

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7410161号
(P7410161)

(45)発行日 令和6年1月9日(2024.1.9)

(24)登録日 令和5年12月25日(2023.12.25)

(51)国際特許分類		F I	
G 0 6 F	21/60 (2013.01)	G 0 6 F	21/60 3 2 0
G 0 6 F	21/14 (2013.01)	G 0 6 F	21/14
G 0 6 F	12/14 (2006.01)	G 0 6 F	12/14

請求項の数 6 (全39頁)

(21)出願番号	特願2021-549577(P2021-549577)	(73)特許権者	390009531
(86)(22)出願日	令和2年3月6日(2020.3.6)		インターナショナル・ビジネス・マシー
(65)公表番号	特表2022-522664(P2022-522664		ンズ・コーポレーション
	A)		INTERNATIONAL BUSI
(43)公表日	令和4年4月20日(2022.4.20)		NESS MACHINES CORPO
(86)国際出願番号	PCT/IB2020/051941		RATION
(87)国際公開番号	WO2020/183308		アメリカ合衆国10504 ニューヨー
(87)国際公開日	令和2年9月17日(2020.9.17)		ク州 アーモンク ニュー オーチャード
審査請求日	令和4年8月24日(2022.8.24)		ロード
(31)優先権主張番号	16/296,303		New Orchard Road, A
(32)優先日	平成31年3月8日(2019.3.8)		rmonk, New York 105
(33)優先権主張国・地域又は機関	米国(US)		04, United States of
			America
		(74)代理人	100112690
			弁理士 太佐 種一

最終頁に続く

(54)【発明の名称】 ページ変更検出によるセキュアなページング

(57)【特許請求の範囲】

【請求項1】

コンピュータ・システムのメモリのページのハッシュ値を計算することと、
前記ハッシュ値を前記ページの以前に計算されたハッシュ値と比較することと、
前記ハッシュ値が前記以前に計算されたハッシュ値に一致するということの決定に基づ
いて、ページごと、暗号化ごとの値を前記ページの暗号化において使用することと、
前記ハッシュ値が前記以前に計算されたハッシュ値に一致しないということの決定に基
づいて、前記ページごと、暗号化ごとの値の修正値を前記ページの暗号化において使用す
ることとを含む、方法。

【請求項2】

前記ページをセキュア・ページから非セキュア・ページに変換するというホストの要求
に応答して、セキュア制御インターフェイスによって前記暗号化が実行される、請求項1
に記載の方法。

【請求項3】

暗号化された前記非セキュア・ページを、格納のために前記ホストに提供することをさ
らに含む、請求項2に記載の方法。

【請求項4】

メモリと、
処理ユニットと、
前記処理ユニットおよび前記メモリとインターフェイスをとるセキュア・インターフェ

イス制御とを備えているシステムであって、前記セキュア・インターフェイス制御が、
前記メモリのページのハッシュ値を計算することと、
前記ハッシュ値を前記ページの以前に計算されたハッシュ値と比較することと、
前記ハッシュ値が前記以前に計算されたハッシュ値に一致するということの決定に基づいて、ページごと、暗号化ごとの値を前記ページの暗号化において使用することと、
前記ハッシュ値が前記以前に計算されたハッシュ値に一致しないということの決定に基づいて、前記ページごと、暗号化ごとの値の修正値を前記ページの暗号化において使用することを含む動作を実行するように構成される、システム。

【請求項 5】

コンピュータに、
コンピュータ・システムのメモリのページのハッシュ値を計算することと、
前記ハッシュ値を前記ページの以前に計算されたハッシュ値と比較することと、
前記ハッシュ値が前記以前に計算されたハッシュ値に一致するということの決定に基づいて、ページごと、暗号化ごとの値を前記ページの暗号化において使用することと、
前記ハッシュ値が前記以前に計算されたハッシュ値に一致しないということの決定に基づいて、前記ページごと、暗号化ごとの値の修正値を前記ページの暗号化において使用することを実行させるためのプログラム。

【請求項 6】

請求項 5 に記載のプログラムを記憶したコンピュータ読取可能な記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、一般に、コンピュータ技術に関し、より詳細には、ページ変更検出によるセキュアなページングに関する。

【背景技術】

【0002】

クラウド・コンピューティングおよびクラウド・ストレージは、それらのデータをサードパーティのデータ・センターに格納して処理する能力をユーザに提供する。クラウド・コンピューティングは、顧客がハードウェアを購入することも、物理的サーバのための床スペースを提供することも必要とせず、VMを顧客のために迅速かつ簡単にプロビジョニングする能力を促進する。顧客は、顧客の嗜好または要件の変化に従って、VMを簡単に拡大または縮小することができる。通常、クラウド・コンピューティング・プロバイダは、プロバイダのデータ・センターで、サーバ上に物理的に存在するVMをプロビジョニングする。特に、コンピューティング・プロバイダが、多くの場合、通常は2人以上の顧客のデータを同じサーバ上に格納するため、顧客は、多くの場合、VM内のデータのセキュリティについて心配する。顧客は、顧客自身のコード／データとプロバイダのサイトで実行されている他のVMのコード／データの間のセキュリティだけでなく、顧客自身のコード／データとクラウド・コンピューティング・プロバイダのコード／データの間のセキュリティを要求することがある。加えて、顧客は、マシン上で実行されている他のコードからの可能性のあるセキュリティ違反に対するセキュリティだけでなく、プロバイダの管理者からのセキュリティを要求することがある。

【0003】

そのような注意を要する状況に対処するために、クラウド・サービス・プロバイダは、適切なデータ分離および論理的ストレージ分離を保証するようにセキュリティ制御を実施することがある。クラウド・インフラストラクチャの実装における仮想化の広範囲に及ぶ使用の結果、仮想化が、オペレーティング・システム(OS: operating system)と基礎になるハードウェアの間の関係を(ハードウェアがコンピューティング、ストレージ、またはネットワークのいずれであっても)変更するため、クラウド・サービスの顧客に固有のセキュリティに関する懸念が生じる。このため仮想化は、それ自体が適切に構成され、管理され、保護されなければならない追加のレイヤとして導入される。

【0004】

一般に、ホスト・ハイパーバイザの制御下でゲストとして実行されるVMは、そのハイパーバイザが仮想化サービスをそのゲストに透過的に提供することに依存する。これらのサービスは、メモリ管理、命令エミュレーション、および割り込み処理を含む。

【0005】

メモリ管理の場合、VMは、そのデータをメモリに常駐させるためにディスクから移動する（ページインする）ことができ、VMは、そのデータをディスクに戻す（ページアウトする）こともできる。ページがメモリに常駐している間、VM（ゲスト）は、動的アドレス変換（DAT：dynamic address translation）を使用してメモリ内のページをゲスト仮想アドレスからゲスト絶対アドレスにマッピングする。加えて、ホスト・ハイパーバイザは、メモリ内のゲスト・ページに関して、それ自身の（ホスト仮想アドレスからホスト絶対アドレスへの）DATマッピングを有しており、ゲスト・ページを、ゲストから独立して透過的にメモリにページインし、メモリからページアウトすることができる。ホストDATテーブルによって、ハイパーバイザは、メモリ分離、または2つの分離したゲストVM間のゲスト・メモリの共有を実現する。ホストは、必要な場合に、ゲスト・メモリにアクセスし、ゲストの代わりにゲストの動作をシミュレートすることもできる。

【発明の概要】

【0006】

本発明の1つまたは複数の実施形態によれば、コンピュータ実装方法は、コンピュータ・システムのメモリのページのハッシュ値を計算することと、このハッシュ値をページの以前に計算されたハッシュ値と比較することを含む。ハッシュ値が以前に計算されたハッシュ値に一致するということの決定に基づいて、ページごと、暗号化ごとの値をページの暗号化において使用することができる。ハッシュ値が以前に計算されたハッシュ値に一致しないということの決定に基づいて、ページごと、暗号化ごとの値の修正値をページの暗号化において使用することができる。利点としては、暗号化ごとの値の組み込みによって暗号化を改良しながら、基礎になるデータに対する変更が発生しなかった場合に、暗号化の更新を制限することが挙げられる。

【0007】

本発明の追加または代替の実施形態に従って、ページをセキュア・ページから非セキュア・ページに変換するというホストの要求に応答して、セキュア制御インターフェイスによって暗号化が実行され得る。利点としては、ページを非セキュアにし、信頼できない実体によってアクセスできるようにする前に、暗号化によって、基礎になるデータへの直接アクセスを防ぐことが挙げられる。

【0008】

本発明の追加または代替の実施形態に従って、暗号化された非セキュア・ページがストレージのホストに提供され得る。利点としては、信頼できないことがある実体によるメモリ・ページの入出力アクセスおよび移動のためのメモリ管理を可能にすることが挙げられる。

【0009】

本発明の追加または代替の実施形態に従って、ハッシュ値を、セキュア制御インターフェイスのセキュア・テーブルに格納することができる。利点としては、変更を比較するための要約値を保存することが挙げられる。

【0010】

本発明の追加または代替の実施形態に従って、非セキュア・ページがセキュア・ページに変換され得る。セキュア・インターフェイス制御（secure interface control）は、ページに関連付けられた暗号化ごとの値に基づいてセキュア・ページを復号し、復号されたページを生成することができる。復号されたページのハッシュ値が計算され得る。復号されたページのハッシュ値は、セキュア・テーブルに格納されているページのハッシュ値と比較され得る。復号されたページのハッシュ値がセキュア・テーブルに格納されているハッシュ値に一致するということの決定に基づいて、復号されたページの妥当性が確認され

得る。利点としては、信頼できない実体に提供されたページが変更されていないことの検証が挙げられる。

【0011】

本発明の追加または代替の実施形態に従って、セキュア・インターフェイス制御は、ファームウェア、ハードウェア、またはファームウェアとハードウェアの組み合わせを含むことができる。セキュア・ページは、ホストによって管理されているセキュア・コンテナまたはセキュア仮想マシンに割り当てられ得る。ホストは、ハイパーバイザまたはオペレーティング・システムであることができる。利点としては、暗号化を使用して、セキュアな実体からのセキュア・ページを信頼できない実体と共有することが挙げられる。

【0012】

本発明の追加または代替の実施形態に従って、ページを暗号化することは、ページに関連付けられたアドレス値、1つまたは複数の乱数値、および暗号化ごとの値と、暗号によって保護されたハッシュ関数との組み合わせを組み込むことができる。利点としては、暗号化の複雑さ向上させてセキュリティを強化することが挙げられる。

【0013】

本発明の他の実施形態は、コンピュータ・システムおよびコンピュータ・プログラム製品において、前述した方法の特徴を実装する。

【0014】

追加の特徴および利点が、本開示の手法によって実現される。本発明のその他の実施形態および態様は、本明細書において詳細に説明され、本発明の一部と見なされる。本発明を利点および特徴と共によく理解するために、説明および図面を参照されたい。

【0015】

本明細書に記載された専有権の詳細は、本明細書の最後にある特許請求の範囲において具体的に指摘され、明確に請求される。本発明の各実施形態の前述およびその他の特徴と利点は、添付の図面と併せて行われる以下の詳細な説明から明らかになる。

【図面の簡単な説明】

【0016】

【図1】本発明の1つまたは複数の実施形態に従って、ゾーン・セキュリティのためのテーブルを示す図である。

【図2】本発明の1つまたは複数の実施形態に従って、DATを実行するための仮想アドレス空間および絶対アドレス空間を示す図である。

【図3】本発明の1つまたは複数の実施形態に従って、ハイパーバイザの下で実行されている仮想マシン（VM：virtual machine）を支援するためのネストされたマルチパートDAT（multi-partDAT）を示す図である。

【図4】本発明の1つまたは複数の実施形態に従って、セキュア・ゲスト・ストレージのマッピングを示す図である。

【図5】本発明の1つまたは複数の実施形態に従って、動的アドレス変換（DAT）動作のシステム概略図である。

【図6】本発明の1つまたは複数の実施形態に従って、セキュア・インターフェイス制御メモリのシステム概略図である。

【図7】本発明の1つまたは複数の実施形態に従って、インポート動作のプロセス・フローを示す図である。

【図8】本発明の1つまたは複数の実施形態に従って、インポート動作のプロセス・フローを示す図である。

【図9】本発明の1つまたは複数の実施形態に従って、提供されたメモリの動作のプロセスを示す図である。

【図10】本発明の1つまたは複数の実施形態に従って、セキュア・インターフェイス制御のセキュア・ページへの非セキュア・ハイパーバイザ・ページの遷移のプロセス・フローを示す図である。

【図11】本発明の1つまたは複数の実施形態に従って、セキュア・インターフェイス制

10

20

30

40

50

御によって行われるセキュア・ストレージ・アクセスのプロセス・フローを示す図である。

【図 1 2】本発明の 1 つまたは複数の実施形態に従って、セキュア・インターフェイス制御およびハードウェアによるアクセスのタグ付けのプロセス・フローを示す図である。

【図 1 3】本発明の 1 つまたは複数の実施形態に従って、プログラムおよびセキュア・インターフェイス制御によってセキュア・アクセスおよび非セキュア・アクセスを支援するための変換のプロセス・フローを示す図である。

【図 1 4】本発明の 1 つまたは複数の実施形態に従って、プログラムおよびセキュア・インターフェイス制御によるセキュア・ストレージの保護を伴う D A T のプロセス・フローを示す図である。

【図 1 5】本発明の 1 つまたは複数の実施形態に従って、暗号化の前にページの内容が変更されたかどうかを判定するためのプロセス・フローを示す図である。

10

【図 1 6】本発明の 1 つまたは複数の実施形態に従って、セキュア・インターフェイス制御による暗号化および復号の制御のためのデータ・フローを示す図である。

【図 1 7】本発明の 1 つまたは複数の実施形態に従って、クラウド・コンピューティング環境を示す図である。

【図 1 8】本発明の 1 つまたは複数の実施形態に従って、抽象モデル・レイヤを示す図である。

【図 1 9】本発明の 1 つまたは複数の実施形態に従って、システムを示す図である。

【図 2 0】本発明の 1 つまたは複数の実施形態に従って、処理システムを示す図である。

【発明を実施するための形態】

20

【 0 0 1 7 】

本明細書において示される図は、実例である。本発明の思想から逸脱することなく、本明細書に記載された図または動作の多くの変形が存在することが可能である。例えば、動作は異なる順序で実行されることが可能であり、あるいは動作は追加、削除、または変更されることが可能である。また、「結合される」という用語およびその変形は、2つの要素間に通信経路が存在することを表しており、それらの要素間に要素／接続が介在しない要素間の直接的接続を意味していない。これらのすべての変形は、本明細書の一部であると見なされる。

【 0 0 1 8 】

本発明の 1 つまたは複数の実施形態は、基礎になるデータにおける変化が検出されたかどうかに基づいて、暗号化プロセスを変更するカウンタを使用してメモリ・ページの暗号化を実行する。セキュア・ドメイン内のデータのページがセキュアでない実体によってアクセスできないことを保証するように、セキュア・ストレージが管理され得る。セキュア・ドメイン内のデータが、ディスクへの格納などのために非セキュア・ドメインに転送されるときに、データが、非セキュア・ドメインに転送する前に暗号化され得る。セキュリティを強化するために、データのページを暗号化するとき、データに対する変更の検出時にインクリメントされるカウンタ値などの複数のアイテムが初期化ベクトルに結合され得る。したがって、データのページが変化しないままである場合、暗号化は、暗号化データの他の以前のコピーに一致し、既存のページング・アルゴリズムを維持する。しかし、基礎になるデータが変化した場合、暗号化ごとのカウンタの暗号化ごとの値が変更され、暗号化プロセスの一部として使用され得る。暗号化ごとの値を組み込むことによって、データ値の変更および暗号化ごとの値の変更の組み合わせに基づいて暗号化が変化するため、暗号化データ内のパターンを検出することをさらに困難にすることができる。暗号化ごとのカウンタと組み合わせてページ変更検出を含むことは、基礎になる内容が変化しないままである限り、ページアウトされた暗号化済みの内容の複数回の再使用を支援することができ、読み取り専用のメモリ・ページのための処理システムの効率向上に役立つことができる。これに対して、ページ変更検出が使用されず、例えばページの各アクセスで暗号化ごとのカウンタが変更された場合、基礎になるデータが変更されなかった場合でも、ページアウトされたページを複数回再使用することができない。

30

40

【 0 0 1 9 】

50

仮想マシン、コンテナ、ハイパーバイザ、オペレーティング・システム、およびその他のコンピュータ・システムの要素を支援するためのページングを参照して、本発明の実施形態が説明される。本明細書に記載されたプロセスが、多数のシステム構成に適用可能であり、本明細書でより詳細に説明される例示的なシステムに限定される必要がないということが、理解されるであろう。

【0020】

本発明の1つまたは複数の実施形態は、ソフトウェアとマシンの間の効率的な軽量のセキュア・インターフェイス制御を活用して、セキュリティを強化する。

【0021】

ホスト・ハイパーバイザの制御下でゲストとして実行される仮想マシン（VM）は、そのハイパーバイザが仮想化サービスをそのゲストに透過的に提供することに依存する。これらのサービスは、セキュアな実体と別の信頼できない実体の間の、この他の実体によるセキュア・リソースへのアクセスを従来は許可していた任意のインターフェイスに適用され得る。前述したように、これらのサービスは、メモリ管理、命令エミュレーション、および割り込み処理を含むことができるが、これらに限定されない。例えば、割り込みおよび例外の投入の場合、ハイパーバイザは、通常、ゲストのプレフィックス領域（ロー・コア）に対して、読み取りまたは書き込みあるいはその両方を行う。「仮想マシン」または「VM」という用語は、本明細書において使用されるとき、物理マシン（コンピューティング・デバイス、プロセッサなど）およびその処理環境（オペレーティング・システム（OS）、ソフトウェア・リソースなど）の論理的表現のことを指す。VMは、基礎になるホスト・マシン（物理プロセッサまたはプロセッサのセット）上で実行されるソフトウェアとして維持される。ユーザまたはソフトウェア・リソースの視点からは、VMは、それ自身が独立した物理マシンであるように見える。「ハイパーバイザ」および「VMモニタ（VMM: VM Monitor）」という用語は、本明細書において使用されるとき、同じホスト・マシン上で複数の（しばしば異なる）OSを使用して実行するように、複数のVMを管理および許可する処理環境またはプラットフォーム・サービスのことを指す。VMをデプロイすることが、VMのインストール・プロセスおよびVMの有効化（または起動）プロセスを含むということが、理解されるべきである。別の例では、VMをデプロイすることは、VMの有効化（または起動）プロセスを含む（例えば、VMがすでにインストールされているか、またはすでに存在する場合）。

【0022】

セキュア・ゲストを促進し、支援するためには、ハイパーバイザがVMのデータにアクセスできず、したがって前述した方法でサービスを提供できないように、ハイパーバイザに依存しない、ハイパーバイザとセキュア・ゲストの間のセキュリティの向上が必要になるという技術的課題が存在する。

【0023】

本明細書に記載されたセキュアな実行は、セキュア・ストレージと非セキュア・ストレージの間の分離、および異なるセキュアなユーザに属するセキュア・ストレージ間の分離を保証するためのハードウェア・メカニズムを提供する。セキュア・ゲストの場合、「信頼できない」非セキュア・ハイパーバイザとセキュア・ゲストの間のセキュリティが強化される。これを行うには、通常はゲストの代わりにハイパーバイザが実行する機能の多くがマシンに組み込まれる必要がある。ハイパーバイザとセキュア・ゲストの間のセキュア・インターフェイスを提供するための新しいセキュア・インターフェイス制御（本明細書では、「UV」とも呼ばれる）が、本明細書において説明される。セキュア・インターフェイス制御およびUVという用語は、本明細書では交換可能なように使用される。セキュア・インターフェイス制御は、ハードウェアと連携して機能し、このセキュリティの向上を実現する。加えて、下位レベルのハイパーバイザが、この信頼できないハイパーバイザに仮想化を提供していることがあり、この下位レベルのハイパーバイザは、信頼できるコードで実装されている場合、セキュア・インターフェイス制御の一部になることもできる。

【0024】

セキュア・インターフェイス制御は、1つの例では、内部のセキュアな信頼できるハードウェアまたはファームウェアあるいはその両方に実装される。この信頼できるファームウェアは、例えば、プロセッサのミリコードまたはP R / S Mの論理分割コードを含むことができる。セキュア・ゲストまたはセキュアな実体に関して、セキュア・インターフェイス制御は、セキュアな環境の初期化および維持に加えて、ハードウェア上でこれらのセキュアな実体のディスパッチの調整を行う。セキュア・ゲストがデータを活発に使用しており、ホスト・ストレージに常駐している間、このセキュア・ゲストは、セキュア・ストレージ内で「疑いが晴れた状態」に保たれる。その単一のセキュア・ゲストによって、セキュア・ゲスト・ストレージにアクセスすることができ、このアクセスは、ハードウェアによって厳密に実施される。すなわち、ハードウェアは、任意のセキュアでない実体（ハイパーバイザまたはその他の非セキュア・ゲストを含む）または異なるセキュア・ゲストがそのデータにアクセスするのを防ぐ。この例では、セキュア・インターフェイス制御は、最低レベルのファームウェアの信頼できる部分として実行される。この最低レベル、またはミリコードは、実際にはハードウェアの拡張であり、例えばI B Mのz A r c h i t e c t u r e (R) において定義されている複雑な命令および機能を実装するために使用される。ミリコードは、セキュアな実行との関連において、それ自身のセキュアU V ストレージ、非セキュア・ハイパーバイザ・ストレージ、セキュア・ゲスト・ストレージ、および共有ストレージを含む、ストレージのすべての部分にアクセスすることができる。これによって、ミリコードは、セキュア・ゲストによって、またはそのゲストの支援においてハイパーバイザによって必要とされるすべての機能を提供することができる。セキュア・インターフェイス制御は、ハードウェアに直接アクセスすることもでき、セキュア・インターフェイス制御によって確立された条件の制御下で、ハードウェアが効率的にセキュリティ・チェックを実行できるようにする。

【0025】

本発明の1つまたは複数の実施形態に従って、ソフトウェアは、U V 呼び出し（U V C : U V C a l l）命令を使用して、セキュア・インターフェイス制御に対して特定の動作を実行するよう要求する。例えば、U V C 命令は、ハイパーバイザによって、セキュア・インターフェイス制御を初期化し、セキュア・ゲスト・ドメイン（例えば、セキュア・ゲスト構成）を作成し、そのセキュアな構成内で仮想C P Uを作成するために使用され得る。U V C 命令は、ハイパーバイザのページイン動作またはページアウト動作の一部として、セキュア・ゲスト・ページをインポートすること（復号してセキュア・ゲスト・ドメインに割り当てること）、およびエクスポートすること（暗号化してホストがアクセスできるようにすること）にも使用され得る。加えて、セキュア・ゲストは、ハイパーバイザと共有されるストレージを定義し、セキュア・ストレージを共有にし、共有ストレージをセキュアにする能力を有する。

【0026】

これらのU V C コマンドは、多くの他の設計された命令と同様に、マシンのファームウェアによって実行され得る。マシンは、セキュア・インターフェイス制御モードに移行せず、代わりにマシンは、現在実行されているモードでセキュア・インターフェイス制御機能を実行する。ハードウェアは、これらの動作を処理するためのコンテキストの切り替えがないように、ファームウェアとソフトウェアの両方の状態を維持する。この少ないオーバーヘッドによって、必要なレベルのセキュリティを引き続き提供しながら、セキュア・インターフェイス制御における複雑さを最小限に抑えて減らすような方法で、ソフトウェア、信頼できるファームウェア、およびハードウェアの異なるレイヤ間の緊密な連携を可能にする。

【0027】

本発明の1つまたは複数の実施形態に従って、セキュア・ゲストを適切に維持するためにセキュア・インターフェイス制御およびハードウェアによって必要とされる制御ブロック構造の支援、およびハイパーバイザ環境の支援において、ハイパーバイザは、セキュア・ゲスト環境を初期化しながら、ストレージをセキュア・インターフェイス制御に提供す

る。その結果、（１）セキュア・ゲストを実行するようにゾーンを初期化すること、（２）セキュア・ゲスト・ドメインを作成すること、および（３）ドメインの各々において実行されるセキュアCPUを作成することに備えて、ハイパーバイザは、特に、提供する必要のあるストレージの量を決定するために、UVC照会命令を発行する。ストレージが提供された後に、このストレージは、セキュアとしてマーク付けされ、セキュア・インターフェイス制御に属しているとして登録され、どの非セキュア・ゲストまたはセキュア・ゲストの実体によるアクセスも禁止される。関連する実体（例えば、セキュア・ゲストCPU、セキュア・ゲスト・ドメインまたはゾーン）が破棄されるまで、この状況が維持される。

【0028】

1つの例では、ゾーン固有のUV制御ブロックを支援するためのUVストレージの第1のセクションが、UVC初期化の一部としてセキュア・インターフェイス制御に提供され、本明細書ではUV2ストレージと呼ばれるものに存在する。（セキュア・ゲスト・ドメインごとに）ベースおよび可変セキュア・ゲスト構成制御ブロック（base and variable secure-guest-configuration control blocks）を支援するためのUVストレージの第2および第3のセクションが、セキュア・ゲスト構成作成UVC（create-secure-guest-configuration UVC）の一部として提供され、UVSストレージおよびUVVストレージにそれぞれ存在する。セキュアCPU制御ブロックを支援するためのUVストレージの第4および最後のセクションも、UVS空間に存在し、セキュア・ゲストCPU作成UVC（create-secure-guest-CPU UVC）の一部として提供される。これらの領域の各々が提供されるときに、セキュア制御インターフェイスは、これらの領域を、（すべてのセキュアでない実体によるアクセスを防ぐために）セキュアとしてマーク付けし、（すべてのセキュア・ゲストの実体によるアクセスを防ぐために）セキュア・インターフェイス制御に属しているとしてゾーン・セキュリティ・テーブルにも登録する。UV空間内のさらなる分離を提供するために、UV2空間（どの特定のセキュア・ゲスト・ドメインにも関連付けられていない）も、一意のUV2セキュア・ドメインでタグ付けされ、一方、UVS空間およびUVV空間の両方が、関連する特定のセキュア・ゲスト・ドメインでさらにタグ付けされる。この例では、UVV空間がホスト仮想空間に存在し、したがって、ホスト仮想からホスト絶対へのマッピングによってさらに識別され得る。

【0029】

セキュア・インターフェイス制御はすべてのストレージ（非セキュア・ストレージ、セキュア・ゲスト・ストレージ、およびUVストレージ）にアクセスすることができるが、本発明の1つまたは複数の実施形態は、非常に具体的には、セキュア・インターフェイス制御がUVストレージにアクセスすることを許可するメカニズムを提供する。本発明の実施形態は、セキュア・ゲスト・ドメイン間の分離を提供する同じハードウェア・メカニズムを使用して、UVストレージ内で同様の分離を提供することができる。これによって、セキュア・インターフェイス制御が、意図され、指定された場合にのみUVストレージにアクセスすること、望ましい指定されたセキュア・ゲストのセキュア・ゲスト・ストレージのみにアクセスすること、および指定された場合にのみ非セキュア・ストレージにアクセスすることを保証する。すなわち、セキュア・インターフェイス制御は、アクセスしようとするストレージを極めて明示的に指定することができ、ハードウェアが、そのストレージに実際にアクセスすることを保証できるようにする。加えて、セキュア・インターフェイス制御は、指定されたセキュア・ゲスト・ドメインに関連付けられたUVストレージのみにアクセスしようとすることを、さらに指定することができる。

【0030】

セキュリティを提供するために、ハイパーバイザがセキュア・ゲストのデータを透過的にページインおよびページアウトしているときに、ハードウェアと連携しているセキュア・インターフェイス制御は、データの復号および暗号化を提供し、保証する。これを実現するために、ハイパーバイザは、ゲストのセキュア・データをページインおよびページアウトするときに、新しいUVCを発行する必要がある。ハードウェアは、これらの新しい

10

20

30

40

50

UVCの間にセキュア・インターフェイス制御によって設定された制御に基づいて、これらのUVCがハイパーバイザによって実際に発行されることを保証する。

【0031】

この新しいセキュアな環境では、ハイパーバイザは、セキュア・ページをページアウトしているときに常に、新しいセキュア・ストレージからの変換（エクスポート）UVCを発行する必要がある。セキュア・インターフェイス制御は、このエクスポートUVCに 응답して、（１）ページがUVによって「ロックされている」ことを示し、（２）ページを暗号化し、（３）ページを非セキュアに設定し、（４）UVのロックをリセットする。エクスポートUVCが完了した後に、ハイパーバイザは、次に暗号化されたゲスト・ページをページアウトできるようになる。

10

【0032】

加えて、ハイパーバイザは、セキュア・ページにページインしているときに常に、新しいセキュア・ストレージへの変換（インポート）UVCを発行しなければならない。UVまたはセキュア・インターフェイス制御は、このインポートUVCに 응답して、（１）ページをハードウェア内でセキュアとしてマーク付けし、（２）ページがUVによって「ロックされている」ことを示し、（３）ページを復号し、（４）特定のセキュア・ゲスト・ドメインに対する権限を設定し、（５）UVのロックをリセットする。アクセスがセキュアな実体によって行われるときに常に、変換中にハードウェアは、そのページに対して許可チェックを実行する。これらのチェックは、（１）ページが、このページにアクセスしようとしているセキュア・ゲスト・ドメインに実際に属していることを検証するためのチェック、および（２）このページがゲスト・メモリに常駐している間にハイパーバイザがこのページのホストのマッピングを変更していないことを確認するためのチェックを含む。ページがセキュアとしてマーク付けされた後に、ハードウェアは、ハイパーバイザまたは非セキュア・ゲストVMのいずれかによるすべてのセキュア・ページへのアクセスを防ぐ。追加の変換ステップが、別のセキュアVMによるアクセスを防ぎ、ハイパーバイザによる再マッピングを防ぐ。

20

【0033】

ここで図1を参照すると、本発明の1つまたは複数の実施形態に従って、ゾーン・セキュリティのためのテーブル100が概して示されている。図1に示されているゾーン・セキュリティ・テーブル100は、セキュアな実体によってアクセスされるすべてのページへのセキュアなアクセスを保証するために、セキュア・インターフェイス制御によって維持され、セキュア・インターフェイス制御およびハードウェアによって使用される。ゾーン・セキュリティ・テーブル100は、ホスト絶対アドレス110によってインデックス付けされる。すなわち、ホスト絶対ストレージのページごとに1つのエントリが存在する。各エントリは、アクセスを行っているセキュアな実体に属しているとしてエントリを検証するために使用される情報を含んでいる。

30

【0034】

さらに、図1に示されているように、ゾーン・セキュリティ・テーブル100は、セキュア・ドメインID120（ページに関連付けられたセキュア・ドメインを識別する）と、UVビット130（このページがセキュア・インターフェイス制御に提供されており、セキュア・インターフェイス制御によって所有されていることを示す）と、アドレス比較無効化（DA）ビット140（ホスト絶対として定義されたセキュア・インターフェイス制御のページに関連するホスト仮想アドレスを有していないなどの場合に、特定の環境内でホスト・アドレス対の比較を無効化するために使用される）と、共有（SH）ビット150（ページが非セキュア・ハイパーバイザと共有されていることを示す）と、ホスト仮想アドレス160（このホスト絶対アドレスの登録されたホスト仮想アドレスを示し、これらのアドレスはホスト・アドレス対と呼ばれる）とを含んでいる。ホスト・アドレス対が、ホスト絶対アドレスと、関連する登録されたホスト仮想アドレスとを示すということに注意する。ホスト・アドレス対は、ハイパーバイザによってインポートされた後の、このページのマッピングを表し、比較は、このページがゲストによって使用されている間に

40

50

、ホストがこのページを再マッピングしていないことを保証する。

【0035】

動的アドレス変換（D A T）は、仮想ストレージを実ストレージにマッピングするために使用される。ゲストV Mがハイパーバイザの制御下でページング可能なゲストとして実行されている場合、ゲストは、D A Tを使用してメモリに常駐するページを管理する。加えて、ホストは、ゲスト・ページがメモリに常駐しているときに、独立してD A Tを使用して、それらのゲスト・ページを（ホスト自身のページと共に）管理する。ハイパーバイザは、D A Tを使用して、異なるV M間のストレージの分離または共有あるいはその両方を提供するだけでなく、ハイパーバイザ・ストレージへのゲストのアクセスを防ぐ。ハイパーバイザは、ゲストが非セキュア・モードで実行されているときに、すべてのゲストのストレージにアクセスすることができる。

10

【0036】

D A Tは、アプリケーションが共有リソースを共有することを引き続き許可しながら、アプリケーション間の分離を可能にする。また、D A TはV Mの実装を許可し、V Mは、アプリケーション・プログラムの同時処理と共に、O Sの新しいバージョンの設計およびテストにおいて使用することができる。仮想アドレスは、仮想ストレージ内の位置を識別する。アドレス空間は、連続する一連の仮想アドレスであり、各仮想アドレスに関連する絶対アドレスに変換できるようにする特定の変換パラメータ（D A Tテーブルを含む）を伴っており、絶対アドレスは、ストレージ内のバイト位置でそのアドレスを識別する。

【0037】

D A Tは、複数の検索テーブルを使用して、仮想アドレスに関連する絶対アドレスに変換する。このテーブル構造は、通常、ストレージ・マネージャによって定義され、維持される。このストレージ・マネージャは、例えば、あるページをページアウトし、別のページを取り込むことによって、複数のプログラム間で絶対ストレージを透過的に共有する。ページがページアウトされるときに、ストレージ・マネージャは、例えば、関連するページ・テーブル内で無効ビットを設定する。プログラムが、ページアウトされたページにアクセスしようとするときに、ハードウェアがプログラム割り込み（多くの場合、ページ・フォールトと呼ばれる）をストレージ・マネージャに提示する。それに応じて、ストレージ・マネージャは、要求されたページをページインし、無効ビットをリセットする。これは、プログラムにとってすべて透過的に実行され、ストレージ・マネージャがストレージを仮想化し、さまざまな異なるユーザ間で共有することを可能にする。

20

【0038】

C P Uによって仮想アドレスが使用されて主記憶装置にアクセスする場合、仮想アドレスは、まずD A Tを用いて実アドレスに変換され、次にプレフィックス変換を用いて絶対アドレスに変換される。特定のアドレス空間に対する最高レベルのテーブルの指定（原点および長さ）は、アドレス空間制御要素（A S C E : address-space-control element）と呼ばれ、関連するアドレス空間を定義する。

30

【0039】

ここで図2を参照すると、本発明の1つまたは複数の実施形態に従って、D A Tを実行するための例示的な仮想アドレス空間202および204ならびに絶対アドレス空間206が概して示されている。図2に示されている例では、仮想アドレス空間202（アドレス空間制御要素（A S C E）A208によって定義される）および仮想アドレス空間204（A S C E B210によって定義される）という2つの仮想アドレス空間が存在する。ストレージ・マネージャによって、A S C E A208を使用して、仮想ページA1．V212a1、A2．V212a2、およびA3．V212a3が、複数の検索テーブル（セグメント230およびページ・テーブル232a、232b）内で、絶対ページA1．A220a1、A2．A220a2、およびA3．A220a3にマッピングされる。同様に、A S C E B210を使用して、仮想ページB1．V214b1およびB2．V214b2が、2つの検索テーブル234および236内で、絶対ページB1．A222b1およびB2．A222b2にそれぞれマッピングされる。

40

50

【0040】

ここで図3を参照すると、本発明の1つまたは複数の実施形態に従って、ハイパーバイザの下で実行されているVMを支援するために使用されるネストされたマルチパートDAT変換の例が、概して示されている。図3に示されている例では、ゲストAの仮想アドレス空間A302（ゲストASCE（GASCE）A304によって定義される）およびゲストBの仮想アドレス空間B306（GASCEB308によって定義される）の両方が、共有ホスト（ハイパーバイザ）仮想アドレス空間325に存在する。図に示されているように、ゲストAのストレージ・マネージャによって、GASCEA304を使用して、ゲストAに属している仮想ページA1．GV310a1、A2．GV310a2、およびA3．GV310a3が、ゲスト絶対ページ（guest absolute pages）A1．HV340a1、A2．HV340a2、およびA3．HV340a3にそれぞれマッピングされ、ゲストBのストレージ・マネージャによって、独立してGASCEB308を使用して、ゲストBに属している仮想ページB1．GV320b1およびB2．GV320b2が、ゲスト絶対ページB1．HV360b1およびB2．HV360b2にそれぞれマッピングされる。この例では、これらのゲスト絶対ページは、共有ホスト仮想アドレス空間325に直接マッピングされ、その後、ホスト絶対アドレス空間330への追加のホストDAT変換を受ける。図に示されているように、ホストのストレージ・マネージャによって、ホストASCE（HASCE）350を使用して、ホスト仮想アドレスA1．HV340a1、A3．HV340a3、およびB1．HV360b1が、A1．HA370a1、A3．HA370a3、およびB1．HA370b1にマッピングされる。ゲストAに属しているホスト仮想アドレスA2．HV340a2、およびゲストBに属しているB2．HV360b2の両方が、同じホスト絶対ページ（host absolute page）AB2．HA380にマッピングされる。これによって、これら2つのゲスト間でデータを共有できるようにする。ゲストDAT変換中に、ゲストのテーブル・アドレスの各々が、ゲスト絶対として扱われ、追加のネストされたホストDAT変換を受ける。

【0041】

本明細書に記載された本発明の実施形態は、セキュア・ゲストおよびUVストレージの保護を実現する。非セキュア・ゲストおよびハイパーバイザによるセキュア・ストレージへのアクセスが禁止される。ハイパーバイザは、特定の常駐するセキュア・ゲスト・ページに関して、次のことを発生させる。関連するホスト絶対アドレスが、単一のハイパーバイザ（ホスト）DATマッピングのみによってアクセス可能になる。すなわち、セキュア・ゲストに割り当てられた特定のホスト絶対アドレスにマッピングされる単一のホスト仮想アドレスが存在する。特定のセキュア・ゲスト・ページに関連付けられたハイパーバイザの（ホスト仮想からホスト絶対への）DATマッピングは、このページがページインされている間に変化しない。セキュア・ゲスト・ページに関連付けられたホスト絶対ページは、単一のセキュア・ゲストに関してマッピングされる。

【0042】

本発明の1つまたは複数の実施形態に従って、セキュア・ゲスト間のストレージの共有も禁止される。ストレージは、セキュア・ゲストの制御下で、単一のセキュア・ゲストとハイパーバイザの間で共有される。UVストレージは、セキュア・ストレージであり、セキュア・インターフェイス制御によってアクセス可能であるが、ゲスト／ホストによるアクセスは不可能である。ストレージは、ハイパーバイザによってセキュア・インターフェイス制御に割り当てられる。本発明の1つまたは複数の実施形態によれば、これらのルールの試みられたすべての違反が、ハードウェアおよびセキュア・インターフェイス制御によって禁止される。

【0043】

ここで図4を参照すると、本発明の1つまたは複数の実施形態に従って、セキュア・ゲスト・ストレージのマッピングの例が概して示されている。図4は図3に似ているが、図4の例が、セキュア・ゲストAとセキュア・ゲストBの間のストレージの共有を可能にしない点が異なっている。図3のセキュアでない例では、ゲストAに属しているホスト仮想

アドレス A 2 . H V 3 4 0 a 2、およびゲスト B に属している B 2 . H V 3 6 0 b 2 の両方が、同じホスト絶対ページ A B 2 . H A 3 8 0 にマッピングされる。図 4 のセキュア・ゲスト・ストレージの例では、ゲスト A に属しているホスト仮想アドレス A 2 . H V 3 4 0 a 2 がホスト絶対アドレス A 2 . H A 4 9 0 a にマッピングされ、一方、ゲスト B に属している B 2 . H V 3 6 0 b 2 が、それ自身の B 2 . H A 4 9 0 b にマッピングされる。この例では、セキュア・ゲスト間に共有が存在しない。

【 0 0 4 4 】

セキュア・ゲスト・ページは、ディスク上に存在する間、暗号化されている。ハイパーバイザは、セキュア・ゲスト・ページをページインするときに U V 呼び出し (U V C) を発行し、この U V C は、セキュア・インターフェイス制御に、(共有されていない限り) ページをセキュアとしてマーク付けし、(共有されていない限り) 復号し、適切なセキュア・ゲスト (例えば、ゲスト A) に属しているとして (ゾーン・セキュリティ・テーブルに) 登録することを実行させる。加えて、ハイパーバイザは、関連するホスト仮想アドレス (例えば、A 3 . H V 3 4 0 a 3) を、そのホスト絶対ページ (ホスト・アドレス対と呼ばれる) に登録する。ハイパーバイザは、正しい U V C を発行できない場合、セキュア・ゲスト・ページにアクセスしようとするときに、例外を受信する。ハイパーバイザがゲスト・ページをページアウトするときに同様の U V C が発行され、この U V C は、ゲスト・ページを非セキュアとしてマーク付けして、非セキュアとしてゾーン・セキュリティ・テーブルに登録する前に、(共有されていない限り) ゲスト・ページを暗号化する。

【 0 0 4 5 】

5 つの特定のホスト絶対ページ K、P、L、M、および N を含んでいる例では、ハイパーバイザがこれらのホスト絶対ページをページインするときに、セキュア・インターフェイス制御によってホスト絶対ページの各々がセキュアとしてマーク付けされる。これによって、非セキュア・ゲストおよびハイパーバイザがこれらのホスト絶対ページにアクセスするのを防ぐ。ハイパーバイザがホスト絶対ページ K、P、および M をページインするときに、これらのホスト絶対ページが、ゲスト A に属しているとして登録され、ホスト絶対ページ L および N が、ハイパーバイザによってページインされるときに、ゲスト B に登録される。共有ページ (単一のセキュア・ゲストとハイパーバイザの間で共有されたページ) は、ページング中に暗号化も復号も実行されない。これらの共有ページは、セキュアとしてマーク付けされない (ハイパーバイザによるアクセスを許可する) が、単一のセキュア・ゲスト・ドメインと共にゾーン・セキュリティ・テーブルに登録される。

【 0 0 4 6 】

本発明の 1 つまたは複数の実施形態に従って、非セキュア・ゲストまたはハイパーバイザが、セキュア・ゲストによって所有されているページにアクセスしようとするときに、ハイパーバイザがセキュア・ストレージ・アクセス (P I C 3 D) 例外を受信する。これを決定するための追加の変換ステップは不要である。

【 0 0 4 7 】

1 つまたは複数の実施形態に従って、セキュアな実体がページにアクセスしようとするときに、ハードウェアが追加の変換チェックを実行し、ストレージがその特定のセキュア・ゲストに実際に属していることを検証する。ストレージがその特定のセキュア・ゲストに属していない場合、非セキュア・アクセス (P C I 3 E) 例外がハイパーバイザに提示される。加えて、変換されているホスト仮想アドレスが、ゾーン・セキュリティ・テーブル内の登録済みのホスト・アドレス対のホスト仮想アドレスに一致しない場合、セキュア・ストレージ違反 (「 3 F 」 x) 例外が認識される。ハイパーバイザとの共有を可能にするために、セキュア・ゲストは、変換チェックがアクセスを許す限り、セキュアとしてマーク付けされていないストレージにアクセスすることができる。

【 0 0 4 8 】

ここで図 5 を参照すると、本発明の 1 つまたは複数の実施形態に従って、D A T 動作のシステム概略図 5 0 0 が概して示されている。システム概略図 5 0 0 は、一次ホスト仮想アドレス空間 (host primary virtual address space) 5 1 0 およびホーム・ホスト仮

10

20

30

40

50

想アドレス空間 (host home virtual address space) 520を含んでおり、これらの仮想アドレス空間のページが、ハイパーバイザ (ホスト) 絶対アドレス空間530に変換される (例えば、ホストDAT変換525を参照 (点線がDAT変換525によるマッピングを表していることに注意する))。例えば、図5は、2つの異なるホスト仮想アドレス空間によるホスト絶対ストレージの共有を示しており、2つのゲスト間だけでなく、ホスト自体とのそれらのホスト仮想アドレスのうちの1つの共有も示している。これに関して、一次ホスト仮想アドレス空間510およびホーム・ホスト仮想アドレス空間520は、2つのホスト仮想アドレス空間の例であり、これらのホスト仮想アドレス空間の各々は、別々のASCE (一次ホストASCE (HPASCE: host primary ASCE) 591 およびホーム・ホストASCE (HHASCE: host home ASCE) 592) によってそれぞれアドレス指定される。すべてのセキュア・インターフェイス制御のストレージ (仮想および現実の両方) が、ハイパーバイザによって提供され、セキュアとしてマーク付けされるということに注意する。セキュア・インターフェイス制御のストレージは、提供された後に、関連するセキュアな実体が存在する限り、セキュア・インターフェイス制御のみによってアクセスされ得る。

【0049】

図に示されているように、一次ホスト仮想アドレス空間510は、ゲストAの絶対ページA1・HV、ゲストAの絶対ページA2・HV、ゲストBの絶対ページB1・HV、およびホスト仮想ページH3・HVを含んでいる。ホーム・ホスト仮想アドレス空間520は、セキュア・インターフェイス制御の仮想ページU1・HV、ホスト仮想ページH1・HV、およびホスト仮想ページH2・HVを含んでいる。

【0050】

本発明の1つまたは複数の実施形態に従って、すべてのセキュア・ゲスト (例えば、セキュア・ゲストAおよびセキュア・ゲストB) ストレージが、本明細書に記載されたゾーン・セキュリティ・テーブルに、セキュア・ゲスト構成に属しているとして登録され、関連するホスト仮想アドレス (例えば、A1・HV、A2・HV、B1・HV) も、ホスト・アドレス対の一部として登録される。1つまたは複数の実施形態では、すべてのセキュア・ゲスト・ストレージが、一次ホスト仮想空間内でマッピングされる。加えて、すべてのセキュア・インターフェイス制御のストレージが、やはりゾーン・セキュリティ・テーブルに、セキュア・インターフェイス制御に属しているとして登録され、関連するセキュア・ゲスト・ドメインに基づいて、ゾーン・セキュリティ・テーブル内でさらに区別されてよい。本発明の1つまたは複数の実施形態に従って、UV仮想ストレージが、ホーム・ホスト仮想空間 (host home virtual space) 内でマッピングされ、関連するホスト仮想アドレスがホスト・アドレス対の一部として登録される。1つまたは複数の実施形態に従って、UV実ストレージが、関連するホスト仮想マッピングを有しておらず、ゾーン・セキュリティ・テーブル内のDAビット (仮想アドレスの比較が無効化されることを示す) が、そのことを示すように設定される。ホスト・ストレージが、非セキュアとしてマーク付けされ、非セキュアとしてゾーン・セキュリティ・テーブルにも登録される。

【0051】

したがって、「ゲスト絶対=ホスト仮想」である場合、ハイパーバイザ (ホスト) 一次DATテーブル (HPASCE 591によって定義される) は、一次ホスト仮想アドレス空間510のページを、次のように変換する。ゲストAの絶対ページA1・HVが、セキュア・ゲストAに属しているホスト絶対A1・HAにマッピングされ、ゲストAの絶対ページA2・HVが、セキュア・ゲストAに属しているホスト絶対A2・HAにマッピングされ、ゲストBの絶対ページB1・HAが、セキュア・ゲストBに属しているホスト絶対B1・HAにマッピングされ、ホスト仮想ページH3・HVが、ホスト絶対ページH3・HA 非セキュア・ホストにマッピングされる (非セキュアであるため、ホスト・アドレス対が存在しない)。さらに、ハイパーバイザ (ホスト) ホームDATテーブル (HHASCE 592によって定義される) が、ホーム・ホスト仮想アドレス空間520のページを、次のように変換する。セキュア・インターフェイス制御の仮想ページU1・HVが、

セキュアUV仮想として定義されたホスト絶対ページU1．HAにマッピングされ、ホスト仮想ページH1．HVが、非セキュアとして定義されたホスト絶対ページH1．HAにマッピングされ、ホスト仮想ページH2．HVが、非セキュアとして定義されたホスト絶対ページH2．HAにマッピングされる。H1．HAおよびH2．HAは非セキュアであるため、これらのいずれかに関連付けられたホスト・アドレス対は存在しない。

【0052】

動作中に、セキュア・ゲストが、セキュア・インターフェイス制御に割り当てられたセキュア・ページにアクセスしようとした場合、ハードウェアによってセキュア・ストレージ違反（「3F」X）例外がハイパーバイザに提示される。非セキュア・ゲストまたはハイパーバイザが、いずれかのセキュア・ページ（セキュア・インターフェイス制御に割り当てられたセキュア・ページを含む）にアクセスしようとした場合、ハードウェアによってセキュア・ストレージ・アクセス（「3D」X）例外がハイパーバイザに提示される。代替として、セキュア・インターフェイス制御の空間に対して試みられたアクセスに関して、エラー状態が提示され得る。ハードウェアが、セキュア・インターフェイス制御のアクセス時に、セキュアな割り当てにおいて不一致を検出した場合（例えば、ストレージが、セキュア・インターフェイス制御ではなくセキュア・ゲストに属しているとしてゾーン・セキュリティ・テーブルに登録されたか、または使用されているホスト・アドレス対に、登録済みの対との不一致が存在する場合）、チェックが提示される。

【0053】

言い換えると、一次ホスト仮想アドレス空間510は、ホスト仮想ページA1．HVおよびA2．HV（セキュア・ゲストAに属している）ならびにB1．HV（セキュア・ゲストBに属している）を含んでおり、これらのホスト仮想ページは、ホスト絶対A1．HA、A2．HA、およびB1．HAにそれぞれマッピングされる。加えて、一次ホスト仮想アドレス空間510は、ホスト絶対H3．HAにマッピングされるホスト（ハイパーバイザ）ページH3．HVを含んでいる。ホーム・ホスト仮想空間520は、ホスト絶対ページH1．HAおよびH2．HAにマッピングされる2つのホスト仮想ページH1．HVおよびH2．HVを含んでいる。一次ホスト仮想アドレス空間510およびホーム・ホスト仮想アドレス空間520の両方は、単一のホスト絶対530にマッピングされる。セキュア・ゲストAおよびセキュア・ゲストBに属しているストレージ・ページは、セキュアとしてマーク付けされ、それらのセキュア・ドメインおよび関連するホスト仮想アドレスと共に、図1に示されているゾーン・セキュリティ・テーブル100に登録される。一方、ホスト・ストレージは、非セキュアとしてマーク付けされる。ハイパーバイザは、セキュア・ゲストを定義している場合、これらのセキュア・ゲストの支援において必要なセキュア制御ブロック（secure control blocks）に使用するために、ホスト・ストレージをセキュア・インターフェイス制御に提供しなければならない。このストレージは、ホスト絶対空間またはホスト仮想空間のいずれかにおいて、および1つの例では、特に、ホーム・ホスト仮想空間において定義され得る。図5に戻り、ホスト絶対ページU1．HAおよびU2．HA セキュアUV絶対は、ホスト絶対ストレージとして定義されているセキュア・インターフェイス制御のストレージである。その結果、これらのページは、セキュアとしてマーク付けされ、セキュア・インターフェイス制御に属しているとして、関連するセキュア・ドメインと共に、図1に示されているゾーン・セキュリティ・テーブル100に登録される。これらのページはホスト絶対アドレスとして定義されるため、関連するホスト仮想アドレスが存在せず、そのためゾーン・セキュリティ・テーブル100内のDAビットが設定される。

【0054】

変換後のハイパーバイザ（ホスト）絶対アドレス空間530の例が、図6で見出すことができる。図6では、本発明の1つまたは複数の実施形態に従って、セキュア・インターフェイス制御メモリに関するシステム概略図600が示されている。システム概略図600は、ホスト絶対ページA2．HA セキュア・ゲストA（A2．HV用）、ホスト絶対ページB1．HA セキュア・ゲストB（B1．HV用）、ホスト絶対ページH1．HA

10

20

30

40

50

非セキュア（ホスト）、ホスト絶対ページ H 2 . H A 非セキュア（ホスト）、ホスト絶対ページ U 3 . H A セキュア U V 現実（H V マッピングなし）、ホスト絶対ページ U 1 . H A セキュア U V 仮想（U 1 . H V 用）、およびホスト絶対ページ A 1 . H A セキュア・ゲスト A（A 1 . H V 用）を含んでいるハイパーバイザ（ホスト）絶対アドレス空間 6 3 0 を示している。

【0055】

ここで図 7 を参照すると、本発明の 1 つまたは複数の実施形態に従って、インポート動作のプロセス・フロー 7 0 0 が概して示されている。セキュア・ゲストが、ハイパーバイザによってページアウトされたページにアクセスするときに、そのページを安全に取り戻すために、プロセス・フロー 7 0 0 に示されているイベントなどの一連のイベントが発生する。プロセス・フロー 7 0 0 はブロック 7 0 5 で開始し、ブロック 7 0 5 で、セキュア・ゲストがゲスト仮想ページにアクセスする。例えばこのページが無効であるため、ハードウェアが、プログラム割り込みコード 1 1（P I C 1 1）によって示されたホスト・ページ・フォールトをハイパーバイザに提示する（ブロック 7 1 5 を参照）。次に、ハイパーバイザは、このゲスト・ページの使用可能なセキュアでないホスト絶対ページを識別し（ブロック 7 2 0 を参照）、暗号化されたゲスト・ページを識別されたホスト絶対ページにページインする（ブロック 7 2 5 を参照）。

【0056】

ブロック 7 3 0 で、次にホスト絶対ページが、（ホスト仮想アドレスに基づいて）適切なホスト D A T テーブル内でマッピングされる。ブロック 7 3 5 で、次にハイパーバイザ（ホスト）が、セキュア・ゲストを再ディスパッチする。ブロック 7 4 0 で、セキュア・ゲストがセキュア・ゲスト・ページに再アクセスする。ページ・フォールトはすでに存在しないが、このセキュア・ゲストのアクセスおよびページが、図 1 のゾーン・セキュリティ・テーブル 1 0 0 内でセキュアとしてマーク付けされていないため、ブロック 7 4 5 で、ハードウェアが非セキュア・ストレージ例外（P I C 3 E）をハイパーバイザに提示する。この P I C 3 E は、必要なインポートが発行されるまで、ゲストによるこのセキュア・ページへのアクセスを防ぐ。次に、プロセス・フロー 7 0 0 は、図 8 に接続されている「A」に進む。

【0057】

ここで図 8 を参照すると、本発明の 1 つまたは複数の実施形態に従って、インポート動作を実行するためのプロセス・フロー 8 0 0 が概して示されている。正常に動作する（例えば、エラーのない期待される方法で動作している）ハイパーバイザが、P I C 3 E に応答して、インポート U V C を発行する（ブロック 8 0 5 を参照）。この時点で、インポートされるページが、非セキュアとしてマーク付けされ、ハイパーバイザ、他のセキュアでない実体、およびセキュア・インターフェイス制御のみによってアクセス可能であるということに注意する。セキュア・ゲストによって、このページにアクセスすることはできない。

【0058】

インポート U V C の一部として、セキュア・インターフェイス制御として機能する信頼できるファームウェアが、セキュア・インターフェイス制御によってこのページがすでにロックされているかどうかをチェックして確認する（判定ブロック 8 1 0 を参照）。このページがロックされている場合、プロセス・フロー 8 0 0 がブロック 8 2 0 に進む。ブロック 8 2 0 で、「ビジー」復帰コードがハイパーバイザに返され、それに応じてハイパーバイザは、遅延し（ブロック 8 2 5 を参照）、インポート U V C を再発行する（プロセス・フロー 8 0 0 がブロック 8 0 5 に戻る）。このページがまだロックされていない場合、プロセス・フロー 8 0 0 が判定ブロック 8 2 2 に進む。

【0059】

判定ブロック 8 2 2 で、セキュア・インターフェイス制御が、このページが、非セキュア・ハイパーバイザと共有されたページであるかどうかをチェックして確認する。このページが共有されている場合（プロセス・フロー 8 0 0 が判定ブロック 8 2 4 に進む）、セ

セキュア・インターフェイス制御が、ホスト絶対アドレスを、関連するセキュア・ゲスト・ドメイン、ホスト仮想アドレスと共に、共有されているとしてゾーン・セキュリティ・テーブルに登録する。このページは、非セキュアとしてマーク付けされたままである。これでインポートUVCが完了し、ゲストによってこのページにアクセスできるようになった。処理は、ハイパーバイザがゲストを再ディスパッチすること（ブロック830）、およびセキュア・ゲストがこのページに正常にアクセスすること（ブロック835）に進む。

【0060】

インポートされるホスト仮想ページがハイパーバイザと共有されていない場合（プロセス・フロー800がブロック840に進む）、セキュア・インターフェイス制御は、ハイパーバイザがこのページにアクセスできなくなるように、このページをセキュアとしてマーク付けする。ブロック845で、セキュア・インターフェイス制御は、他のUVCがページの状態を変更できないように、ページをロックする。（ブロック850で）ロックが設定された後に、セキュア・インターフェイス制御は、ゲスト・ページの内容が、暗号化されている間に変化しなかったことを検証する。ゲスト・ページの内容が変化していた場合、エラー復帰コードがハイパーバイザに返され、そうでない場合、セキュア・インターフェイス制御がセキュア・ページを復号する。

【0061】

ブロック855で、セキュア・インターフェイス制御がページのロックを解除して、他のUVCによるアクセスを許可し、ページを、セキュアとして、HV→HAホスト・アドレス対を完成させるための適切なゲスト・ドメインおよびホスト仮想アドレスに関連付けて、ゾーン・セキュリティ・テーブルに登録する。これによって、ゲストによるアクセスを許可し、UVCを完了する。

【0062】

ここで図9を参照すると、本発明の1つまたは複数の実施形態に従って、提供されたメモリの動作に関するプロセス・フロー900が概して示されている。プロセス・フロー900はブロック905で開始し、ブロック905で、ハイパーバイザが照会UVCをセキュア・インターフェイス制御に発行する。ブロック910で、セキュア・インターフェイス制御がデータ（例えば、照会UVC）を返す。このデータは、ゾーン固有のベース・ホスト絶対ストレージの必要な量、セキュア・ゲスト・ドメイン固有のベース・ホスト絶対ストレージの必要な量、MBごとのセキュア・ゲスト・ドメイン固有の可変ホスト仮想ストレージの必要な量、またはセキュア・ゲストCPU固有のベース・ホスト絶対ストレージの必要な量、あるいはその組み合わせを含むことができる。

【0063】

ブロック915で、ハイパーバイザは、ホスト絶対ゾーン固有のベース・ストレージを（例えば、照会UVCによって返されたサイズに基づいて）確保する。ブロック920で、ハイパーバイザが、初期化をセキュア・インターフェイス制御に発行する。これに関して、ハイパーバイザは、ゾーン全体のセキュア・ゲスト構成間の調整に必要とされるUV制御ブロックのために提供されるストレージを提供する、初期化UVCを発行することができる。初期化UVCは、ゾーン固有のベース・ストレージの原点を指定する。

【0064】

ブロック925で、セキュア・インターフェイス制御が、提供されたストレージをUVに登録し、セキュアとしてマーク付けすることによって、初期化（例えば、初期化UVC）を実施する。初期化UVCの場合、セキュア・インターフェイス制御は、提供されたストレージをセキュアとしてマーク付けし、その提供されたストレージの一部をゾーン・セキュリティ・テーブルに割り当て、提供されたストレージを、一意のセキュア・ドメインと共に、ただし関連するセキュア・ゲスト・ドメインなしで、関連するホスト仮想アドレス対を有していないとして、UV使用のためのゾーン・セキュリティ・テーブルに登録することができる。

【0065】

ブロック930で、ハイパーバイザがストレージ（例えば、セキュア・ゲスト・ドメイ

10

20

30

40

50

ン固有のベースおよび可変ストレージ）を確保する。例えば、ハイパーバイザは、（例えば、セキュア・ゲスト・ドメインのストレージのサイズに基づいて）セキュア・ゲスト・ドメイン固有のベースおよび可変ストレージ（例えば、照会UVCによって返されたサイズ）を確保する。ブロック935で、ハイパーバイザが、構成作成をセキュア・インターフェイス制御に発行する。これに関して、ハイパーバイザは、セキュア・ゲスト・ドメイン固有のベースおよび可変ストレージの原点を指定するセキュア・ゲスト構成作成UVCを発行することができる。さらに、セキュア・ゲスト構成作成UVCは、このセキュア・ゲスト構成を支援するために必要なUV制御ブロックのために提供されるストレージを提供する。

【0066】

ブロック940で、セキュア・インターフェイス制御が、構成作成（例えば、セキュア・ゲスト構成作成UVC）を実施する。セキュア・ゲスト構成作成UVCの場合、セキュア・インターフェイス制御は、提供されたストレージをセキュアとしてマーク付けし、提供されたストレージをUV使用のためのゾーン・セキュリティ・テーブルに登録し、提供されたストレージに関連するセキュア・ゲスト・ドメインと共に登録することができる。提供されたベース（ホスト絶対）ストレージは、関連するホスト仮想アドレス対を有していないとして登録される。提供された可変（ホスト仮想）ストレージは、関連するホスト仮想アドレス対と共に登録される。

【0067】

ブロック945で、ハイパーバイザは、セキュア・ゲストCPU固有のベース・ストレージ（例えば、照会UVによって返されたサイズ）を確保する。ブロック950で、ハイパーバイザがストレージの原点を指定する。例えば、ハイパーバイザは、セキュア・ゲストCPU固有のベース・ストレージの原点を指定するセキュア・ゲストCPU作成をUVに発行する。ブロック955で、セキュア・インターフェイス制御が、CPU作成（例えば、セキュア・ゲストCPU作成UVC）を実施する。セキュア・ゲストCPU作成UVCの場合、セキュア・インターフェイス制御は、提供されたストレージをセキュアとしてマーク付けし、提供されたストレージを、関連するセキュア・ゲスト・ドメインなしで、関連するホスト仮想アドレス対を有していないとして、UV使用のためのゾーン・セキュリティ・テーブルに登録することができる。

【0068】

ここで図10を参照すると、本発明の1つまたは複数の実施形態に従って、セキュア・インターフェイス制御のセキュア・ページへの非セキュア・ハイパーバイザ・ページの遷移に関するプロセス・フロー1000が概して示されている。プロセス・フロー1000では、3つのハイパーバイザのページ（例えば、非セキュア・ハイパーバイザのページA、非セキュア・ハイパーバイザのページB、および非セキュア・ハイパーバイザのページC）が示されている。

【0069】

ハイパーバイザ（非セキュア）のページA、B、およびCは、セキュアでない実体（ハイパーバイザを含む）によってアクセスされ得る。さらに、ハイパーバイザ（非セキュア）のページA、B、およびCは、非セキュアおよび非共有としてゾーン・セキュリティ・テーブル（例えば、図1に示されているゾーン・セキュリティ・テーブル100）に登録されるのと同時に、非セキュアとして（NS：non-secure）としてマーク付けされる。矢印1005で、初期化UVCが発行され、ゲスト・ページAを、ゾーン全体（UV2）に関連付けられたセキュア・インターフェイス制御の実ストレージ・ページ1010に遷移させる。セキュア・インターフェイス制御の実ストレージ1010は、UVとして、セキュア・ゲスト・ドメインなし、かつ絶対（HV→HA）マッピングをホストするためのハイパーバイザなしで、ゾーン・セキュリティ・テーブル（例えば、図1に示されているゾーン・セキュリティ・テーブル100）に登録されるのと同時に、セキュアとしてとてマーク付けされ得る。代わりに、実ストレージ1010は、一意のUV2セキュア・ドメインと共に登録され、DAビットが1に設定される。セキュア・インターフェイス制

10

20

30

40

50

御の実ストレージ1010が、セキュア・インターフェイス制御によって現実としてアクセスされ得るということに注意する。

【0070】

ハイパーバイザ（非セキュア）のページBから、矢印1025で、SG構成作成またはSG-CPU作成UV Cが発行され、このページを、セキュア・ゲスト・ドメイン（UV S）に関連付けられたセキュア・インターフェイス制御の実ストレージ1030に遷移させる。セキュア・インターフェイス制御の実ストレージ1030は、UVとして、関連するセキュア・ゲスト・ドメインと共に、絶対（HV→HA）マッピングをホストするためのハイパーバイザなし（すなわち、DAビット=1）で、ゾーン・セキュリティ・テーブル（例えば、図1に示されているゾーン・セキュリティ・テーブル100）に登録されるのと同時に、セキュアとしてとしてマーク付けされ得る。セキュア・インターフェイス制御の実ストレージ1010が、セキュア・ゲスト・ドメインの代わりに、セキュア・インターフェイス制御によって現実としてアクセスされ得るということに注意する。

10

【0071】

ハイパーバイザ（非セキュア）のページCから、矢印1045で、SG構成作成UV Cが発行され、このページを、セキュア・ゲスト・ドメイン（UV V）に関連付けられたセキュア・インターフェイス制御の仮想ストレージ1050に遷移させる。セキュア・インターフェイス制御の仮想ストレージ1050は、UVとして、セキュア・ゲスト・ドメイン、および絶対（HV→HA）マッピングをホストするためのハイパーバイザと共に、ゾーン・セキュリティ・テーブル（例えば、図1に示されているゾーン・セキュリティ・テーブル100）に登録されるのと同時に、セキュアとしてとしてマーク付けされ得る。セキュア・インターフェイス制御の仮想ストレージ1050が、セキュア・ゲスト・ドメインの代わりに、UV仮想としてアクセスされ得るということに注意する。

20

【0072】

ここで図11を参照すると、1つまたは複数の実施形態に従って、プログラムまたはセキュア・インターフェイス制御によって行われるセキュア・ストレージ・アクセスに関するプロセス・フロー1100が示されている。この図は、セキュア・インターフェイス制御がゲスト・ストレージまたはセキュア・インターフェイス制御のストレージにアクセスしようとしており、ハードウェアがそのアクセスのセキュリティを検証できるようにするために、そのアクセスに正しくタグ付けしなければならない状況を表している。1100は、セキュア・インターフェイス制御によるストレージ・アクセスのこのタグ付けを表している。プロセス・フロー1100はブロック1110で開始し、ブロック1110で、セキュア・インターフェイス制御が、セキュア・インターフェイス制御のストレージへのアクセスを行っているかどうかを判定する。

30

【0073】

このアクセスが、セキュア・インターフェイス制御のストレージへのアクセスでない場合、プロセス・フロー1100が判定ブロック1112に進む（「いいえ」の矢印で示されている）。判定ブロック1112で、セキュア・インターフェイス制御が、セキュア・ゲスト・ストレージへのアクセスを行っているかどうかを判定する。このアクセスが、セキュア・ゲスト・ストレージへのアクセスでない場合、プロセス・フロー1100が、非セキュア・アクセスのデフォルト設定を使用する「B」（図12のプロセス・フロー1200に接続されている）に進む。このアクセスが、セキュア・ゲスト・ストレージへのアクセスである場合、プロセス・フロー1100が判定ブロック1113に進み、判定ブロック1113で、セキュア・インターフェイス制御が、デフォルトのセキュア・ゲスト・ドメインが使用されているかどうかを判定する。デフォルトのセキュア・ゲスト・ドメインが使用されている場合、プロセス・フロー1100が、セキュア・ゲスト・アクセスのデフォルト設定を使用する「B」（図12のプロセス・フロー1200に接続されている）に進む。デフォルトのセキュア・ゲスト・ドメインが使用されていない場合、プロセス・フロー1100がブロック1114に進む。ブロック1114で、適切なセキュア・ゲスト・ドメインがSGセキュア・ドメイン・レジスタに読み込まれる（図12のプロセス

40

50

・フロー１２００に接続されている「Ｂ」に進む）。

【００７４】

このアクセスが、セキュア・インターフェイス制御のストレージへのアクセスである場合、プロセス・フロー１１００がブロック１１２０に進む（「はい」の矢印で示されている）。ブロック１１２０で、アクセスが、セキュアＵＶとしてタグ付けされる（例えば、ＵＶセキュア・ドメイン・レジスタを使用する）。

【００７５】

次に、プロセス・フロー１１００が判定ブロック１１３０に進み、判定ブロック１１３０で、セキュア・インターフェイス制御が、このアクセスがＵＶ空間（例えば、ＳＧ構成可変テーブル）へのアクセスであるかどうかを判定する。このアクセスがＵＶ空間へのアクセスである場合、プロセス・フロー１１００がブロック１１３４に進む（「はい」の矢印で示されている）。ブロック１１３４で、アクセスが仮想としてタグ付けされる。ブロック１１３６で、適用可能なセキュア・ゲスト・ドメインがＵＶセキュア・ドメイン・レジスタに読み込まれる。ブロック１１３８で、ＤＡＴ変換およびストレージ・アクセスを開始する準備ができる。判定ブロック１１３０に戻り、このアクセスがＵＶ空間へのアクセスでない場合、プロセス・フロー１１００がブロック１１４０に進む（「いいえ」の矢印で示されている）。ブロック１１４０で、アクセスが現実としてタグ付けされる。

【００７６】

判定ブロック１１５０で、セキュア・インターフェイス制御が、このアクセスがＵＶ空間（例えば、ＳＧ構成またはＣＰＵテーブル）へのアクセスであるかどうかを判定する。このアクセスがＵＶ空間へのアクセスである場合、プロセス・フロー１１００がブロック１１３６に進む（「はい」の矢印で示されている）。このアクセスがＵＶ空間へのアクセスでない場合、プロセス・フロー１１００がブロック１１７０に進む（「いいえ」の矢印で示されている）。このアクセスは、ＵＶ２空間（例えば、ゾーン・セキュリティ・テーブル）へのアクセスである。ブロック１１７０で、一意のＵＶ２セキュア・ドメインがＵＶセキュア・ドメイン・レジスタに読み込まれる。

【００７７】

図１２は、本発明の１つまたは複数の実施形態に従って、プロセス・フロー１２００を示している。ゲストがディスパッチされるときに、ＳＩＥエントリ・ファームウェア（ＳＩＥ Entry firmware）が、ゲストが実行中である（例えば、ゲスト・モードが有効である）ことをハードウェアに示すことができ、ゲストがセキュアであるかどうかを示すことができる。ゲストがセキュアである場合、関連するセキュア・ゲスト・ドメインがハードウェア（例えば、ＳＧセキュア・ドメイン・レジスタ）に読み込まれ得る。プログラムがストレージにアクセスしているときに、ハードウェアが、アクセスの時点でのプログラムの現在の状態に基づいてアクセスにタグ付けすることができる。図１２は、このプロセスの例をプロセス・フロー１２００に示している。ブロック１２０５で、ハードウェアが、マシンがゲスト・モードで現在実行中であるかどうかを判定し、ゲスト・モードで実行中でない場合、ブロック１２１０でホストのアクセスであるとして、およびブロック１２１５で非セキュア・アクセスであるとして、アクセスにタグ付けすることができる。ブロック１２０５で、マシンがゲスト・モードで実行中である場合、ブロック１２２０で、ゲストのアクセスとしてアクセスにタグ付けすることができ、ブロック１２２５で、現在のゲストがセキュア・ゲストであるかどうかをさらに判定する。ゲストがセキュアでない場合、ブロック１２１５で、非セキュアとしてアクセスにタグ付けすることができる。ゲストがセキュアである場合、ブロック１２３０で、ハードウェアがセキュアとしてゲストにタグ付けすることができ、セキュア・ゲストがディスパッチされたときに読み込まれたＳＧセキュア・ドメイン・レジスタに、セキュア・ゲストを関連付けることができる。ブロック１２３５で、非セキュア・ゲストおよびセキュア・ゲストの両方に関して、ＤＡＴの状態をチェックすることができる。ＤＡＴがオフである場合、ブロック１２４０で、現実としてアクセスにタグ付けすることができる。ＤＡＴがオンである場合、ブロック１２４５で、仮想としてアクセスにタグ付けすることができる。アクセスが、ＤＡＴがオフの場合に、

10

20

30

40

50

ブロック1240で現実としてタグ付けされるか、またはDATがオンの場合に、ブロック1245で仮想としてタグ付けされた後に、ブロック1250で、ハードウェアは、図13でさら説明されるように、変換およびストレージ・アクセスを開始する準備ができる。
【0078】

図13は、本発明の1つまたは複数の実施形態に従って、セキュア・アクセスおよび非セキュア・アクセスの両方を支援するためにハードウェアによって実行される変換の例をプロセス・フロー1300に示している。ブロック1305で、ハードウェアは、アクセスがゲスト変換としてタグ付けされているかどうかを判定することができ、アクセスがゲスト変換としてタグ付けされており、ブロック1310で、アクセスが仮想である場合、ブロック1315で、ゲストDATが実行され得る。ゲストDAT変換の間に、ゲストDATテーブルに対するネストされた中間フェッチが存在することができる。テーブルのフェッチが、ゲスト現実として、および元の変換がセキュアとしてタグ付けされている場合はセキュアとして、タグ付けされ得る。テーブルのフェッチも、プロセス・フロー1300の変換プロセスに従うことができる。ブロック1315で、ゲスト仮想としてタグ付けされたアクセスに対してゲストDATが実行され、ブロック1310で、ゲスト現実（仮想=いいえ）としてタグ付けされたアクセスに対してゲストDATが実行された後に、ブロック1320で、ゲスト・プレフィックス変換およびゲスト・メモリ・オフセットが適用され得る。ゲスト変換プロセスの完了時に、ブロック1325で、得られたアドレスが、ホスト仮想として、および元のゲスト変換がセキュアとしてタグ付けされている場合はセキュアとして、タグ付けされ得る。プロセス1300は、ホスト仮想としてタグ付けされた任意のアクセスに関して、続行することができる。元のアクセスが、ブロック1305でホストのアクセスであり（ゲスト=いいえ）、ブロック1330で仮想である場合、ブロック1335でホストDATが実行され得る。ブロック1335で、ホストのテーブルのフェッチが非セキュアとしてマーク付けされ得る。ブロック1335でホストDATが実行された後に、またはブロック1330で元のホストのアクセスが現実（仮想=いいえ）としてタグ付けされた場合に、ブロック1340でホスト・プレフィックス変換が適用され得る。ブロック1345で、得られたアドレスは、ホスト絶対アドレスであることができる。

【0079】

図14は、本発明の1つまたは複数の実施形態に従って、ハードウェアによって実行され得るセキュア・ストレージ保護を伴うDAT変換の例をプロセス・フロー1400に示している。図13のブロック1345から続けて、ブロック1405で、セキュアUVアクセスが識別された場合、ブロック1410で、ハードウェアは、ストレージがセキュアUVストレージとして登録されているかどうかを検証することができ、ストレージがセキュアUVストレージとして登録されていない場合、ブロック1415でエラーが提示される。UVストレージにアクセスしている場合、セキュア・インターフェイス制御によってセキュアUVアクセスが行われ得る。ブロック1410で、ストレージがセキュアUVストレージとして登録されている場合、任意のセキュア・アクセスに対して実行されるように、保護チェックが継続することができるが、処理が継続する場合、ブロック1420で、UVセキュア・ドメイン・レジスタ（セキュアUVアクセスを行う前にセキュア・インターフェイス制御によって設定される）が、ドメイン・チェックのために指定されたセキュア・ドメインとして使用され得るという点が異なる。加えて、ブロック1425でセキュア・ゲスト違反（セキュアUV=いいえ）に対して実行されるような、ブロック1435でのハイパーバイザへの例外ではなく、ブロック1425でUVアクセスに対して検出された任意の違反（エントリ・ポイントD）が、ブロック1430でエラーとして提示され得る。

【0080】

ブロック1405でセキュアUVアクセスとしてタグ付けされていないアクセスの場合、ブロック1440で、ハードウェアは、アクセスがセキュア・ゲストのアクセスであるかどうかを判定し、アクセスがセキュア・ゲストのアクセスでなく、かつブロック144

10

20

30

40

50

5でページがセキュアとしてマーク付けされている場合、ブロック1435で、例外がハイパーバイザに提示され得る。しかし、ブロック1440でアクセスがセキュア・ゲストのアクセスではなく、ブロック1445でページがセキュアとしてマーク付けされていない場合、ブロック1450で変換が成功する。

【0081】

アクセスが、ブロック1440で、セキュア・ゲストのアクセスであるか、またはブロック1410で、セキュアUVストレージとして登録されたストレージへのセキュアUVアクセスである場合、ブロック1420で、ハードウェアは、ストレージがアクセスに関連付けられたセキュアな実体に登録されていることをチェックして確認することができる。アクセスがセキュアUVアクセスである場合、指定されたセキュア・ドメインをUVセキュア・ドメイン・レジスタから取得することができ（セキュアUVストレージがアクセスされているということに基づいて、セキュア・インターフェイス制御によって読み込まれる）、セキュア・ゲストのアクセスの場合、指定されたセキュア・ドメインをSGセキュア・ドメイン・レジスタから取得することができる（セキュアな実体がディスパッチされるときに読み込まれる）。ブロック1420で、アクセスされているストレージが指定されたセキュア・ドメインに登録されていない場合、ブロック1425でセキュアUVアクセスである場合は、ブロック1430でエラーが選択され、ブロック1425でセキュア・ゲストのアクセス（セキュアUV=いいえ）である場合は、ブロック1435で例外がハイパーバイザに提示される。

【0082】

ブロック1420で指定されたセキュア・ドメインに登録された、ブロック1440およびブロック1410でのストレージへのセキュア・アクセスに関して、ブロック1455で仮想アドレス・チェックが無効化されている（すなわち、DAビット=1）であり、かつブロック1460でアクセスが現実である場合、ブロック1450で変換が完了する。しかし、ブロック1455でDAビット=1であるが、ブロック1460でアクセスが仮想（現実=いいえ）である場合、ブロック1425でセキュアUVアクセスである場合は、ブロック1430でエラーが選択され、ブロック1425でセキュア・ゲストのアクセス（セキュアUV=いいえ）である場合は、ブロック1435で例外がハイパーバイザに提示される。ブロック1455でDAビット=0であり、かつブロック1475でアクセスが仮想アクセスである場合、ブロック1470で、ハードウェアは、ホスト仮想からホスト絶対へのアクセスのマッピングが、このホスト絶対アドレスに関して登録されているマッピングに一致するかどうかを判定することができる。一致する場合、ブロック1450で変換が正常に完了する。ブロック1470で、マッピングが一致しない場合、ブロック1425でセキュアUVアクセスである場合は、ブロック1430でエラーが選択され、ブロック1425でセキュア・ゲストのアクセス（セキュアUV=いいえ）である場合は、ブロック1435で例外がハイパーバイザに提示される。DAビット=0であり、かつブロック1475でアクセスが現実のアクセス（仮想=いいえ）である場合、ブロック1425でセキュアUVアクセスである場合は、ブロック1430でエラーが選択され、ブロック1425でセキュア・ゲストのアクセス（セキュアUV=いいえ）である場合は、ブロック1435で例外がハイパーバイザに提示され、代替として、ブロック1450で変換が正常に完了してよい。ブロック1480でのI/Oサブシステムによるアクセスは、ブロック1445でページがセキュアとしてマーク付けされているかどうかをチェックして確認することができ、ページがセキュアである場合、ブロック1435で例外をハイパーバイザに提示することができ、ページがセキュアとしてマーク付けされていない場合、ブロック1450で変換が成功する。

【0083】

ゾーン・セキュリティ・テーブル・インターフェイス1485を介して、ストレージの登録およびマッピングのさまざまなチェックが集散的に管理され得る。例えば、ブロック1410、1420、1455、1470、および1475は、さまざまなアクセスを管理するために同じゾーンに関連付けられているゾーン・セキュリティ・テーブルとインタ

ーフェイスをとることができる。

【0084】

ここで図15を参照すると、本発明の1つまたは複数の実施形態に従って、暗号化の前にページの内容が変化しているかどうかを判定するためのプロセス・フロー1500が概して示されている。プロセス・フロー1500は、図16のデータ・フロー1600を参照してさらに説明される。ブロック1505で、プロセス・フロー1500が開始する。ブロック1510で、コンピュータ・システムのメモリのセキュア・ページのハッシュ値が計算される。例えば、コンピュータ・システムの一部であってよいセキュア・インターフェイス制御1605は、セキュア・ページ1610の内容が非セキュア・ドメイン1625内の非セキュア・ページ1620（すなわち、アクセス可能だが暗号化されている）として提供される前に、セキュア・ドメイン1615内のセキュア・ページ1610のハッシュ関数を計算することができる。ブロック1510のハッシュ関数は、セキュア・ページ1610の1つまたは複数のデータ・ビットにおける変化の検出を支援するためのチェックサムの形態であることができる。セキュア・ページ1610は、非セキュア・ハイパーバイザまたはオペレーティング・システムなどのセキュアでない実体によって、直接アクセス可能でなくてよい。非セキュア・ページ1620は、図7のプロセス・フロー700のブロック725などのさまざまなプロセスによって使用され得る。セキュア・インターフェイス制御1605の暗号化／復号制御1630によって、暗号化および復号が管理され得る。暗号化および復号は、図8のプロセス・フロー800などのさまざまなプロセスの一部としても、呼び出され得る。セキュア・インターフェイス制御1605は、ファームウェア、ハードウェア、またはファームウェアとハードウェアの組み合わせを含むことができる。例えば、セキュア・インターフェイス制御1605は、処理ユニット（例えば、コンピュータ・プロセッサの処理回路）の一部であるか、または処理ユニットによって呼び出し可能であることができる。セキュア・ページ1610を、ホストによって管理されているセキュア・コンテナまたはセキュア仮想マシンに割り当てることができ、ホストは、例えばハイパーバイザまたはオペレーティング・システムである。

【0085】

プロセス・フロー1500のブロック1515で、ブロック1510で計算されたハッシュ値が、ページの以前に計算されたハッシュ値と比較される。例えば、セキュア・インターフェイス制御1605のセキュア・テーブル1635は、関連するページの複数のページ識別子1640およびハッシュ値1645を格納することができる。セキュア・ページ1610のハッシュ値が計算されるときに、セキュア・インターフェイス制御1605は、ページ識別子1640において関連するアドレスの検索を実行し、ハッシュ値1645がセキュア・ページ1610の以前に計算されたハッシュ値を含んでいるかどうかを判定することができる。

【0086】

ハッシュ値が同じページに対して以前に計算されたハッシュ値に一致しないということの決定に基づいて、ブロック1520で、ページごと、暗号化ごとの値の修正値を、ブロック1525で、ページの暗号化において使用することができる。ハッシュ値が同じページに対して以前に計算されたハッシュ値に一致するというものの決定に基づいて、ブロック1525で、暗号化ごとのカウンタ1650の暗号化ごとの値を、変更／インクリメントなしで、ページの暗号化において使用することができる。暗号化ごとのカウンタ1650の値は、ページごとに管理され得る。暗号化ごとのカウンタ1650の初期値は、暗号化ごとの値が使用され、関連するページが参照される前に、確立され得る。例えば、暗号化ごとのカウンタ1650は、暗号化された値と復号された値の間の関係をさらにランダム化するために、暗号化／復号制御1630によって、暗号化または復号の一部として使用され得る。1つの例として、ページを暗号化することは、ページに関連付けられたアドレス値、1つまたは複数の乱数値、および暗号化ごとの値と、暗号によって保護されたハッシュ関数との組み合わせを組み込むことができる。基礎になるデータに対する変更が存在する場合にのみ、暗号化ごとの値を変更することによって、変更されていないデータの

得られた暗号化が、同じデータの以前に実行された暗号化のコピーと一致することができる。ハイパーバイザまたはオペレーティング・システムなどのセキュアでない実体が、非セキュア・ページ 1620 の暗号化された内容を理解できなくても、変更されていない状態を識別することは、変更されていない暗号化データのコピーをさらに作成するのを避けること、または変更されていないページの更新を防ぐこと、あるいはその両方によって、より効率的なメモリ管理を可能にすることができる。ブロック 1530 で、プロセス・フロー 1500 が終了する。

【0087】

本発明の実施形態では、暗号化は、ページをセキュア・ページ 1610 から非セキュア・ページ 1620 に変換するというホスト（例えば、ハイパーバイザまたはオペレーティング・システム）の要求にตอบสนองして、セキュア制御インターフェイス 1605 の暗号化／復号制御 1630 によって実行され得る。暗号化された非セキュア・ページ 1620 は、例えばメモリ管理動作の一部として、格納のためにホストに提供され得る。計算されたハッシュ値は、セキュア制御インターフェイス 1605 のセキュア・テーブル 1635 のハッシュ値 1645 に格納され得る。その後の動作時に、セキュア制御インターフェイス 1605 は、非セキュア・ページ 1620 をセキュア・ページ 1610 に変換することができる、暗号化／復号制御 1630 は、ページに関連付けられた暗号化ごとの値に基づいてセキュア・ページ 1610 を復号し、復号されたページを生成することができる。セキュア・インターフェイス制御 1605 は、復号されたページのハッシュ値を計算し、復号されたページのハッシュ値を、セキュア・テーブル 1635 に格納されたハッシュ値 1645 と比較することができる。復号されたページのハッシュ値が、セキュア・テーブル 1635 に格納されたハッシュ値 1645 のうちの 1 つに一致するということの決定に基づいて、復号されたページの妥当性を確認することができ、妥当性が確認されて復号されたページを、セキュア・ドメイン 1615 においてセキュア・ページ 1610 として使用可能にすることができる。この妥当性確認によって、非セキュア・ページ 1620 が、非セキュア・ドメイン 1625 内にある間に変更されなかった（例えば、ディスクにキャッシュされて取り出されなかった）ことを確認することができる。セキュアと非セキュアの間のページ変換は、ページのアクセス性を制限するために、ページに関連付けられたビットまたはタグを設定することを含むことができる。

【0088】

本開示にはクラウド・コンピューティングに関する詳細な説明が含まれているが、本明細書において示された教示の実装は、クラウド・コンピューティング環境に限定されないと理解されるべきである。むしろ、本発明の実施形態は、現在既知であるか、または今後開発される任意のその他の種類のコンピューティング環境と組み合わせて実装できる。

【0089】

クラウド・コンピューティングは、構成可能な計算リソース（例えば、ネットワーク、ネットワーク帯域幅、サーバ、処理、メモリ、ストレージ、アプリケーション、VM、およびサービス）の共有プールへの便利なオンデマンドのネットワーク・アクセスを可能にするためのサービス提供モデルであり、管理上の手間またはサービス・プロバイダとのやりとりを最小限に抑えて、これらのリソースを迅速にプロビジョニングおよび解放することができる。このクラウド・モデルは、少なくとも 5 つの特徴、少なくとも 3 つのサービス・モデル、および少なくとも 4 つのデプロイメント・モデルを含むことができる。

【0090】

特徴は、次のとおりである。

【0091】

オンデマンドのセルフ・サービス：クラウドの利用者は、サーバの時間およびネットワーク・ストレージなどの計算能力を一方的に、サービス・プロバイダとの人間的なやりとりを必要とせず、必要に応じて自動的にプロビジョニングすることができる。

【0092】

幅広いネットワーク・アクセス：能力は、ネットワークを経由して利用可能であり、標

10

20

30

40

50

準的なメカニズムを使用してアクセスできるため、異種のシン・クライアントまたはシック・クライアント・プラットフォーム（例えば、携帯電話、ラップトップ、およびPDA）による利用を促進する。

【0093】

リソース・プール：プロバイダの計算リソースは、プールされ、マルチテナント・モデルを使用して複数の利用者に提供される。さまざまな物理的および仮想的リソースが、要求に従って動的に割り当ておよび再割り当てされる。場所に依存しないという感覚があり、利用者は通常、提供されるリソースの正確な場所に関して管理することも知ることがもないが、さらに高い抽象レベルでは、場所（例えば、国、州、またはデータセンター）を指定できる場合がある。

10

【0094】

迅速な順応性：能力は、迅速かつ柔軟に、場合によっては自動的にプロビジョニングされ、素早くスケールアウトし、迅速に解放されて素早くスケールインすることができる。プロビジョニングに使用できる能力は、利用者には、多くの場合、任意の量をいつでも無制限に購入できるように見える。

【0095】

測定されるサービス：クラウド・システムは、計測機能を活用することによって、サービスの種類（例えば、ストレージ、処理、帯域幅、およびアクティブなユーザのアカウント）に適した、ある抽象レベルで、リソースの使用を自動的に制御および最適化する。リソースの使用量は監視、制御、および報告することができ、利用されるサービスのプロバイダと利用者の両方に透明性が提供される。

20

【0096】

サービス・モデルは、次のとおりである。

【0097】

SaaS（Software as a Service）：利用者に提供される能力は、クラウド・インフラストラクチャ上で稼働しているプロバイダのアプリケーションの利用である。それらのアプリケーションは、Webブラウザ（例えば、Webベースの電子メール）などのシン・クライアント・インターフェイスを介して、さまざまなクライアント・デバイスからアクセスできる。利用者は、ネットワーク、サーバ、オペレーティング・システム、ストレージ、または個々のアプリケーション機能でさえも含む基盤になるクラウド・インフラストラクチャを、限定的なユーザ固有のアプリケーション構成設定を行う可能性を除き、管理することも制御することもない。

30

【0098】

PaaS（Platform as a Service）：利用者に提供される能力は、プロバイダによって支援されるプログラミング言語およびツールを使用して作成された、利用者が作成または取得したアプリケーションをクラウド・インフラストラクチャにデプロイすることである。利用者は、ネットワーク、サーバ、オペレーティング・システム、またはストレージを含む基盤になるクラウド・インフラストラクチャを管理することも制御することもないが、デプロイされたアプリケーション、および場合によってはアプリケーション・ホスティング環境の構成を制御することができる。

40

【0099】

IaaS（Infrastructure as a Service）：利用者に提供される能力は、処理、ストレージ、ネットワーク、およびその他の基本的な計算リソースのプロビジョニングであり、利用者は、オペレーティング・システムおよびアプリケーションを含むことができる任意のソフトウェアをデプロイして実行できる。利用者は、基盤になるクラウド・インフラストラクチャを管理することも制御することもないが、オペレーティング・システム、ストレージ、デプロイされたアプリケーションを制御することができ、場合によっては、選択されたネットワーク・コンポーネント（例えば、ホスト・ファイアウォール）を限定的に制御できる。

【0100】

50

デプロイメント・モデルは、次のとおりである。

【0101】

プライベート・クラウド：このクラウド・インフラストラクチャは、組織のためにのみ運用される。この組織またはサード・パーティによって管理することができ、オンプレミスまたはオフプレミスに存在することができる。

【0102】

コミュニティ・クラウド：このクラウド・インフラストラクチャは、複数の組織によって共有され、関心事（例えば、任務、セキュリティ要件、ポリシー、およびコンプライアンスに関する考慮事項）を共有している特定のコミュニティを支援する。これらの組織またはサード・パーティによって管理することができ、オンプレミスまたはオフプレミスに存在することができる。

10

【0103】

パブリック・クラウド：このクラウド・インフラストラクチャは、一般ユーザまたは大規模な業界団体が使用できるようになっており、クラウド・サービスを販売する組織によって所有される。

【0104】

ハイブリッド・クラウド：このクラウド・インフラストラクチャは、データとアプリケーションの移植を可能にする標準化された技術または独自の技術（例えば、クラウド間の負荷バランスを調整するためのクラウド・バースト）によって固有の実体を残したまま互いに結合された2つ以上のクラウド（プライベート、コミュニティ、またはパブリック）の複合である。

20

【0105】

クラウド・コンピューティング環境は、ステートレス、疎結合、モジュール性、および意味的相互運用性に重点を置いたサービス指向の環境である。クラウド・コンピューティングの中心になるのは、相互接続されたノードのネットワークを含んでいるインフラストラクチャである。

【0106】

ここで図17を参照すると、例示的なクラウド・コンピューティング環境50が示されている。図示されているように、クラウド・コンピューティング環境50は、クラウドの利用者によって使用されるローカル・コンピューティング・デバイス（例えば、パーソナル・デジタル・アシスタント（PDA：personal digital assistant）または携帯電話54A、デスクトップ・コンピュータ54B、ラップトップ・コンピュータ54C、または自動車コンピュータ・システム54N、あるいはその組み合わせなど）が通信できる1つまたは複数のクラウド・コンピューティング・ノード10を含んでいる。ノード10は、互いに通信してよい。ノード10は、1つまたは複数のネットワーク内で、本明細書において前述されたプライベート・クラウド、コミュニティ・クラウド、パブリック・クラウド、またはハイブリッド・クラウド、あるいはこれらの組み合わせなどに、物理的または仮想的にグループ化されてよい（図示されていない）。これによって、クラウド・コンピューティング環境50は、クラウドの利用者がローカル・コンピューティング・デバイス上でリソースを維持する必要のないインフラストラクチャ、プラットフォーム、またはSaaS、あるいはその組み合わせを提供できる。図17に示されたコンピューティング・デバイス54A～Nの種類は、例示のみが意図されており、コンピューティング・ノード10およびクラウド・コンピューティング環境50は、任意の種類のネットワークまたはネットワーク・アドレス可能な接続（例えば、Webブラウザを使用した接続）あるいはその両方を經由して任意の種類のコンピュータ制御デバイスと通信することができると理解される。

30

40

【0107】

ここで図18を参照すると、クラウド・コンピューティング環境50（図17）によって提供される機能的抽象レイヤのセットが示されている。図18に示されたコンポーネント、レイヤ、および機能は、例示のみが意図されており、本発明の実施形態がこれらに限

50

定されないということが、あらかじめ理解されるべきである。図示されているように、次のレイヤおよび対応する機能が提供される。

【0108】

ハードウェアおよびソフトウェア・レイヤ60は、ハードウェア・コンポーネントおよびソフトウェア・コンポーネントを含む。ハードウェア・コンポーネントの例としては、メインフレーム61、RISC (Reduced Instruction Set Computer) アーキテクチャベースのサーバ62、サーバ63、ブレード・サーバ64、ストレージ・デバイス65、ならびにネットワークおよびネットワーク・コンポーネント66が挙げられる。一部の実施形態では、ソフトウェア・コンポーネントは、ネットワーク・アプリケーション・サーバ・ソフトウェア67およびデータベース・ソフトウェア68を含む。

10

【0109】

仮想化レイヤ70は、仮想サーバ71、仮想ストレージ72、仮想プライベート・ネットワークを含む仮想ネットワーク73、仮想アプリケーションおよびオペレーティング・システム74、ならびに仮想クライアント75などの仮想的実体の例を提供できる抽象レイヤを備える。

【0110】

一例を挙げると、管理レイヤ80は、以下で説明される機能を提供することができる。リソース・プロビジョニング81は、クラウド・コンピューティング環境内でタスクを実行するために利用される計算リソースおよびその他のリソースの動的調達を行う。計測および価格設定82は、クラウド・コンピューティング環境内でリソースが利用される際のコスト追跡、およびそれらのリソースの利用に対する請求書の作成と送付を行う。一例を挙げると、それらのリソースは、アプリケーション・ソフトウェア・ライセンスを含んでよい。セキュリティは、クラウドの利用者およびタスクのID検証を行うとともに、データおよびその他のリソースの保護を行う。ユーザ・ポータル83は、クラウド・コンピューティング環境へのアクセスを利用者およびシステム管理者に提供する。サービス・レベル管理84は、必要なサービス・レベルを満たすように、クラウドの計算リソースの割り当てと管理を行う。サービス水準合意 (SLA: Service Level Agreement) 計画および実行85は、今後の要求が予想されるクラウドの計算リソースの事前準備および調達を、SLAに従って行う。

20

【0111】

ワークロード・レイヤ90は、クラウド・コンピューティング環境で利用できる機能の例を示している。このレイヤから提供されてよいワークロードおよび機能の例としては、マッピングおよびナビゲーション91、ソフトウェア開発およびライフサイクル管理92、仮想クラスルーム教育の配信93、データ解析処理94、トランザクション処理95、および仮想マシンのセキュア・ストレージへのアクセスの制御96が挙げられる。これらが単なる例であり、他の実施形態では、各レイヤが異なるサービスを含むことができるということが理解される。

30

【0112】

ここで図19を参照すると、本発明の1つまたは複数の実施形態に従って、システム1900が示されている。システム1900は、ネットワーク165などを介して1つまたは複数のクライアント・デバイス20A~20Eと直接的または間接的に通信する例示的なノード10 (例えば、ホスティング・ノード) を含んでいる。ノード10は、クラウド・コンピューティング・プロバイダのデータセンターまたはホスト・サーバであることができる。ノード10は、1つまたは複数のVM15 (15A~15N) のデプロイを容易にするハイパーバイザ12を実行する。ノード10は、セキュア・インターフェイス制御11を含むハードウェア/ファームウェア・レイヤ13をさらに含んでいる。セキュア・インターフェイス制御11は、ハイパーバイザ12が1つまたは複数のサービスを仮想マシン15に提供するのを容易にする1つまたは複数のハードウェア・モジュールおよびファームウェアを含んでいる。ハイパーバイザ12とセキュア・インターフェイス制御11の間、セキュア・インターフェイス制御11と1つまたは複数のVM15の間、ハイパー

40

50

バイザ 12 と 1 つまたは複数の VM 15 の間、およびセキュア・インターフェイス制御 11 を介したハイパーバイザ 12 と VM 15 の間には、通信が存在することができる。セキュアな VM 環境を促進するために、本発明の 1 つまたは複数の実施形態に従うホスティング・ノード 10 は、ハイパーバイザ 12 と 1 つまたは複数の VM 15 の間に、直接通信を何も含まない。

【0113】

例えば、ホスティング・ノード 10 は、クライアント・デバイス 20 A が VM 15 A ~ 15 N のうちの 1 つまたは複数のデプロイするのを容易にすることができる。個別のクライアント・デバイス 20 A ~ 20 E からの各要求に応答して、VM 15 A ~ 15 N がデプロイされてよい。例えば、クライアント・デバイス 20 A によって VM 15 A がデプロイされてよく、クライアント・デバイス 20 B によって VM 15 B がデプロイされてよく、クライアント・デバイス 20 C によって VM 15 C がデプロイされてよい。ノード 10 は、クライアントが (VM として実行するのではなく) 物理的サーバをプロビジョニングするのを容易にすることもできる。本明細書に記載された例は、VM の一部としてノード 10 内のリソースのプロビジョニングを具現化するが、説明された技術的解決策は、物理的サーバの一部としてリソースをプロビジョニングするように適用されてもよい。

【0114】

1 つの例では、クライアント・デバイス 20 A ~ 20 E は、人、企業、政府機関、会社内の部門、または任意のその他の実体などの、同じ実体に属してよく、ノード 10 は、実体のプライベート・クラウドとして運用されてよい。この場合、ノード 10 は、実体に属しているクライアント・デバイス 20 A ~ 20 E によってデプロイされている VM 15 A ~ 15 N のみをホストする。別の例では、クライアント・デバイス 20 A ~ 20 E は、個別の実体に属してよい。例えば、第 1 の実体はクライアント・デバイス 20 A を所有してよく、第 2 の実体はクライアント・デバイス 20 B を所有してよい。この場合、ノード 10 は、異なる実体の VM をホストするパブリック・クラウドとして運用されてよい。例えば、VM 15 A ~ 15 N は、VM 15 A が VM 15 B へのアクセスを容易にしないような、覆い隠される方法でデプロイされてよい。例えば、ノード 10 は、IBM z System (R) プロセッサ・リソース／システム・マネージャ (PR/SM: Processor Resource/System Manager) 論理パーティション (LPAR: Logical Partition) 機能を使用して VM 15 A ~ 15 N を覆い隠してよい。PR/SM LPAR などのこれらの機能は、パーティション間を分離することによって、ノード 10 が、同じ物理ノード 10 上の異なる実体のために、異なる論理パーティション内で、2 つ以上の VM 15 A ~ 15 N をデプロイするのを容易にする。PR/SM LPAR ハイパーバイザは、この分離を提供するために特定のハードウェアと共に、信頼できる内部ファームウェアにおいて実装される。

【0115】

クライアント・デバイス 20 A ~ 20 E からのクライアント・デバイス 20 A は、コンピュータ、スマートフォン、タブレット・コンピュータ、デスクトップ・コンピュータ、ラップトップ・コンピュータ、サーバ・コンピュータ、またはノード 10 のハイパーバイザ 12 による VM のデプロイメントを要求する任意のその他の通信装置などの、通信装置である。クライアント・デバイス 20 A は、ネットワーク 165 を介してハイパーバイザ 12 によって受信するための要求を送信してよい。VM 15 A ~ 15 N からの VM 15 A は、クライアント・デバイス 20 A ~ 20 E からのクライアント・デバイス 20 A からの要求に応答してハイパーバイザ 12 がデプロイする VM イメージである。ハイパーバイザ 12 は、VM モニタ (VMM) であり、VM を作成して実行するソフトウェア、ファームウェア、またはハードウェアであってよい。ハイパーバイザ 12 は、VM 15 A がノード 10 のハードウェア・コンポーネントを使用してプログラムを実行すること、またはデータを格納すること、あるいはその両方を容易にする。ハイパーバイザ 12 は、適切な機能および変更を伴って、IBM z System (R)、Oracle の VM Server、Citrix の XenServer、Vmware の ESX、Microsoft Hy

per-Vハイパーバイザ、または任意のその他のハイパーバイザであってよい。ハイパーバイザ12は、ノード10上で直接実行されるネイティブ・ハイパーバイザであるか、または別のハイパーバイザ上で実行されるホストされたハイパーバイザであってよい。

【0116】

ここで図20を参照すると、本発明の1つまたは複数の実施形態に従って、本明細書の教示を実装するためのノード10が示されている。ノード10は、本明細書において説明されているように、さまざまな通信技術を利用するコンピューティング・デバイスおよびネットワークの任意の数および組み合わせを備えるか、または採用するか、あるいはその両方である、電子的コンピュータ・フレームワークであることができる。ノード10は、容易にスケーリング可能であり、拡張可能であり、モジュール式であり、異なるサービスに変化する能力、または他の機能とは無関係に、一部の機能を再構成する能力を有することができる。

【0117】

この実施形態では、ノード10が、1つまたは複数の中央処理装置（CPU：central processing units）2201a、2001b、2001cなどを含むことができるプロセッサ2001を含んでいる。プロセッサ2001は、処理回路、マイクロプロセッサ、コンピューティング・ユニットとも呼ばれ、システム・バス2002を介してシステム・メモリ2003およびさまざまな他のコンポーネントに結合される。システム・メモリ2003は、読み取り専用メモリ（ROM：read only memory）2004およびランダム・アクセス・メモリ（RAM：random access memory）2005を含む。ROM2004は、システム・バス2002に結合され、ノード10の特定の基本機能を制御する基本入出力システム（BIOS：basic input/output system）を含んでよい。RAMは、プロセッサ2001で使用するためにシステム・バス2002に結合された読み取り書き込みメモリである。

【0118】

図20のノード10は、プロセッサ2001による読み取りおよび実行が可能な有形のストレージ媒体の例であるハード・ディスク2007を含んでいる。ハード・ディスク2007は、ソフトウェア2008およびデータ2009を格納する。ソフトウェア2008は、（図1～19を参照して説明されたプロセスなどのプロセスを実行するために）プロセッサ2001によってノード10上で実行される命令として格納される。データ2009は、ソフトウェア2008の動作を支援し、ソフトウェア2008の動作によって使用されるさまざまなデータ構造に構造化された定性的変数または定量的変数の値のセットを含む。

【0119】

図20のノード10は、ノード10のプロセッサ2001、システム・メモリ2003、ハード・ディスク2007、およびその他のコンポーネント（例えば、周辺機器および外部デバイス）の間を相互接続し、これらの間の通信を支援する1つまたは複数のアダプタ（例えば、ハード・ディスク・コントローラ、ネットワーク・アダプタ、グラフィックス・アダプタなど）を含む。本発明の1つまたは複数の実施形態では、1つまたは複数のアダプタを、中間バス・ブリッジを介してシステム・バス2002に接続された1つまたは複数のI/Oバスに接続することができ、1つまたは複数のI/Oバスが、PCI（Peripheral Component Interconnect）などの一般的なプロトコルを利用することができる。

【0120】

図に示されているように、ノード10は、キーボード2021、マウス2022、スピーカ2023、およびマイクロホン2024をシステム・バス2002に相互接続するインターフェイス・アダプタ2020を含んでいる。ノード10は、システム・バス2002をディスプレイ2031に相互接続するディスプレイ・アダプタ2030を含んでいる。ディスプレイ・アダプタ2030（またはプロセッサ2001あるいはその両方）は、GUI2032の表示および管理などのグラフィックス性能を提供するために、グラフィ

ックス・コントローラを含むことができる。通信アダプタ 2041 は、システム・バス 2002 をネットワーク 2050 と相互接続し、ノード 10 が、サーバ 2051 およびデータベース 2052 などの他のシステム、デバイス、データ、およびソフトウェアと通信できるようにする。本発明の 1 つまたは複数の実施形態では、ソフトウェア 2008 およびデータ 2009 の動作が、サーバ 2051 およびデータベース 2052 によってネットワーク 2050 上に実装され得る。例えば、ネットワーク 2050、サーバ 2051、およびデータベース 2052 は、組み合わせさせて、PaaS (Platform as a Service)、SaaS (Software as a Service)、または IaaS (Infrastructure as a Service)、あるいはその組み合わせとして (例えば、分散システム内の Web アプリケーションとして)、ソフトウェア 2008 およびデータ 2009 の内部の反復を提供することができる。

10

【0121】

本明細書に記載された実施形態は、必然的にコンピュータ技術に根差しており、特に、VM をホストするコンピュータ・サーバに根差している。さらに、本発明の 1 つまたは複数の実施形態は、コンピューティング技術自体の動作に対する改良を促進し、特に、ハイパーバイザがセキュア VM に関連付けられたメモリ、レジスタ、およびその他のそのようなデータにアクセスすることを禁止されていても、VM をホストするコンピュータ・サーバがセキュア VM をホストするのを容易にすることによって、VM をホストするコンピュータ・サーバの動作に対する改良を促進する。加えて、本発明の 1 つまたは複数の実施形態は、ハードウェア、ファームウェア (例えば、ミリコード)、またはこれらの組み合わせを含むセキュア・インターフェイス制御 (本明細書では「UV」とも呼ばれる) を使用して、セキュア VM とハイパーバイザの分離を促進し、このようにして、コンピューティング・サーバによってホストされる VM のセキュリティを維持することによって、コンピューティング・サーバをホストする VM の改善に向かう重要な手順を提供する。セキュア・インターフェイス制御は、本明細書において説明されているように、VM の初期化／終了時に、VM の状態を保護することに大きなオーバーヘッドを追加せずに、セキュリティを促進するための軽量の間接動作を提供する。

20

【0122】

本明細書で開示された本発明の実施形態は、VM のセキュア・ストレージへのアクセスを制御するシステム、方法、またはコンピュータ・プログラム製品 (本明細書ではシステム)、あるいはその組み合わせを含んでよい。説明ごとに、要素の識別子が、異なる図の他の類似する要素に再使用されるということに注意する。

30

【0123】

本明細書では、関連する図面を参照して、本発明のさまざまな実施形態が説明される。本発明の範囲を逸脱することなく、本発明の代替の実施形態が考案され得る。以下の説明および図面において、要素間のさまざまな接続および位置関係 (例えば、上、下、隣接など) が示される。それらの接続または位置関係あるいはその両方は、特に規定されない限り、直接的または間接的であることができ、本発明はこの点において限定するよう意図されていない。したがって、実体の結合は、直接的結合または間接的結合を指すことができ、実体間の位置関係は、直接的位置関係または間接的位置関係であることができる。さらに、本明細書に記載されたさまざまな作業および工程段階は、本明細書に詳細に記載されない追加の段階または機能を含んでいるさらに包括的な手順または工程に組み込まれ得る。

40

【0124】

以下の定義および略称が、特許請求の範囲および本明細書の解釈に使用される。本明細書において使用されているように、「備える」、「備えている」、「含む」、「含んでいる」、「有する」、「有している」、「含有する」、もしくは「含有している」という用語、またはこれらの任意のその他の変形は、非排他的包含をカバーするよう意図されている。例えば、要素のリストを含んでいる組成、混合、工程、方法、製品、または装置は、それらの要素のみに必ずしも限定されず、明示的に列記されていないか、またはそのような組成、混合、工程、方法、製品、もしくは装置に固有の、その他の要素を含むことがで

50

きる。

【0125】

さらに、「例示的」という用語は、本明細書では「例、事例、または実例としての役割を果たす」ことを意味するために使用される。「例示的」として本明細書に記載された任意の実施形態または設計は、必ずしも他の実施形態もしくは設計よりも好ましいか、または有利であると解釈されるべきではない。「少なくとも1つ」および「1つまたは複数」という用語は、1以上の任意の整数（すなわち、1、2、3、4など）を含んでいると理解されてよい。「複数」という用語は、2以上の任意の整数（すなわち、2、3、4、5など）を含んでいると理解されてよい。「接続」という用語は、間接的「接続」および直接的「接続」の両方を含んでよい。

10

【0126】

「約」、「実質的に」、「近似的に」、およびこれらの変形の実用用語は、本願書の出願時に使用できる機器に基づいて、特定の量の測定に関連付けられた誤差の程度を含むよう意図されている。例えば、「約」は、特定の値の±8%または5%、あるいは2%の範囲を含むことができる。

【0127】

本発明は、任意の可能な統合の技術的詳細レベルで、システム、方法、またはコンピュータ・プログラム製品、あるいはその組み合わせであってよい。コンピュータ・プログラム製品は、プロセッサに本発明の態様を実行させるためのコンピュータ可読プログラム命令を含んでいる1つ（または複数）のコンピュータ可読ストレージ媒体を含んでよい。

20

【0128】

コンピュータ可読ストレージ媒体は、命令実行デバイスによって使用するための命令を保持および格納できる有形のデバイスであることができる。コンピュータ可読ストレージ媒体は、例えば、電子ストレージ・デバイス、磁気ストレージ・デバイス、光ストレージ・デバイス、電磁ストレージ・デバイス、半導体ストレージ・デバイス、またはこれらの任意の適切な組み合わせであってよいが、これらに限定されない。コンピュータ可読ストレージ媒体のさらに具体的な例の非網羅的リストは、ポータブル・フロッピー（R）・ディスク、ハード・ディスク、ランダム・アクセス・メモリ（RAM：random access memory）、読み取り専用メモリ（ROM：read-only memory）、消去可能プログラマブル読み取り専用メモリ（EPROM：erasable programmable read-only memoryまたはフラッシュ・メモリ）、スタティック・ランダム・アクセス・メモリ（SRAM：static random access memory）、ポータブル・コンパクト・ディスク読み取り専用メモリ（CD-ROM：compact disc read-only memory）、デジタル多用途ディスク（DVD：digital versatile disk）、メモリ・スティック、フロッピー（R）・ディスク、パンチカードまたは命令が記録されている溝の中の隆起構造などの機械的にエンコードされるデバイス、およびこれらの任意の適切な組み合わせを含む。本明細書において使用されるとき、コンピュータ可読ストレージ媒体は、それ自体が、電波またはその他の自由に伝搬する電磁波、導波管またはその他の送信媒体を伝搬する電磁波（例えば、光ファイバ・ケーブルを通過する光パルス）、あるいはワイヤを介して送信される電気信号などの一過性の信号であると解釈されるべきではない。

30

40

【0129】

本明細書に記載されたコンピュータ可読プログラム命令は、コンピュータ可読ストレージ媒体から各コンピューティング・デバイス／処理デバイスへ、またはネットワーク（例えば、インターネット、ローカル・エリア・ネットワーク、広域ネットワーク、または無線ネットワーク、あるいはその組み合わせ）を介して外部コンピュータもしくは外部ストレージ・デバイスへダウンロードされ得る。このネットワークは、銅伝送ケーブル、光伝送ファイバ、無線送信、ルータ、ファイアウォール、スイッチ、ゲートウェイ・コンピュータ、またはエッジ・サーバ、あるいはその組み合わせを備えてよい。各コンピューティング・デバイス／処理デバイス内のネットワーク・アダプタ・カードまたはネットワーク・インターフェイスは、コンピュータ可読プログラム命令をネットワークから受信し、そ

50

これらのコンピュータ可読プログラム命令を各コンピューティング・デバイス／処理デバイス内のコンピュータ可読ストレージ媒体に格納するために転送する。

【0130】

本発明の動作を実行するためのコンピュータ可読プログラム命令は、アセンブラ命令、命令セット・アーキテクチャ（ISA：instruction-set-architecture）命令、マシン命令、マシン依存命令、マイクロコード、ファームウェア命令、状態設定データ、集積回路のための構成データ、あるいは、Smalltalk（R）、C++などのオブジェクト指向プログラミング言語、および「C」プログラミング言語または同様のプログラミング言語などの手続き型プログラミング言語を含む、1つもしくは複数のプログラミング言語の任意の組み合わせで記述されたソース・コードまたはオブジェクト・コードのいずれか
10
であってよい。コンピュータ可読プログラム命令は、ユーザのコンピュータ上で全体的に実行すること、ユーザのコンピュータ上でスタンドアロン・ソフトウェア・パッケージとして部分的に実行すること、ユーザのコンピュータ上およびリモート・コンピュータ上でそれぞれ部分的に実行すること、あるいはリモート・コンピュータ上またはサーバ上で全体的に実行することができる。後者のシナリオでは、リモート・コンピュータは、ローカル・エリア・ネットワーク（LAN：local area network）または広域ネットワーク（WAN：wide area network）を含む任意の種類のネットワークを介してユーザのコンピュータに接続されてよく、または接続は、（例えば、インターネット・サービス・プロバイダを使用してインターネットを介して）外部コンピュータに対して行われてよい。一部
20
の実施形態では、本発明の態様を実行するために、例えばプログラマブル論理回路、フィールドプログラマブル・ゲート・アレイ（FPGA：field-programmable gate arrays）、またはプログラマブル・ロジック・アレイ（PLA：programmable logic arrays）を含む電子回路は、コンピュータ可読プログラム命令の状態情報を利用することによって、電子回路をカスタマイズするためのコンピュータ可読プログラム命令を実行してよい。

【0131】

本発明の態様は、本明細書において、本発明の実施形態に従って、方法、装置（システム）、およびコンピュータ・プログラム製品のフローチャート図またはブロック図あるいはその両方を参照して説明される。フローチャート図またはブロック図あるいはその両方の各ブロック、ならびにフローチャート図またはブロック図あるいはその両方に含まれるブロックの組み合わせが、コンピュータ可読プログラム命令によって実装され得るという
30
ことが理解されるであろう。

【0132】

これらのコンピュータ可読プログラム命令は、コンピュータまたはその他のプログラム可能なデータ処理装置のプロセッサを介して実行される命令が、フローチャートまたはブロック図あるいはその両方の1つまたは複数のブロックに指定される機能／動作を実施する手段を作り出すべく、汎用コンピュータ、専用コンピュータ、または他のプログラム可能なデータ処理装置のプロセッサに提供されてマシンを生成するものであってよい。これらのコンピュータ可読プログラム命令は、命令が格納されたコンピュータ可読ストレージ媒体がフローチャートまたはブロック図あるいはその両方の1つまたは複数のブロックに指定される機能／動作の態様を実施する命令を含んでいる製品を備えるように、コンピュータ可読ストレージ媒体に格納され、コンピュータ、プログラム可能なデータ処理装置、または他のデバイス、あるいはその組み合わせに特定の方式で機能するように指示できる
40
ものであってもよい。

【0133】

コンピュータ可読プログラム命令は、コンピュータ上、その他のプログラム可能な装置上、またはその他のデバイス上で実行される命令が、フローチャートまたはブロック図あるいはその両方の1つまたは複数のブロックに指定される機能／動作を実施するように、コンピュータ、その他のプログラム可能なデータ処理装置、またはその他のデバイスに読み込まれてもよく、それによって、一連の動作可能なステップを、コンピュータ上、その他のプログラム可能な装置上、またはコンピュータ実装プロセスを生成するその他のデバ
50

イス上で実行させる。

【0134】

図内のフローチャートおよびブロック図は、本発明のさまざまな実施形態に従って、システム、方法、およびコンピュータ・プログラム製品の可能な実装のアーキテクチャ、機能、および動作を示す。これに関連して、フローチャートまたはブロック図内の各ブロックは、規定された論理機能を実装するための1つまたは複数の実行可能な命令を備える、命令のモジュール、セグメント、または部分を表してよい。一部の代替の実装では、ブロックに示された機能は、図に示された順序とは異なる順序で発生してよい。例えば、連続して示された2つのブロックは、実際には、含まれている機能に応じて、実質的に同時に実行されるか、または場合によっては逆の順序で実行されてよい。ブロック図またはフローチャート図あるいはその両方の各ブロック、ならびにブロック図またはフローチャート図あるいはその両方に含まれるブロックの組み合わせは、規定された機能または動作を実行するか、あるいは専用ハードウェアとコンピュータ命令の組み合わせを実行する専用ハードウェアベースのシステムによって実装され得るということにも注意する。

10

【0135】

本明細書で使用される用語は、特定の実施形態を説明することのみを目的としており、制限することを意図していない。本明細書において使用されるとき、単数形「a」、「an」、および「the」は、特に明示的に示されない限り、複数形も含むことが意図されている。「備える」または「備えている」あるいはその両方の用語は、本明細書で 사용되는場合、記載された機能、整数、ステップ、動作、要素、またはコンポーネント、あるいはその組み合わせの存在を示すが、1つまたは複数のその他の機能、整数、ステップ、動作、要素コンポーネント、またはこれらのグループ、あるいはその組み合わせの存在または追加を除外していないということが、さらに理解されるであろう。

20

【0136】

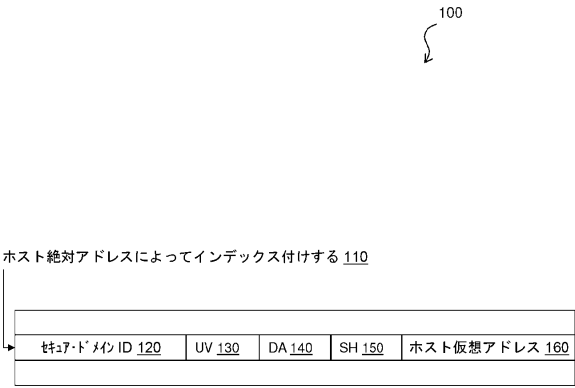
本明細書におけるさまざまな実施形態の説明は、例示の目的で提示されているが、網羅的であることは意図されておらず、開示された実施形態に制限されない。記載された実施形態の範囲および思想を逸脱することなく多くの変更および変形が可能であることは、当業者にとって明らかであろう。本明細書で使用された用語は、実施形態の原理、実際の適用、または市場で見られる技術を超える技術的改良を最も適切に説明するため、あるいは他の当業者が本明細書で開示された実施形態を理解できるようにするために選択されている。

30

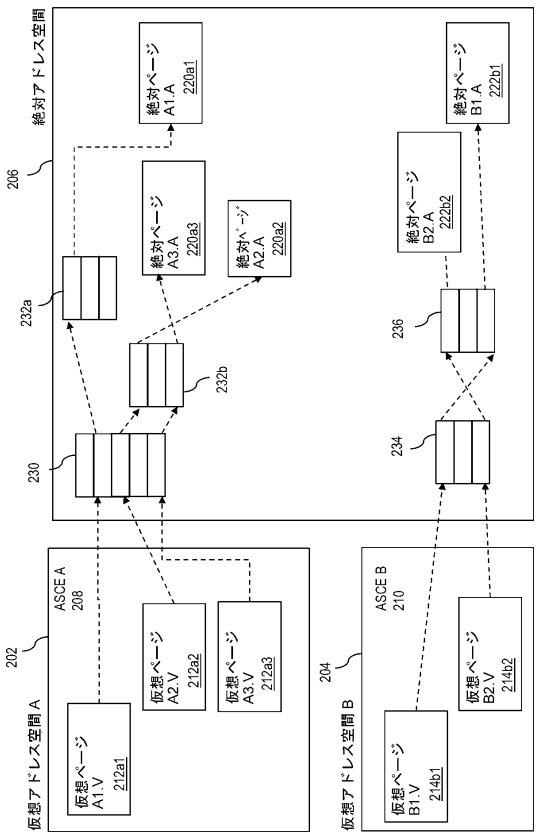
40

50

【図面】
【図 1】



【図 2】



10

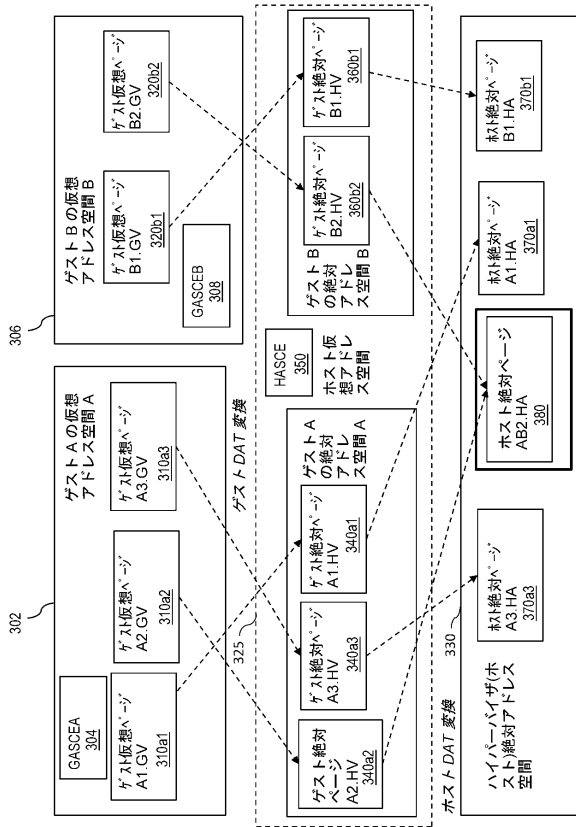
20

30

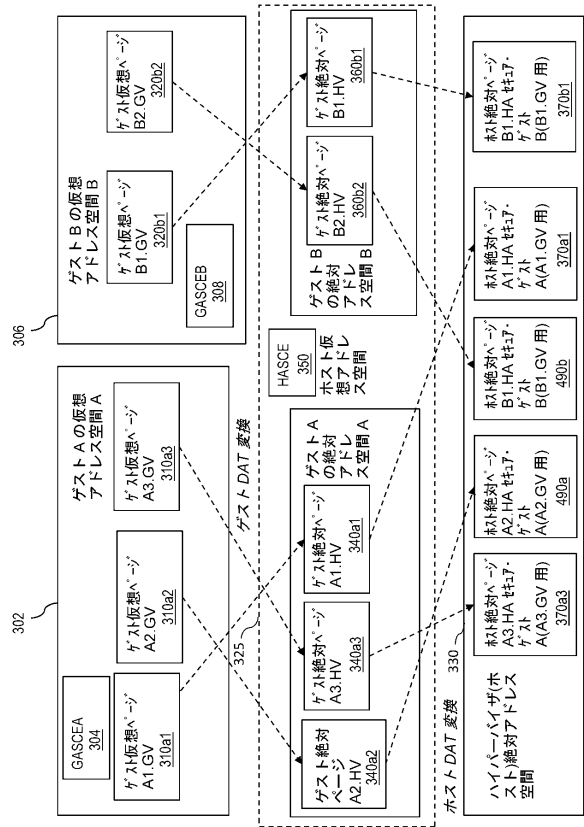
40

50

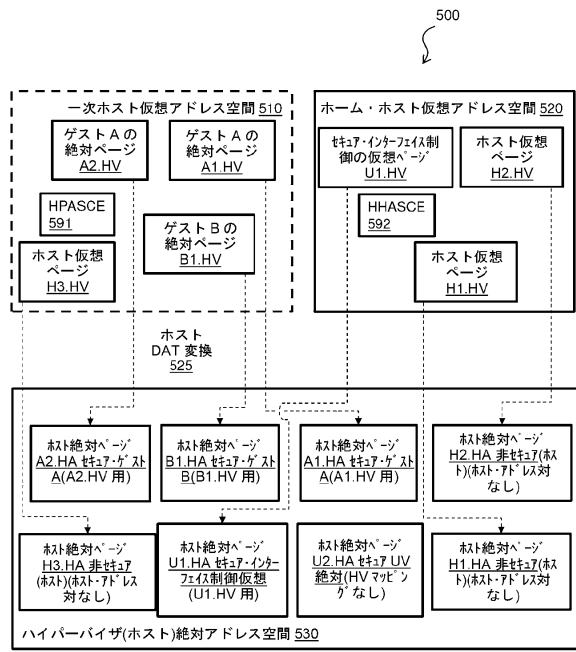
【図 3】



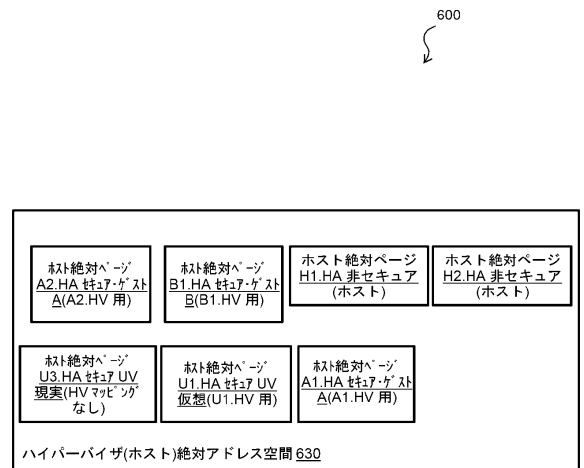
【図 4】



【図 5】



【図 6】



10

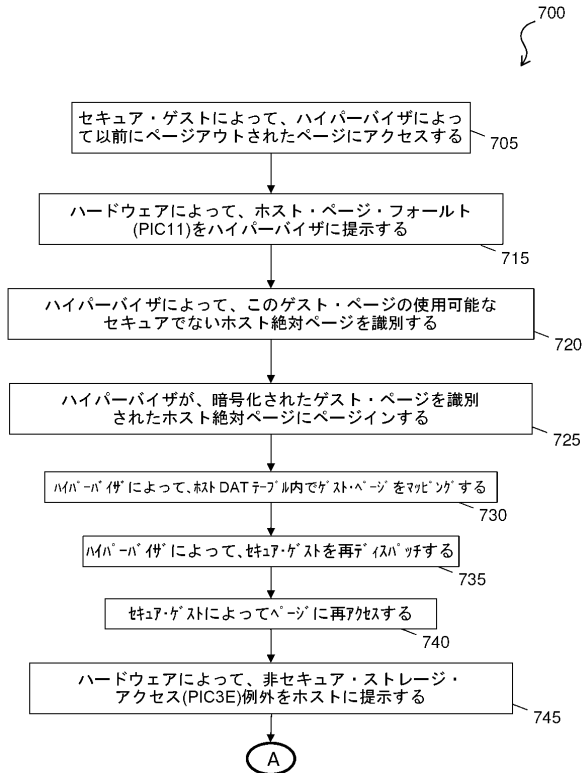
20

30

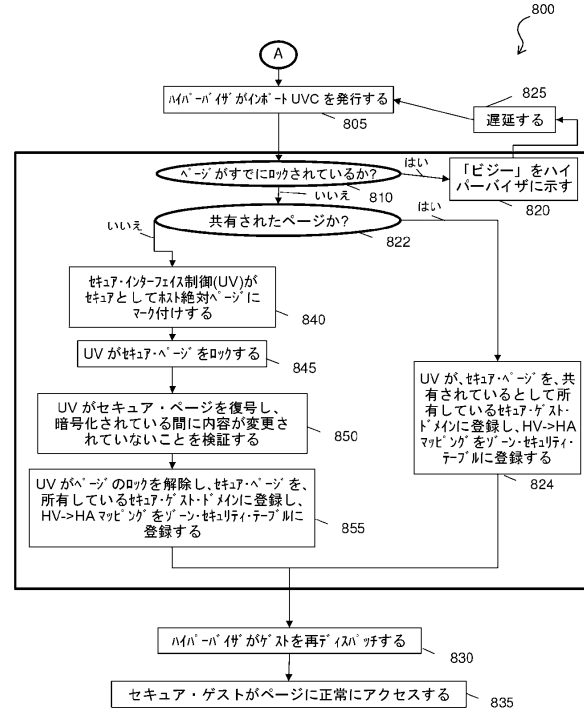
40

50

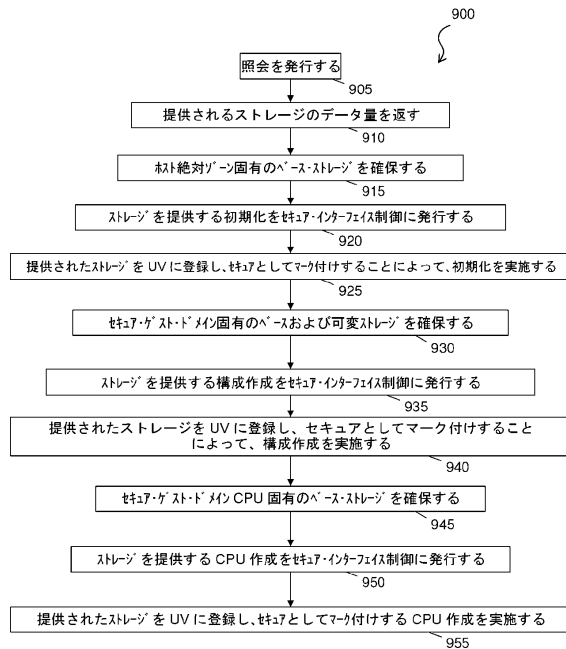
【図 7】



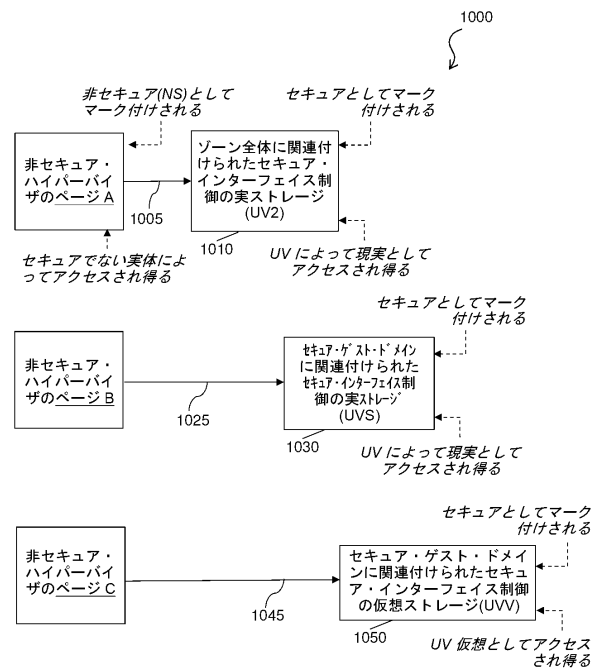
【図 8】



【図 9】



【図 10】



10

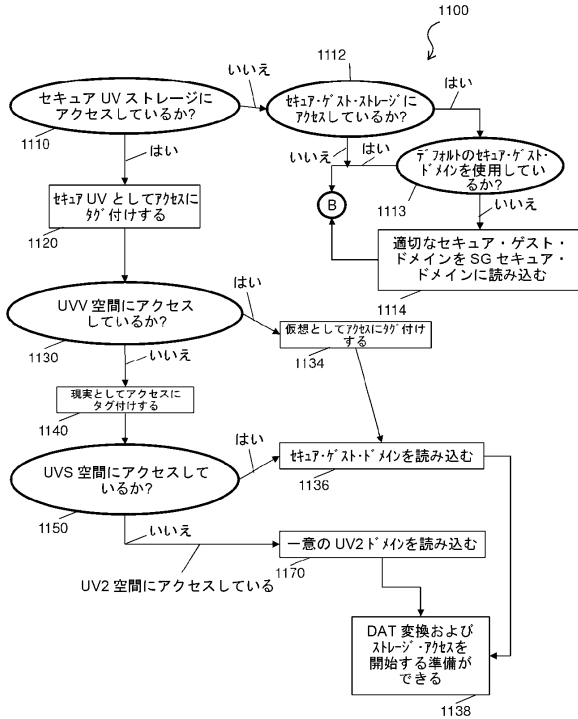
20

30

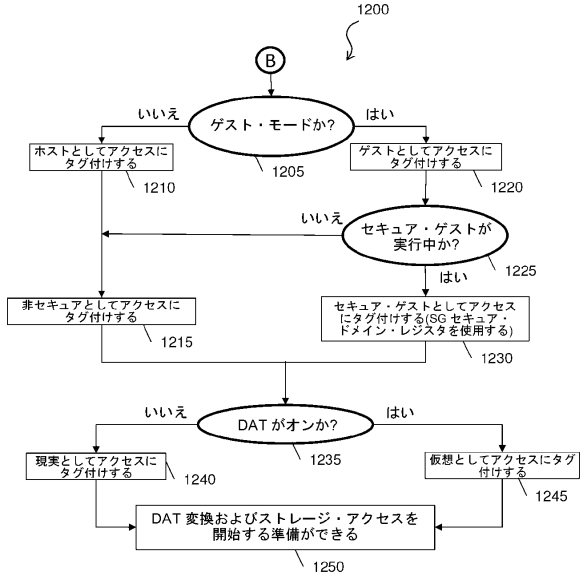
40

50

【図 1 1】



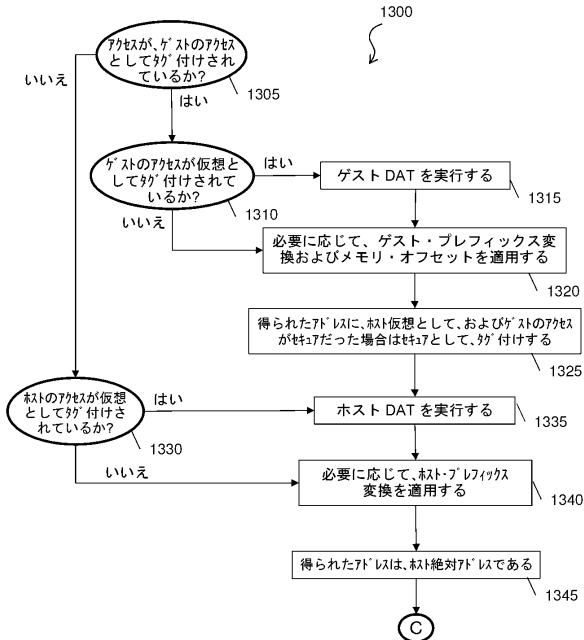
【図 12】



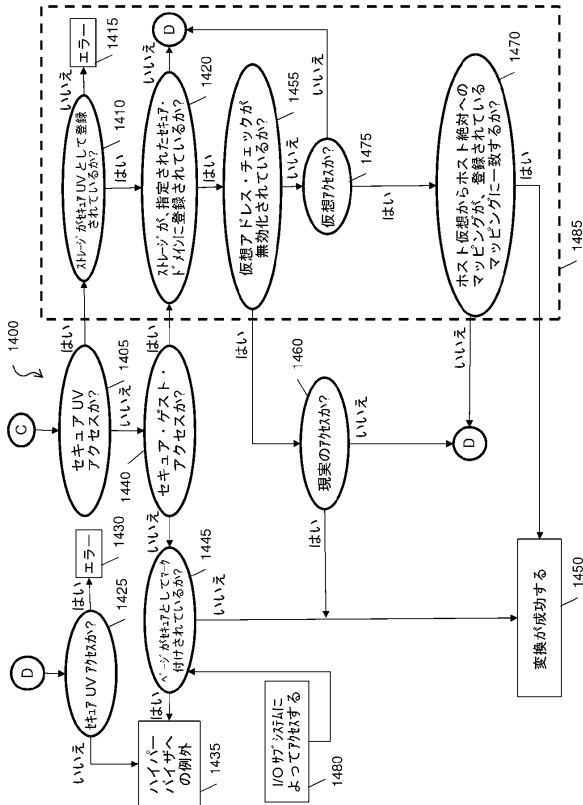
10

20

【図 13】



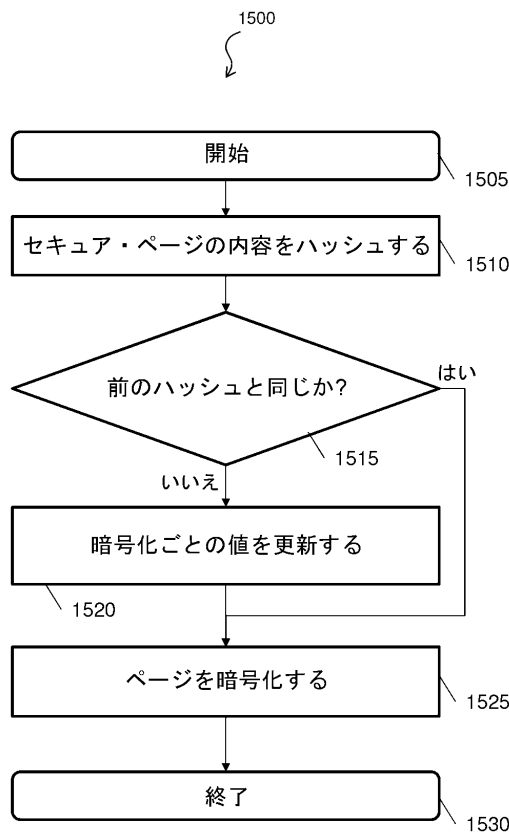
【図 14】



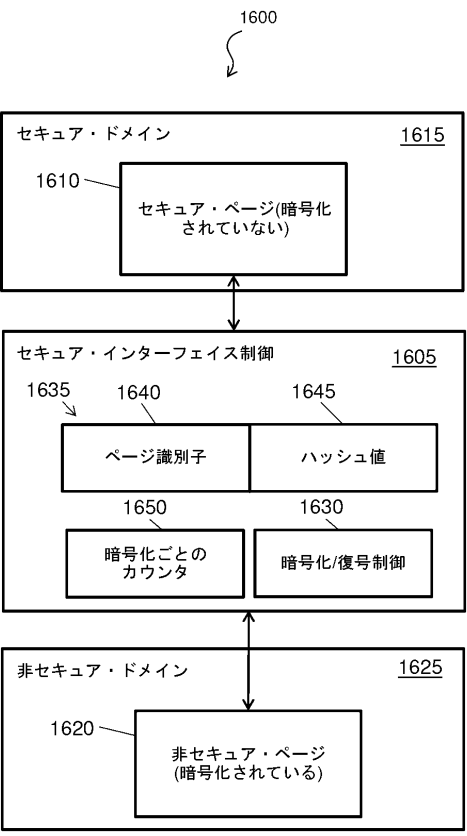
30

40

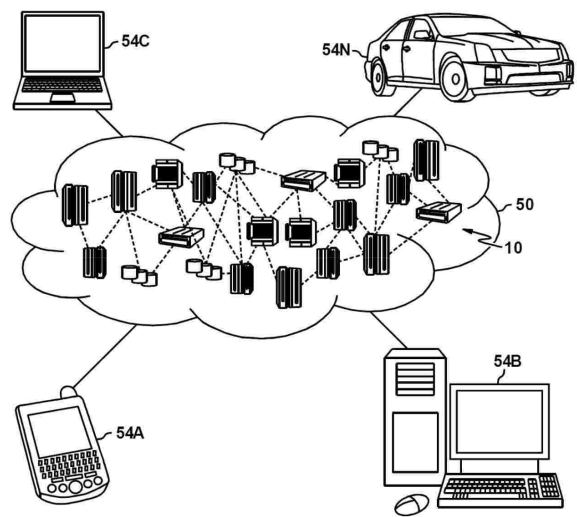
【図 15】



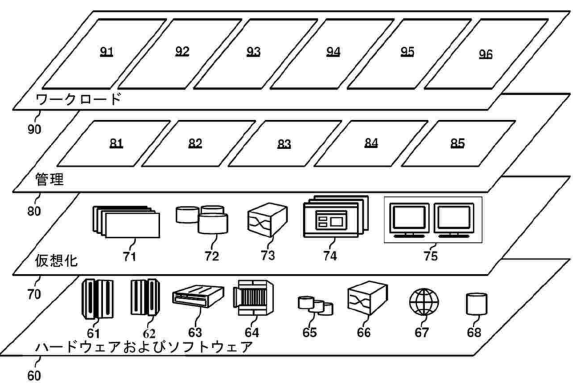
【図 16】



【図 17】



【図 18】



10

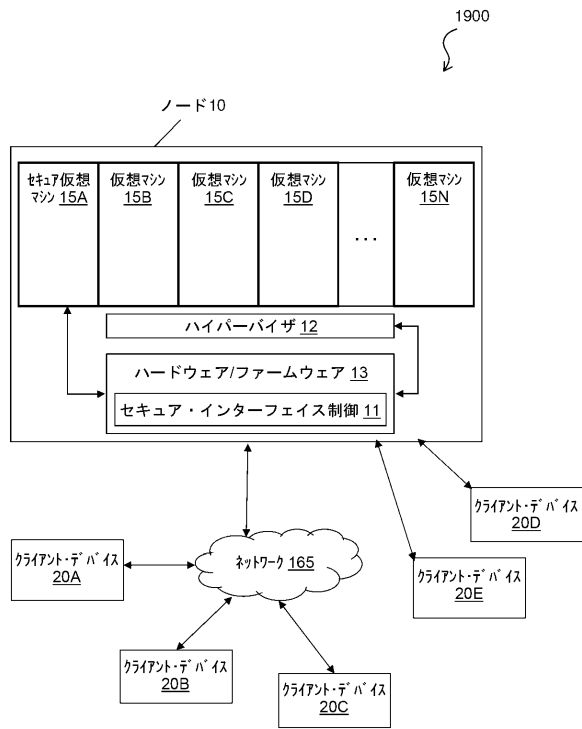
20

30

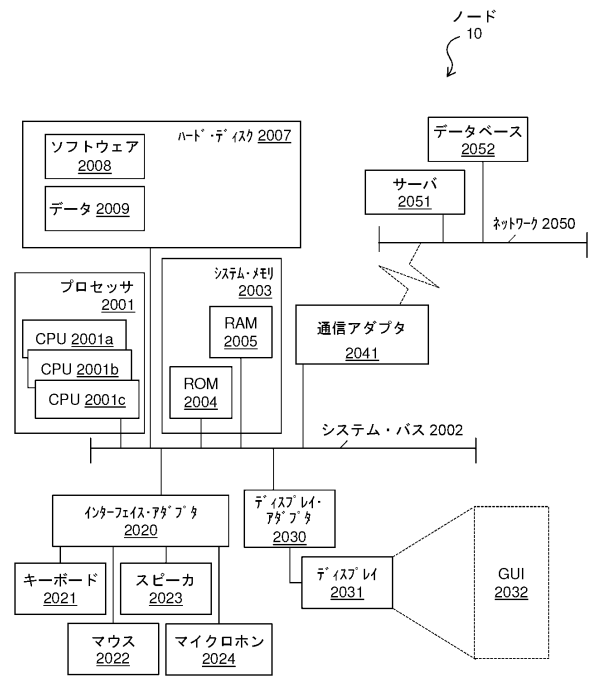
40

50

【図 19】



【図 20】



10

20

30

40

50

フロントページの続き

- (72)発明者 ブラッドベリー、ジョナサン
 アメリカ合衆国12601 ニューヨーク州ボキプシー サウス・ロード2455
- (72)発明者 ボントレーガー、クリスチャン
 ドイツ 71032 ベー布林ゲン シェーナイヒャー・シュトラーセ 220
- (72)発明者 カーステンス、ハイコ
 ドイツ 71032 ベー布林ゲン シェーナイヒャー・シュトラーセ 220
- (72)発明者 シュヴィデフスキー、マーティン
 ドイツ 71032 ベー布林ゲン シェーナイヒャー・シュトラーセ 220
- (72)発明者 ブエントゲン、ラインハルト
 ドイツ 71032 ベー布林ゲン シェーナイヒャー・シュトラーセ 220
- 審査官 吉田 歩
- (56)参考文献 米国特許出願公開第2018/0285140 (US, A1)
 特表2018-502371 (JP, A)
 特開2011-181107 (JP, A)
- (58)調査した分野 (Int.Cl., DB名)
 G06F 21/60
 G06F 21/14
 G06F 12/14