(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4636366号
(P4636366)
(45)発行日　平成23年2月23日(2011.2.23)　(24)登録日　平成22年12月3日(2010.12.3)

(51)Int.Cl.　　F I
*G06F 21/20 (2006.01)*　　G06F 15/00 330D
*G09C 1/00 (2006.01)*　　G09C 1/00 660D

請求項の数 6　(全 19 頁)

(21)出願番号　特願2004-537309 (P2004-537309)
(86)(22)出願日　平成15年9月19日 (2003.9.19)
(65)公表番号　特表2006-517690 (P2006-517690A)
(43)公表日　平成18年7月27日 (2006.7.27)
(86)国際出願番号　PCT/GB2003/004063
(87)国際公開番号　WO2004/027612
(87)国際公開日　平成16年4月1日 (2004.4.1)
審査請求日　平成18年7月11日 (2006.7.11)
(31)優先権主張番号　10/246,909
(32)優先日　平成14年9月19日 (2002.9.19)
(33)優先権主張国　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　ＭＡＳＣＨＩＮＥＳ　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
(74)代理人　100108501
弁理士　上野　剛史
(74)代理人　100112690
弁理士　太佐　種一
(74)代理人　100091568
弁理士　市位　嘉宏

最終頁に続く

(54)【発明の名称】分散コンピューティング・ドメインのためのアプリケーション・サーバのオブジェクト・レベル・セキュリティ

(57)【特許請求の範囲】
【請求項１】
管理オブジェクト及びユーザ・オブジェクトを含む１つ又は複数のアプリケーション・サーバにおいて、オブジェクト単位でセキュリティ機能を実行する方法であって、
管理オブジェクトのアプリケーション・サーバ・レベル・セキュリティのために定義され、管理オブジェクトへのインターフェースに関連付けられたアプリケーション・サーバ・セキュリティ・フラグを記憶手段が記憶するステップと、
管理オブジェクトのドメイン・レベル・セキュリティのために定義されたグローバル・セキュリティ・フラグ及び前記アプリケーション・サーバ・セキュリティ・フラグが有効である場合に、ユーザ・オブジェクト及び管理オブジェクトをセキュアな通信で保護する第１モードと、前記グローバル・セキュリティ・フラグが有効であり、かつ前記アプリケーション・サーバ・セキュリティ・フラグが無効である場合に、ユーザ・オブジェクトを非セキュアな通信で用いるが、管理オブジェクトをセキュアな通信で保護する第２モードと、前記グローバル・セキュリティ・フラグが無効である場合に、ユーザ・オブジェクト及び管理オブジェクトを非セキュアな通信で用いる第３モードとを含むモードのうちの１つに従って、アプリケーション・サーバが、管理オブジェクト及びユーザ・オブジェクトごとに別個のセキュアな通信または非セキュアな通信をクライアント・プロセスと実行するステップと
を含む、前記方法。
【請求項２】

アプリケーション・サーバ・セキュリティ・フラグの管理オブジェクト・インターフェースへの関連付けが、アプリケーション・サーバのユーザ・オブジェクト・セキュリティが有効である場合に、アプリケーション・サーバ上のユーザ・オブジェクトについてタグ付きコンポーネントを備えたＣＯＲＢＡ　相互運用オブジェクト参照（ＩＯＲ）をエクスポートするアクション、及びアプリケーション・サーバのユーザ・オブジェクト・セキュリティが有効である場合に、アプリケーション・サーバ上のユーザ・オブジェクトについてＵＤＤＩレジストリにオブジェクト・タイプを提供するアクションを含む群から選択されるアクションを実行することによって行われる、請求項１に記載の方法。

【請求項３】

前記セキュアな通信を実行することが、認証、許可、及びトランスポート保護から選択されたセキュリティ・オペレーションを実行することを含む、請求項１に記載の方法。

【請求項４】

前記セキュアな通信を実行することが、シンプル・オブジェクト・アクセス・プロトコル（ＳＯＡＰ）及びオブジェクト・リクエスト・ブローカ（ＯＲＢ）間プロトコルを含む群から選択されたセキュリティ・プロトコルを使用することによって行われる、請求項１に記載の方法。

【請求項５】

管理オブジェクト及びユーザ・オブジェクトを含む１つ又は複数のアプリケーション・サーバにおいて、オブジェクト単位でセキュリティ機能を実行するためのコンピュータ・プログラムであって、コンピュータに請求項１に記載の方法の各ステップを実行させる前記コンピュータ・プログラム。

【請求項６】

管理オブジェクト及びユーザ・オブジェクトを含む１つ又は複数のアプリケーション・サーバにおいて、オブジェクト単位でセキュリティ機能を実行するオブジェクト・レベル・セキュリティ・システムであって、

管理オブジェクトのアプリケーション・サーバ・レベル・セキュリティのために定義され、管理オブジェクトへのインターフェースに関連付けられたアプリケーション・サーバ・セキュリティ・フラグを記憶する手段と、

前記記憶する手段に記憶されたアプリケーション・サーバ・セキュリティ・フラグを読み取り、管理オブジェクトのドメイン・レベル・セキュリティのために定義されたグローバル・セキュリティ・フラグ及び前記アプリケーション・サーバ・セキュリティ・フラグが有効である場合に、ユーザ・オブジェクト及び管理オブジェクトをセキュアな通信で保護する第１モードと、前記グローバル・セキュリティ・フラグが有効であり、かつ前記アプリケーション・サーバ・セキュリティ・フラグが無効である場合に、ユーザ・オブジェクトを非セキュアな通信で用いるが、管理オブジェクトをセキュアな通信で保護する第２モードと、前記グローバル・セキュリティ・フラグが無効である場合に、ユーザ・オブジェクト及び管理オブジェクトを非セキュアな通信で用いる第３モードとを含むモードのうちの１つに従って、アプリケーション・サーバに、管理オブジェクト及びユーザ・オブジェクトごとに別個のセキュアな通信または非セキュアな通信をクライアント・プロセスと実行させる手段と

を備えている、前記システム。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は、アプリケーション・サーバ・セキュリティの管理システム及びツールの分野に関する。

【背景技術】

【０００２】

アプリケーション・サーバは、イントラネット又はインターネットなどの分散型ネットワークにおいて、コンピュータ又はコンピュータ・プラットフォームによって実行される

サーバ・プログラムであり、それは、アプリケーション又はサービスのためのビジネス・ロジックを提供する。Ｉｎｔｅｒｎａｔｉｏｎａｌ Ｂｕｓｉｎｅｓｓ Ｍａｃｈｉｎｅｓ（ＩＢＭ（登録商標））、Ｍｉｃｒｏｓｏｆｔ（登録商標） Ｃｏｒｐｏｒａｔｉｏｎ、及びＳｕｎ Ｍｉｃｒｏｓｙｓｔｅｍｓなどの企業は、多種多様なハードウェア・プラットフォーム上のアプリケーション・サーバを開発し、管理し、配置し、かつ実行するためのソフトウェア・スイートを提供しており、そのハードウェア・プラットフォームには、これらに限定されるものではないが、パーソナル・コンピュータ、ネットワーク・サーバ、メインフレーム、及びワークステーションが含まれ、これらは、以下に限定されるものではないが、ＵＮＩＸ（登録商標）、Ｌｉｎｕｘ、ＩＢＭのＡＩＸ（登録商標）、ＯＳ／２（登録商標）、及びＯＳ／３９０（登録商標）、ＭｉｃｒｏｓｏｆｔのＷｉｎｄｏｗｓ（登録商標）、並びにＳｕｎのＳｏｌａｒｉｓを含む多種多様なオペレーティング・システムを対象にする。ＩＢＭは、ＷｅｂＳｐｈｅｒｅ（登録商標）として知られた周知かつ広く用いられているアプリケーション・サーバ・スイートを提供する。（ＩＢＭ、ＡＩＸ、ＯＳ／２、及びＯＳ／３９０は、Ｉｎｔｅｒｎａｔｉｏｎａｌ Ｂｕｓｉｎｅｓｓ Ｍａｃｈｉｎｅｓ Ｃｏｒｐｏｒａｔｉｏｎの登録商標であり、Ｍｉｃｒｏｓｏｆｔ及びＷｉｎｄｏｗｓは、米国、その他の国々、又はその両方におけるＭｉｃｒｏｓｏｆｔ Ｃｏｒｐｏｒａｔｉｏｎの商標であり、ＵＮＩＸは、米国及びその他の国々におけるＴｈｅ Ｏｐｅｎ Ｇｒｏｕｐの登録商標である。）

【０００３】

アプリケーション・サーバは、「Ｃ」などの高水準言語（「ＨＬＬ」）、Ｓｕｎ ＭｉｃｒｏｓｙｓｔｅｍｓのＪａｖａ（商標）などの移植可能言語、及びそれらの組み合わせで開発することができる。アプリケーション・ロジックの一部は、アプリケーション・サーバ・プログラムの開発に用いられる設計手法及び言語に応じて、Ｅｎｔｅｒｐｒｉｓｅ Ｊａｖａ Ｂｅａｎｓ（「ＥＪＢ」）、ダイナミック・リンク・ライブラリ（「ＤＬＬ」）、プログラム・オブジェクト、アプレット、及びサーブレットにおいて、具体化することができる。（ＪａｖａとＪａｖａベースの商標及びロゴのすべては、米国、その他の国々、又はその両方におけるＳｕｎ Ｍｉｃｒｏｓｙｓｔｅｍｓ， Ｉｎｃ．の商標である。）

【０００４】

アプリケーション・サーバは、一般に、必要とされる機能又はサービスを実行するために、Ｗｅｂブラウザ及び遠隔端末装置などのクライアント・コンピュータ及びプロセスと協同し、ユーザとのインターフェースをとる。クライアント・コンピュータ及びプロセスと相互作用するための共通プロトコルは、以下に限定されるものではないが、ハイパーテキスト転送プロトコル（「ＨＴＴＰ」）、伝送制御プロトコル／インターネット・プロトコル（「ＴＣＰ／ＩＰ」）、セキュア・ソケット・レイヤー（「ＳＳＬ」）、及びファイル転送プロトコル（「ＦＴＰ」）を含む。アプリケーション・サーバと協同するためのクライアント・プロセスは、以下に限定されるものではないが、スタンド・アローン（例えばブラウザ独立）のクライアント・プログラム、ハイパーテキスト・マークアップ言語（「ＨＴＭＬ」）リソース、ブラウザ拡張及びプラグイン、拡張マークアップ言語（「ＸＭＬ」）リソース、（例えばＴＩＦＦ、ＧＩＦ、ＪＰＥＧ、ＡＶＩなどの）グラフィックス及びマルチメディア・オブジェクト、Ｊａｖａサーバ・ページ（「ＪＳＰ」）、アクティブ・サーバ・ページ（「ＡＳＰ」）、個人ホームページ（「ＰＨＰ」）、Ｊａｖａモジュール、並びにＪａｖａ Ｂｅａｎｓを含む。

【０００５】

アプリケーション・サーバはまた、ファイル・システム及びデータベースなどの他のシステム・リソースと相互作用し、データの読み取り、格納、作成、及びコピーを行うことができる。これらのシステム・リソースは、ローカル・キャッシュなどのアプリケーション・サーバよって直接管理することができ、又は他のサーバを介してアクセスし、かつそれによって管理することもできる。

【０００６】

アプリケーション・サーバはまた、多くの場合にセキュリティ許可・認証サーバと協同して、ユーザが本人であると主張しているユーザであるかどうか、及び要求されたアクションの実行、必要とされるリソース又はデータへのアクセスなどの許可又は特権を、ユーザが有しているかどうかを判断する。

【０００７】

アプリケーション・サーバとそうした許可及び認証プロセスとの間の通信のみならず、他のシステム・リソースとの間の通信用プロトコルは、クライアント通信のために上でリストしたプロトコルのいずれをも含むことができ、またプロプライエタリ仕様によって定義された他のプロトコルのみならず、Ｊａｖａ Ｖｅｒｓｉｏｎ ２ Ｅｎｔｅｒｐｒｉｓｅ Ｅｄｉｔｉｏｎ（「Ｊ２ＥＥ」）のインターフェース及び通信のモデルや、Ｃｏｍｍｏｎ Ｂｒｏｋｅｒ Ｒｅｑｕｅｓｔ Ａｒｃｈｉｔｅｃｔｕｒｅ（「ＣＯＲＢＡ」）などの「オープン」スタンダードを含むことが、間々ある。

【０００８】

業界において、ＩＢＭのＳＯＭ及びＤＳＯＭアーキテクチャと、Ｎｅｔｓｃａｐｅ Ｃｏｒｐｏｒａｔｉｏｎのオープン・ネットワーク環境（「ＯＮＥ」）と、Ｓｕｎ Ｍｉｃｒｏｓｙｓｔｅｍｓ のＲＭＩと、ＭｉｃｒｏｓｏｆｔのＣＯＭ及びＤＣＯＭアーキテクチャとを含む、ＣＯＲＢＡのいくつかの実装形態が入手可能である。これらのアーキテクチャは、クライアント・アプリケーション・プログラム・オブジェクトが、タスク又はサービスの完了に有用であり得るか又は必要とされ得る他のプログラム・オブジェクトの存在を「発見」することを可能にする。オブジェクト・リクエスト・ブローカ（「ＯＲＢ」）は、そのようなアーキテクチャのコンポーネントであり、それに対してクライアントは、特定機能の要求を送信することができる。ＯＲＢは次に、ＯＲＢが認識しており、かつクライアントによって求められる機能を処理できる適当なサーバに、これらの要求を転送する。クライアント及びサーバは次に、使用する適当なインターフェースを決定するために対話し、それにより、クライアントは、サーバが提供するプログラム・オブジェクトを用いることができる。

【０００９】

ＣＯＲＢＡアーキテクチャにおける特定用途については、相互運用オブジェクト参照（「ＩＯＲ」）の定義がある。ＩＯＲは、オブジェクトが存在するプラットフォームに依存せず、ＯＲＢ実装にも依存しないオブジェクト参照を提供する。

【００１０】

オブジェクトを分散したこれらのタイプのアプリケーションが一層複雑になるにともない、多くのアプリケーション管理システムが市場にもたらされた。アプリケーション管理は、アプリケーションのインストール及び設定から、その有効な製品寿命を通しての当該アプリケーションの制御及びモニタリングを含み、例えばメトリックを収集したり、効率及び応答性を最大化するための調整を行ったりする。アプリケーション管理システムによって採用されたモデルは、マネージャーエージェント・モデルであることが多い。アプリケーションは、管理システムのエージェントからの情報アクション要求に応答し、管理システム・エージェントと通信する管理システムに、特定情報を提供する役割を担う。管理システム・エージェントは、通常、管理されているアプリケーションと同じハードウェア・プラットフォーム上で実行する。

【００１１】

管理システムは、管理されているアプリケーションと自らが接触している１つ又は複数のエージェントと通信する。管理システムは、プロプライエタリ・プロトコルを用いて、そのエージェントと通信することができ、或いはＳｉｍｐｌｅ Ｎｅｔｗｏｒｋ Ｍａｎａｇｅｍｅｎｔ Ｐｒｏｔｏｃｏｌ（「ＳＮＭＰ」）又はＪａｖａ Ｍａｎａｇｅｍｅｎｔ Ｅｘｔｅｎｓｉｏｎｓ（「ＪＭＸ」）などの「オープン」プロトコルを用いることもできる。市販の管理システムには、例えばＴｉｖｏｌｉのＭａｎａｇｅｍｅｎｔ Ｅｎｖｉｒｏｎｍｅｎｔ及びＩＢＭのＷｅｂＳｐｈｅｒｅ管理アプリケーションなどがある。

【００１２】

図３を参照すると、ＪＭＸ　Ｍａｎａｇｅｍｅｎｔ　ＢｅａｎｓすなわちＭＢｅａｎｓを用いる、こうした管理システムの１つの実施形態が示される。ＭＢｅａｎｓは、標準ＪＭＸインターフェースを介してアクセス可能なオブジェクトであり、それは、開発者にアプリケーション固有管理インターフェースを公開する能力を提供する。ＪＭＸ　ＭＢｅａｎサーバ（３３）は、管理アプリケーション（３７）へのアプリケーション・サーバ（３１）のインターフェースを公開する役割を担うアプリケーション・サーバＭＢｅａｎ（３４）を実行する。同様に、ＪＭＸ　ＭＢｅａｎサーバ（３３）はまた、アプリケーション（３２）のためのアプリケーション固有管理インターフェースを管理アプリケーション（３７）に公開するアプリケーションＭＢｅａｎ（３５）を実行する。ＭＢｅａｎサーバ（３３）はまた、１つ又は複数の管理システムをアプリケーションにブリッジするのに必要なアダプタ（３６）を提供し、それにより、アプリケーションと共にアプリケーション・マネージャの管理機能を用いることができる。

【００１３】

ＩＢＭのＷｅｂＳｐｈｅｒｅアプリケーション・サーバ製品の現バージョンにおいては、管理サーバが中央コンタクト・ポイントであり、そこに管理要求のすべてが向けられる。この管理サーバを保護するためには、アプリケーション・サーバをセキュア・クライアントとして構成することだけが必要であったが、アプリケーション・サーバをセキュア・サーバとして構成する必要はなかった。管理サーバ上ではいかなるユーザ・オブジェクトも許可されず、かつアプリケーション・サーバ上ではいかなる管理オブジェクトも許可されなかったため、このアプローチは、しばらくの間、当該技術分野における必要性に合致した。

【発明の開示】

【発明が解決しようとする課題】

【００１４】

しかしながら、アプリケーションが進化して、管理機能を含むようになり、かつアプリケーション・サーバ上で管理オブジェクトを用いるようになったことにともない、（サーバ・レベル・セキュリティに代わる）オブジェクト・レベル・セキュリティが求められるが、既存のアーキテクチャはそれをサポートしない。いくつかのシステムにおいては、ユーザ・リソースを保護しなければならないかどうかには関係なく、アプリケーション・プロセスのすべては、保護しなければならない管理リソースを含む。サーバ・レベル・セキュリティ・スキームである現在のセキュリティ・スキームでは、ユーザ・オブジェクト・セキュリティが無効にされると、管理オブジェクト・セキュリティも無効になる。実際にセキュアである必要のないユーザ・オブジェクト及びリソースにとっては、これは、認証及び許可プロセスをユーザ・オブジェクトに適用する必要があるため、パフォーマンスを不必要にかつ望ましくなく低下させる。システムは、デフォルトで管理リソース・セキュリティを無効にすることを含む、セキュリティを無効にするように構成されると、管理機能を介して、不必要にかつ望ましくなくセキュリティ・ブリーチに曝される。

【課題を解決するための手段】

【００１５】

従って、本発明は、分散型コンピュータ・ドメインにおける方法であって、管理オブジェクト及びユーザ・オブジェクトを1つ又は複数のアプリケーション・サーバに分散し、管理オブジェクトのドメイン・レベル・セキュリティのためのグローバル・セキュリティ・フラグを定義し、1つ又は複数のアプリケーション・サーバ・セキュリティ・フラグを前記分散ユーザ・オブジェクトへのインターフェースに関連付け、前記グローバル・セキュリティ・フラグ及び前記関連付けられたアプリケーション・サーバ・セキュリティ・フラグが有効である場合に、ユーザ・オブジェクト及び管理オブジェクトを保護する第１モードと、前記グローバル・セキュリティ・フラグが有効であり、かつ前記関連付けられたアプリケーション・サーバ・セキュリティ・フラグが無効である場合に、セキュリティ・オペレーションなしにユーザ・オブジェクトを用いるが、管理オブジェクトを保護する第２モードと、前記グローバル・セキュリティ・フラグが無効である場合に、セキュリティ

・オペレーションなしにユーザ・オブジェクト及び管理オブジェクトを用いる第３モードとからなる３つのモードのうちの１つにおいて、1つ又は複数のセキュリティ・オペレーションを、アプリケーション・サーバによってクライアント・プロセスと協同して実行することを含む方法を、提供する。

【００１６】

ユーザ・オブジェクト及び管理オブジェクトを含むアプリケーション・サーバ用のオブジェクト・レベル・セキュリティ構成を可能にするシステム及び方法を提供することが好ましい。

【００１７】

セキュリティ・ドメイン内のアプリケーション・サーバごとに、ユーザ・オブジェクト・セキュリティのためのセキュリティ有効化とは別に、セキュリティ管理オブジェクト及び命名オブジェクトのためのセキュリティが、別個に、かつ宣言的に有効及び無効にされることを可能にするプロセス及びメカニズムを提供することが好ましい。これは、好ましくは、特にＩＢＭ　ＷｅｂＳｐｈｅｒｅアプリケーション・サーバ製品に基づくアプリケーション環境内で、同一プロセスの管理セキュリティを維持しつつ、ユーザ・セキュリティを無効にする能力を提供する。

【００１８】

顧客（例えばアプリケーション・サーバ所有者及びオペレータ）は、セキュリティ・ドメイン内のいくつかのアプリケーション・サーバ上のユーザ・オブジェクト・セキュリティを、他のアプリケーション・サーバ上のユーザ・オブジェクトのセキュリティを維持しつつ、かつセキュリティ・ドメイン全体を通して、好ましくは、グローバル管理オブジェクト・セキュリティを有しつつ、選択的に無効にできることが好ましい。したがって、好ましくは、ユーザ・セキュリティとは別個に管理セキュリティ、内部システム管理などの管理機能を管理することができる。

【００１９】

ここに開示したプロセス及び方法を用いて、クライアント・プロセスは、アプリケーション・オブジェクトがユーザ・オブジェクトであるのか管理オブジェクトであるのかを認識しないが、セキュリティ機能を実行するか否かの各オブジェクトの要件だけは認識するのが好ましい。さらに、本発明は、好ましくは、クラス及びパッケージによってどのオブジェクトが保護されることになるかの宣言型定義を可能にする。

【００２０】

一般に、アプリケーション・サーバのオブジェクト・レベル・セキュリティを構成可能にするために、好ましくは、ＯＲＢがＩＯＲを共有オブジェクトのネーム・サーバにエクスポートするＣＯＲＢＡアーキテクチャが採用される。管理オブジェクトなどの保護すべきオブジェクトのためのＩＯＲは、そのサービスでの使用のためのセキュリティ詳細を指示するタグ付きコンポーネントを提供されるのが好ましい。ユーザ・オブジェクトなどの非セキュアなオブジェクトのためのＩＯＲは、タグ付きコンポーネントなしにエクスポートするのが好ましい。所与のオブジェクトをエクスポートするときに、ＩＯＲインターセプタが呼び出されることが好ましい。ＩＯＲインターセプタは、好ましくは、保護する必要のある管理オブジェクト又は管理パッケージのすべてを列挙した記述子ファイルをロードするのが好ましい。これは、好ましくは、グローバル・セキュリティがセキュリティ・ドメイン内で有効にされたときに、単一オブジェクト、個々のオブジェクトのリスト、又はオブジェクトのパッケージの仕様が保護されるようにする。

【００２１】

この記述子ファイルにおけるオブジェクトのために、セキュリティ・タグ付きコンポーネントを有するＩＯＲをエクスポートするのが好ましい。クライアントは、あとで、タグ付きＩＯＲをピックアップしたときに、好ましくは、そのサービスの要求を呼び出すためのセキュリティ要件を認識させられる。いかなるタグもＩＯＲに提供されていなければ、セキュリティ処置なしにサービスを要求できるのが好ましい。こうした非セキュア・オブジェクトについて、クライアントは、アプリケーション・サーバにいかなるユーザ・セキ

ュリティ情報も送信せず、またアプリケーション・サーバへのトランスポートも保護しないのが好ましく、その結果、これらのユーザ・オブジェクトについてパフォーマンスが向上し、ユーザ・オブジェクトを不必要に保護するのを避けることができる。

【００２２】

一般に、１つのドメインにおいて、（ａ）グローバル（ドメイン全体にわたる管理セキュリティ）セキュリティ・フラグ、及び（ｂ）１つ又は複数のアプリケーション・サーバ・セキュリティ・フラグの２つのタイプのフラグを利用し、オブジェクト・レベルのセキュリティを構成することが好ましい。前者は、好ましくは、ドメイン・レベルで維持され、後者は、好ましくは、各アプリケーション・サーバに固有である。グローバル・セキュリティ・フラグは、好ましくは、ドメイン全体を通して管理オブジェクトのすべてが保護されているか否かを判断する。これらの管理オブジェクトはまた、好ましくは、前記の記述子ファイルにリストされる。各アプリケーション・サーバ・セキュリティ・フラグは、好ましくは、それだけが関係するアプリケーション・サーバ上のユーザ・オブジェクトのセキュリティを有効又は無効にするが、そのサーバ上の管理オブジェクトのセキュリティには影響しない。

【００２３】

セキュリティ・オペレーションは、管理オブジェクト及びユーザ・オブジェクトのための別個のセキュリティ・オペレーションを提供することによって、実行されるのが好ましい。

【００２４】

アプリケーション・サーバのユーザ・オブジェクト・セキュリティが有効である場合に、アプリケーション・サーバ上のユーザ・オブジェクトについてタグ付きコンポーネントを備えたＣＯＲＢＡ　ＩＯＲをエクスポートすること、及びアプリケーション・サーバのユーザ・オブジェクト・セキュリティが有効である場合に、アプリケーション・サーバ上のユーザ・オブジェクトについてＵＤＤＩレジストリにオブジェクト・タイプを提供することからなるグループからアクションを選択することによって、アプリケーション・サーバ・セキュリティ・フラグを管理オブジェクト・インターフェースに関連付けできることが好ましい。

【００２５】

保護されるべきオブジェクトの宣言リストにアクセスできることが好ましい。

【００２６】

アプリケーション・サーバ（例えばＷｅｂＳｐｈｅｒｅアプリケーション・サーバ）を提供することが好ましい。

【００２７】

Ｅｎｔｅｒｐｒｉｓｅ　Ｊａｖａ　Ｂｅａｎとしてクライアント・プロセスを提供することが好ましい。

【００２８】

Ｅｎｔｅｒｐｒｉｓｅ　Ｊａｖａ　Ｂｅａｎｓとしてユーザ・オブジェクトを提供することが好ましい。

【００２９】

ＭＢｅａｎとして管理オブジェクトを提供することが好ましい。

【００３０】

認証、許可、及びトランスポート保護のリストから選択されたセキュリティ・オペレーションを実行することが好ましい。

【００３１】

シンプル・オブジェクト・アクセス・プロトコル及びＯＲＢ間プロトコルのグループから選択されたセキュリティ・プロトコルを使用することが好ましい。

【００３２】

ＩＤＬと、Ｗｅｂサービス・エンドポイント言語及びインターフェース定義言語を備えたＷｅｂサービス定義言語とからなるグループから選択されたサービス記述モデルを使用

することが好ましい。

【００３３】

別の態様により、本発明は、分散型コンピュータ・ドメインにおいて用いるためのコンピュータ・プログラムであって、該プログラムがコンピュータ上で実行されるときに、管理オブジェクト及びユーザ・オブジェクトを1つ又は複数のアプリケーション・サーバに分散するステップと、管理オブジェクトのドメイン・レベル・セキュリティのためのグローバル・セキュリティ・フラグを定義するステップと、1つ又は複数のアプリケーション・サーバ・セキュリティ・フラグを前記分散ユーザ・オブジェクトへのインターフェースに関連付けるステップと、前記グローバル・セキュリティ・フラグ及び前記関連付けられたアプリケーション・サーバ・セキュリティ・フラグが有効である場合に、ユーザ・オブジェクト及び管理オブジェクトを保護する第１モードと、前記グローバル・セキュリティ・フラグが有効であり、かつ前記関連付けられたアプリケーション・サーバ・セキュリティ・フラグが無効である場合に、セキュリティ・オペレーションなしにユーザ・オブジェクトを用いるが、管理オブジェクトを保護する第２モードと、前記グローバル・セキュリティ・フラグが無効である場合に、セキュリティ・オペレーションなしにユーザ・オブジェクト及び管理オブジェクトを用いる第３モードとからなる３つのモードのうちの１つにおいて、1つ又は複数のセキュリティ・オペレーションを、アプリケーション・サーバによってクライアント・プロセスと協同して実行するステップとを実行するように構成されたプログラム・コード手段を含むコンピュータ・プログラムを、提供する。

【００３４】

Ｅｎｔｅｒｐｒｉｓｅ　Ｊａｖａ　Ｂｅａｎクライアント・プロセスがあることが好ましい。

【００３５】

Ｅｎｔｅｒｐｒｉｓｅ　Ｊａｖａ　Ｂｅａｎｓユーザ・オブジェクトがあることが好ましい。

【００３６】

ＭＢｅａｎ管理オブジェクトがあることが好ましい。

【００３７】

別の態様により、分散型コンピュータ・ドメインにおけるオブジェクト・レベル・セキュリティ・システムであって、1つ又は複数のアプリケーション・サーバに分散された1つ又は複数の管理オブジェクト及び1つ又は複数のユーザ・オブジェクトと、ネットワーク・コンピュータ・ドメイン・レベル内の前記管理オブジェクトのセキュリティを定義するグローバル・セキュリティ・フラグと、前記分散ユーザ・オブジェクトへのインターフェースに関連付けられた1つ又は複数のアプリケーション・サーバ・セキュリティ・フラグと、前記グローバル・セキュリティ・フラグ及び前記関連付けられたアプリケーション・サーバ・セキュリティ・フラグが有効である場合に、ユーザ・オブジェクト及び管理オブジェクトを保護する第１モードと、前記グローバル・セキュリティ・フラグが有効であり、かつ前記関連付けられたアプリケーション・サーバ・セキュリティ・フラグが無効である場合に、セキュリティ・オペレーションなしにユーザ・オブジェクトを用いるが、管理オブジェクトを保護する第２モードと、前記グローバル・セキュリティ・フラグが無効である場合に、セキュリティ・オペレーションなしにユーザ・オブジェクト及び管理オブジェクトを用いる第３モードとからなる３つのモードのうちの１つにおいて、1つ又は複数のセキュリティ・オペレーションを、アプリケーション・サーバによってクライアント・プロセスと協同して実行可能な1つ又は複数のセキュリティ・オペレーションとを含むシステムが、提供される。

【００３８】

セキュリティ・オペレーションが、管理オブジェクト及びユーザ・オブジェクトのための別個のセキュリティ・オペレーションを含むことが好ましい。

【００３９】

管理オブジェクト・インターフェースに関連付けられたアプリケーション・サーバ・セ

キュリティ・フラグは、アプリケーション・サーバのユーザ・オブジェクト・セキュリティが有効である場合に、アプリケーション・サーバ上のユーザ・オブジェクト用のタグ付きコンポーネントを備えたＣＯＲＢＡ　ＩＯＲと、アプリケーション・サーバのユーザ・オブジェクト・セキュリティが有効である場合に、アプリケーション・サーバ上のユーザ・オブジェクト用のＵＤＤＩレジストリにおけるオブジェクト・タイプとからなるグループから選択されたオブジェクト・タイプ・インジケータを含むことが好ましい。

【００４０】

ＩＯＲか又はＵＤＤＩレジストリのいずれのエントリがユーザ・オブジェクト・セキュリティを有効にさせるのに修正されるべきかを判断するために、保護されるべきオブジェクトの宣言リストがあることが好ましい。

【発明を実施するための最良の形態】

【００４１】

本発明の好ましい実施形態は、ここに、実例のみを目的に、添付図面を参照して記載することとする。

【００４２】

本発明は、好ましくは、パーソナル・コンピュータ、Ｗｅｂサーバ、及びＷｅｂブラウザなどの周知のコンピュータ・プラットフォーム上にある既に考え出されたエンタープライズ・サーバ・ソフトウェアの機能、拡張、又は改良として実現される。これらの共通のコンピュータ・プラットフォームは、パーソナル・コンピュータ、ワークステーション、及びメインフレームを含み、かつ携帯情報端末（「ＰＤＡ」）、Ｗｅｂ対応無線電話、及び他のタイプの個人情報管理（「ＰＩＭ」）デバイスなど、考えられるところでは適当に携帯可能なコンピュータ・プラットフォームを含むことができる。

【００４３】

そのため、コンピュータ・プラットフォームの汎用アーキテクチャを再検討することは有用であり、それは、ハイエンドＷｅｂ又はエンタープライズ・サーバ・プラットフォームから、パーソナル・コンピュータ、ポータブルＰＤＡ又はＷｅｂ対応無線電話までの実装範囲に及ぶことができるものである。

【００４４】

図１を参照すると、具体的にはランダム・アクセス・メモリ（「ＲＡＭ」）（４）と読み出し専用メモリ（「ＲＯＭ」）（５）とに関連付けられたマイクロプロセッサ（２）を備えた中央演算処理ユニット（１）（「ＣＰＵ」）を含む汎用アーキテクチャが、提示される。ＣＰＵ（１）にはまた、キャッシュ・メモリ（３）とプログラム可能なフラッシュＲＯＭ（６）とが提供されることがある。マイクロプロセッサ（２）と様々なタイプのＣＰＵメモリとの間にあるインターフェース（７）は、よく「ローカルバス」と呼ばれるが、より一般的な、すなわち業界標準バスにすることもできる。

【００４５】

コンピュータ・プラットフォームの多くには、また、ハードディスク・ドライブ（「ＨＤＤ」）、フロッピィディスク・ドライブ、（ＣＤ、ＣＤ－Ｒ、ＣＤ－ＲＷ、ＤＶＤ、ＤＶＤ－Ｒなどの）コンパクトディスク・ドライブ、及び（例えばＩｏｍｅｇａ　Ｚｉｐ及びＪａｚ、Ａｄｄｏｎｉｃｓ　ＳｕｐｅｒＤｉｓｋのような）プロプライエタリ・ディスク及びテープ・ドライブなどの１つ又は複数のストレージ・ドライブ（９）が、提供される。さらに、ストレージ・ドライブのいくつかは、コンピュータ・ネットワークを通じてアクセス可能にすることができる。

【００４６】

コンピュータ・プラットフォームの多くには、コンピュータ・プラットフォームの所期機能に応じて、１つ又は複数の通信インターフェースが、提供される。例えば、パーソナル・コンピュータには、多くの場合（ＲＳ－２３２、ＲＳ－４２２などの）高速シリアル・ポート、拡張パラレル・ポート（「ＥＰＰ」）、及び１つ又は複数のユニバーサル・シリアル・バス（「ＵＳＢ」）ポートが提供される。コンピュータ・プラットフォームには、また、イーサネット・カードなどのローカル・エリア・ネットワーク（「ＬＡＮ」）・

インターフェース、及びハイ・パフォーマンス・シリアル・バスＩＥＥＥ－１３９４などの他の高速インターフェースを提供することができる。

【００４７】

無線電話及び無線ネットワークＰＤＡなどのコンピュータ・プラットフォームには、また、さらにアンテナを備えた無線周波数（「ＲＦ」）インターフェースを提供することができる。場合によっては、コンピュータ・プラットフォームに、赤外線通信（ＩｒＤＡ）インターフェースを提供することができる。

【００４８】

コンピュータ・プラットフォームは、多くの場合、サウンドカード、メモリボード、及びグラフィック・アクセラレータなどの他のハードウェアを追加するための、業界標準アーキテクチャ（ＩＳＡ）、拡張業界標準アーキテクチャ（ＥＩＳＡ）、周辺コンポーネント相互接続（ＰＣＩ）、又はプロプライエタリ・インターフェース・スロットなどの、１つ又は複数の内部拡張スロット（１１）を搭載している。

【００４９】

さらに、ラップトップ・コンピュータ及びＰＤＡなどのユニットの多くには、１つ又は複数の外部拡張スロット（１２）が提供されており、それによりユーザは、ＰＣＭＣＩＡカード、ＳｍａｒｔＭｅｄｉａ（登録商標）カード、及び、取り外し可能ハード・ドライブ、ＣＤドライブ、フロッピィ・ドライブのような様々なプロプライエタリ・モジュールなどのハードウェア拡張デバイスを容易にインストールし、かつ取り外すことができる。

【００５０】

ストレージ・デバイス（９）、通信インターフェース（１０）、内部拡張スロット（１１）、及び外部拡張スロット（１２）は、ＩＳＡ、ＥＩＳＡ、又はＰＣＩなどの標準すなわち業界オープンのバス・アーキテクチャ（８）を介し、ＣＰＵ（１）と相互接続されることが多い。多くの場合、プロプライエタリ設計のバス（８）でもよい。

【００５１】

コンピュータ・プラットフォームには、通常、キーボード又はキーパッド（１６）などの１つ又は複数のユーザ入力デバイス、マウス又はポインタ・デバイス（１７）、及び／又はタッチスクリーン式ディスプレイが、提供される。パーソナル・コンピュータの場合は、トラックボール又はトラックポイントなどのマウス又はポインタ・デバイスと共にフルサイズのキーボードが、提供されることが多い。Ｗｅｂ対応無線電話の場合は、簡易キーパッドに、１つ又は複数の特定機能キーを提供することができる。ＰＤＡの場合は、通常、タッチスクリーン（１８）が提供され、多くの場合、手書き認識機能を備えている。

【００５２】

さらに、Ｗｅｂ対応無線電話のマイクロフォン又はパーソナル・コンピュータのマイクロフォンなどのマイクロフォン（１９）が、コンピュータ・プラットフォームに供給される。このマイクロフォンは、単にオーディオ及び音声信号を伝えるために用いることができ、また、音声認識機能を用いて、Ｗｅｂサイトの音声ナビゲーション又は電話番号の自動ダイヤルなどのユーザ選択を入力するために、用いることもできる。

【００５３】

コンピュータ・プラットフォームの多くはまた、デジタル静止画カメラ又はフルモーションビデオ・デジタルカメラなどのカメラ・デバイス（１００）を装備している。

【００５４】

ディスプレイ（１３）などの１つ又は複数のユーザ出力デバイスも、ほとんどのコンピュータ・プラットフォームに提供される。ディスプレイ（１３）は、陰極線管（「ＣＲＴ」）、薄膜トランジスタ（「ＴＦＴ」）・アレイ、又は簡易な一組の発光ダイオード（「ＬＥＤ」）もしくは液晶ディスプレイ（「ＬＣＤ」）・インジケータを含む多くの形態をとることができる。

【００５５】

１つ又は複数のスピーカ（１４）及び／又は報知器（１５）が、コンピュータ・プラットフォームに関連付けられることも多い。スピーカ（１４）は、無線電話のスピーカ又は

パーソナル・コンピュータのスピーカのように、音声及び音楽を再生するのに用いることができる。報知器（１５）は、ＰＤＡ及びＰＩＭなどの特定のデバイスに広く見られる簡易ビープ音エミッタ又はブザーの形態をとることができる。

【００５６】

これらのユーザ入出力デバイスは、プロプライエタリ・バス構造及び／又はインターフェースを介して、直接ＣＰＵ（１）に相互接続（８′、８′′）することができるか、又はＩＳＡ、ＥＩＳＡ、ＰＣＩなどの１つ又は複数の業界オープンのバスを介して相互接続することができる。コンピュータ・プラットフォームには、また、１つ又は複数のソフトウェア及びファームウェア（１０１）プログラムが提供され、コンピュータ・プラットフォームの所望の機能を実装する。

【００５７】

ここで図２を参照すると、このクラスのコンピュータ・プラットフォームにおける汎用ソフトウェア及びファームウェア（１０１）が、より詳細に示されている。コンピュータ・プラットフォームには、ワード・プロセッサ、スプレッドシート、連絡先管理ユーティリティ、アドレス帳、カレンダー、電子メール・クライアント、プレゼンテーション、財務及び会計のプログラムなどの１つ又は複数のオペレーティング・システム（「ＯＳ」）ネイティブ・アプリケーション・プログラム（２３）を提供することができる。

【００５８】

さらに、Ｊａｖａスクリプト及びプログラムなどのＯＳネイティブのプラットフォーム固有インタープリタ（２５）によって解釈されなければならない１つ又は複数の「移植可能」プログラムすなわちデバイス独立プログラム（２４）を提供することができる。

【００５９】

コンピュータ・プラットフォームにはまた、ブラウザ・プラグイン（２７）などの１つ又は複数のブラウザ拡張機能を含むこともできるＷｅｂブラウザ又はマイクロブラウザの一形態（２６）が、提供されることが多い。

【００６０】

コンピュータ・デバイスには、一般に、Ｍｉｃｒｏｓｏｆｔ Ｗｉｎｄｏｗｓ、ＵＮＩＸ、ＩＢＭ ＯＳ／２、ＬＩＮＵＸ、ＭＡＣ ＯＳ、又は他のプラットフォーム固有オペレーティング・システムなどのオペレーティング・システム（２０）が提供される。ＰＤＡ及び無線電話などのより小型のデバイスは、リアルタイム・オペレーティング・システム（「ＲＴＯＳ」）又はＰａｌｍ ＣｏｍｐｕｔｉｎｇのＰａｌｍＯＳなどの他の形態のオペレーティング・システムを装備することができる。

【００６１】

基本入出力機能（「ＢＩＯＳ」）とハードウェア・デバイス・ドライバとの組みが、多くの場合に提供され、オペレーティング・システム（２０）及びプログラムが、コンピュータ・プラットフォームに提供される特定ハードウェア機能とインターフェースし、それらを制御することを可能にする。

【００６２】

さらに、１つ又は複数の組み込みファームウェア・プログラム（２２）は、コンピュータ・プラットフォームの多くに広く提供されており、それらは、マイクロコントローラもしくはハード・ドライブ、通信プロセッサ、ネットワーク・インターフェース・カード、又はサウンドカードかもしくはグラフィックス・カードなどの周辺装置の一部として、オンボードの「組み込み」マイクロプロセッサによって、実行される。

【００６３】

よって、図１及び図２は、これらに限定されるものではないが、パーソナル・コンピュータ、ＰＤＡ、ＰＩＭ、Ｗｅｂ対応電話、及びＷｅｂＴＶ（商標）ユニットなどの他のアプライアンスを含む多岐にわたるコンピュータ・プラットフォームの様々なハードウェアコンポーネント、ソフトウェア及びファームウェア・プログラムを、一般的な意味で記述している。本発明は、好ましくは、そうしたコンピュータ・プラットフォーム上のソフトウェア及びファームウェアとして実装される。本発明は、代替的にハードウェア機能とし

て実現できることが、当業者であれば容易に認識されるであろう。

【００６４】

１つの可能な実施形態において、本発明は、以下の段落でより詳細に記載されるように、特定のシステムファイル及びリソースにアクセスし、修正するＪａｖａ　Ｂｅａｎｓの集合体又は組みとして実現されるが、本発明は、Ｗｅｂサーバ・スイートなどの既存のソフトウェアに統合することができる。

【００６５】

例示的な実施形態において、ＣＯＲＢＡアーキテクチャが使用され、そこでは、オブジェクトが、ＯＭＧインターフェース定義言語（「ＩＤＬ」）によって定義され、オブジェクトが、サーバ及びクライアント中に分散され、利用可能なオブジェクト及びサービスの発見及び使用を調整するためにオブジェクト・リクエスト・ブローカ（「ＯＲＢ」）が使用される。ＩＤＬコンパイラは、ＣＯＲＢＡバインディングを実装し、各オブジェクトのサービスをサーバ環境からクライアントにマッピングするクライアント・スタブ及びサーバ・スケルトンを生成する。クライアントとサーバとの間及びサーバ同士の間の通信プロトコルは、汎用ＯＲＢ間プロトコル（「ＧＩＯＰ」）と、伝送制御プロトコル／インターネット・プロトコル（「ＴＣＰ／ＩＰ」）上にＧＩＯＰを実装するＯＲＢ間プロトコル（「ＩＩＯＰ」）である。相互運用オブジェクト参照（「ＩＯＲ」）は、プラットフォーム独立及びＯＲＢ実装独立のオブジェクト参照を提供する。各ＣＯＲＢＡオブジェクトをパブリック・ネームにバインドする命名サービスが、ＣＯＲＢＡアーキテクチャ内に提供される。

【００６６】

さらに、この例示的実施形態において、Ｅｎｔｅｒｐｒｉｓｅ　Ｊａｖａ　Ｂｅａｎ（「ＥＪＢ」）プログラミング・モデルが使用され、そこでは、オブジェクト・インターフェース及びサーバ実装がＪａｖａ定義に従い、オブジェクトが遠隔操作でアクセスできるように分散され、かつリモート・メソッド呼び出し（「ＲＭＩ」）を含むＪａｖａ通信プロトコルが用いられる。そのようにして、この例示的実施形態は、ＩＢＭエンタープライズ・サーバ上のＡＩＸ又はＬＩＮＵＸオペレーティング・システムの元で稼働するＩＢＭ　ＷｅｂＳｐｈｅｒｅアプリケーション・サーバ製品を用いて、実装される。現バージョンのＷｅｂＳｐｈｅｒｅは、ＲＭＩ－ＩＩＯＰしかサポートしないので、クライアントは、好ましくは、ＪａｖａＲＭＩプロトコルのＲＭＩ－ＪＲＭＰフォーマットを用いないようにすべきである。他のウェブ・アプリケーション環境を用いる他の実施形態では、この制限は必要でないかもしれない。なお、この例示的実施形態におけるネーミング・サービスには、Ｊａｖａネーミング・サービス（「ＪＮＤＩ」）が使用される。

【００６７】

ＣＯＲＢＡ及びＥＪＢの一般概念、プロトコル、プログラミング方法論、及び機能は、通常、当該技術分野において周知であり、かつ、ＷｅｂＳｐｈｅｒｅ製品が普及し、それ自体が周知であるが、当業者であれば、本発明は、他のプログラミング方法論、プロトコル、コンピュータ・プラットフォーム、オペレーティング・システム・アーキテクチャ、及びウェブ・アプリケーション・サーバ製品によって、本発明の範囲を逸脱することなく実現することができるということを認識するであろう。例えば、本発明を実現するのに用いられる可能性があるアプリケーション・サーバのいくつかは、Ｏｒａｃｌｅの９ｉ製品、ＢＥＡ　Ｓｙｓｔｅｍｓ　ＷｅｂＬｏｇｉｃプラットフォーム、Ｓｕｎ　ＭｉｃｒｏｓｙｓｔｅｍｓのＯＮＥアプリケーション・サーバ、Ｈｅｗｌｅｔｔ－Ｐａｃｋａｒｄのアプリケーション・サーバ、及びＪＢｏｓｓを含む。オペレーティング・システムに関して、Ｕｎｉｘ、Ｌｉｎｕｘ、ＮｏｖｅｌｌのＮｅｔｗａｒｅ、ＩＢＭのＡＩＸ及びＯＳ／２、Ｓｕｎ　ＭｉｃｒｏｓｙｓｔｅｍｓのＳｏｌａｒｉｓ、Ｈｅｗｌｅｔｔ－ＰａｃｋａｒｄのＨＰ－ＵＸ、並びにＭｉｃｒｏｓｏｆｔのＷｉｎｄｏｗｓ製品はまた、代替実施形態において、代わりに使用することができる。

【００６８】

ここで、図４を参照すると、本発明の好ましい実施形態によるクライアント・システム

（４３）と複数のアプリケーション・サーバ（４１、４２）との相互作用及び関係（４０）が、管理ノード・エージェント（４７）及びネーム・サーバ（４６）を含めて、示される。アプリケーション・サーバ（４１、４２）と、管理ノード・エージェント（４７）と、ネーム・サーバ（４６）とは、ドメイン（４５）の一部であり、ここから、アプリケーションがクライアント・システム（４３）に提供される。

【００６９】

アプリケーションが始動すると、ＩＯＲインターセプタがロードされる特定の初期化手順が生じ、共有オブジェクトへの参照が、各アプリケーション・サーバ（４１、４２）からネーム・サーバ（４６）にエクスポートされ（４１２、４１３、４１４、４１５、４１６）、それにより、クライアント（４３）は共有オブジェクトを見出すことができる。

【００７０】

オブジェクトがネーム・サーバにエクスポートされているときには、ＩＯＲインターセプタは、すべてのオブジェクトに対して呼び出される。ＯＲＢがＩＯＲを作成するこの作成時間中に、どれもが１つ有するコンポーネント用のＩＯＲインターセプタのすべてが、呼び出される。これらのＩＯＲインターセプタの目的は、タグ付きコンポーネントをＩＯＲに追加することである。タグ付きコンポーネントは、サービスによって実行されるサービスに関する情報をサーバがエクスポートできるようにする追加情報である。この情報は次に、サービスを用いることが必要なクライアントによって読み出すことができる。例えばセキュリティ・サービスの場合、これらのタグ付きコンポーネントに配置された情報は、サーバが受信待機中であるセキュア・ソケット・レイヤー（「ＳＳＬ」）・ポートの標識、クライアントによって、どのタイプのＳＳＬ接続がサーバにより確立されなければならないのか、サーバによって、どのセキュリティ・メカニズムがサポートされるのか、クライアントからは、どのタイプの情報（例えばユーザ名、パスワード、証明書など）が必要とされているのか、及びクライアントが、ログイン目的のために、ユーザにサーバ・レルムのプロンプトを出せるように、サーバのレルムが何であるのか、といったような情報含むパラメータを含むことができる。セキュリティが無効にされた場合には、ＩＯＲには、いかなるタグ付きコンポーネントも追加されない。

【００７１】

ＩＯＲインターセプタがロードされると、インターセプタは、テキストファイルなどのファイルをロードする。そのファイルは、アプリケーション・サーバ上の他のユーザ・オブジェクトがセキュアでないときでも保護される必要がある管理オブジェクトのクラス及びパッケージの名前を含む。管理オブジェクトは、アプリケーション・サーバ上のユーザ・オブジェクトがセキュアであるか否かに関わらず、常にセキュアであるべきものと、管理者が判断することができる。例えば、この図において、アプリケーション・サーバ１（４１）は、すべてがセキュアであるユーザ・オブジェクト（４００）及び管理オブジェクト（４０１）を提供する。それにより、セキュアなＩＯＲがネーム・サーバ（４６）にエクスポートされる（４１２、４１３）。しかしながら、アプリケーション・サーバ２（４２）は、セキュアである必要がないユーザ・オブジェクト（４０３）を提供し、それにより、これらのオブジェクト（４０３）用のセキュアでないＩＯＲがネーム・サーバ（４６）にエクスポートされるが、一方で、同じサーバ（４２）上のセキュアな管理オブジェクト（４０４）についてはセキュアなＩＯＲが、エクスポートされる（４１６）。

【００７２】

このことは、管理セキュリティを損なうことなく、顧客が、サーバごとにセキュリティを無効にすることを可能にする。管理者がアプリケーション・サーバ・セキュリティを操作しているときは、グローバル・セキュリティ（例えばドメイン全体のセキュリティ）は、好ましくは、なお有効であるべきである。特定のアプリケーション・サーバのセキュリティが無効にされ、グローバル・セキュリティが有効にされると、ＩＯＲインターセプタは、ユーザ・オブジェクトのセキュリティ状態に関係なくセキュアに保たれるべきものであって、かつ、保護された管理オブジェクトごとに、タグ付きコンポーネントをエクスポートすべきかどうかと、どの種類のタグ付きコンポーネントをエクスポートすべきかとを

判断するための、クラス及びパッケージのリスト（例えばテキスト・ファイル）をロードすることができる。ユーザ・オブジェクトは、セキュアなものとして維持するこのパッケージ又はクラスのリストのメンバーではなく、かくして、ドメイン内でどのクラスがセキュアのままであるかを修正するために、宣言メソッドが管理者に提供される。このように、ユーザ・オブジェクトはリストに含まれず、かくして、それらのＩＯＲはタグ付きコンポーネントを受け取らないが、管理オブジェクトはリストに含まれ、それらのＩＯＲは、それらがセキュリティ依存型であると判断された場合に、タグ付きコンポーネントを受け取る。ユーザ・オブジェクトか管理オブジェクトかには関係なく、すべてのオブジェクトは、ネーム・サーバ（４６）によって処理されるネーム・スペースに追加される。これは、宣言的に構成可能なクラスベースのオブジェクト・セキュリティ・スキームを作成する。

【００７３】

この配置及びプロセスを用いて、アプリケーション・サーバ１（４１）のようなセキュリティが完全に有効であるサーバからのオブジェクト使用を、クライアント・オブジェクト（４０５）が要求したときには、クライアント（４３）とオブジェクト（４００、４０１）との間のトランザクション（４０７）のすべては、セキュアに実施される。クライアント・オブジェクト（４０５）は、アプリケーション・サーバ上のオブジェクト用のＩＯＲをネーム・サーバ（４６）で検索（４０６）したときに、すべてのオブジェクト（４００、４０１）が保護されており、安全に用いられなければならない（４０７）ことを示すタグを受信する。タグによって指示されるように、ＷｅｂＳｐｈｅｒｅ認証エンジン及び許可エンジンなどの認証及び許可プロセスは、安全に使用され（４１０）、管理ノード（４７）及び１つ又は複数の管理オブジェクト（４０２）が提供するように、セキュリティ・サービスを提供することができる。

【００７４】

ユーザ・オブジェクト・セキュリティを無効にしているアプリケーション・サーバ２（４２）が提供するユーザ・オブジェクト（４０３）をクライアント・プロセスが用いようと試みたときに、検索（４０６）アクションは、結果として、いかなるタグもクライアント・プロセスに受信させることはなく、かくして、安全でない態様でユーザ・オブジェクト（４０３）と対話する（４０８）ときに、いかなるセキュリティ・プロセスも使用されないことになる。かくして、これらのユーザ・オブジェクト（４０３）との通信及び対話（４０８）は、（例えば、認証又は識別情報なしに）匿名で、及び／又は（例えばトランスポート保護なしに）「平文で」実施することができる。しかしながら、アプリケーション・サーバ２（４２）が提供する管理オブジェクト（４０４）を用いたときに、検索アクションは、結果として、ＷｅｂＳｐｈｅｒｅの許可エンジン及び認証エンジンなどの、管理ノードからのセキュリティ機能を呼び出させることになるタグ受信を、もたらす。かくして、アプリケーション・サーバ２は、管理オブジェクト（４０４）がセキュアに保たれたままで、ユーザ・オブジェクト（４０３）が、セキュリティなしに、かつ暗示される本来的なパフォーマンスの向上（例えば、クライアントによる認証が不要であり、サーバによる許可が不要であり、クリデンシャルの作成及び格納がない、など）を伴い、首尾よくクライアントによって用いられるようにする。

【００７５】

図４の理解を容易にするために、網掛けされた囲み矢印（幅広の矢印）は、非セキュアな通信又は対話（例えば４０８）を表し、一方、網掛けのない囲み矢印は、セキュアな通信又は対話（例えば４０７、４１０、４１１）を表す。同様に、細い矢印（例えば４１２、４１３、４１４、４１５、４１６）は、タグ付きコンポーネントを伴う、セキュア・オブジェクト用のＩＯＲをエクスポートするオペレーションを表し、一方、矢印付きのより太い線（例えば４１５）は、タグ付きコンポーネントなしの、非セキュア・オブジェクト用のＩＯＲをエクスポートすることを表す。

【００７６】

この新規なプロセスを要約し、レビューするために、タグ付きコンポーネントを伴う又

はタグ付きコンポーネントなしのＩＯＲが適切に作成された後に、クライアントから、ユーザ・オブジェクトに対する非セキュアな未認証の要求があると、クライアント・オブジェクトは、ＩＯＲを検索して受信し、かつＯＲＢは、セキュリティ要求インターセプタを含む１つ又は複数の要求インターセプタを呼び出す。タグ付きコンポーネントがＩＯＲに含まれないことから、それは、認証及びＳＳＬなしにＴＣＰ／ＩＰ要求を行う。これは、非セキュア・ユーザ・オブジェクトがクリデンシャル、認証、又は許可なく用いられることを可能にする。しかしながら、いかなる管理オブジェクト又は保護されたユーザ・オブジェクトの使用も、たとえこれらの保護されたオブジェクトが非セキュア・ユーザ・オブジェクトと同じドメインにあっても、セキュアな通信（例えばＳＳＬ）の使用と同様に、認証及び許可のプロセスをトリガーする。

【００７７】

図５は、本発明の好ましい実施形態により、タグ付きコンポーネントを伴う、共有オブジェクトのためのＩＯＲをエクスポートする前述のプロセス（５０）を図式的に表す。サーバ又はアプリケーションのスタートアップ（５１）に際して、管理クラス記述子ファイル（５００）がアクセス又はロードされ（５２）、各共有オブジェクトについて（５３）、ＩＯＲインターセプタが呼び出される（５４）。直接指定により又は指定パッケージの一部として、オブジェクトが管理クラス内にあれば（５５）、ＩＯＲは、セキュリティ・タグを伴ってエクスポートされる（５７）。オブジェクトが管理クラス内になければ、そのオブジェクトのためのＩＯＲは、特別なセキュリティ・タグなしに普通にエクスポートされる（５６）。これは、共有され、かつＩＯＲがエクスポートされるべきすべてのオブジェクトについて、繰り返される（５８）。

【００７８】

図６は、本発明の好ましい実施形態により、クライアントが共有オブジェクト又はサービスを要求する前述のプロセス（６０）を図式的に表す。必要なサービスについえＩＯＲが検索され（６１）、受信されたＩＯＲがセキュリティ・タグを検査される（６２）。要求及びサービスを用いるのに、認証、許可、及び／又はトランスポート保護などのセキュリティが要求されていると（６３）、要求されたアクションが実行され（６５）、アプリケーション・サーバへのセキュアな通信リンクを認証し、許可し、及び／又は確立する。次に、サービス／オブジェクトが、セキュリティ及びトランスポート保護を伴って用いられる（６６）。さもなければ（６３）、サービスは、セキュリティ措置なしに普通に要求され、用いられる（６４）。

【００７９】

さらに、ＷｅｂＳｐｈｅｒｅプラットフォームを用いる例示的実施形態により、ユーザ・オブジェクト用の許可エンジンとは別個の管理オブジェクト用の許可エンジンが提供される。ＷｅｂＳｐｈｅｒｅ管理許可エンジンは、ロール・ベースの許可モデル又はプロセスを提供する。ユーザ・オブジェクトのために別個の許可エンジンを有することによって、ユーザ・オブジェクト・セキュリティのために任意のＪ２ＥＥを用いることができる。これら別個のユーザ・オブジェクト・セキュリティ・フラグ及び管理オブジェクト・フラグ（例えばグローバル・セキュリティ・フラグ）のユニークな使用法のために、システムは、異なる許可及び認証エンジンによってユーザ・オブジェクト及び管理オブジェクトのセキュリティを別々に定義し、管理するようにされる。これは、３つのセキュリティ・モードの構成を可能にする：

（ａ）　アプリケーション・サーバ・セキュリティが有効であり、かつグローバル・セキュリティが有効であるときには、ドメイン内の所与のサーバ上の（ユーザ及び管理）オブジェクトのすべてが保護される；

（ｂ）　アプリケーション・サーバ・セキュリティが無効であり、かつグローバル・セキュリティが有効であるときには、特定サーバ上でユーザ・オブジェクトのみが非セキュアであり、管理オブジェクトがセキュアに維持される；

（ｃ）　グローバル・セキュリティが無効であるときには、ドメイン内のすべてのサーバについてすべてのオブジェクト（ユーザ・オブジェクト及び管理オブジェクト）が非セキ

ュアである。

【００８０】

ＷｅｂＳｐｈｅｒｅ及びＣＯＲＢＡを用いるときの図４の実施形態において、クライアント・オブジェクト（４０５）及びユーザ・オブジェクト（４００、４０３）は、Ｅｎｔｅｒｐｒｉｓｅ　Ｊａｖａ　Ｂｅａｎｓ（ＥＪＢ）であり、管理オブジェクト（４０１、４０２、及び４０４）は、Ｊａｖａ　Ｍａｎａｇｅｍｅｎｔ　ＭＢｅａｎｓである。他の適当なオブジェクト指向プログラミング言語を代替的に用いることができる。

【００８１】

以上、本開示は、オブジェクト・レベル・セキュリティ・プロセス及びメカニズムを記載すべくＣＯＲＢＡ、ＩＩＯＰ、及びＩＯＲを用いて、その実施形態及び詳細を提示してきた。本発明は、好ましくは、限定されるものではないが、Ｗｅｂサービス・セキュリティを含む他のセキュリティ・プロトコル及びオブジェクト・モデルにも同様に適用され得る。Ｗｅｂサービス・セキュリティ・モデルでは、メッセージ受け渡し及びメソッド呼び出しは、ＩＩＯＰセキュリティ・プロトコルの代わりに、シンプル・オブジェクト・アクセス・プロトコル（「ＳＯＡＰ」）のメッセージによって行われ、サービスは、ＩＤＬの代わりに、Ｗｅｂサービス定義言語（「ＷＳＤＬ」）及びＷｅｂサービス・エンドポイント言語（「ＷＳＥＬ」）を用いて記述される。これに対応して、Ｗｅｂサービス・モデルにおいて、ＣＯＲＢＡのＣｏｓＮａｍｉｎｇの代わりに、Ｕｎｉｖｅｒｓａｌ　Ｄｅｓｃｒｉｐｔｉｏｎ，Ｄｉｓｃｏｖｅｒｙ　ａｎｄ　Ｉｎｔｅｇｒａｔｉｏｎ（「ＵＤＤＩ」）が用いられるが、前者は、ＩＯＲを検索するためのＡＰＩを提供し、一方、後者は、ビジネス・エントリ及びＷｅｂサービス・エントリを検索するためのＡＰＩを提供する。このように、ここで開示された例示的実施形態において、ＩＯＲは、オブジェクト・インスタンスのためのオブジェクト・タイプ情報を運ぶメカニズムとして用いられており、それは、ＷｅｂサービスＵＤＤＩオブジェクト・レジストリにオブジェクト・タイプ情報を提供することなどの代替的オブジェクト・モデル及びセキュリティ・プロトコルで、実現させることができる。

【図面の簡単な説明】

【００８２】

【図１】パーソナル・コンピュータ、サーバ・コンピュータ、携帯情報端末、Ｗｅｂ対応無線電話、又は他のプロセッサ・ベースのデバイスなどの汎用コンピューティング・プラットフォーム・アーキテクチャを示す。

【図２】図１の汎用アーキテクチャに関連付けられたソフトウェア及びファームウェアの一般化された編成を示す。

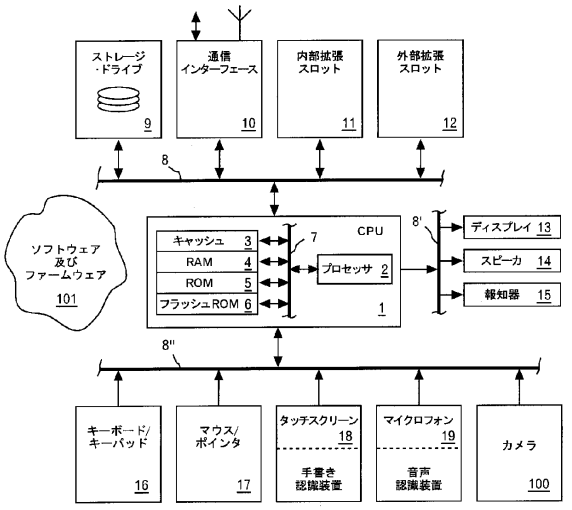
【図３】管理システム及びアプリケーションの典型的配置を例示する。

【図４】本発明の好ましい実施形態による本発明のプロセス及び対話を示す。

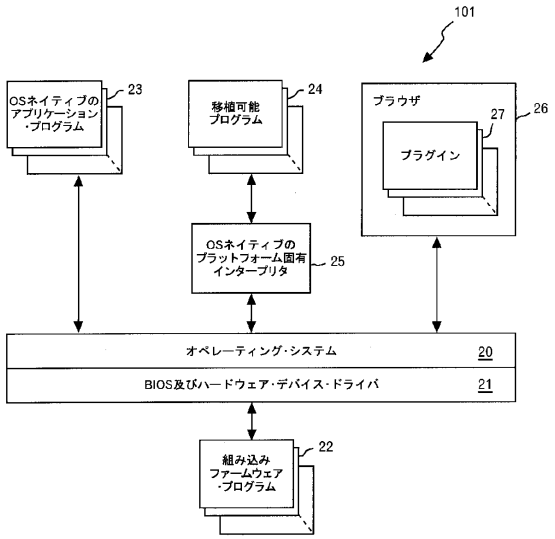
【図５】本発明の好ましい実施形態によるＩＩＯＰモデルの実装において、タグ付きコンポーネントを伴う、共有オブジェクト用のＩＯＲをエクスポートするプロセスを図形的に示す。

【図６】本発明の好ましい実施形態によるＩＩＯＰモデルの実装において、クライアントが共有オブジェクト又はサービスを要求するプロセスを図形的に示す。

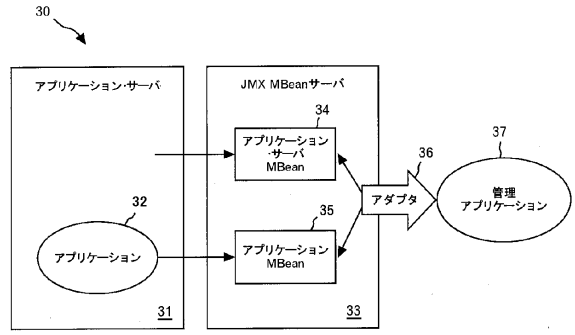
【図１】


ストレージ・ドライブ
9
通信インターフェース
10
内部拡張スロット
11
外部拡張スロット
12
8
ソフトウェア及びファームウェア
101
CPU
キャッシュ 3
RAM 4
ROM 5
フラッシュROM 6
7
プロセッサ 2
1
8'
ディスプレイ 13
スピーカ 14
報知器 15
8"
キーボード/キーパッド
16
マウス/ポインタ
17
タッチスクリーン
18
手書き認識装置
マイクロフォン
19
音声認識装置
カメラ
100

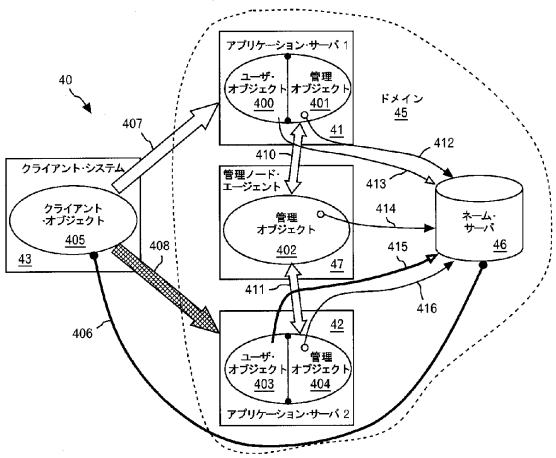
【図２】


101
OSネイティブのアプリケーション・プログラム
23
移植可能プログラム
24
ブラウザ
27
26
プラグイン
OSネイティブのプラットフォーム固有インタープリタ
25
オペレーティング・システム
20
BIOS及びハードウェア・デバイス・ドライバ
21
組み込みファームウェア・プログラム
22

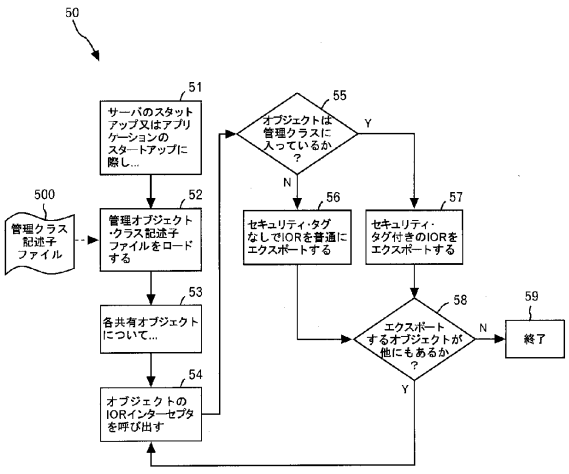
【図３】


30
アプリケーション・サーバ
JMX MBeanサーバ
34
アプリケーション・サーバMBean
37
36
管理アプリケーション
アダプタ
32
35
アプリケーションMBean
アプリケーション
31
33

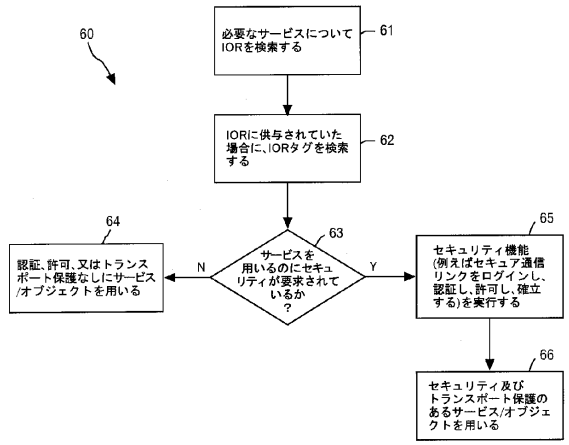
【図４】


40
アプリケーション・サーバ 1
ユーザ・オブジェクト 400
管理オブジェクト 401
41
ドメイン 45
407
412
410
413
クライアント・システム
クライアント・オブジェクト 405
43
管理ノード・エージェント
管理オブジェクト 402
47
414
ネーム・サーバ 46
408
415
411
416
406
42
ユーザ・オブジェクト 403
管理オブジェクト 404
アプリケーション・サーバ 2

【図５】


50
51
サーバのスタートアップ又はアプリケーションのスタートアップに際し...
55
オブジェクトは管理クラスに入っているか？
Y
N
500
管理クラス記述子ファイル
52
管理オブジェクト・クラス記述子ファイルをロードする
56
セキュリティ・タグなしでIORを普通にエクスポートする
57
セキュリティ・タグ付きのIORをエクスポートする
53
各共有オブジェクトについて...
58
エクスポートするオブジェクトが他にもあるか？
N
59
終了
Y
54
オブジェクトのIORインターセプタを呼び出す

【図６】


60
必要なサービスについてIORを検索する
61
IORに供与されていた場合に、IORタグを検索する
62
サービスを用いるのにセキュリティが要求されているか？
63
N
Y
64
認証、許可、又はトランスポート保護なしにサービス/オブジェクトを用いる
65
セキュリティ機能(例えばセキュア通信リンクをログインし、認証し、許可し、確立する)を実行する
66
セキュリティ及びトランスポート保護のあるサービス/オブジェクトを用いる

フロントページの続き

(74)代理人 100086243
弁理士 坂口 博
(72)発明者 バーク、ピーター
アメリカ合衆国 ７８７２７ テキサス州 オースチン チェースウッド・ドライブ １９２５
(72)発明者 チャオ、チンユン
アメリカ合衆国 ７８７５０ テキサス州 オースチン ラスティック・ロック・ドライブ １１５１８
(72)発明者 チャン、ヒョン
アメリカ合衆国 ７８６８１ テキサス州 ラウンド・ロック プリースト・リバー・ドライブ ８６１９
(72)発明者 メイソン、カールトン
アメリカ合衆国 ７８７２７ テキサス州 オースチン バリックス・コーブ １２６０２
(72)発明者 レディー、アジャイクマール
アメリカ合衆国 ７８７２７ テキサス州 オースチン オムロ・コーブ １５５１２
(72)発明者 ベンカタラマッパ、ビシュワナス
アメリカ合衆国 ７８７１７ テキサス州 オースチン フリッチュ・ドライブ ８７０４

審査官 市川 武宜

(56)参考文献 特開平１０－１８７６３７（ＪＰ，Ａ）
特開平１０－１７１６５７（ＪＰ，Ａ）
特開２０００－００３３４８（ＪＰ，Ａ）
特開２００２－２１５４８６（ＪＰ，Ａ）
特表２００２－５２８８１５（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
G06F 21/20
G09C 1/00