(19)日本国特許庁(JP)

再公表特許(A1)

(11)国際公開番号
WO2009/157142

発行日　平成23年12月8日(2011.12.8)

(43)国際公開日　平成21年12月30日(2009.12.30)

(51)Int.Cl.			F I			テーマコード（参考）
H04L	9/08	(2006.01)	H04L	9/00	601B	5J104
H04L	9/14	(2006.01)	H04L	9/00	601E	
H04L	9/10	(2006.01)	H04L	9/00	641	
			H04L	9/00	621A	

審査請求 未請求　予備審査請求 未請求　（全 100 頁）

出願番号	特願2010-517691 (P2010-517691)
(21)国際出願番号	PCT/JP2009/002531
(22)国際出願日	平成21年6月4日(2009.6.4)
(31)優先権主張番号	特願2008-163071 (P2008-163071)
(32)優先日	平成20年6月23日(2008.6.23)
(33)優先権主張国	日本国(JP)

(71)出願人　000005821
パナソニック株式会社
大阪府門真市大字門真１００６番地
(74)代理人　100090446
弁理士　中島　司朗
(74)代理人　100125597
弁理士　小林　国人
(74)代理人　100146798
弁理士　川畑　孝二
(74)代理人　100121027
弁理士　木村　公一
(72)発明者　芳賀　智之
大阪府門真市大字門真１００６番地　パナソニック株式会社内

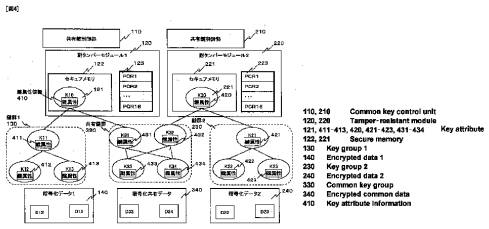
最終頁に続く

(54)【発明の名称】情報処理装置、暗号鍵の管理方法、コンピュータプログラム及び集積回路

(57)【要約】

複数のステークホルダーの各々にルート鍵から構成される鍵ツリー群の鍵を、複数のステークホルダー間で共有鍵を生成し、生成した共有鍵に対するアクセス制限を柔軟に設定する。

複数のステークホルダー毎に共有鍵制御部と、耐タンパーモジュールを備え、ステークホルダーの依存関係に基いて共有鍵の設定を行い、共有鍵設定後、セキュリティレベルを維持するために不正なステークホルダーからはアクセスできないように共有鍵へのアクセス制御を行う。



【特許請求の範囲】
【請求項１】
第１のステークホルダーに対応する第１共有鍵制御部と、
第２のステークホルダーに対応する第２共有鍵制御部と、
前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、
前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデータ保持部と、
前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備し、
前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、
前記第１共有鍵制御部は、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断し、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーし、
前記第２共有鍵制御部は、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することを特徴とする情報処理装置。
【請求項２】
前記第２共有鍵制御部は、前記第１共有鍵制御部との間の依存関係を証明した証明書を有し、前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を送付する際、前記証明書を送付し、
前記第１共有鍵制御部は、前記証明書に基づいて、前記第２共有鍵制御部に対応する第２のステークホルダーが、少なくとも前記第１のステークホルダーに対応する第１の耐タンパーモジュールを利用するステークホルダーモデルであると判断した場合に、
前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であると判断することを特徴とする請求項１記載の情報処理装置。
【請求項３】
前記第１の暗号鍵群に含まれる各鍵は、当該鍵が前記第１の暗号鍵群からコピー可能か否かを示す属性情報を有し、
前記第１共有鍵制御部は、前記属性情報を参照して、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探すことを特徴とする請求項２記載の情報処理装置。
【請求項４】
前記第１共有鍵制御部は、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵が存在しない場合、コピー可能な鍵を生成して、この生成した鍵を前記第２の暗号鍵群の中にコピーすることを特徴とする請求項１記載の情報処理装置。
【請求項５】
前記第１共有鍵制御部は、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵の位置を示す位置情報を前記所定の鍵のリンク情報として生成し、このリンク情報と共に前記所定の鍵を前記第２の暗号鍵群の中にコピーすることを特徴とする請求項３記載の情報処理装置。
【請求項６】
前記第１共有鍵制御部は、前記第１の暗号鍵群に含まれる所定の鍵の位置情報及び前記第２の暗号鍵群に含まれる所定の鍵の位置情報を、前記第１の暗号鍵群に含まれる前記所

定の鍵より下層の鍵のリンク情報として生成することを特徴とする請求項５記載の情報処理装置。

【請求項７】

前記第１共有鍵制御部と前第２共有鍵制御部とは、共用の共有鍵制御部であることを特徴とする請求項１記載の情報処理装置。

【請求項８】

前記第１のステークホルダーが管理する第１ステークホルダー環境は、前記第２のステークホルダーが管理する第２のステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１共有鍵制御部は、前記所定の鍵と置換える代替鍵を前記第１の耐タンパーモジュールに生成させ、前記所定の鍵を親鍵とするツリー構造に含まれる鍵を前記代替鍵で再暗号化させると共に前記前記所定の鍵の親鍵を用いて前記代替鍵を暗号化させて、前記第２共有鍵制御部による前記所定のデータの利用を排除することを特徴とする請求項１記載の情報処理装置。

【請求項９】

前記第１のステークホルダーが管理する第１ステークホルダー環境は、前記第２のステークホルダーが管理する第２のステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵を用いて前記第１の耐タンパーモジュールに前記所定のデータを暗号化し直させて、前記第２共有鍵制御部による前記所定の鍵の使用を排除することを特徴とする請求項１記載の情報処理装置。

【請求項１０】

前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵は、属性情報として、前記改竄のない第２のステークホルダーが管理する第２ステークホルダー環境のハッシュ値から生成された期待値として鍵利用制限情報を有し、

第２の耐タンパーモジュールは、前記第２のステークホルダー環境のハッシュ値から生成された実際の値としての環境情報を記憶し、

第２の共有鍵制御部から前記第１の共有鍵制御部に対して前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵の利用を依頼するときに、第２の共有鍵制御部は、前記鍵利用制限情報と前記環境情報とを比較し、比較結果が正しい場合にのみ前記鍵を利用させるように制限をすることを特徴する請求項１記載の情報処理装置。

【請求項１１】

前記第１ステークホルダーが管理する第１のステークホルダー環境は、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵を、第２の共有鍵制御部から利用できないように、前記鍵利用制限情報を書き換えることを特徴とする請求項１０の情報処理端末。

【請求項１２】

前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データは、属性情報として、前記改竄のない第２のステークホルダーが管理する第２ステークホルダー環境のハッシュ値から生成された期待値である暗号化データ利用制限情報を有し、

第２の耐タンパーモジュールは、前記第２のステークホルダー環境のハッシュ値から生成された実際の値としての環境情報を記憶し、

第２の共有鍵制御部から前記第１の共有鍵制御部に対して前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化されたデータの復号処理を依頼するときに、第２の共有鍵制御部は、前記暗号化データ利用制限情報と前記環境情報とを比較し、比較結果が正しい場合にのみ前記暗号化データの復号処理させるように制限をすることを特徴する請求項１記載の情報処理装置。

【請求項１３】

前記第１ステークホルダーが管理する第１のステークホルダー環境は、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データを、第２の共有鍵制御部から利用できないように、前記暗号化データ利用制限情報を書き換えることを特徴とする請求項１２の情報処理端末。

【請求項１４】

前記第１共有鍵制御部は、第１ステークホルダー環境および第２ステークホルダー環境を環境に対して完全性をチェックしてから改竄されていない環境のみを起動する機能であるセキュアブートによってブートする際に、第２ステークホルダー環境が改竄された、もしくはリボーク対象であることを検知することを特徴とする請求項８乃至請求項１３のいずれかに記載の情報処理装置。

【請求項１５】

前記第１共有鍵制御部は、外部のサーバーから、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象である旨の通知を受けることで前記第第２ステークホルダー環境が改竄された、もしくはリボーク対象であることを検知することを特徴とする請求項８乃至請求項１３のいずれかに記載の情報処理装置。

【請求項１６】

第１のステークホルダーに対応する第１共有鍵制御部と、

第２のステークホルダーに対応する第２共有鍵制御部と、

前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、

前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、

前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置における暗号鍵の鍵管理方法であって、

前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、

前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断し、

前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーし、

前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することを特徴とする暗号鍵の管理方法。

【請求項１７】

第１のステークホルダーに対応する第１共有鍵制御部と、

第２のステークホルダーに対応する第２共有鍵制御部と、

前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、

前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、

前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置における暗号鍵の鍵管理に用いるコンピュータプログラムであって、

コンピュータに対して、

前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、

前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断する処理と、
前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーする処理と、を実行させ、
前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用する処理を実行させることを特徴とするコンピュータプログラム。

【請求項１８】
第１のステークホルダーに対応する第１共有鍵制御部と、
第２のステークホルダーに対応する第２共有鍵制御部と、
前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、
前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、
前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置に用いる集積回路であって、
情報処理部と、
この情報処理部に対して、
前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、
前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断する処理と、
前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーする処理と、を実行させ、
前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用する処理を実行させる処理プログラムを格納したメモリと、
を具備した集積回路。

【請求項１９】
前記第１共有鍵制御部は、
前記第１の暗号鍵群に含まれる暗号鍵で暗号化されている前記所定の鍵を、前記暗号鍵を用いて復号し、
復号された鍵を、前記第２の暗号鍵群に含まれる暗号鍵で再暗号化し、
再暗号化された鍵を、前記第２の暗号鍵群の中にコピーする
ことを特徴とする請求項１記載の情報処理装置。

【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、複数のステークホルダーを備える装置において、ツリー構造で管理される鍵を、複数のステークホルダー間で共有して利用できる共有鍵の制御を行う情報処理装置、暗号鍵の管理方法、コンピュータプログラム及び集積回路に関するものである。

【背景技術】
【０００２】

近年、情報セキュリティへの意識の高まりと共に、データを保護する技術ニーズが高まってきている。

【０００３】

これを背景として、セキュアなコンピュータプラットフォームを開発、普及させることを目的として、Ｔｒｕｓｔｅｄ Ｃｏｍｐｕｔｉｎｇ Ｇｒｏｕｐ（ＴＣＧ）が設立された。ＴＣＧでは、Ｔｒｕｓｔｅｄ Ｐｌａｔｆｏｒｍ Ｍｏｄｕｌｅ（ＴＰＭ）と呼ばれるセキュリティコアモジュールを利用し、安全な端末環境を実現している（非特許文献１～３参照）。ＴＣＧの基本機能として、特徴的な機能が３つある。

【０００４】

まず１つ目の特徴機能として、端末起動時からＯＳやアプリケーションが起動するまでの各モジュールのインテグリティ情報を計測し、計測した値を、それより前のモジュールのインテグリティ情報を連鎖させる累積処理（ＴＣＧのＴＰＭ＿Ｅｘｔｅｎｄコマンドに相当する処理）をし、その累積値をＴＰＭ内のＰｌａｔｆｏｒｍ Ｃｏｎｆｉｇｒａｔｉｏｎ Ｒｅｇｉｓｔｅｒｓ（ＰＣＲ）と呼ばれるレジスタに格納するＴｒｕｓｔｅｄ Ｂｏｏｔ機能がある。

【０００５】

２つ目の特徴機能として、ＰＣＲに蓄積した値を端末環境情報として、外部のサーバーに通知し、外部のサーバーで、端末環境情報が期待される環境情報であるかどうかを検証するというＡｔｔｅｓｔａｔｉｏｎ機能がある。これらの機能を利用し、リモートで端末の環境の正当性を検証可能であることが、ＴＣＧ技術のメリットの１つである。

【０００６】

そして、３つ目の特徴機能として、Ｐｒｏｔｅｃｔｅｄ Ｓｔｏｒａｇｅ機能というストレージデータに対する保護機能がある。この機能は、ＴＰＭ内のセキュアなメモリ領域に、ＴＰＭ内部で生成した暗号鍵であるＳｔｏｒａｇｅ Ｒｏｏｔ Ｋｅｙ（ＳＲＫ）を保持する。そして、ＳＲＫをルート鍵とし、保護対象となる複数の暗号鍵をルート鍵で暗号化してＴＰＭ外のメモリ上で安全に保護する機能である。

【０００７】

具体的には、暗復号処理や署名処理に利用する鍵を保護するために、ＳＲＫをルート鍵とした階層的なツリー構造のノードに、それら保護対象の鍵を対応づけ、親のノード鍵が子供のノード鍵を暗号化する方法である。ルート鍵であるＳＲＫがＴＰＭ内のセキュアメモリ上で保持されているため、ルート鍵以外の暗号化された鍵は、ＴＰＭ外のメモリ上で管理しても安全となる。また、これらの鍵をバックアップする目的として、マイグレート機能も有する。マイグレート機能は、他のＴＰＭのＳＲＫの下に、鍵をコピーする機能である。

【０００８】

また、ＴＣＧは、ＴＰＭ搭載端末として、携帯電話機もターゲットとしており、携帯電話機向けのＴＰＭの仕様も規格化されている（非特許文献３、４）。携帯電話機向けのＴＰＭは、Ｍｏｂｉｌｅ Ｔｒｕｓｔｅｄ Ｍｏｄｕｌｅ（ＭＴＭ）と呼ばれている。ＭＴＭは、ＴＰＭの機能を実現しながらも、一部のコマンドを携帯電話機向けに修正したり、新規コマンドが追加されている。その追加機能として、セキュアブート機能と、マルチステークホルダーモデルを定義している。

【０００９】

セキュアブートとは、携帯電話機の端末起動時から、ＯＳやアプリケーションが起動するまでの各モジュールのインテグリティ情報を計測し、計測した値が期待される値であるかを、ローカル端末内で検証しながらブートする方式である。

【００１０】

また、マルチステークホルダーモデルとは、デバイスメーカー、キャリア、アプリケーションサービス提供者、ユーザーといった携帯電話機端末内に存在する複数のステークホルダーが所有する権利物を安全に利用するための実装モデルを定義したものである。各ステークホルダーの権利物として、例えば、デバイスメーカーであれば、Ｉｎｔｅｒｎａｔ

ｉｏｎａｌ Ｍｏｂｉｌｅ Ｅｑｕｉｐｍｅｎｔ Ｉｄｅｎｔｉｔｙ（ＩＭＥＩ）であり、キャリアであれば、Ｓｕｂｓｃｒｉｂｅｒ Ｉｄｅｎｔｉｆｉｃａｔｉｏｎ Ｍｏｄｕｌｅ（ＳＩＭ）関連情報であり、アプリケーションサービス提供者であれば、サービス提供されたデータであり、ユーザーであれば、アドレス帳が挙げられる。

【００１１】

要するに、マルチステークホルダーモデルとは、それぞれのステークホルダーが利用するＭＴＭを、個別に割り当てることでそれぞれの権利物を安全に利用するモデルである。仮想化技術を用いることで、１つの端末内に、複数のＭＴＭを仮想的に実現することが可能となる。

【００１２】

特許文献１は、暗号鍵をツリー構造に構造化し、暗号鍵更新時に管理する方法が開示されている。

【先行技術文献】

【特許文献】

【００１３】

【特許文献１】特開平１１－１８７０１３

【非特許文献】

【００１４】

【非特許文献１】ＴＰＭ Ｍａｉｎ，Ｐａｒｔ１ Ｄｅｓｉｇｎ Ｐｒｉｎｃｉｐｌｅｓ，Ｓｐｅｃｉｆｉｃａｔｉｏｎ ｖｅｒｓｉｏｎ １．２ Ｌｅｖｅｌ２ Ｒｅｖｉｓｉｏｎ１０３（９ Ｊｕｌｙ ２００７）

【非特許文献２】ＴＰＭ Ｍａｉｎ，Ｐａｒｔ２ ＴＰＭ Ｓｔｒｕｃｔｕｒｅｓ，Ｓｐｅｃｉｆｉｃａｔｉｏｎ ｖｅｒｉｏｎ１．２ Ｌｅｖｅｌ２ Ｒｅｖｉｓｉｏｎ１０３（９ Ｊｕｌｙ ２００７）

【非特許文献３】ＴＰＭ Ｍａｉｎ Ｐａｒｔ３ Ｃｏｍｍａｎｄｓ，Ｓｐｅｃｉｆｉｃａｔｉｏｎ ｖｅｒｉｏｎ １．２ Ｌｅｖｅｌ２ Ｒｅｖｉｓｉｏｎ１０３（９ Ｊｕｌｙ ２００７）

【非特許文献４】ＴＣＧ Ｍｏｂｉｌｅ Ｔｒｕｓｔｅｄ Ｍｏｄｕｌｅ Ｓｐｅｃｉｆｉｃａｔｉｏｎ ｖｅｒｓｉｏｎ １．０ Ｒｅｖｉｓｉｏｎ １（１２ Ｊｕｎｅ ２００７）

【非特許文献５】ＴＣＧ Ｍｏｂｉｌｅ Ｒｅｆｅｒｅｎｃｅ Ａｒｃｈｉｔｅｃｔｕｒｅ Ｓｐｅｃｉｆｉｃａｔｉｏｎ ｖｅｒｓｉｏｎ１．０ Ｒｅｖｉｓｉｏｎ１（１２ Ｊｕｎｅ ２００７）

【発明の概要】

【発明が解決しようとする課題】

【００１５】

上述したマルチステークホルダーモデルにおいて、各ステークホルダーが、各々ＳＲＫをルートとした鍵ツリーを保持する。そのため、一つの端末内に、その鍵のツリーが、ステークホルダーの個数だけ存在することになる。

【００１６】

ここで、ステークホルダーＡが、ステークホルダーＢが管理しているデータや機能を利用することが想定される。特に、ステークホルダーＡが、ステークホルダーＢのＴＰＭで管理されているＳＲＫをルートするツリーのノード鍵で暗号化されているデータに対してアクセスしたい場合、ステークホルダーＡは、ステークホルダーＢに対して、ステークホルダーＢのＳＲＫを用いたデータの復号処理を要求しなければならない。そして、ステークホルダーＢは、ステークホルダーＢのＳＲＫを利用して復号処理後、復号データをステークホルダーＡに送信する。

【００１７】

このように、異なるステークホルダー間で共有したいデータがあった場合、その共有データのアクセス要求のたびに、鍵を管理しているステークホルダーが復号処理をして、安

全に復号データを渡すといったオーバーヘッド処理が必要になる。

【００１８】

これらのオーバーヘッド処理を回避するために、異なるステークホルダー間でアクセスされるセキュアな共有データに対しては、異なるステークホルダー間で共有な鍵で暗号化することが必要となる。言い換えると、異なるステークホルダー間で共有な鍵を保持させるような仕組みが必要となる。

【００１９】

現状のＴＣＧ仕様で、ステークホルダーＡとＢの２者間で鍵を共有させる場合、すなわち、ステークホルダーＢの鍵を、ステークホルダーＡでも利用できるようにするには、ＴＣＧのマイグレート機能を利用する。マイグレート元をステークホルダーＢとし、マイグレート先をステークホルダーＡとし、ステークホルダーＢの鍵をステークホルダーＡへマイグレートする。その結果、ステークホルダーＢからマイグレートされた鍵をステークホルダーＡが管理する鍵ツリーのノードとして構成することが可能となる。

【００２０】

これにより、共有鍵を保持しているステークホルダーＡとＢの間では、マイグレートされた鍵が、共有鍵となり、その共有鍵で暗号化したデータに対して、ステークホルダーを跨ることなく、直接アクセス可能となる。

【００２１】

しかしながら、マイグレート機能を利用した場合、マイグレート対象となった鍵の実体は、ステークホルダーＡとＢの各々で保持しているため、１つの端末に、共有鍵が２重持ちされることになり、非効率であるという課題があった。特に、共有鍵を複数設定する場合、２重持ちされる鍵が複数存在することになる。

【００２２】

また、特許文献１は、暗号鍵をツリー構造で管理し、子供のノードが親のノードを暗号化する構成であり、リーフからルートに至る経路の鍵群をユーザー鍵としてユーザーに配布し、ユーザーが脱退した際の鍵ツリーの更新方法を開示している。

【００２３】

しかしながら、ルート鍵は全てのユーザーで共有しているため、一人のユーザーが脱退する度に、更新後のルート鍵を全ユーザーに再配布しなければならないため、管理が複雑になってしまうという課題があった。

【００２４】

そこで、本発明は、これらの課題を解決するもので、マルチステークホルダーモデルにおいて、ステークホルダー毎に異なるＳＲＫをルート鍵とする鍵ツリー間のノードの一部を共有鍵をとして設定する情報処理装置、暗号鍵の管理方法、コンピュータプログラム及び集積回路と、共有鍵を共有しているステークホルダーに不正があった場合に、不正なステークホルダーからの共有鍵へのアクセスを、より柔軟に無効化する情報処理装置、暗号鍵の管理方法、コンピュータプログラム及び集積回路を提供することを目的とする。

【課題を解決するための手段】

【００２５】

上記の課題を解決するために、本発明に係る情報処理装置は、第１のステークホルダーに対応する第１共有鍵制御部と、第２のステークホルダーに対応する第２共有鍵制御部と、前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデータ保持部と、前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備し、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第１共有鍵制御部は、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断し、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホル

ダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーし、前記第２共有鍵制御部は、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することを特徴とする。

【発明の効果】

【００２６】

本発明によると、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を前記第２の暗号鍵群の中にコピーすることにより、前記第２のステークホルダーが前記第１のステークホルダーに依存する関係であることを条件に、前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体をコピーするのではなく、前記所定の鍵のみをコピーする。これにより、前記第１の耐タンパーモジュール及び前記第２の耐タンパーモジュールとで前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体を二重持ちする非効率を回避できる。

【００２７】

また、前記第２共有鍵制御部側で、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持し、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することにより、前記所定の鍵をコピーするだけで、前記第２共有鍵制御部側では、第１共有鍵制御部に対応する第１の耐タンパーモジュールが管理する第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用できる。これにより、前記第２共有鍵制御部は、前記データ保持部で保持された暗号化された所定のデータを簡易な構成で利用できる。

【図面の簡単な説明】

【００２８】

【図１】本発明の実施の形態１におけるシステム概要を示す図である。

【図２】本発明の実施の形態１におけるマルチステークホルダーを示す図である。

【図３】本発明の実施の形態１における情報処理端末の構成を示す図である。

【図４】本発明の実施の形態１における耐タンパーモジュールが有する鍵ツリーの構成を示す図である。

【図５】本発明の実施の形態１における鍵属性情報を示す図である。

【図６】本発明の実施の形態１における鍵のリンク情報示した図である。

【図７】本発明の実施の形態１における鍵属性情報の例を示した図である。

【図８】本発明の実施の形態１における鍵管理テーブルを示した図である。

【図９】本発明の実施の形態１におけるステークホルダー証明書の構成を示した図である。

【図１０】本発明の実施の形態１におけるステークホルダー証明書を用いてトラストモデルを表現した例を示した図である。

【図１１】本発明の実施の形態１における共有鍵の設定のシーケンス図である。

【図１２】本発明の実施の形態１における共有鍵の設定のシーケンス図である。

【図１３】本発明の実施の形態１における共有鍵を利用シーケンス図である。

【図１４】本発明の実施の形態１における共有鍵の無効化概要フローチャートを示す図である。

【図１５】本発明の実施の形態１における共有鍵の無効化詳細フローチャートを示す図である。

【図１６】本発明の実施の形態１における共有鍵の無効化前後における鍵ツリーの構成を示す図である。

【図１７】本発明の実施の形態２における共有鍵の無効化詳細フローチャートを示す図で

ある。
【図１８】本発明の実施の形態２における共有鍵の無効化前後における鍵ツリーの構成を示す図である。
【図１９】本発明の実施の形態３における共有鍵の無効化詳細フローチャートを示す図である。
【図２０】本発明の実施の形態４における暗号化共有データ構造を示す図である。
【図２１】本発明の実施の形態４における暗号化共有データ構造を利用した暗号化共有データの復号処理のシーケンス図である。
【図２２】本発明の実施の形態４における共有鍵の無効化詳細フローチャートを示す図である。
【図２３】本発明の実施の形態３における共有鍵の再共有化のフローチャートを示す図である。
【発明を実施するための形態】
【００２９】
本発明の請求項１に記載の情報処理装置は、第１のステークホルダーに対応する第１共有鍵制御部と、第２のステークホルダーに対応する第２共有鍵制御部と、前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデータ保持部と、前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備し、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第１共有鍵制御部が、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断し、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーし、前記第２共有鍵制御部は、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することを特徴とする。
【００３０】
本態様により、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を前記第２の暗号鍵群の中にコピーすることにより、前記第２のステークホルダーが前記第１のステークホルダーに依存する関係であることを条件に、前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体をコピーするのではなく、前記所定の鍵のみをコピーする。これにより、前記第１の耐タンパーモジュール及び前記第２の耐タンパーモジュールとで前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体を二重持ちする非効率を回避できる。
【００３１】
また、前記第２共有鍵制御部側で、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持し、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することにより、前記所定の鍵をコピーするだけで、前記第２共有鍵制御部側では、第１共有鍵制御部に対応する第１の耐タンパーモジュールが管理する第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用できる。これにより、前記第２共有鍵制御部は、前記データ保持部で保持された暗号化された所定のデータを簡易な構成で利用できる。
【００３２】

さらに、前記第２共有鍵制御部は、前記第２のステークホルターが第１のステークホルダーに依存する関係にある場合にのみ、前記データ保持部内に保持された暗号化された所定のデータを利用できる。これにより、前記所定のデータを管理する鍵構成を前記第１の耐タンパーモジュール及び前記第２の耐タンパーモジュールで簡易にしつつ、前記所定のデータの機密性を保証できる。

【００３３】

本発明の請求項２に記載の情報処理装置は、前記第２共有鍵制御部が、前記第１共有鍵制御部との間の依存関係を証明した証明書を有し、前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を送付する際、前記証明書を送付し、前記第１共有鍵制御部が、前記証明書に基づいて、前記第２共有鍵制御部に対応する第２のステークホルダーが、少なくとも前記第１のステークホルダーに対応する第１の耐タンパーモジュールを利用するステークホルダーモデルであると判断した場合に、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であると判断するものである。

【００３４】

本態様によると、前記第１共有鍵制御部は、前記第２共有鍵制御部に対応する第２のステークホルダーが、少なくとも前記第１のステークホルダーに対応する第１の耐タンパーモジュールを利用するステークホルダーモデルであると証明書に基づいて判断した場合に、前記第２のステークホルダーが前記第１のステークホルダーに依存する関係であると判断する。これにより、前記第２のステークホルダーの前記第１のステークホルダーに対する依存関係を確実に判断できるので、前記所定のデータを管理する鍵構成を前記第１の耐タンパーモジュール及び前記第１の耐タンパーモジュールで簡易にしつつ、不正なステークホルダーからの前記所定のデータへのアクセスを確実に禁止できる。

【００３５】

本発明の請求項３に記載の情報処理装置は、前記第１の暗号鍵群に含まれる各鍵が、当該鍵が前記第１の暗号鍵群からコピー可能か否かを示す属性情報を有し、前記第１共有鍵制御部は、前記属性情報を参照して、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探すことを特徴としている。

【００３６】

本態様によると、当該鍵が前記第１の暗号鍵群からコピー可能か否かを示す属性情報を参照して、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探す。これにより、前記属性情報を参照するだけでコピー可能な鍵を探せるので、コピー可能な鍵を簡易にサーチできる。

【００３７】

本発明の請求項４に記載の情報処理装置は、前記第１共有鍵制御部が、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵が存在しない場合、コピー可能な鍵を生成して、この生成した鍵を前記第２の暗号鍵群の中にコピーすることを特徴としている。

【００３８】

本態様によると、前記第１共有鍵制御部は、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵が存在しない場合、コピー可能な鍵を生成して、この生成した鍵を前記第２の暗号鍵群の中にコピーする。これにより、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵が存在しない場合であっても、前記第２共有鍵制御部は前記第１の暗号鍵群に含まれる鍵を共用できるので、前記第２共有鍵制御部は前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデータ保持部にアクセスできる。

【００３９】

本発明の請求項５に記載の情報処理装置は、前記第１共有鍵制御部が、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵の位置を示す位置情報を前記所定の鍵のリンク情報として生成し、このリンク情報と共に前記所定の鍵を前記第２の暗号鍵群の中にコピ

ーすることを特徴とする請求項３記載の情報処理装置。

【００４０】

本態様によると、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵の位置を示す位置情報を前記所定の鍵のリンク情報として生成してコピーすることにより、第２共有鍵制御部では、前記所定の鍵のリンク情報を参照すれば、前記所定の鍵より下層の鍵の位置を確認できるので、前記所定の鍵より下層の鍵を前記第２の耐タンパーモジュールにコピーすることなく、前記第１の耐タンパーモジュールとの間で前記所定の鍵より下層の鍵を共用できる。その結果、前記第１の耐タンパーモジュール及び前記第１の耐タンパーモジュールとで前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体を二重持ちする非効率を回避できる。

【００４１】

本発明の請求項６に記載の情報処理装置は、前記第１共有鍵制御部が、前記第１の暗号鍵群に含まれる所定の鍵の位置情報及び前記第２の暗号鍵群に含まれる所定の鍵の位置情報を、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵のリンク情報として生成することを特徴としている。

【００４２】

本態様により、前記第１の暗号鍵群に含まれる所定の鍵の位置情報及び前記第２の暗号鍵群に含まれる所定の鍵の位置情報を、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵のリンク情報として生成することにより、前記下層の鍵のリンク情報を参照すれば前記下層の鍵を暗号化した親鍵の所在を認識できる。この結果、前記第１の耐タンパーモジュール及び前記第１の耐タンパーモジュールとで前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用する場合であっても、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵がどの鍵で暗号化されているかを容易に識別できる。

【００４３】

本発明の請求項７に記載の情報処理装置は、前記第１共有鍵制御部と前第２共有鍵制御部とは、共用の共有鍵制御部であることを特徴としている。

【００４４】

本態様により、前記第１共有鍵制御部と前第２共有鍵制御部とは、共用の共有鍵制御部で構成が可能となり、１つの共有鍵制御部にて、２つのステークホルダー間の共有鍵を統括的に制御することが可能なので、より柔軟にアクセス制御を行うことが可能となる。

【００４５】

また、同じ共有鍵を保持している場合、自身以外の共有先のステークホルダーの脆弱性が原因で、共有していた鍵が暴露される危険性がある。そのため、その自身以外の共有先のステークホルダーが、不正と判断（リボーク対象もしくは、改竄されていると検知）された場合には、そのステークホルダーから、共有鍵で暗号化されているデータを利用不可にするためのアクセス制御が必要となる。

【００４６】

本発明の請求項８に記載の情報処理装置は、前記第１のステークホルダーが管理する第１ステークホルダー環境が、前記第２のステークホルダーが管理する第２のステークホルダー環境が改竄された、もしくはリボーク対象であることを検知した場合、前記第１共有鍵制御部は、前記所定の鍵と置換える代替鍵を前記第１の耐タンパーモジュールに生成させ、前記所定の鍵を親鍵とするツリー構造に含まれる鍵を前記代替鍵で再暗号化させると共に前記前記所定の鍵の親鍵を用いて前記代替鍵を暗号化させて、前記第２共有鍵制御部による前記所定のデータの利用を排除することを特徴としている。

【００４７】

本態様によると、前記第１のステークホルダーが管理する第１ステークホルダー環境が、前記第２のステークホルダーが管理する第２のステークホルダー環境が改竄された、もしくはリボーク対象であることを検知した場合、前記第１の耐タンパーモジュールは前記所定の鍵と置換える代替鍵を生成して前記所定の鍵を親鍵とするツリー構造に含まれる鍵を前記代替鍵で再暗号化すると共に前記所定の鍵の親鍵を用いて前記代替鍵を暗号化する

。この結果、前記第２共有鍵制御部は前記所定の鍵を用いて前記代替鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、前記代替鍵を親鍵とするツリー構造に含まれる鍵で暗号化された所定のデータを利用できず、前記所定データを不正な利用から保護できる。

【００４８】

本発明の請求項９に記載の情報処理装置は、前記第１のステークホルダーが管理する第１ステークホルダー環境が、前記第２のステークホルダーが管理する第２のステークホルダー環境が改竄された、もしくはリボーク対象であることを検知した場合、前記第１共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵を用いて前記第１の耐タンパーモジュールに前記所定のデータを暗号化し直させて、前記第２共有鍵制御部による前記所定の鍵の使用を排除することを特徴とする。

【００４９】

本態様によると、前記第２のステークホルダー環境が攻撃されたことを、第１のステークホルダー環境が検知した場合、前記第１の耐タンパーモジュールは前記所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵を用いて前記所定のデータを暗号化し直す。この結果、前記第２共有鍵制御部は前記所定の鍵を親鍵とするツリー構造に含まれる鍵を用いては前記所定のデータを復号化できないので、前記所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵で暗号化された所定のデータを利用できず、前記所定データを不正な利用から保護できる。

【００５０】

本発明の請求項１０に記載の情報処理装置は、前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵が、属性情報として、前記改竄のない第２のステークホルダーが管理する第２ステークホルダー環境のハッシュ値から生成された期待値として鍵利用制限情報を有し、第２の耐タンパーモジュールは、前記第２のステークホルダー環境のハッシュ値から生成された実際の値としての環境情報を記憶し、第２の共有鍵制御部から前記第１の共有鍵制御部に対して前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵の利用を依頼するときに、第２の共有鍵制御部は、前記鍵利用制限情報と前記環境情報とを比較し、比較結果が正しい場合にのみ前記鍵を利用させるように制限をすることを特徴する。

【００５１】

本態様によると、第２の共有鍵制御部は、前記改竄のない第２のステークホルダー環境から生成された鍵利用制限情報と前記第２のステークホルダー環境から実際に得られた環境情報とを比較し、比較結果が正しい場合にのみ前記鍵を利用させる。この結果、前記第２ステークホルダー環境が改竄され若しくはリボークされた場合には前記比較結果は不一致となって、前記第２共有鍵制御部は前記所定の鍵を用いては前記所定の鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、前記所定の鍵を親鍵とするツリー構造に含まれる鍵用いて暗号化されている前記所定のデータを復号化できず、前記所定データを不正な利用から保護できる。

【００５２】

本発明の請求項１１に記載の情報処理装置は、前記第１ステークホルダーが管理する第１のステークホルダー環境が、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵を、第２の共有鍵制御部から利用できないように、前記鍵利用制限情報を書き換えることを特徴とする。

【００５３】

本態様によると、前記第１ステークホルダーが管理する第１のステークホルダー環境は、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵を、第２の共有鍵制御部から利用できないように、前記鍵利用制限情報を書き換える。この結果、前記第２共有鍵制御部は前記所定の鍵を用いては前記所定の鍵を親鍵とするツリ

ー構造に含まれる鍵を復号化できないので、前記所定の鍵を親鍵とするツリー構造に含まれる鍵用いて暗号化されている前記所定のデータを復号化できず、前記所定データを不正な利用から保護できる。

【００５４】

本発明の請求項１２に記載の情報処理装置は、前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データが、属性情報として、前記改竄のない第２のステークホルダーが管理する第２ステークホルダー環境のハッシュ値から生成された期待値である暗号化データ利用制限情報を有し、第２の耐タンパーモジュールは、前記第２のステークホルダー環境のハッシュ値から生成された実際の値としての環境情報を記憶し、第２の共有鍵制御部から前記第１の共有鍵制御部に対して前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化されたデータの復号処理を依頼するときに、第２の共有鍵制御部は、前記暗号化データ利用制限情報と前記環境情報とを比較し、比較結果が正しい場合にのみ前記暗号化データの復号処理させるように制限をすることを特徴する。

【００５５】

本態様によると、第２の共有鍵制御部は、前記改竄のない第２のステークホルダー環境から生成された暗号化データ利用制限情報と前記第２のステークホルダー環境から実際に得られた環境情報とを比較し、比較結果が正しい場合にのみ暗号化データを復号させる。この結果、前記第２ステークホルダー環境が改竄され若しくはリボークされた場合には前記比較結果は不一致となって、前記第２共有鍵制御部は前記所定の鍵を用いては前記所定の鍵を親鍵とするツリー構造に含まれる鍵を用いて暗号化されている前記暗号化データを復号化できず、前記暗号化データを不正な利用から保護できる。

【００５６】

本発明の請求項１３に記載の情報処理装置は、前記第１ステークホルダーが管理する第１のステークホルダー環境が、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データを、第２の共有鍵制御部から利用できないように、前記暗号化データ利用制限情報を書き換えることを特徴とする。

【００５７】

本態様によると、前記第１ステークホルダーが管理する第１のステークホルダー環境は、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データを、第２の共有鍵制御部から利用できないように、前記暗号化データ利用制限情報を書き換える。この結果、前記第２共有鍵制御部は前記所定の鍵を用いては前記所定の鍵を親鍵とするツリー構造に含まれる鍵を用いた復号処理ができないので、前記所定データを不正な利用から保護できる。

【００５８】

本発明の請求項１４に記載の情報処理装置は、前記第１共有鍵制御部は第１ステークホルダー環境および第２ステークホルダー環境を環境に対して完全性をチェックしてから改竄されていない環境のみを起動する機能であるセキュアブートによってブートする際に、第２ステークホルダー環境が改竄された、もしくはリボーク対象であることを検知することを特徴とする。

【００５９】

本態様によると、前記第１共有鍵制御部は第１ステークホルダー環境および第２ステークホルダー環境を環境に対して完全性をチェックしてから改竄されていない環境のみを起動する機能であるセキュアブートによってブートする際に、第２ステークホルダー環境が改竄された、もしくはリボークされたことを検知することにより、前記第２共有鍵制御部の外部からの攻撃を判断できる。

【００６０】

本発明の請求項１５に記載の情報処理装置は、前記第１共有鍵制御部が、外部のサーバ

ーから、前記第２共有鍵制御部が改竄されたもしくはリボーク対象である旨の通知を受けることで前記第２共有鍵制御部が外部から攻撃されたことを検知することを特徴とする。

【００６１】

本態様によると、前記第１共有鍵制御部は、外部のサーバーから、前記第２共有鍵制御部が外部から攻撃された旨の通知を受けることにより、前記第２共有鍵制御部の外部からの攻撃を検知できる。

【００６２】

本発明の請求項１６に記載の暗号鍵の管理方法は、第１のステークホルダーに対応する第１共有鍵制御部と、第２のステークホルダーに対応する第２共有鍵制御部と、前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置における暗号鍵の鍵管理方法であって、前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断し、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーし、前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することを特徴とする。

【００６３】

本発明の請求項１７に記載のコンピュータプログラムは、第１のステークホルダーに対応する第１共有鍵制御部と、第２のステークホルダーに対応する第２共有鍵制御部と、前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置における暗号鍵の鍵管理に用いるコンピュータプログラムであって、コンピュータに対して、前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断する処理と、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーする処理と、を実行させ、前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用する処理を実行させることを特徴とする。

【００６４】

本発明の請求項１８に記載の集積回路は、第１のステークホルダーに対応する第１共有鍵制御部と、第２のステークホルダーに対応する第２共有鍵制御部と、前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備

する情報処理装置に用いる集積回路であって、情報処理部と、この情報処理部に対して、前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断する処理と、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーする処理と、を実行させ、前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用する処理を実行させる処理プログラムを格納したメモリと、を具備した集積回路とする。

【００６５】

発明の請求項１９に記載の情報処理装置は、前記第１共有鍵制御部は、前記第１の暗号鍵群に含まれる暗号鍵で暗号化されている前記所定の鍵を、前記暗号鍵を用いて復号し、復号された鍵を、前記第２の暗号鍵群に含まれる暗号鍵で再暗号化し、再暗号化された鍵を、前記第２の暗号鍵群の中にコピーすることを特徴とする。

【００６６】

以下、本発明の実施の形態について、図面を参照しながら説明する。

【００６７】

（実施の形態１）

本発明の実施の形態について、説明する。以降、本実施の形態で示すＴＰＭは、ＴＣＧのＭｏｂｉｌｅ Ｐｈｏｎｅ Ｗｏｒｋｉｎｇ Ｇｒｏｕｐで仕様化されているＭｏｂｉｌｅ Ｔｒｕｓｔｅｄ Ｍｏｄｕｌｅ（ＭＴＭ）の機能を有する耐タンパーモジュールであるとして説明する。

【００６８】

＜図１：システム概要＞

まず、マルチステークホルダーについて携帯電話機を例として説明する。携帯電話機には、デバイスメーカー、キャリア、アプリケーションサービス提供者、ユーザーといった複数のステークホルダーが存在する。各ステークホルダーは、各自の権利物を所有している。

【００６９】

各ステークホルダーの権利物は、例えば、デバイスメーカーであれば、Ｉｎｔｅｒｎａｔｉｏｎａｌ Ｍｏｂｉｌｅ Ｅｑｕｉｐｍｅｎｔ Ｉｄｅｎｔｉｔｙ（ＩＭＥＩ）であり、キャリアであれば、Ｓｕｂｓｃｒｉｂｅｒ Ｉｄｅｎｔｉｆｉｃａｔｉｏｎ Ｍｏｄｕｌｅ（ＳＩＭ）関連情報であり、アプリケーションサービス提供者であれば、サービス提供されたデータであり、ユーザーであれば、アドレス帳が挙げられる。

【００７０】

マルチステークホルダーモデルとは、それぞれのステークホルダーが利用するＴＰＭを、個別に割り当てることでそれぞれの権利物を安全に利用するモデルである。

【００７１】

図１は、マルチステークホルダーモデルにおけるシステム全体を示した図である。

【００７２】

情報処理端末１０は、ＭＴＭを搭載したモバイル端末である。実施の形態１では、以降、情報処理装置１０内には、第１のステークホルダーと第２のステークホルダーとの２つのステークホルダーが存在するという例を用いて説明する。なお、ステークホルダーは前述しているように、２つ以上であってもよい。また、情報処理端末は、携帯電話機であってもよいし、ＰＤＡなどのモバイル端末、あるいは、ＴＶやＤＶＤやＢＤプレイヤーやＳＴＢなどの据え置き型の電子機器であってもよい。

【００７３】

第１のステークホルダー管理サーバー１１は、第１のステークホルダー環境を提供するステークホルダーが管理しているサーバーである。第１のステークホルダー管理サーバー１１は、認証ＰＣＲデータベース１２、証明書管理データベース１３及びリボケーションリスト１４を管理している。

【００７４】

認証ＰＣＲデータベース１２は、第１のステークホルダー管理サーバー１１が、情報処理端末１０の正当性を検証するＡｔｔｅｓｔａｔｉｏｎ処理の際に利用するデータベースであり、正当な情報処理端末１０の環境情報であるＰＣＲの期待値のデータベースである。

【００７５】

第１のステークホルダー管理サーバー１１は、Ａｔｔｅｓｔａｔｉｏｎ時に、情報処理端末１０から送信されたＰＣＲ値と、認証ＰＣＲデータベース１２の記録している値を比較し、一致すれば正当な情報処理端末１０と判断し、適切なサービス等を提供する。

【００７６】

証明書データベース１３は、第１のステークホルダーから提供されるソフトウェアの証明書のデータベースである。第１のステークホルダーのモジュールの更新が必要であれば、更新すべきモジュールとともに、更新モジュールの証明書を送信する。また、更新モジュールの証明書も、証明書データベース１３で管理される。

【００７７】

リボケーションリスト１４は、リボーク対象であるステークホルダーのリストを記録しているデータベースである。リボケーションリスト１４は、第１のステークホルダー管理サーバー１１から情報処理端末１０へ送信され、情報処理端末１０内で完全性を保護した状態で管理される。

【００７８】

なお、リボケーションリスト１４は、無効化すべきステークホルダーの情報を記載したブラックリスト方式として説明するが、有効なステークホルダーの情報を記載したホワイトリスト方式を用いてもよい。

【００７９】

なお、図１では、認証ＰＣＲデータベース１２、証明書データベース１３及びリボケーションリスト１４は、第１のステークホルダー管理サーバー１１が、全て管理しているが、複数の管理サーバーで管理してもよい。

【００８０】

第２のステークホルダー管理サーバー１６は、第２のステークホルダー環境を提供するステークホルダーが管理しているサーバーである。図１で図示していないが、第２のステークホルダーも、第１のステークホルダー管理サーバー１１と同様に、認証ＰＣＲデータベース１２、証明書管理データベース１３及びリボケーションリスト１４を管理している。

【００８１】

＜図２：マルチステークホルダーモデルにおけるトラストモデル＞

図２は、マルチステークホルダーモデルにおけるトラストモデルを示した図である。

【００８２】

トラストモデルとして、３つのモデルが定義される。また、図２には図示していないが、各ステークホルダーは、各自のステークホルダーの権利物を管理しており、ステークホルダーの所有する権利物は、各ステークホルダーに対応づけられたＴＰＭを利用して、安全にアクセスされる。以下、それぞれ３つのモデルについて説明する。

【００８３】

図２（ａ）は、Ｉｎｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌを示している。このモデルは、各ステークホルダー間に信頼の依存関係はないモデルである。例えば、ステークホルダー１（２１）が、ＴＰＭ１（２３）を利用し、ステークホルダー２（２２）は、ＴＰＭ２（２４）を利用するモデルである。

【００８４】
図２（ｂ）は、Ｉｎｔｅｒｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌを示している。このモデルは、ステークホルダー間で、部分的に依存関係のあるモデルである。例えば、ステークホルダー１（３１）は、ＴＰＭ１（３３）を利用し、ステークホルダー２（３２）は、ＴＰＭ２（３４）を利用する。ここまでは、Ｉｎｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌと同じであるが、図２（ｂ）に示しているように、一部領域が重なっている部分が存在する。これは、ステークホルダー２（３２）が、ＴＰＭ２（３４）以外に、ＴＰＭ１（３３）の機能を利用することを概念的に表している。
【００８５】
例えば、携帯電話機の場合、ステークホルダー２（３２）が、キャリアであって、ステークホルダー１（３１）がデバイスメーカーであった場合、キャリアが、デバイスメーカーの権利物であるＩＭＥＩにアクセスするといった例である。この場合、ステークホルダー２（３２）は、ステークホルダー１（３１）に対して、ＩＭＥＩアクセス要求をすることになるため、ステークホルダー２（３２）は、ステークホルダー１(３１)経由でＴＰＭ１（３３）を利用することになる。
【００８６】
図２（ｃ）は、Ｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌを示している。このモデルは、ステークホルダー間で、あるステークホルダーが、別のステークホルダーに完全に依存するモデルである。これは、ステークホルダー１（４１）がＴＰＭ１（４３）を利用し、ステークホルダー２（４２）もＴＰＭ１（４３）を利用するモデルである。携帯電話機の場合、ステークホルダー２が、キャリアであって、ステークホルダー１がデバイスメーカーであった場合、デバイスメーカーの権利物であるＩＭＥＩは、ＴＰＭ１（４３）の機能により保護され、キャリアの権利物であるＳＩＭ情報もＴＰＭ１（４３）の機能を利用して安全に保護される。
【００８７】
＜図３：情報処理端末＞
図３は、マルチステークホルダーモデルの情報処理端末１０の全体構成図である。
【００８８】
情報処理端末１０は、第１のステークホルダープログラム１００、第２のステークホルダープログラム２００、共有鍵制御部（１１１，２１０）、耐タンパーモジュール（１２０，２２０）、鍵格納部３０、暗号化データ格納部４０、及び、ステークホルダー証明書格納部５０から構成される。また、図示していないが、情報処理端末１０は、ＣＰＵ、Ｉ／Ｏデバイス、ＲＡＭなどの揮発メモリ、ＲＯＭやＦｌａｓｈメモリなどの不揮発メモリなどのハードウェア群を保持する。
【００８９】
＜ステークホルダープログラム＞
第１のステークホルダープログラム１００は、第１のステークホルダーから提供されるプログラム群であり、第１のステークホルダー管理サーバー１１から配布されるものである。第１のステークホルダープログラム１００は、耐タンパーモジュール１２０のセキュアブート機能により、正当性を検証されたプログラムのみが起動される。
【００９０】
第２のステークホルダープログラムは、第２のステークホルダーから提供されるプログラム群であり、第２のステークホルダー管理サーバー１６から配布されるものである。第２のステークホルダープログラム２００は、耐タンパーモジュール２２０のセキュアブート機能により、正当性を検証されたプログラムのみが起動される。
【００９１】
なお、セキュアブートの仕様については、非特許文献４に詳細に記載されているので説明を省略する。
【００９２】
＜共有鍵制御部（１１０、２１０）＞

共有鍵制御部１（１１０）は、鍵群１（１３０）と共有鍵群（３３０）との利用制御を行う部であり、マルチステークホルダーモデル判定部１（１１１）と、共有許可設定部１（１１２）と鍵管理テーブル１（１１３）とから構成される。

【００９３】

マルチステークホルダーモデル判定部１（１１１）は、共有鍵制御部１（１１０）が管理している鍵群１（１１３）もしくは共有鍵（３３０）に対して、ステークホルダー１以外のステークホルダーから共有鍵の設定の要求があった場合に、要求元のステークホルダーが、鍵を共有してよいステークホルダーであるかどうかをステークホルダー証明書格納部５０に格納しているステークホルダー証明書（１５０，２５０）を参照して判断する。

【００９４】

共有許可設定部１（１１２）は、鍵群１（１３０）に属する鍵を、共有鍵群（３３０）の共有鍵として設定したり、新規に鍵を生成したり、鍵のマイグレート処理を制御したりと、各種共有鍵を設定する際に必要な鍵処理をする部である。ここでの鍵処理は、共有鍵制御部１（１１２）と、耐タンパーモジュール１（１２０）とが連携して行う。

【００９５】

鍵管理テーブル１（１１３）は、共有鍵制御部１（１１０）から、鍵群１（１３０）と共有鍵群（３３０）とにアクセスするために必要な情報が記載されているテーブルである。鍵管理テーブル１（１１３）は、図８を用いて後述する。

【００９６】

共有鍵制御部２（２１０）は、鍵群２（２３０）と共有鍵群（３３０）の利用制御を行う部であり、マルチステークホルダーモデル判定部２（２１１）と、共有許可設定部２（２１２）と、鍵管理テーブル２（２１３）から構成される。

【００９７】

マルスステークホルダーモデル判定部２（２１１）は、共有鍵制御部２（２１０）が管理している鍵群２（２３０）もしくは共有鍵（３３０）に対して、ステークホルダー２以外のステークホルダーから共有鍵の設定の要求があった場合に、要求元のステークホルダーが、鍵を共有してよいステークホルダーであるかどうかをステークホルダー証明書格納部５０に格納しているステークホルダー証明書（１５０，２５０）を参照して判断する。

【００９８】

共有許可設定部２（２１２）は、鍵群２（２３０）の鍵を、共有鍵群（３３０）の共有鍵として設定したり、新規に鍵を生成したり、鍵のマイグレート処理を制御したりと、各種共有鍵を設定する際に必要な鍵処理をする部である。ここでの鍵処理は、共有鍵制御部２（２１２）と、耐タンパーモジュール２（２２０）とが連携して行う。

【００９９】

鍵管理テーブル２（２１３）は、鍵群２（２３０）と共有鍵群（３３０）とにアクセスするために必要な情報が記載されているテーブルである。鍵管理テーブル２（２１３）は、図８を用いて後述する。

【０１００】

なお、共有鍵制御部１（１１０）及び共有鍵制御部２（２１０）は、それぞれ、第１のステークホルダープログラム、第２のステークホルダープログラムとして実現されていてもよい。これにより、共有鍵制御部１（１１０）及び共有鍵制御部２（２１０）は、ＴＣＧのモバイル仕様で規定されるセキュアブートで完全性が検証されてから起動することが可能となる。

【０１０１】

＜耐タンパーモジュール（１２０、２２０）＞

耐タンパーモジュール１（１２０）は、ＭＴＭ機能を有するものとして実装されているとして説明する。そのため、耐タンパーモジュール１（１２０）は、暗復号処理や署名生成・検証処理やＴＰＭ機能処理などのセキュアな処理や、鍵の制御処理をする際に、第１のステークホルダープログラム及び共有鍵制御部１（１１０）などから利用される。さらに、耐タンパーモジュール１（１２０）は、耐タンパーモジュール１内の不揮発性メモリ

上にルート鍵１（１２１）を保持する。このルート鍵１（１２１）は、ＴＣＧにおけるＳＲＫに相当する鍵である。

【０１０２】

同様に、耐タンパーモジュール２（２２０）は、ＭＴＭ機能を有するものとして実装されているとして説明する。そのため、耐タンパーモジュール２（２２０）は、暗復号処理や署名生成・検証処理やＴＰＭ機能処理などのセキュアな処理や、鍵の制御処理をする際に、第２のステークホルダープログラムおよび共有鍵制御部２（２１０）などから利用される。さらに、耐タンパーモジュール２（２２０）は、耐タンパーモジュール１内の不揮発なメモリ上にルート鍵２（２２１）を保持する。このルート鍵２（２２１）は、ＴＣＧにおけるＳＲＫに相当する鍵である。

【０１０３】

＜鍵格納部３０＞

鍵群１（１３０）と共有鍵群３３０とは、ルート鍵１（１２１）をルートとした階層的ツリー構造のノード鍵として構成される。これは、ＴＣＧのＰｒｏｔｅｃｔｅｄ　Ｓｔｏｒａｇｅ機能を実現するための鍵ツリー構造に相当する。

【０１０４】

鍵群１（１３０）は、１つ以上の鍵から構成されたツリー構造を有する鍵群である。鍵群１（１３０）の個々の鍵は、共有鍵制御部１（１１０）から耐タンパーモジュール１（１２０）を経由して暗復号化処理や署名生成・検証処理に利用される。

【０１０５】

鍵群２（２３０）と共有鍵群３３０とは、ルート鍵２（２２１）をルートとした階層的ツリー構造のノード鍵として構成される。これは、ＴＣＧのＰｒｏｔｅｃｔｅｄ　Ｓｔｏｒａｇｅ機能を実現するための鍵ツリー構造に相当する。

【０１０６】

鍵群２（２３０）は、１つ以上の鍵から構成されたツリー構造を有する鍵群である。鍵群２（２３０）の個々の鍵は、共有鍵制御部２（２１０）から耐タンパーモジュール２（２２０）を経由して暗復号化処理や署名生成・検証処理に利用される。

【０１０７】

一方、共有鍵群３３０は、１つ以上の鍵から構成されたツリー構造を有する鍵群である。共有鍵群３３０の個々の鍵は、共有鍵制御部１（１１０）から耐タンパーモジュール１（１１０）経由で暗復号化処理や署名生成・検証処理に利用される。共有鍵群３３０は、共有鍵制御部２（２１０）から耐タンパーモジュール２（２２０）経由でも暗復号化処理や署名生成・検証処理に利用される。

【０１０８】

共有鍵群３３０は、共有鍵と、その共有鍵を保護するための共有鍵の親鍵とから構成される。共有鍵の親鍵は、共有鍵制御部１（１１０）から耐タンパーモジュール１（１１０）経由でのみ利用される鍵と、共有鍵制御部２（２１０）から耐タンパーモジュール２（２１０）経由でのみ利用される鍵とから構成される。　これら鍵群１（１３０）、鍵群２（２３０）及び共有鍵群（３３０）は、鍵格納部３０に格納される。鍵群１（１３０）、鍵群２（２３０）及び共有鍵群（３３０）の構造については、図４から図７を用いてさらに詳しく説明する。

【０１０９】

＜暗号化データ格納部４０＞

暗号化データ１（１４０）は、鍵群１（１３０）の鍵で暗号化されたデータである。暗号化共有データ３４０は、共有鍵群３３０の鍵で暗号化されたデータである。暗号化データ２（２４０）は、鍵群２（２３０）の鍵で暗号化されたデータである。

【０１１０】

暗号化データ１（１４０）、暗号化データ２（２４０）、及び暗号化共有データ３４０は、暗号化データ格納部４０に格納される。暗号化データ格納部４０は、ＨＤＤやフラッシュメモリなどの不揮発メモリで構成される。

【０１１１】
なお、図３では、暗号化データ１（１４０）、暗号化データ２（２４０）、及び暗号化共有データ３４０は、暗号化データとしているが、暗号化データに限定されず、それぞれの鍵で署名生成したデータであってもよい。
【０１１２】
＜ステークホルダー証明書格納部５０＞
ステークホルダー証明書格納部５０は、ステークホルダー証明書１（１５０）とステークホルダー証明書２（２５０）とを格納する部である。ステークホルダー証明書格納部５０は、不揮発メモリで実現され、完全性が保護された形で管理される。
【０１１３】
ステークホルダー証明書１（２５０）は、第１のステークホルダープログラム、共有鍵制御部１（１１０）、及び耐タンパーモジュール１（１２０）が、正規のステークホルダーから提供されてことを示す証明書である。
【０１１４】
ステークホルダー証明書２（２５０）は、第２のステークホルダープログラム、共有鍵制御部２（２１０）、及び耐タンパーモジュール２（２２０）が、正規のステークホルダーから提供されてことを示す証明書である。
【０１１５】
ステークホルダー証明書（１５０、２５０）は、それぞれ、依存関係のあるステークホルダーを識別できる情報が記載される。ステークホルダー証明書（１５０，２５０）の構成などの詳細は、図９及び図１０を用いて後述する。
【０１１６】
＜図４：共有鍵の鍵ツリー構成＞
図４は、鍵群１（１３０）と鍵群２（２３０）と共有鍵群（３３０）とのツリー構成を表した図である。なお、図３で既に説明している構成要素については、説明を省略する。図４では、図３における耐タンパーモジュール（１２０，２２０）、鍵群１（１３０）、鍵群２（２３０）、及び共有鍵群（３３０）を、より詳細に示したものである。
【０１１７】
耐タンパーモジュール１（１２０）は、耐タンパーモジュール１（１２０）の外部から不正なアクセスができないように保護されたセキュアメモリ（１２２）と、１６個のＰＣＲ（１２３）とを備えている。ルート鍵１（１２１）は、セキュアメモリ（１２２）に安全に保持される。
【０１１８】
耐タンパーモジュール２（２２０）は、耐タンパーモジュール２（２２０）の外部から不正なアクセスができないように保護されたセキュアメモリ（２２２）と、１６個のＰＣＲ（２２３）とを備えている。ルート鍵２（２２１）は、セキュアメモリ（２２２）に安全に保持される。
【０１１９】
なお、ＰＣＲ（１２０、２２０）は、Ｐｌａｔｆｏｒｍ Ｃｏｎｆｉｇｒａｔｉｏｎ Ｒｅｇｉｓｔｅｒｓと呼ばれるレジスタであり、ＴＣＧのＴＰＭ＿Ｅｘｔｅｎｄコマンドにより生成されたインテグリティ値が格納される。なお、ＰＣＲの個数は、１６個に限定されるわけではなく、それより多くても少なくても良い。実施の形態では、ＴＣＧの仕様で決められている個数以上の数を備えるものとする。
【０１２０】
以降、ルート鍵１（１２１）をＫ１０、ルート鍵２（２２１）をＫ２０として説明する。
【０１２１】
鍵群１（１３０）は、３つの鍵（Ｋ１１、Ｋ１２、Ｋ１３）から構成されている。Ｋ１１は、Ｋ１０の子供の鍵としてツリー構造化され、Ｋ１１はＫ１０により暗号化される。Ｋ１２及びＫ１３は、共にＫ１１の子供の鍵としてツリー構造化され、Ｋ１２及びＫ１３

はＫ１１で暗号化される。

【０１２２】

鍵群２（２３０）は、３つの鍵（Ｋ２１、Ｋ２２、Ｋ２３）から構成されている。Ｋ２１は、Ｋ２０の子供の鍵としてツリー構造化され、Ｋ２１はＫ２０により暗号化される。Ｋ２２及びＫ２３は、共にＫ２１の子供の鍵としてツリー構造化され、Ｋ２２及びＫ２３はＫ３１で暗号化される。

【０１２３】

共有鍵群３３０は、４つの鍵（Ｋ３１、Ｋ３２、Ｋ３３、Ｋ３４）から構成されている。Ｋ３１は、Ｋ１０の子供の鍵としてツリー構造化され、Ｋ３１はＫ１０により暗号化される。したがって、Ｋ３１は、共有鍵制御部１（１１０）から耐タンパーモジュール１（１１０）経由でのみ利用される鍵となる。なぜなら、Ｋ３１は、Ｋ１０で暗号化されているため、Ｋ３１の鍵値を利用する場合は、Ｋ１０を用いてＫ３１を復号しなければならないからである。

【０１２４】

一方、Ｋ３２は、Ｋ２０の子供の鍵としてツリー構造化され、Ｋ３２はＫ２０により暗号化される。したがって、Ｋ３２は、共有鍵制御部２（２１０）から耐タンパーモジュール２（２１０）経由でのみ利用される鍵となる。なぜなら、Ｋ３２は、Ｋ２０で暗号化されているため、Ｋ３２の鍵値を利用する場合は、Ｋ２０を用いてＫ３２を復号しなければならないからである。

【０１２５】

続いて、Ｋ３３及びＫ３４について説明する。Ｋ３３は、Ｋ３１及びＫ３２の子供の鍵としてツリー構造化され、Ｋ３４は、Ｋ３１及びＫ３２の子供の鍵としてツリー構造化されている。

【０１２６】

Ｋ３３及びＫ３４は、共有鍵制御部１（１１０）から耐タンパーモジュール１（１１０）経由で暗復号化処理や署名生成・検証処理に利用され、さらに共有鍵制御部２（２１０）から耐タンパーモジュール２（２２０）経由でも暗復号化処理や署名生成・検証処理に利用される共有鍵である。そのため、Ｋ３３及びＫ３４は、上述したように耐タンパーモジュール１（１１０）、及び、耐タンパーモジュール２（２１０）の両モジュールから利用可能な鍵である。

【０１２７】

このようにするために、Ｋ３１及びＫ３２は、同じ鍵値とする。親鍵が子供の鍵を暗号化する方式をとったツリー構造としているため、Ｋ３１及びＫ３２が同じ鍵値であれば、ルート鍵がＫ１０とＫ２０のように異なっていても、Ｋ３３及びＫ３４は復号できる。

【０１２８】

Ｋ３１及びＫ３２を同じ鍵値に設定する方法については、ＴＣＧにおけるマイグレート機能を利用する。具体的なフローについては図１１及び図１２を用いて後述するので、ここでの説明は省略する。

【０１２９】

また、暗号化データ１（１４０）は、暗号化データＤ１２とＤ１３とから構成されている。Ｄ１２は、Ｋ１２で暗号化されたデータであり、Ｄ１３は、Ｋ１３で暗号化されたデータである。なお、暗号化データ１（１４０）の暗号化データは、これに限定されることなく、Ｋ１２及びＫ１３のそれぞれを署名鍵として用い、署名されたデータであってもよい。また、Ｋ１２で、暗号化されたデータが複数あってもよいし、Ｋ１３で、暗号化されたデータが複数あってもよい。

【０１３０】

また、暗号化データ２（２４０）は、暗号化データＤ２２とＤ２３とから構成されている。Ｄ２２は、Ｋ２２で暗号化されたデータであり、Ｄ２３は、Ｋ２３で暗号化されたデータである。なお、暗号化データ２（２４０）の暗号化データは、これに限定されることなく、Ｋ２２及びＫ２３をそれぞれ署名鍵として用い、署名されたデータであってもよい

。また、Ｋ２２で、暗号化されたデータが複数あってもよいし、Ｋ２３で、暗号化されたデータが複数あってもよい。

【０１３１】

また、暗号化共有データ３４０は、暗号化共有データＤ３３とＤ３４とから構成されている。Ｄ３３は、Ｋ３３で暗号化されたデータであり、Ｄ３４は、Ｋ３４で暗号化されたデータである。なお、暗号化共有データ３４０の暗号化データは、これに限定されることなく、Ｋ３３及びＫ３４をそれぞれ署名鍵として用い、署名されたデータであってもよい。また、Ｋ３３で、暗号化されたデータが複数あってもよいし、Ｋ３４で、暗号化されたデータが複数あってもよい。

【０１３２】

なお、鍵群１（１３０）、鍵群２（２３０）、及び共有鍵群３３０の鍵の個数と、暗号化データ１（１４０）、暗号化データ２（２４０）、及び暗号化共有データ３４０のデータの個数は、図４で示しているものに限定されない。また、Ｋ１０、Ｋ２０をルートする鍵ツリー構造は、２分木で構成しているが、３分木、Ｎ分木（Ｎは整数）であってもよい。

【０１３３】

＜図５：鍵属性情報＞

ルート鍵１（１２１）、ルート鍵２（２２１）、鍵群１（１３０）、鍵群２（２３０）、共有鍵群３３０のそれぞれの鍵値は、鍵属性情報の１要素として鍵値を持つ。

【０１３４】

例えば、ルート鍵１（１２１）であるＫ１０の鍵値は、鍵属性情報４１０内に記録されている。ルート鍵２（２２１）であるＫ２０の鍵値は、鍵属性情報４２０内に記録されている。鍵群１（１３０）の鍵であるＫ１１の鍵値は、鍵属性情報４１１内に記録されている。鍵群２（２３０）の鍵であるＫ２１の鍵値は、鍵属性情報４２１内に記録されている。共有鍵群（３３０）の鍵であるＫ３１の鍵値は、鍵属性情報４３１内に記録されている。そのほかの鍵値も同様に記録されているので、説明を省略する。

【０１３５】

ここで、鍵属性情報は、鍵の値と共に、鍵の属性を示す情報も同じデータ構造として記録されている。

【０１３６】

図５は、鍵属性情報の構成を示した図である。鍵属性情報４３４は、マイグレート許可フラグ５０１と、共有許可フラグ５０２と、鍵のアルゴリズムを識別するための情報である暗号アルゴリズムと鍵値のサイズを示した５０３と、鍵値５０４とを備える。

【０１３７】

マイグレート許可フラグ５０１は、鍵のマイグレートが許可されているかどうかを示すフラグ情報であり、「０」であればマイグレート不可、「１」であればマイグレート許可を示す。

【０１３８】

共有許可フラグ５０２は、共有鍵として利用可能かどうかを示すフラグ情報であり、「０」であれば共有不可、「１」であれば共有許可を示す。

【０１３９】

鍵属性情報４１０は、さらに、鍵が、複数のステークホルダーから共有鍵としてアクセス可能な場合に、そのステークホルダーの情報を格納するステークホルダーフィールド５０５を有する。複数のステークホルダーからアクセス可能な鍵であれば、ステークホルダーフィールド５０５は、アクセスされるステークホルダーの数だけ列挙される。

【０１４０】

図５に示した鍵属性情報４３４は、ステークホルダー識別子１からｎまでのステークホルダーフィールド５０５が設定されている。

【０１４１】

図４では、Ｋ３４は、ステークホルダー１とステークホルダー２とからアクセス可能な

例であるので、ステークホルダーフィールド５０５は、２つ存在することになる。

【０１４２】

ステークホルダーフィールド５０５は、ステークホルダー識別子５０６と、鍵の利用制限を示す鍵利用制限情報５０７と、リンク情報５０８とを含む。

【０１４３】

鍵利用制限情報５０７は、鍵を利用する際に耐タンパーモジュールの備えるＰＣＲ（１２３、２２３）に記録されていることが期待されるＰＣＲ値である。鍵利用制限情報５０７は、耐タンパーモジュールの備えるＰＣＲ（１２３、２２３）に記録されている実際の値と比較され、実際のＰＣＲ値と期待されるＰＣＲ値とが等しい場合にのみ、鍵が利用できるように制限するための情報である。

【０１４４】

リンク情報５０８は、個々の鍵に対する親鍵を識別するためのリンク情報、もしくは、個々の鍵に対する子供の鍵を識別するためのリンク情報である。

【０１４５】

＜図６：リンク情報＞

ここで、図６を用いてリンク情報５０８の構造について説明する。

【０１４６】

図６（ａ）のリンク情報５０８は、個々の鍵に対する親鍵を識別するための情報である。複数の親鍵が存在するのであれば、図に示すようにリンク情報５０８には、複数の親鍵へのポインタ（６０１，６０２，６０３）が格納される。

【０１４７】

図６（ｂ）のリンク情報５０８は、個々の鍵に対する子供の鍵を識別するための情報である。複数の子供の鍵が存在するのであれば、図に示すようにリンク情報５０８には、複数の子供の鍵へのポインタ（６１１，６１２，６１３）が格納される。

【０１４８】

＜図７：鍵属性情報の例＞

図７は、鍵のツリー構成と鍵属性情報の関係の例を表した図である。図７では、図４に図示している一部の鍵について抜粋して説明する。

【０１４９】

ルート鍵１（Ｋ１０）の鍵属性情報４１０は、マイグレート不許可であり、共有不許可であり、暗号アルゴリズムがＲＳＡアルゴリズムで鍵長が２０４８ビットであることを示している。また、鍵値５０４のフィールドには、Ｋ１０の公開鍵の鍵値と、Ｋ１０の秘密鍵の鍵値とが設定されている。

【０１５０】

そして、Ｋ１０はステークホルダー１に対してアクセスを許可させるため、ステークホルダーフィールド５０５には、ステークホルダー識別子５０６として、ステークホルダー１の識別子「ＳＨ１」、鍵利用制限情報５０７に期待されるＰＣＲの情報としてＰＣＲ＿１０が示されている。さらに、リンク情報５０８には、図６（ａ）の親鍵へのリンク情報が設定される。Ｋ１０はルート鍵なので、親鍵は存在しないので、リンク情報５０８には「ＮＵＬＬ」と設定される。さらに、Ｋ１０は、親鍵が存在しないので、Ｋ１０の鍵値５０４のフィールドには、平文の鍵値が設定される。

【０１５１】

鍵群１（１３０）の鍵Ｋ１１の鍵属性情報４１１は、マイグレート許可であり、共有許可であり、暗号アルゴリズムがＲＳＡアルゴリズムで、鍵長が２０４８ビットであることを示している。また、鍵値５０４のフィールドには、Ｋ１１の公開鍵の鍵値と、Ｋ１０の公開鍵で暗号化されたＫ１１の秘密鍵の鍵値とが設定される。そしてＫ１１はステークホルダー１に対してアクセスを許可させるため、ステークホルダーフィールド５０５には、ステークホルダー識別子５０６として、ステークホルダー１の識別子「ＳＨ１」が設定される。さらに、リンク情報５０８には、図６（ａ）で説明した親鍵へのリンク情報が記載されている。Ｋ１１の親鍵はＫ１０であるので、リンク情報５０８には「Ｋ１０へのポイ

ンタ情報」が設定される。具体的に、このポインタ情報は、Ｋ１０へ鍵属性情報４１０を参照できる情報であれば、アドレスでもよいし、識別ＩＤでもよい。鍵利用制限情報５０７に期待されるＰＣＲの情報として「ＮＵＬＬ」として設定される。この「ＮＵＬＬ」は、Ｋ１１を利用する際のＰＣＲの制限はない鍵であることを示している。

【０１５２】

共有鍵群（３３０）のＫ３３の鍵属性情報４３３は、マイグレート許可であり、共有許可であり、暗号アルゴリズムがＡＥＳアルゴリズムで鍵長が２５６ビットであることを示している。また、鍵値５０４のフィールドには、Ｋ３１もしくはＫ３２の公開鍵で暗号化されたＫ３３鍵値が設定される。そしてＫ３３はステークホルダー１とステークホルダー２との間で共有できる共有鍵であるため、ステークホルダーフィールド５０５には、ステークホルダー識別子５０６として、ステークホルダー１の識別子「ＳＨ１」と「ＳＨ２」とが設定される。また、鍵利用制限情報５０７には、期待されるＰＣＲの情報として「ＳＨ１」からの利用制限に使うＰＣＲ＿３３＿１と「ＳＨ１」からの利用制限に使うＰＣＲ＿３３＿２とが記載されている。そして、リンク情報５０８には、図６（ａ）で説明した親鍵へのリンク情報が記載されている。Ｋ３３は共有鍵であり、親鍵はＫ３１とＫ３２であるので、リンク情報５０８には「Ｋ３１へのポインタ情報」と「Ｋ３２へのポインタ情報」が設定される。

【０１５３】

他の鍵（Ｋ２０、Ｋ３１、Ｋ３２、Ｋ３４）についても同様なので、説明を省略する。

【０１５４】

なお、実施の形態１では、暗号化アルゴリズムをＲＳＡ、または、ＡＥＳとしているが、暗号アルゴリズムはこれに限定されない。公開鍵暗号系であれば、ＲＳＡでなく楕円曲線暗号でもよい、また、共通鍵暗号系ではＡＥＳ以外のアルゴリズムでもよい。鍵長も本実施の形態１の例に限定はされない。また、親子関係で親鍵が公開鍵暗号系のアルゴリズムであれば、その鍵は、親鍵の公開鍵で暗号化される。また親鍵が共通鍵暗号系のアルゴリズムであれば、その鍵は、親鍵で暗号化される。また、子供鍵の公開鍵暗号系アルゴリズムの鍵であれば秘密鍵が暗号化対象となり、共通鍵暗号系であれば、その暗号鍵が暗号化対象となる。

【０１５５】

なお、鍵属性情報４３４は、鍵値５０５を含む構成としているが、鍵値５０５と、それ以外の属性情報は別のデータとして構成するようにしてもよい。

【０１５６】

＜図８：鍵管理テーブル＞

次に、鍵管理テーブルについて説明する。

【０１５７】

鍵管理テーブル１（１１３）は、共有鍵制御部１（１１０）が利用するテーブルである。鍵管理テーブル１（１１３）は、鍵ＩＤ８１１と鍵属性情報アドレス８１２とから構成される。鍵ＩＤ８１１は、各鍵を識別するための識別子である。鍵属性情報アドレス８１２は、各鍵ＩＤ８１１に対応する鍵属性情報が格納されているアドレス値が設定される。この２つの情報を利用することで、共有鍵制御部１（１１３）は、所望の鍵にアクセスする。

【０１５８】

鍵管理テーブル２（２１３）は、共有鍵制御部２（２１０）が利用するテーブルである。鍵管理テーブル２（２１３）は、鍵ＩＤ８２１と鍵属性情報アドレス８２２とから構成される。鍵ＩＤ８２１は、各鍵を識別するための識別子である。鍵属性情報アドレス８２２は、各鍵ＩＤ８２１に対応する鍵属性情報が格納されているアドレス値が設定される。この２つの情報を利用することで、共有鍵制御部２（２１３）は、所望の鍵にアクセスする。

【０１５９】

ここで、共有鍵として設定されている鍵Ｋ３３及びＫ３４は、鍵管理テーブル１（１１

３）と鍵管理テーブル２（２１３）との両テーブルに登録されている。

【０１６０】

＜図９：ステークホルダー証明書＞

次に、図９を用いて、ステークホルダー証明書について説明する。ステークホルダー証明書１（１５０）、及び、ステークホルダー証明書２（２５０）は、すべて同じフォーマットであるとする。具体的には、Ｘ．５０９形式のフォーマットを利用する。

【０１６１】

ステークホルダー証明書（ＴＰＭ証明書）は、Ｘ．５０９のバージョンを示す証明書バージョン９０１、発行者によって一意な値を割り振られたシリアルナンバー９０２、証明書の署名検証に用いる署名アルゴリズムを示す署名アルゴリズム情報９０３、発行者情報９０４、証明書の有効期間９０５、証明書を受ける対象を示したサブジェクト９０６、鍵値や公開鍵アルゴリズムを示す公開鍵情報９０７、ＴＰＭバージョン９０８、トラストモデル識別情報９０９、依存ステークホルダー証明書識別情報９１０、拡張領域９１１、及び、これらのデータに対する署名データ９１２から構成される。

【０１６２】

拡張領域９１１には、ＣＲＬやＩＳＯ９０００００などの製造プロセスや、ＥＡＬなどのコモンクライテリアといったセキュリティ関連情報を記載してもよいし、機能制御の条件と機能制御の内容が記載されてもよい。

【０１６３】

本実施の形態では、トラストモデル識別情報９０９と依存ステークホルダー証明書識別情報９１０とを用いてトラストモデルを定義している。以下、これらの構成について詳細に説明する。

【０１６４】

なお、本実施の形態では、Ｘ．５０９形式のフォーマットとしているが、これ以外のフォーマットであってもよい。例えば、ＭＴＭ仕様で規定されているＲＩＭ証明書のフォーマットを利用してもよい。ＲＩＭ証明書を利用することで、ＭＴＭの証明書検証用のコマンドを利用して、証明書検証を行うことが可能となる。ＲＩＭ証明書については、非特許文献４に詳細に記載されているので、ここでの説明を省略する。

【０１６５】

＜図１０：ステークホルダー間の依存関係＞

トラストモデル情報識別情報９０９は、３つのトラストモデルであるＩｎｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌ、Ｉｎｔｅｒｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌ、及び、Ｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌを識別するため情報が記載される。

【０１６６】

依存ステークホルダー証明書識別情報９１０は、トラストモデルにおける信頼関係のあるステークホルダー証明書へのポインタ情報を格納する。

【０１６７】

図１０（ａ）は、Ｉｎｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌの具体例である。この例では、Ｉｎｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌを示すトラストモデル識別情報を「００１」としている。ステークホルダー１とステークホルダー２との間で依存関係がないモデルであるので、ＣＥＲＴ００１とＣＥＲＴ００２の依存ステークホルダーモデル識別情報９１０には「ＮＵＬＬ」と設定されている。

【０１６８】

図１０（ｂ）は、Ｉｎｔｅｒｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌの具体例である。この例は、ステークホルダー２がステークホルダー１に対して信頼の依存関係があるモデルである。そのため、ＣＥＲＴ００２の依存ステークホルダー識別情報９１０には、信頼の依存先ステークホルダーである「ＣＥＲＴ００１」と設定されている。

【０１６９】

図１０（ｃ）は、Ｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌの具体例である。この例は、ステークホルダー２がステークホルダー１に対して信頼の依存関係があるモデルである。そのた

め、ＣＥＲＴ００２の依存ステークホルダー識別情報９１０には、信頼の依存先ステークホルダーである「ＣＥＲＴ００１」と設定されている。

【０１７０】

＜図１１、１２：共有鍵設定フロー＞

図１１及び図１２は、共有鍵制御部１（１１０）で管理している鍵群１（１３０）の鍵について、共有鍵制御部２（２１０）から共有鍵としての利用要求があった場合のフローである。

【０１７１】

図４の鍵構成を例にフローの概要を説明すると、耐タンパーモジュール１（１２０）は、Ｋ３１を耐タンパーモジュール２（２２０）にマイグレートし、耐タンパーモジュール２（２２０）は、マイグレートされたＫ３１をＫ３２として管理し、Ｋ３１の子供鍵であるＫ３３とＫ３４とを、ステークホルダー１とステークホルダー２の共有鍵として設定する。

【０１７２】

ここでは、図１１及び図１２を用いて、共有鍵の設定フローの詳細を説明する。

【０１７３】

まず、共有鍵制御部２（２１０）は、共有鍵制御部１（１１０）に対して、共有鍵設定要求データとして、ステークホルダー２を識別するＩＤと、ルート鍵２であるＫ２０の公開鍵を送付する（ステップＳ１１０１）。ここで、共有鍵制御部２（２１０）から直接共有鍵制御部１（１１０）に対してデータを送信しているが、第２のステークホルダープログラムから第１のステークホルダープログラムに対してＳ１１０１の要求を出すようにし、第１のステークホルダープログラムは、共有鍵制御部１（１１０）に対して処理要求を出し、第２のステークホルダープログラムは、共有鍵制御部２（２１０）に対して処理要求を出すようにしてもよい。

【０１７４】

次に、共有鍵制御部１（１１０）は、ステークホルダー証明書格納部５０から、Ｓ１１０１で受信したＩＤに対応するステークホルダー証明書をリードする（ステップＳ１１０３）。図４の例では、ステークホルダー証明書２（２５０）をリードする。

【０１７５】

次に、共有鍵制御部１（１１０）は、耐タンパーモジュール１（１２０）を利用してステークホルダー証明書の検証を行う（ステップＳ１１０３）。Ｓ１１０３の検証の結果、ステークホルダー証明書が正当でないと判断されたら、共有鍵制御部１（１１０）は、エラー処理へと処理を移す（Ｓ１１３０）。Ｓ１１０３の検証の結果、証明書が正当であれば、ステップＳ１１０４に処理を移す。

【０１７６】

次に、ステークホルダー間の依存関係を、Ｓ１１０３で検証したステークホルダー証明書を用いてチェックする。ステークホルダー証明書のトラストモデル識別情報をチェックし、ＩｎｔｅｒｄｅｐｅｎｄｅｎｔもしくはＤｅｐｅｎｄｅｎｔ ｍｏｄｅｌであることを確認する。

【０１７７】

そして、ＩｎｔｅｒｄｅｐｅｎｄｅｎｔもしくはＤｅｐｅｎｄｅｎｔ ｍｏｄｅｌであることが確認できたら、ステークホルダー証明書格納部５０から、依存ステークホルダー識別子が参照しているステークホルダー証明書を参照し、依存先のステークホルダー証明書が正当であるかどうかを確認する（ステップＳ１１０４）。

【０１７８】

この結果、正当であると判断されれば、Ｓ１１０６へ処理を移す。そうでない場合、すなわち、トラストモデルがＩｎｄｅｐｅｎｄｅｎｔモデルである、もしくは依存先のステークホルダー証明書が正当でないと判断されれば、エラー処理へと処理を移る（ステップＳ１１３０）。

【０１７９】

図４の例では、ステークホルダー２とステークホルダー１とが依存関係であるかどうかを、ステークホルダー証明書１（１５０）と、ステークホルダー証明書２（２５０）ｔｐを用いて検証する。

【０１８０】

次に、共有鍵制御部１（１１０）は、鍵群１（１３０）の中にマイグレート可能で且つ共有許可な鍵が存在するかどうか探索する（ステップＳ１１０５）。図４の例では、Ｋ１０，Ｋ１１，Ｋ１２，Ｋ１３，Ｋ３１，Ｋ３３，Ｋ３４の中から探索する。

【０１８１】

もし、そのような鍵が存在しなければ、新たにマイグレート可能で且つ共有許可な鍵を生成する（ステップＳ１１０６）。鍵の生成処理は、耐タンパーモジュール１（１２０）で行う。

【０１８２】

そして、共有鍵制御部１（１１０）は、生成した鍵を鍵群１（１３０）の鍵として鍵管理テーブル１（１１３）に登録する（ステップＳ１１０７）。

【０１８３】

次に、Ｓ１１０５の探索により見つけたマイグレート可能で共有許可な鍵の中で、子供鍵を有するものがあるかどうかチェックする（ステップＳ１１０８）。もし、そのような鍵がなければ、ステップＳ１１０６へ処理を移し、マイグレート可能で共有許可な鍵の子供の鍵を生成する。図４の例では、Ｋ３１が子供鍵を有し、且つ、マイグレート可能で共有許可な鍵として選択される。

【０１８４】

次に、Ｓ１１０７で子供鍵を有するマイグレート可能で共有許可な鍵として選択された鍵のロード処理を行う（ステップＳ１１０９）。ここでのロード処理とは、親鍵で子供の鍵が暗号化されているので、ロード要求のあった鍵を、ルート鍵であるＫ１０からリーフ方向へ辿り、親子関係を元に復号処理することである。図４の例では、Ｋ３１をロードする。

【０１８５】

次に、共有鍵制御部１（１１０）は、Ｓ１１０９でロードした鍵を耐タンパーモジュール１（１２０）から耐タンパーモジュール２（２２０）にマイグレートするために、耐タンパーモジュール１（１２０）に対してマイグレート処理依頼を行う（ステップＳ１１１０）。以降、マイグレートされる鍵をマイグレート鍵と呼ぶことにする。図４の例では、Ｋ３１がマイグレート鍵であって、Ｋ３３とＫ３４が共有鍵として設定される。

【０１８６】

次に、マイグレート処理依頼を受けた耐タンパーモジュール１（１２０）は、マイグレート鍵をＳ１１０１で受信した公開鍵で暗号化し、共有鍵制御部１（１１０）に返す（ステップＳ１１１１）。図４の例では、Ｋ３１をＫ２０の公開鍵で暗号化する。

【０１８７】

次に、共有鍵制御部１（１１０）は、暗号化マイグレート鍵を共有鍵制御部２（２１０）へ送信する（ステップＳ１１１２）。なお、ここでは、共有鍵制御部１（１１０）及び共有鍵制御部２（２１０）間で直接データの送受信を行うことができるものとして説明しているが、直接でなく、別の第三者の制御部を経由して行うようにしてもよい。

【０１８８】

次に、共有鍵制御部２（２１０）は、耐タンパーモジュール２（２２０）に対して、鍵のマイグレート処理の完了依頼を行う（ステップＳ１１１３）。

【０１８９】

次に、耐タンパーモジュール２（２２０）は、暗号化マイグレート鍵を、ルート鍵２（２２１）の秘密鍵で復号する（ステップＳ１１１４）。図４の例では、Ｋ２０の秘密鍵で、暗号化Ｋ３１を復号する。

【０１９０】

次に、耐タンパーモジュール２（２２０）は、平文になったマイグレート鍵の鍵属性情

報のステークホルダーフィールド５０５のステークホルダー識別子５０６と鍵利用制限情報５０７とを、ステークホルダー２における鍵利用制限情報に設定する（ステップＳ１１１５）。

【０１９１】

また、Ｓ１１１５にて、リンク情報５０８が、図６（ａ）の親鍵へのリンク情報であれば、ステークホルダーフィールド５０５のリンク情報５０８を、マイグレート先の親鍵に設定し、リンク情報５０８が、図６（ｂ）の子供鍵へのリンク情報であれば変更しない。

【０１９２】

図７の例では、Ｋ３１及びＫ３２の鍵属性情報は、ステークホルダーフィールド５０５のステークホルダー識別子５０６と鍵利用制限情報５０７とリンク情報５０８とが異なり、他は同じ値として設定される。このように設定することで、Ｋ３１がステークホルダー１の環境からのみ利用可能であり、ステークホルダー２の環境からのみ利用可能となる。

【０１９３】

次に、耐タンパーモジュール２（２２０）は、ステップＳ１１１５で設定された鍵を、共有鍵群（３３０）の指定の位置に設定する（Ｓ１１１６）。図４の例では、Ｋ３２を、Ｋ２０の子供として設定している。そのため、Ｋ３２の秘密鍵は、Ｋ２０の公開鍵で暗号化される。

【０１９４】

次に、耐タンパーモジュール２（２２０）から共有鍵制御部２（２１０）を経由し、共有鍵制御部１（１１０）に対して、ステークホルダー２の鍵利用制限情報が送信される（ステップＳ１１１７）。

【０１９５】

次に、共有鍵制御部２（１１０）は、共有鍵として設定される鍵(マイグレート鍵の子供の鍵）の鍵属性情報に、ステークホルダー２の鍵利用制限情報を設定する（ステップＳ１１１８）。

【０１９６】

図７の例では、Ｋ３３及びＫ３４の鍵利用制限情報のステークホルダーフィールド５０５にステークホルダー２の情報が追加される。この処理により、ステークホルダーフィールド５０５には、「ＳＨ１」と「ＳＨ２」との２つのステークホルダーフィールドが設定され、それぞれの鍵利用制限情報としてステークホルダー１の鍵利用制限情報（ＰＣＲ）とステークホルダー２の鍵利用制限情報（ＰＣＲ）とが設定される。このように設定することで、Ｋ３３及びＫ３４は、ステークホルダー１の環境とステークホルダー２の環境とから共用利用が可能な鍵となる。

【０１９７】

次に、共有鍵制御部１（１１０）は、鍵管理テーブル１（１１３）内にあるＳ１１１８で設定した鍵ＩＤと鍵属性情報アドレスとを、共有鍵制御部２（２１０）に送信する（ステップＳ１１１９）。図４の例では、Ｋ３３とＫ３４の鍵ＩＤと鍵属性情報を送る。

【０１９８】

次に、共有鍵制御部２（２１０）は、Ｓ１１１９で受信した鍵ＩＤと鍵属性情報アドレスとを、鍵管理テーブル２（２１２）に登録する。

【０１９９】

以上で、共有鍵Ｋ３３及びＫ３４が、共有鍵制御部１（１１０）及び共有鍵制御部２（２１０）から利用可能な状態となる。

【０２００】

以上で、図１１のフローの説明を終了する。

【０２０１】

以上説明したように、本実施の態様によると、共有鍵制御部２（２１０）から共有鍵制御部１（１１０）に、鍵群１（１３０）に含まれる鍵を共有したい旨の通知を受けると、共有鍵制御部２（２１０）に対応するステークホルダー２が共有鍵制御部１（１１０）に対応するステークホルダー１に依存する関係である場合、鍵群１（１３０）に含まれる鍵

の中から鍵群２（２３０）にマイグレート可能な所定の鍵Ｋ３１を鍵群２（２３０）の中にマイグレートする。

【０２０２】

即ち、ステークホルダー２がステークホルダー１に依存する関係であることを条件に、所定の鍵Ｋ３１を親鍵とするツリー構造に含まれる鍵群の全体をコピーするのではなく、所定の鍵Ｋ３１のみをコピーし、耐タンパーモジュール１（２１０）及び耐タンパーモジュール２（２２０）で所定の鍵Ｋ３１を親鍵とするツリー構造に含まれる鍵群を共有鍵群３３０とすることで、所定の鍵Ｋ３１を親鍵とするツリー構造に含まれる鍵群の全体を二重持ちする非効率を回避できる。

【０２０３】

また、共有鍵制御部２（２１０）側で、鍵群２（２３０）に含まれる鍵を用いて所定の鍵Ｋ３１を暗号化して鍵群２（２３０）の中に保持し、鍵群１（１３０）に含まれる所定の鍵Ｋ３１より下層の鍵、例えば、Ｋ３３、Ｋ３４を共用することにより、所定の鍵Ｋ３１をマイグレートするだけで、共有鍵制御部２（２１０）側では、共有鍵制御部１（１１０）に対応する耐タンパーモジュール１（２１０）が管理する鍵群１（１３０）に含まれる所定の鍵Ｋ３１より下層の鍵、例えば、Ｋ３３、Ｋ３４を共用できる。この結果、共有鍵制御部２（２１０）は、共有鍵群３３０あるいは暗号化データ格納部４０で保持された暗号化共有データ（３４０）を簡易な構成で利用できる。

【０２０４】

さらに、共有鍵制御部２（２１０）は、ステークホルター２がステークホルダー１に依存する関係にある場合にのみ、暗号化データ格納部（４０）で保持された暗号化共有データ（３４０）を利用できる。この結果、前記所定のデータを管理する鍵構成を耐タンパーモジュール１及び耐タンパーモジュール２（２２０）で簡易にしつつ、共有鍵群３３０あるいは暗号化共有データ３４０の機密性を保証できる。

【０２０５】

また、共有鍵制御部１（１１０）は、共有鍵制御部２（２１０）に対応するステークホルダー２が、少なくともものステークホルダー１に対応する耐タンパーモジュール１（１２０）を利用するステークホルダーモデルであるとステークホルダー証明書（１５０、２５０）に基づいて判断した場合に、ステークホルダー２がステークホルダー１に依存する関係であると判断する。これにより、ステークホルダー２のステークホルダー１に対する依存関係を確実に判断できるので、前記所定のデータを管理する鍵構成を耐タンパーモジュール１（１２０）及び耐タンパーモジュール２（２２０）で簡易にしつつ、不正なステークホルダーからの共有鍵群３３０あるいは、暗号化共有データ３４０へのアクセスを確実に禁止できる。

【０２０６】

また、鍵のマイグレート処理をする際に、鍵群１（１３０）からマイグレート可能か否かを示す属性情報を参照して、鍵群１（１３０）に含まれる鍵の中から鍵群２（２３０）にマイグレート可能な所定の鍵、例えばＫ３１を探すことにより、鍵属性情報を参照するだけでマイグレート可能な鍵を探せるので、マイグレート可能な鍵を簡易にサーチできる。

【０２０７】

また、共有鍵制御部１（１１０）は、鍵群１（１３０）に含まれる鍵の中から鍵群２（２３０）にマイグレート可能な鍵が存在しない場合、マイグレート可能な鍵を生成して、この生成した鍵を鍵群２（２３０）の中にマイグレートする。この結果、鍵群１（１３０）に含まれる鍵の中から鍵群２（２３０）にマイグレート可能な所定の鍵が存在しない場合であっても、共有鍵制御部２（２１０）は鍵群１（１３０）に含まれる鍵を共用できるので、共有鍵制御部１（１１０）は鍵群１（１３０）に含まれる暗号鍵を用いて暗号化された暗号化データ１（１４０）を暗号化共有データ３４０として暗号化データ格納部で保持し、暗号化共有データ３４０にアクセスできる。

【０２０８】

また、鍵群１（１３０）に含まれる所定の鍵より下層の鍵の位置を示す位置情報を所定の鍵のリンク情報として生成してマイグレートすることにより、共有鍵制御部２（２１０）では、所定の鍵のリンク情報５０８を参照すれば、前記所定の鍵より下層の鍵の位置を確認できるので、前記所定の鍵より下層の鍵を前記耐タンパーモジュール２（２２０）の管理する鍵としてコピーすることなく、耐タンパーモジュール１（１２０）との間で所定の鍵より下層の鍵を共用できる。その結果、耐タンパーモジュール１（１２０）及び耐タンパーモジュール２（２２０）とで前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体を二重持ちする非効率を回避できる。例えば、Ｋ３３とＫ３４を共有鍵とする際に、Ｋ３１のリンク情報５０８を含んだ情報をマイグレートすることで可能となる。例えば、図６（ｂ）のリンク情報５０８を利用することで実現可能である。

【０２０９】

また、鍵群１（１３０）に含まれる所定の鍵の位置情報及び鍵群２（２３０）に含まれる所定の鍵の位置情報を、鍵群１（１３０）に含まれる前記所定の鍵より下層の鍵のリンク情報５０８として生成することにより、その下層の鍵のリンク情報５０８を参照すればその下層の鍵を暗号化した親鍵の所在を認識できるので、前記耐タンパーモジュール１（１２０）及び耐タンパーモジュール２（２２０）とで鍵群１（１３０）に含まれる所定の鍵より下層の鍵を共用する場合であっても、鍵群１（１３０）に含まれる所定の鍵より下層の鍵がどの鍵で暗号化されているかを容易に識別できる。例えば、図６（ａ）のリンク情報５０８を利用することで実現可能である。

【０２１０】

＜図１３：共有鍵利用フロー＞

図１３は、図１２で設定した共有鍵を利用する際のフローであり、第２のステークホルダープログラム（２００）が、共有鍵制御部２（２１０）を介して、共有鍵Ｋ３３あるいはＫ３４を利用する場合のフローである。

【０２１１】

まず、第２のステークホルダープログラム２００は、共有鍵制御部２（２１０）に対して共有鍵による暗号化要求を、共有鍵のＩＤと共に送信する（ステップＳ１３００）。

【０２１２】

次に、共有鍵制御部２（２１０）は、Ｓ１３００で受信したＩＤをもとに、鍵管理テーブル２（２１３）から、共有鍵を選択する（ステップＳ１３０１）。

【０２１３】

次に、共有鍵制御部２は、耐タンパーモジュール２（２２０）に、暗号化対象データと、共有鍵の鍵属性情報とを送信する（ステップＳ１３０２）。

【０２１４】

次に、耐タンパーモジュール２（２２０）は、鍵属性情報内の利用制限情報であるＰＣＲの情報と、ＰＣＲに記録されている実際の値とを比較し、両者が等しいかどうかチェックする（ステップＳ１３０３）。

【０２１５】

もし、チェックの結果、両者の値が等しいと判断されれば、Ｓ１３０４へ処理を移す。そうでなく、両者の値が等しくないと判断されれば、共有鍵制御部２（２１０）にエラーを返す。エラーを受信した共有鍵制御部２（２１０）は、図示しないエラー処理へ処理を移す。

【０２１６】

次に、耐タンパーモジュール２（２２０）は、共有データを共有鍵で暗号化し、暗号化データを共有鍵制御部２（２１０）へ返す（ステップＳ１３０４、Ｓ１３０５）。

【０２１７】

次に、共有鍵制御部２（２１０）は、共有鍵により暗号化された暗号化データを、暗号化データ格納部に書き込む（ステップＳ１３０６）。

【０２１８】

最後に、書き込み完了したことを、共有鍵制御部２（２１０）を通して第２のステーク

ホルダープログラム２００に通知する。

【０２１９】

以上で、図１３のフローの説明を終了する。

【０２２０】

なお、実施の形態１では鍵利用制限情報をＰＣＲの情報としているが、鍵利用制限情報は、これに限定されない。例えば、ＴＰＭが有するセキュアなカウンターの値や、生体認証との照合情報などの、各種認証情報であってもよい。また、それら認証情報の組合せたもので鍵の利用を制限させてもよい。

【０２２１】

例えば、カウンターの値とＰＣＲとを組み合わせて、両者の一致した場合のみ共有鍵を利用できるようにしてもよい。また、鍵利用制限情報を、利用許可フラグとし、鍵利用制限情報が「１」だったら利用可とし、鍵利用制限情報が「０」だったら利用不可とするようにしてもよい。

【０２２２】

＜図１４：共有鍵の無効化の概要フロー＞

図１４は、共有鍵の無効化の概要フローである。ステークホルダー１及びステークホルダー２間で共有されている鍵を、どちらかのステークホルダー環境がリボーク対象であるか、もしくは、ステークホルダー環境に改竄があると判定された場合、リボーク対象もしくは改竄されたステークホルダー側から、共有鍵を利用させないようにするためのフローである。

【０２２３】

まず、共有鍵制御部が端末内のステークホルダー環境がリボーク対象、もしくは、端末内のステークホルダー環境が改竄されたことを判定する（ステップＳ１４０１）。リボーク対象の判断は、リボケーションリストにリボーク対象のＩＤが列挙されているので、そのＩＤを元に判断可能である。また改竄チェックについても、ステークホルダープログラムのハッシュ値を含んだ証明書を情報処理端末に持たせ、静的もしくは動的にプログラムに改竄チェックを行い、改竄があったかどうかを判断することが可能となる。

【０２２４】

次に、共有鍵制御部は、リボーク対象もしくは改竄されたと判定されたステークホルダーが、他のステークホルダーと共有している共有鍵が、存在するかどうかを確認する（ステップＳ１４０２）。もし、存在するなら、ステップＳ１４０３へ処理を移す。そうでなく、存在しないのなら、無効化する共有鍵が存在しないので、無効化処理は終了となる。

【０２２５】

次に、共有鍵制御部が、リボーク対象もしくは改竄されたと判定されたステークホルダー環境から、共有鍵を利用できないように無効化設定する（ステップＳ１４０３）。以降、無効化処理について、例を用いて詳細に説明する

＜図１５：共有鍵の無効化の詳細フロー＞

図１５は、共有鍵の無効化の詳細フローであり、図１４のＳ１４０３を詳細化したフローである。

【０２２６】

まず、リボーク対象もしくは改竄されたと判定されたステークホルダーが他のステークホルダーと共有している共有鍵を判断する（ステップＳ１５０１）、この判断処理は、鍵属性情報のステークホルダーフィールド５０５を参照することで可能となる。

【０２２７】

次に、Ｓ１５０１により共有鍵と判断された鍵の親鍵にアクセスし、その鍵で暗号化共有鍵を復号する（ステップＳ１５０２）。

【０２２８】

次に、共有鍵制御部は、乱数生成機能を利用し、親鍵の鍵属性情報の鍵値を乱数で更新する（ステップＳ１５０３）。なお、鍵値を乱数で更新するとしているが、これに限定されない。当初の鍵値と異なる値で上書きすれば乱数以外の値でもよい。

【０２２９】
次に、更新した親鍵で、共有鍵を暗号化する（ステップＳ１５０４）。
【０２３０】
図１６は、図１５の無効化フローの実施前（無効化前）と実施後（無効化後）の状態を示した例である。
【０２３１】
図１６（ａ）は、無効化前を示したものである。この例では、共有鍵Ｋ３３とＫ３４とが、ステークホルダー１及びステークホルダー２間で共有されている。
【０２３２】
図１６（ｂ）は、ステークホルダー２の環境がリボーク対象もしくは改竄ありと判断された場合に、ステークホルダー２の環境から、Ｋ３３及びＫ３４を利用できないように無効化した後の例である。
【０２３３】
Ｋ３１の鍵値を、乱数で書換えて、書換え後の鍵をＫ３５としている。このようにすると、共有鍵Ｋ３３及びＫ３４は、Ｋ３５で暗号化され、Ｋ３５はＫ１０で暗号化されることになる。そのため、ステークホルダー２側からは、Ｋ３５を復号することができない。その結果、Ｋ３３及びＫ３４も復号することもできない。したがって、Ｋ３３及びＫ３４で暗号化されたデータを、不正なステークホルダー環境による不正利用から保護することが可能となる。
【０２３４】
以上で実施の形態１の説明を終わる。
【０２３５】
以上のように、第２のステークホルダープログラム環境（第２のステークホルダープログラム２００、共有鍵制御部２（２１０）、耐タンパーモジュール２（２２０））が外部から攻撃されたことを検知した場合、耐タンパーモジュール１（１２０）は所定の鍵と置換える代替鍵を生成して前記所定の鍵を親鍵とするツリー構造に含まれる鍵を前記代替鍵で再暗号化すると共に前記所定の鍵の親鍵を用いて前記代替鍵を暗号化する。この結果、共有鍵制御部２（２１０）は前記所定の鍵を用いて前記代替鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、前記代替鍵を親鍵とするツリー構造に含まれる鍵で暗号化された所定のデータを利用できず、前記所定データを不正な利用から保護できる。例えば、この場合、所定の鍵がＫ３１であり、代替鍵がＫ３５となる。
【０２３６】
なお、実施の形態１では、共有鍵制御部１（１１０）と共有鍵制御部２（２１０）とは、それぞれ別の制御部として構成しているが、共有鍵制御部１（１１０）と共有鍵制御部２（２１０）と両機能を１つの制御部として実現してもよい。
【０２３７】
これにより、共有鍵制御部１（１１０）と共有鍵制御部２（２１０）とは、共用の共有鍵制御部で構成が可能となり、１つの共有鍵制御部で２つのステークホルダー間の共有鍵を統括的に制御することが可能なので、より柔軟にアクセス制御を行うことが可能となる。
【０２３８】
なお、実施の形態１では、共有鍵の設定と、共有鍵設定後の不正なステークホルダーによる共有鍵の不正利用を防ぐための共有鍵の無効化について説明しているが、無効化後の鍵に対して、再度共有鍵として設定するようにしてもよい。
【０２３９】
例えば、不正ステークホルダーが、ステークホルダー管理サーバーを利用して更新処理したことにより、正規のステークホルダー環境になった場合、無効化されていた共有鍵を、再度共有鍵として設定してもよい。
【０２４０】
より具体的には、ステークホルダー１とステークホルダー２との間で共用できる共有鍵

があり、ステークホルダー２が不正と判断されると、ステークホルダー２からその共有鍵が利用できないように無効化される。

【０２４１】

その後、不正なステークホルダー２の環境が、第２のステークホルダー管理サーバーから更新モジュールをダウンロードし、不正なステークホルダー２の環境を更新し、更新の結果、正規ステークホルダー環境となれば、無効化されていた共有鍵をステークホルダー２から再度利用できるように復活させる。

【０２４２】

再度共有鍵として設定する処理は、実施の形態１で説明した共有鍵の設定処理を、更新されたステークホルダー２に対して行うことで実現できるので説明は省略する。

【０２４３】

（実施の形態２）

実施の形態２では、実施の形態１とは異なる方法で、共有鍵の無効化をする例である。実施の形態１と同じ構成やフローである部分について説明を省略し、実施の形態２に特有の処理について図面を用いて説明する。

【０２４４】

＜図１７：無効化詳細フロー＞

図１７は、共有鍵の無効化の詳細フローであり、図１４のＳ１４０３を詳細化したフローである。また、図１８は、図１７の無効化フローの実施前（無効化前）と実施後（無効化後）とを示したものでる。以降、図１７及び図１８を用いて説明する。

【０２４５】

ステップＳ１７０１とステップＳ１７０２とは、実施の形態１のＳ１５０１とＳ１５０２と同じであるので説明を省略する。

【０２４６】

次に、Ｓ１７０２における共有鍵群（３３０）以外から、共有鍵の親鍵とは異なるマイグレート可能で共有許可である鍵をツリー内から選択する。もし、そのような鍵が存在しなければ、新たに鍵を生成する。（ステップＳ１７０３）。

【０２４７】

後述するように、本実施の形態では、選択した鍵を共有鍵の新たな親鍵とすることで、ステークホルダー２から共有鍵を利用できないようにする。この目的を達成するためのみならば、マイグレートの可否や共有許可の有無とは関係なく共有鍵の親鍵と異なる鍵を選べばよい。しかし、ここでは、さらにマイグレートが可能であり、共有許可である鍵を選択している。

【０２４８】

なぜ、このような鍵を選択するかについて、次の２つのケースを用いて説明する。

【０２４９】

まず１つ目のケースは、不正なステークホルダー環境と判断されたことにより共有鍵が無効化された不正なステークホルダーが、更新処理により、正規のステークホルダー環境になった場合、再度共有鍵を利用できるように復活させるケースである。再度共有鍵を利用できるようにするには、共有鍵の親鍵をマイグレートする必要があるため、その場合に備えてマイグレートが可であり共有許可のある鍵を新たな親鍵としておく必要がある。

【０２５０】

２つ目のケースは、３つ以上のステークホルダーで共有されている共有鍵があるケースである。例えば、ステークホルダー１，２，３で共有されている共有鍵に対し、ステークホルダー２に対してのみ無効化する場合、ステークホルダー１とステークホルダー３との間では、共有鍵を共有したままの設定にする必要がある。共有鍵の親鍵を変更すると、ステークホルダー３も共有鍵を利用することができなくなってしまう。そのため、共有鍵がステークホルダー３と共有されたままの状態を維持するために、共有鍵の親鍵の変更後、ステークホルダー１から３へ、共有鍵の親鍵をマイグレートする処理が必要となる。ステークホルダー１から３へ共有鍵の親鍵がマイグレートされれば、両ステークホルダーで共

有鍵の利用が可能となる。

【０２５１】

次に、Ｓ１７０３で選択された鍵を、共有鍵の親鍵として設定するので、Ｓ１７０３で選択された鍵と共有鍵とが親子関係になるように鍵属性情報のリンク情報５０８を更新する（ステップＳ１７０４）。

【０２５２】

次に、Ｓ１７０３で選択された鍵で、共有鍵を暗号化する（ステップＳ１７０５）。

【０２５３】

最後に、鍵管理テーブルを更新する（ステップＳ１７０６）。

【０２５４】

図１８は、図１７の無効化フローの実施前（無効化前）と実施後（無効化後）との状態を示した例である。

【０２５５】

図１８（ａ）は、無効化前を示したものである。この例では、共有鍵Ｋ３３とＫ３４とがステークホルダー１及びステークホルダー２間で共有されている。

【０２５６】

図１８（ｂ）は、ステークホルダー２の環境がリボーク対象もしくは改竄ありと判断された場合に、ステークホルダー２の環境からはＫ３３とＫ３４とを利用できないように無効化した後の例である。

【０２５７】

Ｋ３３及びＫ３４の親鍵を、Ｋ３１ではないマイグレート可能かつ共有可能なＫ１１としている。このようにすると、Ｋ３３とＫ３４とがＫ１１で暗号化され、Ｋ１１はＫ１０で暗号化されることになる。そのため、ステークホルダー２側からは、Ｋ１１を復号することができず、その結果、Ｋ３３とＫ３４とも復号することもできない。したがって、Ｋ３３とＫ３４とで暗号化されたデータを、不正なステークホルダー環境による不正利用から保護することが可能となる。

【０２５８】

以上で実施の形態２の説明を終わる。

【０２５９】

以上のように、第２のステークホルダープログラム環境（第２のステークホルダープログラム２００、共有鍵制御部２（２１０）、耐タンパーモジュール２（２２０））が外部から攻撃されたことを検知した場合、耐タンパーモジュール１（１２０）は前記所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵を用いて所定のデータを暗号化し直す。この結果、共有鍵制御部２（２１０）は所定の鍵を親鍵とするツリー構造に含まれる鍵を用いては前記所定のデータを復号化できないので、所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵で暗号化された所定のデータを利用できず、所定データを不正な利用から保護できる。例えば、この場合、所定の鍵が、Ｋ３１であり、所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵がＫ１１として実現可能である。

【０２６０】

なお、実施の形態２では、実施の形態１と同様、不正なステークホルダー環境が、更新などにより正当なステークホルダー環境になった場合に、共有鍵を再度共有させることも可能である。再度共有鍵として設定する処理は、実施の形態２で説明した共有鍵の設定処理を、更新されたステークホルダー２に対して行うことで実現できるので説明は省略する。

【０２６１】

（実施の形態３）

本実施の形態３では、実施の形態１と実施の形態２とは異なる方法で、共有鍵の無効化をする例である。実施の形態１と実施の形態２と同じ構成やフローである部分について説明を省略し、実施の形態３に特有の処理について図面を用いて説明する。

【０２６２】

＜図１９：無効化詳細フロー＞

図１９は、共有鍵の無効化の詳細フローであり、図１４のＳ１４０３を詳細化したフローである。

【０２６３】

ステップＳ１９０１とステップＳ１９０２とは、Ｓ１５０１とＳ１５０２と同じであるので説明を省略する。

【０２６４】

次に、Ｓ１９０１によりリボーク対象もしくは改竄されたと判定されたステークホルダーが他のステークホルダーと共有していると判断された共有鍵にアクセスし、リボーク対象もしくは改竄されたステークホルダーに対応する鍵利用制限情報にアクセスする（ステップＳ１９０２）。

【０２６５】

次に、共有鍵制御部は、乱数生成機能を利用し、Ｓ１９０２で選択された共有鍵に対応する鍵属性情報の鍵利用制限情報を、乱数で更新する（ステップＳ１５０３）。なお、鍵利用制限情報を乱数で更新するとしているが、これに限定されない。

【０２６６】

本実施の形態によると、第１ステークホルダーが管理する第１のステークホルダー環境は、第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、共有鍵制御部１（１１０）は共有鍵で暗号化されたデータを、共有鍵制御部２（２１０）から利用できないように、鍵属性情報の鍵利用制限情報５０８を書き換える。この結果、共有鍵制御部２（２２０）は共有鍵の鍵ロード処理が出来ないので、共有鍵で暗号化されたデータを不正な利用から保護できる。

【０２６７】

なお、共有鍵の鍵利用制限情報だけを更新するとしているが、共有鍵の親鍵の鍵利用制限情報を更新してもよい。これは、共有鍵が複数に階層に渡って構成されている場合、共有鍵からルート方向に辿り共有鍵群（３３０）内の最上位層の鍵だけを無効化することで、それより子供の鍵は無効化対象とすることが可能となるからである。

【０２６８】

以上のように、共有鍵制御部２（２１０）は、改竄のない第２のステークホルダー環境から生成された鍵利用制限情報５０７と第２のステークホルダープログラム（２００）から実際に得られたＰＣＲ（２２３）に記録されている環境情報とを比較し、比較結果が正しい場合にのみ前記鍵を利用させる。

【０２６９】

この結果、第２のステークホルダープログラム（２００）が改竄され若しくはリボークされた場合には前記比較結果は不一致となって、共有鍵制御部２（２１０）は所定の鍵を用いて、所定の鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、前記所定の鍵を親鍵とするツリー構造に含まれる鍵用いて暗号化されている所定のデータを復号化できず、所定データを不正な利用から保護できる。

【０２７０】

例えば、第２のステークホルダープログラム（２００）が改竄されていれば、共有鍵制御部２（２１０）からＫ３３のロードができないため、Ｋ３３で暗号化されたデータＤ３４の不正な利用から保護できる。

【０２７１】

また、第１ステークホルダーが管理する第１のステークホルダー環境は、第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、共有鍵制御部１（１１０）は、所定の鍵を親鍵とするツリー構造に含まれる鍵を、共有鍵制御部２（２１０）から利用できないように、鍵利用制限情報を書き換える。

【０２７２】

この結果、共有鍵制御部２（２１０）は所定の鍵を用いては所定の鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、所定の鍵を親鍵とするツリー構造に含まれる

鍵用いて暗号化されている前記所定のデータを復号化できず、前記所定データを不正な利用から保護できる。この場合、例えば、所定の鍵がＫ３２で、所定の鍵を親鍵とするツリー構造に含まれる鍵がＫ３３，Ｋ３４であると、Ｋ３２の利用制限情報を書き換える、もしくはＫ３３，Ｋ３４の鍵利用制限情報を書き換えることで実現できる。

【０２７３】

なお、実施の形態１及び実施の形態２と同様、不正なステークホルダー環境が、更新などにより正当なステークホルダー環境になった場合に、共有鍵を再度共有させることも可能である。再度共有させる処理は、更新されたステークホルダーについての鍵利用制限情報を、更新後のステークホルダーの鍵利用制限情報（ＰＣＲ値）に更新すればよい。

【０２７４】

＜図２３：再共有化＞

図２３は、実施の形態３における不正なステークホルダー環境が、更新などにより正当なステークホルダー環境になった場合に、共有鍵を再度共有させるフローを示している。ここでは、ステークホルダー２の環境が、更新などにより正当なステークホルダー環境になり、共有が無効化されていたステークホルダー１の環境との共有鍵を、再度共有化する例で説明する。

【０２７５】

まず、ステークホルダー１の環境は、不正と判断されていたステークホルダー２の環境が正当な環境に更新されたと判断する（ステップＳ２３０１）。この判断は、例えば、周期的に行われたり、電源が投入されたときや、ステークホルダー２の環境の更新が完了したときに行われる。

【０２７６】

次に、ステークホルダー１の環境の、共有鍵制御部１（１１０）は、共有鍵群（３３０）内の鍵属性情報を参照し、ステークホルダーフィールド５０５のステークホルダー識別子５０６にステークホルダー２の識別子が登録されている鍵を選択する（ステップＳ２３０２）。

【０２７７】

最後に、Ｓ２３０３で選択した鍵の鍵属性情報内の鍵利用制限情報を正当になったステークホルダーの環境情報に更新する（ステップＳ２３０３）。この例では、鍵属性情報内のステークホルダー２の識別子に対応する鍵利用制限情報が、更新などにより正当になった後のステークホルダー２の環境情報に更新される。

【０２７８】

もし、ステークホルダー２と共有すべき鍵が、複数存在するのであれば、Ｓ２３０２とＳ２３０３とを繰り返し処理すればよい。

【０２７９】

このようにすることで、ステークホルダー２の環境は、共有が無効化されていたステークホルダー１と共有する共有鍵を、再度共有鍵として利用可能となる。

【０２８０】

（実施の形態４）

本実施の形態４では、実施の形態１、実施の形態２、及び実施の形態３とは異なる方法で、暗号化共有データを保護するする例である。他の実施の形態と同じ構成やフローである部分について説明を省略し、実施の形態３に特有の処理について図面を用いて説明する。

【０２８１】

＜図２０：暗号化共有データの構造＞

図２０は、暗号化共有データの構造を示した図である。暗号化共有データは、暗号化データサイズ２００１と、暗号化データ２００２と、ステークホルダーフィールド２００３とを備える。

【０２８２】

ステークホルダーフィールド２００３には、共有データにアクセス権のあるステークホ

ルダーの情報が列挙される。図４に示した暗号化共有データＤ３３及びＤ３４は、ステークホルダー１とステークホルダー２とからアクセスされる例であるので、ステークホルダーフィールド５０５は、２つ存在することになる。

【０２８３】

ステークホルダーフィールド２００３は、ステークホルダー識別子２００４と、鍵の利用制限を示す利用制限情報２００５とを含む。

【０２８４】

利用制限情報２００５は、期待されるＰＣＲ値であり、耐タンパーモジュールの備えるＰＣＲ（１２３、２２３）に記録されている実際の値と比較され、実際のＰＣＲ値と期待されるＰＣＲ値が等しかった場合にのみ耐タンパーモジュールから復号結果が得られるように制限するための情報である。

【０２８５】

＜図２１：暗号化共有データの復号フロー＞

図２１は、共有鍵制御部２（２１０）の暗号化共有データ復号処理のフローを示した図である。

【０２８６】

まず、共有鍵制御部２（２１０）は、第２のステークホルダープログラム（２００）から暗号化共有データの復号要求を受け、暗号化データ格納部４０から復号対象となる暗号化共有データ３４０をリードする（ステップＳ２１０１）。

【０２８７】

次に、共有鍵制御部２（２１０）は、鍵管理テーブル２（２１３）から、共有鍵を選択する（ステップＳ２１０２）。

【０２８８】

次に、共有鍵制御部２（２１０）は、Ｓ２１０２で選択した共有鍵と、Ｓ２１０１でリードした暗号化共有データとをパラメータとし、耐タンパーモジュール２（２２０）に復号処理の要求をする（ステップＳ２１０３）。

【０２８９】

次に、Ｓ２１０３で指定された共有鍵を鍵ツリーのルート鍵２からリーフ方向へと、親子関係を元に鍵を復号していき、平文の共有鍵を得て、暗号化共有データを復号する（ステップＳ２１０４）。

【０２９０】

次に、耐タンパーモジュール２（２２０）は、復号要求のあった暗号化共有データの利用制限情報として設定されている期待ＰＣＲ値と、実際のＰＣＲ（２２３）に記録されている値とを比較する（ステップＳ２１０５）。比較の結果、両者が等しければ、復号データを共有鍵制御部２（２１０）へ返す（ステップＳ２１０６）。比較の結果、両者が等しくないと判定されれば、復号データは、共有鍵制御部２（２１０）へ返されず、エラー通知のみを返す。

【０２９１】

＜図２２：無効化＞

図２２は、暗号化共有データの復号処理を無効化するための詳細フローであり、実施の形態４における図１４のＳ１４０３を詳細化したフローである。

【０２９２】

まず、リボーク対象もしくは改竄されたと判定されたステークホルダーが、他のステークホルダーと共有している共有データを判断する（ステップＳ２２０１）。この判断処理は、暗号化共有データのデータ構造のステークホルダーフィールド２００３を参照することで可能となる。

【０２９３】

次に、Ｓ２２０１で判断された暗号化共有データの利用制限情報にアクセスする（ステップＳ２２０２）。

【０２９４】

次に、共有鍵制御部は、乱数生成機能を利用し、Ｓ２２０２でアクセスされた利用制限情報を、乱数で更新する（ステップＳ１５０３）。なお、利用制限情報を乱数で更新するとしているが、これに限定されない。

【０２９５】

このようにすると、リボーク対象もしくは改竄のある不正なステークホルダー環境から暗号化共有データの復号要求があったとしても、Ｓ２１０５の比較結果は不一致となって、暗号化共有データの復号データを得ることができず、暗号化共有データの不正利用から保護できる。

【０２９６】

以上説明したように、共有鍵制御部２（２２０）は、改竄のない第２のステークホルダー環境から生成された暗号化データ利用制限情報と第２のステークホルダー環境から実際に得られた耐タンパーモジュール２（２２０）内のＰＣＲ（２２３）に記録されている環境情報とを比較し、比較結果が正しい場合にのみ暗号化データを復号させる。

【０２９７】

この結果、第２ステークホルダー環境が改竄され若しくはリボークされた場合には前記比較結果は不一致となって、共有鍵制御部２（２１０）は所定の鍵を用いて、所定の鍵を親鍵とするツリー構造に含まれる鍵を用いて暗号化されている暗号化データを復号化できず、暗号化データを不正な利用から保護できる。

【０２９８】

例えば、第２のステークホルダープログラム（２００）が改竄されていれば、耐タンパーモジュール２（２２０）から、Ｋ３３で暗号化されたＤ３３の復号データを得ようとしても、復号結果が得られないので、Ｄ３４の不正な利用から保護できる。

【０２９９】

また、前記第１ステークホルダーが管理する第１のステークホルダー環境は、第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、共有鍵制御部１（１１０）は、所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データを、共有鍵制御部２（２１０）から利用できないように、暗号化データ利用制限情報２００５を書き換える。

【０３００】

この結果、共有鍵制御部２（２１０）は所定の鍵を用いて所定の鍵を親鍵とするツリー構造に含まれる鍵を用いた復号処理ができないので、前記所定データを不正な利用から保護できる

なお、実施の形態１から実施の形態４は、組み合わせて実現されてもよい。

【０３０１】

また、実施の形態１から実施の形態４は、耐タンパーモジュール（１５０，２５０）を、ＴＰＭあるいはＭＴＭを用いて実現する形態としている。そのためＴＣＧで規定されるＴｒｕｓｔｅｄ　Ｂｏｏｔでもよいし、ＴＣＧ　Ｍｏｂｉｌｅ仕様で規定されるＳｅｃｕｒｅ　Ｂｏｏｔでもよい。また、実行されるプログラムの完全性を検証できる仕組みであればよい。

【０３０２】

これにより、前記第１共有鍵制御部はセキュアブートする際に、前記第２共有鍵制御部が外部から攻撃されたことを検知することにより、前記第２共有鍵制御部の外部からの攻撃を判断できる。

【０３０３】

また、ステークホルダー環境の改竄チェックは、ＴＣＧで規定されているＡｔｔｅｓｔａｔｉｏｎ機能を利用してもよい。Ａｔｔｅｓｔａｔｉｏｎ機能のサーバー側の判定結果を情報処理端末１０に送付し、ステークホルダーの改竄を検知してもよいし、上述したようにサーバーから配信されるリボケーションリスト１４を利用してもよい。

【０３０４】

これにより、共有鍵制御部１（１１０）は、外部のサーバーから、共有鍵制御部２（２

１０）の改竄の検知、もしくはリボーク対象である旨の通知を受けることにより、不正な共有鍵制御部２（２１０）から共有鍵や暗号化共有データの不正利用を防止することができる。

【０３０５】

なお、不正なステークホルダー環境が、更新などにより正当なステークホルダー環境になった場合に、暗号化共有データを再度共有させることも可能である。再度共有させる処理は、暗号化共有データの、更新されたステークホルダーについての鍵利用制限情報を、更新後のステークホルダーの鍵利用制限情報（ＰＣＲ値）に更新すればよい。

【０３０６】

なお、実施の形態４における共有鍵の再共有化のフローは、図２３における処理のＳ２２０２とＳ２２０３で参照する情報が、図５の鍵属性情報でなく、図２０の共有鍵暗号化データになり、更新する情報が、鍵利用制限情報５０８でなく、鍵利用制限情報２００５になるだけなので、説明を省略する。

【０３０７】

なお、実施の形態１から実施の形態４では、共有鍵の再共有化の処理を、不正と判断されていたステークホルダー以外のステークホルダー環境が、再共有化処理のトリガーとなっているが、不正なステークホルダー自身が、更新後、自ら自身が正当であることを検証してから、共有先であるステークホルダーに対して再度共有化の依頼を行うようにしてもよい。

（その他変形例）

なお、本発明を上記実施の形態に基づいて説明してきたが、本発明は、上記の実施の形態に限定されないのはもちろんである。以下のような場合も本発明に含まれる。

【０３０８】

（１）上記の各装置は、具体的には、マイクロプロセッサ、ＲＯＭ、ＲＡＭ、ハードディスクユニット、ディスプレイユニット、キーボード、マウスなどから構成されるコンピュータシステムである。前記ＲＡＭまたはハードディスクユニットには、コンピュータプログラムが記憶されている。前記マイクロプロセッサが、前記コンピュータプログラムにしたがって動作することにより、各装置は、その機能を達成する。ここでコンピュータプログラムは、所定の機能を達成するために、コンピュータに対する指令を示す命令コードが複数個組み合わされて構成されたものである。また、各装置は、マイクロプロセッサ、ＲＯＭ、ＲＡＭ、ハードディスクユニット、ディスプレイユニット、キーボード、マウスなどの全てを含むものではなく、これらの一部から構成されているとしてもよい。

【０３０９】

（２）上記の各装置を構成する構成要素の一部または全部は、１個のシステムＬＳＩ（Ｌａｒｇｅ　Ｓｃａｌｅ　Ｉｎｔｅｇｒａｔｉｏｎ：大規模集積回路）から構成されているとしてもよい。システムＬＳＩは、複数の構成部を１個のチップ上に集積して製造された超多機能ＬＳＩであり、具体的には、マイクロプロセッサ、ＲＯＭ、ＲＡＭなどを含んで構成されるコンピュータシステムである。前記ＲＡＭには、コンピュータプログラムが記憶されている。前記マイクロプロセッサが、前記コンピュータプログラムにしたがって動作することにより、システムＬＳＩは、その機能を達成する。

【０３１０】

また、上記の各装置を構成する構成要素の各部は、個別に１チップ化されていても良いし、一部又は全てを含むように１チップ化されてもよい。

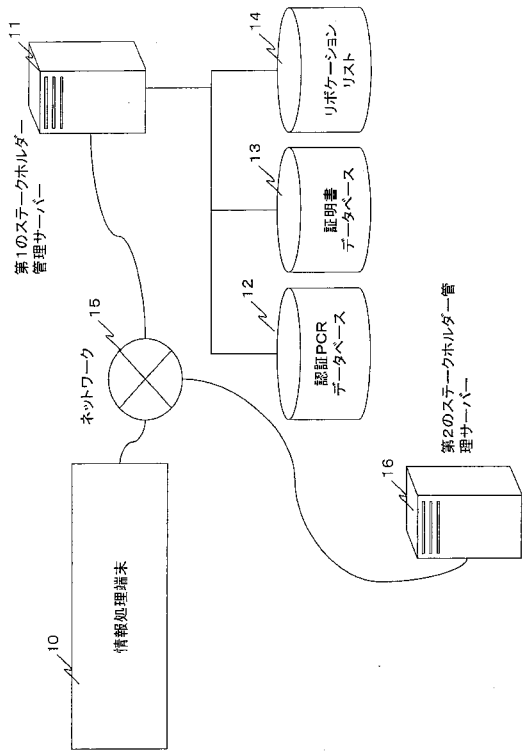
【０３１１】

また、ここでは、システムＬＳＩとしたが、集積度の違いにより、ＩＣ、ＬＳＩ、スーパーＬＳＩ、ウルトラＬＳＩと呼称されることもある。また、集積回路化の手法はＬＳＩに限るものではなく、専用回路又は汎用プロセッサで実現してもよい。ＬＳＩ製造後に、プログラムすることが可能なＦＰＧＡ（Ｆｉｅｌｄ　Ｐｒｏｇｒａｍｍａｂｌｅ　Ｇａｔｅ　Ａｒｒａｙ）や、ＬＳＩ内部の回路セルの接続や設定を再構成可能なリコンフィギュラブル・プロセッサーを利用しても良い。

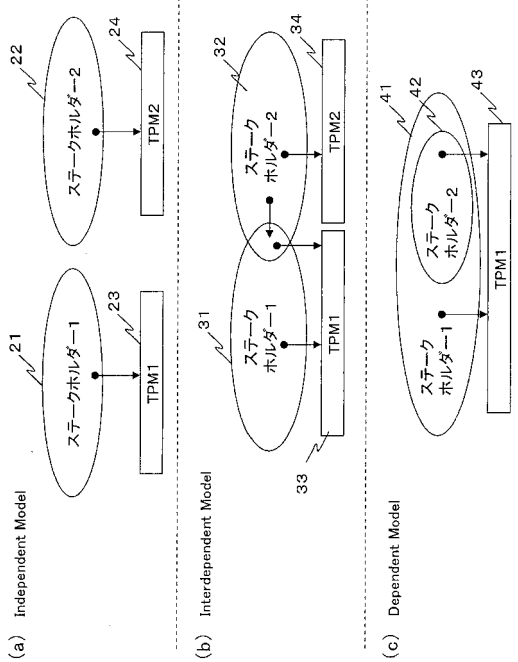
【０３１２】
さらには、半導体技術の進歩又は派生する別技術によりＬＳＩに置き換わる集積回路化の技術が登場すれば、当然、その技術を用いて機能ブロックの集積化を行ってもよい。バイオ技術の適用等が可能性としてありえる。
【０３１３】
（３）上記の各装置を構成する構成要素の一部または全部は、各装置に脱着可能なＩＣカードまたは単体のモジュールから構成されているとしてもよい。前記ＩＣカードまたは前記モジュールは、マイクロプロセッサ、ＲＯＭ、ＲＡＭなどから構成されるコンピュータシステムである。前記ＩＣカードまたは前記モジュールは、上記の超多機能ＬＳＩを含むとしてもよい。マイクロプロセッサが、コンピュータプログラムにしたがって動作することにより、前記ＩＣカードまたは前記モジュールは、その機能を達成する。このＩＣカードまたはこのモジュールは、耐タンパ性を有するとしてもよい。
【０３１４】
（４）本発明は、上記に示す方法であるとしてもよい。また、これらの方法をコンピュータにより実現するコンピュータプログラムであるとしてもよいし、前記コンピュータプログラムからなるデジタル信号であるとしてもよい。
【０３１５】
また、本発明は、前記コンピュータプログラムまたは前記デジタル信号をコンピュータ読み取り可能な記録媒体、例えば、フレキシブルディスク、ハードディスク、ＣＤ－ＲＯＭ、ＭＯ、ＤＶＤ、ＤＶＤ－ＲＯＭ、ＤＶＤ－ＲＡＭ、ＢＤ（Ｂｌｕ－ｒａｙ　Ｄｉｓｃ）、半導体メモリなどに記録したものとしてもよい。また、これらの記録媒体に記録されている前記デジタル信号であるとしてもよい。
【０３１６】
また、本発明は、前記コンピュータプログラムまたは前記デジタル信号を、電気通信回線、無線または有線通信回線、インターネットを代表とするネットワーク、データ放送等を経由して伝送するものとしてもよい。
【０３１７】
また、本発明は、マイクロプロセッサとメモリを備えたコンピュータシステムであって、前記メモリは、上記コンピュータプログラムを記憶しており、前記マイクロプロセッサは、前記コンピュータプログラムにしたがって動作するとしてもよい。
【０３１８】
また、前記プログラムまたは前記デジタル信号を前記記録媒体に記録して移送することにより、または前記プログラムまたは前記デジタル信号を前記ネットワーク等を経由して移送することにより、独立した他のコンピュータシステムにより実施するとしてもよい。
【０３１９】
（５）上記実施の形態及び上記変形例をそれぞれ組み合わせるとしてもよい。
【産業上の利用可能性】
【０３２０】
本発明は、例えばセキュアなデータを扱う情報処理装置を製造及び販売する産業において、複数のステークホルダー間において依存関係を持たせた証明書を用いて、依存関係に従った形で鍵を共有し、複数のステークホルダー間で、セキュアな共有データを効率的に暗号化する仕組みとして利用することができる。また、本発明は、不正なステークホルダーから共有鍵へのアクセスを制限させる仕組みとして利用することができる。
【符号の説明】
【０３２１】
１０　情報処理端末
１１　第１のステークホルダー管理サーバー
１２　認証ＰＣＲデータベース
１３　証明書データベース
１４　リボケーションリスト

１５　ネットワーク
１６　第２のステークホルダー管理サーバー
２１、２２、３１、３２、４１，４２　ステークホルダー
２３、２４、３３、３４、４３　ＴＰＭ
１００、２００　ステークホルダープグラム
１１０、２１０　共有鍵制御部
１１１、２１１　マルチステークホルダー判定部
１１２、２１２　共有許可設定部
１１３、２１３　鍵管理テーブル
１２０、２２０　耐タンパーモジュール
１２１、２２１　ルート鍵
３０　鍵格納部
４０　暗号化データ格納部
５０　証明書格納部
１３０　鍵群１
２３０　鍵群２
３３０　共有鍵群
１４０、２４０　暗号化データ
３４０　暗号化共有データ
１５０、２５０　ステークホルダー証明書
１２２、２２１　セキュアメモリ
４１０、４１１、４１２、４１３、４２０、４２１、４２２、４２３、４３１、４３２、４３４　鍵属性情報
１２３、２２３　ＰＣＲ
５０１　マイグレート許可フラグ
５０２　共有許可フラグ
５０３　暗号アルゴリズム・鍵サイズ
５０４　鍵値
５０５、２００３　ステークホルダーフィールド
５０６、２００４　ステークホルダー識別子
５０７、２００５　鍵利用制限情報
５０８、２００６　リンク情報
６０１、６０２、６０３、６１１、６１２、６１３　鍵ポインタ
８１１、８２１　鍵ＩＤ
８１２、８２２　鍵属性情報アドレス
９０１　証明書バージョン
９０２　シリアルナンバー
９０３　署名アルゴリズム
９０４　発行者情報
９０５　有効期間
９０６　サブジェクト
９０７　公開鍵情報
９０８　ＴＰＭバージョン
９０９　トラストモデル識別情報
９１０　依存ステークホルダー証明書識別情報
９１１　拡張領域
９１２　署名データ
２００１　暗号化データサイズ
２００２　暗号化データ

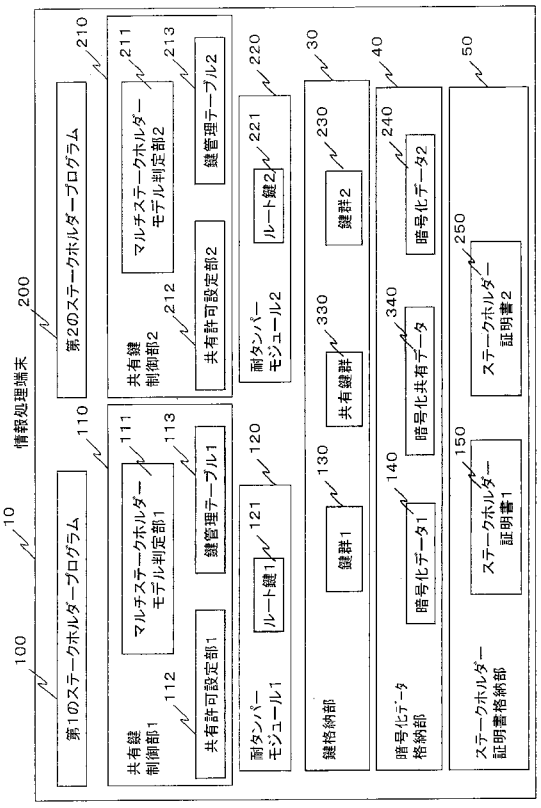
【図１】


第1のステークホルダー管理サーバー
11
ネットワーク
15
情報処理端末
10
認証PCRデータベース
12
証明書データベース
13
リボケーションリスト
14
第2のステークホルダー管理サーバー
16

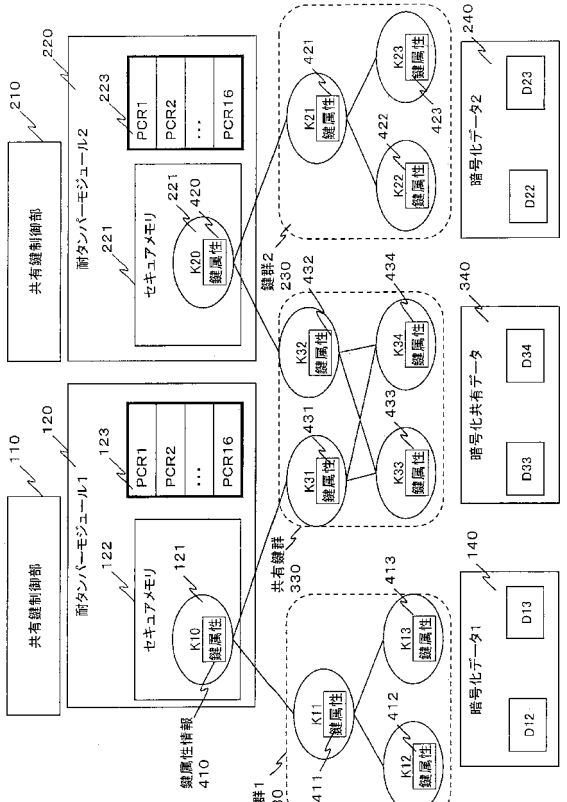
【図２】


(a) Independent Model
ステークホルダー1
21
TPM1
23
ステークホルダー2
22
TPM2
24
(b) Interdependent Model
ステークホルダー1
31
TPM1
33
ステークホルダー2
32
TPM2
34
(c) Dependent Model
ステークホルダー1
41
TPM1
43
ステークホルダー2
42

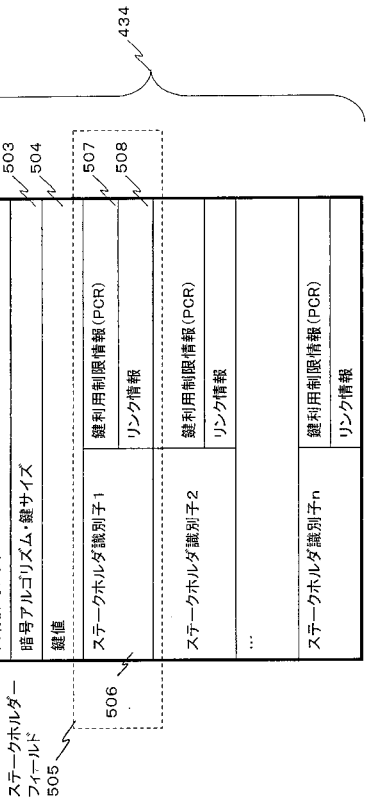
【図３】


情報処理端末
10
第1のステークホルダープログラム
100
第2のステークホルダープログラム
200
共有鍵制御部1
110
マルチステークホルダーモデル判定部1
111
共有許可設定部1
112
鍵管理テーブル1
113
耐タンパーモジュール1
120
ルート鍵1
121
鍵格納部
鍵群1
130
暗号化データ格納部
暗号化データ1
140
ステークホルダー証明書格納部
ステークホルダー証明書1
150
共有鍵制御部2
210
マルチステークホルダーモデル判定部2
211
共有許可設定部2
212
鍵管理テーブル2
213
耐タンパーモジュール2
220
ルート鍵2
221
鍵群2
230
共有鍵群
330
暗号化データ2
240
ステークホルダー証明書2
250
30
40
50

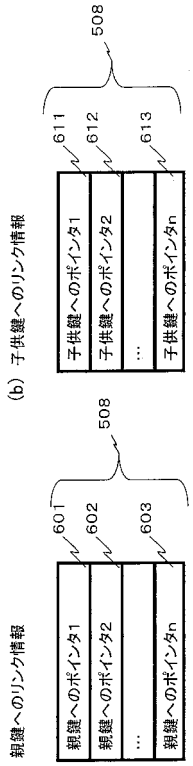
【図４】


共有鍵制御部
110
耐タンパーモジュール1
120
セキュアメモリ
121
K10
鍵属性
PCR1
PCR2
PCR16
123
鍵属性情報
410
共有鍵制御部
210
耐タンパーモジュール2
220
セキュアメモリ
221
K20
鍵属性
420
PCR1
PCR2
PCR16
223
鍵群1
130
共有鍵群
330
K11
K12
K13
鍵属性
411
412
413
暗号化データ1
140
D12
D13
鍵群2
230
K21
K22
K23
421
422
423
暗号化データ2
240
D22
D23
K31
K32
K33
K34
431
432
433
434
暗号化共有データ
340
D33
D34

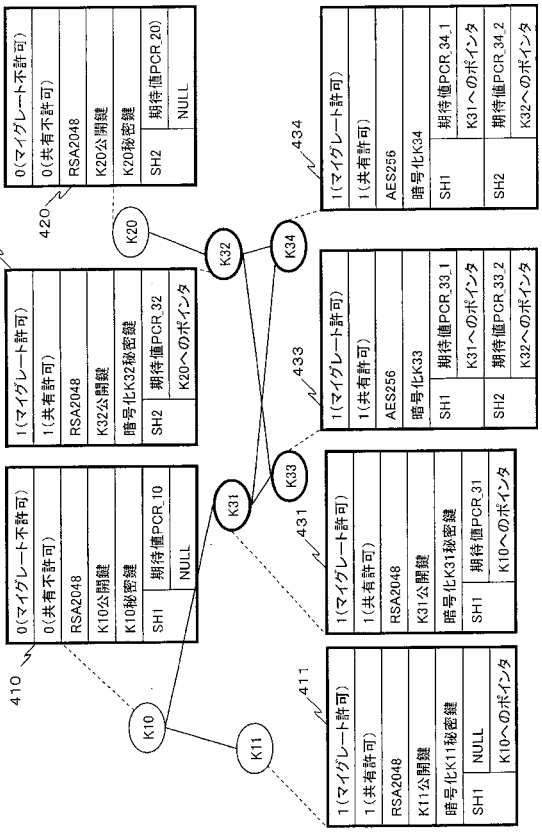
【図５】


鍵属性情報
マイグレート許可フラグ
共有許可フラグ
暗号アルゴリズム・鍵サイズ
鍵値
ステークホルダ識別子1
鍵利用制限情報(PCR)
リンク情報
ステークホルダ識別子2
鍵利用制限情報(PCR)
リンク情報
…
ステークホルダ識別子n
鍵利用制限情報(PCR)
リンク情報
501
502
503
504
507
508
434
506
ステークホルダーフィールド
505

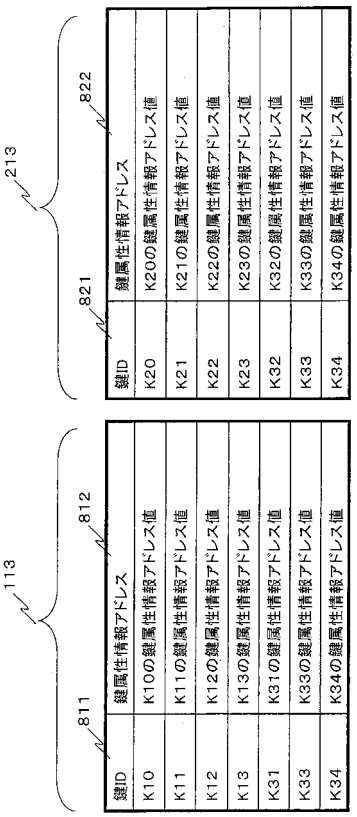
【図６】


(a) 親鍵へのリンク情報
親鍵へのポインタ1
親鍵へのポインタ2
…
親鍵へのポインタn
601
602
603
508
(b) 子供鍵へのリンク情報
子供鍵へのポインタ1
子供鍵へのポインタ2
…
子供鍵へのポインタn
611
612
613
508

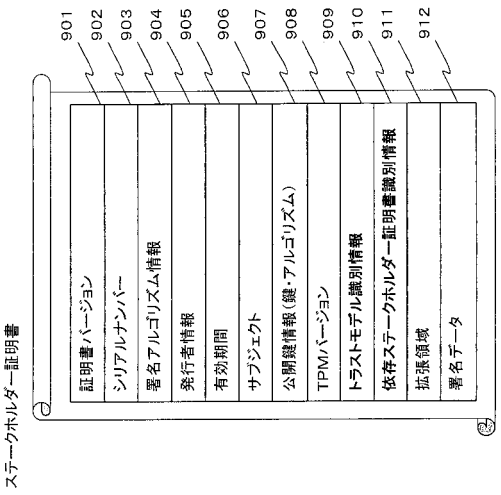
【図７】


410
0(マイグレート不許可)
0(共有不許可)
RSA2048
K10公開鍵
K10秘密鍵
SH1 期待値PCR_10
NULL
K10
411
1(マイグレート許可)
1(共有許可)
RSA2048
K11公開鍵
暗号化K11秘密鍵
SH1 NULL
K10へのポインタ
K11
420
0(マイグレート不許可)
0(共有不許可)
RSA2048
K20公開鍵
K20秘密鍵
SH2 期待値PCR_20
NULL
K20
432
1(マイグレート許可)
1(共有許可)
RSA2048
K32公開鍵
暗号化K32秘密鍵
SH2 期待値PCR_32
K20へのポインタ
K32
431
1(マイグレート許可)
1(共有許可)
RSA2048
K31公開鍵
暗号化K31秘密鍵
SH1 期待値PCR_31
K10へのポインタ
K31
433
1(マイグレート許可)
1(共有許可)
AES256
暗号化K33
SH1 期待値PCR_33_1
K31へのポインタ
SH2 期待値PCR_33_2
K32へのポインタ
K33
434
1(マイグレート許可)
1(共有許可)
AES256
暗号化K34
SH1 期待値PCR_34_1
K31へのポインタ
SH2 期待値PCR_34_2
K32へのポインタ
K34

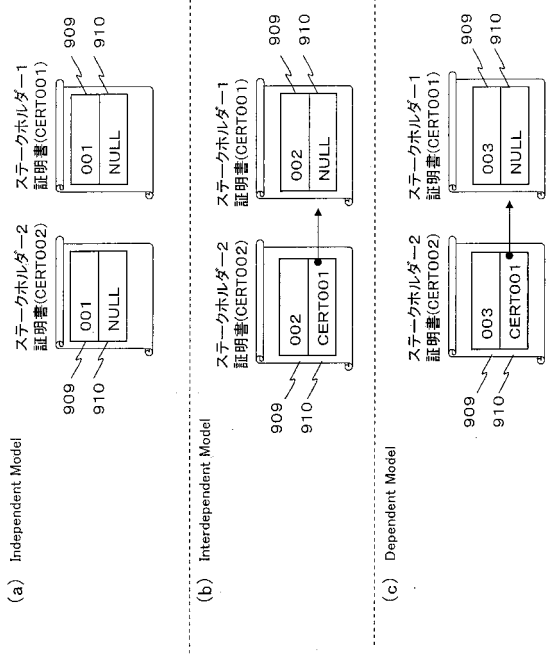
【図８】


113
811
812
鍵ID
鍵属性情報アドレス
K10 K10の鍵属性情報アドレス値
K11 K11の鍵属性情報アドレス値
K12 K12の鍵属性情報アドレス値
K13 K13の鍵属性情報アドレス値
K31 K31の鍵属性情報アドレス値
K33 K33の鍵属性情報アドレス値
K34 K34の鍵属性情報アドレス値
(a) 鍵管理テーブル1
213
821
822
鍵ID
鍵属性情報アドレス
K20 K20の鍵属性情報アドレス値
K21 K21の鍵属性情報アドレス値
K22 K22の鍵属性情報アドレス値
K23 K23の鍵属性情報アドレス値
K32 K32の鍵属性情報アドレス値
K33 K33の鍵属性情報アドレス値
K34 K34の鍵属性情報アドレス値
(b) 鍵管理テーブル2

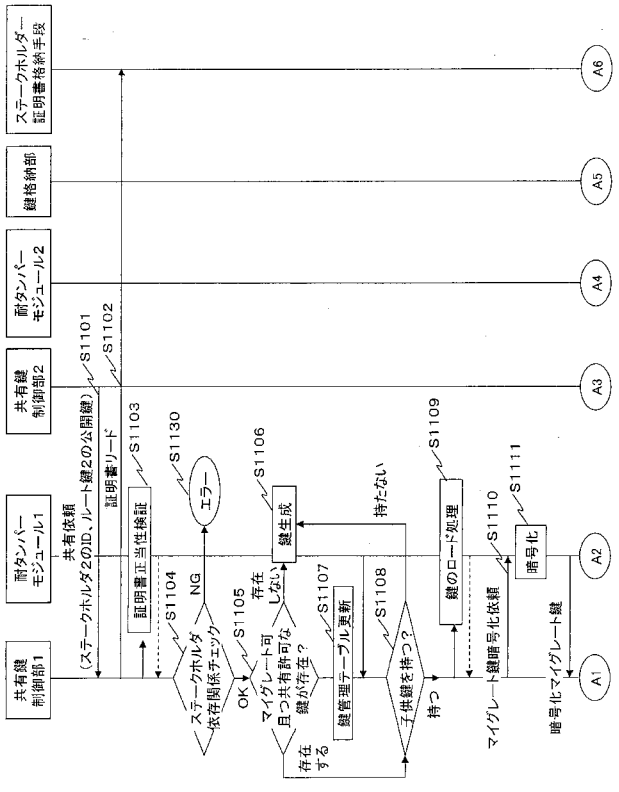
【図９】



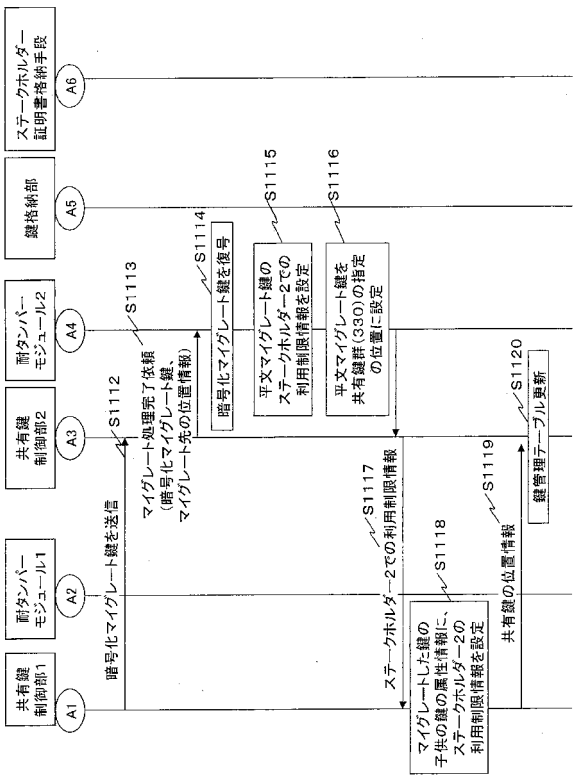
【図１０】



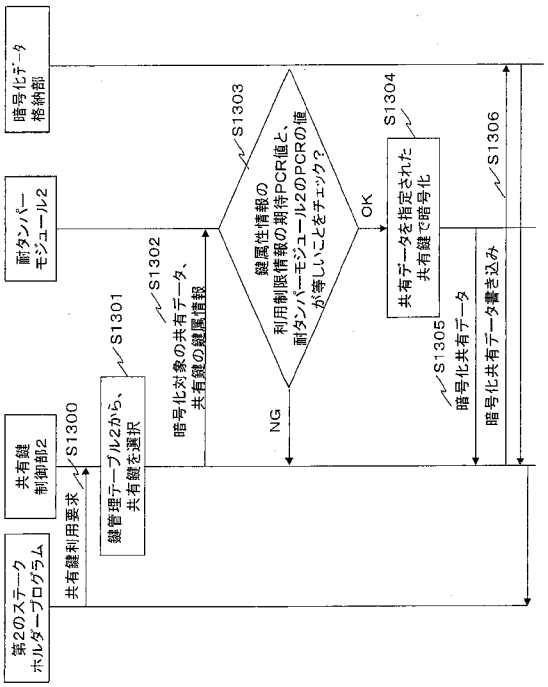
【図１１】



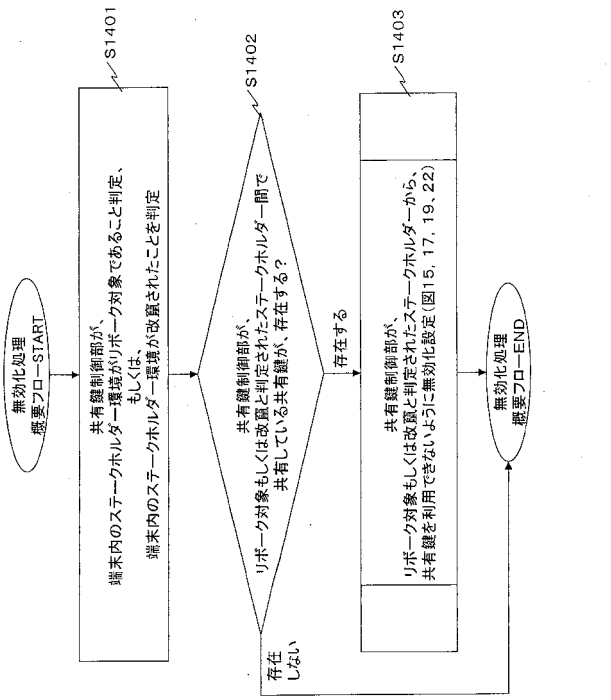
【図１２】



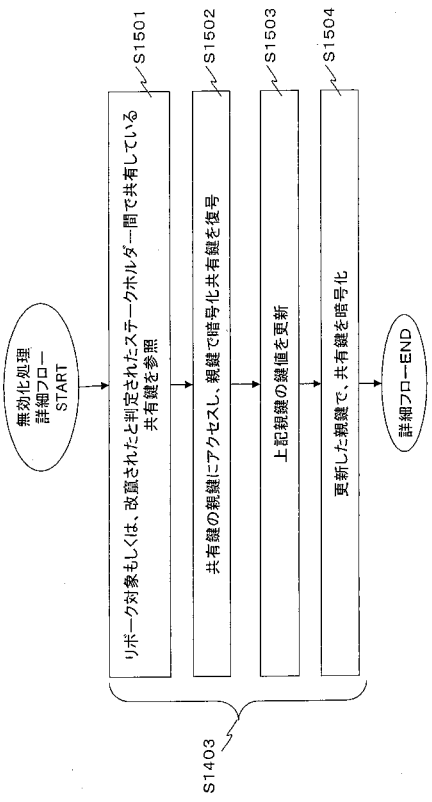
【図１３】



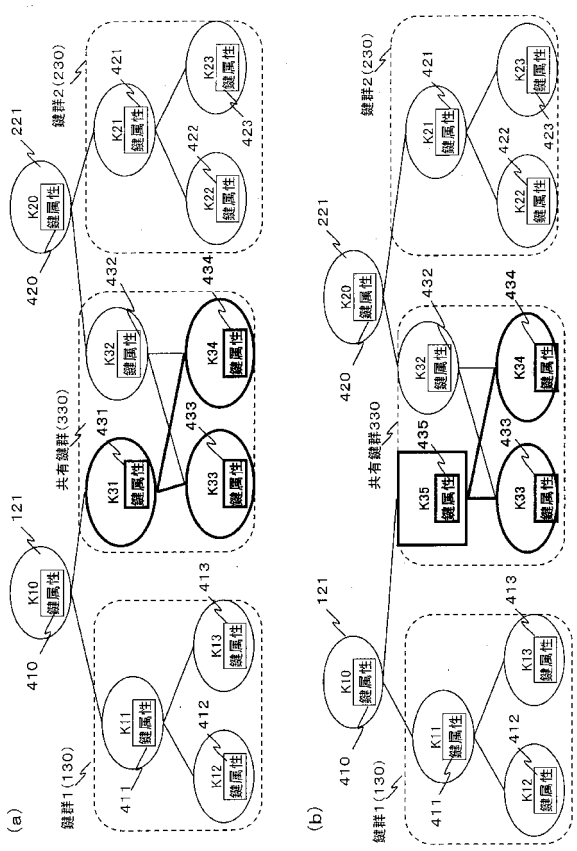
【図１４】



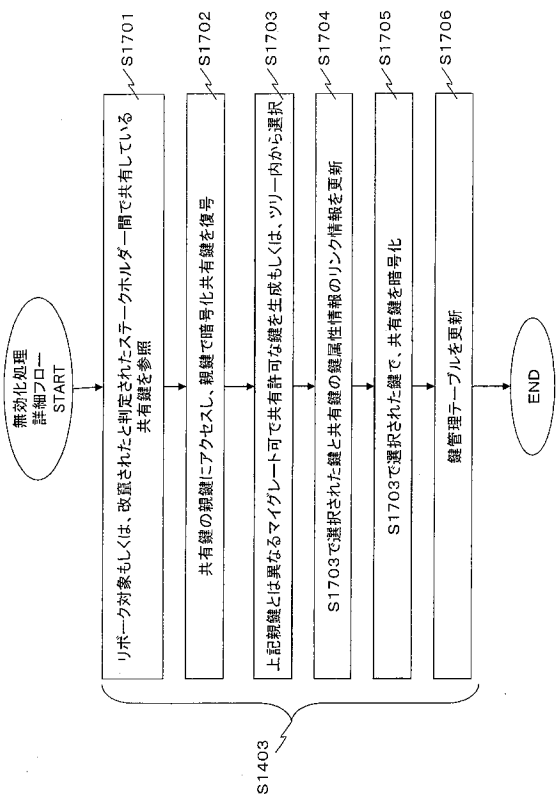
【図１５】



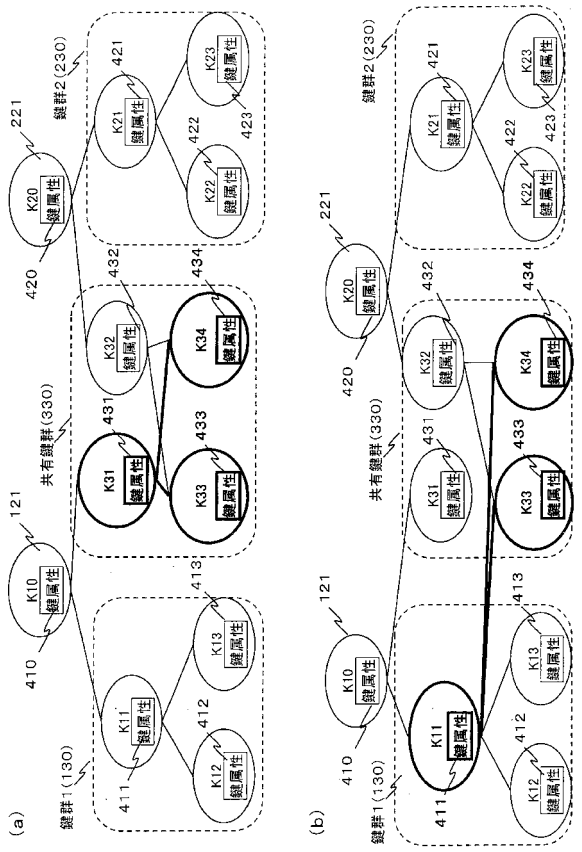
【図１６】



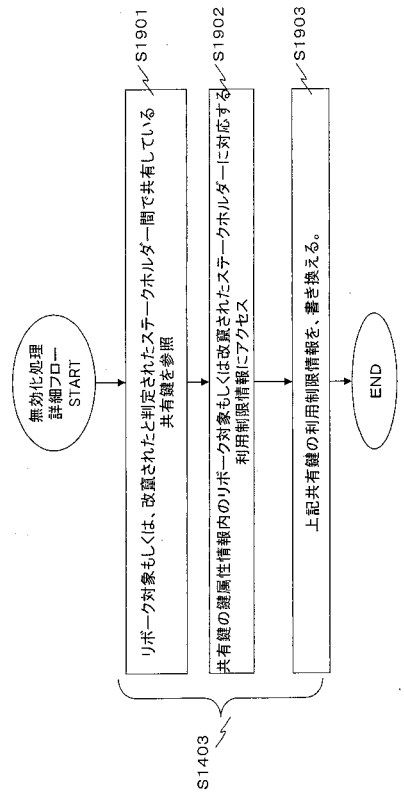
【図１７】



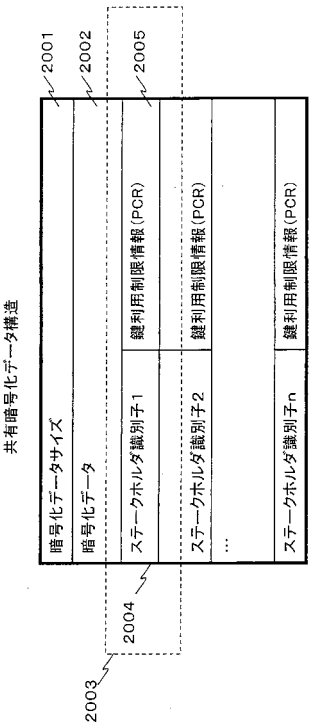
【図１８】



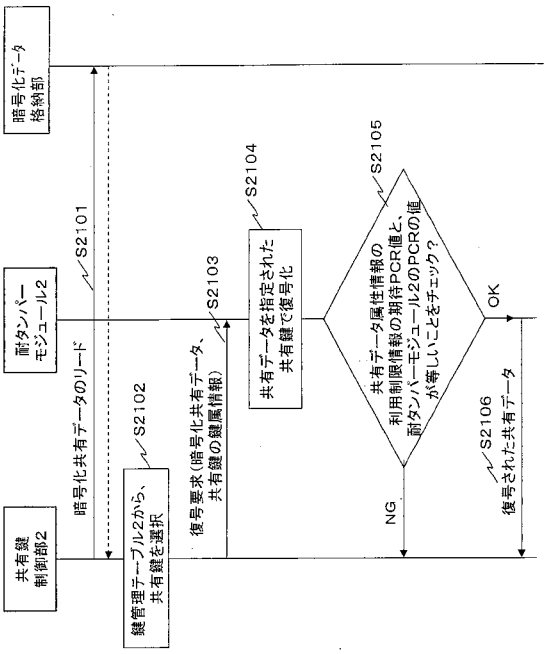
【図１９】



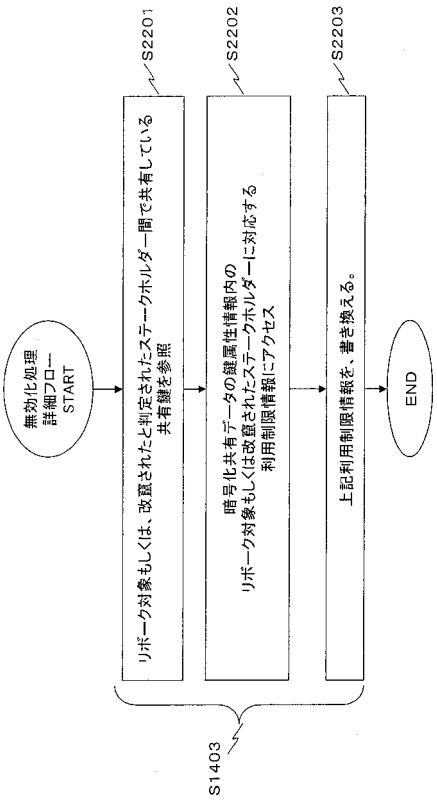
【図２０】



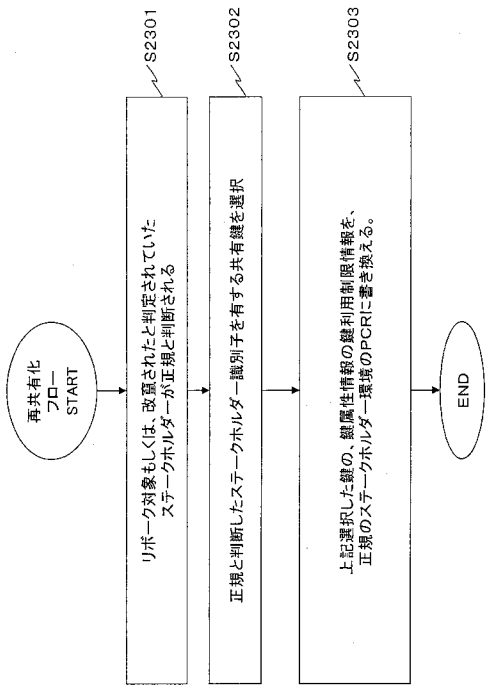
【図２１】


暗号化データ格納部
耐タンパーモジュール2
共有鍵制御部2
S2101
暗号化共有データのリード
S2102
鍵管理テーブル2から、共有鍵を選択
S2103
復号要求（暗号化共有データ、共有鍵の鍵属情報）
S2104
共有データを指定された共有鍵で復号化
S2105
共有データ属性情報の利用制限情報の期待PCR値と、耐タンパーモジュール2のPCRの値が等しいことをチェック？
OK
NG
S2106
復号された共有データ

【図２２】


無効化処理詳細フロー START
S2201
リボーク対象もしくは、改竄されたと判定されたステークホルダー間で共有している共有鍵を参照
S2202
暗号化共有データの鍵属性情報内のリボーク対象もしくは改竄されたステークホルダーに対応する利用制限情報にアクセス
S2203
上記利用制限情報を、書き換える。
END
S1403

【図２３】


再共有化フロー START
S2301
リボーク対象もしくは、改竄されたと判定されていたステークホルダーが正規と判断される
S2302
正規と判断したステークホルダー識別子を有する共有鍵を選択
S2303
上記選択した鍵の、鍵属性情報の鍵利用制限情報を、正規のステークホルダー環境のPCRに書き換える。
END

【手続補正書】
【提出日】平成22年11月18日(2010.11.18)
【手続補正１】
【補正対象書類名】明細書
【補正対象項目名】全文
【補正方法】変更
【補正の内容】
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、複数のステークホルダーを備える装置において、ツリー構造で管理される鍵を、複数のステークホルダー間で共有して利用できる共有鍵の制御を行う情報処理装置、暗号鍵の管理方法、コンピュータプログラム及び集積回路に関するものである。
【背景技術】
【０００２】
近年、情報セキュリティへの意識の高まりと共に、データを保護する技術ニーズが高まってきている。
これを背景として、セキュアなコンピュータプラットフォームを開発、普及させることを目的として、Ｔｒｕｓｔｅｄ　Ｃｏｍｐｕｔｉｎｇ　Ｇｒｏｕｐ（ＴＣＧ）が設立された。ＴＣＧでは、Ｔｒｕｓｔｅｄ　Ｐｌａｔｆｏｒｍ　Ｍｏｄｕｌｅ（ＴＰＭ）と呼ばれるセキュリティコアモジュールを利用し、安全な端末環境を実現している（非特許文献１～３参照）。ＴＣＧの基本機能として、特徴的な機能が３つある。
【０００３】
まず１つ目の特徴機能として、端末起動時からＯＳやアプリケーションが起動するまでの各モジュールのインテグリティ情報を計測し、計測した値を、それより前のモジュールのインテグリティ情報を連鎖させる累積処理（ＴＣＧのＴＰＭ＿Ｅｘｔｅｎｄコマンドに相当する処理）をし、その累積値をＴＰＭ内のＰｌａｔｆｏｒｍ　Ｃｏｎｆｉｇｒａｔｉｏｎ　Ｒｅｇｉｓｔｅｒｓ（ＰＣＲ）と呼ばれるレジスタに格納するＴｒｕｓｔｅｄ　Ｂｏｏｔ機能がある。
【０００４】
２つ目の特徴機能として、ＰＣＲに蓄積した値を端末環境情報として、外部のサーバーに通知し、外部のサーバーで、端末環境情報が期待される環境情報であるかどうかを検証するというＡｔｔｅｓｔａｔｉｏｎ機能がある。これらの機能を利用し、リモートで端末の環境の正当性を検証可能であることが、ＴＣＧ技術のメリットの１つである。
【０００５】
そして、３つ目の特徴機能として、Ｐｒｏｔｅｃｔｅｄ　Ｓｔｏｒａｇｅ機能というストレージデータに対する保護機能がある。この機能は、ＴＰＭ内のセキュアなメモリ領域に、ＴＰＭ内部で生成した暗号鍵であるＳｔｏｒａｇｅ　Ｒｏｏｔ　Ｋｅｙ（ＳＲＫ）を保持する。そして、ＳＲＫをルート鍵とし、保護対象となる複数の暗号鍵をルート鍵で暗号化してＴＰＭ外のメモリ上で安全に保護する機能である。
【０００６】
具体的には、暗復号処理や署名処理に利用する鍵を保護するために、ＳＲＫをルート鍵とした階層的なツリー構造のノードに、それら保護対象の鍵を対応づけ、親のノード鍵が子供のノード鍵を暗号化する方法である。ルート鍵であるＳＲＫがＴＰＭ内のセキュアメモリ上で保持されているため、ルート鍵以外の暗号化された鍵は、ＴＰＭ外のメモリ上で管理しても安全となる。また、これらの鍵をバックアップする目的として、マイグレート機能も有する。マイグレート機能は、他のＴＰＭのＳＲＫの下に、鍵をコピーする機能である。
【０００７】
また、ＴＣＧは、ＴＰＭ搭載端末として、携帯電話機もターゲットとしており、携帯電

話機向けのＴＰＭの仕様も規格化されている（非特許文献３、４）。携帯電話機向けのＴＰＭは、Ｍｏｂｉｌｅ Ｔｒｕｓｔｅｄ Ｍｏｄｕｌｅ（ＭＴＭ）と呼ばれている。ＭＴＭは、ＴＰＭの機能を実現しながらも、一部のコマンドを携帯電話機向けに修正したり、新規コマンドが追加されている。その追加機能として、セキュアブート機能と、マルチステークホルダーモデルを定義している。

【０００８】

セキュアブートとは、携帯電話機の端末起動時から、ＯＳやアプリケーションが起動するまでの各モジュールのインテグリティ情報を計測し、計測した値が期待される値であるかを、ローカル端末内で検証しながらブートする方式である。

【０００９】

また、マルチステークホルダーモデルとは、デバイスメーカー、キャリア、アプリケーションサービス提供者、ユーザーといった携帯電話機端末内に存在する複数のステークホルダーが所有する権利物を安全に利用するための実装モデルを定義したものである。各ステークホルダーの権利物として、例えば、デバイスメーカーであれば、Ｉｎｔｅｒｎａｔｉｏｎａｌ Ｍｏｂｉｌｅ Ｅｑｕｉｐｍｅｎｔ Ｉｄｅｎｔｉｔｙ（ＩＭＥＩ）であり、キャリアであれば、Ｓｕｂｓｃｒｉｂｅｒ Ｉｄｅｎｔｉｆｉｃａｔｉｏｎ Ｍｏｄｕｌｅ（ＳＩＭ）関連情報であり、アプリケーションサービス提供者であれば、サービス提供されたデータであり、ユーザーであれば、アドレス帳が挙げられる。

【００１０】

要するに、マルチステークホルダーモデルとは、それぞれのステークホルダーが利用するＭＴＭを、個別に割り当てることでそれぞれの権利物を安全に利用するモデルである。仮想化技術を用いることで、１つの端末内に、複数のＭＴＭを仮想的に実現することが可能となる。

【００１１】

特許文献１は、暗号鍵をツリー構造に構造化し、暗号鍵更新時に管理する方法が開示されている。

【先行技術文献】

【特許文献】

【００１２】

【特許文献１】特開平１１－１８７０１３

【非特許文献】

【００１３】

【非特許文献１】ＴＰＭ Ｍａｉｎ，Ｐａｒｔ１ Ｄｅｓｉｇｎ Ｐｒｉｎｃｉｐｌｅｓ，Ｓｐｅｃｉｆｉｃａｔｉｏｎ ｖｅｒｓｉｏｎ １．２ Ｌｅｖｅｌ２ Ｒｅｖｉｓｉｏｎ１０３（９ Ｊｕｌｙ ２００７）

【非特許文献２】ＴＰＭ Ｍａｉｎ，Ｐａｒｔ２ ＴＰＭ Ｓｔｒｕｃｔｕｒｅｓ，Ｓｐｅｃｉｆｉｃａｔｉｏｎ ｖｅｒｉｏｎ１．２ Ｌｅｖｅｌ２ Ｒｅｖｉｓｉｏｎ１０３（９ Ｊｕｌｙ ２００７）

【非特許文献３】ＴＰＭ Ｍａｉｎ Ｐａｒｔ３ Ｃｏｍｍａｎｄｓ，Ｓｐｅｃｉｆｉｃａｔｉｏｎ ｖｅｒｉｏｎ １．２ Ｌｅｖｅｌ２ Ｒｅｖｉｓｉｏｎ１０３（９ Ｊｕｌｙ ２００７）

【非特許文献４】ＴＣＧ Ｍｏｂｉｌｅ Ｔｒｕｓｔｅｄ Ｍｏｄｕｌｅ Ｓｐｅｃｉｆｉｃａｔｉｏｎ ｖｅｒｓｉｏｎ １．０ Ｒｅｖｉｓｉｏｎ １（１２ Ｊｕｎｅ ２００７）

【非特許文献５】ＴＣＧ Ｍｏｂｉｌｅ Ｒｅｆｅｒｅｎｃｅ Ａｒｃｈｉｔｅｃｔｕｒｅ Ｓｐｅｃｉｆｉｃａｔｉｏｎ ｖｅｒｓｉｏｎ１．０ Ｒｅｖｉｓｉｏｎ１（１２ Ｊｕｎｅ ２００７）

【発明の概要】

【発明が解決しようとする課題】

【００１４】

上述したマルチステークホルダーモデルにおいて、各ステークホルダーが、各々ＳＲＫをルートとした鍵ツリーを保持する。そのため、一つの端末内に、その鍵のツリーが、ステークホルダーの個数だけ存在することになる。

【００１５】

ここで、ステークホルダーＡが、ステークホルダーＢが管理しているデータや機能を利用することが想定される。特に、ステークホルダーＡが、ステークホルダーＢのＴＰＭで管理されているＳＲＫをルートするツリーのノード鍵で暗号化されているデータに対してアクセスしたい場合、ステークホルダーＡは、ステークホルダーＢに対して、ステークホルダーＢのＳＲＫを用いたデータの復号処理を要求しなければならない。そして、ステークホルダーＢは、ステークホルダーＢのＳＲＫを利用して復号処理後、復号データをステークホルダーＡに送信する。

【００１６】

このように、異なるステークホルダー間で共有したいデータがあった場合、その共有データのアクセス要求のたびに、鍵を管理しているステークホルダーが復号処理をして、安全に復号データを渡すといったオーバーヘッド処理が必要になる。

【００１７】

これらのオーバーヘッド処理を回避するために、異なるステークホルダー間でアクセスされるセキュアな共有データに対しては、異なるステークホルダー間で共有な鍵で暗号化することが必要となる。言い換えると、異なるステークホルダー間で共有な鍵を保持させるような仕組みが必要となる。

【００１８】

現状のＴＣＧ仕様で、ステークホルダーＡとＢの２者間で鍵を共有させる場合、すなわち、ステークホルダーＢの鍵を、ステークホルダーＡでも利用できるようにするには、ＴＣＧのマイグレート機能を利用する。マイグレート元をステークホルダーＢとし、マイグレート先をステークホルダーＡとし、ステークホルダーＢの鍵をステークホルダーＡへマイグレートする。その結果、ステークホルダーＢからマイグレートされた鍵をステークホルダーＡが管理する鍵ツリーのノードとして構成することが可能となる。

【００１９】

これにより、共有鍵を保持しているステークホルダーＡとＢの間では、マイグレートされた鍵が、共有鍵となり、その共有鍵で暗号化したデータに対して、ステークホルダーを跨ることなく、直接アクセス可能となる。

【００２０】

しかしながら、マイグレート機能を利用した場合、マイグレート対象となった鍵の実体は、ステークホルダーＡとＢの各々で保持しているため、１つの端末に、共有鍵が２重持ちされることになり、非効率であるという課題があった。特に、共有鍵を複数設定する場合、２重持ちされる鍵が複数存在することになる。

【００２１】

また、特許文献１は、暗号鍵をツリー構造で管理し、子供のノードが親のノードを暗号化する構成であり、リーフからルートに至る経路の鍵群をユーザー鍵としてユーザーに配布し、ユーザーが脱退した際の鍵ツリーの更新方法を開示している。

【００２２】

しかしながら、ルート鍵は全てのユーザーで共有しているため、一人のユーザーが脱退する度に、更新後のルート鍵を全ユーザーに再配布しなければならないため、管理が複雑になってしまうという課題があった。

【００２３】

そこで、本発明は、これらの課題を解決するもので、マルチステークホルダーモデルにおいて、ステークホルダー毎に異なるＳＲＫをルート鍵とする鍵ツリー間のノードの一部を共有鍵をとして設定する情報処理装置、暗号鍵の管理方法、コンピュータプログラム及び集積回路と、共有鍵を共有しているステークホルダーに不正があった場合に、不正なステークホルダーからの共有鍵へのアクセスを、より柔軟に無効化する情報処理装置、暗号

鍵の管理方法、コンピュータプログラム及び集積回路を提供することを目的とする。
【課題を解決するための手段】
【００２４】
上記の課題を解決するために、本発明に係る情報処理装置は、第１のステークホルダーに対応する第１共有鍵制御部と、第２のステークホルダーに対応する第２共有鍵制御部と、前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデータ保持部と、前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備し、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第１共有鍵制御部は、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断し、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーし、前記第２共有鍵制御部は、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することを特徴とする。
【発明の効果】
【００２５】
本発明によると、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を前記第２の暗号鍵群の中にコピーすることにより、前記第２のステークホルダーが前記第１のステークホルダーに依存する関係であることを条件に、前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体をコピーするのではなく、前記所定の鍵のみをコピーする。これにより、前記第１の耐タンパーモジュール及び前記第２の耐タンパーモジュールとで前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体を二重持ちする非効率を回避できる。
【００２６】
また、前記第２共有鍵制御部側で、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持し、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することにより、前記所定の鍵をコピーするだけで、前記第２共有鍵制御部側では、第１共有鍵制御部に対応する第１の耐タンパーモジュールが管理する第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用できる。これにより、前記第２共有鍵制御部は、前記データ保持部で保持された暗号化された所定のデータを簡易な構成で利用できる。
【図面の簡単な説明】
【００２７】
【図１】本発明の実施の形態１におけるシステム概要を示す図である。
【図２】本発明の実施の形態１におけるマルチステークホルダーを示す図である。
【図３】本発明の実施の形態１における情報処理端末の構成を示す図である。
【図４】本発明の実施の形態１における耐タンパーモジュールが有する鍵ツリーの構成を示す図である。
【図５】本発明の実施の形態１における鍵属性情報を示す図である。
【図６】本発明の実施の形態１における鍵のリンク情報示した図である。
【図７】本発明の実施の形態１における鍵属性情報の例を示した図である。
【図８】本発明の実施の形態１における鍵管理テーブルを示した図である。

【図９】本発明の実施の形態１におけるステークホルダー証明書の構成を示した図である。
【図１０】本発明の実施の形態１におけるステークホルダー証明書を用いてトラストモデルを表現した例を示した図である。
【図１１】本発明の実施の形態１における共有鍵の設定のシーケンス図である。
【図１２】本発明の実施の形態１における共有鍵の設定のシーケンス図である。
【図１３】本発明の実施の形態１における共有鍵を利用シーケンス図である。
【図１４】本発明の実施の形態１における共有鍵の無効化概要フローチャートを示す図である。
【図１５】本発明の実施の形態１における共有鍵の無効化詳細フローチャートを示す図である。
【図１６】本発明の実施の形態１における共有鍵の無効化前後における鍵ツリーの構成を示す図である。
【図１７】本発明の実施の形態２における共有鍵の無効化詳細フローチャートを示す図である。
【図１８】本発明の実施の形態２における共有鍵の無効化前後における鍵ツリーの構成を示す図である。
【図１９】本発明の実施の形態３における共有鍵の無効化詳細フローチャートを示す図である。
【図２０】本発明の実施の形態４における暗号化共有データ構造を示す図である。
【図２１】本発明の実施の形態４における暗号化共有データ構造を利用した暗号化共有データの復号処理のシーケンス図である。
【図２２】本発明の実施の形態４における共有鍵の無効化詳細フローチャートを示す図である。
【図２３】本発明の実施の形態３における共有鍵の再共有化のフローチャートを示す図である。
【発明を実施するための形態】
【００２８】
本発明の請求項１に記載の情報処理装置は、第１のステークホルダーに対応する第１共有鍵制御部と、第２のステークホルダーに対応する第２共有鍵制御部と、前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデータ保持部と、前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備し、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第１共有鍵制御部が、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断し、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーし、前記第２共有鍵制御部は、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することを特徴とする。
【００２９】
本態様により、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を前記第２の暗号鍵群の中にコピーすることにより、前記第２のステーク

ホルダーが前記第１のステークホルダーに依存する関係であることを条件に、前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体をコピーするのではなく、前記所定の鍵のみをコピーする。これにより、前記第１の耐タンパーモジュール及び前記第２の耐タンパーモジュールとで前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体を二重持ちする非効率を回避できる。

【００３０】

また、前記第２共有鍵制御部側で、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持し、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することにより、前記所定の鍵をコピーするだけで、前記第２共有鍵制御部側では、第１共有鍵制御部に対応する第１の耐タンパーモジュールが管理する第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用できる。これにより、前記第２共有鍵制御部は、前記データ保持部で保持された暗号化された所定のデータを簡易な構成で利用できる。

【００３１】

さらに、前記第２共有鍵制御部は、前記第２のステークホルターが第１のステークホルダーに依存する関係にある場合にのみ、前記データ保持部内に保持された暗号化された所定のデータを利用できる。これにより、前記所定のデータを管理する鍵構成を前記第１の耐タンパーモジュール及び前記第２の耐タンパーモジュールで簡易にしつつ、前記所定のデータの機密性を保証できる。

【００３２】

本発明の請求項２に記載の情報処理装置は、前記第２共有鍵制御部が、前記第１共有鍵制御部との間の依存関係を証明した証明書を有し、前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を送付する際、前記証明書を送付し、前記第１共有鍵制御部が、前記証明書に基づいて、前記第２共有鍵制御部に対応する第２のステークホルダーが、少なくとも前記第１のステークホルダーに対応する第１の耐タンパーモジュールを利用するステークホルダーモデルであると判断した場合に、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であると判断するものである。

【００３３】

本態様によると、前記第１共有鍵制御部は、前記第２共有鍵制御部に対応する第２のステークホルダーが、少なくとも前記第１のステークホルダーに対応する第１の耐タンパーモジュールを利用するステークホルダーモデルであると証明書に基づいて判断した場合に、前記第２のステークホルダーが前記第１のステークホルダーに依存する関係であると判断する。これにより、前記第２のステークホルダーの前記第１のステークホルダーに対する依存関係を確実に判断できるので、前記所定のデータを管理する鍵構成を前記第１の耐タンパーモジュール及び前記第１の耐タンパーモジュールで簡易にしつつ、不正なステークホルダーからの前記所定のデータへのアクセスを確実に禁止できる。

【００３４】

本発明の請求項３に記載の情報処理装置は、前記第１の暗号鍵群に含まれる各鍵が、当該鍵が前記第１の暗号鍵群からコピー可能か否かを示す属性情報を有し、前記第１共有鍵制御部は、前記属性情報を参照して、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探すことを特徴としている。

【００３５】

本態様によると、当該鍵が前記第１の暗号鍵群からコピー可能か否かを示す属性情報を参照して、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探す。これにより、前記属性情報を参照するだけでコピー可能な鍵を探せるので、コピー可能な鍵を簡易にサーチできる。

【００３６】

本発明の請求項４に記載の情報処理装置は、前記第１共有鍵制御部が、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵が存在しない場合

、コピー可能な鍵を生成して、この生成した鍵を前記第２の暗号鍵群の中にコピーすることを特徴としている。

【００３７】

本態様によると、前記第１共有鍵制御部は、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵が存在しない場合、コピー可能な鍵を生成して、この生成した鍵を前記第２の暗号鍵群の中にコピーする。これにより、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵が存在しない場合であっても、前記第２共有鍵制御部は前記第１の暗号鍵群に含まれる鍵を共用できるので、前記第２共有鍵制御部は前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデータ保持部にアクセスできる。

【００３８】

本発明の請求項５に記載の情報処理装置は、前記第１共有鍵制御部が、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵の位置を示す位置情報を前記所定の鍵のリンク情報として生成し、このリンク情報と共に前記所定の鍵を前記第２の暗号鍵群の中にコピーすることを特徴とする請求項３記載の情報処理装置。

【００３９】

本態様によると、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵の位置を示す位置情報を前記所定の鍵のリンク情報として生成してコピーすることにより、第２共有鍵制御部では、前記所定の鍵のリンク情報を参照すれば、前記所定の鍵より下層の鍵の位置を確認できるので、前記所定の鍵より下層の鍵を前記第２の耐タンパーモジュールにコピーすることなく、前記第１の耐タンパーモジュールとの間で前記所定の鍵より下層の鍵を共用できる。その結果、前記第１の耐タンパーモジュール及び前記第１の耐タンパーモジュールとで前記所定の鍵を親鍵とするツリー構造に含まれる鍵群の全体を二重持ちする非効率を回避できる。

【００４０】

本発明の請求項６に記載の情報処理装置は、前記第１共有鍵制御部が、前記第１の暗号鍵群に含まれる所定の鍵の位置情報及び前記第２の暗号鍵群に含まれる所定の鍵の位置情報を、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵のリンク情報として生成することを特徴としている。

【００４１】

本態様により、前記第１の暗号鍵群に含まれる所定の鍵の位置情報及び前記第２の暗号鍵群に含まれる所定の鍵の位置情報を、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵のリンク情報として生成することにより、前記下層の鍵のリンク情報を参照すれば前記下層の鍵を暗号化した親鍵の所在を認識できる。この結果、前記第１の耐タンパーモジュール及び前記第１の耐タンパーモジュールとで前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用する場合であっても、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵がどの鍵で暗号化されているかを容易に識別できる。

【００４２】

本発明の請求項７に記載の情報処理装置は、前記第１共有鍵制御部と前第２共有鍵制御部とは、共用の共有鍵制御部であることを特徴としている。

本態様により、前記第１共有鍵制御部と前第２共有鍵制御部とは、共用の共有鍵制御部で構成が可能となり、１つの共有鍵制御部にて、２つのステークホルダー間の共有鍵を統括的に制御することが可能なので、より柔軟にアクセス制御を行うことが可能となる。

【００４３】

また、同じ共有鍵を保持している場合、自身以外の共有先のステークホルダーの脆弱性が原因で、共有していた鍵が暴露される危険性がある。そのため、その自身以外の共有先のステークホルダーが、不正と判断（リボーク対象もしくは、改竄されていると検知）された場合には、そのステークホルダーから、共有鍵で暗号化されているデータを利用不可にするためのアクセス制御が必要となる。

【００４４】

本発明の請求項８に記載の情報処理装置は、前記第１のステークホルダーが管理する第１ステークホルダー環境が、前記第２のステークホルダーが管理する第２のステークホルダー環境が改竄された、もしくはリボーク対象であることを検知した場合、前記第１共有鍵制御部は、前記所定の鍵と置換える代替鍵を前記第１の耐タンパーモジュールに生成させ、前記所定の鍵を親鍵とするツリー構造に含まれる鍵を前記代替鍵で再暗号化させると共に前記前記所定の鍵の親鍵を用いて前記代替鍵を暗号化させて、前記第２共有鍵制御部による前記所定のデータの利用を排除することを特徴としている。

【００４５】

本態様によると、前記第１のステークホルダーが管理する第１ステークホルダー環境が、前記第２のステークホルダーが管理する第２のステークホルダー環境が改竄された、もしくはリボーク対象であることを検知した場合、前記第１の耐タンパーモジュールは前記所定の鍵と置換える代替鍵を生成して前記所定の鍵を親鍵とするツリー構造に含まれる鍵を前記代替鍵で再暗号化すると共に前記所定の鍵の親鍵を用いて前記代替鍵を暗号化する。この結果、前記第２共有鍵制御部は前記所定の鍵を用いて前記代替鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、前記代替鍵を親鍵とするツリー構造に含まれる鍵で暗号化された所定のデータを利用できず、前記所定データを不正な利用から保護できる。

【００４６】

本発明の請求項９に記載の情報処理装置は、前記第１のステークホルダーが管理する第１ステークホルダー環境が、前記第２のステークホルダーが管理する第２のステークホルダー環境が改竄された、もしくはリボーク対象であることを検知した場合、前記第１共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵を用いて前記第１の耐タンパーモジュールに前記所定のデータを暗号化し直させて、前記第２共有鍵制御部による前記所定の鍵の使用を排除することを特徴とする。

【００４７】

本態様によると、前記第２のステークホルダー環境が攻撃されたことを、第１のステークホルダー環境が検知した場合、前記第１の耐タンパーモジュールは前記所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵を用いて前記所定のデータを暗号化し直す。この結果、前記第２共有鍵制御部は前記所定の鍵を親鍵とするツリー構造に含まれる鍵を用いては前記所定のデータを復号化できないので、前記所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵で暗号化された所定のデータを利用できず、前記所定データを不正な利用から保護できる。

【００４８】

本発明の請求項１０に記載の情報処理装置は、前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵が、属性情報として、前記改竄のない第２のステークホルダーが管理する第２ステークホルダー環境のハッシュ値から生成された期待値として鍵利用制限情報を有し、第２の耐タンパーモジュールは、前記第２のステークホルダー環境のハッシュ値から生成された実際の値としての環境情報を記憶し、第２の共有鍵制御部から前記第１の共有鍵制御部に対して前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵の利用を依頼するときに、第２の共有鍵制御部は、前記鍵利用制限情報と前記環境情報とを比較し、比較結果が正しい場合にのみ前記鍵を利用させるように制限をすることを特徴する。

【００４９】

本態様によると、第２の共有鍵制御部は、前記改竄のない第２のステークホルダー環境から生成された鍵利用制限情報と前記第２のステークホルダー環境から実際に得られた環境情報とを比較し、比較結果が正しい場合にのみ前記鍵を利用させる。この結果、前記第２ステークホルダー環境が改竄され若しくはリボークされた場合には前記比較結果は不一致となって、前記第２共有鍵制御部は前記所定の鍵を用いては前記所定の鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、前記所定の鍵を親鍵とするツリー構造に含まれる鍵用いて暗号化されている前記所定のデータを復号化できず、前記所定データを

不正な利用から保護できる。

【００５０】

本発明の請求項１１に記載の情報処理装置は、前記第１ステークホルダーが管理する第１のステークホルダー環境が、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵を、第２の共有鍵制御部から利用できないように、前記鍵利用制限情報を書き換えることを特徴とする。

【００５１】

本態様によると、前記第１ステークホルダーが管理する第１のステークホルダー環境は、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵を、第２の共有鍵制御部から利用できないように、前記鍵利用制限情報を書き換える。この結果、前記第２共有鍵制御部は前記所定の鍵を用いては前記所定の鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、前記所定の鍵を親鍵とするツリー構造に含まれる鍵用いて暗号化されている前記所定のデータを復号化できず、前記所定データを不正な利用から保護できる。

【００５２】

本発明の請求項１２に記載の情報処理装置は、前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データが、属性情報として、前記改竄のない第２のステークホルダーが管理する第２ステークホルダー環境のハッシュ値から生成された期待値である暗号化データ利用制限情報を有し、第２の耐タンパーモジュールは、前記第２のステークホルダー環境のハッシュ値から生成された実際の値としての環境情報を記憶し、第２の共有鍵制御部から前記第１の共有鍵制御部に対して前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化されたデータの復号処理を依頼するときに、第２の共有鍵制御部は、前記暗号化データ利用制限情報と前記環境情報とを比較し、比較結果が正しい場合にのみ前記暗号化データの復号処理させるように制限をすることを特徴する。

【００５３】

本態様によると、第２の共有鍵制御部は、前記改竄のない第２のステークホルダー環境から生成された暗号化データ利用制限情報と前記第２のステークホルダー環境から実際に得られた環境情報とを比較し、比較結果が正しい場合にのみ暗号化データを復号させる。この結果、前記第２ステークホルダー環境が改竄され若しくはリボークされた場合には前記比較結果は不一致となって、前記第２共有鍵制御部は前記所定の鍵を用いては前記所定の鍵を親鍵とするツリー構造に含まれる鍵を用いて暗号化されている前記暗号化データを復号化できず、前記暗号化データを不正な利用から保護できる。

【００５４】

本発明の請求項１３に記載の情報処理装置は、前記第１ステークホルダーが管理する第１のステークホルダー環境が、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データを、第２の共有鍵制御部から利用できないように、前記暗号化データ利用制限情報を書き換えることを特徴とする。

【００５５】

本態様によると、前記第１ステークホルダーが管理する第１のステークホルダー環境は、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データを、第２の共有鍵制御部から利用できないように、前記暗号化データ利用制限情報を書き換える。この結果、前記第２共有鍵制御部は前記所定の鍵を用いては前記所定の鍵を親鍵とするツリー構造に含まれる鍵を用いた復号処理ができないので、前記所定データを不正な利用から保護できる。

【００５６】

本発明の請求項１４に記載の情報処理装置は、前記第１共有鍵制御部は第１ステークホルダー環境および第２ステークホルダー環境を環境に対して完全性をチェックしてから改竄されていない環境のみを起動する機能であるセキュアブートによってブートする際に、第２ステークホルダー環境が改竄された、もしくはリボーク対象であることを検知することを特徴とする。

【００５７】

本態様によると、前記第１共有鍵制御部は第１ステークホルダー環境および第２ステークホルダー環境を環境に対して完全性をチェックしてから改竄されていない環境のみを起動する機能であるセキュアブートによってブートする際に、第２ステークホルダー環境が改竄された、もしくはリボークされたことを検知することにより、前記第２共有鍵制御部の外部からの攻撃を判断できる。

【００５８】

本発明の請求項１５に記載の情報処理装置は、前記第１共有鍵制御部が、外部のサーバーから、前記第２共有鍵制御部が改竄されたもしくはリボーク対象である旨の通知を受けることで前記第２共有鍵制御部が外部から攻撃されたことを検知することを特徴とする。

【００５９】

本態様によると、前記第１共有鍵制御部は、外部のサーバーから、前記第２共有鍵制御部が外部から攻撃された旨の通知を受けることにより、前記第２共有鍵制御部の外部からの攻撃を検知できる。

【００６０】

本発明の請求項１６に記載の暗号鍵の管理方法は、第１のステークホルダーに対応する第１共有鍵制御部と、第２のステークホルダーに対応する第２共有鍵制御部と、前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置における暗号鍵の鍵管理方法であって、前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断し、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーし、前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することを特徴とする。

【００６１】

本発明の請求項１７に記載のコンピュータプログラムは、第１のステークホルダーに対応する第１共有鍵制御部と、第２のステークホルダーに対応する第２共有鍵制御部と、前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置における暗号鍵の鍵管理に用いるコンピュータプログラムであって、コンピュータに対して、前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断する処理と、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対

応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーする処理と、を実行させ、前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用する処理を実行させることを特徴とする。

【００６２】

本発明の請求項１８に記載の集積回路は、第１のステークホルダーに対応する第１共有鍵制御部と、第２のステークホルダーに対応する第２共有鍵制御部と、前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置に用いる集積回路であって、情報処理部と、この情報処理部に対して、前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断する処理と、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーする処理と、を実行させ、前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用する処理を実行させる処理プログラムを格納したメモリと、を具備した集積回路とする。

【００６３】

発明の請求項１９に記載の情報処理装置は、前記第１共有鍵制御部は、前記第１の暗号鍵群に含まれる暗号鍵で暗号化されている前記所定の鍵を、前記暗号鍵を用いて復号し、復号された鍵を、前記第２の暗号鍵群に含まれる暗号鍵で再暗号化し、再暗号化された鍵を、前記第２の暗号鍵群の中にコピーすることを特徴とする。

【００６４】

以下、本発明の実施の形態について、図面を参照しながら説明する。

（実施の形態１）

本発明の実施の形態について、説明する。以降、本実施の形態で示すＴＰＭは、ＴＣＧのＭｏｂｉｌｅ Ｐｈｏｎｅ Ｗｏｒｋｉｎｇ Ｇｒｏｕｐで仕様化されているＭｏｂｉｌｅ Ｔｒｕｓｔｅｄ Ｍｏｄｕｌｅ（ＭＴＭ）の機能を有する耐タンパーモジュールであるとして説明する。

【００６５】

＜図１：システム概要＞

まず、マルチステークホルダーについて携帯電話機を例として説明する。携帯電話機には、デバイスメーカー、キャリア、アプリケーションサービス提供者、ユーザーといった複数のステークホルダーが存在する。各ステークホルダーは、各自の権利物を所有している。

【００６６】

各ステークホルダーの権利物は、例えば、デバイスメーカーであれば、Ｉｎｔｅｒｎａｔｉｏｎａｌ Ｍｏｂｉｌｅ Ｅｑｕｉｐｍｅｎｔ Ｉｄｅｎｔｉｔｙ（ＩＭＥＩ）であり、キャリアであれば、Ｓｕｂｓｃｒｉｂｅｒ Ｉｄｅｎｔｉｆｉｃａｔｉｏｎ Ｍｏｄｕｌｅ（ＳＩＭ）関連情報であり、アプリケーションサービス提供者であれば、サービス提供されたデータであり、ユーザーであれば、アドレス帳が挙げられる。

【００６７】

マルチステークホルダーモデルとは、それぞれのステークホルダーが利用するＴＰＭを、個別に割り当てることでそれぞれの権利物を安全に利用するモデルである。

図１は、マルチステークホルダーモデルにおけるシステム全体を示した図である。

【００６８】

情報処理端末１０は、ＭＴＭを搭載したモバイル端末である。実施の形態１では、以降、情報処理装置１０内には、第１のステークホルダーと第２のステークホルダーとの２つのステークホルダーが存在するという例を用いて説明する。なお、ステークホルダーは前述しているように、２つ以上であってもよい。また、情報処理端末は、携帯電話機であってもよいし、ＰＤＡなどのモバイル端末、あるいは、ＴＶやＤＶＤやＢＤプレイヤーやＳＴＢなどの据え置き型の電子機器であってもよい。

【００６９】

第１のステークホルダー管理サーバー１１は、第１のステークホルダー環境を提供するステークホルダーが管理しているサーバーである。第１のステークホルダー管理サーバー１１は、認証ＰＣＲデータベース１２、証明書管理データベース１３及びリボケーションリスト１４を管理している。

【００７０】

認証ＰＣＲデータベース１２は、第１のステークホルダー管理サーバー１１が、情報処理端末１０の正当性を検証するＡｔｔｅｓｔａｔｉｏｎ処理の際に利用するデータベースであり、正当な情報処理端末１０の環境情報であるＰＣＲの期待値のデータベースである。

【００７１】

第１のステークホルダー管理サーバー１１は、Ａｔｔｅｓｔａｔｉｏｎ時に、情報処理端末１０から送信されたＰＣＲ値と、認証ＰＣＲデータベース１２の記録している値を比較し、一致すれば正当な情報処理端末１０と判断し、適切なサービス等を提供する。

【００７２】

証明書データベース１３は、第１のステークホルダーから提供されるソフトウェアの証明書のデータベースである。第１のステークホルダーのモジュールの更新が必要であれば、更新すべきモジュールとともに、更新モジュールの証明書を送信する。また、更新モジュールの証明書も、証明書データベース１３で管理される。

【００７３】

リボケーションリスト１４は、リボーク対象であるステークホルダーのリストを記録しているデータベースである。リボケーションリスト１４は、第１のステークホルダー管理サーバー１１から情報処理端末１０へ送信され、情報処理端末１０内で完全性を保護した状態で管理される。

【００７４】

なお、リボケーションリスト１４は、無効化すべきステークホルダーの情報を記載したブラックリスト方式として説明するが、有効なステークホルダーの情報を記載したホワイトリスト方式を用いてもよい。

【００７５】

なお、図１では、認証ＰＣＲデータベース１２、証明書データベース１３及びリボケーションリスト１４は、第１のステークホルダー管理サーバー１１が、全て管理しているが、複数の管理サーバーで管理してもよい。

【００７６】

第２のステークホルダー管理サーバー１６は、第２のステークホルダー環境を提供するステークホルダーが管理しているサーバーである。図１で図示していないが、第２のステークホルダーも、第１のステークホルダー管理サーバー１１と同様に、認証ＰＣＲデータベース１２、証明書管理データベース１３及びリボケーションリスト１４を管理している。

【００７７】

＜図２：マルチステークホルダーモデルにおけるトラストモデル＞

図２は、マルチステークホルダーモデルにおけるトラストモデルを示した図である。

トラストモデルとして、３つのモデルが定義される。また、図２には図示していないが、各ステークホルダーは、各自のステークホルダーの権利物を管理しており、ステークホルダーの所有する権利物は、各ステークホルダーに対応づけられたＴＰＭを利用して、安全にアクセスされる。以下、それぞれ３つのモデルについて説明する。

【００７８】

図２（ａ）は、Ｉｎｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌを示している。このモデルは、各ステークホルダー間に信頼の依存関係はないモデルである。例えば、ステークホルダー１（２１）が、ＴＰＭ１（２３）を利用し、ステークホルダー２（２２）は、ＴＰＭ２（２４）を利用するモデルである。

【００７９】

図２（ｂ）は、Ｉｎｔｅｒｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌを示している。このモデルは、ステークホルダー間で、部分的に依存関係のあるモデルである。例えば、ステークホルダー１（３１）は、ＴＰＭ１（３３）を利用し、ステークホルダー２（３２）は、ＴＰＭ２（３４）を利用する。ここまでは、Ｉｎｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌと同じであるが、図２（ｂ）に示しているように、一部領域が重なっている部分が存在する。これは、ステークホルダー２（３２）が、ＴＰＭ２（３４）以外に、ＴＰＭ１（３３）の機能を利用することを概念的に表している。

【００８０】

例えば、携帯電話機の場合、ステークホルダー２（３２）が、キャリアであって、ステークホルダー１（３１）がデバイスメーカーであった場合、キャリアが、デバイスメーカーの権利物であるＩＭＥＩにアクセスするといった例である。この場合、ステークホルダー２（３２）は、ステークホルダー１（３１）に対して、ＩＭＥＩアクセス要求をすることになるため、ステークホルダー２（３２）は、ステークホルダー１(３１)経由でＴＰＭ１（３３）を利用することになる。

【００８１】

図２（ｃ）は、Ｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌを示している。このモデルは、ステークホルダー間で、あるステークホルダーが、別のステークホルダーに完全に依存するモデルである。これは、ステークホルダー１（４１）がＴＰＭ１（４３）を利用し、ステークホルダー２（４２）もＴＰＭ１（４３）を利用するモデルである。携帯電話機の場合、ステークホルダー２が、キャリアであって、ステークホルダー１がデバイスメーカーであった場合、デバイスメーカーの権利物であるＩＭＥＩは、ＴＰＭ１（４３）の機能により保護され、キャリアの権利物であるＳＩＭ情報もＴＰＭ１（４３）の機能を利用して安全に保護される。

【００８２】

＜図３：情報処理端末＞

図３は、マルチステークホルダーモデルの情報処理端末１０の全体構成図である。

情報処理端末１０は、第１のステークホルダープログラム１００、第２のステークホルダープログラム２００、共有鍵制御部（１１１，２１０）、耐タンパーモジュール（１２０，２２０）、鍵格納部３０、暗号化データ格納部４０、及び、ステークホルダー証明書格納部５０から構成される。また、図示していないが、情報処理端末１０は、ＣＰＵ、Ｉ／Ｏデバイス、ＲＡＭなどの揮発メモリ、ＲＯＭやＦｌａｓｈメモリなどの不揮発メモリなどのハードウェア群を保持する。

【００８３】

＜ステークホルダープログラム＞

第１のステークホルダープログラム１００は、第１のステークホルダーから提供されるプログラム群であり、第１のステークホルダー管理サーバー１１から配布されるものである。第１のステークホルダープログラム１００は、耐タンパーモジュール１２０のセキュアブート機能により、正当性を検証されたプログラムのみが起動される。

【００８４】

第２のステークホルダープログラムは、第２のステークホルダーから提供されるプログラム群であり、第２のステークホルダー管理サーバー１６から配布されるものである。第２のステークホルダープログラム２００は、耐タンパーモジュール２２０のセキュアブート機能により、正当性を検証されたプログラムのみが起動される。

【００８５】

なお、セキュアブートの仕様については、非特許文献４に詳細に記載されているので説明を省略する。

＜共有鍵制御部（１１０、２１０）＞

共有鍵制御部１（１１０）は、鍵群１（１３０）と共有鍵群（３３０）との利用制御を行う部であり、マルチステークホルダーモデル判定部１（１１１）と、共有許可設定部１（１１２）と鍵管理テーブル１（１１３）とから構成される。

【００８６】

マルチステークホルダーモデル判定部１（１１１）は、共有鍵制御部１（１１０）が管理している鍵群１（１１３）もしくは共有鍵（３３０）に対して、ステークホルダー１以外のステークホルダーから共有鍵の設定の要求があった場合に、要求元のステークホルダーが、鍵を共有してよいステークホルダーであるかどうかをステークホルダー証明書格納部５０に格納しているステークホルダー証明書（１５０，２５０）を参照して判断する。

【００８７】

共有許可設定部１（１１２）は、鍵群１（１３０）に属する鍵を、共有鍵群（３３０）の共有鍵として設定したり、新規に鍵を生成したり、鍵のマイグレート処理を制御したりと、各種共有鍵を設定する際に必要な鍵処理をする部である。ここでの鍵処理は、共有鍵制御部１（１１２）と、耐タンパーモジュール１（１２０）とが連携して行う。

【００８８】

鍵管理テーブル１（１１３）は、共有鍵制御部１（１１０）から、鍵群１（１３０）と共有鍵群（３３０）とにアクセスするために必要な情報が記載されているテーブルである。鍵管理テーブル１（１１３）は、図８を用いて後述する。

【００８９】

共有鍵制御部２（２１０）は、鍵群２（２３０）と共有鍵群（３３０）の利用制御を行う部であり、マルチステークホルダーモデル判定部２（２１１）と、共有許可設定部２（２１２）と、鍵管理テーブル２（２１３）から構成される。

【００９０】

マルスステークホルダーモデル判定部２（２１１）は、共有鍵制御部２（２１０）が管理している鍵群２（２３０）もしくは共有鍵（３３０）に対して、ステークホルダー２以外のステークホルダーから共有鍵の設定の要求があった場合に、要求元のステークホルダーが、鍵を共有してよいステークホルダーであるかどうかをステークホルダー証明書格納部５０に格納しているステークホルダー証明書（１５０，２５０）を参照して判断する。

【００９１】

共有許可設定部２（２１２）は、鍵群２（２３０）の鍵を、共有鍵群（３３０）の共有鍵として設定したり、新規に鍵を生成したり、鍵のマイグレート処理を制御したりと、各種共有鍵を設定する際に必要な鍵処理をする部である。ここでの鍵処理は、共有鍵制御部２（２１２）と、耐タンパーモジュール２（２２０）とが連携して行う。

【００９２】

鍵管理テーブル２（２１３）は、鍵群２（２３０）と共有鍵群（３３０）とにアクセスするために必要な情報が記載されているテーブルである。鍵管理テーブル２（２１３）は、図８を用いて後述する。

【００９３】

なお、共有鍵制御部１（１１０）及び共有鍵制御部２（２１０）は、それぞれ、第１のステークホルダープログラム、第２のステークホルダープログラムとして実現されていてもよい。これにより、共有鍵制御部１（１１０）及び共有鍵制御部２（２１０）は、ＴＣＧのモバイル仕様で規定されるセキュアブートで完全性が検証されてから起動することが

可能となる。

【００９４】

＜耐タンパーモジュール（１２０、２２０）＞

耐タンパーモジュール１（１２０）は、ＭＴＭ機能を有するものとして実装されているとして説明する。そのため、耐タンパーモジュール１（１２０）は、暗復号処理や署名生成・検証処理やＴＰＭ機能処理などのセキュアな処理や、鍵の制御処理をする際に、第１のステークホルダープログラム及び共有鍵制御部１（１１０）などから利用される。さらに、耐タンパーモジュール１（１２０）は、耐タンパーモジュール１内の不揮発性メモリ上にルート鍵１（１２１）を保持する。このルート鍵１（１２１）は、ＴＣＧにおけるＳＲＫに相当する鍵である。

【００９５】

同様に、耐タンパーモジュール２（２２０）は、ＭＴＭ機能を有するものとして実装されているとして説明する。そのため、耐タンパーモジュール２（２２０）は、暗復号処理や署名生成・検証処理やＴＰＭ機能処理などのセキュアな処理や、鍵の制御処理をする際に、第２のステークホルダープログラムおよび共有鍵制御部２（２１０）などから利用される。さらに、耐タンパーモジュール２（２２０）は、耐タンパーモジュール１内の不揮発なメモリ上にルート鍵２（２２１）を保持する。このルート鍵２（２２１）は、ＴＣＧにおけるＳＲＫに相当する鍵である。

【００９６】

＜鍵格納部３０＞

鍵群１（１３０）と共有鍵群３３０とは、ルート鍵１（１２１）をルートとした階層的ツリー構造のノード鍵として構成される。これは、ＴＣＧのＰｒｏｔｅｃｔｅｄ　Ｓｔｏｒａｇｅ機能を実現するための鍵ツリー構造に相当する。

【００９７】

鍵群１（１３０）は、１つ以上の鍵から構成されたツリー構造を有する鍵群である。鍵群１（１３０）の個々の鍵は、共有鍵制御部１（１１０）から耐タンパーモジュール１（１２０）を経由して暗復号化処理や署名生成・検証処理に利用される。

【００９８】

鍵群２（２３０）と共有鍵群３３０とは、ルート鍵２（２２１）をルートとした階層的ツリー構造のノード鍵として構成される。これは、ＴＣＧのＰｒｏｔｅｃｔｅｄ　Ｓｔｏｒａｇｅ機能を実現するための鍵ツリー構造に相当する。

【００９９】

鍵群２（２３０）は、１つ以上の鍵から構成されたツリー構造を有する鍵群である。鍵群２（２３０）の個々の鍵は、共有鍵制御部２（２１０）から耐タンパーモジュール２（２２０）を経由して暗復号化処理や署名生成・検証処理に利用される。

【０１００】

一方、共有鍵群３３０は、１つ以上の鍵から構成されたツリー構造を有する鍵群である。共有鍵群３３０の個々の鍵は、共有鍵制御部１（１１０）から耐タンパーモジュール１（１１０）経由で暗復号化処理や署名生成・検証処理に利用される。共有鍵群３３０は、共有鍵制御部２（２１０）から耐タンパーモジュール２（２２０）経由でも暗復号化処理や署名生成・検証処理に利用される。

【０１０１】

共有鍵群３３０は、共有鍵と、その共有鍵を保護するための共有鍵の親鍵とから構成される。共有鍵の親鍵は、共有鍵制御部１（１１０）から耐タンパーモジュール１（１１０）経由でのみ利用される鍵と、共有鍵制御部２（２１０）から耐タンパーモジュール２（２１０）経由でのみ利用される鍵とから構成される。　これら鍵群１（１３０）、鍵群２（２３０）及び共有鍵群（３３０）は、鍵格納部３０に格納される。鍵群１（１３０）、鍵群２（２３０）及び共有鍵群（３３０）の構造については、図４から図７を用いてさらに詳しく説明する。

【０１０２】

＜暗号化データ格納部４０＞

暗号化データ１（１４０）は、鍵群１（１３０）の鍵で暗号化されたデータである。暗号化共有データ３４０は、共有鍵群３３０の鍵で暗号化されたデータである。暗号化データ２（２４０）は、鍵群２（２３０）の鍵で暗号化されたデータである。

【０１０３】

暗号化データ１（１４０）、暗号化データ２（２４０）、及び暗号化共有データ３４０は、暗号化データ格納部４０に格納される。暗号化データ格納部４０は、ＨＤＤやフラッシュメモリなどの不揮発メモリで構成される。

【０１０４】

なお、図３では、暗号化データ１（１４０）、暗号化データ２（２４０）、及び暗号化共有データ３４０は、暗号化データとしているが、暗号化データに限定されず、それぞれの鍵で署名生成したデータであってもよい。

【０１０５】

＜ステークホルダー証明書格納部５０＞

ステークホルダー証明書格納部５０は、ステークホルダー証明書１（１５０）とステークホルダー証明書２（２５０）とを格納する部である。ステークホルダー証明書格納部５０は、不揮発メモリで実現され、完全性が保護された形で管理される。

【０１０６】

ステークホルダー証明書１（２５０）は、第１のステークホルダープログラム、共有鍵制御部１（１１０）、及び耐タンパーモジュール１（１２０）が、正規のステークホルダーから提供されてことを示す証明書である。

【０１０７】

ステークホルダー証明書２（２５０）は、第２のステークホルダープログラム、共有鍵制御部２（２１０）、及び耐タンパーモジュール２（２２０）が、正規のステークホルダーから提供されてことを示す証明書である。

【０１０８】

ステークホルダー証明書（１５０、２５０）は、それぞれ、依存関係のあるステークホルダーを識別できる情報が記載される。ステークホルダー証明書（１５０，２５０）の構成などの詳細は、図９及び図１０を用いて後述する。

【０１０９】

＜図４：共有鍵の鍵ツリー構成＞

図４は、鍵群１（１３０）と鍵群２（２３０）と共有鍵群（３３０）とのツリー構成を表した図である。なお、図３で既に説明している構成要素については、説明を省略する。図４では、図３における耐タンパーモジュール（１２０，２２０）、鍵群１（１３０）、鍵群２（２３０）、及び共有鍵群（３３０）を、より詳細に示したものである。

【０１１０】

耐タンパーモジュール１（１２０）は、耐タンパーモジュール１（１２０）の外部から不正なアクセスができないように保護されたセキュアメモリ（１２２）と、１６個のＰＣＲ（１２３）とを備えている。ルート鍵１（１２１）は、セキュアメモリ（１２２）に安全に保持される。

【０１１１】

耐タンパーモジュール２（２２０）は、耐タンパーモジュール２（２２０）の外部から不正なアクセスができないように保護されたセキュアメモリ（２２２）と、１６個のＰＣＲ（２２３）とを備えている。ルート鍵２（２２１）は、セキュアメモリ（２２２）に安全に保持される。

【０１１２】

なお、ＰＣＲ（１２０、２２０）は、Ｐｌａｔｆｏｒｍ Ｃｏｎｆｉｇｒａｔｉｏｎ Ｒｅｇｉｓｔｅｒｓと呼ばれるレジスタであり、ＴＣＧのＴＰＭ＿Ｅｘｔｅｎｄコマンドにより生成されたインテグリティ値が格納される。なお、ＰＣＲの個数は、１６個に限定されるわけではなく、それより多くても少なくても良い。実施の形態では、ＴＣＧの仕様

で決められている個数以上の数を備えるものとする。
【０１１３】
以降、ルート鍵１（１２１）をＫ１０、ルート鍵２（２２１）をＫ２０として説明する。
鍵群１（１３０）は、３つの鍵（Ｋ１１、Ｋ１２、Ｋ１３）から構成されている。Ｋ１１は、Ｋ１０の子供の鍵としてツリー構造化され、Ｋ１１はＫ１０により暗号化される。Ｋ１２及びＫ１３は、共にＫ１１の子供の鍵としてツリー構造化され、Ｋ１２及びＫ１３はＫ１１で暗号化される。
【０１１４】
鍵群２（２３０）は、３つの鍵（Ｋ２１、Ｋ２２、Ｋ２３）から構成されている。Ｋ２１は、Ｋ２０の子供の鍵としてツリー構造化され、Ｋ２１はＫ２０により暗号化される。Ｋ２２及びＫ２３は、共にＫ２１の子供の鍵としてツリー構造化され、Ｋ２２及びＫ２３はＫ３１で暗号化される。
【０１１５】
共有鍵群３３０は、４つの鍵（Ｋ３１、Ｋ３２、Ｋ３３、Ｋ３４）から構成されている。Ｋ３１は、Ｋ１０の子供の鍵としてツリー構造化され、Ｋ３１はＫ１０により暗号化される。したがって、Ｋ３１は、共有鍵制御部１（１１０）から耐タンパーモジュール１（１１０）経由でのみ利用される鍵となる。なぜなら、Ｋ３１は、Ｋ１０で暗号化されているため、Ｋ３１の鍵値を利用する場合は、Ｋ１０を用いてＫ３１を復号しなければならないからである。
【０１１６】
一方、Ｋ３２は、Ｋ２０の子供の鍵としてツリー構造化され、Ｋ３２はＫ２０により暗号化される。したがって、Ｋ３２は、共有鍵制御部２（２１０）から耐タンパーモジュール２（２１０）経由でのみ利用される鍵となる。なぜなら、Ｋ３２は、Ｋ２０で暗号化されているため、Ｋ３２の鍵値を利用する場合は、Ｋ２０を用いてＫ３２を復号しなければならないからである。
【０１１７】
続いて、Ｋ３３及びＫ３４について説明する。Ｋ３３は、Ｋ３１及びＫ３２の子供の鍵としてツリー構造化され、Ｋ３４は、Ｋ３１及びＫ３２の子供の鍵としてツリー構造化されている。
【０１１８】
Ｋ３３及びＫ３４は、共有鍵制御部１（１１０）から耐タンパーモジュール１（１１０）経由で暗復号化処理や署名生成・検証処理に利用され、さらに共有鍵制御部２（２１０）から耐タンパーモジュール２（２２０）経由でも暗復号化処理や署名生成・検証処理に利用される共有鍵である。そのため、Ｋ３３及びＫ３４は、上述したように耐タンパーモジュール１（１１０）、及び、耐タンパーモジュール２（２１０）の両モジュールから利用可能な鍵である。
【０１１９】
このようにするために、Ｋ３１及びＫ３２は、同じ鍵値とする。親鍵が子供の鍵を暗号化する方式をとったツリー構造としているため、Ｋ３１及びＫ３２が同じ鍵値であれば、ルート鍵がＫ１０とＫ２０のように異なっていても、Ｋ３３及びＫ３４は復号できる。
【０１２０】
Ｋ３１及びＫ３２を同じ鍵値に設定する方法については、ＴＣＧにおけるマイグレート機能を利用する。具体的なフローについては図１１及び図１２を用いて後述するので、ここでの説明は省略する。
【０１２１】
また、暗号化データ１（１４０）は、暗号化データＤ１２とＤ１３とから構成されている。Ｄ１２は、Ｋ１２で暗号化されたデータであり、Ｄ１３は、Ｋ１３で暗号化されたデータである。なお、暗号化データ１（１４０）の暗号化データは、これに限定されることなく、Ｋ１２及びＫ１３のそれぞれを署名鍵として用い、署名されたデータであってもよ

い。また、Ｋ１２で、暗号化されたデータが複数あってもよいし、Ｋ１３で、暗号化されたデータが複数あってもよい。

【０１２２】

また、暗号化データ２（２４０）は、暗号化データＤ２２とＤ２３とから構成されている。Ｄ２２は、Ｋ２２で暗号化されたデータであり、Ｄ２３は、Ｋ２３で暗号化されたデータである。なお、暗号化データ２（２４０）の暗号化データは、これに限定されることなく、Ｋ２２及びＫ２３をそれぞれ署名鍵として用い、署名されたデータであってもよい。また、Ｋ２２で、暗号化されたデータが複数あってもよいし、Ｋ２３で、暗号化されたデータが複数あってもよい。

【０１２３】

また、暗号化共有データ３４０は、暗号化共有データＤ３３とＤ３４とから構成されている。Ｄ３３は、Ｋ３３で暗号化されたデータであり、Ｄ３４は、Ｋ３４で暗号化されたデータである。なお、暗号化共有データ３４０の暗号化データは、これに限定されることなく、Ｋ３３及びＫ３４をそれぞれ署名鍵として用い、署名されたデータであってもよい。また、Ｋ３３で、暗号化されたデータが複数あってもよいし、Ｋ３４で、暗号化されたデータが複数あってもよい。

【０１２４】

なお、鍵群１（１３０）、鍵群２（２３０）、及び共有鍵群３３０の鍵の個数と、暗号化データ１（１４０）、暗号化データ２（２４０）、及び暗号化共有データ３４０のデータの個数は、図４で示しているものに限定されない。また、Ｋ１０、Ｋ２０をルートする鍵ツリー構造は、２分木で構成しているが、３分木、Ｎ分木（Ｎは整数）であってもよい。

【０１２５】

＜図５：鍵属性情報＞

ルート鍵１（１２１）、ルート鍵２（２２１）、鍵群１（１３０）、鍵群２（２３０）、共有鍵群３３０のそれぞれの鍵値は、鍵属性情報の１要素として鍵値を持つ。

【０１２６】

例えば、ルート鍵１（１２１）であるＫ１０の鍵値は、鍵属性情報４１０内に記録されている。ルート鍵２（２２１）であるＫ２０の鍵値は、鍵属性情報４２０内に記録されている。鍵群１（１３０）の鍵であるＫ１１の鍵値は、鍵属性情報４１１内に記録されている。鍵群２（２３０）の鍵であるＫ２１の鍵値は、鍵属性情報４２１内に記録されている。共有鍵群（３３０）の鍵であるＫ３１の鍵値は、鍵属性情報４３１内に記録されている。そのほかの鍵値も同様に記録されているので、説明を省略する。

【０１２７】

ここで、鍵属性情報は、鍵の値と共に、鍵の属性を示す情報も同じデータ構造として記録されている。

図５は、鍵属性情報の構成を示した図である。鍵属性情報４３４は、マイグレート許可フラグ５０１と、共有許可フラグ５０２と、鍵のアルゴリズムを識別するための情報である暗号アルゴリズムと鍵値のサイズを示した５０３と、鍵値５０４とを備える。

【０１２８】

マイグレート許可フラグ５０１は、鍵のマイグレートが許可されているかどうかを示すフラグ情報であり、「０」であればマイグレート不可、「１」であればマイグレート許可を示す。

【０１２９】

共有許可フラグ５０２は、共有鍵として利用可能かどうかを示すフラグ情報であり、「０」であれば共有不可、「１」であれば共有許可を示す。

鍵属性情報４１０は、さらに、鍵が、複数のステークホルダーから共有鍵としてアクセス可能な場合に、そのステークホルダーの情報を格納するステークホルダーフィールド５０５を有する。複数のステークホルダーからアクセス可能な鍵であれば、ステークホルダーフィールド５０５は、アクセスされるステークホルダーの数だけ列挙される。

【０１３０】
図５に示した鍵属性情報４３４は、ステークホルダー識別子１からｎまでのステークホルダーフィールド５０５が設定されている。
図４では、Ｋ３４は、ステークホルダー１とステークホルダー２とからアクセス可能な例であるので、ステークホルダーフィールド５０５は、２つ存在することになる。
【０１３１】
ステークホルダーフィールド５０５は、ステークホルダー識別子５０６と、鍵の利用制限を示す鍵利用制限情報５０７と、リンク情報５０８とを含む。
鍵利用制限情報５０７は、鍵を利用する際に耐タンパーモジュールの備えるＰＣＲ（１２３、２２３）に記録されていることが期待されるＰＣＲ値である。鍵利用制限情報５０７は、耐タンパーモジュールの備えるＰＣＲ（１２３、２２３）に記録されている実際の値と比較され、実際のＰＣＲ値と期待されるＰＣＲ値とが等しい場合にのみ、鍵が利用できるように制限するための情報である。
【０１３２】
リンク情報５０８は、個々の鍵に対する親鍵を識別するためのリンク情報、もしくは、個々の鍵に対する子供の鍵を識別するためのリンク情報である。
＜図６：リンク情報＞
ここで、図６を用いてリンク情報５０８の構造について説明する。
【０１３３】
図６（ａ）のリンク情報５０８は、個々の鍵に対する親鍵を識別するための情報である。複数の親鍵が存在するのであれば、図に示すようにリンク情報５０８には、複数の親鍵へのポインタ（６０１，６０２，６０３）が格納される。
【０１３４】
図６（ｂ）のリンク情報５０８は、個々の鍵に対する子供の鍵を識別するための情報である。複数の子供の鍵が存在するのであれば、図に示すようにリンク情報５０８には、複数の子供の鍵へのポインタ（６１１，６１２，６１３）が格納される。
【０１３５】
＜図７：鍵属性情報の例＞
図７は、鍵のツリー構成と鍵属性情報の関係の例を表した図である。図７では、図４に図示している一部の鍵について抜粋して説明する。
【０１３６】
ルート鍵１（Ｋ１０）の鍵属性情報４１０は、マイグレート不許可であり、共有不許可であり、暗号アルゴリズムがＲＳＡアルゴリズムで鍵長が２０４８ビットであることを示している。また、鍵値５０４のフィールドには、Ｋ１０の公開鍵の鍵値と、Ｋ１０の秘密鍵の鍵値とが設定されている。
【０１３７】
そして、Ｋ１０はステークホルダー１に対してアクセスを許可させるため、ステークホルダーフィールド５０５には、ステークホルダー識別子５０６として、ステークホルダー１の識別子「ＳＨ１」、鍵利用制限情報５０７に期待されるＰＣＲの情報としてＰＣＲ＿１０が示されている。さらに、リンク情報５０８には、図６（ａ）の親鍵へのリンク情報が設定される。Ｋ１０はルート鍵なので、親鍵は存在しないので、リンク情報５０８には「ＮＵＬＬ」と設定される。さらに、Ｋ１０は、親鍵が存在しないので、Ｋ１０の鍵値５０４のフィールドには、平文の鍵値が設定される。
【０１３８】
鍵群１（１３０）の鍵Ｋ１１の鍵属性情報４１１は、マイグレート許可であり、共有許可であり、暗号アルゴリズムがＲＳＡアルゴリズムで、鍵長が２０４８ビットであることを示している。また、鍵値５０４のフィールドには、Ｋ１１の公開鍵の鍵値と、Ｋ１０の公開鍵で暗号化されたＫ１１の秘密鍵の鍵値とが設定される。そしてＫ１１はステークホルダー１に対してアクセスを許可させるため、ステークホルダーフィールド５０５には、ステークホルダー識別子５０６として、ステークホルダー１の識別子「ＳＨ１」が設定さ

れる。さらに、リンク情報５０８には、図６（ａ）で説明した親鍵へのリンク情報が記載されている。Ｋ１１の親鍵はＫ１０であるので、リンク情報５０８には「Ｋ１０へのポインタ情報」が設定される。具体的に、このポインタ情報は、Ｋ１０へ鍵属性情報４１０を参照できる情報であれば、アドレスでもよいし、識別ＩＤでもよい。鍵利用制限情報５０７に期待されるＰＣＲの情報として「ＮＵＬＬ」として設定される。この「ＮＵＬＬ」は、Ｋ１１を利用する際のＰＣＲの制限はない鍵であることを示している。

【０１３９】

共有鍵群（３３０）のＫ３３の鍵属性情報４３３は、マイグレート許可であり、共有許可であり、暗号アルゴリズムがＡＥＳアルゴリズムで鍵長が２５６ビットであることを示している。また、鍵値５０４のフィールドには、Ｋ３１もしくはＫ３２の公開鍵で暗号化されたＫ３３鍵値が設定される。そしてＫ３３はステークホルダー１とステークホルダー２との間で共有できる共有鍵であるため、ステークホルダーフィールド５０５には、ステークホルダー識別子５０６として、ステークホルダー１の識別子「ＳＨ１」と「ＳＨ２」とが設定される。また、鍵利用制限情報５０７には、期待されるＰＣＲの情報として「ＳＨ１」からの利用制限に使うＰＣＲ＿３３＿１と「ＳＨ１」からの利用制限に使うＰＣＲ＿３３＿２とが記載されている。そして、リンク情報５０８には、図６（ａ）で説明した親鍵へのリンク情報が記載されている。Ｋ３３は共有鍵であり、親鍵はＫ３１とＫ３２であるので、リンク情報５０８には「Ｋ３１へのポインタ情報」と「Ｋ３２へのポインタ情報」が設定される。

【０１４０】

他の鍵（Ｋ２０、Ｋ３１、Ｋ３２、Ｋ３４）についても同様なので、説明を省略する。

なお、実施の形態１では、暗号化アルゴリズムをＲＳＡ、または、ＡＥＳとしているが、暗号アルゴリズムはこれに限定されない。公開鍵暗号系であれば、ＲＳＡでなく楕円曲線暗号でもよい、また、共通鍵暗号系ではＡＥＳ以外のアルゴリズムでもよい。鍵長も本実施の形態１の例に限定はされない。また、親子関係で親鍵が公開鍵暗号系のアルゴリズムであれば、その鍵は、親鍵の公開鍵で暗号化される。また親鍵が共通鍵暗号系のアルゴリズムであれば、その鍵は、親鍵で暗号化される。また、子供鍵の公開鍵暗号系アルゴリズムの鍵であれば秘密鍵が暗号化対象となり、共通鍵暗号系であれば、その暗号鍵が暗号化対象となる。

【０１４１】

なお、鍵属性情報４３４は、鍵値５０５を含む構成としているが、鍵値５０５と、それ以外の属性情報は別のデータとして構成するようにしてもよい。

＜図８：鍵管理テーブル＞

次に、鍵管理テーブルについて説明する。

【０１４２】

鍵管理テーブル１（１１３）は、共有鍵制御部１（１１０）が利用するテーブルである。鍵管理テーブル１（１１３）は、鍵ＩＤ８１１と鍵属性情報アドレス８１２とから構成される。鍵ＩＤ８１１は、各鍵を識別するための識別子である。鍵属性情報アドレス８１２は、各鍵ＩＤ８１１に対応する鍵属性情報が格納されているアドレス値が設定される。この２つの情報を利用することで、共有鍵制御部１（１１３）は、所望の鍵にアクセスする。

【０１４３】

鍵管理テーブル２（２１３）は、共有鍵制御部２（２１０）が利用するテーブルである。鍵管理テーブル２（２１３）は、鍵ＩＤ８２１と鍵属性情報アドレス８２２とから構成される。鍵ＩＤ８２１は、各鍵を識別するための識別子である。鍵属性情報アドレス８２２は、各鍵ＩＤ８２１に対応する鍵属性情報が格納されているアドレス値が設定される。この２つの情報を利用することで、共有鍵制御部２（２１３）は、所望の鍵にアクセスする。

【０１４４】

ここで、共有鍵として設定されている鍵Ｋ３３及びＫ３４は、鍵管理テーブル１（１１

３）と鍵管理テーブル２（２１３）との両テーブルに登録されている。

＜図９：ステークホルダー証明書＞

次に、図９を用いて、ステークホルダー証明書について説明する。ステークホルダー証明書１（１５０）、及び、ステークホルダー証明書２（２５０）は、すべて同じフォーマットであるとする。具体的には、Ｘ．５０９形式のフォーマットを利用する。

【０１４５】

ステークホルダー証明書（ＴＰＭ証明書）は、Ｘ．５０９のバージョンを示す証明書バージョン９０１、発行者によって一意な値を割り振られたシリアルナンバー９０２、証明書の署名検証に用いる署名アルゴリズムを示す署名アルゴリズム情報９０３、発行者情報９０４、証明書の有効期間９０５、証明書を受ける対象を示したサブジェクト９０６、鍵値や公開鍵アルゴリズムを示す公開鍵情報９０７、ＴＰＭバージョン９０８、トラストモデル識別情報９０９、依存ステークホルダー証明書識別情報９１０、拡張領域９１１、及び、これらのデータに対する署名データ９１２から構成される。

【０１４６】

拡張領域９１１には、ＣＲＬやＩＳＯ９０００００などの製造プロセスや、ＥＡＬなどのコモンクライテリアといったセキュリティ関連情報を記載してもよいし、機能制御の条件と機能制御の内容が記載されてもよい。

【０１４７】

本実施の形態では、トラストモデル識別情報９０９と依存ステークホルダー証明書識別情報９１０とを用いてトラストモデルを定義している。以下、これらの構成について詳細に説明する。

【０１４８】

なお、本実施の形態では、Ｘ．５０９形式のフォーマットとしているが、これ以外のフォーマットであってもよい。例えば、ＭＴＭ仕様で規定されているＲＩＭ証明書のフォーマットを利用してもよい。ＲＩＭ証明書を利用することで、ＭＴＭの証明書検証用のコマンドを利用して、証明書検証を行うことが可能となる。ＲＩＭ証明書については、非特許文献４に詳細に記載されているので、ここでの説明を省略する。

【０１４９】

＜図１０：ステークホルダー間の依存関係＞

トラストモデル情報識別情報９０９は、３つのトラストモデルであるＩｎｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌ、Ｉｎｔｅｒｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌ、及び、Ｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌを識別するため情報が記載される。

【０１５０】

依存ステークホルダー証明書識別情報９１０は、トラストモデルにおける信頼関係のあるステークホルダー証明書へのポインタ情報を格納する。

図１０（ａ）は、Ｉｎｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌの具体例である。この例では、Ｉｎｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌを示すトラストモデル識別情報を「００１」としている。ステークホルダー１とステークホルダー２との間で依存関係がないモデルであるので、ＣＥＲＴ００１とＣＥＲＴ００２の依存ステークホルダーモデル識別情報９１０には「ＮＵＬＬ」と設定されている。

【０１５１】

図１０（ｂ）は、Ｉｎｔｅｒｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌの具体例である。この例は、ステークホルダー２がステークホルダー１に対して信頼の依存関係があるモデルである。そのため、ＣＥＲＴ００２の依存ステークホルダー識別情報９１０には、信頼の依存先ステークホルダーである「ＣＥＲＴ００１」と設定されている。

【０１５２】

図１０（ｃ）は、Ｄｅｐｅｎｄｅｎｔ　Ｍｏｄｅｌの具体例である。この例は、ステークホルダー２がステークホルダー１に対して信頼の依存関係があるモデルである。そのため、ＣＥＲＴ００２の依存ステークホルダー識別情報９１０には、信頼の依存先ステークホルダーである「ＣＥＲＴ００１」と設定されている。

【０１５３】
＜図１１、１２：共有鍵設定フロー＞
図１１及び図１２は、共有鍵制御部１（１１０）で管理している鍵群１（１３０）の鍵について、共有鍵制御部２（２１０）から共有鍵としての利用要求があった場合のフローである。
【０１５４】
図４の鍵構成を例にフローの概要を説明すると、耐タンパーモジュール１（１２０）は、Ｋ３１を耐タンパーモジュール２（２２０）にマイグレートし、耐タンパーモジュール２（２２０）は、マイグレートされたＫ３１をＫ３２として管理し、Ｋ３１の子供鍵であるＫ３３とＫ３４とを、ステークホルダー１とステークホルダー２の共有鍵として設定する。
【０１５５】
ここでは、図１１及び図１２を用いて、共有鍵の設定フローの詳細を説明する。
まず、共有鍵制御部２（２１０）は、共有鍵制御部１（１１０）に対して、共有鍵設定要求データとして、ステークホルダー２を識別するＩＤと、ルート鍵２であるＫ２０の公開鍵を送付する（ステップＳ１１０１）。ここで、共有鍵制御部２（２１０）から直接共有鍵制御部１（１１０）に対してデータを送信しているが、第２のステークホルダープログラムから第１のステークホルダープログラムに対してＳ１１０１の要求を出すようにし、第１のステークホルダープログラムは、共有鍵制御部１（１１０）に対して処理要求を出し、第２のステークホルダープログラムは、共有鍵制御部２（２１０）に対して処理要求を出すようにしてもよい。
【０１５６】
次に、共有鍵制御部１（１１０）は、ステークホルダー証明書格納部５０から、Ｓ１１０１で受信したＩＤに対応するステークホルダー証明書をリードする（ステップＳ１１０３）。図４の例では、ステークホルダー証明書２（２５０）をリードする。
【０１５７】
次に、共有鍵制御部１（１１０）は、耐タンパーモジュール１（１２０）を利用してステークホルダー証明書の検証を行う（ステップＳ１１０３）。Ｓ１１０３の検証の結果、ステークホルダー証明書が正当でないと判断されたら、共有鍵制御部１（１１０）は、エラー処理へと処理を移す（Ｓ１１３０）。Ｓ１１０３の検証の結果、証明書が正当であれば、ステップＳ１１０４に処理を移す。
【０１５８】
次に、ステークホルダー間の依存関係を、Ｓ１１０３で検証したステークホルダー証明書を用いてチェックする。ステークホルダー証明書のトラストモデル識別情報をチェックし、ＩｎｔｅｒｄｅｐｅｎｄｅｎｔもしくはＤｅｐｅｎｄｅｎｔ ｍｏｄｅｌであることを確認する。
【０１５９】
そして、ＩｎｔｅｒｄｅｐｅｎｄｅｎｔもしくはＤｅｐｅｎｄｅｎｔ ｍｏｄｅｌであることが確認できたら、ステークホルダー証明書格納部５０から、依存ステークホルダー識別子が参照しているステークホルダー証明書を参照し、依存先のステークホルダー証明書が正当であるかどうかを確認する（ステップＳ１１０４）。
【０１６０】
この結果、正当であると判断されれば、Ｓ１１０６へ処理を移す。そうでない場合、すなわち、トラストモデルがＩｎｄｅｐｅｎｄｅｎｔモデルである、もしくは依存先のステークホルダー証明書が正当でないと判断されれば、エラー処理へと処理を移る（ステップＳ１１３０）。
【０１６１】
図４の例では、ステークホルダー２とステークホルダー１とが依存関係であるかどうかを、ステークホルダー証明書１（１５０）と、ステークホルダー証明書２（２５０）ｔｐを用いて検証する。

【０１６２】
次に、共有鍵制御部１（１１０）は、鍵群１（１３０）の中にマイグレート可能で且つ共有許可な鍵が存在するかどうか探索する（ステップＳ１１０５）。図４の例では、Ｋ１０，Ｋ１１，Ｋ１２，Ｋ１３，Ｋ３１，Ｋ３３，Ｋ３４の中から探索する。
【０１６３】
もし、そのような鍵が存在しなければ、新たにマイグレート可能で且つ共有許可な鍵を生成する（ステップＳ１１０６）。鍵の生成処理は、耐タンパーモジュール１（１２０）で行う。
【０１６４】
そして、共有鍵制御部１（１１０）は、生成した鍵を鍵群１（１３０）の鍵として鍵管理テーブル１（１１３）に登録する（ステップＳ１１０７）。
次に、Ｓ１１０５の探索により見つけたマイグレート可能で共有許可な鍵の中で、子供鍵を有するものがあるかどうかチェックする（ステップＳ１１０８）。もし、そのような鍵がなければ、ステップＳ１１０６へ処理を移し、マイグレート可能で共有許可な鍵の子供の鍵を生成する。図４の例では、Ｋ３１が子供鍵を有し、且つ、マイグレート可能で共有許可な鍵として選択される。
【０１６５】
次に、Ｓ１１０７で子供鍵を有するマイグレート可能で共有許可な鍵として選択された鍵のロード処理を行う（ステップＳ１１０９）。ここでのロード処理とは、親鍵で子供の鍵が暗号化されているので、ロード要求のあった鍵を、ルート鍵であるＫ１０からリーフ方向へ辿り、親子関係を元に復号処理することである。図４の例では、Ｋ３１をロードする。
【０１６６】
次に、共有鍵制御部１（１１０）は、Ｓ１１０９でロードした鍵を耐タンパーモジュール１（１２０）から耐タンパーモジュール２（２２０）にマイグレートするために、耐タンパーモジュール１（１２０）に対してマイグレート処理依頼を行う（ステップＳ１１１０）。以降、マイグレートされる鍵をマイグレート鍵と呼ぶことにする。図４の例では、Ｋ３１がマイグレート鍵であって、Ｋ３３とＫ３４が共有鍵として設定される。
【０１６７】
次に、マイグレート処理依頼を受けた耐タンパーモジュール１（１２０）は、マイグレート鍵をＳ１１０１で受信した公開鍵で暗号化し、共有鍵制御部１（１１０）に返す（ステップＳ１１１１）。図４の例では、Ｋ３１をＫ２０の公開鍵で暗号化する。
【０１６８】
次に、共有鍵制御部１（１１０）は、暗号化マイグレート鍵を共有鍵制御部２（２１０）へ送信する（ステップＳ１１１２）。なお、ここでは、共有鍵制御部１（１１０）及び共有鍵制御部２（２１０）間で直接データの送受信を行うことができるものとして説明しているが、直接でなく、別の第三者の制御部を経由して行うようにしてもよい。
【０１６９】
次に、共有鍵制御部２（２１０）は、耐タンパーモジュール２（２２０）に対して、鍵のマイグレート処理の完了依頼を行う（ステップＳ１１１３）。
次に、耐タンパーモジュール２（２２０）は、暗号化マイグレート鍵を、ルート鍵２（２２１）の秘密鍵で復号する（ステップＳ１１１４）。図４の例では、Ｋ２０の秘密鍵で、暗号化Ｋ３１を復号する。
【０１７０】
次に、耐タンパーモジュール２（２２０）は、平文になったマイグレート鍵の鍵属性情報のステークホルダーフィールド５０５のステークホルダー識別子５０６と鍵利用制限情報５０７とを、ステークホルダー２における鍵利用制限情報に設定する（ステップＳ１１１５）。
【０１７１】
また、Ｓ１１１５にて、リンク情報５０８が、図６（ａ）の親鍵へのリンク情報であれ

ば、ステークホルダーフィールド５０５のリンク情報５０８を、マイグレート先の親鍵に設定し、リンク情報５０８が、図６（ｂ）の子供鍵へのリンク情報であれば変更しない。

【０１７２】

図７の例では、Ｋ３１及びＫ３２の鍵属性情報は、ステークホルダーフィールド５０５のステークホルダー識別子５０６と鍵利用制限情報５０７とリンク情報５０８とが異なり、他は同じ値として設定される。このように設定することで、Ｋ３１がステークホルダー１の環境からのみ利用可能であり、ステークホルダー２の環境からのみ利用可能となる。

【０１７３】

次に、耐タンパーモジュール２（２２０）は、ステップＳ１１１５で設定された鍵を、共有鍵群（３３０）の指定の位置に設定する（Ｓ１１１６）。図４の例では、Ｋ３２を、Ｋ２０の子供として設定している。そのため、Ｋ３２の秘密鍵は、Ｋ２０の公開鍵で暗号化される。

【０１７４】

次に、耐タンパーモジュール２（２２０）から共有鍵制御部２（２１０）を経由し、共有鍵制御部１（１１０）に対して、ステークホルダー２の鍵利用制限情報が送信される（ステップＳ１１１７）。

【０１７５】

次に、共有鍵制御部２（１１０）は、共有鍵として設定される鍵(マイグレート鍵の子供の鍵）の鍵属性情報に、ステークホルダー２の鍵利用制限情報を設定する（ステップＳ１１１８）。

【０１７６】

図７の例では、Ｋ３３及びＫ３４の鍵利用制限情報のステークホルダーフィールド５０５にステークホルダー２の情報が追加される。この処理により、ステークホルダーフィールド５０５には、「ＳＨ１」と「ＳＨ２」との２つのステークホルダーフィールドが設定され、それぞれの鍵利用制限情報としてステークホルダー１の鍵利用制限情報（ＰＣＲ）とステークホルダー２の鍵利用制限情報（ＰＣＲ）とが設定される。このように設定することで、Ｋ３３及びＫ３４は、ステークホルダー１の環境とステークホルダー２の環境とから共用利用が可能な鍵となる。

【０１７７】

次に、共有鍵制御部１（１１０）は、鍵管理テーブル１（１１３）内にあるＳ１１１８で設定した鍵ＩＤと鍵属性情報アドレスとを、共有鍵制御部２（２１０）に送信する（ステップＳ１１１９）。図４の例では、Ｋ３３とＫ３４の鍵ＩＤと鍵属性情報を送る。

【０１７８】

次に、共有鍵制御部２（２１０）は、Ｓ１１１９で受信した鍵ＩＤと鍵属性情報アドレスとを、鍵管理テーブル２（２１２）に登録する。

以上で、共有鍵Ｋ３３及びＫ３４が、共有鍵制御部１（１１０）及び共有鍵制御部２（２１０）から利用可能な状態となる。

【０１７９】

以上で、図１１のフローの説明を終了する。

以上説明したように、本実施の態様によると、共有鍵制御部２（２１０）から共有鍵制御部１（１１０）に、鍵群１（１３０）に含まれる鍵を共有したい旨の通知を受けると、共有鍵制御部２（２１０）に対応するステークホルダー２が共有鍵制御部１（１１０）に対応するステークホルダー１に依存する関係である場合、鍵群１（１３０）に含まれる鍵の中から鍵群２（２３０）にマイグレート可能な所定の鍵Ｋ３１を鍵群２（２３０）の中にマイグレートする。

【０１８０】

即ち、ステークホルダー２がステークホルダー１に依存する関係であることを条件に、所定の鍵Ｋ３１を親鍵とするツリー構造に含まれる鍵群の全体をコピーするのではなく、所定の鍵Ｋ３１のみをコピーし、耐タンパーモジュール１（２１０）及び耐タンパーモジュール２（２２０）で所定の鍵Ｋ３１を親鍵とするツリー構造に含まれる鍵群を共有鍵群

３３０とすることで、所定の鍵Ｋ３１を親鍵とするツリー構造に含まれる鍵群の全体を二重持ちする非効率を回避できる。

【０１８１】

また、共有鍵制御部２（２１０）側で、鍵群２（２３０）に含まれる鍵を用いて所定の鍵Ｋ３１を暗号化して鍵群２（２３０）の中に保持し、鍵群１（１３０）に含まれる所定の鍵Ｋ３１より下層の鍵、例えば、Ｋ３３、Ｋ３４を共用することにより、所定の鍵Ｋ３１をマイグレートするだけで、共有鍵制御部２（２１０）側では、共有鍵制御部１（１１０）に対応する耐タンパーモジュール１（２１０）が管理する鍵群１（１３０）に含まれる所定の鍵Ｋ３１より下層の鍵、例えば、Ｋ３３、Ｋ３４を共用できる。この結果、共有鍵制御部２（２１０）は、共有鍵群３３０あるいは暗号化データ格納部４０で保持された暗号化共有データ（３４０）を簡易な構成で利用できる。

【０１８２】

さらに、共有鍵制御部２（２１０）は、ステークホルター２がステークホルダー１に依存する関係にある場合にのみ、暗号化データ格納部（４０）で保持された暗号化共有データ（３４０）を利用できる。この結果、前記所定のデータを管理する鍵構成を耐タンパーモジュール１及び耐タンパーモジュール２（２２０）で簡易にしつつ、共有鍵群３３０あるいは暗号化共有データ３４０の機密性を保証できる。

【０１８３】

また、共有鍵制御部１（１１０）は、共有鍵制御部２（２１０）に対応するステークホルダー２が、少なくともものステークホルダー１に対応する耐タンパーモジュール１（１２０）を利用するステークホルダーモデルであるとステークホルダー証明書（１５０、２５０）に基づいて判断した場合に、ステークホルダー２がステークホルダー１に依存する関係であると判断する。これにより、ステークホルダー２のステークホルダー１に対する依存関係を確実に判断できるので、前記所定のデータを管理する鍵構成を耐タンパーモジュール１（１２０）及び耐タンパーモジュール２（２２０）で簡易にしつつ、不正なステークホルダーからの共有鍵群３３０あるいは、暗号化共有データ３４０へのアクセスを確実に禁止できる。

【０１８４】

また、鍵のマイグレート処理をする際に、鍵群１（１３０）からマイグレート可能か否かを示す属性情報を参照して、鍵群１（１３０）に含まれる鍵の中から鍵群２（２３０）にマイグレート可能な所定の鍵、例えばＫ３１を探すことにより、鍵属性情報を参照するだけでマイグレート可能な鍵を探せるので、マイグレート可能な鍵を簡易にサーチできる。

【０１８５】

また、共有鍵制御部１（１１０）は、鍵群１（１３０）に含まれる鍵の中から鍵群２（２３０）にマイグレート可能な鍵が存在しない場合、マイグレート可能な鍵を生成して、この生成した鍵を鍵群２（２３０）の中にマイグレートする。この結果、鍵群１（１３０）に含まれる鍵の中から鍵群２（２３０）にマイグレート可能な所定の鍵が存在しない場合であっても、共有鍵制御部２（２１０）は鍵群１（１３０）に含まれる鍵を共用できるので、共有鍵制御部１（１１０）は鍵群１（１３０）に含まれる暗号鍵を用いて暗号化された暗号化データ１（１４０）を暗号化共有データ３４０として暗号化データ格納部で保持し、暗号化共有データ３４０にアクセスできる。

【０１８６】

また、鍵群１（１３０）に含まれる所定の鍵より下層の鍵の位置を示す位置情報を所定の鍵のリンク情報として生成してマイグレートすることにより、共有鍵制御部２（２１０）では、所定の鍵のリンク情報５０８を参照すれば、前記所定の鍵より下層の鍵の位置を確認できるので、前記所定の鍵より下層の鍵を前記耐タンパーモジュール２（２２０）の管理する鍵としてコピーすることなく、耐タンパーモジュール１（１２０）との間で所定の鍵より下層の鍵を共用できる。その結果、耐タンパーモジュール１（１２０）及び耐タンパーモジュール２（２２０）とで前記所定の鍵を親鍵とするツリー構造に含まれる鍵群

の全体を二重持ちする非効率を回避できる。例えば、Ｋ３３とＫ３４を共有鍵とする際に、Ｋ３１のリンク情報５０８を含んだ情報をマイグレートすることで可能となる。例えば、図６（ｂ）のリンク情報５０８を利用することで実現可能である。

【０１８７】

また、鍵群１（１３０）に含まれる所定の鍵の位置情報及び鍵群２（２３０）に含まれる所定の鍵の位置情報を、鍵群１（１３０）に含まれる前記所定の鍵より下層の鍵のリンク情報５０８として生成することにより、その下層の鍵のリンク情報５０８を参照すればその下層の鍵を暗号化した親鍵の所在を認識できるので、前記耐タンパーモジュール１（１２０）及び耐タンパーモジュール２（２２０）とで鍵群１（１３０）に含まれる所定の鍵より下層の鍵を共用する場合であっても、鍵群１（１３０）に含まれる所定の鍵より下層の鍵がどの鍵で暗号化されているかを容易に識別できる。例えば、図６（ａ）のリンク情報５０８を利用することで実現可能である。

【０１８８】

＜図１３：共有鍵利用フロー＞

図１３は、図１２で設定した共有鍵を利用する際のフローであり、第２のステークホルダープログラム（２００）が、共有鍵制御部２（２１０）を介して、共有鍵Ｋ３３あるいはＫ３４を利用する場合のフローである。

【０１８９】

まず、第２のステークホルダープログラム２００は、共有鍵制御部２（２１０）に対して共有鍵による暗号化要求を、共有鍵のＩＤと共に送信する（ステップＳ１３００）。

次に、共有鍵制御部２（２１０）は、Ｓ１３００で受信したＩＤをもとに、鍵管理テーブル２（２１３）から、共有鍵を選択する（ステップＳ１３０１）。

【０１９０】

次に、共有鍵制御部２は、耐タンパーモジュール２（２２０）に、暗号化対象データと、共有鍵の鍵属性情報とを送信する（ステップＳ１３０２）。

次に、耐タンパーモジュール２（２２０）は、鍵属性情報内の利用制限情報であるＰＣＲの情報と、ＰＣＲに記録されている実際の値とを比較し、両者が等しいかどうかチェックする（ステップＳ１３０３）。

【０１９１】

もし、チェックの結果、両者の値が等しいと判断されれば、Ｓ１３０４へ処理を移す。そうでなく、両者の値が等しくないと判断されれば、共有鍵制御部２（２１０）にエラーを返す。エラーを受信した共有鍵制御部２（２１０）は、図示しないエラー処理へ処理を移す。

【０１９２】

次に、耐タンパーモジュール２（２２０）は、共有データを共有鍵で暗号化し、暗号化データを共有鍵制御部２（２１０）へ返す（ステップＳ１３０４、Ｓ１３０５）。

次に、共有鍵制御部２（２１０）は、共有鍵により暗号化された暗号化データを、暗号化データ格納部に書き込む（ステップＳ１３０６）。

【０１９３】

最後に、書き込み完了したことを、共有鍵制御部２（２１０）を通して第２のステークホルダープログラム２００に通知する。

以上で、図１３のフローの説明を終了する。

【０１９４】

なお、実施の形態１では鍵利用制限情報をＰＣＲの情報としているが、鍵利用制限情報は、これに限定されない。例えば、ＴＰＭが有するセキュアなカウンターの値や、生体認証との照合情報などの、各種認証情報であってもよい。また、それら認証情報の組合せたもので鍵の利用を制限させてもよい。

【０１９５】

例えば、カウンターの値とＰＣＲとを組み合わせて、両者の一致した場合のみ共有鍵を利用できるようにしてもよい。また、鍵利用制限情報を、利用許可フラグとし、鍵利用制

限情報が「１」だったら利用可とし、鍵利用制限情報が「０」だったら利用不可とするようにしてもよい。

【０１９６】

＜図１４：共有鍵の無効化の概要フロー＞

図１４は、共有鍵の無効化の概要フローである。ステークホルダー１及びステークホルダー２間で共有されている鍵を、どちらかのステークホルダー環境がリボーク対象であるか、もしくは、ステークホルダー環境に改竄があると判定された場合、リボーク対象もしくは改竄されたステークホルダー側から、共有鍵を利用させないようにするためのフローである。

【０１９７】

まず、共有鍵制御部が端末内のステークホルダー環境がリボーク対象、もしくは、端末内のステークホルダー環境が改竄されたことを判定する（ステップＳ１４０１）。リボーク対象の判断は、リボケーションリストにリボーク対象のＩＤが列挙されているので、そのＩＤを元に判断可能である。また改竄チェックについても、ステークホルダープログラムのハッシュ値を含んだ証明書を情報処理端末に持たせ、静的もしくは動的にプログラムに改竄チェックを行い、改竄があったかどうかを判断することが可能となる。

【０１９８】

次に、共有鍵制御部は、リボーク対象もしくは改竄されたと判定されたステークホルダーが、他のステークホルダーと共有している共有鍵が、存在するかどうかを確認する（ステップＳ１４０２）。もし、存在するなら、ステップＳ１４０３へ処理を移す。そうでなく、存在しないのなら、無効化する共有鍵が存在しないので、無効化処理は終了となる。

【０１９９】

次に、共有鍵制御部が、リボーク対象もしくは改竄されたと判定されたステークホルダー環境から、共有鍵を利用できないように無効化設定する（ステップＳ１４０３）。以降、無効化処理について、例を用いて詳細に説明する

＜図１５：共有鍵の無効化の詳細フロー＞

図１５は、共有鍵の無効化の詳細フローであり、図１４のＳ１４０３を詳細化したフローである。

【０２００】

まず、リボーク対象もしくは改竄されたと判定されたステークホルダーが他のステークホルダーと共有している共有鍵を判断する（ステップＳ１５０１）、この判断処理は、鍵属性情報のステークホルダーフィールド５０５を参照することで可能となる。

【０２０１】

次に、Ｓ１５０１により共有鍵と判断された鍵の親鍵にアクセスし、その鍵で暗号化共有鍵を復号する（ステップＳ１５０２）。

次に、共有鍵制御部は、乱数生成機能を利用し、親鍵の鍵属性情報の鍵値を乱数で更新する（ステップＳ１５０３）。なお、鍵値を乱数で更新するとしているが、これに限定されない。当初の鍵値と異なる値で上書きすれば乱数以外の値でもよい。

【０２０２】

次に、更新した親鍵で、共有鍵を暗号化する（ステップＳ１５０４）。

図１６は、図１５の無効化フローの実施前（無効化前）と実施後（無効化後）の状態を示した例である。

【０２０３】

図１６（ａ）は、無効化前を示したものである。この例では、共有鍵Ｋ３３とＫ３４とが、ステークホルダー１及びステークホルダー２間で共有されている。

図１６（ｂ）は、ステークホルダー２の環境がリボーク対象もしくは改竄ありと判断された場合に、ステークホルダー２の環境から、Ｋ３３及びＫ３４を利用できないように無効化した後の例である。

【０２０４】

Ｋ３１の鍵値を、乱数で書換えて、書換え後の鍵をＫ３５としている。このようにする

と、共有鍵Ｋ３３及びＫ３４は、Ｋ３５で暗号化され、Ｋ３５はＫ１０で暗号化されることになる。そのため、ステークホルダー２側からは、Ｋ３５を復号することができない。その結果、Ｋ３３及びＫ３４も復号することもできない。したがって、Ｋ３３及びＫ３４で暗号化されたデータを、不正なステークホルダー環境による不正利用から保護することが可能となる。

【０２０５】

以上で実施の形態１の説明を終わる。

以上のように、第２のステークホルダープログラム環境（第２のステークホルダープログラム２００、共有鍵制御部２（２１０）、耐タンパーモジュール２（２２０））が外部から攻撃されたことを検知した場合、耐タンパーモジュール１（１２０）は所定の鍵と置換える代替鍵を生成して前記所定の鍵を親鍵とするツリー構造に含まれる鍵を前記代替鍵で再暗号化すると共に前記所定の鍵の親鍵を用いて前記代替鍵を暗号化する。この結果、共有鍵制御部２（２１０）は前記所定の鍵を用いて前記代替鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、前記代替鍵を親鍵とするツリー構造に含まれる鍵で暗号化された所定のデータを利用できず、前記所定データを不正な利用から保護できる。例えば、この場合、所定の鍵がＫ３１であり、代替鍵がＫ３５となる。

【０２０６】

なお、実施の形態１では、共有鍵制御部１（１１０）と共有鍵制御部２（２１０）とは、それぞれ別の制御部として構成しているが、共有鍵制御部１（１１０）と共有鍵制御部２（２１０）と両機能を１つの制御部として実現してもよい。

【０２０７】

これにより、共有鍵制御部１（１１０）と共有鍵制御部２（２１０）とは、共用の共有鍵制御部で構成が可能となり、１つの共有鍵制御部で２つのステークホルダー間の共有鍵を統括的に制御することが可能なので、より柔軟にアクセス制御を行うことが可能となる。

【０２０８】

なお、実施の形態１では、共有鍵の設定と、共有鍵設定後の不正なステークホルダーによる共有鍵の不正利用を防ぐための共有鍵の無効化について説明しているが、無効化後の鍵に対して、再度共有鍵として設定するようにしてもよい。

【０２０９】

例えば、不正ステークホルダーが、ステークホルダー管理サーバーを利用して更新処理したことにより、正規のステークホルダー環境になった場合、無効化されていた共有鍵を、再度共有鍵として設定してもよい。

【０２１０】

より具体的には、ステークホルダー１とステークホルダー２との間で共用できる共有鍵があり、ステークホルダー２が不正と判断されると、ステークホルダー２からその共有鍵が利用できないように無効化される。

【０２１１】

その後、不正なステークホルダー２の環境が、第２のステークホルダー管理サーバーから更新モジュールをダウンロードし、不正なステークホルダー２の環境を更新し、更新の結果、正規ステークホルダー環境となれば、無効化されていた共有鍵をステークホルダー２から再度利用できるように復活させる。

【０２１２】

再度共有鍵として設定する処理は、実施の形態１で説明した共有鍵の設定処理を、更新されたステークホルダー２に対して行うことで実現できるので説明は省略する。

（実施の形態２）

実施の形態２では、実施の形態１とは異なる方法で、共有鍵の無効化をする例である。実施の形態１と同じ構成やフローである部分について説明を省略し、実施の形態２に特有の処理について図面を用いて説明する。

【０２１３】

＜図１７：無効化詳細フロー＞

図１７は、共有鍵の無効化の詳細フローであり、図１４のＳ１４０３を詳細化したフローである。また、図１８は、図１７の無効化フローの実施前（無効化前）と実施後（無効化後）とを示したものでる。以降、図１７及び図１８を用いて説明する。

【０２１４】

ステップＳ１７０１とステップＳ１７０２とは、実施の形態１のＳ１５０１とＳ１５０２と同じであるので説明を省略する。

次に、Ｓ１７０２における共有鍵群（３３０）以外から、共有鍵の親鍵とは異なるマイグレート可能で共有許可である鍵をツリー内から選択する。もし、そのような鍵が存在しなければ、新たに鍵を生成する。（ステップＳ１７０３）。

【０２１５】

後述するように、本実施の形態では、選択した鍵を共有鍵の新たな親鍵とすることで、ステークホルダー２から共有鍵を利用できないようにする。この目的を達成するためのみならば、マイグレートの可否や共有許可の有無とは関係なく共有鍵の親鍵と異なる鍵を選べばよい。しかし、ここでは、さらにマイグレートが可能であり、共有許可である鍵を選択している。

【０２１６】

なぜ、このような鍵を選択するかについて、次の２つのケースを用いて説明する。

まず１つ目のケースは、不正なステークホルダー環境と判断されたことにより共有鍵が無効化された不正なステークホルダーが、更新処理により、正規のステークホルダー環境になった場合、再度共有鍵を利用できるように復活させるケースである。再度共有鍵を利用できるようにするには、共有鍵の親鍵をマイグレートする必要があるため、その場合に備えてマイグレートが可であり共有許可のある鍵を新たな親鍵としておく必要がある。

【０２１７】

２つ目のケースは、３つ以上のステークホルダーで共有されている共有鍵があるケースである。例えば、ステークホルダー１，２，３で共有されている共有鍵に対し、ステークホルダー２に対してのみ無効化する場合、ステークホルダー１とステークホルダー３との間では、共有鍵を共有したままの設定にする必要がある。共有鍵の親鍵を変更すると、ステークホルダー３も共有鍵を利用することができなくなってしまう。そのため、共有鍵がステークホルダー３と共有されたままの状態を維持するために、共有鍵の親鍵の変更後、ステークホルダー１から３へ、共有鍵の親鍵をマイグレートする処理が必要となる。ステークホルダー１から３へ共有鍵の親鍵がマイグレートされれば、両ステークホルダーで共有鍵の利用が可能となる。

【０２１８】

次に、Ｓ１７０３で選択された鍵を、共有鍵の親鍵として設定するので、Ｓ１７０３で選択された鍵と共有鍵とが親子関係になるように鍵属性情報のリンク情報５０８を更新する（ステップＳ１７０４）。

【０２１９】

次に、Ｓ１７０３で選択された鍵で、共有鍵を暗号化する（ステップＳ１７０５）。

最後に、鍵管理テーブルを更新する（ステップＳ１７０６）。

図１８は、図１７の無効化フローの実施前（無効化前）と実施後（無効化後）との状態を示した例である。

【０２２０】

図１８（ａ）は、無効化前を示したものである。この例では、共有鍵Ｋ３３とＫ３４とがステークホルダー１及びステークホルダー２間で共有されている。

図１８（ｂ）は、ステークホルダー２の環境がリボーク対象もしくは改竄ありと判断された場合に、ステークホルダー２の環境からはＫ３３とＫ３４とを利用できないように無効化した後の例である。

【０２２１】

Ｋ３３及びＫ３４の親鍵を、Ｋ３１ではないマイグレート可能かつ共有可能なＫ１１と

している。このようにすると、Ｋ３３とＫ３４とがＫ１１で暗号化され、Ｋ１１はＫ１０で暗号化されることになる。そのため、ステークホルダー２側からは、Ｋ１１を復号することができず、その結果、Ｋ３３とＫ３４とも復号することもできない。したがって、Ｋ３３とＫ３４とで暗号化されたデータを、不正なステークホルダー環境による不正利用から保護することが可能となる。

【０２２２】

以上で実施の形態２の説明を終わる。

以上のように、第２のステークホルダープログラム環境（第２のステークホルダープログラム２００、共有鍵制御部２（２１０）、耐タンパーモジュール２（２２０））が外部から攻撃されたことを検知した場合、耐タンパーモジュール１（１２０）は前記所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵を用いて所定のデータを暗号化し直す。この結果、共有鍵制御部２（２１０）は所定の鍵を親鍵とするツリー構造に含まれる鍵を用いては前記所定のデータを復号化できないので、所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵で暗号化された所定のデータを利用できず、所定データを不正な利用から保護できる。例えば、この場合、所定の鍵が、Ｋ３１であり、所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵がＫ１１として実現可能である。

【０２２３】

なお、実施の形態２では、実施の形態１と同様、不正なステークホルダー環境が、更新などにより正当なステークホルダー環境になった場合に、共有鍵を再度共有させることも可能である。再度共有鍵として設定する処理は、実施の形態２で説明した共有鍵の設定処理を、更新されたステークホルダー２に対して行うことで実現できるので説明は省略する。

【０２２４】

（実施の形態３）

本実施の形態３では、実施の形態１と実施の形態２とは異なる方法で、共有鍵の無効化をする例である。実施の形態１と実施の形態２と同じ構成やフローである部分について説明を省略し、実施の形態３に特有の処理について図面を用いて説明する。

【０２２５】

＜図１９：無効化詳細フロー＞

図１９は、共有鍵の無効化の詳細フローであり、図１４のＳ１４０３を詳細化したフローである。

【０２２６】

ステップＳ１９０１とステップＳ１９０２とは、Ｓ１５０１とＳ１５０２と同じであるので説明を省略する。

次に、Ｓ１９０１によりリボーク対象もしくは改竄されたと判定されたステークホルダーが他のステークホルダーと共有していると判断された共有鍵にアクセスし、リボーク対象もしくは改竄されたステークホルダーに対応する鍵利用制限情報にアクセスする（ステップＳ１９０２）。

【０２２７】

次に、共有鍵制御部は、乱数生成機能を利用し、Ｓ１９０２で選択された共有鍵に対応する鍵属性情報の鍵利用制限情報を、乱数で更新する（ステップＳ１５０３）。なお、鍵利用制限情報を乱数で更新するとしているが、これに限定されない。

【０２２８】

本実施の形態によると、第１ステークホルダーが管理する第１のステークホルダー環境は、第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、共有鍵制御部１（１１０）は共有鍵で暗号化されたデータを、共有鍵制御部２（２１０）から利用できないように、鍵属性情報の鍵利用制限情報５０８を書き換える。この結果、共有鍵制御部２（２２０）は共有鍵の鍵ロード処理が出来ないので、共有鍵で暗号化されたデータを不正な利用から保護できる。

【０２２９】

なお、共有鍵の鍵利用制限情報だけを更新するとしているが、共有鍵の親鍵の鍵利用制限情報を更新してもよい。これは、共有鍵が複数に階層に渡って構成されている場合、共有鍵からルート方向に辿り共有鍵群（３３０）内の最上位層の鍵だけを無効化することで、それより子供の鍵は無効化対象とすることが可能となるからである。

【０２３０】

以上のように、共有鍵制御部２（２１０）は、改竄のない第２のステークホルダー環境から生成された鍵利用制限情報５０７と第２のステークホルダープログラム（２００）から実際に得られたＰＣＲ（２２３）に記録されている環境情報とを比較し、比較結果が正しい場合にのみ前記鍵を利用させる。

【０２３１】

この結果、第２のステークホルダープログラム（２００）が改竄され若しくはリボークされた場合には前記比較結果は不一致となって、共有鍵制御部２（２１０）は所定の鍵を用いて、所定の鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、前記所定の鍵を親鍵とするツリー構造に含まれる鍵用いて暗号化されている所定のデータを復号化できず、所定データを不正な利用から保護できる。

【０２３２】

例えば、第２のステークホルダープログラム（２００）が改竄されていれば、共有鍵制御部２（２１０）からＫ３３のロードができないため、Ｋ３３で暗号化されたデータＤ３４の不正な利用から保護できる。

【０２３３】

また、第１ステークホルダーが管理する第１のステークホルダー環境は、第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、共有鍵制御部１（１１０）は、所定の鍵を親鍵とするツリー構造に含まれる鍵を、共有鍵制御部２（２１０）から利用できないように、鍵利用制限情報を書き換える。

【０２３４】

この結果、共有鍵制御部２（２１０）は所定の鍵を用いては所定の鍵を親鍵とするツリー構造に含まれる鍵を復号化できないので、所定の鍵を親鍵とするツリー構造に含まれる鍵用いて暗号化されている前記所定のデータを復号化できず、前記所定データを不正な利用から保護できる。この場合、例えば、所定の鍵がＫ３２で、所定の鍵を親鍵とするツリー構造に含まれる鍵がＫ３３，Ｋ３４であると、Ｋ３２の利用制限情報を書き換える、もしくはＫ３３，Ｋ３４の鍵利用制限情報を書き換えることで実現できる。

【０２３５】

なお、実施の形態１及び実施の形態２と同様、不正なステークホルダー環境が、更新などにより正当なステークホルダー環境になった場合に、共有鍵を再度共有させることも可能である。再度共有させる処理は、更新されたステークホルダーについての鍵利用制限情報を、更新後のステークホルダーの鍵利用制限情報（ＰＣＲ値）に更新すればよい。

【０２３６】

＜図２３：再共有化＞

図２３は、実施の形態３における不正なステークホルダー環境が、更新などにより正当なステークホルダー環境になった場合に、共有鍵を再度共有させるフローを示している。ここでは、ステークホルダー２の環境が、更新などにより正当なステークホルダー環境になり、共有が無効化されていたステークホルダー１の環境との共有鍵を、再度共有化する例で説明する。

【０２３７】

まず、ステークホルダー１の環境は、不正と判断されていたステークホルダー２の環境が正当な環境に更新されたと判断する（ステップＳ２３０１）。この判断は、例えば、周期的に行われたり、電源が投入されたときや、ステークホルダー２の環境の更新が完了したときに行われる。

【０２３８】

次に、ステークホルダー１の環境の、共有鍵制御部１（１１０）は、共有鍵群（３３０

）内の鍵属性情報を参照し、ステークホルダーフィールド５０５のステークホルダー識別子５０６にステークホルダー２の識別子が登録されている鍵を選択する（ステップＳ２３０２）。

【０２３９】

最後に、Ｓ２３０３で選択した鍵の鍵属性情報内の鍵利用制限情報を正当になったステークホルダーの環境情報に更新する（ステップＳ２３０３）。この例では、鍵属性情報内のステークホルダー２の識別子に対応する鍵利用制限情報が、更新などにより正当になった後のステークホルダー２の環境情報に更新される。

【０２４０】

もし、ステークホルダー２と共有すべき鍵が、複数存在するのであれば、Ｓ２３０２とＳ２３０３とを繰り返し処理すればよい。

このようにすることで、ステークホルダー２の環境は、共有が無効化されていたステークホルダー１と共有する共有鍵を、再度共有鍵として利用可能となる。

【０２４１】

（実施の形態４）

本実施の形態４では、実施の形態１、実施の形態２、及び実施の形態３とは異なる方法で、暗号化共有データを保護するする例である。他の実施の形態と同じ構成やフローである部分について説明を省略し、実施の形態３に特有の処理について図面を用いて説明する。

【０２４２】

＜図２０：暗号化共有データの構造＞

図２０は、暗号化共有データの構造を示した図である。暗号化共有データは、暗号化データサイズ２００１と、暗号化データ２００２と、ステークホルダーフィールド２００３とを備える。

【０２４３】

ステークホルダーフィールド２００３には、共有データにアクセス権のあるステークホルダーの情報が列挙される。図４に示した暗号化共有データＤ３３及びＤ３４は、ステークホルダー１とステークホルダー２とからアクセスされる例であるので、ステークホルダーフィールド５０５は、２つ存在することになる。

【０２４４】

ステークホルダーフィールド２００３は、ステークホルダー識別子２００４と、鍵の利用制限を示す利用制限情報２００５とを含む。

利用制限情報２００５は、期待されるＰＣＲ値であり、耐タンパーモジュールの備えるＰＣＲ（１２３、２２３）に記録されている実際の値と比較され、実際のＰＣＲ値と期待されるＰＣＲ値が等しかった場合にのみ耐タンパーモジュールから復号結果が得られるように制限するための情報である。

【０２４５】

＜図２１：暗号化共有データの復号フロー＞

図２１は、共有鍵制御部２（２１０）の暗号化共有データ復号処理のフローを示した図である。

【０２４６】

まず、共有鍵制御部２（２１０）は、第２のステークホルダープログラム（２００）から暗号化共有データの復号要求を受け、暗号化データ格納部４０から復号対象となる暗号化共有データ３４０をリードする（ステップＳ２１０１）。

【０２４７】

次に、共有鍵制御部２（２１０）は、鍵管理テーブル２（２１３）から、共有鍵を選択する（ステップＳ２１０２）。

次に、共有鍵制御部２（２１０）は、Ｓ２１０２で選択した共有鍵と、Ｓ２１０１でリードした暗号化共有データとをパラメータとし、耐タンパーモジュール２（２２０）に復号処理の要求をする（ステップＳ２１０３）。

【０２４８】
次に、Ｓ２１０３で指定された共有鍵を鍵ツリーのルート鍵２からリーフ方向へと、親子関係を元に鍵を復号していき、平文の共有鍵を得て、暗号化共有データを復号する（ステップＳ２１０４）。
【０２４９】
次に、耐タンパーモジュール２（２２０）は、復号要求のあった暗号化共有データの利用制限情報として設定されている期待ＰＣＲ値と、実際のＰＣＲ（２２３）に記録されている値とを比較する（ステップＳ２１０５）。比較の結果、両者が等しければ、復号データを共有鍵制御部２（２１０）へ返す（ステップＳ２１０６）。比較の結果、両者が等しくないと判定されれば、復号データは、共有鍵制御部２（２１０）へ返されず、エラー通知のみを返す。
【０２５０】
＜図２２：無効化＞
図２２は、暗号化共有データの復号処理を無効化するための詳細フローであり、実施の形態４における図１４のＳ１４０３を詳細化したフローである。
【０２５１】
まず、リボーク対象もしくは改竄されたと判定されたステークホルダーが、他のステークホルダーと共有している共有データを判断する（ステップＳ２２０１）。この判断処理は、暗号化共有データのデータ構造のステークホルダーフィールド２００３を参照することで可能となる。
【０２５２】
次に、Ｓ２２０１で判断された暗号化共有データの利用制限情報にアクセスする（ステップＳ２２０２）。
次に、共有鍵制御部は、乱数生成機能を利用し、Ｓ２２０２でアクセスされた利用制限情報を、乱数で更新する（ステップＳ１５０３）。なお、利用制限情報を乱数で更新するとしているが、これに限定されない。
【０２５３】
このようにすると、リボーク対象もしくは改竄のある不正なステークホルダー環境から暗号化共有データの復号要求があったとしても、Ｓ２１０５の比較結果は不一致となって、暗号化共有データの復号データを得ることができず、暗号化共有データの不正利用から保護できる。
【０２５４】
以上説明したように、共有鍵制御部２（２２０）は、改竄のない第２のステークホルダー環境から生成された暗号化データ利用制限情報と第２のステークホルダー環境から実際に得られた耐タンパーモジュール２（２２０）内のＰＣＲ（２２３）に記録されている環境情報とを比較し、比較結果が正しい場合にのみ暗号化データを復号させる。
【０２５５】
この結果、第２ステークホルダー環境が改竄され若しくはリボークされた場合には前記比較結果は不一致となって、共有鍵制御部２（２１０）は所定の鍵を用いて、所定の鍵を親鍵とするツリー構造に含まれる鍵を用いて暗号化されている暗号化データを復号化できず、暗号化データを不正な利用から保護できる。
【０２５６】
例えば、第２のステークホルダープログラム（２００）が改竄されていれば、耐タンパーモジュール２（２２０）から、Ｋ３３で暗号化されたＤ３３の復号データを得ようとしても、復号結果が得られないので、Ｄ３４の不正な利用から保護できる。
【０２５７】
また、前記第１ステークホルダーが管理する第１のステークホルダー環境は、第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、共有鍵制御部１（１１０）は、所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データを、共有鍵制御部２（２１０）から利用できないように、暗号化データ利用制

限情報２００５を書き換える。

【０２５８】

この結果、共有鍵制御部２（２１０）は所定の鍵を用いて所定の鍵を親鍵とするツリー構造に含まれる鍵を用いた復号処理ができないので、前記所定データを不正な利用から保護できる

なお、実施の形態１から実施の形態４は、組み合わせて実現されてもよい。

【０２５９】

また、実施の形態１から実施の形態４は、耐タンパーモジュール（１５０，２５０）を、ＴＰＭあるいはＭＴＭを用いて実現する形態としている。そのためＴＣＧで規定されるＴｒｕｓｔｅｄ　Ｂｏｏｔでもよいし、ＴＣＧ　Ｍｏｂｉｌｅ仕様で規定されるＳｅｃｕｒｅ　Ｂｏｏｔでもよい。また、実行されるプログラムの完全性を検証できる仕組みであればよい。

【０２６０】

これにより、前記第１共有鍵制御部はセキュアブートする際に、前記第２共有鍵制御部が外部から攻撃されたことを検知することにより、前記第２共有鍵制御部の外部からの攻撃を判断できる。

【０２６１】

また、ステークホルダー環境の改竄チェックは、ＴＣＧで規定されているＡｔｔｅｓｔａｔｉｏｎ機能を利用してもよい。Ａｔｔｅｓｔａｔｉｏｎ機能のサーバー側の判定結果を情報処理端末１０に送付し、ステークホルダーの改竄を検知してもよいし、上述したようにサーバーから配信されるリボケーションリスト１４を利用してもよい。

【０２６２】

これにより、共有鍵制御部１（１１０）は、外部のサーバーから、共有鍵制御部２（２１０）の改竄の検知、もしくはリボーク対象である旨の通知を受けることにより、不正な共有鍵制御部２（２１０）から共有鍵や暗号化共有データの不正利用を防止することができる。

【０２６３】

なお、不正なステークホルダー環境が、更新などにより正当なステークホルダー環境になった場合に、暗号化共有データを再度共有させることも可能である。再度共有させる処理は、暗号化共有データの、更新されたステークホルダーについての鍵利用制限情報を、更新後のステークホルダーの鍵利用制限情報（ＰＣＲ値）に更新すればよい。

【０２６４】

なお、実施の形態４における共有鍵の再共有化のフローは、図２３における処理のＳ２２０２とＳ２２０３で参照する情報が、図５の鍵属性情報でなく、図２０の共有鍵暗号化データになり、更新する情報が、鍵利用制限情報５０８でなく、鍵利用制限情報２００５になるだけなので、説明を省略する。

【０２６５】

なお、実施の形態１から実施の形態４では、共有鍵の再共有化の処理を、不正と判断されていたステークホルダー以外のステークホルダー環境が、再共有化処理のトリガーとなっているが、不正なステークホルダー自身が、更新後、自ら自身が正当であることを検証してから、共有先であるステークホルダーに対して再度共有化の依頼を行うようにしてもよい。

（その他変形例）

なお、本発明を上記実施の形態に基づいて説明してきたが、本発明は、上記の実施の形態に限定されないのはもちろんである。以下のような場合も本発明に含まれる。

【０２６６】

（１）上記の各装置は、具体的には、マイクロプロセッサ、ＲＯＭ、ＲＡＭ、ハードディスクユニット、ディスプレイユニット、キーボード、マウスなどから構成されるコンピュータシステムである。前記ＲＡＭまたはハードディスクユニットには、コンピュータプログラムが記憶されている。前記マイクロプロセッサが、前記コンピュータプログラムに

したがって動作することにより、各装置は、その機能を達成する。ここでコンピュータプログラムは、所定の機能を達成するために、コンピュータに対する指令を示す命令コードが複数個組み合わされて構成されたものである。また、各装置は、マイクロプロセッサ、ＲＯＭ、ＲＡＭ、ハードディスクユニット、ディスプレイユニット、キーボード、マウスなどの全てを含むものではなく、これらの一部から構成されているとしてもよい。

【０２６７】

（２）上記の各装置を構成する構成要素の一部または全部は、１個のシステムＬＳＩ（Ｌａｒｇｅ　Ｓｃａｌｅ　Ｉｎｔｅｇｒａｔｉｏｎ：大規模集積回路）から構成されているとしてもよい。システムＬＳＩは、複数の構成部を１個のチップ上に集積して製造された超多機能ＬＳＩであり、具体的には、マイクロプロセッサ、ＲＯＭ、ＲＡＭなどを含んで構成されるコンピュータシステムである。前記ＲＡＭには、コンピュータプログラムが記憶されている。前記マイクロプロセッサが、前記コンピュータプログラムにしたがって動作することにより、システムＬＳＩは、その機能を達成する。

【０２６８】

また、上記の各装置を構成する構成要素の各部は、個別に１チップ化されていても良いし、一部又は全てを含むように１チップ化されてもよい。

また、ここでは、システムＬＳＩとしたが、集積度の違いにより、ＩＣ、ＬＳＩ、スーパーＬＳＩ、ウルトラＬＳＩと呼称されることもある。また、集積回路化の手法はＬＳＩに限るものではなく、専用回路又は汎用プロセッサで実現してもよい。ＬＳＩ製造後に、プログラムすることが可能なＦＰＧＡ（Ｆｉｅｌｄ　Ｐｒｏｇｒａｍｍａｂｌｅ　Ｇａｔｅ　Ａｒｒａｙ）や、ＬＳＩ内部の回路セルの接続や設定を再構成可能なリコンフィギュラブル・プロセッサーを利用しても良い。

【０２６９】

さらには、半導体技術の進歩又は派生する別技術によりＬＳＩに置き換わる集積回路化の技術が登場すれば、当然、その技術を用いて機能ブロックの集積化を行ってもよい。バイオ技術の適用等が可能性としてありえる。

【０２７０】

（３）上記の各装置を構成する構成要素の一部または全部は、各装置に脱着可能なＩＣカードまたは単体のモジュールから構成されているとしてもよい。前記ＩＣカードまたは前記モジュールは、マイクロプロセッサ、ＲＯＭ、ＲＡＭなどから構成されるコンピュータシステムである。前記ＩＣカードまたは前記モジュールは、上記の超多機能ＬＳＩを含むとしてもよい。マイクロプロセッサが、コンピュータプログラムにしたがって動作することにより、前記ＩＣカードまたは前記モジュールは、その機能を達成する。このＩＣカードまたはこのモジュールは、耐タンパ性を有するとしてもよい。

【０２７１】

（４）本発明は、上記に示す方法であるとしてもよい。また、これらの方法をコンピュータにより実現するコンピュータプログラムであるとしてもよいし、前記コンピュータプログラムからなるデジタル信号であるとしてもよい。

【０２７２】

また、本発明は、前記コンピュータプログラムまたは前記デジタル信号をコンピュータ読み取り可能な記録媒体、例えば、フレキシブルディスク、ハードディスク、ＣＤ－ＲＯＭ、ＭＯ、ＤＶＤ、ＤＶＤ－ＲＯＭ、ＤＶＤ－ＲＡＭ、ＢＤ（Ｂｌｕ－ｒａｙ　Ｄｉｓｃ）、半導体メモリなどに記録したものとしてもよい。また、これらの記録媒体に記録されている前記デジタル信号であるとしてもよい。

【０２７３】

また、本発明は、前記コンピュータプログラムまたは前記デジタル信号を、電気通信回線、無線または有線通信回線、インターネットを代表とするネットワーク、データ放送等を経由して伝送するものとしてもよい。

【０２７４】

また、本発明は、マイクロプロセッサとメモリを備えたコンピュータシステムであって

、前記メモリは、上記コンピュータプログラムを記憶しており、前記マイクロプロセッサは、前記コンピュータプログラムにしたがって動作するとしてもよい。

【０２７５】

また、前記プログラムまたは前記デジタル信号を前記記録媒体に記録して移送することにより、または前記プログラムまたは前記デジタル信号を前記ネットワーク等を経由して移送することにより、独立した他のコンピュータシステムにより実施するとしてもよい。

【０２７６】

（５）上記実施の形態及び上記変形例をそれぞれ組み合わせるとしてもよい。

【産業上の利用可能性】

【０２７７】

本発明は、例えばセキュアなデータを扱う情報処理装置を製造及び販売する産業において、複数のステークホルダー間において依存関係を持たせた証明書を用いて、依存関係に従った形で鍵を共有し、複数のステークホルダー間で、セキュアな共有データを効率的に暗号化する仕組みとして利用することができる。また、本発明は、不正なステークホルダーから共有鍵へのアクセスを制限させる仕組みとして利用することができる。

【符号の説明】

【０２７８】

１０　情報処理端末
１１　第１のステークホルダー管理サーバー
１２　認証ＰＣＲデータベース
１３　証明書データベース
１４　リボケーションリスト
１５　ネットワーク
１６　第２のステークホルダー管理サーバー
２１、２２、３１、３２、４１，４２　ステークホルダー
２３、２４、３３、３４、４３　ＴＰＭ
１００、２００　ステークホルダープグラム
１１０、２１０　共有鍵制御部
１１１、２１１　マルチステークホルダー判定部
１１２、２１２　共有許可設定部
１１３、２１３　鍵管理テーブル
１２０、２２０　耐タンパーモジュール
１２１、２２１　ルート鍵
３０　鍵格納部
４０　暗号化データ格納部
５０　証明書格納部
１３０　鍵群１
２３０　鍵群２
３３０　共有鍵群
１４０、２４０　暗号化データ
３４０　暗号化共有データ
１５０、２５０　ステークホルダー証明書
１２２、２２１　セキュアメモリ
４１０、４１１、４１２、４１３、４２０、４２１、４２２、４２３、４３１、４３２、４３４　鍵属性情報
１２３、２２３　ＰＣＲ
５０１　マイグレート許可フラグ
５０２　共有許可フラグ
５０３　暗号アルゴリズム・鍵サイズ
５０４　鍵値

５０５、２００３　ステークホルダーフィールド
５０６、２００４　ステークホルダー識別子
５０７、２００５　鍵利用制限情報
５０８、２００６　リンク情報
６０１、６０２、６０３、６１１、６１２、６１３　鍵ポインタ
８１１、８２１　鍵ＩＤ
８１２、８２２　鍵属性情報アドレス
９０１　証明書バージョン
９０２　シリアルナンバー
９０３　署名アルゴリズム
９０４　発行者情報
９０５　有効期間
９０６　サブジェクト
９０７　公開鍵情報
９０８　ＴＰＭバージョン
９０９　トラストモデル識別情報
９１０　依存ステークホルダー証明書識別情報
９１１　拡張領域
９１２　署名データ
２００１　暗号化データサイズ
２００２　暗号化データ

【手続補正２】
【補正対象書類名】特許請求の範囲
【補正対象項目名】全文
【補正方法】変更
【補正の内容】
【特許請求の範囲】
【請求項１】
第１のステークホルダーに対応する第１共有鍵制御部と、
第２のステークホルダーに対応する第２共有鍵制御部と、
前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、
前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデータ保持部と、
前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備し、
前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、
前記第１共有鍵制御部は、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断し、前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーし、
前記第２共有鍵制御部は、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することを特徴とする情報処理装置。
【請求項２】
前記第２共有鍵制御部は、前記第１共有鍵制御部との間の依存関係を証明した証明書を有し、前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を

送付する際、前記証明書を送付し、
前記第１共有鍵制御部は、前記証明書に基づいて、前記第２共有鍵制御部に対応する第２のステークホルダーが、少なくとも前記第１のステークホルダーに対応する第１の耐タンパーモジュールを利用するステークホルダーモデルであると判断した場合に、
前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であると判断することを特徴とする請求項１記載の情報処理装置。
【請求項３】
前記第１の暗号鍵群に含まれる各鍵は、当該鍵が前記第１の暗号鍵群からコピー可能か否かを示す属性情報を有し、
前記第１共有鍵制御部は、前記属性情報を参照して、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探すことを特徴とする請求項２記載の情報処理装置。
【請求項４】
前記第１共有鍵制御部は、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵が存在しない場合、コピー可能な鍵を生成して、この生成した鍵を前記第２の暗号鍵群の中にコピーすることを特徴とする請求項１記載の情報処理装置。
【請求項５】
前記第１共有鍵制御部は、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵の位置を示す位置情報を前記所定の鍵のリンク情報として生成し、このリンク情報と共に前記所定の鍵を前記第２の暗号鍵群の中にコピーすることを特徴とする請求項３記載の情報処理装置。
【請求項６】
前記第１共有鍵制御部は、前記第１の暗号鍵群に含まれる所定の鍵の位置情報及び前記第２の暗号鍵群に含まれる所定の鍵の位置情報を、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵のリンク情報として生成することを特徴とする請求項５記載の情報処理装置。
【請求項７】
前記第１共有鍵制御部と前第２共有鍵制御部とは、共用の共有鍵制御部であることを特徴とする請求項１記載の情報処理装置。
【請求項８】
前記第１のステークホルダーが管理する第１ステークホルダー環境は、前記第２のステークホルダーが管理する第２のステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１共有鍵制御部は、前記所定の鍵と置換える代替鍵を前記第１の耐タンパーモジュールに生成させ、前記所定の鍵を親鍵とするツリー構造に含まれる鍵を前記代替鍵で再暗号化させると共に前記前記所定の鍵の親鍵を用いて前記代替鍵を暗号化させて、前記第２共有鍵制御部による前記所定のデータの利用を排除することを特徴とする請求項１記載の情報処理装置。
【請求項９】
前記第１のステークホルダーが管理する第１ステークホルダー環境は、前記第２のステークホルダーが管理する第２のステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵以外の鍵を用いて前記第１の耐タンパーモジュールに前記所定のデータを暗号化し直させて、前記第２共有鍵制御部による前記所定の鍵の使用を排除することを特徴とする請求項１記載の情報処理装置。
【請求項１０】
前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵は、属性情報として、前記改竄のない第２のステークホルダーが管理する第２ステークホルダー環境のハッシュ値から生成された期待値として鍵利用制限情報を有し、

第２の耐タンパーモジュールは、前記第２のステークホルダー環境のハッシュ値から生成された実際の値としての環境情報を記憶し、

第２の共有鍵制御部から前記第１の共有鍵制御部に対して前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵の利用を依頼するときに、第２の共有鍵制御部は、前記鍵利用制限情報と前記環境情報とを比較し、比較結果が正しい場合にのみ前記鍵を利用させるように制限をすることを特徴する請求項１記載の情報処理装置。

【請求項１１】

前記第１ステークホルダーが管理する第１のステークホルダー環境は、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵を、第２の共有鍵制御部から利用できないように、前記鍵利用制限情報を書き換えることを特徴とする請求項１０の情報処理端末。

【請求項１２】

前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データは、属性情報として、前記改竄のない第２のステークホルダーが管理する第２ステークホルダー環境のハッシュ値から生成された期待値である暗号化データ利用制限情報を有し、

第２の耐タンパーモジュールは、前記第２のステークホルダー環境のハッシュ値から生成された実際の値としての環境情報を記憶し、

第２の共有鍵制御部から前記第１の共有鍵制御部に対して前記第１の暗号化鍵群に含まれる前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化されたデータの復号処理を依頼するときに、第２の共有鍵制御部は、前記暗号化データ利用制限情報と前記環境情報とを比較し、比較結果が正しい場合にのみ前記暗号化データの復号処理させるように制限をすることを特徴する請求項１記載の情報処理装置。

【請求項１３】

前記第１ステークホルダーが管理する第１のステークホルダー環境は、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象であることを検知した場合、前記第１の共有鍵制御部は、前記所定の鍵を親鍵とするツリー構造に含まれる鍵で暗号化された暗号化データを、第２の共有鍵制御部から利用できないように、前記暗号化データ利用制限情報を書き換えることを特徴とする請求項１２の情報処理端末。

【請求項１４】

前記第１共有鍵制御部は、第１ステークホルダー環境および第２ステークホルダー環境を環境に対して完全性をチェックしてから改竄されていない環境のみを起動する機能であるセキュアブートによってブートする際に、第２ステークホルダー環境が改竄された、もしくはリボーク対象であることを検知することを特徴とする請求項８乃至請求項１３のいずれかに記載の情報処理装置。

【請求項１５】

前記第１共有鍵制御部は、外部のサーバーから、前記第２ステークホルダー環境が改竄されたもしくはリボーク対象である旨の通知を受けることで前記第第２ステークホルダー環境が改竄された、もしくはリボーク対象であることを検知することを特徴とする請求項８乃至請求項１３のいずれかに記載の情報処理装置。

【請求項１６】

第１のステークホルダーに対応する第１共有鍵制御部と、

第２のステークホルダーに対応する第２共有鍵制御部と、

前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、

前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、

前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構

造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置における暗号鍵の鍵管理方法であって、
前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、
前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断し、
前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーし、
前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用することを特徴とする暗号鍵の管理方法。
【請求項１７】
第１のステークホルダーに対応する第１共有鍵制御部と、
第２のステークホルダーに対応する第２共有鍵制御部と、
前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、
前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、
前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置における暗号鍵の鍵管理に用いるコンピュータプログラムであって、
コンピュータに対して、
前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、
前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断する処理と、
前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーする処理と、を実行させ、
前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用する処理を実行させることを特徴とするコンピュータプログラム。
【請求項１８】
第１のステークホルダーに対応する第１共有鍵制御部と、
第２のステークホルダーに対応する第２共有鍵制御部と、
前記第１のステークホルダーに対応し、複数の暗号鍵を含む第１の暗号鍵群をツリー構造で管理する第１の耐タンパーモジュールと、
前記第１の暗号鍵群に含まれる暗号鍵を用いて暗号化された所定のデータを保持するデー保持部と、
前記第２のステークホルダーに対応し、複数の暗号鍵を含む第２の暗号鍵群をツリー構造で管理する第２の耐タンパーモジュールと、を具備する情報処理装置に用いる集積回路であって、
情報処理部と、
この情報処理部に対して、
前記第１共有鍵制御部において、前記第２共有鍵制御部から前記第１共有鍵制御部に前記第１の暗号鍵群に含まれる鍵を共有したい旨の通知を受けると、

前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係であるか否かを判断する処理と、

前記第２共有鍵制御部に対応する第２のステークホルダーが前記第１共有鍵制御部に対応する第１のステークホルダーに依存する関係である場合、前記第１の暗号鍵群に含まれる鍵の中から前記第２の暗号鍵群にコピー可能な所定の鍵を探して、この所定の鍵を前記第２の暗号鍵群の中にコピーする処理と、を実行させ、

前記第２共有鍵制御部において、前記第２の暗号鍵に含まれる鍵を用いて前記所定の鍵を暗号化して前記第２の暗号鍵群の中に保持することで、前記第１の暗号鍵群に含まれる前記所定の鍵より下層の鍵を共用する処理を実行させる処理プログラムを格納したメモリと、

を具備した集積回路。

【請求項１９】

前記第１共有鍵制御部は、

前記第１の暗号鍵群に含まれる暗号鍵で暗号化されている前記所定の鍵を、前記暗号鍵を用いて復号し、

復号された鍵を、前記第２の暗号鍵群に含まれる暗号鍵で再暗号化し、

再暗号化された鍵を、前記第２の暗号鍵群の中にコピーする

ことを特徴とする請求項１記載の情報処理装置。

【手続補正３】

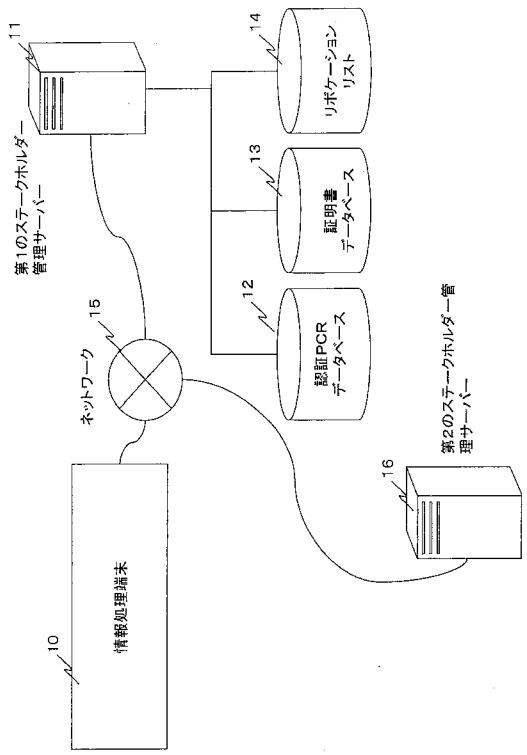
【補正対象書類名】図面

【補正対象項目名】全図

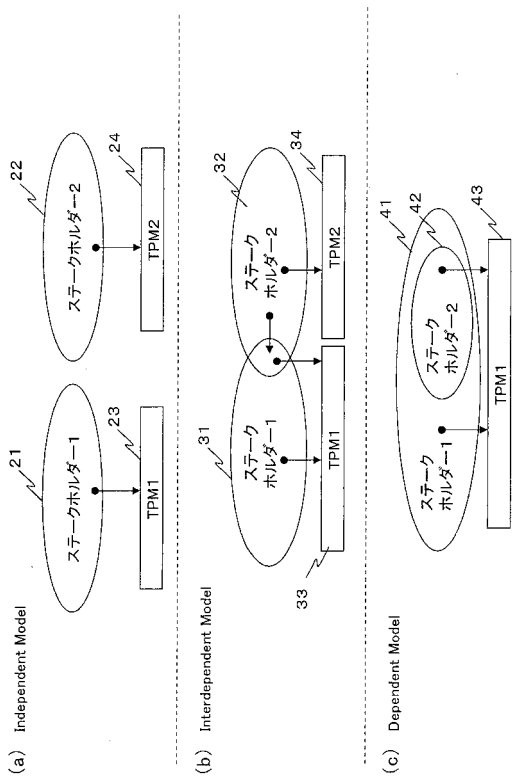
【補正方法】変更

【補正の内容】

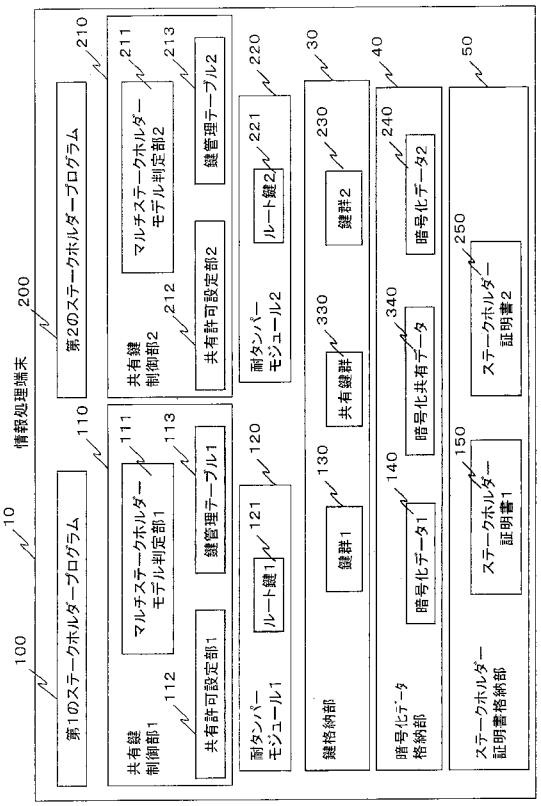
【図１】



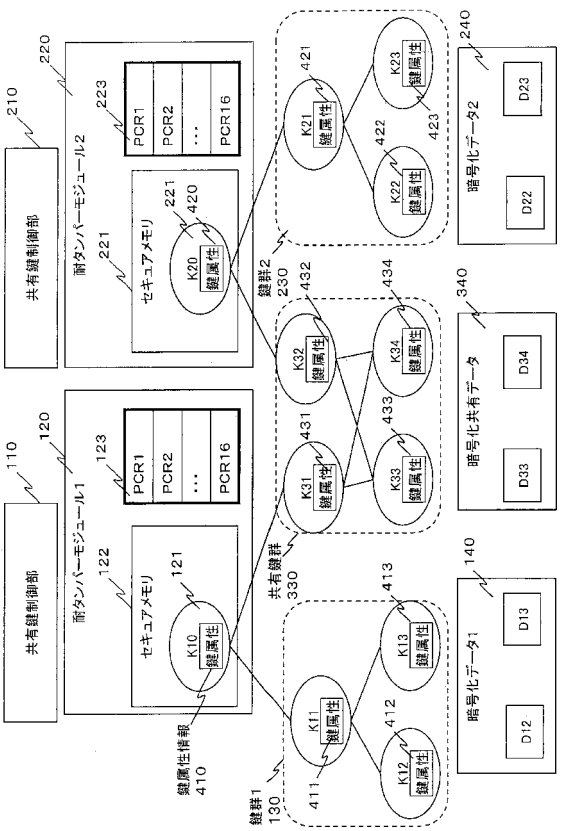
【図２】



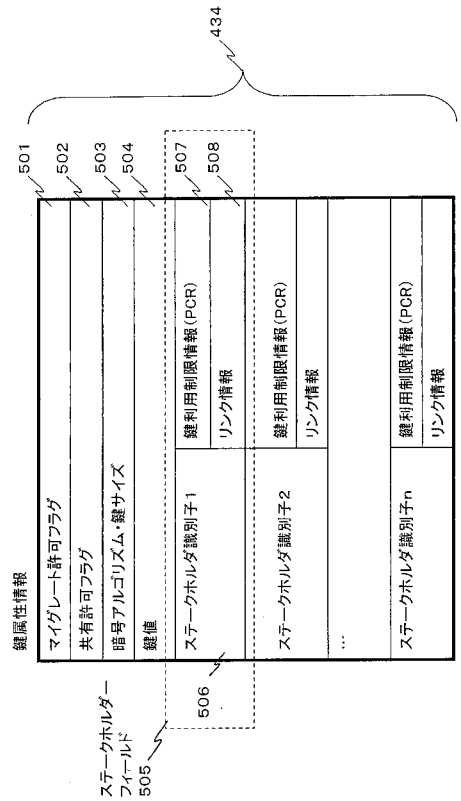
【図３】


情報処理端末
10
100
200
第1のステークホルダープログラム
110
第2のステークホルダープログラム
210
共有鍵制御部1
112
マルチステークホルダーモデル判定部1
111
共有許可設定部1
鍵管理テーブル1
113
共有鍵制御部2
212
マルチステークホルダーモデル判定部2
211
共有許可設定部2
鍵管理テーブル2
213
耐タンパーモジュール1
ルート鍵1
121
120
耐タンパーモジュール2
ルート鍵2
221
220
鍵格納部
鍵群1
130
共有鍵群
330
鍵群2
230
30
暗号化データ格納部
暗号化データ1
140
暗号化共有データ
340
暗号化データ2
240
40
ステークホルダー証明書格納部
ステークホルダー証明書1
150
ステークホルダー証明書2
250
50

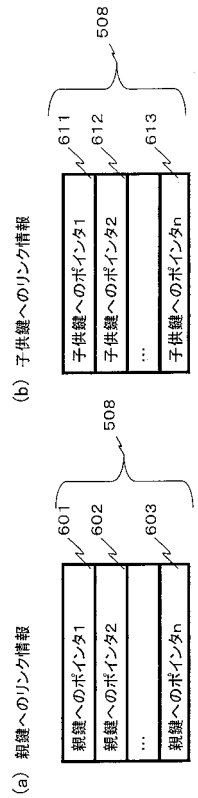
【図４】


共有鍵制御部
110
耐タンパーモジュール1
120
セキュアメモリ
122
K10
鍵属性
121
PCR1
PCR2
…
PCR16
123
鍵属性情報
410
共有鍵群
330
鍵群1
130
K11
K12
K13
412
413
411
暗号化データ1
140
D12
D13
共有鍵制御部
210
耐タンパーモジュール2
220
セキュアメモリ
221
K20
420
PCR1
PCR2
…
PCR16
223
鍵群2
230
K32
K31
K33
K34
431
432
433
434
K21
K22
K23
421
422
423
暗号化共有データ
340
D33
D34
暗号化データ2
240
D22
D23

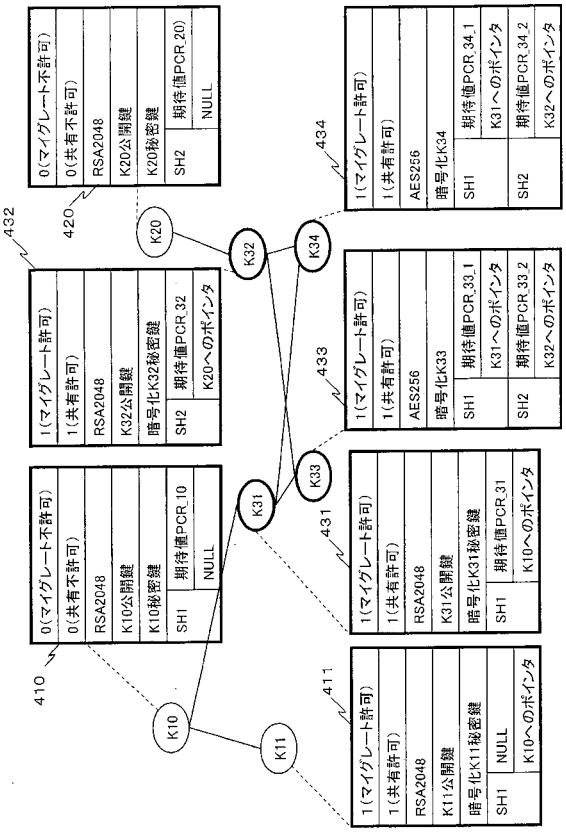
【図５】


鍵属性情報
501
マイグレート許可フラグ
502
共有許可フラグ
503
暗号アルゴリズム，鍵サイズ
504
鍵値
505
ステークホルダー識別子1
鍵利用制限情報(PCR)
507
リンク情報
508
ステークホルダー識別子2
鍵利用制限情報(PCR)
リンク情報
…
ステークホルダー識別子n
鍵利用制限情報(PCR)
リンク情報
506
434
ステークホルダーフィールド

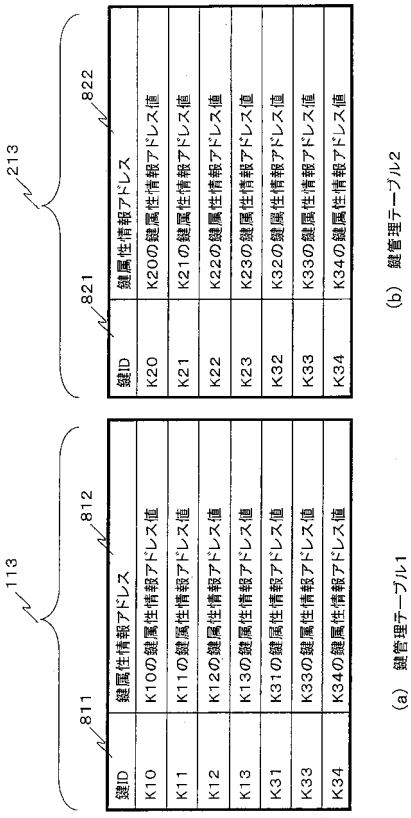
【図６】


(a) 親鍵へのリンク情報
601
親鍵へのポインタ1
602
親鍵へのポインタ2
…
603
親鍵へのポインタn
508
(b) 子供鍵へのリンク情報
611
子供鍵へのポインタ1
612
子供鍵へのポインタ2
…
613
子供鍵へのポインタn
508

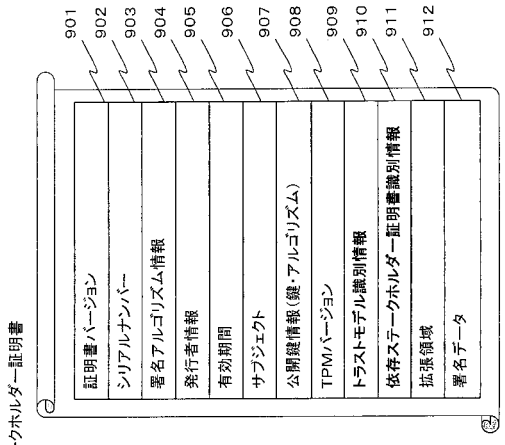
【図７】


410
411
420
431
432
433
434
K10
K11
K20
K31
K32
K33
K34
0(マイグレート不許可)
0(共有不許可)
RSA2048
K10公開鍵
K10秘密鍵
SH1
期待値PCR_10
NULL
1(マイグレート許可)
1(共有許可)
K11公開鍵
暗号化K11秘密鍵
K10へのポインタ
K20公開鍵
K20秘密鍵
SH2
期待値PCR_20
K32公開鍵
暗号化K32秘密鍵
期待値PCR_32
K20へのポインタ
K31公開鍵
暗号化K31秘密鍵
期待値PCR_31
K33公開鍵
暗号化K33秘密鍵
期待値PCR_33_1
期待値PCR_33_2
K34公開鍵
暗号化K34秘密鍵
期待値PCR_34_1
期待値PCR_34_2
K31へのポインタ
K32へのポインタ

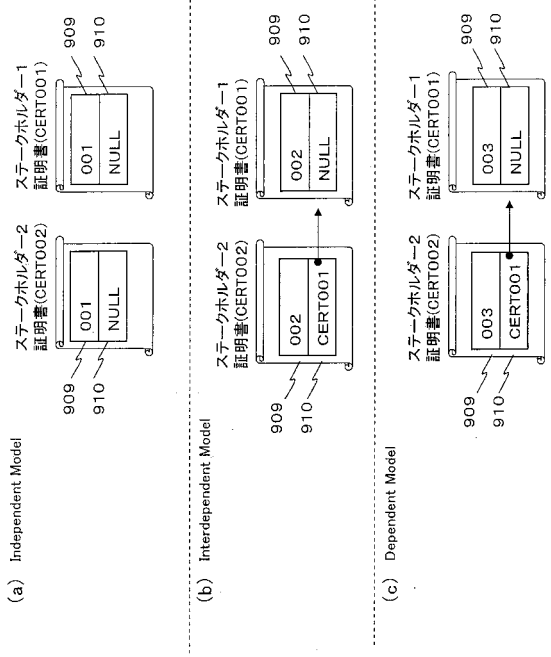
【図８】


113
213
811
812
821
822
鍵ID
鍵属性情報アドレス
K10
K10の鍵属性情報アドレス値
K11
K11の鍵属性情報アドレス値
K12
K12の鍵属性情報アドレス値
K13
K13の鍵属性情報アドレス値
K31
K31の鍵属性情報アドレス値
K33
K33の鍵属性情報アドレス値
K34
K34の鍵属性情報アドレス値
K20
K20の鍵属性情報アドレス値
K21
K21の鍵属性情報アドレス値
K22
K22の鍵属性情報アドレス値
K23
K23の鍵属性情報アドレス値
K32
K32の鍵属性情報アドレス値
(a) 鍵管理テーブル1
(b) 鍵管理テーブル2

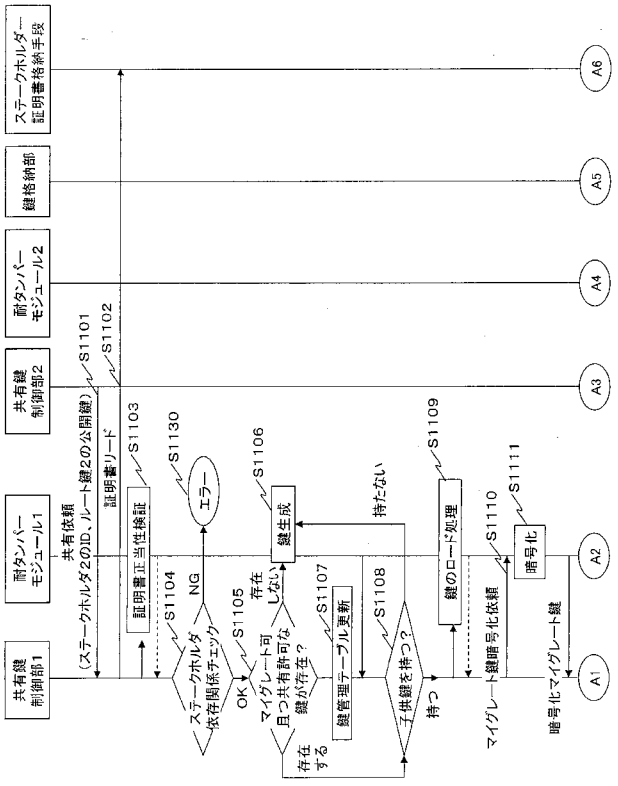
【図９】


ステークホルダー証明書
901
902
903
904
905
906
907
908
909
910
911
912
証明書バージョン
シリアルナンバー
署名アルゴリズム情報
発行者情報
有効期間
サブジェクト
公開鍵情報(鍵・アルゴリズム)
TPMバージョン
トラストモデル識別情報
依存ステークホルダー証明書識別情報
拡張領域
署名データ

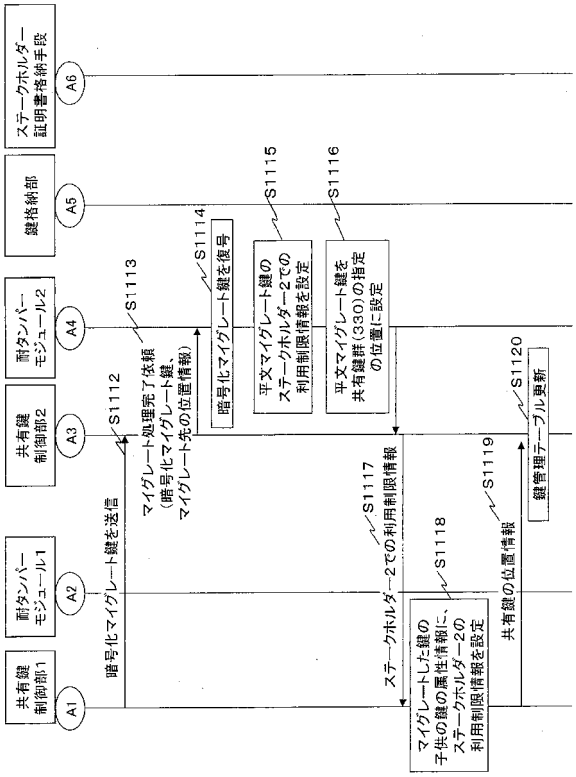
【図１０】


(a) Independent Model
(b) Interdependent Model
(c) Dependent Model
ステークホルダー2
証明書(CERT002)
ステークホルダー1
証明書(CERT001)
001
002
003
NULL
CERT001
909
910

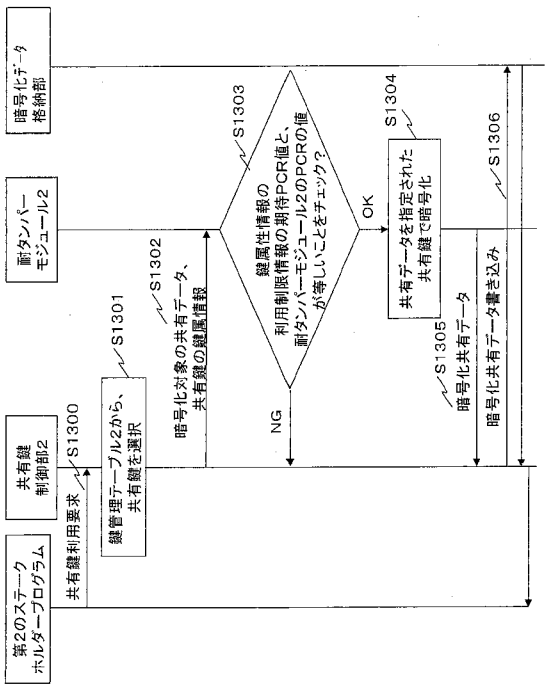
【図１１】



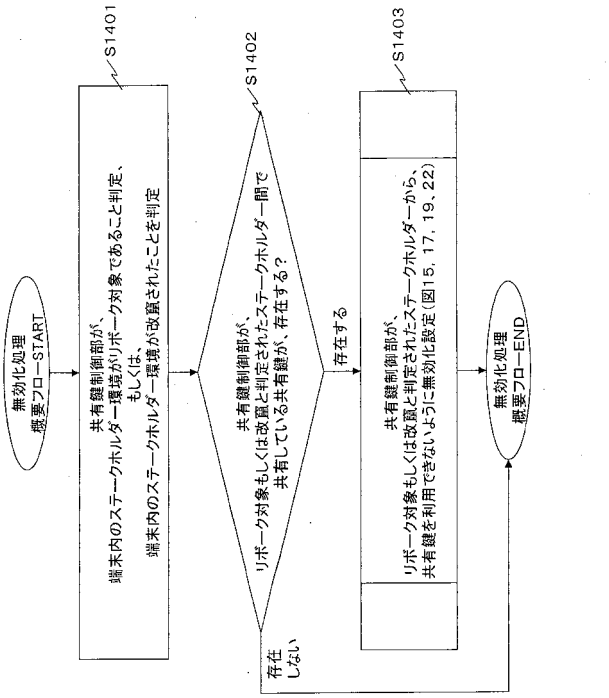
【図１２】



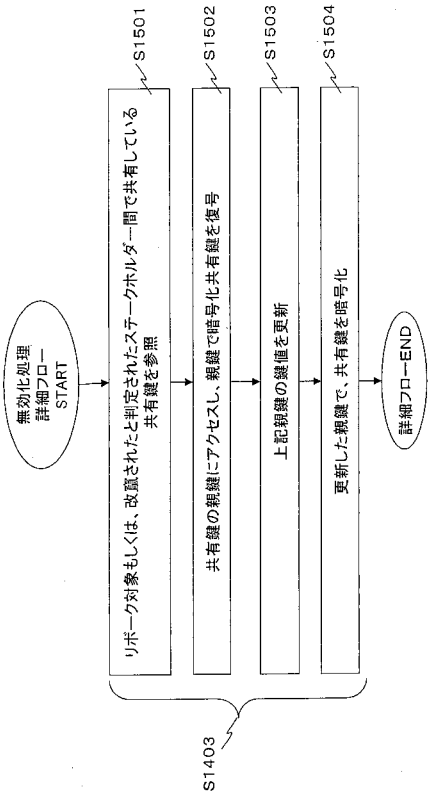
【図１３】



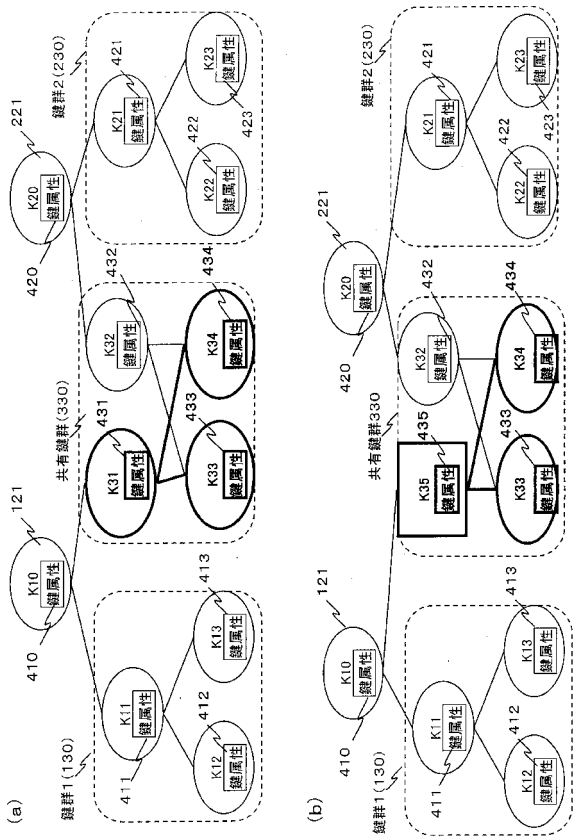
【図１４】



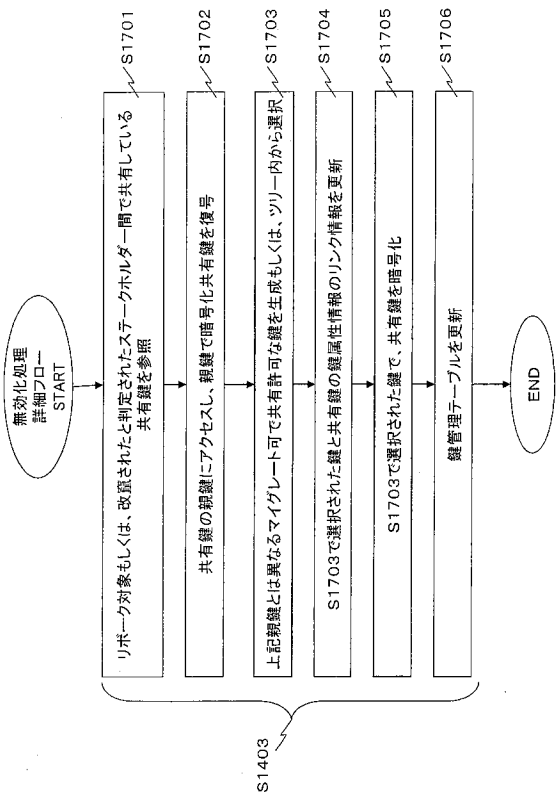
【図１５】



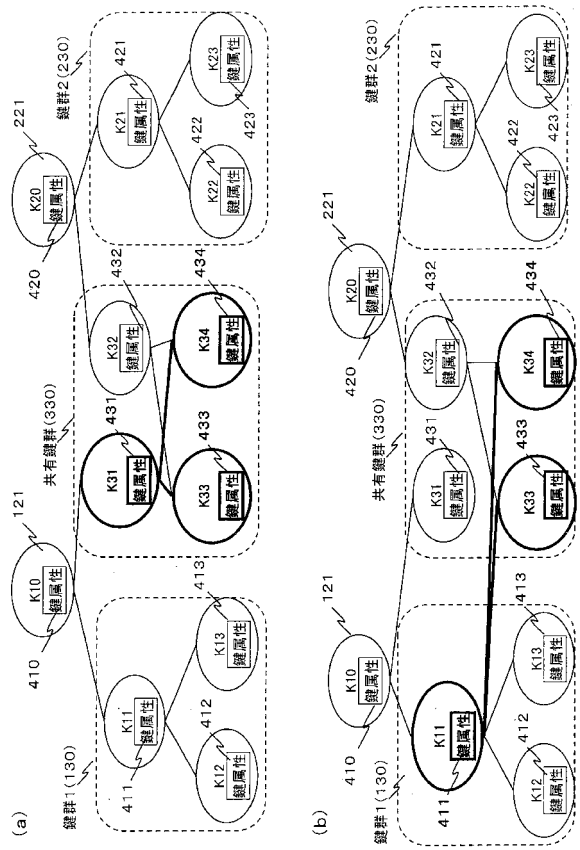
【図１６】



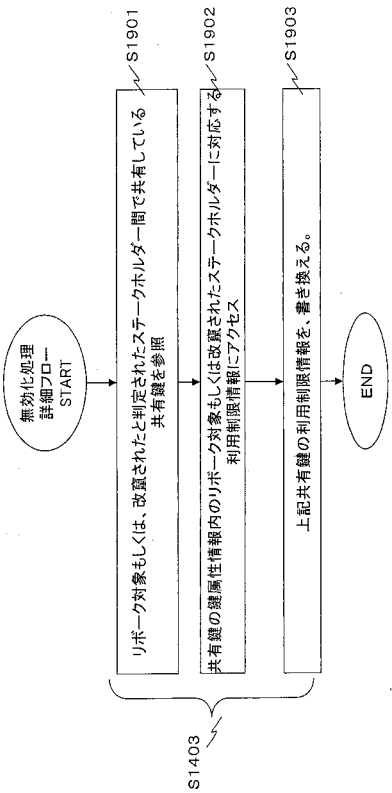
【図１７】



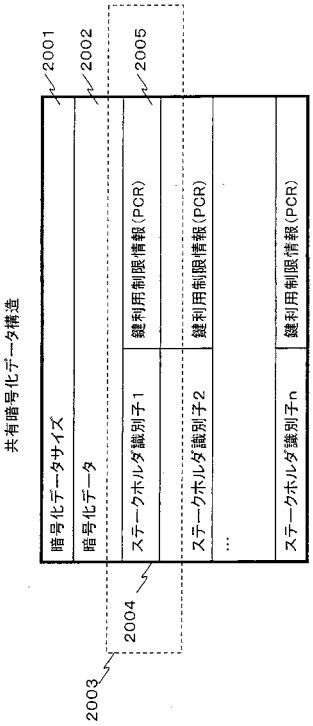
【図１８】



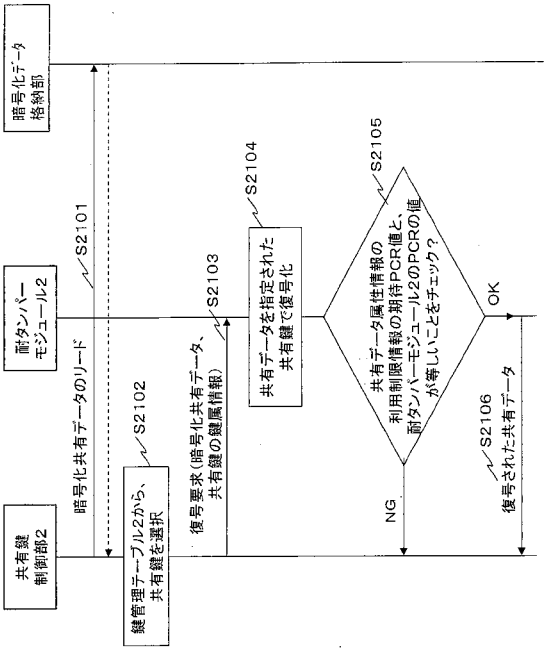
【図１９】



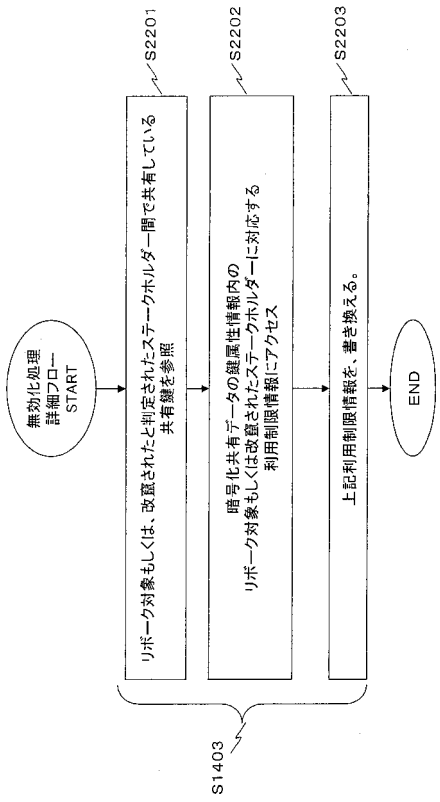
【図２０】



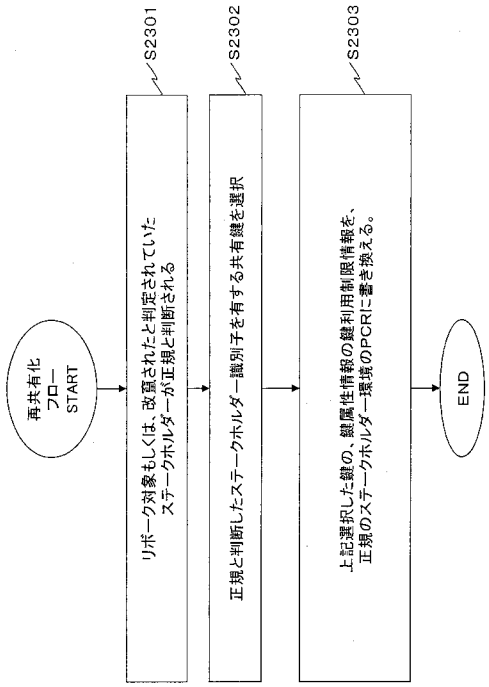
【図２１】



【図２２】



【図２３】


再共有化
フロー
START
S2301
リボーク対象もしくは、改竄されたと判定されていた
ステークホルダーが正規と判断される
S2302
正規と判断したステークホルダー識別子を有する共有鍵を選択
S2303
上記選択した鍵の、鍵属性情報の鍵利用制限情報を、
正規のステークホルダー環境のPCRに書き換える。
END

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP2009/002531

A. CLASSIFICATION OF SUBJECT MATTER
H04L9/08(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L9/08

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Jitsuyo Shinan Koho 1922-1996 Jitsuyo Shinan Toroku Koho 1996-2009
Kokai Jitsuyo Shinan Koho 1971-2009 Toroku Jitsuyo Shinan Koho 1994-2009

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
JSTPlus(JDreamII), JMEDPlus(JDreamII), JST7580(JDreamII)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Onur ACIICMEZ, Afshin LATIFI, Jean-Pierre SEIFERT, Xinwen ZHANG, "A Trusted Mobile Phone Prototype", 2008 Consumer Communications and Networking Conference, 2008.01.10, Volume 3 of 3, p.1208-1209	1-19
A	Takaaki MIZUKI, and Takao NISHIZEKI, "Necessary and Sufficient Numbers of Cards for Sharing Secret Keys on Hierarchical Groups", IEICE transactions on information and systems, 2002.02.01, VOL.E85-D, NO.2, p.333-345, Special Issue on Selected Papers from LA Symposium	1-19
E,A	JP 2009-3855 A (Panasonic Corp.), 08 January, 2009 (08.01.09), Full text; all drawings & US 2009/0019551 A1	1-19

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
26 August, 2009 (26.08.09)	08 September, 2009 (08.09.09)
Name and mailing address of the ISA/ Japanese Patent Office Facsimile No.	Authorized officer Telephone No.

Form PCT/ISA/210 (second sheet) (April 2007)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP2009/002531

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
E,A	Jen-Shiun LIN, Kuo-Hsuan HUANG, Feipei LAI, Hung-Chang LEE, "Secure and efficient group key management with shared key derivation", COMPUTER STANDARDS & INTERFACES, Volume 31, Issue 1, 2009.01, p.192-208	1-19

国際調査報告	国際出願番号 ＰＣＴ／ＪＰ２００９／００２５３１

A． 発明の属する分野の分類（国際特許分類（ＩＰＣ））

Int.Cl. H04L9/08(2006.01)i

B． 調査を行った分野

調査を行った最小限資料（国際特許分類（ＩＰＣ））

Int.Cl. H04L9/08

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	１９２２－１９９６年
日本国公開実用新案公報	１９７１－２００９年
日本国実用新案登録公報	１９９６－２００９年
日本国登録実用新案公報	１９９４－２００９年

国際調査で使用した電子データベース（データベースの名称、調査に使用した用語）

JSTPlus(JDreamII), JMEDPlus(JDreamII), JST7580(JDreamII)

C． 関連すると認められる文献

引用文献のカテゴリー＊	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する請求項の番号
A	Onur ACIICMEZ, Afshin LATIFI, Jean-Pierre SEIFERT, Xinwen ZHANG, “A Trusted Mobile Phone Prototype”, 2008 Consumer Communications and Networking Conference, 2008.01.10, Volume 3 of 3, p.1208-1209	１－１９

☑ Ｃ欄の続きにも文献が列挙されている。 ☐ パテントファミリーに関する別紙を参照。

＊ 引用文献のカテゴリー
「Ａ」特に関連のある文献ではなく、一般的技術水準を示すもの
「Ｅ」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの
「Ｌ」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す）
「Ｏ」口頭による開示、使用、展示等に言及する文献
「Ｐ」国際出願日前で、かつ優先権の主張の基礎となる出願の日の後に公表された文献
「Ｔ」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの
「Ｘ」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの
「Ｙ」特に関連のある文献であって、当該文献と他の１以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの
「＆」同一パテントファミリー文献

国際調査を完了した日	国際調査報告の発送日
２６．０８．２００９	０８．０９．２００９

国際調査機関の名称及びあて先	特許庁審査官（権限のある職員）	５Ｓ	４２２９
日本国特許庁（ＩＳＡ／ＪＰ） 郵便番号１００－８９１５ 東京都千代田区霞が関三丁目４番３号	青木 重徳 電話番号 ０３－３５８１－１１０１ 内線 ３５４６		

様式ＰＣＴ／ＩＳＡ／２１０（第２ページ）（２００７年４月）

国際調査報告 国際出願番号 ＰＣＴ／ＪＰ２００９／００２５３１

Ｃ（続き）． 関連すると認められる文献

引用文献のカテゴリー＊	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する請求項の番号
A	Takaaki MIZUKI, and Takao NISHIZEKI, “Necessary and Sufficient Numbers of Cards for Sharing Secret Keys on Hierarchical Groups”, IEICE transactions on information and systems, 2002.02.01, VOL.E85-D, NO.2, p.333-345, Special Issue on Selected Papers from LA Symposium	１－１９
E, A	JP 2009-3855 A (パナソニック株式会社) 2009.01.08, 全文, 全図 & US 2009/0019551 A1	１－１９
E, A	Jen-Shiun LIN, Kuo-Hsuan HUANG, Feipei LAI, Hung-Chang LEE, “Secure and efficient group key management with shared key derivation”, COMPUTER STANDARDS & INTERFACES, Volume 31, Issue 1, 2009.01, p.192-208	１－１９

フロントページの続き

(81)指定国　　AP(BW,GH,GM,KE,LS,MW,MZ,NA,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HR,HU,IE,IS,IT,LT,LU,LV,MC,MK,MT,NL,NO,PL,PT,RO,SE,SI,SK,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AO,AT,AU,AZ,BA,BB,BG,BH,BR,BW,BY,BZ,CA,CH,CL,CN,CO,CR,CU,CZ,DE,DK,DM,DO,DZ,EC,EE,EG,ES,FI,GB,GD,GE,GH,GM,GT,HN,HR,HU,ID,IL,IN,IS,JP,KE,KG,KM,KN,KP,KR,KZ,LA,LC,LK,LR,LS,LT,LU,LY,MA,MD,ME,MG,MK,MN,MW,MX,MY,MZ,NA,NG,NI,NO,NZ,OM,PG,PH,PL,PT,RO,RS,RU,SC,SD,SE,SG,SK,SL,SM,ST,SV,SY,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VC,VN,ZA,ZM,ZW

(72)発明者　ニコルソン　ケネス　アレクサンダー
大阪府門真市大字門真１００６番地　パナソニック株式会社内
(72)発明者　松島　秀樹
大阪府門真市大字門真１００６番地　パナソニック株式会社内
(72)発明者　伊藤　孝幸
大阪府門真市大字門真１００６番地　パナソニック株式会社内
(72)発明者　高山　久
大阪府門真市大字門真１００６番地　パナソニック株式会社内
(72)発明者　前田　学
大阪府門真市大字門真１００６番地　パナソニック株式会社内

Ｆターム(参考) 5J104 AA16　EA06　EA07　NA02