用ファームウェア（以下、配信ファームウェア）で書き換える機能を持つソフトウェアである。配信ファームウェアは、例えばサーバやSDカードなどの記憶媒体経由で情報処理装置内にダウンロードできる。
【００１４】
TPM103は、耐タンパー性を有するセキュリティチップである。耐タンパー性とは、外部からの解析を困難にすると共に、外部から解析しようとした場合に内部に記憶されているソフトウェア、或いはデータを破壊することにより自己防衛する特性である。また、TPM103は、NVRAM115、PCR0 118、PCR1 119、PCR2 120、PCR3 121、PCR4 122、単調増加カウンタ117、及び制御部116から構成されている。単調増加カウンタ117は、カウンタ値を保持し、且つ、その値を減少が不可能なカウンタであり、例えば、単調に増加するハードウェアカウンタとして実現できる。NVRAM115は後述するアクセス制御が可能な不揮発性メモリである。
【００１５】

PCRは、Platform Configration Registerと略称で、揮発性メモリである。実施形態のTPM103は、PCR0,PCR1,PCR2,PCR3,PCR4の５つPCRを有するものとしているが、この数に制限はない。また、PCR0,PCR1,PCR2,PCR3,PCR4は、前述したBIOS110、ブートローダ111、OS112、ソフトウェアA 113、ソフトウェアB 114、アップデータ123などのハッシュ値を記憶する。制御部116はPCR0、PCR1、PCR2、PCR3、PCR4へのハッシュ値保存処理、暗号・復号処理、カウンタ値増加処理などを実行する。

【００１６】

ここで、制御部116によるPCRx(ｘは0,1,2,3,4のいずれか）へのハッシュ値保存処理を説明する。ハッシュ値保存処理では、所定のPCRに既に保存されているハッシュ値Hash1と、TPM103の外部から入力されたソフトウェアまたはデータのハッシュ値Hash2を用いて次の式(1)を計算し、計算した値Result1を当該PCRに保存する。

Result1＝H(Hash1｜Hash2)　　…(1)

ここで、H(x)は値xに対するハッシュ関数である。ハッシュ関数としては公知のSHA1、SHA256、SHA512等のアルゴリズムが適応可能である。「x｜y」は値xと値yの連結を示している。以上説明したPCRへのハッシュ値保存処理は、情報処理装置100が起動する際などに実行される。

【００１７】

ここで、まず情報処理装置100の通常の起動処理について説明する。情報処理装置100に電源が投入されると、まずBIOS110が実行される。その後、ブートローダ111、OS112、ソフトウェアA113、及びソフトウェアB114の順にロード・実行する。ここで、ソフトウェアA113、及びソフトウェアB114は選択的にロード・実行することも可能である。即ち、ロード・実行されないソフトウェアもある。また、ソフトウェアA113、及びソフトウェアB114のロード・実行の順序は特に定める必要はなく、必要な時に必要なソフトウェアがロード・実行される。また、ソフトウェアのロード・実行に関わらず、任意の値のハッシュ値をPCRに保存することもできる。

【００１８】

次に、情報処理装置100のファームウェアアップデート時の起動処理について説明する。ファームウェアアップデート時には、OSなどの基本ソフトウェアも更新するため、OSが起動していない状態でアップデータを起動し、ファームウェアをアップデートをする必要がある。従って、BIOS110の起動後、ブートローダ111を起動し、OS112を起動せずにアップデータ123を起動し、ファームウェアをアップデートする。

【００１９】

本実施形態では、前述したPCRへのハッシュ値保存処理を、以上説明した起動処理中に実行する。即ち、CPU105は、BIOS110に従って起動中に、BIOS110自身のハッシュ値を算出し、TPM103を介して、算出したハッシュ値を式1に従ってPCR0へ保存する。そして、CPU105は、BIOS110に従って、次にブートローダ111のハッシュ値を算出し、算出したハッシュ値を式1に従ってPCR1へ保存する。その後、CPU105は、ブートローダ111を起動する。CPU105は、ブートローダ111を起動すると、通常起動時には、OS112のハッシュ値を算出し、算出したハッシュ値を式1に従ってPCR3へ保存する。その後、CPU105はOS112を起動する。OS112を起動すると、CPU105は、ソフトウェア（ソフトウェアA113、ソフトウェアB114）を起動する前にソフトウェアのハッシュ値を算出し、算出したハッシュ値を式1に従ってPCR4へ保存し、ソフトウェアを起動する。CPU105は、OS112に従ってソフトウェアを起動する毎にハッシュ値保存処理を繰り返し実行する。一方で、ファームウェアアップデート時には、CPU105は、ブートローダ111に従って、OS112ではなく、アップデータ123のハッシュ値を算出し、算出したハッシュ値を式1に従ってPCR2に保存する。その後、CPU105は、アップデータ123を起動する。アップデータを起動すると、CPU105は、そのアップデータに従い、ダウンロードした配信ファームウェアで情報処理装置100の現在のファームウェアを更新する。

【００２０】

［機能構成］

図２は、実施形態における情報処理装置100の機能ブロック図である。この機能構成は、CPU105が実施形態の情報処理プログラム（アップデータ１２３）を実行することで実現される。本機能構成により、情報処理装置100のファームウェアが古いバージョンのファームウェアに書き換えられること（ロールバック）を防止できる。

【００２１】

第1の検証部201は、サーバやSDカード経由でダウンロードした配信ファームウェア及び当該配信ファームウェアのバージョン番号が正規のものである（改竄されていない）か否かを検証し、正規でなければファームウェアのアップデートを中止する。例えば、図７に示すように配信ファームウェア情報700として、配信ファームウェアのバージョン番号701、配信ファームウェアのハッシュ値702及びデジタル署名703で構成することで、配信ファームウェア及びバージョン番号が正規か否かを検証できる。具体的には配信ファームウェア情報のデジタル署名703が正しいか否かを署名検証鍵（公開鍵）209を用いて検証する。署名検証鍵（公開鍵）209は、デジタル署名703の生成に用いた署名生成鍵（秘密鍵）と対になる公開鍵である。配信ファームウェア情報700のデジタル署名703が正しいことが検証されれば、それに含まれる配信ファームウェアのバージョン番号701及び配信ファームウェアのハッシュ値702が正しいこと（正当性）が保証される。そして、配信ファームウェア208から算出したハッシュ値と、配信ファームウェア情報700に含まれる配信ファームウェアのハッシュ値702が一致するか否かを検証することで配信ファームウェア208が改竄されているか否かを検証できる。これはあくまで一例であり、例えば配信ファームウェアのハッシュ値ではなく、配信ファームウェアのデジタル署名を配信ファームウェアと一緒に配信し、当該デジタル署名を検証することでも配信ファームウェアの改竄を検証できる。また、署名検証鍵（公開鍵）209を公開鍵証明書とすることで、情報処理装置100内に保存するルート証明書で署名検証鍵（公開鍵）が正規のものかを検証するようにしても良い。

【００２２】

ロールバック検知部202は、ダウンロードした配信ファームウェアのバージョンが古いバージョンか否かを検知し、古いバージョンであればファームウェアのアップデートを中止する。具体的には、第1の検証部201で検証した配信ファームウェア情報700に含まれる配信ファームウェアのバージョン番号701、後述するTPM103が具備するカウンタ部204（単調増加カウンタ117に対応）で管理する現在のファームウェアのバージョン番号の比較が行われる。配信ファームウェアのバージョン番号701がカウンタ部204で管理する現在のファームウェアのバージョン番号より小さい場合は、配信ファームウェアが古いバージョンであると判断し、アップデートを中止する。なお、カウンタ部204で管理する現在のファームウェアのバージョン番号は、後述するバージョン管理部203を介して取得する。

【００２３】

バージョン管理部203は、後述するTPM103が具備する単調増加カウンタ117であるカウンタ部204のカウンタ値を取得したり、増加させる機能を持つ。取得したカウンタ値は、要求に応じてロールバック検知部202や後述する第2の検証部207に送信する。

【００２４】

カウンタ部204は、バージョン管理部203の指示で増加する単調増加なカウンタであり、例えば、TPM103の単調増加カウンタ117で実現できる。また、認証が成功した時だけ、カウンタ部204が増加できるように制御することもできる。例えば、カウンタ部204の増加時に、パスワード（以下、認可シークレット）を要求し、認可シークレットが正しい時だけ増加可能に制御できる。

【００２５】

保存部205は、アクセス制御可能な不揮発性メモリであり、例えば、TPM103のNVRAM115で実現できる。以下、保存部205のアクセス制御機能について説明する。

【００２６】

保存部205のアクセス制御機能を用いることで、PCRに保存されているハッシュ値が正解ハッシュ値と一致した場合にのみ、保存部205に保存するデータを読み書きできるように制御できる。例えば、BIOS110、ブートローダ111、アップデータ123が改竄されていない

場合のハッシュ値を正解ハッシュ値として、保存部205に保存されるデータへのアクセス条件に設定できる。このとき、BIOS110、ブートローダ111、アップデータ123の何れかが改竄されていると、アクセス条件に設定した正解ハッシュ値とPCRのハッシュ値とが不一致となり、NVRAM115に保存されるデータにアクセスできなくなる。これにより、改竄されたソフトウェアによる、データへの不正アクセスを防止できる。なお、上述した保存部205のアクセス制御機能を、以下ではTPMのNVRAM機能と呼称する場合がある。

【００２７】

アップデート部206は、配信ファームウェア208で、情報処理装置100のファームウェアを更新する。ここで、配信ファームウェア208は、秘匿にするために暗号化されていても良い。この場合、アップデート部206は、情報処理装置100に保存される復号鍵を使い、暗号化された配信ファームウェア208を復号する。ここで、復号鍵の秘匿化には、上述したTPMのNVRAM機能やシール機能を用いることができる。例えば、TPMのNVRAM機能で保存部205にアクセス制御を付与し、そこに復号鍵を保存することで改竄されたソフトウェアによる復号鍵への不正アクセスを防止できる。

【００２８】

次にTPM103のシール機能について説明する。TPM103のシール機能は、PCRに保存されているハッシュ値が正解ハッシュ値と一致した場合にのみ復号可能となるように暗号化する機能である。例えば、シール機能を用いることで、BIOS110、ブートローダ111、アップデータ123が改竄されていない場合のハッシュ値を正解ハッシュ値として復号条件に設定し、復号鍵を暗号化できる。このとき、BIOS110、ブートローダ111、アップデータ123の何れかが改竄されていると、復号条件に設定した正解ハッシュ値とPCRのハッシュ値とが不一致となり、暗号化した復号鍵を復号（アンシール）できなくなる。これにより、改竄されたソフトウェアによる復号鍵の不正読み取りを防止できる。

【００２９】

なお、前述した正解ハッシュ値は一例であり、例えば、BIOS110、ブートローダ111、OS112のハッシュ値を正解ハッシュ値とすることもできる。また、BIOS110、ブートローダ111、OS112、アップデータ123のハッシュ値を正解ハッシュ値としてもよい。

【００３０】

第2の検証部207は、アップデート部206によるアップデートが正常に行われたか否かを検証する。例えば、アップデート部206によりROMやHDDに書き込まれた配信ファームウェア208からハッシュ値を計算し、配信ファームウェア情報700の配信ファームウェアのハッシュ値702と比較することで、ファームウェアが正しく更新されたかを検証できる。
正しく更新されていない場合は、元のファームウェア（更新前のファームウェア）に戻してからアップデートを中止する。更新前に予め元のファームウェアをHDDなどに退避しておくなどすることで、元のファームウェアに戻すことが可能となる。これは一例であり、例えば、更新ファームウェアを元のファームウェアとは別の領域に書き込むことでも元のファームウェアに戻すことができる。

【００３１】

ハッシュ値比較により正しく更新されていることが確認できた場合、第2の検証部207は、バージョン管理部203にカウンタ部204で管理するバージョン番号を増加するように指示する。具体的には、配信ファームウェア情報700の配信ファームウェアのバージョン番号701の値と一致するまで、カウンタ部204で管理するバージョン番号を増加させる。
上述した第1の検証部201、ロールバック検知部202、バージョン管理部203、アップデート部206及び第2の検証部207は、例えばアップデータ123が具備する機能として実現できる。

【００３２】

ここで、改竄されていないアップデータ123のみ起動するように制御することもできる。例えば、BIOS110、ブートローダ111、アップデータ123が改竄されていない状態のハッシュ値を正解ハッシュ値（期待値）としてTPM103のNVRAM115に保存しておく。なお、正解ハッシュ値はNVRAM115に保存せず、デジタル署名付きでHDD102に保存するようにしてもよい。BIOS110は、ブートローダ111の起動前にTPM103のPCR1に保存したハッシュ値とNVRAM1

15に保存してあるブートローダ111の正解ハッシュ値を比較し、一致する場合にのみブートローダ111を起動する。ブートローダ111は、アップデータ123の起動前にTPM103のPCR2に保存したハッシュ値とNVRAM115に保存してあるアップデータ123の正解ハッシュ値を比較し、一致する場合にのみアップデータ123を起動する。これにより、改竄のない正常なアップデータ123のみ、起動可能となるため、改竄されたアップデータ123によるファームウェアの不正アップデートを防止できる。

【００３３】

尚、上述した正常なソフトウェアの起動機能を、以下ではTPMのセキュアブート機能と呼称する場合がある。

【００３４】

[ファームウェアアップデートにおけるロールバック検知処理]

図３のフローチャートにより本実施形態のファームウェアアップデートにおけるロールバック検知処理を説明する。

【００３５】

以下では、サーバからダウンロードした配信ファームウェアで直接ファームウェアをアップデートする処理を説明するが、これは例示である。他にも、SDカードなどの記憶媒体に保存した配信ファームウェアで、情報処理装置100のファームウェアを更新することもできる。また、以下では配信ファームウェアを暗号化しているが、秘匿にする必要がなければ暗号化しなくてもよい。

【００３６】

情報処理装置100のCPU105は、ソフトウェアA113やソフトウェアB114が具備する通信機能を使い、サーバにファームウェアのアップデートのリクエストをする。そして、暗号化された配信ファームウェア208及び配信ファームウェア情報700をサーバからダウンロードする(S301、S302)。ここで、情報処理装置100内に、署名検証鍵（公開鍵）209がない場合は、S302で署名検証鍵（公開鍵）209もダウンロードする。ダウンロード後、リブートし、BIOS、ブートローダ、アップデータを順次起動する(S303、S304)。起動したアップデータは、第1の検証部201で署名検証鍵（公開鍵）209が正しいか否かを情報処理装置100内に保存するルート証明書で検証する(S305)。検証に失敗した場合は、アップデート失敗として処理を終了する(S306)。検証に成功した場合は、第1の検証部201で配信ファームウェア情報700に含まれるデジタル署名703を署名検証鍵（公開鍵）209で検証する(S307)。検証に失敗した場合は、アップデート失敗として処理を終了する(S308)。検証に成功した場合は、ロールバック検知部202で、配信ファームウェアのバージョン番号701と、カウンタ部204で管理する現在のファームウェアのバージョン番号とを比較し、バージョンが古いか否かを検証する(S309)。

【００３７】

配信ファームウェアのバージョン番号701がカウンタ部204で管理する現在のファームウェアのバージョン番号以下の場合は、アップデート失敗として処理を終了する(S310)。配信ファームウェアのバージョン番号701の方が大きい場合は、アップデート部206で、TPMのシール機能で暗号化された配信ファームウェア208の復号鍵を復号（アンシール）する(S311)。アンシールに失敗した場合は、何れかのモジュールが改竄されていると判断し、アップデート失敗として処理を終了する(S312)。アンシールに成功した場合は、アップデート部206で、アンシールした復号鍵を用い、暗号化された配信ファームウェア208を復号する(S313)。次に、アップデート部206は、復号した配信ファームウェアから計算したハッシュ値と、配信ファームウェア情報700に含まれる配信ファームウェアのハッシュ値702が一致するか検証する(S314)。ハッシュ値が一致しない場合は、アップデート失敗として処理を終了する(S315)。ハッシュ値が一致した場合は、アップデート部206は、配信ファームウェア208で情報処理装置100内のファームウェアを更新する(S316)。第2の検証部207は、情報処理装置100のファームウェアが配信ファームウェア208で正しく更新できたか否かをハッシュ値比較で検証する(S317)。正しく更新できていない場合は、元のファームウェアに戻し、アップデート失敗として処理を終了する。正しく更新できていた場合は、第

2の検証部207は、カウンタ部204で管理する現在のファームウェアのバージョン番号を、配信ファームウェア700に含まれる配信ファームウェアのバージョン番号701と一致するまで増加させる(S318)。

【００３８】

このように、TPM103のカウンタ部204でロールバックを検知し、かつファームウェアが正しく書き換えられたことを確認した後にカウンタ部204を増加するようにすることで、ロールバック検知とアップデート失敗時の対応を両立することができる。

【００３９】

［変形例1]

以下、本発明に係る変形例1の処理を説明する。なお、変形例1において、上記実施形態と略同様の構成については、同一符号を付して、その詳細説明を省略する。

【００４０】

変形例1では、改竄されていないアップデータだけがカウンタ部204の増加ができるように制限する。カウンタ部の増加を制限しない場合、次の脅威が想定される。例えば、改竄されたアップデータや他のソフトウェアを介して、攻撃者がカウンタ部204のカウンタ値を不正に増加させる攻撃が考えられる。このとき、最新のファームウェアにアップデートしようとしても、カウンタ部204のカウンタが不正に大きい値になってしまっているため、ロールバック検知部202で古いファームウェアと判定され、アップデートできない。

【００４１】

本変形例では、上述の脅威に対抗するために、正常なアップデータのみカウンタ部204の増加ができるように、認可シークレットを使いカウンタ部204の増加を制御する。

【００４２】

［機能構成］

図２のブロック図により変形例1の情報処理装置100の機能構成例を説明する。

【００４３】

本変形例1の機能構成は図２に示すように、実施形態と実質的に同じで、保存部205が具備するTPMのNVRAM機能を利用し、認可シークレットへのアクセス制御を実現する。そして、正しい認可シークレットを利用した場合にのみ、カウンタ部204のカウンタ値を増加できるようにカウンタ部204を制御する。

【００４４】

以下、図５(A)を用い認可シークレットへのアクセス制御を説明する。認可シークレット501は、情報処理装置100が具備するTPM103のNVRAM115に、前述したTPMのNVRAM機能でアクセス制御された状態で保存される。例えば、認可シークレット501のアクセス条件502にBIOSの正解ハッシュ値503、ブートローダの正解ハッシュ値504、アップデータの正解ハッシュ値505を設定することができる。このとき、BIOS110、ブートローダ111、アップデータ123の何れかが改竄されていると、認可シークレットへのアクセスができなくなる。

【００４５】

従って、カウンタ部204の増加には正しい認可シークレット501が必要なので、上述のようにTPMのNVRAM機能で認可シークレット501にアクセス制御をすることで、改竄されたモジュールによるカウンタ部204の増加を防止できる。

【００４６】

［カウンタ部の不正増加の防止処理］

図４のフローチャートにより、本変形例1のカウンタ部204の不正な増加を防止する処理を説明する。図３で示す実施形態のフローチャートと略同様の処理には同一番号を付記し、説明を省略する。

【００４７】

情報処理装置100のアップデータ123は、ダウンロードした配信ファームウェア配信ファームウェアの検証及び復号をした後で、ファームウェアの書き換えを実施する(S301～S317)。アップデータ123の第2の検証部207は、保存部205にTPMのNVRAM機能でアクセス制御して保存した認可シークレットを取得する(S401)。モジュールの改竄等により認可シークレ

ットが取得できないかった場合は、アップデート失敗として処理を終了する(S402)。取得できた場合、当該認可シークレットを用い、カウンタ部204で管理するバージョン番号の増加を試みる。認可シークレットが正しくない場合は、カウンタ部204で管理するバージョン番号の増加に失敗し、アップデート失敗として処理を終了する(S403)。正しい場合は、配信ファームウェアのバージョン番号までカウンタ部204で管理するバージョン番号を増加させる(S318)。

【００４８】

このように、カウンタ部204の増加に認可シークレットを要求するようにし、かつ改竄されたモジュールがアクセスできないようにアクセス制御された領域に認可シークレットを保存することで、カウンタ部204の不正な増加を防止できる。

【００４９】

［変形例2]

以下、本発明にかかる変形例2の処理を説明する。なお、変形例2において、上記実施形態及び変形例1と略同様の構成については、同一符号を付して、その詳細説明を省略する。

【００５０】

本変形例2では、TPM103のNVRAM機能による認可シークレットへのアクセス条件に、OSの正解ハッシュ値としてPCRの初期値を追加する。初期値は例えば、20byte分の「0」が並ぶ値である。本変形例により、以下の脅威に対抗できるようになる。

【００５１】

例えば、通常のブートでOS112を起動し、OSプロセス上の改竄モジュールがアップデータ123からハッシュ値を計算し、PCR2に保存することで、PCR0、PCR1、PCR2の値が認可シークレット501のアクセス条件502と一致する。このとき、改竄モジュールであるにもかかわらず、認可シークレット501にアクセスできてしまう。

【００５２】

本変形例2は上述の脅威に対抗するために、図５(B)に示すように認可シークレットへのアクセス条件507としてOSの正解ハッシュ値（PCR初期値）506を追加している。OSの正解ハッシュ値としてPCRの初期値を設定することで、OSが一度でも起動すると、そのOSのハッシュ値がPCRに保存されるため、ハッシュ値が不一致となり認可シークレット501へのアクセスができなくなる。従って、本変形例2により、OSプロセス上で起動する改竄モジュールも認可シークレット501にアクセスできなくなる。具体的に説明すると、OS112が一度でも起動すると場合、PCR3にOS112のハッシュ値が保存されるため、アクセス条件507のOSの正解ハッシュ値（PCR初期値）と不一致になり、認可シークレット501にアクセスできなくなる。

【００５３】

このように、認可シークレットへのアクセス条件として、PCR初期値をOSの正解ハッシュ値に設定することで、OSプロセス上の改竄モジュールによる認可シークレットへの不正アクセスを防止できる。

【００５４】

［変形例3］

以下、本発明にかかる変形例3の処理を説明する。なお、本変形例3において、上記実施形態、変形例1及び変形例2と略同様の構成については、同一符号を付して、その詳細説明を省略する。

【００５５】

本変形例3では、図３中のS305の処理である署名検証鍵の検証に用いるルート証明書をTPM103のNVRAM機能を用い保護する。ルート証明書は、署名検証鍵のような公開鍵証明書とは異なり、大元の証明書であるため、例えばデジタル署名などを用いてルート証明書を検証することはできない。従って、ルート証明書は適切に保護する必要がある。本変形例3では、上述したTPM103のNVRAM機能により、ルート証明書にアクセス制御を付与する。なお、ルート証明書は、公開鍵証明書のため秘匿にする必要はなく、改竄防止だけを行えば

よい。従って、書き込みに対してのみアクセス制御を行えばよい。例えば、図６(A)のように、TPM103のNVRAM機能により、ルート証明書601の書き込み条件602として、BIOSの正解ハッシュ値503、ブートローダの正解ハッシュ値504、アップデータの正解ハッシュ値505を設定する。これにより、改竄されたアップデータによるルート証明書の改竄を防止できる。また、図６(B)のように、ルート証明書601の書き込み条件603として、OSの正解ハッシュ値（PCR初期値）を設定することで、OSプロセス上で起動した不正なモジュールによるルート証明書の改竄を防止できる。

【００５６】

［変形例4］

変形例2、3、4では、TPM103のNVRAM機能で認可シークレットやルート証明書を保護したが、TPMのシール機能を使っても良い。この場合、TPMのNVRAM機能で設定したアクセス条件をそのままTPMのシール機能の復号条件に設定することができる。

【００５７】

［変形例5］

上述の実施形態及び各変形例では、セキュリティチップ103をTPMとして説明したが、これは例示であり、例えばTPMのNVRAM機能及びシール機能と同等の機能を持ち、かつ単調増加カウンタ117を具備する解析困難なチップであればよい。例えば、上述の機能を持たせた画像処理を行うハードウェアチップを、セキュリティチップ103とすることもできる。また、ハードウェアチップではなく、同等の機能をもつソフトウェアをセキュリティチップ103の代わりに用いることもできる。この場合、TPMと同等の機能を持つソフトウェアを例えば、ROMに保存し改竄困難にしたり、通常のモジュールからはアクセスできないようにアクセス制御した領域に保存することで、保護することができる。

【００５８】

［変形例6］

上述の実施形態及び各変形例では、配信ファームウェアに対してロールバック検知を行うものとして説明したが、これは例示であり、例えば基本ソフトウェア以外のソフトウェアであるアプリケーションなどのロールバック検知にも適応できる。

【００５９】

［変形例7］

上述の実施形態及び各変形例では、第1の検証部201でダウンロードした配信ファームウェアの正当性検証、及び第2の検証部での配信ファームウェアが正しく更新されたか否かの検証にハッシュ値を用いたがこれは例示である。例えば、ハッシュ値と同等の役割を持つ値であればよいため、CRCを用いることもできる。

【００６０】

また、第2の検証部は、配信ファームウェアが正しく書き込まれたか否かを検証できればよく、上記はあくまで例示である。例えば別の検証方法として、ROMやHDDに書き込んだ配信ファームウェアと、ダウンロードした配信ファームウェアのバイナリデータを直接比較することでも、配信ファームウェアが正しく書き込まれたか否かを検証できる。

【００６１】

また、上述の実施形態及び変形例では、配信ファームウェアが正しく更新されたか否かを第2の検証部で検証したが、これは必須ではなく、不要であれば検証しなくても良い。この場合、第2の検証部の機能は不要となる。

【００６２】

（その他の実施例）

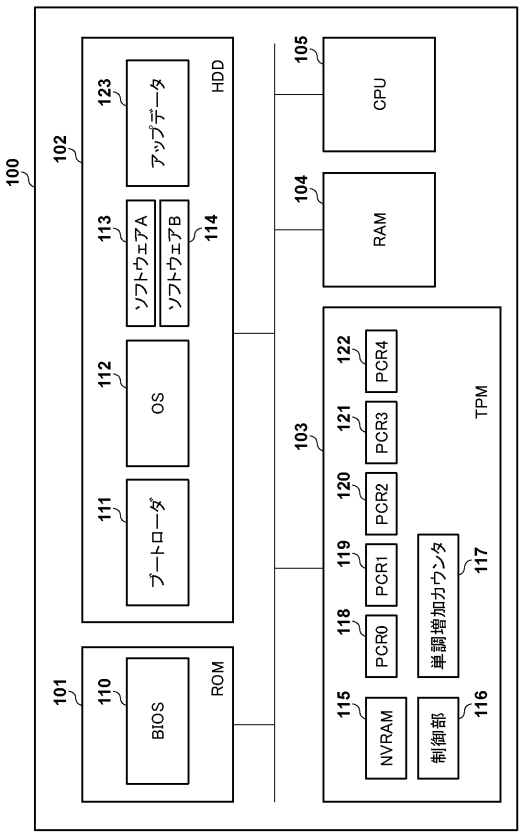
また、本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア（プログラム）を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ（またはＣＰＵやＭＰＵ等）がプログラムを読み出して実行する処理である。

【符号の説明】

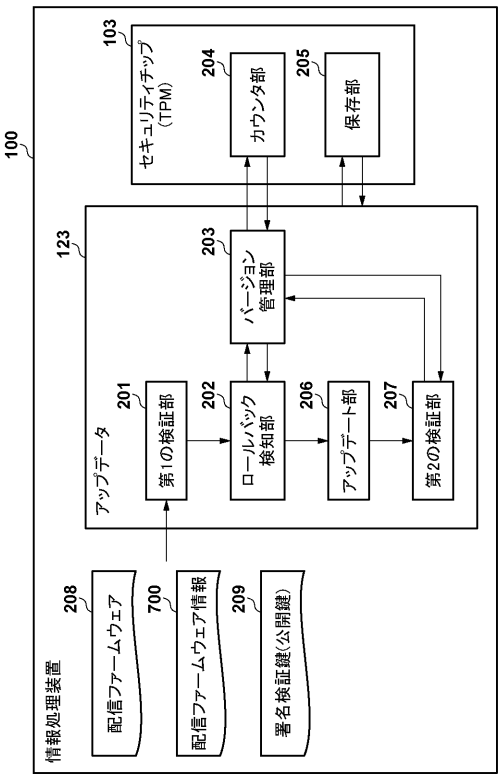
【００６３】

１００…情報処理装置、１０１…ＲＯＭ、１０２…ＨＤＤ、１０３…ＴＰＭ、１０４…ＲＡＭ、１０５…ＣＰＵ、１１５…ＮＶＲＡＭ、１１６…制御部、１１７…単調増加カウンタ、１１８～１２２…ＰＣＲ、１２３…アップデータ、２０１…第１の検証部、２０２…ロールバック検知部、２０３…バージョン解離部、２０４…カウンタ部２０４、２０５…保存部、２０６…アップデート部、２０７…第２の検証部

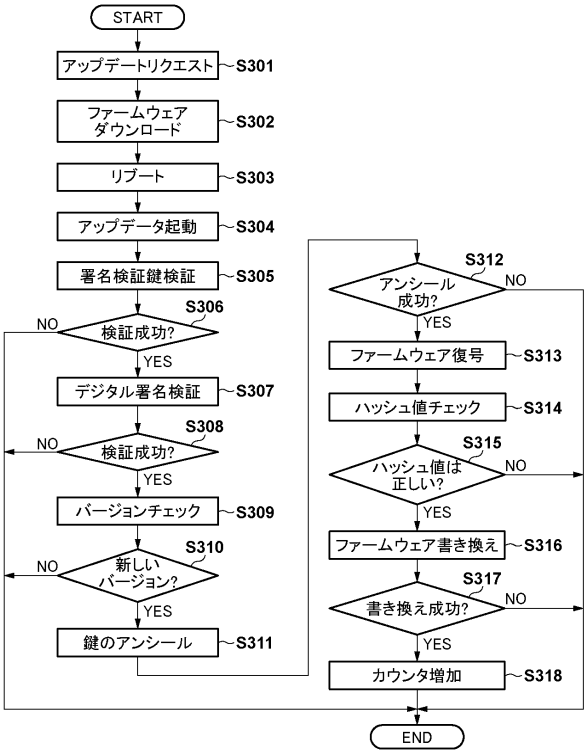
【図１】



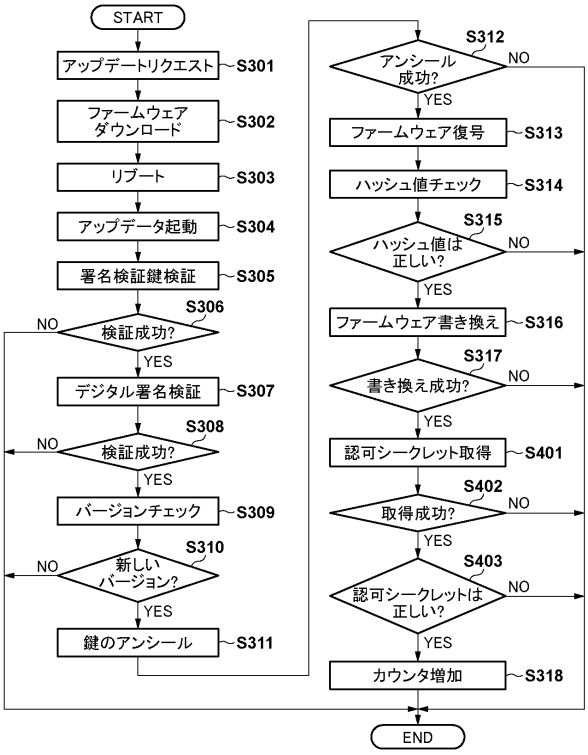
【図２】



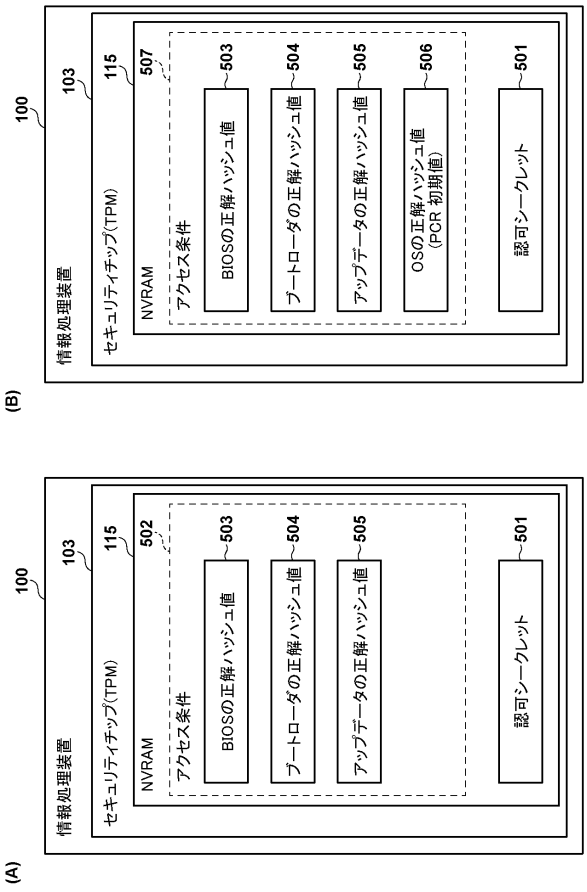
【図３】


START
アップデートリクエスト
S301
ファームウェア
ダウンロード
S302
リブート
S303
アップデータ起動
S304
署名検証鍵検証
S305
検証成功?
S306
NO
YES
デジタル署名検証
S307
検証成功?
S308
NO
YES
バージョンチェック
S309
新しい
バージョン?
S310
NO
YES
鍵のアンシール
S311
アンシール
成功?
S312
NO
YES
ファームウェア復号
S313
ハッシュ値チェック
S314
ハッシュ値は
正しい?
S315
NO
YES
ファームウェア書き換え
S316
書き換え成功?
S317
NO
YES
カウンタ増加
S318
END

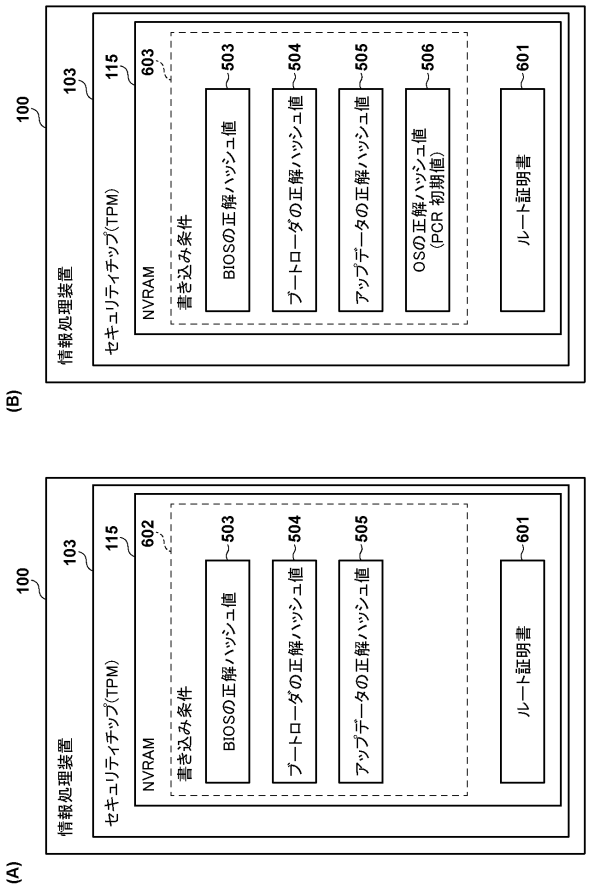
【図４】


START
アップデートリクエスト
S301
ファームウェア
ダウンロード
S302
リブート
S303
アップデータ起動
S304
署名検証鍵検証
S305
検証成功?
S306
NO
YES
デジタル署名検証
S307
検証成功?
S308
NO
YES
バージョンチェック
S309
新しい
バージョン?
S310
NO
YES
鍵のアンシール
S311
アンシール
成功?
S312
NO
YES
ファームウェア復号
S313
ハッシュ値チェック
S314
ハッシュ値は
正しい?
S315
NO
YES
ファームウェア書き換え
S316
書き換え成功?
S317
NO
YES
認可シークレット取得
S401
取得成功?
S402
NO
YES
認可シークレットは
正しい?
S403
NO
YES
カウンタ増加
S318
END

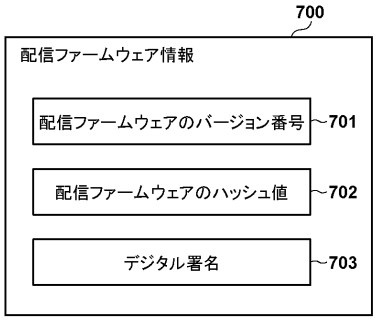
【図５】


(B)
100
情報処理装置
103
セキュリティチップ(TPM)
115
NVRAM
507
アクセス条件
503
BIOSの正解ハッシュ値
504
ブートローダの正解ハッシュ値
505
アップデータの正解ハッシュ値
506
OSの正解ハッシュ値
(PCR 初期値)
501
認可シークレット
(A)
100
情報処理装置
103
セキュリティチップ(TPM)
115
NVRAM
502
アクセス条件
503
BIOSの正解ハッシュ値
504
ブートローダの正解ハッシュ値
505
アップデータの正解ハッシュ値
501
認可シークレット

【図６】


(B)
100
情報処理装置
103
セキュリティチップ(TPM)
115
NVRAM
603
書き込み条件
503
BIOSの正解ハッシュ値
504
ブートローダの正解ハッシュ値
505
アップデータの正解ハッシュ値
506
OSの正解ハッシュ値
(PCR 初期値)
601
ルート証明書
(A)
100
情報処理装置
103
セキュリティチップ(TPM)
115
NVRAM
602
書き込み条件
503
BIOSの正解ハッシュ値
504
ブートローダの正解ハッシュ値
505
アップデータの正解ハッシュ値
601
ルート証明書

【図７】


700
配信ファームウェア情報
配信ファームウェアのバージョン番号
701
配信ファームウェアのハッシュ値
702
デジタル署名
703

フロントページの続き

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

審査官　行田　悦資

(56)参考文献　国際公開第２０１４／０７４６７４（WO，Ａ１）
国際公開第２００９／０４４５３３（WO，Ａ１）
特開２０００－２９３３６６（ＪＰ，Ａ）
特開２００９－２９４８５９（ＪＰ，Ａ）
特表２００９－５３４７６５（ＪＰ，Ａ）
宗藤誠治ほか，Ｌｉｎｕｘのセキュリティ機能，情報処理，日本，一般社団法人情報処理学会，２０１０年１０月１５日，第51巻，第10号，p.1284-1293
中村智久ほか，ＰＣ搭載セキュリティチップ（ＴＰＭ）の概要と最新動向，情報処理　第４７巻第５号，日本，社団法人情報処理学会，２００６年　５月１５日，第47巻，第5号，p.473-478

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　２１／５７
Ｇ０６Ｆ　２１／１２