(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7379516号
(P7379516)

(45)発行日　令和5年11月14日(2023.11.14)

(24)登録日　令和5年11月6日(2023.11.6)

(51)国際特許分類
G06F 12/14 (2006.01)

FI
G06F 12/14 510E

請求項の数 20 （全40頁）

(21)出願番号　特願2021-551611(P2021-551611)
(86)(22)出願日　令和2年3月6日(2020.3.6)
(65)公表番号　特表2022-523785(P2022-523785A)
(43)公表日　令和4年4月26日(2022.4.26)
(86)国際出願番号　PCT/EP2020/056033
(87)国際公開番号　WO2020/182664
(87)国際公開日　令和2年9月17日(2020.9.17)
審査請求日　令和4年8月24日(2022.8.24)
(31)優先権主張番号　16/296,301
(32)優先日　平成31年3月8日(2019.3.8)
(33)優先権主張国・地域又は機関
米国(US)

(73)特許権者　390009531
インターナショナル・ビジネス・マシーンズ・コーポレーション
INTERNATIONAL BUSINESS MACHINES CORPORATION
アメリカ合衆国１０５０４ ニューヨーク州 アーモンク ニュー オーチャード ロード
New Orchard Road, Armonk, New York 10504, 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】　セキュア・インターフェース制御ストレージのためのホスト仮想アドレス空間使用方法、システム、プログラム

(57)【特許請求の範囲】

【請求項１】

コンピュータ・システムのセキュア・インターフェース制御において、前記コンピュータ・システムのセキュア・ドメイン内のセキュア・エンティティに関連するデータ構造へのアクセス要求を受信することと、

前記セキュア・インターフェース制御によって、前記データ構造の位置に関連付けられた仮想ストレージ・アドレスをチェックすることと、

前記セキュア・インターフェース制御によって、前記データ構造の前記位置が前記仮想ストレージ・アドレスに関連付けられているとの判定に基づいて、前記コンピュータ・システムの非セキュア・エンティティの仮想アドレス空間を使用するアドレス変換を要求することと、

前記セキュア・インターフェース制御によって、前記アドレス変換の結果に基づいて前記データ構造にアクセスすることと

を含む、方法。

【請求項２】

前記セキュア・インターフェース制御によって、前記データ構造の前記位置が前記仮想ストレージ・アドレスに関連付けられていないとの判定に基づいて、絶対アドレスを使用して前記データ構造にアクセスすることをさらに含む、請求項１に記載の方法。

【請求項３】

前記非セキュア・エンティティによって提供された前記仮想ストレージ・アドレスのマ

ッピングを検証することをさらに含む、請求項１または２に記載の方法。
【請求項４】
前記仮想ストレージ・アドレスの前記マッピングを検証することが、以前のマッピングと比較して前記マッピングの変化をチェックすることを含む、請求項３に記載の方法。
【請求項５】
前記セキュア・ドメイン内の前記セキュア・エンティティに関連する前記データ構造が、メモリの複数のページ間に分散される、請求項１ないし４のいずれか一項に記載の方法。
【請求項６】
前記非セキュア・エンティティによって提供されたメモリの前記ページが、連続した範囲の仮想アドレスに常駐する、請求項５に記載の方法。
【請求項７】
前記仮想ストレージ・アドレスをチェックすることが、ホスト仮想アドレスに関連付けられた仮想アドレス比較が有効であるか無効であるかを判定するために、ゾーン・セキュリティ・テーブルを検査することをさらに含む、請求項１ないし６のいずれか一項に記載の方法。
【請求項８】
前記セキュア・インターフェース制御が、ファームウェア、ハードウェア、信頼できるソフトウェア、またはファームウェアと、ハードウェアと、信頼できるソフトウェアとの組合せを含む、請求項１ないし７のいずれか一項に記載の方法。
【請求項９】
前記非セキュア・エンティティが、１つまたは複数のセキュア・ゲストを前記セキュア・エンティティとしてホストするように構成されたハイパーバイザを含む、請求項１ないし８のいずれか一項に記載の方法。
【請求項１０】
システムであって、
メモリと、
処理ユニットと、
セキュア・インターフェース制御と
を備え、前記セキュア・インターフェース制御が、
セキュア・ドメイン内のセキュア・エンティティに関連するデータ構造へのアクセス要求を受信することと、
前記メモリ内の前記データ構造の位置に関連付けられた仮想ストレージ・アドレスをチェックすることと、
前記データ構造の前記位置が前記仮想ストレージ・アドレスに関連付けられているとの判定に基づいて、前記処理ユニットの非セキュア・エンティティの仮想アドレス空間を使用するアドレス変換を要求することと、
前記アドレス変換の結果に基づいて前記データ構造にアクセスすることと
を含む複数の動作を実行するように構成される、システム。
【請求項１１】
前記セキュア・インターフェース制御が、
前記データ構造の前記位置が前記仮想ストレージ・アドレスに関連付けられていないとの判定に基づいて、絶対アドレスを使用して前記データ構造にアクセスすることを含む動作を実行するように構成される、請求項１０に記載のシステム。
【請求項１２】
前記セキュア・インターフェース制御が、
前記非セキュア・エンティティによって提供された前記仮想ストレージ・アドレスのマッピングを検証することを含む動作を実行するように構成される、請求項１０または１１に記載のシステム。
【請求項１３】
前記仮想ストレージ・アドレスの前記マッピングを検証することが、以前のマッピング

と比較して前記マッピングの変化をチェックすることを含む、請求項１２に記載のシステム。
【請求項１４】
前記セキュア・ドメイン内の前記セキュア・エンティティに関連する前記データ構造が、前記メモリの複数のページ間に分散される、請求項１０ないし１３のいずれか一項に記載のシステム。
【請求項１５】
前記非セキュア・エンティティによって提供された前記メモリの前記ページが、連続した範囲の仮想アドレスに常駐する、請求項１４に記載のシステム。
【請求項１６】
前記仮想ストレージ・アドレスをチェックすることが、ホスト仮想アドレスに関連付けられた仮想アドレス比較が有効であるか無効であるかを判定するために、ゾーン・セキュリティ・テーブルを検査することをさらに含む、請求項１０ないし１５のいずれか一項に記載のシステム。
【請求項１７】
前記セキュア・インターフェース制御が、ファームウェア、ハードウェア、信頼できるソフトウェア、またはファームウェアと、ハードウェアと、信頼できるソフトウェアとの組合せを含む、請求項１０ないし１６のいずれか一項に記載のシステム。
【請求項１８】
前記非セキュア・エンティティが、１つまたは複数のセキュア・ゲストを前記セキュア・エンティティとしてホストするように構成されたハイパーバイザを含む、請求項１０ないし１７のいずれか一項に記載のシステム。
【請求項１９】
請求項１～８の何れか1項に記載の方法をコンピュータに実行させる、コンピュータ・プログラム。
【請求項２０】
請求項１９に記載の前記コンピュータ・プログラムをコンピュータ可読記憶媒体に記録した、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、セキュア・インターフェース制御ストレージのためのホスト仮想アドレス空間の使用に関する。
【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、サード・パーティのデータ・センターにデータを格納してデータをそこで処理する機能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、物理サーバ用のフロアスペースを設けることも必要とせずに、顧客向けにＶＭを迅速かつ簡単にプロビジョニングする能力を容易化する。顧客は、顧客の好みまたは要件の変化に依って、ＶＭを簡単に拡張または縮小することができる。典型的には、クラウド・コンピューティング・プロバイダがＶＭをプロビジョニングし、ＶＭは、プロバイダのデータ・センターにあるサーバ上に物理的に常駐している。特にコンピューティング・プロバイダが２人以上の顧客のデータを同じサーバ上に格納することが多いので、顧客は、しばしばＶＭ内のデータのセキュリティについて懸念する。顧客は、自分のコード／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／データとの間のセキュリティ、および自分のコード／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／データとの間のセキュリティを望む場合がある。さらに、顧客は、プロバイダの管理者からのセキュリティ、およびマシン上で実行されている他のコードからの潜在的なセキュリティ違反に対するセキュリティを望む場合がある。

【０００３】
このような機密な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ストレージの区分化を保証するためのセキュリティ制御を実装する場合がある。クラウド・インフラストラクチャを実装する際に仮想化を広範に使用すると、仮想化によって、オペレーティング・システム（ＯＳ）と、コンピューティング・ハードウェア、ストレージ・ハードウェア、またはさらにはネットワーク・ハードウェアという基盤となるハードウェアとの間の関係が変化するので、クラウド・サービスの顧客に固有のセキュリティ上の懸念が生じる。これにより、仮想化は、層自体を適切に構成、管理、および保護する必要がある追加の層として導入される。
【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に依存して、そのゲストに仮想化サービスを透過的に提供する。これらのサービスには、メモリ管理、命令エミュレーション、および割り込み処理が含まれる。
【０００５】
メモリ管理の場合、ＶＭはそのデータをディスクから移動（ページ・イン）させてメモリに常駐させることができ、ＶＭはそのデータをディスクに戻す（ページ・アウト）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。さらに、ホスト・ハイパーバイザは、メモリ内のゲスト・ページの独自の（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、単独で、またゲストにとって透過的に、ゲスト・ページをメモリにページ・インおよびメモリからページ・アウトすることができる。ハイパーバイザが２つの別々のゲストＶＭ間でゲスト・メモリのメモリ分離または共有を実現する場合、ホストＤＡＴテーブルを介して実現される。ホストは、ゲスト・メモリにアクセスし、必要に応じてゲストに代わってゲスト動作をシミュレートすることも可能である。
【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、コンピュータ実施方法は、コンピュータ・システムのセキュア・インターフェース制御において、コンピュータ・システムのセキュア・ドメイン内のセキュア・エンティティに関連するデータ構造へのアクセス要求を受信することを含む。セキュア・インターフェース制御は、データ構造の位置に関連付けられた仮想ストレージ・アドレスをチェックすることができる。セキュア・インターフェース制御は、データ構造の位置が仮想ストレージ・アドレスに関連付けられているとの判定に基づいて、コンピュータ・システムの非セキュア・エンティティの仮想アドレス空間を使用するアドレス変換を要求することができる。セキュア・インターフェース制御は、アドレス変換の結果に基づいてデータ構造にアクセスすることができる。利点として、セキュア・インターフェース制御ストレージのためのホスト仮想アドレス空間を提供することが含まれ得る。
【０００７】
本発明の追加のまたは代替の実施形態によれば、セキュア・インターフェース制御は、データ構造の位置が仮想ストレージ・アドレスに関連付けられていないとの判定に基づいて、絶対アドレスを使用してデータ構造にアクセスすることができる。利点として、絶対アドレス指定におけるアドレス指定の柔軟性が含まれ得る。
【０００８】
本発明の追加のまたは代替の実施形態によれば、非セキュア・エンティティによって提供された仮想ストレージ・アドレスのマッピングが検証され得る。利点として、セキュア・エンティティに対する非セキュア・エンティティによって管理されているデータの整合性をチェックすることが含まれ得る。
【０００９】
本発明の追加のまたは代替の実施形態によれば、仮想ストレージ・アドレスのマッピン

グを検証することが、以前のマッピングと比較してマッピングの変化をチェックすることを含むことができる。利点として、非セキュア・エンティティがセキュア・エンティティのアドレス・マッピングを変更していないと確認することが含まれ得る。

【００１０】

本発明の追加のまたは代替の実施形態によれば、セキュア・ドメイン内のセキュア・エンティティに関連するデータ構造は、メモリの複数のページ間に分散され得る。利点として、データ構造の断片化をサポートすることが含まれ得る。

【００１１】

本発明の追加のまたは代替の実施形態によれば、非セキュア・エンティティによって提供されたメモリのページは、連続した範囲の仮想アドレスに常駐することができる。利点として、断片化された絶対アドレスを連続して表示させることが含まれ得る。

【００１２】

本発明の追加のまたは代替の実施形態によれば、仮想ストレージ・アドレスをチェックすることは、ホスト仮想アドレスに関連付けられた仮想アドレス比較が有効であるか無効であるかを判定するために、ゾーン・セキュリティ・テーブルを検査することをさらに含むことができる。利点として、ドメインまたはページごとにアドレス指定モード選択を管理すること含まれ得る。

【００１３】

本発明の追加のまたは代替の実施形態によれば、セキュア・インターフェース制御は、ファームウェア、ハードウェア、信頼できるソフトウェア、またはファームウェアと、ハードウェアと、信頼できるソフトウェアとの組合せとすることができる。利点として、システム全体の性能に対する関連する動作上の影響が少ないセキュア・インターフェース制御を実施することが含まれ得る。

【００１４】

本発明の追加のまたは代替の実施形態によれば、非セキュア・エンティティは、１つまたは複数のセキュア・ゲストをセキュア・エンティティとしてホストするように構成されたハイパーバイザとすることができる。利点として、非セキュア・ハイパーバイザによってセキュア・ゲストをホストすることが含まれ得る。

【００１５】

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ける上記の方法の特徴を実装する。

【００１６】

本開示の技術を通じて、追加の特徴および利点が実現される。本発明の他の実施形態および態様は、本明細書に詳細に記載されており、本発明の一部と見なされる。利点および特徴を備えた本発明をより良く理解するために、説明および図面を参照されたい。

【００１７】

本明細書の添付の特許請求の範囲において、本明細書に記載の排他的権利の詳細を具体的に取り上げ、明確に特許請求している。添付図面と併せて行う以下の詳細な説明から、本発明の実施形態の前述および他の特徴ならびに利点が明らかになる。

【図面の簡単な説明】

【００１８】

【図１】本発明の１つまたは複数の実施形態による、ゾーン・セキュリティに関するテーブルである。

【図２】本発明の１つまたは複数の実施形態による、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よる、ハイパーバイザの下で実行している仮想マシン（ＶＭ：virtual machine）をサポートするためのネストされたマルチ・パートＤＡＴ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よる、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よる、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よる、セキュア・インターフェー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よる、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よる、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よる、提供されたメモリ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よる、セキュア・インターフェース制御の非セキュア・ハイパーバイザ・ページからセキュア・ページへの移行のプロセス・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よる、セキュア・インターフェース制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１２】本発明の１つまたは複数の実施形態による、セキュア・インターフェース制御およびハードウェアによるアクセス・タグ付けのプロセス・フローを示す図である。
【図１３】本発明の１つまたは複数の実施形態による、プログラムおよびセキュア・インターフェース制御によってセキュア・アクセスおよび非セキュア・アクセスをサポートするための変換のプロセス・フローを示す図である。
【図１４】本発明の１つまたは複数の実施形態による、プログラムおよびセキュア・インターフェース制御によるセキュア・ストレージ保護を備えたＤＡＴのプロセス・フローを示す図である。
【図１５】本発明の１つまたは複数の実施形態による、アドレス指定モード決定のためのプロセス・フローを示す図である。
【図１６】本発明の１つまたは複数の実施形態による、セキュア・インターフェース制御ストレージのためのホスト仮想アドレス空間を使用するためのプロセス・フローを示す図である。
【図１７】本発明の１つまたは複数の実施形態による、クラウド・コンピューティング環境を示す図である。
【図１８】本発明の１つまたは複数の実施形態による、抽象化モデル層を示す図である。
【図１９】本発明の１つまたは複数の実施形態による、システムを示す図である。
【図２０】本発明の１つまたは複数の実施形態による、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】

【００１９】

本明細書に示されている図は例示的なものである。本発明の思想から逸脱することなく、図またはそこに記載されている動作に対する多くの変形形態が存在し得る。例えば、アクションを異なる順序で実行することができ、またはアクションを追加、削除、もしくは変更することができる。また、「結合された」という用語、およびその変形は、２つの要素間に通信経路を有することを表しており、要素間に介在する要素／接続がない、要素間の直接接続を意味するものではない。これらの変形形態はすべて、本明細書の一部と見なされる。

【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との間の効率的で軽いセキュア・インターフェース制御を活用して、追加のセキュリティを提供する。

【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、そのゲストに仮想化サービスを透過的に提供するそのハイパーバイザに依存する。これらのサービスは、セキュア・エンティティと、他のエンティティによるセキュア・リソースへのアクセスを従来から許可している別の信頼できないエンティティとの間の任意のインタ

ーフェースに適用することができる。前述のように、これらのサービスには、メモリ管理、命令エミュレーション、および割り込み処理が含まれるが、これらに限定されない。例えば、割り込みおよび例外の挿入の場合、ハイパーバイザは通常、ゲストのプレフィックス領域（低コア）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。本明細書で使用する「仮想マシン」または「ＶＭ」という用語は、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）の論理表現を指す。ＶＭは、基盤となるホスト・マシン（物理プロセッサまたはプロセッサのセット）上で実行す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・リソースの観点からは、ＶＭは独自の独立した物理マシンのように見える。本明細書で使用する「ハイパーバイザ」および「ＶＭモニタ（ＶＭＭ）」という用語は、同じホスト・マシン上で複数の（場合によっては異なる）ＯＳを使用して複数のＶＭを実行するように管理および許可する処理環境またはプラットフォーム・サービスを指す。ＶＭの展開は、ＶＭのインストール・プロセスおよびＶＭのアクティブ化（または開始）プロセスを含むことを理解されたい。別の例では、（例えば、ＶＭが以前にインストールされているか、すでに存在する場合）ＶＭの展開は、ＶＭのアクティブ化（または開始）プロセスを含む。

【００２２】

セキュア・ゲストを円滑化およびサポートするためには、ハイパーバイザがＶＭからのデータにアクセスできず、したがって上記の方法でサービスを提供できないように、ハイパーバイザとセキュア・ゲストとの間にハイパーバイザに依存することのない追加のセキュリティが必要となるという技術上の問題が存在する。

【００２３】

本明細書に記載のセキュアな実行は、セキュア・ストレージと非セキュア・ストレージとの間、および異なるセキュア・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・ゲストとの間に追加のセキュリティが提供される。これを実施するには、典型的にはハイパーバイザがゲストに代わって実行する機能の多くをマシンに組み込む必要がある。本明細書において、ハイパーバイザとセキュア・ゲストとの間のセキュアなインターフェースを提供するための、本明細書では「ＵＶ」とも呼ぶ新しいセキュア・インターフェース制御について説明する。本明細書において、セキュア・インターフェース制御およびＵＶという用語は同義で使用される。セキュア・インターフェース制御は、ハードウェアと連携して作用し、この追加のセキュリティを提供する。さらに、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、下位レベルのハイパーバイザが、信頼できるコード／ソフトウェアに実装されている場合、その下位レベルのハイパーバイザもまた、セキュア・インターフェース制御の一部とすることができる。

【００２４】

一例において、セキュア・インターフェース制御は、セキュアで信頼できる内部の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には、例えば、プロセッサ・ミリコードまたはＰＲ／ＳＭ論理パーティション・コードが含まれ得る。セキュア・ゲストまたはセキュア・エンティティの場合、セキュア・インターフェース制御は、セキュア環境の初期化および保守、ならびにハードウェア上のこれらのセキュア・エンティティのディスパッチの調整を提供する。セキュア・ゲストがデータを能動的に使用しており、データがホスト・ストレージに常駐している間、データは、セキュア・ストレージ内で「クリア状態」に保たれる。セキュア・ゲスト・ストレージには、その単一のセキュア・ゲストからアクセスすることができ、これは、ハードウェアによって厳密に強制される。すなわち、ハードウェアは、任意の非セキュア・エンティティ（ハイパーバイザ、もしくは他の非セキュア・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止する。この例では、セキュア・インターフェース制御は、最下位レベルのファームウェアの信頼できる部分として実行する。最下位

レベル、すなわちミリコードは、実際にはハードウェアの拡張であり、例えばＩＢＭが提供しているｚＡｒｃｈｉｔｅｃｔｕｒｅ（Ｒ）で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、セキュアな実行のコンテキストで独自のセキュアＵＶ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。これにより、セキュア・ゲストまたはそのゲストをサポートするハイパーバイザが必要とする任意の機能を提供することができる。セキュア・インターフェース制御はまた、ハードウェアに直接アクセスするので、ハードウェアは、セキュア・インターフェース制御によって確立された条件の制御下でセキュリティ・チェックを効率的に提供することができる。

【００２５】

本発明の１つまたは複数の実施形態によれば、セキュア・ページにマークするために、ハードウェア内にセキュア・ストレージ・ビットが提供される。このビットが設定されている場合、ハードウェアは、任意の非セキュア・ゲストまたはハイパーバイザがこのページにアクセスするのを防止する。さらに、各セキュア・ページまたは共有ページは、ゾーン・セキュリティ・テーブルに登録され、セキュア・ゲスト・ドメイン識別情報（ＩＤ）でタグ付けされる。ページがセキュリティで非セキュアである場合、ページは、ゾーン・セキュリティ・テーブルにおいて非セキュアとしてマークされる。このゾーン・セキュリティ・テーブルは、区分またはゾーンごとのセキュア・インターフェース制御によって維持される。ホスト絶対ページごとに１つのエントリがあり、セキュア・エンティティによって行われる任意のＤＡＴ変換時に、ハードウェアがエントリを使用して、ページを所有するセキュア・ゲストまたはエンティティによってのみページがアクセスされることを検証する。

【００２６】

本発明の１つまたは複数の実施形態によれば、セキュア・インターフェース制御は、セキュア・インターフェース制御自体によってのみアクセスされ得る独自のセキュアＵＶストレージを有する。このストレージは、セキュア・ゲストに必要なセキュリティを提供するために、セキュア・インターフェース制御およびハードウェアによって使用される。セキュア・インターフェース制御は、このセキュア・ストレージを使用して、セキュア・インターフェース制御自体、セキュア・ゲストを実行することが可能なゾーン、セキュア・ゲスト、およびセキュア仮想ＣＰＵに関する情報を記憶する。セキュア・ゲスト・ストレージと同様に、セキュア・インターフェース制御ストレージも、非セキュア・エンティティによるアクセスを防止するためにセキュア・ページとしてマークされる。さらに、セキュア・インターフェース制御ストレージは、任意の他のセキュア・エンティティがセキュア・インターフェース制御ストレージにアクセスするのを防止するために使用される独自のセキュア・ドメインＩＤを有する。

【００２７】

本発明の１つまたは複数の実施形態は、セキュア・インターフェース制御ストレージのためのホスト仮想アドレス空間を使用する。セキュア・インターフェース制御、ゲスト、および仮想ＣＰＵに関する情報を記憶するためのメモリ領域は、通常、絶対メモリに配置されており、これはセキュア・インターフェース制御がアドレス変換無しでメモリ領域に直接アクセスできることを意味する。セキュアなスワッピングを可能にするために、ホストからセキュア・インターフェース制御に提供されたメモリ領域内で、整合性テーブルなどのデータ構造を作成、維持、および使用することができる。整合性テーブルは、ページがディスクにスワップ・アウトされるときに各ページの内容および関連情報のチェックサムを保持することができるので、ページが再びスワップ・インされると、ページの内容をチェックすることが可能となる。整合性テーブルのサイズは、セキュア・ゲストのストレージのサイズに依存するので広範なメモリ領域にまたがることがある。整合性テーブルやその他のセキュア・インターフェース制御のデータ構造は、実行時に割り当てられることが可能であり、データ構造ごとに連続メモリとして表示される必要があり得る。システムがある程度の間実行されていると、メモリ断片化に起因して、ストレージの広範な連続領

域を見つけることが困難になることがある。本発明の１つまたは複数の実施形態は、システムがある程度の間実行されている場合でも、セキュア・インターフェース制御のための広範な連続領域のストレージの割り当てを可能にすることによって、メモリ断片化の問題に対する回避策を提供する。セキュア・インターフェース制御を絶対アドレスとしてアドレス・メモリのみに限定するのではなく、セキュア・インターフェース制御は、信頼できないハイパーバイザのアドレス空間内の仮想ストレージを使用して、整合性テーブルなどの１つまたは複数のデータ構造を記憶することができる。これにより、セキュア・インターフェース制御は、動的アドレス変換（ＤＡＴ）を使用して、セキュア・インターフェース制御ストレージのホスト仮想アドレス空間内の構造体にアクセスすることが可能になる。

【００２８】

本発明の１つまたは複数の実施形態によれば、ソフトウェアは、ＵＶ呼出し（ＵＶＣ：UV Call）命令を使用して、特定のアクションを実行するためのセキュア・インターフェース制御を要求する。例えば、ＵＶＣ命令は、ハイパーバイザがセキュア・インターフェース制御を初期化し、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、そのセキュア構成内に仮想ＣＰＵを作成するために使用され得る。ＵＶＣ命令はまた、セキュア・ゲスト・ページをハイパーバイザのページ・インまたはページ・アウト動作の一部としてインポートし（復号し、セキュア・ゲスト・ドメインに割り当て）、エクスポートする（暗号化し、ホスト・アクセスを許可する）ためにも使用され得る。さらに、セキュア・ゲストは、ハイパーバイザと共有するストレージを定義し、セキュア・ストレージを共有させ、共有ストレージをセキュアにするための能力を有する。

【００２９】

これらのＵＶＣコマンドは、他の多くの設計された命令と同様に、マシン・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・インターフェース制御モードに入らないが、代わりにマシンは、現在実行しているモードでセキュア・インターフェース制御機能を実行する。ハードウェアは、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ので、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はない。この低いオーバーヘッドにより、必要なレベルのセキュリティを提供しながらセキュア・インターフェース制御の複雑さを最小限に抑えて軽減する方法で、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様々な層間の緊密な連携が可能になる。

【００３０】

本発明の１つまたは複数の実施形態によれば、セキュア・インターフェース制御およびハードウェアがセキュア・ゲストを適切に維持してハイパーバイザ環境をサポートするために必要とされる制御ブロック構造をサポートするために、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、セキュア・インターフェース制御にストレージを提供する。その結果、１）セキュア・ゲストを実行するためのゾーンを初期化する、２）セキュア・ゲスト・ドメインを作成する、および３）ドメインの各々で実行されるセキュアＣＰＵを作成するための準備として、ハイパーバイザは、とりわけ、提供に必要なストレージの容量を決定するためのクエリＵＶＣ命令を発行する。ストレージが提供されると、ストレージはセキュアとしてマークされ、セキュア・インターフェース制御に属するものとして登録され、非セキュアまたはセキュアなゲスト・エンティティによるアクセスは禁止される。これは、関連するエンティティ（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継続する。

【００３１】

一例において、ゾーン固有ＵＶ制御ブロックをサポートするためのＵＶストレージの第１のセクションは、セキュア・インターフェース制御に初期化ＵＶＣの一部として提供され、本明細書でＵＶ２ストレージと呼ぶ場所に常駐する。（セキュア・ゲスト・ドメインごとに）基本および可変のセキュア・ゲスト構成制御ブロックをサポートするための、ＵＶストレージの第２のセクションおよび第３のセクションは、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常駐する。セキュアＣＰＵ制御ブロックをサ

ポートするためのＵＶストレージの第４の最後のセクションも、ＵＶＳ空間に常駐し、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。これらのエリアの各々がそれぞれ提供されると、セキュア制御インターフェースは、（各エリアがすべての非セキュア・エンティティによってアクセスされないように）これらのエリアをセキュアとしてマークし、また（これらのエリアがすべてのセキュア・ゲスト・エンティティによってアクセスされないように）これらのエリアをゾーン・セキュリティ・テーブルにセキュア・インターフェース制御に属するものとして登録する。ＵＶ空間内でさらなる分離を実現するために、（どの特定のセキュア・ゲスト・ドメインに関連付けられていない）ＵＶ２空間も、一意のＵＶ２セキュア・ドメインでタグ付けされ、ＵＶＳ空間とＵＶＶ空間はどちらも、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空間はホスト仮想空間に常駐し、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。

【００３２】

セキュア・インターフェー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）にアクセスできるが、本発明の１つまたは複数の実施形態は、セキュア・インターフェース制御が非常に具体的にＵＶストレージにアクセスすることを可能に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。これにより、セキュア・インターフェース制御が、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、所望の指定されたセキュア・ゲストのセキュア・ゲスト・ストレージにのみアクセスすること、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が保証される。すなわち、セキュア・インターフェース制御は、セキュア・インターフェース制御がアクセスしようとするストレージにハードウェアが実際にアクセスすることを保証できるように、そのストレージを極めて明示的に指定してもよい。また、セキュア・インターフェース制御はさらに、セキュア・インターフェース制御が、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ストレージにのみアクセスしようとするように指定することができる。

【００３３】

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・データを透過的にページ・インおよびページ・アウトするとき、ハードウェアと連携するセキュア・インターフェース制御は、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は、ハイパーバイザは、ゲスト・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ＵＶＣ中にセキュア・インターフェース制御によってセットアップされた制御に基づいて、これらのＵＶＣが実際にハイパーバイザによって発行されることを保証する。

【００３４】

この新しいセキュア環境では、ハイパーバイザは、セキュア・ページをページ・アウトしている場合は常に、セキュア・ストレージからの新しい変換（エクスポート）ＵＶＣを発行する必要がある。セキュア・インターフェース制御は、このエクスポートＵＶＣに応答して、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ロックをリセットする。エクスポートＵＶＣが完了すると、ここでハイパーバイザは、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。

【００３５】

さらに、ハイパーバイザは、セキュア・ページをページ・インしている場合は常に、セキュア・ストレージへの新しい変換（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・インターフェース制御は、このインポートＵＶＣに応答して、１）ハードウェアにおいてページをセキュアとしてマークし、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

ンに権限を設定し、５）ＵＶロックをリセットする。セキュア・エンティティによってアクセスが行われる場合は常に、ハードウェアは、変換中にそのページに対して権限チェックを実行する。これらのチェックには、１）アクセスしようとしているセキュア・ゲスト・ドメインにページが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、ハイパーバイザがこのページのホスト・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が含まれる。ページがセキュアとしてマークされると、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止する。追加の変換ステップは、別のセキュアＶＭによるアクセスを防止し、ハイパーバイザによる再マッピングを防止する。

【００３６】

次に図１を参照すると、本発明の１つまたは複数の実施形態によるゾーン・セキュリティのテーブル１００が概略的に示されている。図１に示すゾーン・セキュリティ・テーブル１００は、セキュア・エンティティによってアクセスされる任意のページへのセキュア・アクセスを保証するために、セキュア・インターフェース制御によって維持され、セキュア・インターフェー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つのエントリがある。各エントリは、エントリが、アクセスを行うセキュア・エンティティに属していることを検証するために使用される情報を含む。

【００３７】

さらに、図１に示すように、ゾーン・セキュリティ・テーブル１００は、セキュア・ドメインＩＤ１２０（この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・インターフェース制御に提供され、セキュア・インターフェー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（ホスト絶対と定義されているセキュア・インターフェース制御ページが、関連するホスト仮想アドレスを有していない場合などに、特定の状況でホスト・アドレス・ペアの比較を無効に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に対して登録されているホスト仮想アドレスを示し、これをホスト・アドレス・ペアと呼ぶ）とを含む。なお、ホスト・アドレス・ペアは、ホスト絶対アドレスと、関連する登録済みのホスト仮想アドレスとを示す。ホスト・アドレス・ペアは、ハイパーバイザによってインポートされた時点でのこのページのマッピングを表し、この比較は、ページがゲストによって使用されている間にホストがそのページを再マッピングしないことを保証する。

【００３８】

動的アドレス変換（ＤＡＴ）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ＶＭ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とき、ゲストは、ＤＡＴを使用してそのメモリに常駐するページを管理する。さらに、ホストは、ページがメモリに常駐しているとき、単独でＤＡＴを使用してそれらのゲスト・ページを（ホスト自体のページと共に）管理する。ハイパーバイザは、ＤＡＴを使用して、異なるＶＭ間でのストレージの分離または共有あるいはその両方を実現し、ハイパーバイザ・ストレージへのゲスト・アクセスを防止する。ゲストが非セキュア・モードで実行しているとき、ハイパーバイザは、ゲストのストレージのすべてにアクセスする。

【００３９】

ＤＡＴにより、アプリケーションを別のアプリケーションから分離しながら、依然としてそれらのアプリケーションに共通リソースを共有させることが可能になる。また、ＤＡＴによりＶＭの実装も可能であり、ＶＭは、アプリケーション・プログラムの同時処理、ならびに新しいバージョンのＯＳの設計およびテストに使用されてもよい。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、仮想アドレスの連続シーケンスであり、特定の変換パラメータ（ＤＡＴテーブルを含む）を用いて、各仮想アドレスを、その各仮想アドレスをストレージ内のバイト位置で識別する関連する絶対アドレスに変換できるようにする。

【００４０】

ＤＡＴは、マルチ・テーブル・ルックアップを使用して、仮想アドレスを関連する絶対アドレスに変換する。このテーブル構造は、典型的には、ストレージ・マネージャによって定義および維持される。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・アウトして別のページを取り込むことによって、絶対ストレージを複数のプログラム間で透過的に共有する。例えば、ページがページ・アウトされると、ストレージ・マネージャは、関連するページ・テーブルに無効ビットを設定する。ページ・アウトされたページにプログラムがアクセスしようとすると、ハードウェアは、しばしばページ・フォールトと呼ばれるプログラム割り込みをストレージ・マネージャに提示する。それに応答してストレージ・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。これはすべて、プログラムに対して透過的に行われ、ストレージ・マネージャがストレージを仮想化して様々な異なるユーザ間で共有できるようにする。

【００４１】

ＣＰＵが仮想アドレスを使用してメイン・ストレージにアクセスする場合、仮想アドレスは最初に、ＤＡＴによって実アドレスに変換され、次いでプレフィックス変換（prefixing）によっ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上位レベルのテーブルの指定（起点および長さ）は、アドレス空間制御要素（ＡＳＣＥ：address-space-control element）と呼ばれ、関連するアドレス空間を定義する。

【００４２】

次に図２を参照すると、本発明の１つまたは複数の実施形態による、ＤＡＴ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が概略的に示されている。図２に示す例では、仮想アドレス空間２０２（アドレス空間制御要素（ＡＳＣＥ）Ａ２０８によって定義される）と、仮想アドレス空間２０４（ＡＳＣＥ　Ｂ２１０によって定義される）との２つの仮想アドレス空間が存在する。マルチ・テーブル（セグメント２３０およびページ・テーブル２３２ａ、２３２ｂ）ルックアップにおいて、ストレージ・マネージャによってＡＳＣＥ　Ａ２０８を使用して、仮想ページＡ１．Ｖ２１２ａ１、Ａ２．Ｖ２１２ａ２、およびＡ３．Ｖ２１２ａ３は、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、２３４および２３６の２つのテーブル・ルックアップにおいて、ＡＳＣＥ　Ｂ２１０を使用して、仮想ページＢ１．Ｖ２１４ｂ１およびＢ２．Ｖ２１４ｂ２は、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。

【００４３】

次に図３を参照すると、本発明の１つまたは複数の実施形態による、ハイパーバイザの下で実行しているＶＭをサポートするために使用されるネストされたマルチ・パートＤＡＴ変換の例が概略的に示されている。図３に示す例では、ゲストＡ仮想アドレス空間Ａ３０２（ゲストＡＳＣＥ（ＧＡＳＣＥ）Ａ３０４によって定義される）とゲストＢ仮想アドレス空間Ｂ３０６（ＧＡＳＣＥＢ３０８によって定義される）はどちらも共有ホスト（ハイパーバイザ）仮想アドレス空間３２５内に常駐する。図示のように、ゲストＡストレージ・マネージャによってＧＡＳＣＥＡ３０４を使用して、ゲストＡに属する仮想ページＡ１．ＧＶ３１０ａ１、Ａ２．ＧＶ３１０ａ２、およびＡ３．ＧＶ３１０ａ３は、ゲスト絶対ページ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ストレージ・マネージャによって単独でＧＡＳＣＥＢ３０８を使用して、ゲストＢに属する仮想ページＢ１．ＧＶ３２０ｂ１およびＢ２．ＧＶ３２０ｂ２は、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、追加のホストＤＡＴ変換を経て、ホス

ト絶対アドレス空間３３０まで進む。図示のように、ホスト・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は、Ａ１．ＨＡ３７０ａ１、Ａ３．ＨＡ３７０ａ３、およびＢ１．ＨＡ３７０ｂ１にマッピングされる。ゲストＡに属するホスト仮想アドレスＡ２．ＨＶ３４０ａ２とゲストＢに属するホスト仮想アドレスＢ２．ＨＶ３６０ｂ２はどちらも、同じホスト絶対ページＡＢ２．ＨＡ３８０にマッピングされる。これにより、この２人のゲスト間でデータを共有することが可能になる。ゲストＤＡＴ変換中、ゲスト・テーブル・アドレスはそれぞれ、ゲスト絶対アドレスとして扱われ、追加のネストされたホストＤＡＴ変換の対象となる。

【００４４】

本明細書に記載の本発明の実施形態は、セキュア・ゲストおよびＵＶストレージ保護を提供する。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は禁止される。ハイパーバイザは、常駐している所与のセキュア・ゲスト・ページに対して、以下の動作が行われるようにする。関連するホスト絶対アドレスは、単一のハイパーバイザ（ホスト）ＤＡＴマッピングを介してのみアクセス可能である。すなわち、セキュア・ゲストに割り当てられた任意の所与のホスト絶対アドレスにマッピングする単一のホスト仮想アドレスが存在する。所与のセキュア・ゲスト・ページに関連する（ホスト仮想からホスト絶対への）ハイパーバイザＤＡＴマッピングは、ページ・インされている間は変更されない。セキュア・ゲスト・ページに関連するホスト絶対ページは、単一のセキュア・ゲストにマッピングされる。

【００４５】

本発明の１つまたは複数の実施形態によれば、セキュア・ゲスト間でのストレージの共有も禁止される。ストレージは、単一のセキュア・ゲストとセキュア・ゲストの制御下にあるハイパーバイザとの間で共有される。ＵＶストレージは、セキュア・ストレージであり、セキュア・インターフェース制御からアクセスできるが、ゲスト／ホストからはアクセスできない。ストレージは、ハイパーバイザによってセキュア・インターフェー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、これらの規則に対するすべての違反を試みることは、ハードウェアおよびセキュア・インターフェース制御によって禁止されている。

【００４６】

次に図４を参照すると、本発明の１つまたは複数の実施形態によるセキュア・ゲスト・ストレージのマッピングの例が概略的に示されている。図４の例がセキュア・ゲストＡとセキュア・ゲストＢとの間のストレージの共有を可能にしないことを除いて、図４は図３と類似している。図３の非セキュアの例では、ゲストＡに属するホスト仮想アドレスＡ２．ＨＶ３４０ａ２とゲストＢに属するホスト仮想アドレスＢ２．ＨＶ３６０ｂ２のどちらも、同じホスト絶対ページＡＢ２．ＨＡ３８０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に属するホスト仮想アドレスＡ２．ＨＶ３４０ａ２は、ホスト絶対アドレスＡ２．ＨＡ４９０ａにマッピングされ、一方、ゲストＢに属するホスト仮想アドレスＢ２．ＨＶ３６０ｂ２は、それ自体のＢ２．ＨＡ４９０ｂにマッピングされる。この例では、セキュア・ゲスト間での共有はない。

【００４７】

セキュア・ゲスト・ページは、ディスク上に常駐しているが、暗号化されている。ハイパーバイザがセキュア・ゲスト・ページにページ・インすると、ハイパーバイザはＵＶ呼出し（ＵＶＣ）を発行し、ＵＶ呼出し（ＵＶＣ）は、セキュア・インターフェース制御に、ページをセキュアとしてマークさせ（共有されていない場合）、ページを復号させ（共有されていない場合）、ページを適切なセキュア・ゲスト（例えば、ゲストＡ）に属するものとして（ゾーン・セキュリティ・テーブルに）登録させる。さらに、セキュア・インターフェース制御は、関連するホスト仮想アドレス（例えば、Ａ３．ＨＶ３４０ａ３）をそのホスト絶対ページ（ホスト・アドレス・ペアと呼ぶ）に登録する。ハイパーバイザが正しいＵＶＣ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

、ハイパーバイザは例外を受信する。ハイパーバイザがゲスト・ページをページ・アウトすると、同様のＵＶＣが発行され、ハイパーバイザは、ゲスト・ページを非セキュアとしてマークしてそのゲスト・ページをゾーン・セキュリティ・テーブルに非セキュアとして登録する前に、（共有されていない場合）ゲスト・ページを暗号化する。

【００４８】

５つの所与のホスト絶対ページＫ、Ｐ、Ｌ、Ｍ、およびＮがある例では、ハイパーバイザがホスト絶対ページをページ・インするとき、ホスト絶対ページはそれぞれ、セキュア・インターフェース制御によってセキュアとしてマークされる。これにより、非セキュア・ゲストおよびハイパーバイザがそのページにアクセスするのを防止する。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、ホスト絶対ページＫ、Ｐ、およびＭは、ゲストＡに属するものとして登録され、ホスト絶対ページＬおよびＮは、ハイパーバイザによってページ・インされるとき、ゲストＢに登録される。共有ページ、すなわち単一のセキュア・ゲストとハイパーバイザとの間で共有されるページは、ページング中に暗号化も復号もされない。共有ページは、（ハイパーバイザによるアクセスを許可する）セキュアとしてマークされないが、ゾーン・セキュリティ・テーブル内の単一のセキュア・ゲスト・ドメインに登録される。

【００４９】

本発明の１つまたは複数の実施形態によれば、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、ハイパーバイザは、セキュア・ストレージ・アクセス（ＰＩＣ３Ｄ）例外を受信する。これを判定するために追加の変換ステップは必要ない。

【００５０】

１つまたは複数の実施形態によれば、セキュア・エンティティがページにアクセスしようとすると、ハードウェアは、ストレージが実際にその特定のセキュア・ゲストに属していることを検証する追加の変換チェックを実行する。属していない場合、ハイパーバイザに非セキュア・アクセス（ＰＩＣ３Ｅ）例外が提示される。さらに、変換されるホスト仮想アドレスが、ゾーン・セキュリティ・テーブルに登録されているホスト・アドレス・ペアからのホスト仮想アドレスと一致しない場合、セキュア・ストレージ違反（「３Ｆ」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、変換チェックでアクセスが許可されている限り、セキュアとしてマークされていないストレージにアクセスしてもよい。

【００５１】

次に図５を参照すると、本発明の１つまたは複数の実施形態による、ＤＡＴ動作のシステム概略図５００が概略的に示されている。システム概略図５００は、ホスト１次仮想アドレス空間５１０およびホスト・ホーム仮想アドレス空間５２０を含み、ページはこれらの空間から、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。なお、点線は、ＤＡＴ変換５２５を介したマッピングを表す）。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、さらに２人のゲスト間だけでなくホスト自体との、これらのホスト仮想アドレスのうちの１つの共有も示す。これに関して、ホスト１次仮想アドレス空間５１０およびホスト・ホーム仮想アドレス空間５２０は、２つのホスト仮想アドレス空間の例であり、各ホスト仮想アドレス空間は、別個のＡＳＣＥ、ホスト１次ＡＳＣＥ（ＨＰＡＳＣＥ）５９１、およびホスト・ホームＡＳＣＥ（ＨＨＡＳＣＥ）５９２によってそれぞれアドレス指定される。なお、すべてのセキュア・インターフェース制御ストレージ（仮想ストレージと実ストレージの両方）は、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される。セキュア・インターフェース制御ストレージが提供されると、関連するセキュア・エンティティが存在する限り、セキュア・インターフェース制御ストレージには、セキュア・インターフェース制御からのみアクセスすることができる。

【００５２】

図示のように、ホスト１次仮想アドレス空間５１０は、ゲストＡ絶対ページＡ１．ＨＶ

、ゲストＡ絶対ページＡ２．ＨＶ、ゲストＢ絶対ページＢ１．ＨＶ、およびホスト仮想ページＨ３．ＨＶを含む。ホスト・ホーム仮想アドレス空間５２０は、セキュア・インターフェース制御仮想ページ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を含む。

【００５３】

本発明の１つまたは複数の実施形態によれば、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）ストレージはすべて、本明細書に記載のゾーン・セキュリティ・テーブルに、セキュア・ゲスト構成に属するものとして登録され、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・ペア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、セキュア・ゲスト・ストレージはすべて、ホスト１次仮想空間にマッピングされる。さらに、セキュア・インターフェース制御ストレージはすべて、同様にゾーン・セキュリティ・テーブルに、セキュア・インターフェース制御に属するものとして登録され、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もよい。本発明の１つまたは複数の実施形態によれば、ＵＶ仮想ストレージは、ホスト・ホーム仮想空間にマッピングされ、関連するホスト仮想アドレスは、ホスト・アドレス・ペアの一部として登録される。１つまたは複数の実施形態によれば、ＵＶ実ストレージは、関連するホスト仮想マッピングを有さず、ゾーン・セキュリティ・テーブル内の（仮想アドレス比較が無効であることを示す）ＤＡビットは、これを示すように設定される。ホスト・ストレージは、非セキュアとしてマークされ、ゾーン・セキュリティ・テーブルにも非セキュアとして登録される。

【００５４】

したがって、「ゲスト絶対＝ホスト仮想」の場合、（ＨＰＡＳＣＥ５９１によって定義される）ハイパーバイザ（ホスト）１次ＤＡＴテーブルは、ホスト１次仮想アドレス空間５１０のページを次のように変換する。ゲストＡ絶対ページＡ１．ＨＶは、セキュア・ゲストＡに属するホスト絶対Ａ１．ＨＡにマッピングされ、ゲストＡ絶対ページＡ２．ＨＶは、セキュア・ゲストＡに属するホスト絶対Ａ２．ＨＡにマッピングされ、ゲストＢ絶対ページＢ１．ＨＶは、セキュア・ゲストＢに属するホスト絶対Ｂ１．ＨＡにマッピングされ、ホスト仮想ページＨ３．ＨＶは、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・ペアはない）。さらに、（ＨＨＡＳＣＥ５９２によって定義される）ハイパーバイザ（ホスト）・ホームＤＡＴテーブルは、ホスト・ホーム仮想アドレス空間５２０のページを次のように変換する。セキュア・インターフェース制御仮想ページＵ１．ＨＶは、セキュアＵＶ仮想と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は、非セキュアと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は、非セキュアと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、Ｈ１．ＨＡまたはＨ２．ＨＡのいずれかに関連するホスト・アドレス・ペアはない。

【００５５】

動作中、セキュア・インターフェース制御に割り当てられたセキュア・ページにセキュア・ゲストがアクセスしようとすると、ハードウェアによってハイパーバイザにセキュア・ストレージ違反（「３Ｆ」Ｘ）例外が提示される。非セキュア・ゲストまたはハイパーバイザが（セキュア・インターフェース制御に割り当てられたページを含む）いずれかのセキュア・ページにアクセスしようとすると、ハードウェアによってハイパーバイザにセキュア・ストレージ・アクセス（「３Ｄ」Ｘ）例外が提示される。代替として、セキュア・インターフェース制御空間に対して行われたアクセスの試行について、エラー状態を提示することができる。ハードウェアがセキュア・インターフェース制御アクセスにおいてセキュアな割り当ての不一致を検出した場合（例えば、ストレージがセキュア・インターフェース制御ではなくセキュア・ゲストに属するものとしてゾーン・セキュリティ・テーブルに登録されている場合、または登録済みのペアで使用されているホスト・アドレス・

ペアに不一致がある場合）、チェックが提示される。

【００５６】

言い換えれば、ホスト１次仮想アドレス空間５１０は、（セキュア・ゲストＡに属する）ホスト仮想ページＡ１．ＨＶおよびＡ２．ＨＶ、ならびに（セキュア・ゲストＢに属する）ホスト仮想ページＢ１．ＨＶを含み、これらは、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。さらに、ホスト１次仮想アドレス空間５１０は、ホスト（ハイパーバイザ）・ページＨ３．ＨＶを含み、これは、ホスト絶対Ｈ３．ＨＡにマッピングされる。ホスト・ホーム仮想空間５２０は、２つのホスト仮想ページＨ１．ＨＶおよびＨ２．ＨＶを含み、これらは、ホスト絶対ページＨ１．ＨＡおよびＨ２．ＨＡにマッピングされる。ホスト１次仮想アドレス空間５１０とホスト・ホーム仮想アドレス空間５２０とはどちらも、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に属するストレージ・ページは、セキュアとしてマークされ、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に示すゾーン・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される。ハイパーバイザがセキュア・ゲストを定義している場合、ハイパーバイザは、これらのセキュア・ゲストのサポートに必要なセキュア制御ブロックに使用するために、セキュア・インターフェース制御にホスト・ストレージを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、一例では、具体的にはホスト・ホーム仮想空間内で定義することができる。図５に戻ると、ホスト絶対ページＵ１．ＨＡおよびホスト絶対ページＵ２．ＨＡセキュアＵＶ絶対は、ホスト絶対ストレージと定義されているセキュア・インターフェース制御ストレージである。結果として、これらのページはセキュアとしてマークされ、関連するセキュア・ドメインと共に、図１に示すゾーン・セキュリティ・テーブル１００にセキュア・インターフェース制御に属するものとして登録される。ページはホスト絶対アドレスと定義されるので、関連するホスト仮想アドレスは存在せず、したがってゾーン・セキュリティ・テーブル１００にＤＡビットが設定される。

【００５７】

図６において、変換後のハイパーバイザ（ホスト）絶対アドレス空間５３０の例を見出すことができる。図６には、本発明の１つまたは複数の実施形態による、セキュア・インターフェー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非セキュア（ホスト）、ホスト絶対ページＨ２．ＨＡ非セキュア（ホスト）、ホスト絶対ページＵ３．ＨＡセキュアＵＶ実（ＨＶマッピング無し）、ホスト絶対ページＵ１．ＨＡセキュアＵＶ仮想（Ｕ１．ＨＶ用）、ホスト絶対ページＡ１．ＨＡセキュア・ゲストＡ（Ａ１．ＨＶ用）を含む、ハイパーバイザ（ホスト）絶対アドレス空間６３０を示す。

【００５８】

次に図７を参照すると、本発明の１つまたは複数の実施形態による、インポート動作のためのプロセス・フロー７００が概略的に示されている。ハイパーバイザによってページ・アウトされたページにセキュア・ゲストがアクセスすると、そのページを安全に戻すために、プロセス・フロー７００に示すような一連のイベントが発生する。プロセス・フロー７００はブロック７０５で開始し、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、ページが無効であるため、ハードウェアは、プログラム割り込みコード１１（ＰＩＣ１１）によって示されるホスト・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７１５を参照）。次に、ハイパーバイザは、このゲスト・ページに対して使用可能な非セキュア・ホスト絶対ページを識別し（ブロック７２０を参照）、識別されたホスト絶対ページに暗号化されたゲスト・ページをページ・インする（ブロック７２５を参照）。

【００５９】

次いで、ブロック７３０において、ホスト絶対ページは、（ホスト仮想アドレスに基づ

く）適切なホストＤＡＴテーブルにマッピングされる。次いで、ブロック７３５において、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７４０において、セキュア・ゲストは、ゲスト・セキュア・ページに再アクセスする。ページ・フォールトはもはや存在しないが、このセキュア・ゲスト・アクセスおよびページは図１のゾーン・セキュリティ・テーブル１００にセキュアとしてマークされないので、ブロック７４５においてハードウェアは、非セキュア・ストレージ例外（ＰＩＣ３Ｅ）をハイパーバイザに提示する。このＰＩＣ３Ｅ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止する。次に、プロセス・フロー７００は、図８に接続されている「Ａ」に進む。

【００６０】

次に図８を参照すると、本発明の１つまたは複数の実施形態による、インポート動作を実行するためのプロセス・フロー８００が概略的に示されている。正常に動作する（例えば、エラー無しで期待どおりに実行する）ハイパーバイザは、ＰＩＣ３Ｅに応答して、インポートＵＶＣを発行する（ブロック８０５を参照）。なお、この時点で、インポートされるページは、非セキュアとしてマークされ、ハイパーバイザ、他の非セキュア・エンティティ、およびセキュア・インターフェース制御によってのみアクセスされ得る。インポートされるページに、セキュア・ゲストによってアクセスすることはできない。

【００６１】

インポートＵＶＣの一部として、セキュア・インターフェース制御として機能する信頼できるファームウェアは、このページがセキュア・インターフェース制御によってすでにロックされているかどうかを確認するためにチェックする（判定ブロック８１０を参照）。すでにロックされている場合、プロセス・フロー８００はブロック８２０に進む。ブロック８２０において、「ビジー」リターン・コードがハイパーバイザに返され、ハイパーバイザは、応答して、遅延し（ブロック８２５を参照）、インポートＵＶＣを再発行する（プロセス・フロー８００はブロック８０５に戻る）。ページがまだロックされていない場合、プロセス・フロー８００は、判定ブロック８２２に進む。

【００６２】

判定ブロック８２２において、セキュア・インターフェース制御は、ページが非セキュア・ハイパーバイザと共有されているページであるかどうかを確認するためにチェックする。ページが共有されている場合（プロセス・フロー８００は判定ブロック８２４に進み）、セキュア・インターフェース制御は、ゾーン・セキュリティ・テーブル内のホスト絶対アドレスを、関連するセキュア・ゲスト・ドメインおよびホスト仮想アドレスに、共有として登録する。このページは、非セキュアとしてマークされたままである。これでインポートＵＶＣが完了し、ここでゲストは、ページにアクセスして利用できるようになる。処理は、ハイパーバイザがゲストを再ディスパッチし（ブロック８３０）、セキュア・ゲストがページに正常にアクセスする（ブロック８３５）ことで続行する。

【００６３】

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はブロック８４０に進み）、セキュア・インターフェース制御は、ハイパーバイザはそのページにアクセスできなくなるようにページをセキュアとしてマークする。ブロック８４５において、他のＵＶＣはページのステータスを変更することができないように、セキュア・インターフェース制御はページをロックする。ロックが設定されると、（ブロック８５０において）セキュア・インターフェース制御は、ゲスト・ページの内容が、暗号化されている間に変更されていないことを検証する。ゲスト・ページの内容が変更された場合、ハイパーバイザにエラー・リターン・コードが返され、変更されていない場合、セキュア・インターフェース制御はセキュア・ページを復号する。

【００６４】

ブロック８５５において、セキュア・インターフェース制御は、ページをロック解除して他のＵＶＣによるアクセスを許可し、ページをゾーン・セキュリティ・テーブルにセキュアとして登録し、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、ホ

スト・アドレスＨＶ－＞ＨＡペアを完成させる。これにより、ゲストによるアクセスが可能になり、ＵＶＣが完了する。

【００６５】

次に図９を参照すると、本発明の１つまたは複数の実施形態による、提供されたメモリの動作に関するプロセス・フロー９００が概略的に示されている。プロセス・フロー９００はブロック９０５で開始し、ハイパーバイザがセキュア・インターフェース制御にクエリＵＶＣを発行する。ブロック９１０において、セキュア・インターフェース制御は、データ（例えば、クエリＵＶＣ）を返す。このデータは、必要とされる基本ゾーン固有ホスト絶対ストレージの容量、必要とされる基本セキュア・ゲスト・ドメイン固有ホスト絶対ストレージの容量、ＭＢごとの必要となる可変セキュア・ゲスト・ドメイン固有ホスト仮想ストレージの容量、または必要となる基本セキュア・ゲストＣＰＵ固有ホスト絶対ストレージの容量、あるいはその組合せを含むことができる。

【００６６】

ブロック９１５において、ハイパーバイザは、（例えば、クエリＵＶＣによって返されるサイズに基づいて）基本ホスト絶対ゾーン固有ストレージを確保する。ブロック９２０において、ハイパーバイザは、セキュア・インターフェース制御に初期化を発行する。この点に関連して、ハイパーバイザは、提供されたストレージを、ゾーン全体のセキュア・ゲスト構成間で調整するために必要なＵＶ制御ブロックに提供する、初期化ＵＶＣを発行することができる。初期化ＵＶＣは、基本ゾーン固有ストレージの起点を指定する。

【００６７】

ブロック９２５において、セキュア・インターフェース制御は、提供されたストレージをＵＶに登録してセキュアとしてマークすることによって、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・インターフェース制御は、提供されたストレージをセキュアとしてマークし、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、ＵＶで使用するために、ゾーン・セキュリティ・テーブルに一意のセキュア・ドメインと共に登録することができるが、関連するセキュア・ゲスト・ドメインはなく、関連するホスト仮想アドレス・ペアはないものとして登録する。

【００６８】

ブロック９３０において、ハイパーバイザは、ストレージ（例えば、基本および可変のセキュア・ゲスト・ドメイン固有ストレージ）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・ストレージのサイズに基づく）基本および可変のセキュア・ゲスト・ドメイン固有ストレージ（例えば、クエリＵＶＣによって返されるサイズ）を確保する。ブロック９３５において、ハイパーバイザは、セキュア・インターフェース制御に構成作成を発行する。この点に関連して、ハイパーバイザは、基本および可変のセキュア・ゲスト・ドメイン固有ストレージの起点を指定する、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、提供されたストレージを、このセキュア・ゲスト構成をサポートするために必要なＵＶ制御ブロックに提供する。

【００６９】

ブロック９４０において、セキュア・インターフェース制御は、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・インターフェース制御は、提供されたストレージをセキュアとしてマークし、提供されたストレージをＵＶで使用するためにゾーン・セキュリティ・テーブルに登録し、提供されたストレージを関連するセキュア・ゲスト・ドメインに登録することができる。提供された基本（ホスト絶対）ストレージは、関連するホスト仮想アドレス・ペアを有していないものとして登録される。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・ペアに登録される。

【００７０】

ブロック９４５において、ハイパーバイザは基本セキュア・ゲストＣＰＵ固有ストレー

ジ（例えば、クエリＵＶによって返されるサイズ）を確保する。ブロック９５０において、ハイパーバイザは、ストレージの起点を指定する。例えば、ハイパーバイザは、基本セキュア・ゲストＣＰＵ固有ストレージの起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において、セキュア・インターフェース制御は、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・インターフェース制御は、提供されたストレージをセキュアとしてマークし、提供されたストレージを、ＵＶで使用するために、ゾーン・セキュリティ・テーブルに登録することができるが、関連するセキュア・ゲスト・ドメインはなく、関連するホスト仮想アドレス・ペアはないものとして登録する。

【００７１】

次に図１０を参照すると、本発明の１つまたは複数の実施形態による、セキュア・インターフェース制御の非セキュア・ハイパーバイザ・ページからセキュア・ページへの移行に関するプロセス・フロー１０００が概略的に示されている。プロセス・フロー１０００において、３つのハイパーバイザ・ページ（例えば、非セキュア・ハイパーバイザ・ページＡ、非セキュア・ハイパーバイザ・ページＢ、および非セキュア・ハイパーバイザ・ページＣ）を示す。

【００７２】

ハイパーバイザ（非セキュア）・ページＡ、Ｂ、およびＣには、非セキュア・エンティティ（ハイパーバイザを含む）によってアクセスすることができる。さらに、ハイパーバイザ（非セキュア）・ページＡ、Ｂ、およびＣは、非セキュア（ＮＳ）としてマークされるとともに、ゾーン・セキュリティ・テーブル（例えば、図１に示すゾーン・セキュリティ・テーブル１００）に非セキュアおよび非共有として登録される。矢印１００５において、初期化ＵＶＣが発行され、これにより、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・インターフェース制御実ストレージ・ページ１０１０に移行させる。セキュア・インターフェース制御実ストレージ１０１０は、セキュアとしてマークされるとともに、ゾーン・セキュリティ・テーブル（例えば、図１に示すゾーン・セキュリティ・テーブル１００）に、セキュア・ゲスト・ドメイン無し、ハイパーバイザからホスト絶対への（ＨＶ－＞ＨＡ）マッピング無しで、ＵＶとして登録され得る。代わりに、セキュア・インターフェース制御実ストレージ１０１０は、一意のＵＶ２セキュア・ドメインに登録され、ＤＡビットが１に設定される。なお、セキュア・インターフェース制御実ストレージ１０１０は、セキュア・インターフェース制御によって実アクセスとしてアクセスされ得る。

【００７３】

ハイパーバイザ（非セキュア）・ページＢからの場合、矢印１０２５において、ＳＧ構成作成またはＳＧ－ＣＰＵ作成ＵＶＣが発行され、これにより、このページをセキュア・ゲスト・ドメイン（ＵＶＳ）に関連付けられたセキュア・インターフェース制御実ストレージ１０３０に移行させる。セキュア・インターフェース制御実ストレージ１０３０は、セキュアとしてマークされるとともに、ゾーン・セキュリティ・テーブル（例えば、図１に示すゾーン・セキュリティ・テーブル１００）に、関連するセキュア・ゲスト・ドメイン有り、ハイパーバイザからホスト絶対への（ＨＶ－＞ＨＡ）マッピング無し（すなわち、ＤＡビット＝１）で、ＵＶとして登録され得る。なお、セキュア・インターフェース制御実ストレージ１０１０は、セキュア・ゲスト・ドメインに代わってセキュア・インターフェース制御によって実アクセスとしてアクセスされ得る。

【００７４】

ハイパーバイザ（非セキュア）・ページＣの場合、矢印１０４５において、ＳＧ構成作成ＵＶＣが発行され、これにより、このページをセキュア・ゲスト・ドメイン（ＵＶＶ）に関連付けられたセキュア・インターフェース制御仮想ストレージ１０５０に移行させる。セキュア・インターフェース制御仮想ストレージ１０５０は、セキュアとしてマークされるとともに、ゾーン・セキュリティ・テーブル（例えば、図１に示すゾーン・セキュリティ・テーブル１００）に、セキュア・ゲスト・ドメイン有り、ハイパーバイザからホス

ト絶対への（ＨＶ－＞ＨＡ）マッピング有りで、ＵＶとして登録され得る。なお、セキュア・インターフェース制御仮想ストレージ１０５０は、セキュア・ゲスト・ドメインに代わってＵＶ仮想アクセスとしてアクセスされ得る。

【００７５】

次に図１１を参照すると、１つまたは複数の実施形態による、プログラムまたはセキュア・インターフェー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。このフローは、セキュア・インターフェース制御がゲスト・ストレージまたはセキュア・インターフェース制御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という状況を表す。１１００は、セキュア・インターフェース制御によるストレージ・アクセスのこのタグ付けについて説明している。プロセス・フロー１１００はブロック１１１０で開始し、セキュア・インターフェース制御は、セキュア・インターフェース制御がセキュア・インターフェース制御ストレージにアクセスしているかどうかを判定する。

【００７６】

アクセスがセキュア・インターフェース制御ストレージへのアクセスでない場合、プロセス・フロー１１００は、（いいえの矢印で示すように）判定ブロック１１１２に進む。判定ブロック１１１２において、セキュア・インターフェース制御は、セキュア・インターフェース制御がセキュア・ゲスト・ストレージにアクセスしているかどうかを判定する。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は（図１２のプロセス・フロー１２００に接続されている）「Ｂ」に進み、非セキュア・アクセス用のデフォルト設定を使用することになる。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は判定ブロック１１１３に進み、セキュア・インターフェース制御は、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。はいの場合、プロセス・フロー１１００は、（図１２のプロセス・フロー１２００に接続されている）「Ｂ」に進み、セキュア・ゲスト・アクセス用のデフォルト設定を使用することになる。いいえの場合、プロセス・フロー１１００はブロック１１１４に進む。ブロック１１１４において、適切なセキュア・ゲスト・ドメインがＳＧセキュア・ドメイン・レジスタにロードされる（そして、図１２のプロセス・フロー１２００に接続されている「Ｂ」に進む）。

【００７７】

アクセスがセキュア・インターフェース制御ストレージへのアクセスである場合、プロセス・フロー１１００は、（はいの矢印で示すように）ブロック１１２０に進む。ブロック１１２０において、アクセスは、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）。

【００７８】

次いで、プロセス・フロー１１００は、判定ブロック１１３０に進み、セキュア・インターフェース制御は、アクセスがＵＶＶ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。アクセスがＵＶＶ空間へのアクセスである場合、プロセス・フロー１１００は、（はいの矢印で示すように）ブロック１１３４に進む。ブロック１１３４において、アクセスは仮想アクセスとしてタグ付けされる。ブロック１１３６において、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・レジスタにロードされる。ブロック１１３８において、ＤＡＴ変換およびストレージへのアクセスを開始する準備ができる。判定ブロック１１３０に戻ると、アクセスがＵＶＶ空間へのアクセスでない場合、プロセス・フロー１１００は、（いいえの矢印で示すように）ブロック１１４０に進む。ブロック１１４０において、アクセスは、実アクセスとしてタグ付けされる。

【００７９】

判定ブロック１１５０において、セキュア・インターフェース制御は、このアクセスがＵＶＳ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ＵＶＳ空間へのアクセスである場合、プロセス・フロー１１００は、（はいの矢印で示すように）ブロック１１３６に進む。このアクセスがＵＶＳ空間へのアクセスでない場合、プロセス・フロー１１００は、（いいえの矢印で示すように）ブロック１１７０に進む。このアクセスは、この場合、ＵＶ２空間（例えば、ゾーン・セキュリティ・テーブル）へのアクセスになる。ブロック１１７０において、一意のＵＶ２セキュア・ドメインがＵＶセキュア・ドメイン・レジスタにロードされる。

【００８０】

図１２は、本発明の１つまたは複数の実施形態によるプロセス・フロー１２００を示す。ゲストがディスパッチされると、ＳＩＥエントリ・ファームウェアは、ゲストが実行されていること（例えば、ゲスト・モードがアクティブであること）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、ハードウェアに（例えば、ＳＧセキュア・ドメイン・レジスタに）ロードされ得る。プログラムがストレージにアクセスしているとき、ハードウェアは、アクセス時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、プロセス・フロー１２００におけるこのプロセスの例を示している。ブロック１２０５において、ハードウェアは、マシンが現在ゲスト・モードで実行されているかどうかを判定することができ、ゲスト・モードで実行されていない場合、ブロック１２１０でアクセスをホスト・アクセスであるとしてタグ付けし、ブロック１２１５で非セキュア・アクセスであるとしてタグ付けすることができる。ブロック１２０５において、マシンがゲスト・モードで実行されている場合、ブロック１２２０において、アクセスは、ゲスト・アクセスとしてタグ付され得、ハードウェアはさらに、ブロック１２２５において、現在のゲストがセキュア・ゲストであるかどうかを判定することができる。ゲストがセキュアでない場合、ブロック１２１５において、アクセスは非セキュアとしてタグ付けされ得る。ゲストがセキュアである場合、ハードウェアは、ブロック１２３０において、ゲストをセキュアとしてタグ付けすることができ、これにより、セキュア・ゲストを、セキュア・ゲストがディスパッチされたときにロードされたＳＧセキュア・ドメイン・レジスタに関連付けることができる。ブロック１２３５において、非セキュア・ゲストとセキュア・ゲストの両方について、ＤＡＴステータスがチェックされ得る。ＤＡＴがオフの場合、ブロック１２４０において、アクセスは実アクセスとしてタグ付けされ得る。ＤＡＴがオンの場合、ブロック１２４５において、アクセスは仮想アクセスとしてタグ付けされ得る。ＤＡＴがオフであることによりブロック１２４０においてアクセスが実アクセスとしてタグ付けされるか、ＤＡＴがオンであることによりブロック１２４５でアクセスが仮想アクセスとしてタグ付けされると、ブロック１２５０において、ハードウェアは、図１３でさらに説明するように変換およびストレージへのアクセスを開始する準備ができる。

【００８１】

図１３は、本発明の１つまたは複数の実施形態による、プロセス・フロー１３００におけるセキュア・アクセスと非セキュア・アクセスの両方をサポートするためにハードウェアによって行われる変換の例を示す。ブロック１３０５において、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、ゲスト変換としてタグ付けされており、ブロック１３１０においてアクセスが仮想アクセスである場合、ブロック１３１５において、ゲストＤＡＴが実行され得る。ゲストＤＡＴ変換中、ゲストＤＡＴテーブルについて、ネストされた中間フェッチが発生する可能性がある。元の変換がセキュアとしてタグ付けされている場合、テーブル・フェッチは、ゲスト実アクセスおよびセキュアとしてタグ付けされ得る。テーブル・フェッチは、プロセス・フロー１３００の変換プロセスに従うこともできる。ブロック１３１５において、ゲスト仮想アクセス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において、ゲスト実アクセスとしてタグ付けされた任意のアクセス（仮想アクセス＝いいえ）に対してゲストＤＡＴが実行された後、ブロック１３２０において、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、ブロック１３２５において、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、結果として得られたアドレスは、ホスト仮想およびセキュアとしてタグ付けされ得る。プロセス１３００は、ホスト仮想アクセス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・アクセスであり（ゲスト＝いいえ）、ブロック１３３０で仮想アクセスである場合、ブロック１３３５において、ホストＤＡＴが実行され得る。ブロック１３３５において、ホスト・テーブル・フェッチは、非セキュアとしてマークされ得る。ブロック１３３５でホストＤＡＴが実行された後、またはブロック１３３０で元のホスト・アクセスが実アクセスとしてタグ付けされた場合（仮想アクセス＝いいえ）、ブロック１３４０において、ホスト・プレフィックス変換が適用され得る。ブロック１３４５において、結果として得られたアドレスは、ホスト絶対アドレスであり得る。

【００８２】

図１４は、本発明の１つまたは複数の実施形態による、プロセス・フロー１４００におけるハードウェアによって実行され得るセキュア・ストレージ保護を備えたＤＡＴ変換の例を示す。図１３のブロック１３４５から続いて、ブロック１４０５においてセキュアＵＶアクセスが識別された場合、ブロック１４１０において、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、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においてエラーが提示される。ＵＶストレージにアクセスするとき、セキュア・インターフェース制御によってセキュアＵＶアクセスを行うことができる。ブロック１４１０において、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、（セキュアＵＶアクセスを行う前にセキュア・インターフェース制御によってセットアップされた）ＵＶセキュア・ドメイン・レジスタが、処理が続行するブロック１４２０でのドメイン・チェックのための指定されたセキュア・ドメインとして使用され得ることを除いて、保護チェックは、あらゆるセキュア・アクセスに対して実行され得るように継続することができる。さらに、（エントリ・ポイントＤの）ブロック１４２５でＵＶアクセスについて検出された任意の違反は、ブロック１４２５でのセキュア・ゲスト違反（セキュアＵＶ＝いいえ）に対して行わ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１４３０においてエラーとして提示され得る。

【００８３】

ブロック１４０５でセキュアＵＶアクセスとしてタグ付けされないアクセスの場合、ブロック１４４０において、ハードウェアは、アクセスがセキュア・ゲスト・アクセスであるかどうかを判定し、セキュア・ゲスト・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されている場合、ブロック１４３５において、ハイパーバイザに例外が提示され得る。そうではなく、ブロック１４４０でアクセスがセキュア・ゲスト・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されていない場合、ブロック１４５０で変換が成功する。

【００８４】

ブロック１４４０において、アクセスがセキュア・ゲスト・アクセスである場合、またはブロック１４１０において、アクセスがセキュア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において、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・エンティティに登録されていることを確認するためにチェックすることができる。アクセスがセキュアＵＶアクセスである場合、ＵＶセキュア・ドメイン・レジスタから、（アクセスされているセキュアＵＶストレージに基づいてセキュア・インターフェース制御によってロードされた）指定されたセキュア・ドメインを取得することができ、セキュア・ゲスト・アクセスの場合、ＳＧセキュア・ドメイン・レジスタから、（セキュア・エンティティがディスパッチされたときにロードされた）指定されたセキュア・ドメインを取得する。ブロック１４２０において、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アクセスのとき、ブロック１４３０においてエラーが

発生し、ブロック１４２５でセキュア・ゲスト・アクセスのとき（セキュアＵＶ＝いいえ）、ブロック１４３５においてハイパーバイザに例外が提示される。

【００８５】

ブロック１４４０およびブロック１４１０において、ストレージへのセキュア・アクセスであり、ブロック１４２０において、それらが、指定されたセキュア・ドメインに登録されている場合、ブロック１４５５で仮想アドレス・チェックが無効であり、すなわちＤＡビット＝１であり、ブロック１４６０でアクセスが実アクセスであるとき、ブロック１４５０で変換が完了する。しかしながら、ブロック１４５５でＤＡビット＝１であるが、ブロック１４６０でアクセスが仮想アクセスである場合（実アクセス＝いいえ）、ブロック１４２５でセキュアＵＶアクセスのとき、ブロック１４３０においてエラーが発生し、ブロック１４２５でセキュア・ゲスト・アクセスのとき（セキュアＵＶ＝いいえ）、ブロック１４３５においてハイパーバイザに例外が提示される。ブロック１４５５でＤＡビット＝０であり、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において、ハードウェアは、アクセスのホスト仮想からホスト絶対へのマッピングが、このホスト絶対アドレスについて登録されたマッピングと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で変換は正常に完了する。ブロック１４７０においてマッピングが一致しない場合、ブロック１４２５でセキュアＵＶアクセスのとき、ブロック１４３０においてエラーが発生し、ブロック１４２５でセキュア・ゲスト・アクセスのとき（セキュアＵＶ＝いいえ）、ブロック１４３５においてハイパーバイザに例外が提示される。ＤＡビット＝０であり、ブロック１４７５でアクセスが実アクセスである場合（仮想アクセス＝いいえ）、ブロック１４２５でセキュアＵＶアクセスのとき、ブロック１４３０においてエラーが発生し、ブロック１４２５でセキュア・ゲスト・アクセスのとき（セキュアＵＶ＝いいえ）、ブロック１４３５においてハイパーバイザに例外が提示される。あるいは、ブロック１４５０で変換が正常に完了する場合がある。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されているかどうかを確認するためにチェックすることができ、ページがセキュアである場合、ブロック１４３５においてハイパーバイザに例外を提示することができ、ページがセキュアとしてマークされていない場合、ブロック１４５０で変換が成功する。

【００８６】

ストレージ登録およびマッピングの様々なチェックは、ゾーン・セキュリティ・テーブル・インターフェース１４８５を介して一括して管理され得る。例えば、ブロック１４１０、１４２０、１４５５、１４７０、および１４７５は、同じゾーンに関連付けられたゾーン・セキュリティ・テーブルとインターフェースして、様々なアクセスを管理することができる。

【００８７】

次に図１５を参照すると、本発明の１つまたは複数の実施形態による、アドレス指定モード決定のためのプロセス・フロー１５００が概略的に示されている。図１０に関して前述したように、セキュア・インターフェース制御は、セキュア・インターフェース制御実ストレージ１０３０を介してアクセス可能な実／絶対ストレージとして確立されたＵＶＳ内のページＢなどのページにアクセスしてもよい。さらに、セキュア・インターフェース制御は、図１０のプロセス・フロー１０００においてセキュア・インターフェース制御仮想ストレージ１０５０として確立されたＵＶＶ内のページＣなどのページにアクセスしてもよい。図１１のプロセス・フロー１１００は、ブロック１１３０において、整合性テーブルなどのためのＵＶＶ空間へのアクセスが実行されるべきかどうか、またはブロック１１５０において、ＵＶＳ空間へのアクセスが実行されるべきかどうかを判定することができる。プロセス・フロー１５００はさらに、図１１のプロセス・フロー１１００をサポートすることができるアドレス変換プロセスを示す。図１５のブロック１５０５において、セキュア・インターフェース制御が、図１の整合性テーブルまたはゾーン・セキュリティ・テーブル１００などのセキュア・ゲストに関連する１つまたは複数のデータ構造にアク

セスする必要がある場合、プロセス・フロー１５００はブロック１５１０に進む。ブロック１５１０において、セキュア・インターフェース制御は、アクセスが仮想ストレージ・アドレスに関連付けられているかどうかを判定することができる。アクセスが仮想ストレージ・アドレスに関連付けられている場合、ブロック１５１５において、セキュア・インターフェース制御は、ＤＡＴテーブルなどを介して、ハイパーバイザの仮想アドレス空間を使用して仮想ストレージ・アドレスを変換することができる。ブロック１５２０において、セキュア・インターフェース制御またはサポートするハードウェア／ファームウェアあるいはその両方は、仮想ストレージ・アドレスの期待されるマッピング（例えば、この絶対ページに以前に登録されたマッピング）がハイパーバイザＤＡＴを介したアクセスから返されることを保証するために、１つまたは複数のチェックを実行することができる。一例として、セキュア・インターフェース制御は、以前に構成されたデータ構造の関連するホスト・マッピングをハイパーバイザが変更しなかったことを確認することができる。ハイパーバイザのＤＡＴテーブルを介して絶対アドレスが決定されるか、またはアクセスがすでに絶対アドレスに関連付けられている場合（例えば、ブロック１５１０＝いいえ）、ブロック１５２５において、セキュア・インターフェース制御は、絶対アドレスでデータ構造にアクセスすることができる。

【００８８】

次に図１６を参照すると、本発明の１つまたは複数の実施形態による、セキュア・インターフェース制御ストレージのためのホスト仮想アドレス空間を使用するためのプロセス・フロー１６００が概略的に示されている。プロセス・フロー１６００は、図１５のプロセス・フロー１５００の変形形態である。ブロック１６０５において、コンピュータ・システムのセキュア・インターフェース制御は、コンピュータ・システムのセキュア・ドメイン内のセキュア・エンティティに関連するデータ構造へのアクセス要求を受信することができる。アクセス要求は、セキュア・インターフェース制御を通じてセキュリティを管理する内部シーケンスの一部とすることができる。セキュア・エンティティは、ＶＭなどの１つまたは複数のセキュア・ゲスト、またはセキュア・コンテナとすることができる。ブロック１６１０において、セキュア・インターフェース制御は、データ構造の位置に関連付けられた仮想ストレージ・アドレスをチェックすることができる。セキュア・ドメイン内のセキュア・エンティティに関連するデータ構造は、メモリの複数のページ間に分散され得る。セキュア・インターフェース制御によるセキュアな使用のために、メモリのページは、複数の固定位置にある非セキュア・エンティティによって提供されてもよい。非セキュア・エンティティによって提供されたメモリのページは、連続した範囲の仮想アドレスに常駐することができる。非セキュア・エンティティは、１つまたは複数のセキュア・ゲストまたはセキュア・コンテナをセキュア・エンティティとしてホストするように構成されたハイパーバイザまたはオペレーティング・システムとすることができる。仮想ストレージ・アドレス・マッピングをチェックすることは、ホスト仮想アドレスに関連付けられた仮想アドレス比較が有効であるか無効であるか（例えば、ＤＡビット１４０の状態）を判定するために、図１のゾーン・セキュリティ・テーブル１００を検査することを含むことができる。

【００８９】

ブロック１６１５において、セキュア・インターフェース制御は、データ構造の位置が仮想ストレージ・アドレスに関連付けられているとの判定に基づいて、コンピュータ・システムの非セキュア・エンティティの仮想アドレス空間を使用してアドレス変換を実行することができる。アドレス変換は、前述のプロセスおよび要素のいずれかを使用して実行され得る。非セキュア・エンティティによって提供される仮想ストレージ・アドレスのマッピングは、セキュア・インターフェース制御によって検証され得る。仮想ストレージ・アドレスのマッピングの検証は、以前に登録されたマッピングと比較したマッピングの変更についてチェックすることを含んでもよい。アドレス変換テーブルによるマッピングは、連続していない絶対位置に分散されているメモリのページを連続した仮想アドレスとして提示するように構成され得る。したがって、絶対メモリの連続ブロックに収まることが

できないデータ構造が、ホストの仮想アドレス空間において連続しているように表示され得る。

【００９０】

ブロック１６２０において、セキュア・インターフェース制御は、アドレス変換の結果として得られる絶対アドレスに基づいて、データ構造にアクセスすることができる。代替として、セキュア・インターフェース制御は、データ構造の位置が仮想ストレージ・アドレスに関連付けられていないとの判定に基づいて、絶対アドレスを使用してデータ構造に直接アクセスすることができる。

【００９１】

本開示はクラウド・コンピューティングに関する詳細な説明を含むが、本明細書に記載された教示の実装がクラウド・コンピューティング環境に限定されないことを理解されたい。むしろ、本発明の実施形態は、現在知られている、または後に開発される他の任意のタイプのコンピューティング環境と組み合わせて実装することが可能である。

【００９２】

クラウド・コンピューティングは、最小限の管理労力、またはサービス・プロバイダとの最小限の対話で迅速にプロビジョニングおよび解放され得る構成可能なコンピューティング・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）の共用プールへの簡便かつ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のモデルである。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つの展開モデルを含むこともできる。

【００９３】

特徴は、以下の通りである。

【００９４】

オンデマンド・セルフサービス：クラウド・コンシューマは、サービス・プロバイダとの間で人間の対話を必要とすることなく、必要に応じて自動的に、サーバ時間およびネットワーク・ストレージなどのコンピューティング機能を一方的にプロビジョニングすることができる。

【００９５】

広範なネットワーク・アクセス：機能は、ネットワーク上で利用可能であり、異種のシン・クライアント・プラットフォームまたはシック・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）による使用を促進する標準的なメカニズムを介してアクセスされる。

【００９６】

リソースのプール化：プロバイダのコンピューティング・リソースは、マルチテナント・モデルを使用して複数のコンシューマにサービス提供するようにプール化され、異なる物理リソースおよび仮想リソースが、要求に応じて動的に割り当ておよび再割り当てされる。コンシューマは一般に、提供されるリソースの正確な位置に対して制御も知識も有していないが、より高い抽象化レベルでは位置（例えば、国、州、またはデータ・センター）を特定し得るという点で、位置の独立性があるといえる。

【００９７】

迅速な柔軟性：機能を、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して素早くスケール・アウトし、迅速に解放して素早くスケール・インすることができる。コンシューマにとっては、プロビジョニングに利用可能な機能は、しばしば無制限であるように見え、いつでも任意の数量で購入することができる。

【００９８】

サービスの測定：クラウド・システムは、サービスのタイプ（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・アカウント）に適した一定の抽象化レベルでの計量機能を活用することによって、リソースの使用を自動的に制御および最適化する。リソースの使用状況を監視、制御、および報告することができ、利用するサービスのプロバ

イダとコンシューマの両方に透明性を提供する。
【００９９】
サービス・モデルは、以下の通りである。
【０１００】
ソフトウェア・アズ・ア・サービス（ＳａａＳ）：クラウド・インフラストラクチャ上で動作しているプロバイダのアプリケーションを使用するために、コンシューマに提供される機能である。アプリケーションは、ウェブ・ブラウザ（例えば、ウェブ・ベースの電子メール）などのシン・クライアント・インターフェースを介して様々なクライアント・デバイスからアクセス可能である。限定されたユーザ固有のアプリケーション構成設定を想定される例外として、コンシューマは、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礎となるクラウド・インフラストラクチャを管理も制御もしない。
【０１０１】
プラットフォーム・アズ・ア・サービス（ＰａａＳ）：プロバイダによってサポートされるプログラミング言語およびツールを使用して生成されたコンシューマが生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上に展開するために、コンシューマに提供される機能である。コンシューマは、ネットワーク、サーバ、オペレーティング・システム、またはストレージなどの基礎となるクラウド・インフラストラクチャを管理も制御もしないが、展開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をホストする環境構成を制御する。
【０１０２】
インフラストラクチャ・アズ・ア・サービス（ＩａａＳ）：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み得る任意のソフトウェアをコンシューマが展開および動作させることができる、処理、ストレージ、ネットワーク、および他の基本的なコンピューティング・リソースをプロビジョニングするために、コンシューマに提供される機能である。コンシューマは、基礎となるクラウド・インフラストラクチャを管理も制御もしないが、オペレーティング・システム、ストレージ、展開されたアプリケーションを制御し、場合によっては選択されたネットワーキング・コンポーネント（例えば、ホスト・ファイアウォール）を限定的に制御する。
【０１０３】
展開モデルは、以下の通りである。
【０１０４】
プライベート・クラウド：クラウド・インフラストラクチャは、ある組織のためだけに運用される。このクラウド・インフラストラクチャは、組織またはサード・パーティによって管理されてもよく、オンプレミスまたはオフプレミスで存在してもよい。
【０１０５】
コミュニティ・クラウド：クラウド・インフラストラクチャは複数の組織で共有され、関心事項（例えば、ミッション、セキュリティ要件、ポリシー、およびコンプライアンス上の考慮事項）を共有している特定のコミュニティをサポートする。このクラウド・インフラストラクチャは、組織またはサード・パーティによって管理されてもよく、オンプレミスまたはオフプレミスで存在してもよい。
【０１０６】
パブリック・クラウド：クラウド・インフラストラクチャは、一般公衆または大規模な業界グループにとって利用可能であり、クラウド・サービスを販売する組織によって所有される。
【０１０７】
ハイブリッド・クラウド：クラウド・インフラストラクチャは、固有のエンティティのままであるが、データおよびアプリケーションの移植性（例えば、クラウド間の負荷分散のためのクラウド・バースティング）を可能にする標準化された技術または専用の技術によって互いに結び付けられる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）の合成である。

【０１０８】

クラウド・コンピューティング環境は、ステートレス性、低結合性、モジュール性、および意味的相互運用性に焦点を置くことを重視したサービスである。クラウド・コンピューティングの中心は、相互接続されたノードのネットワークを含むインフラストラクチャである。

【０１０９】

次に図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。図示のように、クラウド・コンピューティング環境５０は、例えば、携帯情報端末（ＰＤＡ：personal digital assistant）もしくは携帯電話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あるいはその組合せなどのクラウド・コンシューマによって使用されるローカル・コンピューティング・デバイスが通信することができる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を含む。ノード１０は、互いに通信してもよい。ノード１０は、本明細書で上述したようなプライベート・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合せなどの１つまたは複数のネットワーク内で物理的にまたは仮想的にグループ化されてもよい（図示せず）。これにより、クラウド・コンピューティング環境５０は、インフラストラクチャ、プラットフォーム、またはソフトウェア・アズ・ア・サービス、あるいはその組合せを、クラウド・コンシューマがローカル・コンピューティング・デバイス上にリソースを保持する必要のないサービスとして提供することが可能になる。図１７に示すコンピューティング・デバイス５４Ａ～５４Ｎのタイプは、例示のみを意図しており、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、（例えば、ウェブ・ブラウザを使用して）任意のタイプのネットワークまたはネットワーク・アドレス指定可能な接続あるいはその両方を介して任意のタイプのコンピュータ化されたデバイスと通信できることを理解されたい。

【０１１０】

次に図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抽象化層のセットが示されている。図１８に示すコンポーネント、層、および機能は、例示のみを意図しており、本発明の実施形態はそれらに限定されないことを予め理解されたい。図示のように、以下の層および対応する機能が提供される。

【０１１１】

ハードウェアおよびソフトウェア層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には、メインフレーム６１、ＲＩＳＣ（縮小命令セット・コンピュータ）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、記憶デバイス６５、ならびにネットワークおよびネットワーキング・コンポーネント６６が含まれる。いくつかの実施形態では、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。

【０１１２】

仮想化層７０は、抽象化層を提供し、この層から仮想エンティティの以下の例、すなわち、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が提供され得る。

【０１１３】

一例において、管理層８２は、以下に記載の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コンピューティング・リソースおよび他のリソースの動的な調達を提供する。計量および価格決定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ときのコスト追跡、およびこれらのリソースの消費に対する課金または請求を提供す

る。一例において、これらのリソースは、アプリケーション・ソフトウェア・ライセンスを含むことがある。セキュリティは、クラウド・コンシューマおよびタスクのための本人確認、ならびにデータおよび他のリソースのための保護を提供する。ユーザ・ポータル８３は、コンシューマおよびシステム管理者にクラウド・コンピューティング環境へのアクセスを提供する。サービス・レベル管理８４は、要求されるサービス・レベルが満たされるように、クラウド・コンピューティング・リソースの割り当ておよび管理を提供する。サービス・レベル・アグリーメント（ＳＬＡ）の計画および履行８５は、ＳＬＡに従って将来において要求されることが予想されるクラウド・コンピューティング・リソースの事前配置および調達を提供する。

【０１１４】

ワークロード層９０は、クラウド・コンピューティング環境が利用され得る機能の例を提供する。この層から提供され得るワークロードおよび機能の例には、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、仮想教室教育配信９３、データ分析処理９４、トランザクション処理９５、ならびに仮想マシンに関連付けられたセキュア・ストレージへのアクセス制御９６が含まれる。これらはほんの一例であり、他の実施形態では層が異なるサービスを含むこともできることを理解されたい。

【０１１５】

次に図１９を参照すると、本発明の１つまたは複数の実施形態によるシステム１９００が示されている。システム１９００は、ネットワーク１６５を介するなどして、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している例示的なノード１０（例えば、ホスティング・ノード）を含む。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とすることができる。ノード１０は、１つまたは複数のＶＭ１５（１５Ａ～１５Ｎ）の展開を容易化するハイパーバイザ１２を実行する。ノード１０は、セキュア・インターフェース制御１１を含むハードウェア／ファームウェア層１３をさらに含む。セキュア・インターフェース制御１１は、ハイパーバイザ１２が仮想マシン１５に１つまたは複数のサービスを提供することを容易化する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む。ハイパーバイザ１２とセキュア・インターフェース制御１１との間の通信、セキュア・インターフェース制御１１と１つまたは複数のＶＭ１５との間の通信、ハイパーバイザ１２と１つまたは複数のＶＭ１５との間の通信、およびセキュア・インターフェース制御１１を介したハイパーバイザ１２からＶＭ１５への通信が存在し得る。セキュアなＶＭ環境を容易化するために、本発明の１つまたは複数の実施形態によるホスティング・ノード１０は、ハイパーバイザ１２と１つまたは複数のＶＭ１５との間の直接の通信を何も含まない。

【０１１６】

例えば、ノード１０は、クライアント・デバイス２０ＡがＶＭ１５Ａ～１５Ｎのうちの１つまたは複数を展開することを容易化することができる。ＶＭ１５Ａ～１５Ｎは、別個のクライアント・デバイス２０Ａ～２０Ｅからのそれぞれの要求に応答して展開されてもよい。例えば、ＶＭ１５Ａは、クライアント・デバイス２０Ａによって展開されてもよく、ＶＭ１５Ｂは、クライアント・デバイス２０Ｂによって展開されてもよく、ＶＭ１５Ｃは、クライアント・デバイス２０Ｃによって展開されてもよい。ノード１０はまた、クライアントが（ＶＭとして実行することなく）物理サーバをプロビジョニングすることを容易化することもできる。本明細書に記載の例は、ノード１０内のリソースをＶＭの一部としてプロビジョニングすることを具現化するが、記載した技術上の解決策を適用して、リソースを物理サーバの一部としてプロビジョニングすることもできる。

【０１１７】

一例において、クライアント・デバイス２０Ａ～２０Ｅは、個人、企業、政府機関、会社内の部門、または他の任意のエンティティなどの同じエンティティに属することができ、ノード１０は、エンティティのプライベート・クラウドとして動作されてもよい。この場合、ノード１０は、エンティティに属するクライアント・デバイス２０Ａ～２０Ｅによ

って展開されるＶＭ１５Ａ～１５Ｎのみをホストする。別の例では、クライアント・デバイス２０Ａ～２０Ｅは、別個のエンティティに属してもよい。例えば、第１のエンティティは、クライアント・デバイス２０Ａを所有してもよく、第２のエンティティは、クライアント・デバイス２０Ｂを所有してもよい。この場合、ノード１０は、異なるエンティティからのＶＭをホストするパブリック・クラウドとして動作されてもよい。例えば、ＶＭ１５Ａ～１５Ｎは、ＶＭ１５ＡがＶＭ１５Ｂへのアクセスを容易化しない、覆い隠す方式（shrouded manner）で展開されてもよい。例えば、ノード１０は、ＩＢＭ ｚ Ｓｙｓｔｅｍｓ（Ｒ）プロセッサ・リソース／システム・マネージャ（ＰＲ／ＳＭ：Resource/Systems Manager）論理パーティション（ＬＰＡＲ：Logical Partition）機能を使用して、ＶＭ１５Ａ～１５Ｎを覆い隠すことができる。ＰＲ／ＳＭ ＬＰＡＲなどのこれらの機能は、パーティション間の分離を実現し、それにより、ノード１０が異なる論理パーティション内の同じ物理ノード１０上の異なるエンティティに対して２つ以上のＶＭ１５Ａ～１５Ｎを展開することを容易化する。この分離を実現するために、特定のハードウェアを備えた信頼できる内部ファームウェアに、ＰＲ／ＳＭ ＬＰＡＲハイパーバイザが実装される。

【０１１８】

クライアント・デバイス２０Ａ～２０Ｅの中のクライアント・デバイス２０Ａ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の展開を要求する他の任意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２０Ａは、ネットワーク１６５を介してハイパーバイザによる受信を求める要求を送信してもよい。ＶＭ１５Ａ～１５Ｎの中のＶＭ１５Ａは、クライアント・デバイス２０Ａ～２０Ｅの中のクライアント・デバイス２０Ａからの要求に応答してハイパーバイザ１２が展開するＶＭイメージである。ハイパーバイザ１２は、ＶＭモニタ（ＶＭＭ：VM monitor）であり、ＶＭモニタは、ＶＭを作成および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とすることができる。ハイパーバイザ１２は、ＶＭ１５Ａ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化する。適切な機能および変更を加えたハイパーバイザ１２は、ＩＢＭ ｚ Ｓｙｓｔｅｍｓ（Ｒ）、ＯｒａｃｌｅのＶＭ Ｓｅｒｖｅｒ、ＣｉｔｒｉｘのＸｅｎＳｅｒｖｅｒ、ＶｍｗａｒｅのＥＳＸ、Ｍｉｃｒｏｓｏｆｔ Ｈｙｐｅｒ－Ｖ ｈｙｐｅｒｖｉｓｏｒ、または他の任意のハイパーバイザとすることができる。ハイパーバイザ１２は、ノード１０上で直接実行するネイティブ・ハイパーバイザ、または別のハイパーバイザ上で実行するホストされたハイパーバイザとすることができる。

【０１１９】

次に図２０を参照すると、本発明の１つまたは複数の実施形態による、本明細書の教示を実施するためのノード１０が示されている。ノード１０は、本明細書に記載した様々な通信技術を利用する任意の数のコンピューティング・デバイスおよびネットワークならびにその組合せを備えるか、または展開するか、あるいはその両方である、電子コンピュータ・フレームワークとすることができる。ノード１０は、異なるサービスに変化する、またはある機能を別の機能とは独立して再構成する能力を有する、容易に拡大縮小可能で拡張可能なモジュール式とすることができる。

【０１２０】

この実施形態において、ノード１０は、プロセッサ２００１を有し、プロセッサ２００１には、１つまたは複数の中央処理装置（ＣＰＵ：central processing unit）２００１ａ、２００１ｂ、２００１ｃなどが含まれ得る。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および他の様々な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）を含んでもよい。ＲＡＭは、プロセッサ２００１によって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。

【０１２１】

図２０のノード１０は、ハード・ディスク２００７を含み、ハード・ディスク２００７は、プロセッサ２００１によって読み取り可能で実行可能な有形記憶媒体の例である。ハード・ディスク２００７は、ソフトウェア２００８およびデータ２００９を記憶する。ソフトウェア２００８は、（図１～図１９を参照して説明し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上で実行するための命令として記憶される。データ２００９は、ソフトウェア２００８の動作をサポートし、ソフトウェア２００８の動作によって使用される、様々なデータ構造で編成された質的変数または量的変数の一連の値を含む。

【０１２２】

図２０のノード１０は、プロセッサ２００１と、システム・メモリ２００３と、ハード・ディスク２００７と、ノード１０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）との間の通信を相互接続およびサポートする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において、１つまたは複数のアダプタは、中間バス・ブリッジを介してシステム・バス２００２に接続される１つまたは複数のＩ／Ｏバスに接続されることが可能であり、１つまたは複数のＩ／Ｏバスは、周辺機器相互接続（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。

【０１２３】

図示のように、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ース・アダプタ２０２０を含む。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を含む。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のグラフィックス・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において、ソフトウェア２００８およびデータ２００９の動作は、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上で実施することができる。例えば、ネットワーク２０５０、サーバ２０５１、およびデータベース２０５２を組み合わせて、ソフトウェア２００８およびデータ２００９の内部反復を、プラットフォーム・アズ・ア・サービス、ソフト・ウェア・アズ・ア・サービス、またはインフラストラクチャ・アズ・ア・サービスあるいはその組合せとして（例えば、分散システムのウェブ・アプリケーションとして）提供することができる。

【０１２４】

本明細書に記載の実施形態は、コンピュータ技術、具体的にはＶＭをホストするコンピュータ・サーバに必然的に根ざしている。さらに、本発明の１つまたは複数の実施形態は、セキュアＶＭに関連付けられたメモリ、レジスタ、および他のそのようなデータへのアクセスがハイパーバイザでさえ禁止されるセキュアＶＭを、ＶＭをホストするコンピュータ・サーバがホストすることを容易化することによって、コンピューティング技術自体、具体的にはＶＭをホストするコンピュータ・サーバの動作の改善を容易化する。さらに、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）、またはその組合せを含むセキュア・インターフェース制御（本明細書では「ＵＶ」とも呼ぶ）を使用してセキュアＶＭおよびハイパーバイザの分離を容易化し、それによ

りコンピュータ・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、コンピュータ・サーバをホストするＶＭの改善に向けた重要なステップを提供する。セキュア・インターフェース制御は、本明細書に記載のように、ＶＭの初期化／終了中にＶＭの状態をセキュアにする際に大幅なオーバーヘッドを追加することなくセキュリティを容易化するための、軽い中間動作を提供する。

【０１２５】

本明細書に開示される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）あるいはその組合せを含むこともできる。なお、説明ごとに、要素の識別子は、異なる図の他の同様の要素に再利用されている。

【０１２６】

本明細書では、本発明の様々な実施形態について、関連する図面を参照して説明している。本発明の範囲から逸脱することなく、本発明の代替の実施形態も考案することができる。以下の説明および図面では、要素間の様々な接続および位置関係（例えば、上、下、隣接など）が記載されている。これらの接続または位置関係あるいはその両方は、特に明記されていない限り、直接的または間接的とすることができ、本発明はこの点に関して限定するよう意図されていない。したがって、エンティティの結合は、直接的または間接的な結合のいずれかを指す場合があり、エンティティ間の位置関係は、直接的または間接的な位置関係である場合がある。さらに、本明細書に記載の様々なタスクおよびプロセス・ステップは、本明細書に詳細に記載されていない追加のステップまたは機能を有するより包括的な手順またはプロセスに組み込むことができる。

【０１２７】

特許請求の範囲および本明細書の解釈のために、以下の定義および略語を使用するものとする。本明細書で使用する「備える（comprises）」、「備えている（comprising）」、「含む（includes）」、「含んでいる（including）」、「有する（has）」、「有している（having）」、「含有する（contains）」、もしくは「含有している（containing）」という用語、またはこれらの他の任意の変形は、非排他的な包含を対象とするよう意図されている。例えば、列挙されている要素を含む組成物、混合物、プロセス、方法、物品、または装置は、必ずしもそれらの要素のみに限定されるわけではなく、明示的に列挙されていないかまたはそのような組成物、混合物、プロセス、方法、物品、または装置に固有の他の要素を含むことができる。

【０１２８】

さらに、本明細書において、「例示的」という用語は、「例、実例、または例示としての役割を果たす」ことを意味するように使用されている。本明細書において「例示的」と記載されているいずれの実施形態または設計も、他の実施形態または設計と比較して、必ずしも好ましい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などを含むと理解され得る。「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などを含むと理解され得る。「接続」という用語は、間接的「接続」および直接的「接続」を含むこともできる。

【０１２９】

「約」、「実質的に」、「おおよそ」という用語およびそれらの変形は、本出願時に利用可能な機器に基づく特定の量の測定に関連した誤差の程度を含むことを意図している。例えば、「約」は、所与の値の±８％もしくは５％または２％の範囲を含むことができる。

【０１３０】

本発明は、任意の可能な技術的詳細の統合レベルでのシステム、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合せとすることができる。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施させるためのコンピュータ可読プログラム命令を有するコンピュータ可読記憶媒体（または複数のコンピュータ可読記憶媒体）を含んでもよい。

【０１３１】
コンピュータ可読記憶媒体は、命令実行デバイスが使用するための命令を保持および記憶することができる有形デバイスとすることができる。コンピュータ可読記憶媒体は、例えば、電子記憶デバイス、磁気記憶デバイス、光学記憶デバイス、電磁気記憶デバイス、半導体記憶デバイス、または上記の任意の適切な組合せとすることができるが、これらに限定されない。コンピュータ可読記憶媒体のより具体的な例の非網羅的なリストには以下のもの、すなわち、ポータブル・コンピュータ・ディスケット、ハード・ディスク、ランダム・アクセス・メモリ（ＲＡＭ）、読み取り専用メモリ（ＲＯＭ）、消去可能プログラマブル読み取り専用メモリ（ＥＰＲＯＭ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）、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）、デジタルバーサタイルディスク（ＤＶＤ）、メモリ・スティック、フロッピ（Ｒ）・ディスク、パンチカードまたは命令が記録された溝内の隆起構造などの機械的に符号化されたデバイス、および上記の任意の適切な組合せが含まれる。本明細書で使用するコンピュータ可読記憶媒体は、電波もしくは他の自由に伝播する電磁波、導波路もしくは他の伝送媒体を介して伝播する電磁波（例えば、光ファイバ・ケーブルを通る光パルス）、または電線を介して送信される電気信号などの、一過性の信号自体であると解釈されるべきではない。

【０１３２】
本明細書に記載のコンピュータ可読プログラム命令は、コンピュータ可読記憶媒体からそれぞれのコンピューティング／処理デバイスに、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、ワイド・エリア・ネットワーク、またはワイヤレス・ネットワークあるいはその組合せを介して外部コンピュータまたは外部記憶デバイスにダウンロードされ得る。ネットワークは、銅伝送ケーブル、光伝送ファイバ、ワイヤレス伝送、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合せを含んでもよい。各コンピューティング／処理デバイスにおけるネットワーク・アダプタ・カードまたはネットワーク・インターフェースは、ネットワークからコンピュータ可読プログラム命令を受信し、そのコンピュータ可読プログラム命令を、それぞれのコンピューティング／処理デバイス内のコンピュータ可読記憶媒体での記憶のために転送する。

【０１３３】
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、インストラクション・セット・アーキテクチャ（ＩＳＡ）命令、機械命令、機械依存命令、マイクロコード、ファームウェア命令、状態設定データ、集積回路用の構成データ、または、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もしくは同様のプログラム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合せで書かれたソース・コードもしくはオブジェクト・コードのいずれかとすることができる。コンピュータ可読プログラム命令は、スタンドアロン・ソフトウェア・パッケージとして全体がユーザのコンピュータ上で、一部がユーザのコンピュータ上で、一部がユーザのコンピュータ上かつ一部がリモート・コンピュータ上で、または全体がコンピュータ上もしくはサーバ上で実行されてもよい。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）もしくはワイド・エリア・ネットワーク（ＷＡＮ）を含む任意のタイプ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もよく、または（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接続されてもよい。いくつかの実施形態では、本発明の態様を実行するために、コンピュータ可読プログラム命令の状態情報を利用して、例えばプログラマブル・ロジック回路、フィールド・プログラマブル・ゲート・アレイ（ＦＰＧＡ）、またはプログラマブル・ロジック・アレイ（ＰＬＡ）を含む電子回路をパーソナライズすることによって、電子回路がコンピュータ可読プログラム命令を実行してもよい。

【０１３４】

本発明の態様は、本発明の実施形態による方法、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ながら本明細書で説明されている。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、および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ブロックの組合せは、コンピュータ可読プログラム命令によって実施され得ることが理解されよう。

【０１３５】

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他のプログラマブル・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で指定された機能／動作を実施するための手段を作り出すように、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マブル・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もよい。また、これらのコンピュータ可読プログラム命令は、命令が記憶されたコンピュータ可読記憶媒体が、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で指定された機能／動作の態様を実施する命令を含む製造品を含むように、コンピュータ可読記憶媒体に記憶され、コンピュータ、プログラマブル・データ処理装置、または他のデバイスあるいはその組合せに対して特定の方式で機能するように指示できるものであってもよい。

【０１３６】

また、コンピュータ可読プログラム命令は、コンピュータ、他のプログラマブル装置、または他のデバイス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で指定された機能／動作を実施するように、コンピュータ実施プロセスを生成するべくコンピュータ、他のプログラマブル・データ処理装置、または他のデバイスにロードされて、コンピュータ、他のプログラマブル装置、または他のデバイス上で一連の動作ステップを実行させるものであってもよい。

【０１３７】

図中のフローチャートおよびブロック図は、本発明の様々な実施形態によるシステム、方法およびコンピュータ・プログラム製品の可能な実装形態のアーキテクチャ、機能、および動作を示す。この点に関連して、フローチャートまたはブロック図の各ブロックは、指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命令を含む、命令のモジュール、セグメント、または一部を表すことがある。いくつかの代替の実装形態では、ブロックに記載された機能は、図に記載された順序とは異なる順序で行われてもよい。例えば、連続して示されている２つのブロックは、実際には、関与する機能に応じて、実質的に同時に実行されてもよく、またはそれらのブロックは、場合によっては逆の順序で実行されてもよい。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、および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ブロックの組合せは、指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との組合せを遂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ことにも留意されたい。

【０１３８】

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、限定することを意図するものではない。本明細書で使用される場合、単数形「ある（ａ）」、「１つの（ａｎ）」および「その（ｔｈｅ）」は、文脈上特に明記されていない限り、複数形も含むことを意図している。「備える」または「備えている」あるいはその両方の用語は、本明細書で使用される場合、記載された特徴、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合せの存在を示すが、１つまたは複数の他の機能、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、あるいはその組合せの存在または追加を除外するものではないことが、さらに理解されよう。

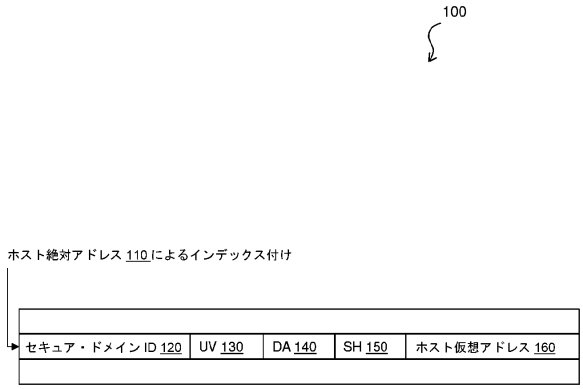
【０１３９】

本明細書における様々な実施形態の説明を例示の目的で提示してきたが、網羅的であることも、開示された実施形態に限定されることも意図されていない。当業者には、説明し

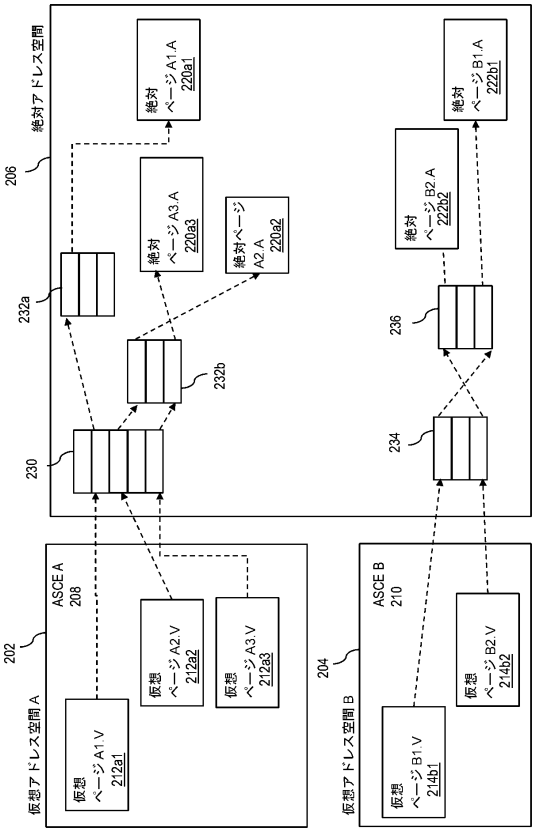
た実施形態の範囲および思想から逸脱することなく多くの変更形態および変形形態が明らかであろう。本明細書で使用される用語は、実施形態の原理、実際の適用例、もしくは市場で見られる技術を超える技術的な改良を最も良く説明するように、または本明細書で開示される実施形態を当業者が理解することが可能になるように選択されたものである。

【図面】

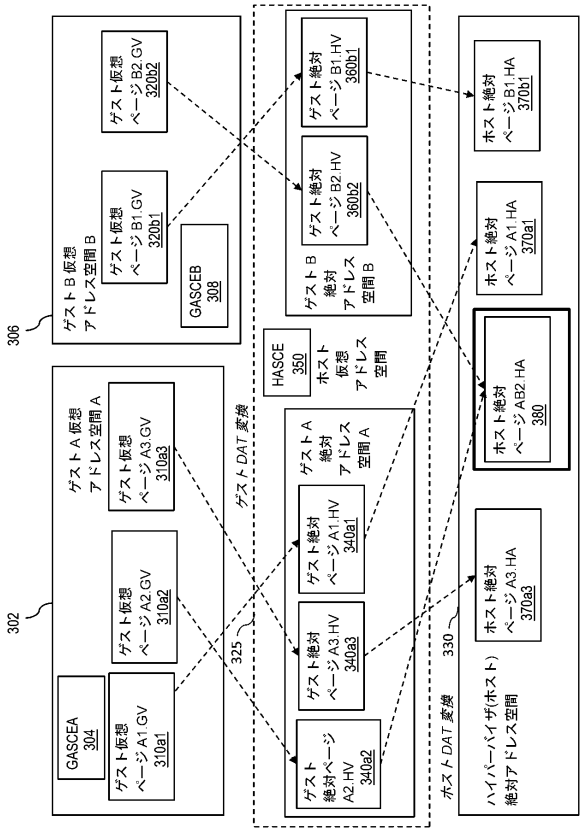
【図１】



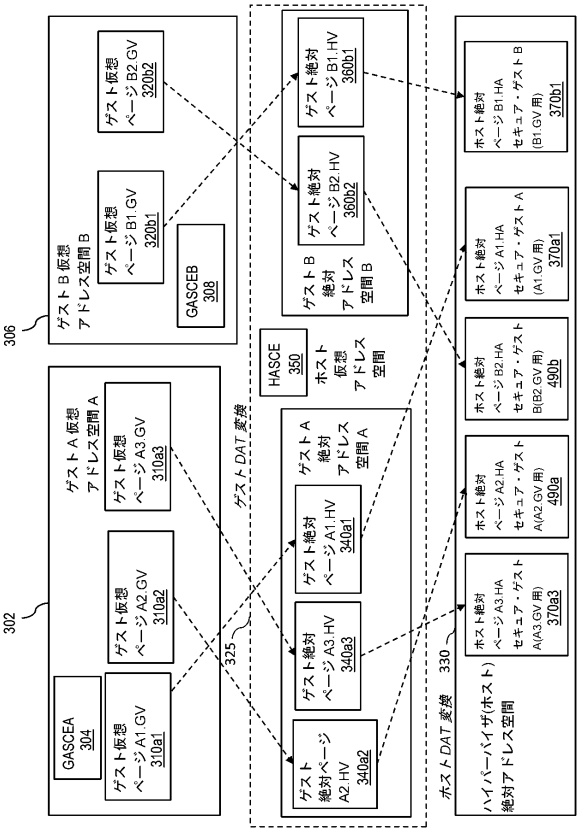
【図２】



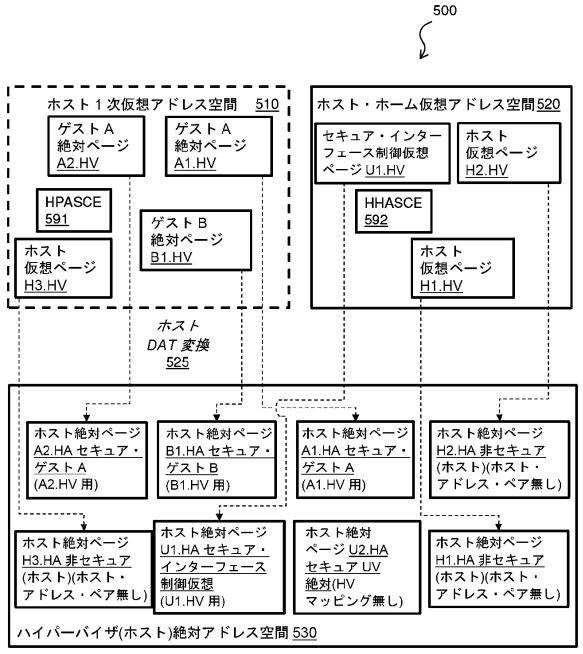
【図３】


306
ゲスト B 仮想
アドレス空間 B
ゲスト仮想
ページ B2.GV
320b2
ゲスト仮想
ページ B1.GV
320b1
GASCEB
308
302
ゲスト A 仮想
アドレス空間 A
ゲスト仮想
ページ A3.GV
310a3
ゲスト仮想
ページ A2.GV
310a2
ゲスト仮想
ページ A1.GV
310a1
GASCEA
304
ゲスト DAT 変換
325
HASCE
350
ホスト
仮想
アドレス
空間
ゲスト B
絶対
アドレス
空間 B
ゲスト絶対
ページ B1.HV
360b1
ゲスト絶対
ページ B2.HV
360b2
ゲスト A
絶対
アドレス
空間 A
ゲスト絶対
ページ A1.HV
340a1
ゲスト絶対
ページ A3.HV
340a3
ゲスト
絶対ページ
A2.HV
340a2
ホスト DAT 変換
330
ハイパーバイザ(ホスト)
絶対アドレス空間
ホスト絶対
ページ B1.HA
370b1
ホスト絶対
ページ A1.HA
370a1
ホスト絶対
ページ AB2.HA
380
ホスト絶対
ページ A3.HA
370a3

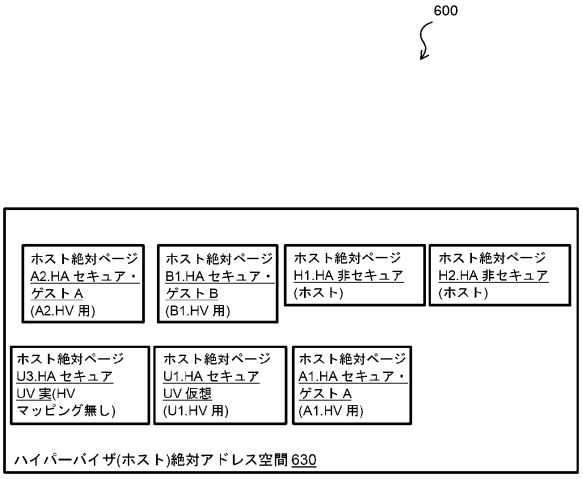
【図４】


306
ゲスト B 仮想
アドレス空間 B
ゲスト仮想
ページ B2.GV
320b2
ゲスト仮想
ページ B1.GV
320b1
GASCEB
308
302
ゲスト A 仮想
アドレス空間 A
ゲスト仮想
ページ A3.GV
310a3
ゲスト仮想
ページ A2.GV
310a2
ゲスト仮想
ページ A1.GV
310a1
GASCEA
304
ゲスト DAT 変換
325
HASCE
350
ホスト
仮想
アドレス
空間
ゲスト B
絶対
アドレス
空間 B
ゲスト絶対
ページ B1.HV
360b1
ゲスト絶対
ページ B2.HV
360b2
ゲスト A
絶対
アドレス
空間 A
ゲスト絶対
ページ A1.HV
340a1
ゲスト絶対
ページ A3.HV
340a3
ゲスト
絶対ページ
A2.HV
340a2
ホスト DAT 変換
330
ハイパーバイザ(ホスト)
絶対アドレス空間
ホスト絶対
ページ B1.HA
セキュア・ゲスト B
(B1.GV 用)
370b1
ホスト絶対
ページ A1.HA
セキュア・ゲスト A
(A1.GV 用)
370a1
ホスト絶対
ページ B2.HA
セキュア・ゲスト
B(B2.GV 用)
490b
ホスト絶対
ページ A2.HA
セキュア・ゲスト
A(A2.GV 用)
490a
ホスト絶対
ページ A3.HA
セキュア・ゲスト
A(A3.GV 用)
370a3

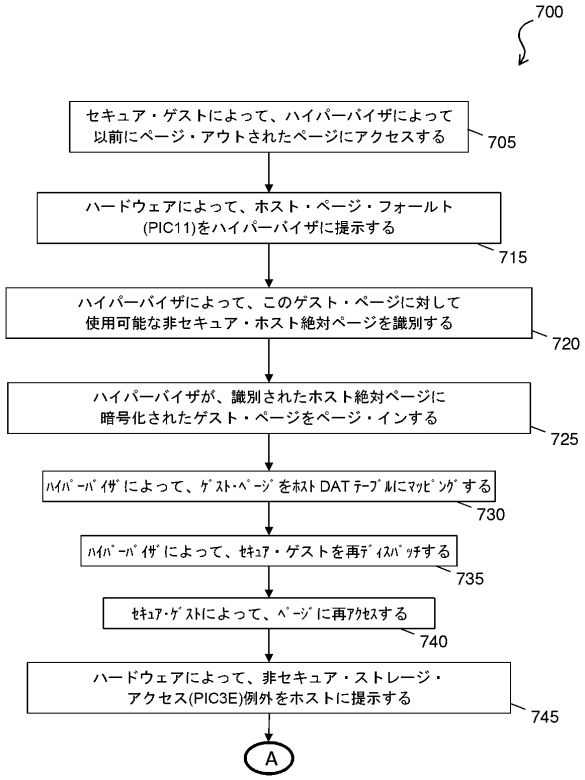
【図５】


500
ホスト 1 次仮想アドレス空間 510
ゲスト A
絶対ページ
A2.HV
ゲスト A
絶対ページ
A1.HV
HPASCE
591
ゲスト B
絶対ページ
B1.HV
ホスト
仮想ページ
H3.HV
ホスト・ホーム仮想アドレス空間 520
セキュア・インター
フェース制御仮想
ページ U1.HV
ホスト
仮想ページ
H2.HV
HHASCE
592
ホスト
仮想ページ
H1.HV
ホスト
DAT 変換
525
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
A1.HA セキュア・
ゲスト A
(A1.HV 用)
ホスト絶対ページ
H2.HA 非セキュア
(ホスト)(ホスト・
アドレス・ペア無し)
ホスト絶対ページ
H3.HA 非セキュア
(ホスト)(ホスト・
アドレス・ペア無し)
ホスト絶対ページ
U1.HA セキュア・
インターフェース
制御仮想
(U1.HV 用)
ホスト絶対
ページ U2.HA
セキュア UV
絶対(HV
マッピング無し)
ホスト絶対ページ
H1.HA 非セキュア
(ホスト)(ホスト・
アドレス・ペア無し)
ハイパーバイザ(ホスト)絶対アドレス空間 530

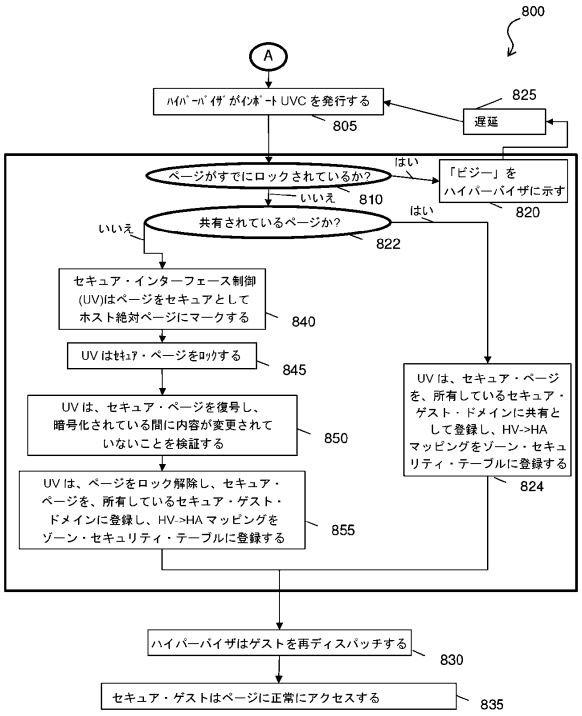
【図６】


600
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 非セキュア
(ホスト)
ホスト絶対ページ
H2.HA 非セキュア
(ホスト)
ホスト絶対ページ
U3.HA セキュア
UV 実(HV
マッピング無し)
ホスト絶対ページ
U1.HA セキュア
UV 仮想
(U1.HV 用)
ホスト絶対ページ
A1.HA セキュア・
ゲスト A
(A1.HV 用)
ハイパーバイザ(ホスト)絶対アドレス空間 630

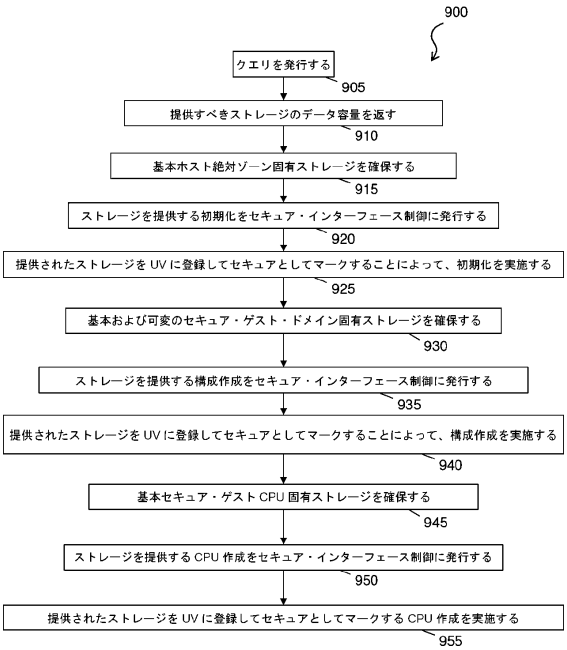
【図７】


700
セキュア・ゲストによって、ハイパーバイザによって以前にページ・アウトされたページにアクセスする
705
ハードウェアによって、ホスト・ページ・フォールト(PIC11)をハイパーバイザに提示する
715
ハイパーバイザによって、このゲスト・ページに対して使用可能な非セキュア・ホスト絶対ページを識別する
720
ハイパーバイザが、識別されたホスト絶対ページに暗号化されたゲスト・ページをページ・インする
725
ﾊｲﾊﾟｰﾊﾞｲｻﾞによって、ｹﾞｽﾄ･ﾍﾟｰｼﾞをﾎｽﾄDATﾃｰﾌﾞﾙにﾏｯﾋﾟﾝｸﾞする
730
ﾊｲﾊﾟｰﾊﾞｲｻﾞによって、ｾｷｭｱ・ゲストを再ﾃﾞｨｽﾊﾟｯﾁする
735
ｾｷｭｱ･ｹﾞｽﾄによって、ﾍﾟｰｼﾞに再ｱｸｾｽする
740
ハードウェアによって、非セキュア・ストレージ・アクセス(PIC3E)例外をホストに提示する
745
A

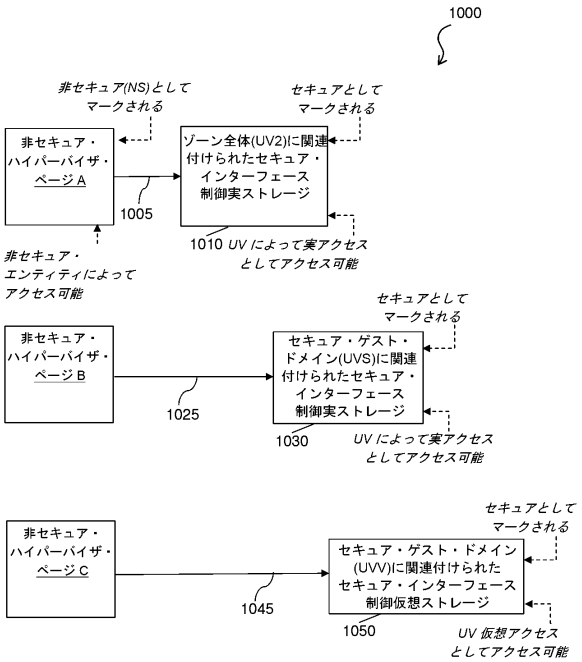
【図８】


800
A
ﾊｲﾊﾟｰﾊﾞｲｻﾞがｲﾝﾎﾞｰﾄUVCを発行する
805
825
遅延
ページがすでにロックされているか?
810
はい
「ビジー」をハイパーバイザに示す
820
いいえ
共有されているページか?
822
はい
いいえ
セキュア・インターフェース制御(UV)はページをセキュアとしてホスト絶対ページにマークする
840
UV はｾｷｭｱ・ページをﾛｯｸする
845
UV は、セキュア・ページを復号し、暗号化されている間に内容が変更されていないことを検証する
850
UV は、ページをロック解除し、セキュア・ページを、所有しているセキュア・ゲスト・ドメインに登録し、HV->HA マッピングをゾーン・セキュリティ・テーブルに登録する
855
UV は、セキュア・ページを、所有しているセキュア・ゲスト・ドメインに共有として登録し、HV->HA マッピングをゾーン・セキュリティ・テーブルに登録する
824
ハイパーバイザはゲストを再ディスパッチする
830
セキュア・ゲストはページに正常にアクセスする
835

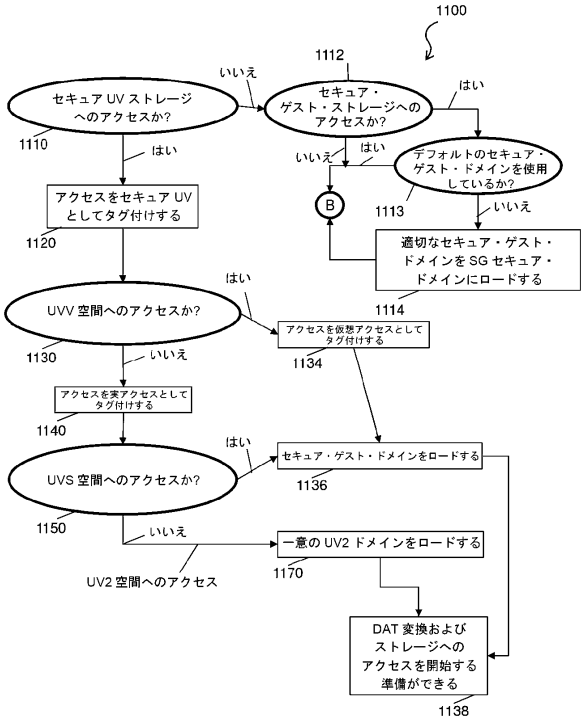
【図９】


900
クエリを発行する
905
提供すべきストレージのデータ容量を返す
910
基本ホスト絶対ゾーン固有ストレージを確保する
915
ストレージを提供する初期化をセキュア・インターフェース制御に発行する
920
提供されたストレージを UV に登録してセキュアとしてマークすることによって、初期化を実施する
925
基本および可変のセキュア・ゲスト・ドメイン固有ストレージを確保する
930
ストレージを提供する構成作成をセキュア・インターフェース制御に発行する
935
提供されたストレージを UV に登録してセキュアとしてマークすることによって、構成作成を実施する
940
基本セキュア・ゲスト CPU 固有ストレージを確保する
945
ストレージを提供する CPU 作成をセキュア・インターフェース制御に発行する
950
提供されたストレージを UV に登録してセキュアとしてマークする CPU 作成を実施する
955

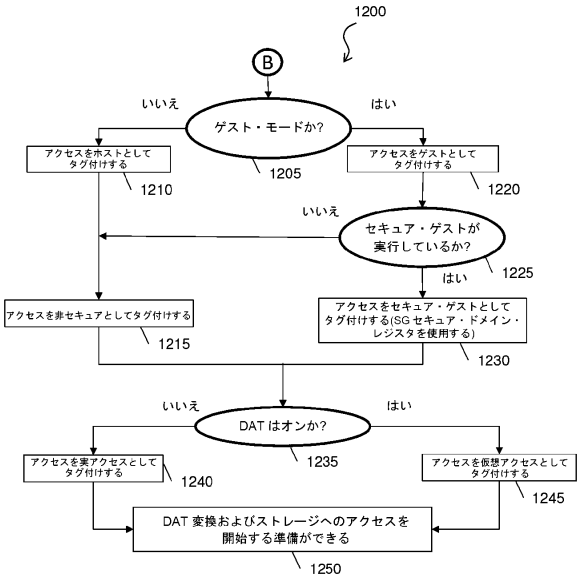
【図１０】


1000
非セキュア(NS)としてマークされる
セキュアとしてマークされる
非セキュア・ハイパーバイザ・ページ A
1005
ゾーン全体(UV2)に関連付けられたセキュア・インターフェース制御実ストレージ
1010 UV によって実アクセスとしてアクセス可能
非セキュア・エンティティによってアクセス可能
セキュアとしてマークされる
非セキュア・ハイパーバイザ・ページ B
1025
セキュア・ゲスト・ドメイン(UVS)に関連付けられたセキュア・インターフェース制御実ストレージ
1030
UV によって実アクセスとしてアクセス可能
セキュアとしてマークされる
非セキュア・ハイパーバイザ・ページ C
1045
セキュア・ゲスト・ドメイン(UVV)に関連付けられたセキュア・インターフェース制御仮想ストレージ
1050
UV 仮想アクセスとしてアクセス可能

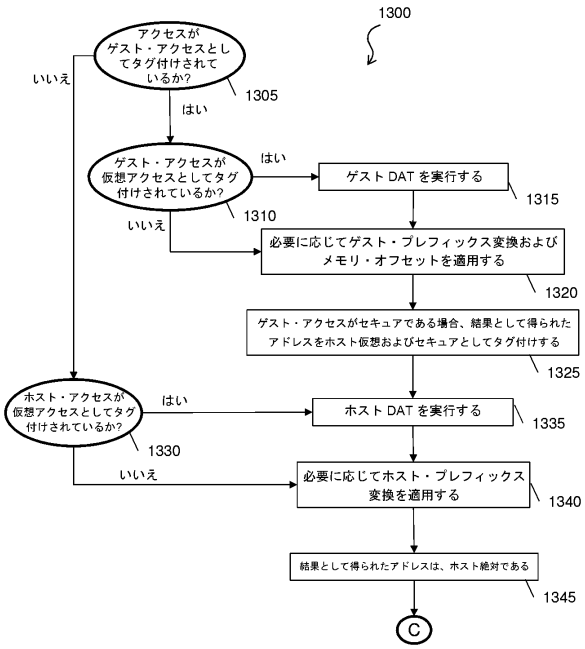
【図１１】



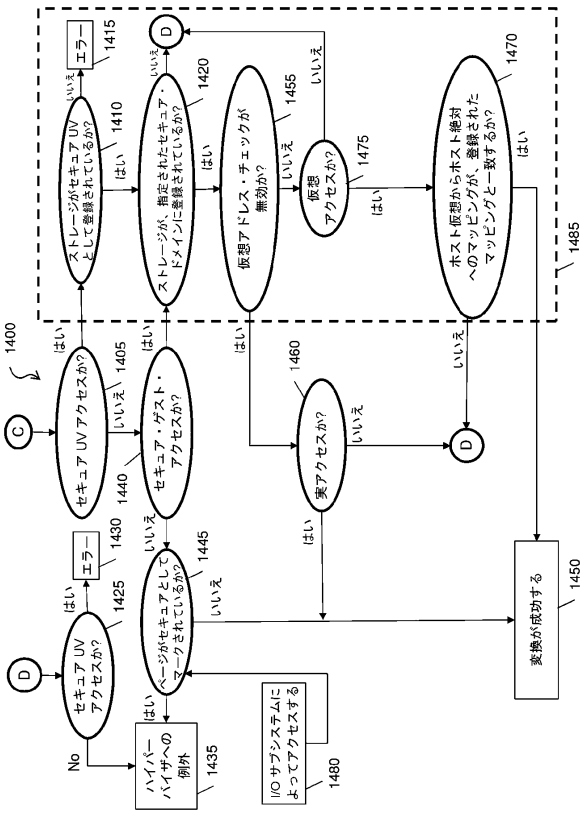
【図１２】



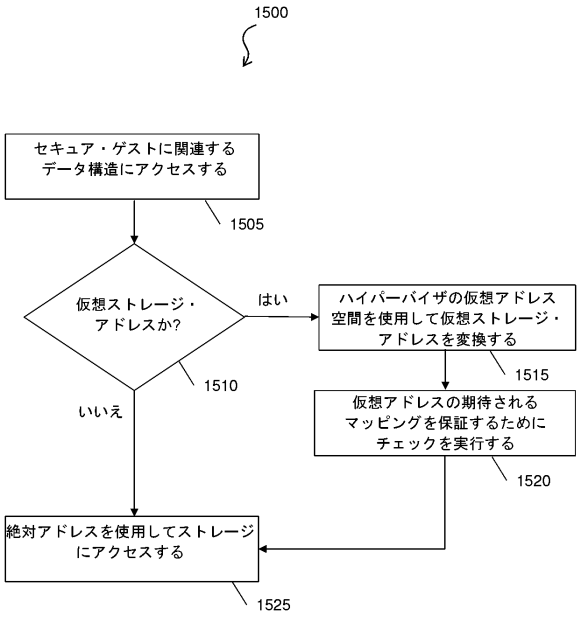
【図１３】



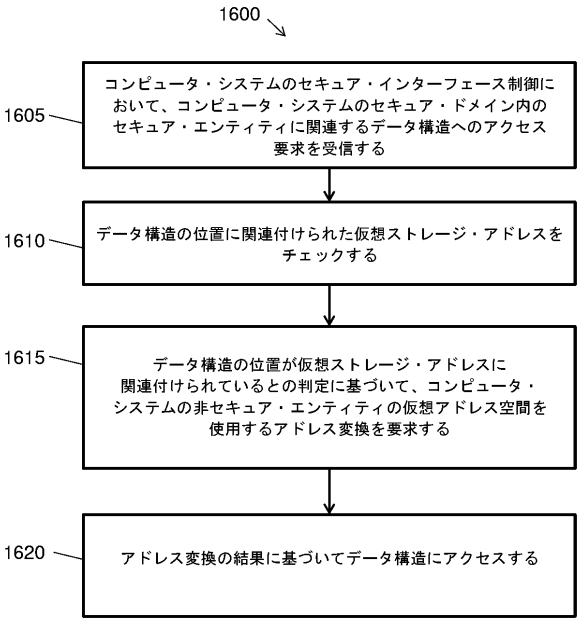
【図１４】



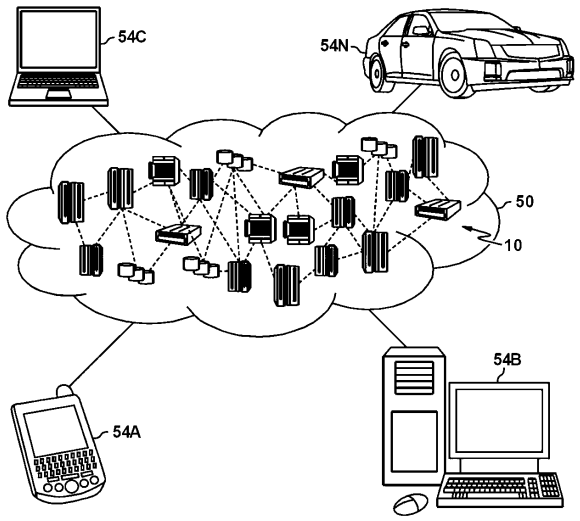
【図１５】



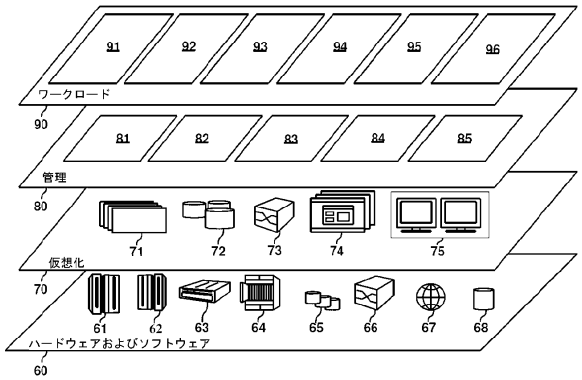
【図１６】



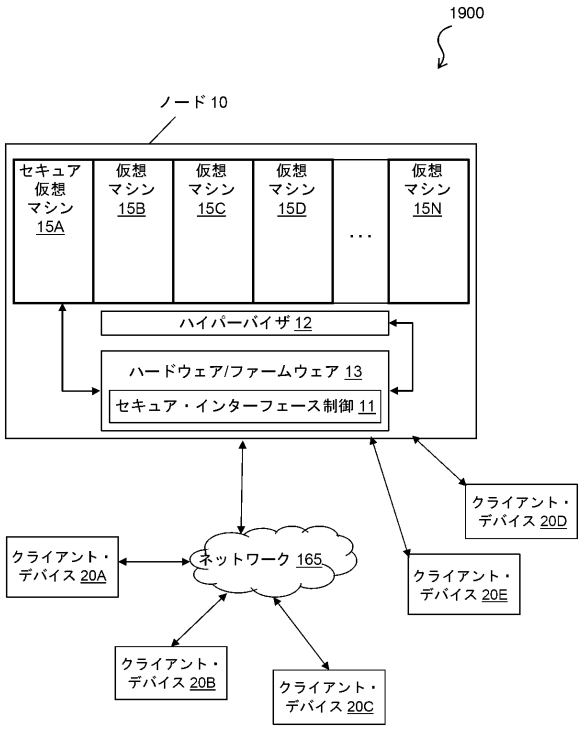
【図１７】



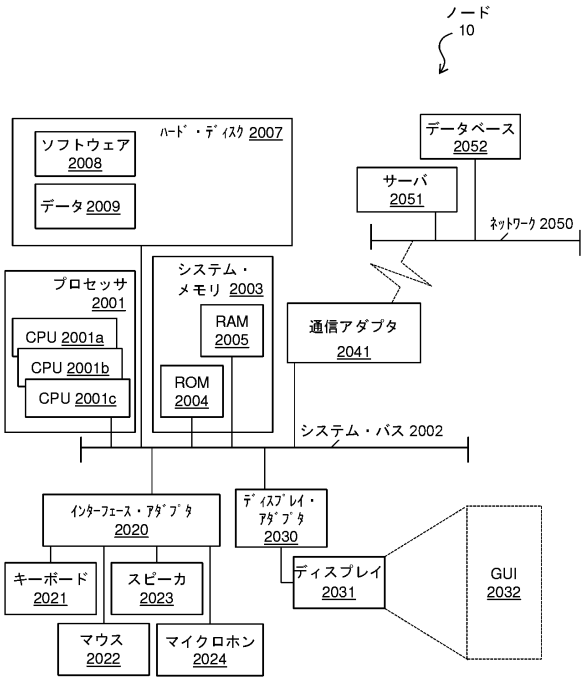
【図１８】



【図１９】


1900
ノード 10
セキュア仮想マシン 15A
仮想マシン 15B
仮想マシン 15C
仮想マシン 15D
...
仮想マシン 15N
ハイパーバイザ 12
ハードウェア/ファームウェア 13
セキュア・インターフェース制御 11
クライアント・デバイス 20A
ネットワーク 165
クライアント・デバイス 20B
クライアント・デバイス 20C
クライアント・デバイス 20D
クライアント・デバイス 20E

【図２０】


ノード 10
ソフトウェア 2008
データ 2009
ﾊｰﾄﾞ・ﾃﾞｨｽｸ 2007
データベース 2052
サーバ 2051
ﾈｯﾄﾜｰｸ 2050
プロセッサ 2001
CPU 2001a
CPU 2001b
CPU 2001c
システム・メモリ 2003
RAM 2005
ROM 2004
通信アダプタ 2041
システム・バス 2002
ｲﾝﾀｰﾌｪｰｽ・ｱﾀﾞﾌﾟﾀ 2020
ﾃﾞｨｽﾌﾟﾚｲ・ｱﾀﾞﾌﾟﾀ 2030
キーボード 2021
スピーカ 2023
ディスプレイ 2031
GUI 2032
マウス 2022
マイクロホン 2024

フロントページの続き

(72)発明者　インブレンダ、クラウディオ
ドイツ　７１０３２　ベーブリンゲン　シェーナイヒャー・シュトラーセ　２２０
(72)発明者　ボーントレーガー、クリスチャン
ドイツ　７１０３２　ベーブリンゲン　シェーナイヒャー・シュトラーセ　２２０
(72)発明者　ヘラー、リサ
アメリカ合衆国１２６０１　ニューヨーク州ポキプシー　サウス・ロード２４５５
(72)発明者　ブサバ、ファディ
アメリカ合衆国１２６０１　ニューヨーク州ポキプシー　サウス・ロード２４５５
(72)発明者　ブラッドベリー、ジョナサン
アメリカ合衆国１２６０１　ニューヨーク州ポキプシー　サウス・ロード２４５５

審査官　平井　誠

(56)参考文献　米国特許出願公開第２０１９／００４２４６３（ＵＳ，Ａ１）
特表２０１６－５３６７２０（ＪＰ，Ａ）
SEONGWOOK JIN; ET AL，ARCHITECTURAL SUPPORT FOR SECURE VIRTUALIZATION UNDER A VULNERABLE HYPERVISOR，PROCEEDINGS OF THE 44TH ANNUAL IEEE/ACM INTERNATIONAL SYMPOSIUM ON MICROARCHITECTURE，米国，2011年，PAGE(S):272-283，http://dx.doi.org/10.1145/2155620.2155652
SEONGWOOK JIN; ET AL，H-SVM: HARDWARE-ASSISTED SECURE VIRTUAL MACHINES UNDER A VULNERABLE HYPERVISOR，IEEE TRANSACTIONS ON COMPUTERS，米国，IEEE，2015年10月，VOL:64, NR:10，PAGE(S):2833-2846，http://dx.doi.org/10.1109/TC.2015.2389792

(58)調査した分野　(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　１２／１４
Ｇ０６Ｆ　２１／００－８８