(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6510568号
(P6510568)
(45)発行日　令和1年5月8日(2019.5.8)　(24)登録日　平成31年4月12日(2019.4.12)

(51)Int.Cl.　F I
G06F 21/62 (2013.01)　G06F 21/62 318
G06F 21/53 (2013.01)　G06F 21/53

請求項の数 9　(全 30 頁)

(21)出願番号　特願2016-575001 (P2016-575001)
(86)(22)出願日　平成27年6月23日 (2015.6.23)
(65)公表番号　特表2017-526048 (P2017-526048A)
(43)公表日　平成29年9月7日 (2017.9.7)
(86)国際出願番号　PCT/US2015/037270
(87)国際公開番号　WO2015/200379
(87)国際公開日　平成27年12月30日 (2015.12.30)
審査請求日　平成30年1月26日 (2018.1.26)
(31)優先権主張番号　62/016,058
(32)優先日　平成26年6月23日 (2014.6.23)
(33)優先権主張国　米国 (US)
(31)優先権主張番号　62/054,912
(32)優先日　平成26年9月24日 (2014.9.24)
(33)優先権主張国　米国 (US)

(73)特許権者　502303739
オラクル・インターナショナル・コーポレイション
アメリカ合衆国カリフォルニア州９４０６５レッドウッド・シティー，オラクル・パークウェイ５００
(74)代理人　110001195
特許業務法人深見特許事務所
(72)発明者　ホプキンス，ウィル
アメリカ合衆国、９４０６５　カリフォルニア州、レッドウッド・シティー、オラクル・パークウェイ、５００

最終頁に続く

(54)【発明の名称】マルチテナントアプリケーションサーバ環境におけるセキュリティをサポートするためのシステムおよび方法

(57)【特許請求の範囲】
【請求項１】
複数のパーティションと、複数のパーティションリソースと、複数のグローバルリソースとを含むマルチテナントアプリケーションサーバ環境におけるセキュリティを提供するための方法であって、前記方法は、
管理セキュリティレルムと第１のセキュリティレルムと第２のセキュリティレルムとを含む複数のセキュリティレルムを規定するステップと、
前記複数のパーティションのうちの第１のパーティションを、前記複数のパーティションリソースのうちの第１の複数のパーティションリソースを有するように構成するステップと、
前記複数のパーティションのうちの第２のパーティションを、前記複数のパーティションリソースのうちの第２の複数のパーティションリソースを有するように構成するステップと、
前記第１のパーティションを前記第１のセキュリティレルムに関連付ける第１のセキュリティ構成を与えるステップと、
前記第２のパーティションを前記第２のセキュリティレルムに関連付ける第２のセキュリティ構成を与えるステップと、
第１のプライマリアイデンティティドメインを前記第１のパーティションに関連付けるステップとを含み、前記第１のプライマリアイデンティティドメインは、第１のテナントに関連付けられた第１の複数のユーザを表わし、

第２のプライマリアイデンティティドメインを前記第２のパーティションに関連付けるステップを含み、前記第２のプライマリアイデンティティドメインは、第２のテナントに関連付けられた第２の複数のユーザを表わし、
前記管理セキュリティレルム、前記第１のセキュリティレルム、および前記第２のセキュリティレルム各々を実行時に同時に動作させることにより、前記複数のパーティションリソースおよび前記複数のグローバルリソースへのアクセスの認証および認可を制御するステップを含み、
前記第１のテナントに関連付けられた前記第１の複数のユーザは、前記第１のパーティションの前記第１の複数のパーティションリソースにアクセスできるが、前記第２のパーティションの前記第２の複数のパーティションリソースにはアクセスできず、
前記第２のテナントに関連付けられた前記第２の複数のユーザは、前記第２のパーティションの前記第２の複数のパーティションリソースにアクセスできるが、前記第１のパーティションの前記第１の複数のパーティションリソースにはアクセスできない、方法。
【請求項２】
前記第１のプライマリアイデンティティドメインを、前記第１のテナントに関連付けられた前記第１の複数のユーザの第１の表現を格納するための第１のアイデンティティ記憶域を参照するように構成するステップと、
前記第２のプライマリアイデンティティドメインを、前記第２のテナントに関連付けられた前記第２の複数のユーザの第２の表現を格納するための、前記第１のアイデンティティ記憶域と異なる第２のアイデンティティ記憶域を参照するように構成するステップとをさらに含む、請求項１に記載の方法。
【請求項３】
前記第１のプライマリアイデンティティドメインを、前記第１のテナントに関連付けられた前記第１の複数のユーザの第１の表現を格納するためのアイデンティティ記憶域の第１の部分を参照するように構成するステップと、
前記第２のプライマリアイデンティティドメインを、前記第２のテナントに関連付けられた前記第２の複数のユーザの第２の表現を格納するための前記アイデンティティ記憶域の第２の部分を参照するように構成するステップとをさらに含む、請求項１に記載の方法。
【請求項４】
管理アイデンティティドメインを前記マルチテナントアプリケーションサーバ環境に関連付けるステップをさらに含み、前記管理アイデンティティドメインは、前記マルチテナントアプリケーションサーバ環境の複数のシステムアドミニストレータを表わし、
前記マルチテナントアプリケーションサーバ環境に関連付けられた前記複数のシステムアドミニストレータは、前記複数のグローバルリソースにアクセス可能である、請求項１～３のいずれかに記載の方法。
【請求項５】
第１の認証サービスを提供するステップをさらに含み、前記第１の認証サービスは、前記第１のテナントに関連付けられた前記第１の複数のユーザを認証するように構成されるとともに、前記第１の複数のユーザのうちの１人以上のユーザと組合わせて前記第１のプライマリアイデンティティドメインを識別する第１の署名付きプリンシパルを生成するように構成される、請求項１～４のいずれかに記載の方法。
【請求項６】
前記第１の複数のパーティションリソース各々を前記第１のプライマリアイデンティティドメインに関連付けるステップと、
前記第２の複数のパーティションリソースを各々前記第２のプライマリアイデンティティドメインに関連付けるステップと、
認可サービスを提供するステップとをさらに含み、前記認可サービスは、リソースへのアクセスのためのコールをユーザから受けたことに応じて、前記ユーザに関連付けられたプライマリアイデンティティドメインと前記リソースに関連付けられたプライマリアイデ

ンティティドメインとを比較し、前記ユーザに関連付けられた前記プライマリアイデンティティドメインと前記リソースに関連付けられた前記プライマリアイデンティティドメインとが一致する場合に限り、前記リソースへのアクセスを認可する、請求項１～５のいずれかに記載の方法。

【請求項７】

前記方法はさらに、

第１の認証サービスを提供するステップを含み、前記第１の認証サービスは、前記第１のテナントに関連付けられた前記第１の複数のユーザを認証するように構成されるとともに、前記第１の複数のユーザのうちの１人以上のユーザと組合わせて前記第１のプライマリアイデンティティドメインを識別する第１の署名付きプリンシパルを生成するように構成され、

第２の認証サービスを提供するステップを含み、前記第２の認証サービスは、前記第２のテナントに関連付けられた前記第２の複数のユーザを認証するように構成されるとともに、前記第２の複数のユーザのうちの１人以上のユーザと組合わせて前記第２のプライマリアイデンティティドメインを識別する第２の署名付きプリンシパルを生成するように構成され、

前記第１の複数のパーティションリソース各々を前記第１のプライマリアイデンティティドメインに関連付けるステップと、

前記第２の複数のパーティションリソース各々を前記第２のプライマリアイデンティティドメインに関連付けるステップと、

認可サービスを提供するステップとを含み、前記認可サービスは、プリンシパルに関連付けられた、リソースへのアクセスのためのコールを受けたことに応じて、前記プリンシパルにおいて識別されたプライマリアイデンティティドメインを、前記リソースに関連付けられたプライマリアイデンティティドメインと比較し、前記プリンシパルに関連付けられたプライマリアイデンティティドメインが前記リソースに関連付けられた前記プライマリアイデンティティドメインと一致する場合に限り、前記リソースへのアクセスを認可するように構成される、請求項１～４のいずれかに記載の方法。

【請求項８】

コンピュータシステムによって実行されると前記コンピュータシステムに請求項１～７のいずれかに記載の方法を実行させる機械読取可能なフォーマットのプログラム命令を含むコンピュータプログラム。

【請求項９】

マルチテナントアプリケーションサーバ環境システムであって、

複数のマイクロプロセッサとメモリとを含むアプリケーションサーバ環境と、

前記アプリケーションサーバ環境に構成された複数のパーティションと、

前記アプリケーションサーバ環境に設けられた複数のパーティションリソースと複数のグローバルリソースと、

前記アプリケーションサーバ環境に構成された管理セキュリティレルムと第１のセキュリティレルムと第２のセキュリティレルムとを含む複数のセキュリティレルムと、

前記複数のパーティションリソースのうちの第１の複数のパーティションリソースを有するように構成された前記複数のパーティションのうちの第１のパーティションと、

前記複数のパーティションリソースのうちの第２の複数のパーティションリソースを有するように構成された前記複数のパーティションのうちの第２のパーティションと、

前記第１のパーティションを前記第１のセキュリティレルムに関連付ける第１のセキュリティ構成と、

前記第２のパーティションを前記第２のセキュリティレルムに関連付ける第２のセキュリティ構成と、

前記第１のパーティションに関連付けられた第１のプライマリアイデンティティドメインとを備え、前記第１のプライマリアイデンティティドメインは、第１のテナントに関連付けられた第１の複数のユーザを表わし、前記マルチテナントアプリケーションサーバ環

境システムはさらに、
前記第２のパーティションに関連付けられた第２のプライマリアイデンティティドメインを備え、前記第２のプライマリアイデンティティドメインは、第２のテナントに関連付けられた第２の複数のユーザを表わし、
前記管理セキュリティレルム、前記第１のセキュリティレルム、および前記第２のセキュリティレルムは、実行時に同時に動作することにより、前記複数のパーティションリソースおよび前記複数のグローバルリソースへのアクセスの認証および認可を制御するように構成され、
前記第１のテナントに関連付けられた前記第１の複数のユーザは、前記第１のパーティションの前記第１の複数のパーティションリソースにアクセスできるが、前記第２のパーティションの前記第２の複数のパーティションリソースにはアクセスできず、
前記第２のテナントに関連付けられた前記第２の複数のユーザは、前記第２のパーティションの前記第２の複数のパーティションリソースにアクセスできるが、前記第１のパーティションの前記第１の複数のパーティションリソースにはアクセスできない、マルチテナントアプリケーションサーバ環境システム。
【発明の詳細な説明】
【技術分野】
【０００１】
著作権に関する注意
本特許文献の開示の一部には、著作権保護の対象となるものが含まれている。著作権者は、この特許文献または特許開示の何者かによる複製が、特許商標庁の特許ファイルまたは記録にある限り、それに対して異議を唱えないが、そうでなければ、いかなる場合もすべての著作権を留保する。
【０００２】
発明の分野：
本発明の実施形態は、概して、アプリケーションサーバおよびクラウド環境に関し、特に、マルチテナントアプリケーションサーバ環境におけるセキュリティを提供するためのシステムおよび方法に関する。
【背景技術】
【０００３】
背景：
ソフトウェアアプリケーションサーバは、その例として、Oracle WebLogic Server（ＷＬＳ）およびGlassfishを含んでおり、概して、エンタープライズソフトウェアアプリケーションを実行するための管理された環境を提供する。近年、クラウド環境において使用するための技術も開発されており、ユーザまたはテナントが、クラウド環境内でそれらのアプリケーションを開発して実行することが可能になり、かつ、環境によって提供される分散型リソースを活用することが可能になっている。
【発明の概要】
【課題を解決するための手段】
【０００４】
概要：
一実施形態に従い、本明細書において、マルチテナントアプリケーションサーバ環境におけるセキュリティを提供するためのシステムおよび方法が説明される。一実施形態に従うと、パーティションごとのセキュリティ構成は、パーティションごとのセキュリティレルム（realm）（認証、認可、資格証明マッピング、監査、パスワード検証、証明書検証、およびユーザロックアウトのための構成を含む）と、キー、証明書、およびその他の構成属性を含むＳＳＬ構成と、パーティションおよびグローバルリソースに対するアクセス制御とを含む。アドミニストレータは、１以上のパーティションユーザを、ロールの付与により、パーティションアドミニストレータとして指定することができる。
【図面の簡単な説明】

【０００５】
【図１】一実施形態に従った、アプリケーションサーバ、クラウドまたは他の環境においてマルチテナンシをサポートするためのシステムを示す図である。
【図２】一実施形態に従った、アプリケーションサーバ、クラウドまたは他の環境においてマルチテナンシをサポートするためのシステムをさらに示す図である。
【図３】一実施形態に従った、アプリケーションサーバ、クラウドまたは他の環境においてマルチテナンシをサポートするためのシステムをさらに示す図である。
【図４】一実施形態に従った、リソースグループテンプレートの代表的な使用を示す図である。
【図５】一実施形態に従った例示的なマルチテナント環境を示す図である。
【図６】一実施形態に従った、マルチテナントアプリケーションサーバ環境におけるパーティションワークマネージャの使用を示す図である。
【図７】一実施形態に従った、マルチテナントアプリケーションサーバ環境におけるパーティションワークマネージャの使用を示す図である。
【図８】マルチテナントアプリケーションサーバ環境において使用するワークマネージャを提供するための代表的な方法のフローチャートである。
【図９】マルチテナントアプリケーションサーバ環境において使用するワークマネージャを提供するための代表的な方法のフローチャートである。
【発明を実施するための形態】
【０００６】
詳細な説明：
一実施形態に従い、本発明は、パーティションごとのセキュリティサービス構成を提供し、この構成は、マルチテナントサーバ環境における、認証、認可、資格証明マッピング、監査、パスワード検証、証明書検証、およびユーザロックアウトのための構成を含む。本発明の実施形態はまた、パーティションおよびグローバルリソースに対するアクセス制御を提供し、この制御によって、特定のパーティションに対してデプロイされたアプリケーションはこの特定のパーティションのユーザしかアクセスできないようにし、特定のパーティションのためのパーティション構成はこの特定のパーティションのアドミニストレータ（およびWebLogicアドミニストレータ）しか利用できないようにし、かつ、グローバルに可視であるリソースおよび構成に対してパーティションアドミニストレータは読取専用アクセスしかできないようにする。図１～図５およびそれに関連する文章は、マルチテナントアプリケーションサーバ環境を説明する。図６～図８およびそれに関連する文章は、たとえば図１～図５について説明するマルチテナントサーバ環境におけるセキュリティをサポートするためのシステムおよび方法を説明する。
【０００７】
以下の説明において、本発明を、添付の図面において、限定ではなく例示のために説明する。本開示においてさまざまな実施形態に言及する場合、これは、かならずしも同一の実施形態であるとは限らず、少なくとも１つの実施形態を意味する。特定の実装について述べるが、これは専ら例示を目的としていることが理解される。当業者は、本発明の範囲および精神から逸脱することなくその他の構成要素および構成を使用し得ることを理解するであろう。
【０００８】
さらに、特定の例では、本発明を十分に説明するために多数の具体的な詳細事項について述べる。しかしながら、本発明はこのような具体的な詳細事項がなくても実施し得ることが当業者には明らかであろう。それ以外の例では、本発明が曖昧にならないよう周知の特徴は具体的に説明していない。
【０００９】
図面および詳細な説明全体を通して、共通する参照番号は同様の要素を示すために用いられている。したがって、ある図面で使用されている参照番号は、この図面に限定される詳細な説明において、もし対応する要素がその他の場所で説明されていれば、言及される

とは限らない。３桁の参照番号のうちの最初の桁は一連の図面を表わし、対応する要素が最初に示される図面を示す。

【００１０】

アプリケーションサーバ（たとえば、マルチテナント（Multi-Tenant：ＭＴ）環境

図１は、一実施形態に従った、アプリケーションサーバ、クラウドまたは他の環境においてマルチテナンシをサポートするためのシステムを示す。

【００１１】

図１に示されるように、一実施形態に従うと、アプリケーションサーバ（たとえばマルチテナント（ＭＴ））環境１００または他のコンピューティング環境は、ソフトウェアアプリケーションのデプロイメントおよび実行を可能にするものであって、アプリケーションサーバドメインを定義するために実行時に用いられるドメイン１０２構成を含み、当該ドメイン１０２構成に従って動作するように構成することができる。

【００１２】

一実施形態に従うと、アプリケーションサーバは、実行時に使用されるよう定義される１つ以上のパーティション１０４を含み得る。各々のパーティションは、グローバルユニークパーティション識別子（identifier：ＩＤ）およびパーティション構成に関連付けることができ、さらに、リソースグループテンプレートの参照１２６および／またはパーティション特有のアプリケーションもしくはリソース１２８とともに、１つ以上のリソースグループ１２４を含み得る。ドメインレベルのリソースグループ、アプリケーションおよび／またはリソース１４０も、任意にはリソースグループテンプレートの参照とともに、ドメインレベルで定義することができる。

【００１３】

各々のリソースグループテンプレート１６０は、１つ以上のアプリケーションＡ１６２、Ｂ１６４、リソースＡ１６６、Ｂ１６８および／または他のデプロイ可能なアプリケーションもしくはリソース１７０を定義することができ、リソースグループによって参照することができる。たとえば、図１に例示されるように、パーティション１０４におけるリソースグループ１２４は、リソースグループテンプレート１６０を参照する（１９０）ことができる。

【００１４】

概して、システムアドミニストレータは、パーティション、ドメインレベルのリソースグループおよびリソースグループテンプレート、ならびにセキュリティ領域を定義することができるとともに、パーティションアドミニストレータは、たとえば、パーティションレベルのリソースグループを作成するか、アプリケーションをパーティションにデプロイするかまたはパーティションについての特定の領域を参照することによって、それら自体のパーティションのアスペクトを定義することができる。

【００１５】

図２は、一実施形態に従った、アプリケーションサーバ、クラウドまたは他の環境においてマルチテナンシをサポートするためのシステムをさらに示す。

【００１６】

図２に示されるように、一実施形態に従うと、パーティション２０２は、たとえば、リソースグループテンプレート２１０の参照２０６を含むリソースグループ２０５と、仮想ターゲット（たとえば仮想ホスト）情報２０７と、プラガブルなデータベース（pluggable database：ＰＤＢ）情報２０８とを含み得る。リソースグループテンプレート（たとえば２１０）は、たとえば、Ｊａｖａ（登録商標）メッセージサーバ（Java Message Server：ＪＭＳ）サーバ２１３、ストア・アンド・フォワード（store-and-forward：ＳＡＦ）エージェント２１５、メールセッションコンポーネント２１６またはＪａｖａデータベースコネクティビティ（Java Database Connectivity：ＪＤＢＣ）リソース２１７などのリソースとともに、複数のアプリケーションＡ２１１およびＢ２１２を定義することができる。

【００１７】

図２に例示されるリソースグループテンプレートが一例として提供される。他の実施形態に従うと、さまざまなタイプのリソースグループテンプレートおよび要素を提供することができる。

【００１８】

一実施形態に従うと、パーティション（たとえば２０２）内のリソースグループが、特定のリソースグループテンプレート（たとえば２１０）を参照する（２２０）と、パーティション特有の情報２３０（たとえば、パーティション特有のＰＤＢ情報）を示すために、特定のパーティションに関連付けられた情報を、参照されたリソースグループテンプレートと組合わせて用いることができる。次いで、パーティション特有の情報は、パーティションによって使用されるリソース（たとえば、ＰＤＢリソース）を構成するようにアプリケーションサーバによって使用可能である。たとえば、パーティション２０２に関連付けられたパーティション特有のＰＤＢ情報は、そのパーティションによって使用されるべき適切なＰＤＢ２３８を備えたコンテナデータベース（container database：ＣＤＢ）２３６を構成する（２３２）ようにアプリケーションサーバによって使用可能である。

【００１９】

同様に、一実施形態に従うと、特定のパーティションに関連付けられた仮想ターゲット情報を用いて、そのパーティションによって使用されるべきパーティション特有の仮想ターゲット２４０（たとえば、ユニフォーム・リソース・ロケータ（uniform resource locator：ＵＲＬ）（たとえば、http://baylandurgentcare.com）によってアクセス可能にすることができるbaylandurgentcare.com）を定義する（２３９）ことができる。

【００２０】

図３は、一実施形態に従った、アプリケーションサーバ、クラウドまたは他の環境においてマルチテナンシをサポートするためのシステムをさらに示す。

【００２１】

一実施形態に従うと、config.xml構成ファイルなどのシステム構成を用いて、パーティションを定義することができる。当該パーティションは、そのパーティションに関連付けられたリソースグループについての構成エレメントおよび／または他のパーティションプロパティを含む。値は、プロパティ名／値の対を用いてパーティションごとに指定することができる。

【００２２】

一実施形態に従うと、複数のパーティションは、管理されたサーバ／クラスタ２４２内で、または、ＣＤＢ２４３にアクセス可能でありかつウェブ層２４４を介してアクセス可能である同様の環境内で、実行することができる。これにより、たとえば、ドメインまたはパーティションを（ＣＤＢの）ＰＤＢのうち１つ以上のＰＤＢに関連付けることが可能となる。

【００２３】

一実施形態に従うと、複数のパーティションの各々、この例においてはパーティションＡ２５０およびパーティションＢ２６０は、そのパーティションに関連付けられた複数のリソースを含むように構成することができる。たとえば、パーティションＡは、アプリケーションＡ１　２５２と、アプリケーションＡ２　２５４と、ＪＭＳ　Ａ　２５６と、さらには、ＰＤＢ　Ａ　２５９に関連付けられたデータソースＡ２５７とをともに含むリソースグループ２５１を含むように構成することができる。この場合、パーティションは仮想ターゲットＡ２５８を介してアクセス可能である。同様に、パーティションＢ２６０は、アプリケーションＢ１　２６２と、アプリケーションＢ２　２６４と、ＪＭＳ　Ｂ　２６６と、さらには、ＰＤＢ　Ｂ　２６９に関連付けられたデータソースＢ２６７とをともに含むリソースグループ２６１を含むように構成することができる。この場合、パーティションは仮想ターゲットＢ２６８を介してアクセス可能である。

【００２４】

上述の例のうちいくつかはＣＤＢおよびＰＤＢの使用を例示しているが、他の実施形態に従うと、他のタイプのマルチテナントのデータベースまたは非マルチテナントのデータ

ベースをサポートすることができる。この場合、特定の構成は、たとえば、スキーマを使用するかまたはさまざまなデータベースを使用することによって、各々のパーティションのために提供することができる。

【００２５】

リソース

一実施形態に従うと、リソースは、環境のドメインにデプロイすることができるシステムリソース、アプリケーションまたは他のリソースもしくはオブジェクトである。たとえば、一実施形態に従うと、リソースは、アプリケーション、ＪＭＳ、ＪＤＢＣ、ＪａｖａＭａｉｌ、ＷＬＤＦもしくはデータソースであり得るか、または、サーバ、クラスタもしくは他のアプリケーションサーバターゲットにデプロイすることができる他のシステムリソースもしくは他のタイプのオブジェクトであり得る。

【００２６】

パーティション

一実施形態に従うと、パーティションは、パーティション識別子（partition identifier：ＩＤ）および構成に関連付けられ得るドメインのうちランタイムおよび管理の区分またはスライスであるとともに、アプリケーションを含み得て、ならびに／または、リソースグループおよびリソースグループテンプレートを使用することによってドメイン全体に渡るリソースを参照し得る。

【００２７】

概して、パーティションは、それ自体のアプリケーションを含み、リソースグループテンプレートを介してドメイン全体に渡るアプリケーションを参照し、それ自体の構成を有し得る。パーティション分割可能なエンティティは、リソース、たとえば、ＪＭＳ、ＪＤＢＣ、ＪａｖａＭａｉｌ、およびＷＬＤＦリソースや、他のコンポーネント、たとえばＪＮＤＩネームスペース、ネットワークトラフィック、ワークマネージャ、セキュリティポリシーおよび領域などを含み得る。マルチテナント環境のコンテキストにおいては、システムは、テナントに関連付けられたパーティションの管理およびランタイムのアスペクトへのアクセスをテナントに提供するように構成することができる。

【００２８】

一実施形態に従うと、パーティション内の各々のリソースグループは、任意には、リソースグループテンプレートを参照することができる。パーティションは、複数のリソースグループを有し得るとともに、それらの各々はリソースグループテンプレートを参照し得る。各々のパーティションは、パーティションのリソースグループが参照するリソースグループテンプレートにおいて指定されていない構成データについてのプロパティを定義することができる。これにより、パーティションが、リソースグループテンプレートで定義されたデプロイ可能なリソースをそのパーティションで使用されるべき特定の値にバインドするものとして機能することが可能となる。場合によっては、パーティションは、リソースグループテンプレートによって指定される構成情報を無効にすることができる。

【００２９】

一実施形態に従うと、パーティション構成は、たとえば、config.xml構成ファイルによって定義されるように、複数の構成エレメントを含み得る。複数の構成エレメントは、たとえば、「パーティション（partition）」（そのパーティションを定義する属性および子エレメントを含む）；「リソース・グループ（resource-group）」（パーティションにデプロイされるアプリケーションおよびリソースを含む）；「リソース・グループ・テンプレート（resource-group-template）」；（そのテンプレートによって定義されるアプリケーションおよびリソースを含む）；「ｊｄｂｃ・システム・リソース・無効化（jdbc-system-resource-override）」（データベース特有のサービス名、ユーザ名およびパスワードを含む）；ならびに、「パーティション・プロパティ（partition-properties）」（リソースグループテンプレートにおいてマクロ置換のために使用可能なプロパティキー値を含む）を含む。

【００３０】

始動後、システムは、構成ファイルによって提供される情報を用いて、リソースグループテンプレートから各々のリソースについてのパーティション特有の構成エレメントを生成することができる。

【００３１】

リソースグループ

一実施形態に従うと、リソースグループは、名前付けされ完全に修飾されたデプロイ可能なリソースの集合であって、ドメインまたはパーティションのレベルで定義することができ、かつ、リソースグループテンプレートを参照することができる。リソースグループにおけるリソースは、完全に修飾されているものと見なされる。というのも、アドミニストレータが、それらのリソースを開始させるのに必要とされるかまたはそれらのリソースに接続するのに必要とされるすべての情報、たとえば、データソースに接続するためのクレデンシャル、またはアプリケーションについての目標情報、を提供しているからである。

【００３２】

システムアドミニストレータは、ドメインレベルで、またはパーティションレベルでリソースグループを公開することができる。ドメインレベルでは、リソースグループは、関連するリソースをグループ化するのに好都合な方法を提供する。システムは、グループ化されていないリソースと同じドメインレベルのリソースグループにおいて公開されたリソースを管理することができる。このため、リソースは、システム起動中に開始させたり、システムのシャットダウン中に停止させたりすることができる。アドミニストレータはまた、グループ内のリソースを個々に停止させるか、開始させるかまたは削除することができ、グループ上で動作させることによって暗黙的にグループ内のすべてのリソースに対して機能することができる。たとえば、あるリソースグループを停止させることにより、まだ停止されていないグループにおけるすべてのリソースを停止させ；リソースグループを始動させることにより、まだ始動させていないグループにおけるいずれのリソースも始動させ、リソースグループを削除することにより、グループに含まれるすべてのリソースを削除する。

【００３３】

パーティションレベルでは、システムまたはパーティションアドミニストレータは、任意のセキュリティ制限下で、或るパーティションにおいて０個以上のリソースグループを構成することができる。たとえば、ＳａａＳ使用事例においては、さまざまなパーティションレベルのリソースグループは、ドメインレベルのリソースグループテンプレートを参照することができる。ＰａａＳ使用事例においては、リソースグループテンプレートを参照しないが代わりにそのパーティション内でのみ利用可能にされるべきアプリケーションおよびそれらの関連するリソースを表わすパーティションレベルのリソースグループを作成することができる。

【００３４】

一実施形態に従うと、リソースグループ化を用いることで、アプリケーションと、それらアプリケーションがドメイン内で別個の管理ユニットとして使用するリソースとをともにグループ化することができる。たとえば、以下に記載される医療記録（ＭｅｄＲｅｃ）アプリケーションにおいては、リソースグループ化によりＭｅｄＲｅｃアプリケーションおよびそのリソースが定義される。複数のパーティションは、各々がパーティション特有の構成情報を用いて、同じＭｅｄＲｅｃリソースグループを実行することができ、このため、各々のＭｅｄＲｅｃインスタンスの一部であるアプリケーションが各々のパーティションにとって特有のものにされる。

【００３５】

リソースグループテンプレート

一実施形態に従うと、リソースグループテンプレートは、リソースグループから参照することができドメインレベルで定義されるデプロイ可能なリソースの集合であり、そのリソースを起動するのに必要な情報のうちいくらかは、パーティションレベル構成の仕様を

サポートするように、テンプレート自体の一部として記憶されない可能性がある。ドメインは、リソースグループテンプレートをいくつ含んでもよく、それらの各々は、たとえば、１つ以上の関連するＪａｖａアプリケーションと、それらのアプリケーションが依存するリソースとを含み得る。このようなリソースについての情報のうちのいくらかは、すべてのパーティションにわたって同じであってもよく、他の情報はパーティションごとに異なっていてもよい。すべての構成がドメインレベルで指定される必要はなく、代わりに、パーティションレベル構成が、マクロまたはプロパティ名／値の対を使用することによってリソースグループテンプレートで指定することができる。

【００３６】

一実施形態に従うと、特定のリソースグループテンプレートは、１つ以上のリソースグループによって参照可能である。概して、任意の所与のパーティション内では、リソースグループテンプレートは一度に１つのリソースグループによってのみ参照することができる。すなわち、同じパーティション内で同時に複数のリソースグループによって参照することはできない。しかしながら、異なるパーティションにおける別のリソースグループによって同時に参照することができる。リソースグループを含むオブジェクト、たとえばドメインまたはパーティションは、プロパティ名／値の割当てを用いて、任意のトークンの値をリソースグループテンプレートで設定することができる。システムは、参照するリソースグループを用いてリソースグループテンプレートを起動させると、それらのトークンを、リソースグループが含むオブジェクトにおいて設定された値と置換えることができる。場合によっては、システムはまた、静的に構成されたリソースグループテンプレートおよびパーティションを用いて、パーティション／テンプレートの組合せごとにランタイム構成を生成することができる。

【００３７】

たとえば、ＳａａＳ使用事例においては、システムは、同じアプリケーションおよびリソースを複数回起動することができるが、そのうちの１回は、それらを用いるであろう各パーティションごとに起動され得る。アドミニストレータがリソースグループテンプレートを定義すると、これらは、どこか他のところで提供されるであろう情報を表わすためにトークンを用いることができる。たとえば、ＣＲＭ関連のデータリソースに接続する際に使用されるユーザ名は、リソースグループテンプレートにおいて＼＄｛ＣＲＭＤａｔａＵｓｅｒｎａｍｅ｝として示すことができる。

【００３８】

テナント

一実施形態に従うと、マルチテナント（ＭＴ）アプリケーションサーバ環境などのマルチテナント環境においては、テナントは、１つ以上のパーティションおよび／もしくは１つ以上のテナント認識型アプリケーションによって表現可能であるエンティティ、または１つ以上のパーティションおよび／もしくは１つ以上のテナント認識型アプリケーションに関連付けることができるエンティティである。

【００３９】

たとえば、テナントは、別個のユーザ組織、たとえばさまざまな外部会社、特定の企業内のさまざまな部門（たとえばＨＲおよび財務部）などを表わすことができ、それら各々は、異なるパーティションに関連付けることができる。テナントのグローバルユニークアイデンティティ（テナントＩＤ）は、特定の時点において特定のユーザを特定のテナントに関連付けるものである。システムは、たとえば、ユーザアイデンティティの記録を参照することによって、ユーザアイデンティティから、特定のユーザがどのテナントに属しているかを導き出すことができる。ユーザアイデンティティにより、ユーザが実行することを認可されているアクションをシステムが実施することが可能となる。ユーザアイデンティティは、ユーザがどのテナントに属し得るかを含むが、これに限定されない。

【００４０】

一実施形態に従うと、システムは、互いに異なるテナントの管理およびランタイムを分離することを可能にする。たとえば、テナントは、それらのアプリケーションのいくつか

の挙動、およびそれらがアクセスできるリソースを構成することができる。システムは、特定のテナントが別のテナントに属するアーティファクトを確実に管理することができないようにし、かつ、実行時に、特定のテナントの代わりに機能するアプリケーションがそのテナントに関連付けられたリソースのみを参照するが他のテナントに関連付けられたリソースは参照しないことを確実にすることができる。

【００４１】

一実施形態に従うと、テナント非認識型アプリケーションは、アプリケーションが応答している要求をどんなユーザが提示したかにかかわらず、アプリケーションが用いる如何なるリソースにもアクセス可能となるように明示的にテナントに対処する論理を含まないものである。対照的に、テナント認識型アプリケーションは、テナントに明示的に対処する論理を含む。たとえば、ユーザのアイデンティティに基づいて、アプリケーションは、ユーザが属するテナントを導き出すことができ、テナント特有のリソースにアクセスするためにその情報を用いることができる。

【００４２】

一実施形態に従うと、システムは、テナント認識型となるように明示的に書き込まれたアプリケーションをユーザがデプロイすることを可能にし、これにより、アプリケーション開発者は、現在のテナントのテナントＩＤを取得することができる。次いで、テナント認識型アプリケーションは、このテナントＩＤを用いて、アプリケーションの単一のインスタンスを用いている複数のテナントを処理することができる。

【００４３】

たとえば、単一の診療室または病院をサポートするＭｅｄＲｅｃアプリケーションは、２つの異なるパーティションまたはテナント（たとえばBayland Urgent Careテナントおよびvalley Healthテナント）に対して公開することができ、その各々は、基礎をなすアプリケーションコードを変更することなく、別個のＰＤＢなどの別個のテナント特有のリソースにアクセスすることができる。

【００４４】

例示的なドメイン構成およびマルチテナント環境

一実施形態に従うと、アプリケーションは、ドメインレベルでリソースグループテンプレートにデプロイすることができるか、または、パーティションに範囲指定されているかもしくはドメインに範囲指定されているリソースグループにデプロイすることができる。アプリケーション構成は、アプリケーションごとまたはパーティションごとに指定されたデプロイメントプランを用いて無効化することができる。デプロイメントプランはまた、リソースグループの一部として指定することができる。

【００４５】

図４は、一実施形態に従った、例示的なマルチテナント環境で使用されるドメイン構成を示す。

【００４６】

一実施形態に従うと、システムがパーティションを始動させると、当該システムは、提供された構成に従って、それぞれのデータベースインスタンスに対して、各パーティションごとに１つずつ、仮想ターゲット（たとえば仮想ホスト）および接続プールを作成する。

【００４７】

典型的には、各々のリソースグループテンプレートは、１つ以上の関連するアプリケーションと、それらアプリケーションが依存するリソースとを含み得る。各々のパーティションは、それが参照するリソースグループテンプレートにおいて指定されていない構成データを提供することができるが、これは、場合によっては、リソースグループテンプレートによって指定されるいくつかの構成情報を無効にすることを含めて、パーティションに関連付けられた特定値に対するリソースグループテンプレートにおけるデプロイ可能なリソースのバインディングを行なうことによって、実行可能である。これにより、システムは、各々のパーティションが定義したプロパティ値を用いて、パーティションごとにリソ

ースグループテンプレートによってさまざまに表わされるアプリケーションを始動させることができる。

【００４８】

いくつかのインスタンスにおいては、パーティションが含み得るリソースグループは、リソースグループテンプレートを参照しないか、または、それら自体のパーティション範囲指定されたデプロイ可能なリソースを直接定義する。パーティション内で定義されるアプリケーションおよびデータソースは、概して、そのパーティションにとってのみ利用可能である。リソースは、パーティション：<partitionName>/<resource JNDI name>、またはドメイン：<resource JNDI name>を用いて、パーティションの中からアクセスすることができるようにデプロイ可能である。

【００４９】

たとえば、ＭｅｄＲｅｃアプリケーションは、複数のＪａｖａアプリケーション、データソース、ＪＭＳサーバおよびメールセッションを含み得る。複数のテナントのためにＭｅｄＲｅｃアプリケーションを実行させるために、システムアドミニストレータは、テンプレートにおけるそれらのデプロイ可能なリソースを公開している単一のＭｅｄＲｅｃリソースグループテンプレート２８６を定義することができる。

【００５０】

ドメインレベルのデプロイ可能なリソースとは対照的に、リソースグループテンプレートにおいて公開されたデプロイ可能なリソースは、テンプレートにおいて完全には構成されない可能性があるか、または、いくつかの構成情報が不足しているので、そのままでは起動させることができない。

【００５１】

たとえば、ＭｅｄＲｅｃリソースグループテンプレートは、アプリケーションによって用いられるデータソースを公開し得るが、データベースに接続するためのＵＲＬを指定しない可能性がある。さまざまなテナントに関連付けられたパーティション、たとえば、パーティションＢＵＣ－Ａ２９０（Bayland Urgent Care：ＢＵＣ）およびパーティションＶＨ－Ａ２９２（Valley Health：ＶＨ）は、各々がＭｅｄＲｅｃリソースグループテンプレートを参照する（２９６，２９７）ＭｅｄＲｅｃリソースグループ２９３，２９４を含むことによって、１つ以上のリソースグループテンプレートを参照することができる。次いで、当該参照を用いて、Bayland Urgent Careテナントによって使用されるＢＵＣ－Ａパーティションに関連付けられた仮想ホストbaylandurgentcare.com３０４と、Valley Healthテナントによって使用されるＶＨ－Ａパーティションに関連付けられた仮想ホストvalleyhealth.com３０８とを含む各々のテナントのための仮想ターゲット／仮想ホストを作成する（３０２，３０６）ことができる。

【００５２】

図５は、一実施形態に従った例示的なマルチテナント環境をさらに示す。図５に示されるように、２つのパーティションがＭｅｄＲｅｃリソースグループテンプレートを参照している上述の例から引続いて、一実施形態に従うと、サーブレットエンジン３１０は、この例においてはBayland Urgent Careの医師テナント環境３２０およびValley Healthの医師テナント環境３３０といった複数のテナント環境をサポートするために用いることができる。

【００５３】

一実施形態に従うと、各々のパーティション３２１および３３１は、そのテナント環境についての入来トラフィックを受入れるための異なる仮想ターゲットと、異なるＵＲＬ３２２，３３２とを定義することができる。異なるＵＲＬ３２２，３３２は、パーティションと、この例ではbayland urgent careデータベースまたはvalley healthデータベースを含むそれぞれのリソース３２４、３３４とに接続するためのものである。同じアプリケーションコードが両方のデータベースに対して実行され得るので、データベースインスタンスは互換性のあるスキーマを用いることができる。システムがパーティションを始動させると、当該システムは、それぞれのデータベースインスタンスに対する接続プールおよび

仮想ターゲットを作成することができる。

【００５４】

パーティションセキュリティ

上記パーティションの特徴は、個々のパーティション間の分離を提供するとともに、パーティションとこれらのパーティションを含むマルチテナントシステムとの間の分離を提供する。先に述べたように、各パーティションは、自身のアプリケーションを含むことができ、ドメイン全体に渡るアプリケーションを、リソースグループテンプレートを介して参照することができ、かつ、自身の構成（configuration）を有することができる。パーティション分割可能なエンティティは、リソース、たとえば、ＪＭＳ、ＪＤＢＣ、ＪａｖａＭａｉｌ、ＷＬＤＦリソース、および、他のコンポーネント、たとえばＪＮＤＩネームスペース、ネットワークトラフィック、ワークマネージャなどを含み得る。マルチテナントシステム内のセキュリティ機能については、パーティション分割された／マルチテナント環境におけるアプリケーションとシステムを安全なものにするために、この分離をモデル化しアクセス制御を適切に強化することが望ましい。本発明の一実施形態に従うと、このシステムは、マルチテナントアプリケーションサーバ環境におけるセキュリティに対するサポートを含み得る。このシステムは、上記マルチテナント環境とパーティションという特徴をサポートするセキュリティサービスを提供する。

【００５５】

本発明の実施形態は、パーティションごとのセキュリティサービス構成を提供し、この構成は、認証、認可、資格証明マッピング、監査、パスワード検証、証明書検証、およびユーザロックアウトのための構成を含む。本発明の実施形態はまた、パーティションおよびグローバルリソースに対するアクセス制御を提供し、この制御によって、特定のパーティションに対してデプロイされたアプリケーションはこの特定のパーティションのユーザしかアクセスできないようにし、特定のパーティションのためのパーティション構成はこの特定のパーティションのアドミニストレータ（およびWebLogicアドミニストレータ）しか利用できないようにし、かつ、グローバルに可視であるリソースおよび構成に対してパーティションアドミニストレータは読取専用アクセスしかできないようにする。

【００５６】

図６は、一実施形態に従う、マルチテナント環境におけるセキュリティをサポートするためのシステムおよび方法の一般的な機能を示す。図６に示されるように、パーティションＡ　２５０およびパーティションＢ　２６０それぞれに、別々のパーティションセキュリティ構成Ａ　６５０およびパーティションセキュリティ構成Ｂ　６６０を設けることができる。パーティションセキュリティ構成Ａ　６５０は、パーティションＡのユーザおよびアドミニストレータに対して、パーティションＡのリソース２５１への安全なアクセスと、グローバルリソース１４０への限定されたアクセスを提供する一方で、これらのアドミニストレータおよびユーザがパーティションＢのリソース２６１にアクセスすることを阻止する。ドメインアドミニストレータ６００は、各ドメインのパーティションセキュリティ構成を初期化することにより、特定のパーティションに対してデプロイされたアプリケーションはこの特定のパーティションのユーザおよびアドミニストレータしかアクセスできないように、パーティションのリソースおよびグローバルリソースに対するアクセスを保証することができる。

【００５７】

本発明の実施形態は、強化されたプラガブルな（pluggable）アイデンティティ管理（identity Management）ｉｄＭシステムをサポートし利用する。ｉｄＭシステムは、システムおよびパーティションの境界内のまたは境界にまたがる、個々のプリンシパル、その認証、認可の管理と、特権制御を提供する。また、本発明の特定の実施形態は、１以上のパーティションユーザを特定のパーティションのパーティションアドミニストレータとして指定する機能を与える管理サービスを提供する。

【００５８】

マルチテナントサーバ環境においてセキュリティに対して提供されるサポートは、主と

して以下のものを含むレルムおよびレルムベースのサービスに関連する多数の特徴および挙動を含む。

【００５９】

・複数のレルム：複数のアクティブなレルムに対するサポートは、各パーティションが異なるレルムに対して実行されるようにすることによって、レルムベースのサービスのパーティションごとの構成を可能にする。また、パーティションは、セキュリティレルムを共有することを選択してもよく、その場合はその結果としてセキュリティ構成とメタデータについて独立性と分離性を失うことになる。

【００６０】

・アイデンティティドメイン：アイデンティティドメインは、典型的には物理的な記憶域内のユーザおよびグループの離散集合を表わすユーザおよびグループのための論理的な名前空間である。アイデンティティドメインは、特定のパーティションに他の目的で関連付けられたユーザを識別するために使用される。

【００６１】

・パーティション認識セキュリティサービス：パーティション認識セキュリティサービスは、このサービスが実行されているパーティションコンテキストを理解するので、たとえば、パーティションが所有するリソースに対するアクセスをこのパーティションに基づいて制御することができる。

【００６２】

これらの特徴および機能領域については以下で詳細に説明する。

複数レルムのサポート

レルムまたはセキュリティレルムは、システム内のセキュリティサービスのための名前付き構成である。レルムは、マルチテナントサーバ環境において使用される、認証、認可、ロールマッピング、資格証明マッピング、監査、およびその他のサービスを構成するために用いられる。本発明の実施形態は、認証、認可、資格証明マッピング、監査、パスワード検証、証明書検証、およびユーザロックアウトのための構成を含むセキュリティサービスの、パーティションごとの構成を提供する。このシステムは複数のアクティブなレルムをサポートする。アクティブなパーティションは各々、異なるセキュリティレルムに関連付けられてもよい。よって、本発明の実施形態が提供するマルチテナントシステムでは、複数のレルムランタイムに対応しかつ複数の異なるパーティションに対応する複数のアクティブな構成が存在し得る。

【００６３】

このシステムが提供するセキュリティサービスは、「レルムベース」または「非レルムベース」のサービスとして特徴付けることができる。レルムベースのサービスは、構成がレルムオブジェクトによって表わされるセキュリティ「レルム」に対して構成されたサービスである。レルムベースのサービスは、認証、認可、資格証明マッピング、監査、およびその他いくつかのサービスを含む。非レルムベースのサービスは、レルムオブジェクトに対して構成されていないサービスである。これらは、セキュリティ構成オブジェクト（ドメインオブジェクトの子、レルムオブジェクトの親コンテナ）およびその他さまざまなサービスに対して構成された、ドメイン全体に渡るサービスおよび設定を含む。

【００６４】

図７は、複数のレルムの使用のために構成されたマルチテナント環境を示す。パーティションセキュリティは、セキュリティサービスを利用して複数のアクティブなレルムをサポートする。複数のレルムオブジェクトインスタンスは、セキュリティ構成オブジェクトに対するアレイ属性として管理される。図７に示されるように、セキュリティ構成オブジェクト７００は、３つのレルムオブジェクト、すなわちレルムＡ　７５０と、デフォルトレルム７０２と、レルムＢ　７６０とを含む。デフォルトレルム７０２は、ドメイン／グローバルランタイムによって使用されるレルムを表わす。しかしながら、レルムＡ　７５０およびレルムＢ　７６０も、実行時にはアクティブである。デフォルトレルムは、「管理レルム」と呼ぶこともできる。なぜなら、システムおよび管理リソースに対してかつそ

の他の管理目的のために認可検査を実施するために用いられるからである。

【００６５】

各パーティションセキュリティ構成は、ドメイン構成のセキュリティ構成オブジェクトに対して構成されたレルムのうちの１つを参照するレルム属性を有する。よって、パーティションセキュリティ構成Ａ　６５０は、レルムＡ　７５０を参照するレルム属性７５２を含み、パーティションセキュリティ構成Ｂ　６６０は、レルムＢ　７６０を参照するレルム属性７６２を含む。パーティションのセキュリティレルムの構成をサポートするための方法がパーティションリソースに追加される。各パーティションセキュリティ構成は、有効なレルムリファレンスを有していなければならない（これはまた、レルムを、パーティションによって参照される場合は削除できないことを示唆する）。パーティションのレルムリファレンスを変更できるのは、システムアドミニストレータ６００のみであり、パーティションアドミニストレータはできない。パーティションは、そのレルム参照が変更できるようになる前に停止しなければならない。

【００６６】

レルムに対するパーティションのマッピングは完全に柔軟であり、すべてのパーティションは異なるレルムを参照できる、または、すべてのパーティションは同一のレルムを共有できる、または、一部のパーティションは異なるレルムを参照することができその他のパーティションは１つのレルムを共有できる。どの組合わせも可能である。パーティションごとに別々のレルムを構成することにより、パーティションについて最大の独立性と分離性が得られる。複数のパーティションで１つのレルムを共有することは、構成を単純化することを可能にし、互いに信頼し合いかつ同様のセキュリティ構成要求を有する関連するパーティションについては良い選択であろう。しかしながら、独立性と分離性は低下し、１つのパーティションのアドミニストレータが、他の共有パーティションに影響するアクションを実施する可能性がある。デフォルト／管理レルムの共有は可能であるが推奨されない。デフォルトレルムを共有する場合は、パーティションアドミニストレータによる／のための特権拡大を回避するよう注意しなければならない。なぜなら、デフォルトポリシーは、デフォルトレルムにおけるロールマッピングおよび認可ポリシーを修正できるからである。

【００６７】

レルムは、それを表わすレルムオブジェクトが、システムアドミニストレータによってインスタンス化されセキュリティ構成オブジェクト内のレルムアレイに追加されたときに、作成される。新たなレルムの作成は、オンラインでもオフラインでも生じ得る。レルム構成は、変更が保存／起動される度に検証される、または、サーバがブートするときにオフライン変更に対して検証される。サーバがブートするためにはデフォルトレルムが構成され有効でなければならない。レルムの作成は、レルムのランタイムライフサイクルを管理するのに必要なランタイムオブジェクトの作成をトリガする。レルムは、一旦作成されたら使用するために利用できる。これは、パーティションからまたはドメイン／グローバルランタイムから参照することができ、ランタイムコードはそのサービスを要求できる。作成されたレルムは、使用パターンと構成変更に応じて、何度でも、始動、停止、または再始動することができる。

【００６８】

レルムは、要求をサービスする必要があるときにまたは管理目的で、オンデマンドで始動される。特に、パーティションが始動されるときにはパーティションのレルムも（既に実行されていなければ）始動される。レルムサービスに対するランタイムリファレンスは、レルムが存在する限り、たとえレルムが再始動されるときでも、有効な状態のままでなければならない。存在しないレルムについてのサービスを取得しようまたは呼び出そうとする試みは、結果としてエラーまたは例外になる。存在するレルムについてのサービスを取得しようまたは呼び出そうとする試みは、要求を満たすにはレルムを始動させねばならない場合であっても、また、サービスリファレンスの取得後にレルムが始動、停止、または再始動された場合でも、常に成功するはずである。

【００６９】
パーティション分割された環境では、アクティブなレルムが複数存在する可能性がある。そのため、レルムサービスを要求するときには常にレルム名が指定される。指定されたレルム名は、特定のレルムまたはデフォルトレルムを特定する。レルム名パラメータは、そのサービスを要求するレルムを特定するストリング値である。ほとんどのサービス呼び出しについて、使用すべき正しいレルムは、呼び出し元のパーティションコンテキスト、呼び出されているサービスおよび方法、およびコールのパラメータに応じて決まる。デフォルトレルムが指定された場合は、ロジックが適用されて、各コールのコンテキストを評価する「サービスプロキシ」によって正しいレルムを選択し、委任先の正しいレルムを決定し、そのレルムにおける適切なサービスを呼び出す。このようにしてレルム選択をカプセル化することによって、複雑なロジックを実装するためにコードを呼び出す必要性または複数のレルムからのサービスに対するリファレンスをキャッシュするためにコードを呼び出す必要性を回避する。ランタイムコードは、セキュリティサービスマネージャから、レルムベースのセキュリティサービスに対するリファレンスを取得する。ほとんどの呼び出し元は、初期化するときに１度必要なサービスに対するリファレンスを取得し、これらを必要である限り保存する。

【００７０】
セキュリティサービスマネージャは、レルムサービスに対する直接的なリファレンスの代わりにサービスプロキシを返す。サービスプロキシの挙動は、サービスを要求したときに与えられたレルム名に基づいて変化する。既存のレルムの実際の名前が指定されると、返されたプロキシはハードワイヤードされてその特定のレルムに委任される。ハードワイヤードされたプロキシサービスは、すべての要求を、構成されたレルムに委任する。以下の部分的なコードは、プリンシパルオーセンティケータ（Principal Authenticator）サービスに対するリファレンスを如何にして取得し得るかを示す。

【００７１】
```
import weblogic.security.service.PrincipalAuthenticator;
import weblogic.security.service.SecurityServiceManager;
PrincipalAuthenticator pa =
SecurityServiceManager.getPrincipalAuthenticator(kernelID, realmName)
```
上記のように、呼び出し元は、特定されたレルム名について、プリンシパルオーセンティケータサービスを要求する。システムのセキュリティサービスは、指定されたレルムによって決まるレルムサービスプロキシリファレンスを返す。

【００７２】
セキュリティサービス要求において「デフォルト」レルムが指定されると、サービスについて返されるプロキシは、返すべき正しいプロキシを選択するためにコンテキストに依存する（または「自動選択」）挙動を提供する。自動選択プロキシサービスは、いくつかの要求を、現在のパーティションのために構成されたレルムに委任し、その一方で、その他の要求を、デフォルト／管理レルムに委任する。認証サービスは、常にローカルであり現在のパーティションのレルムに委任される。認可サービスは、たとえばサーブレットまたはＥＪＢリソース等のパーティションリソースについては、ローカルであり現在のパーティションのレルムに委任されることがある。認可サービスは、たとえばＪＭＸおよび管理リソースであるシステムリソース等のグローバルリソースについては、グローバルでありデフォルト／管理レルムに委任されることがある。監査サービスは、常にローカルであり現在のパーティションのレルムに委任される。レルムサービスは、自身のレルムからの監査サービスを用いて自身を監査する。資格証明マッピングサービス、証明書確立／検証、パスワード検証、およびユーザロックアウトは、常にローカルであり、現在のパーティションのレルムに委任される。

【００７３】
アイデンティティドメイン
アイデンティティドメイン（Identity Domain：ＩＤＤ）は、ユーザおよびグループの

ための論理的な名前空間である。アイデンティティドメインは、さまざまなユーザの組を特定し区別するために用いられる。アイデンティティドメインは、特定の企業（たとえば「Acme社」ＩＤＤ）の、または、その企業内の部署（たとえば「ＨＲ部」ＩＤＤ）のユーザを表わし得る。クラウド環境において、アイデンティティドメインは、システムアドミニストレータを、顧客サービスの代表者から区別することができ、かつこれらをテナントユーザから区別することができる。アイデンティティドメインは、ユーザを異なるパーティションから区別しリソースの所有権の所属先にするのに必要である。したがって、これらは、あるパーティションを別のパーティションから区別できるロールマッピングおよび認可ポリシーのため、および、ユーザまたはリソースを異なるパーティションから区別しなければならないその他のサービスのための、実現技術（enabling technology）である。このように、アイデンティティドメインは、パーティション分割された実質的にすべての環境で使用されることを意図している。

【００７４】

アイデンティティドメインによって表わされる論理名前空間は、対応するユーザ記憶域の構造／トポロジにおいて、物理的に発現される。アイデンティティドメインの物理的表現は、ターゲットユーザ記憶技術によってサポートされるものであればよく、たとえば、アイデンティティドメインごとの別々のＬＤＡＰインスタンス、またはデータベース内のユーザ記録に追加されるアイデンティティドメインフィールドである。共有ＩｄＭ/オラクルパブリッククラウド（Oracle Public Cloud：ＯＰＣ）の場合、アイデンティティドメインは、１つのオラクルインターネットディレクトリ（Oracle Internet Directory：ＯＩＤ）インスタンスのユーザおよびグループ階層における明確なサブツリーとして表わされる。

【００７５】

マルチテナントサーバ環境において、アイデンティドメインは、「テナント」と「パーティション」との間の接続／整列ポイントの役割を果たす。１つのアイデンティティドメインは、テナントのユーザを表わす（すなわち、テナントとアイデンティティドメインとの間には１対１のマッピングがある）。このことは、特定のパーティションを、テナントに関連付けられたアイデンティティドメインを用いるように構成されているのであれば、テナントに有効に「割当てる」またはテナントによって有効に「所有される」ことができる。すなわち、このテナントのユーザはパーティションにアクセスできるが、他のテナントのユーザはそれができない。

【００７６】

パーティションのプライマリアイデンティティドメイン－アイデンティティドメインは、パーティション分割された環境においていくつかの目的に応える。何よりも先ず、アイデンティティドメインは、異なるパーティションにそれぞれ関係付けられたユーザを区別する。各パーティションは、そのパーティションに関連付けられた一組のユーザを特定するプライマリアイデンティティドメイン（Primary Identity Domain:ＰＩＤＤ）で構成される。デフォルトアクセスポリシーにより、上記ユーザ（その他のアイデンティティドメインからのユーザではなく）は、そのパーティションにアクセスできる。複数のパーティションが同一のプライマリアイデンティティドメインを構成することは可能であるが、その効果として、これらのパーティションのユーザ間の区別ができなくなる。共有アイデンティティドメインからのユーザは、共有パーティションすべてにアクセスできる。プライマリアイデンティティドメインはまた、ロールマッピングおよび認可ポリシーを記述するのに好都合な形態でパーティションのリソースの「所有権」を示す役割を果たす。パーティションのリソースを、あたかもパーティションのアイデンティティドメインによって「所有されている」ように扱うことによって、ユーザのアイデンティティドメインとリソースのアイデンティティドメインを比較し易くなる。アイデンティティドメインが使用中のとき、１つのアイデンティティドメインを、そのドメインの管理アイデンティティドメイン（Administrative Identity Domain：ＡＩＤＤ）として指定しなければならない。これは、そのドメインのプライマリアイデンティティドメインであるのが効果的である。アイ

デンティドメインは、システムアドミニストレータの所属先であり、システムおよび管理リソースの「所有権」の帰属先である。

【００７７】

図７は、アイデンティティドメインが使用中であるマルチテナント環境１００を示す。図７に示されるように、パーティションＡ　２５０の構成６５０は、プライマリアイデンティティドメインＰＩＤＤ　Ａ　７５４のＩＤを含む。ＰＩＤＤ　Ａ　７５４はＬＤＡＰＡ　７５６に関連付けられている。パーティションＢ　２６０の構成６６０は、プライマリアイデンティティドメインＰＩＤＤ　Ｂ　７６４のＩＤを含む。ＰＩＤＤ　Ｂ　７６４はＬＤＡＰ　Ｂ　７６６に関連付けられている。ドメインレベルのセキュリティ構成は、管理アイデンティティドメインＡＩＤＤ　７０４を参照する。ＡＩＤＤ７０４はＬＤＡＰ７０６に関連付けられている。なお、図７では独立したＬＤＡＰディレクトリを用いて示されているが、各アイデンティティドメインのユーザ記憶域を、テナントのニーズまたはテナントによって提供される既存のユーザ記憶域に応じて構成することができる。ユーザ記憶域は、マルチテナント環境１００内部でも外部でもよい。これに代えて、ＰＩＤＤＡ７５４およびＰＩＤＤ　Ｂ　７６６は、たとえば、ＬＤＡＰ７０６の別々のサブツリーを参照してもよく、または、データベース内のユーザ記録に追加されたアイデンティティドメインフィールドを参照してもよい。プライマリアイデンティティドメインの参照により、プラガブルなインターフェイスを構成することができる。このインターフェイスは、プライマリアイデンティティドメインに関連付けられたどのようなユーザ記憶域に対するコールも処理する。異なるパーティションは異なる種類のユーザ記憶域を使用し得る。

【００７８】

アイデンティティドメインは、マルチテナントサーバ環境において少なくとも１つのアイデンティティドメインが構成されていれば、「使用中」とみなされる。アイデンティティドメインが使用中であるとき、アイデンティティドメインは、プリンシパルが常にアイデンティティドメインフィールドを有しているという意味において、常に「イネーブルにされ」、認証プロバイダは常に、ゼロでない／空でないアイデンティティドメイン値を有するプリンシパルを作成できる。アイデンティティドメインは、セキュリティ構成オブジェクトの管理アイデンティティドメイン属性、システム内の各パーティションオブジェクトの主アイデンティドメイン属性、および、システム内で構成されたアイデンティティドメイン認識認証プロバイダのアイデンティティドメイン属性で、構成される。これらの属性のうちのいずれかがゼロでなく空でない値に設定されている場合、アイデンティティドメインは使用中とみなされる。

【００７９】

アイデンティティドメインが使用中である場合は、管理アイデンティティドメインを設定しなければならず（構成検証によって検査される）、すべてのパーティションがプライマリアイデンティティドメインを構成しなければならない（構成検証によって検査される）。いずれかのレルムに対してシステム内で構成されたすべてのロールマッピング、認可、資格証明マッピング、および監査プロバイダが、アイデンティティドメイン認識プロバイダのマーカーインターフェイスを実現しなければならない（レルム始動時に検査される）。認証プロバイダは、デフォルト／管理レルム、および、適切なアイデンティティドメインからのユーザを認証できる各パーティションのレルムにおいて構成される（これは構成検証によって検査されない）。このように、図７に示される例を用いると、レルムＡ　７５０は、ＰＩＤＤ　Ａ　７５４からのユーザを認証できる認証プロバイダを構成しなければならず、レルムＢ　７６０は、ＰＩＤＤ　Ｂ　７６４からのユーザを認証できる認証プロバイダを構成しなければならない。同様に、デフォルトレルム７０２は、ＡＩＤＤ７０４からのユーザを認証できる認証プロバイダを構成しなければならない。

【００８０】

アイデンティティドメイン認識プロバイダの使用はまた、セキュリティ構成オブジェクトのアイデンティティドメイン認識プロバイダに必要な属性を設定することによって強制できる。この属性を真に設定すると、たとえシステム内でアイデンティティドメインが構

成されていなくても、すべてのロールマッピング、認可、資格証明マッピング、および監査プロバイダがアイデンティティドメイン認識プロバイダインターフェイスをサポートするように強制することになる。

【００８１】

セキュリティプロバイダ（Security Provider：ＳＳＰＩ）インターフェイスプロバイダは、パーティション分割された環境内で、またはより正確にはアイデンティティドメインがシステム内に構成されている環境内で正しく機能するためには、アイデンティティドメイン認識プロバイダでなければならない。たとえば、アイデンティティドメインを理解しない認可プロバイダは、名前が同一であるがアイデンティティドメインが異なる２人のユーザを正しく区別できないので、有効な認識判定を下すことができない。アイデンティティドメインが使用中のとき、または、アイデンティティドメイン認識プロバイダに必要な属性が真に設定されているとき、以下の種類のプロバイダはすべて、アイデンティドメイン認識プロバイダでなければならずかつ以下のマーカーインターフェイスを実現しなければならない。

【００８２】

・ロールマッピング
・認可
・資格証明
・マッピング監査

アイデンティティドメイン認識プロバイダは、マーカーインターフェイスを実現するとともに、ユーザおよびグループプリンシパルを、これらがアイデンティティドメイン値を有するときに、適切に処理する。これは、同等性（equality）を検査するときにアイデンティティドメインを説明すること、および、アイデンティティドメインを説明するマップキーを構成することを含む。アイデンティティドメイン認識はまた、以下の認可に関するセクションで説明するように、リソース所有権を適切に説明しなければならない。

【００８３】

認証

先に述べたように、アイデンティティドメイン（ＩＤＤ）は、ユーザおよびグループのための論理的な名前空間である。アイデンティティドメインは、さまざまなユーザの組を特定し区別するために用いられる。アイデンティティドメインは、異なるパーティションのユーザを区別するのに必要であり、かつ、リソースの所有権の帰属先にするのに必要である。これらはしたがって、あるパーティションを別のパーティションから区別できるロールマッピングおよび認可ポリシーのため、および、ユーザまたはリソースを異なるパーティションから区別しなければならないその他のサービスのための、実現技術である。したがって、ユーザが認証されるときには、特定のアイデンティティドメインについて認証されることが必要である。一例として、認証メカニズムは、パーティションＡのユーザJohn Smithを、パーティションＢのユーザJohn Smithから区別しなければならない。このように、ユーザは、ユーザ名とユーザが属するアイデンティティドメインについて認証される。

【００８４】

このシステムが提供するセキュリティサービスは、「レルムベース」または「非レルムベース」のサービスとして特徴付けることができる。レルムベースのサービスは、構成がセキュリティレルムオブジェクトによって表わされたセキュリティ「レルム」に対して構成されたサービスである。認証は、レルムオブジェクトに対して構成されたレルムベースのセキュリティサービスである。

【００８５】

本発明の実施形態において、システムは、アイデンティティドメインを利用するマルチテナント環境におけるプリンシパルの認証を提供するために与えられる。アイデンティティドメインを適切に現わす基本的な種類（コールバック、プリンシパル）が与えられる。コンテナおよびアプリケーションがユーザのアイデンティティドメインを特定できるようにするアプリケーションプログラミングインターフェイスが与えられる。加えて、サービ

スおよびプロバイダは、アイデンティティドメインに対する認証および結果として得られるサブジェクトおよびプリンシパルの適切な処理をサポートする特徴を含む。

【００８６】

アイデンティティドメインを利用するマルチテナント環境において、プリンシパルは、ユーザ名と、ユーザが登録されたアイデンティティドメインによって、規定される。インターフェイスは、アイデンティティドメイン情報を保持するプリンシパルを特定し、関連付けられたアイデンティティドメインを得るための方法を宣言する。クラスを用いてこのインターフェイスを実現し、アイデンティティドメイン情報を保持する。プリンシパルの名前およびアイデンティティドメイン双方が比較に用いられる。プリンシパルの署名は、名前およびアイデンティティドメインフィールドいずれもカバーする。プリンシパルファクトリクラスは、アイデンティティドメイン値を有するユーザおよびグループプリンシパルの作成をサポートする。

【００８７】

２つのコールバック、すなわちアイデンティティドメインユーザコールバックおよびアイデンティティドメイングループコールバックが与えられる。これらのコールバックはそれぞれ、関連付けられたアイデンティドメインを有するユーザ名を表わすことができ、各々がアイデンティティドメインに関連付けられている一組のグループ名を表わすことができる。新たなコールバックの処理をサポートするためにコールバックハンドラが与えられる。コールバックおよびコールバックハンドラはともに、アプリケーションおよびコンテナが、認証しているユーザのアイデンティドメインを指定できるようにし、かつ、認証サービスおよびプロバイダが、認証中にユーザのアイデンティティドメイン情報にアクセスできるようにする。

【００８８】

ユーザの資格証明をコールバックハンドラとみなすアプリケーションプログラミングインターフェイスは、関連付けられたアイデンティティドメインを有するユーザ名を表わすことができるようにするアイデンティティドメインユーザコールバックを処理できるコールバックハンドラを用いて、アイデンティティドメインを容易にサポートすることができる。コールバックの代わりにユーザ名ストリングを用いるレガシー認証ＡＰＩは、アイデンティティドメインユーザコールバックおよびアイデンティティドメイングループコールバックをサポートするためにコールバックハンドラのアーギュメント（argument）を使用できるように、修正される。

【００８９】

サーブレットコンテナ（図３参照）の形態のログインの実現は、ログインフォームがユーザのアイデンティティドメインを収集し（そうでなければ決定し）それを標準ユーザおよびパスワードとともにコンテナに送ることができるよう、新たなアイデンティティドメインパラメータをサポートするために拡張される。よって、たとえば、テナントＡの環境３２０にアクセスするユーザは、ユーザＩＤとパスワードを要求するログオンフォームが与えられるであろう。サーブレットコンテナは、デフォルト値（すなわち現在のパーティションのプライマリアイデンティティドメイン値）を与える。よって、たとえば、ログインフォームがテナントＡの環境　３２０に与えられているので、ログインデータは、パーティションＡ　２５０のＰＩＤＤ　Ａ　７５４（図７参照）に自動的に関連付けられるであろう。ユーザはアイデンティティドメインを指定する必要はないであろう。しかしながら、異なるアイデンティティドメイン値が指定された場合、これは、コンテナから与えられたデフォルト値を無効にする。しかしながら、パーティションＡのログオンフォームは、パーティションに対してデフォルトアイデンティドメイン以外のアイデンティティドメインを選択できるように構成されていない場合がある。

【００９０】

コンテナ認証は、ストリングユーザ名パラメータを取るいくつかのアプリケーション認証インターフェイスとともに、パーティション分割された環境においてコールされたときに自動的にアイデンティドメインを提供する。この挙動は、パーティションのために設定

されたプライマリアイデンティティドメイン値に依拠し、レガシーアプリケーションがパーティションにおいて実行されているときに正しく認証できることを保証する。アプリケーションに変更は不要である。デフォルトアイデンティティドメインの挙動が適切に機能するためには、すべてのプロトコル／コンテナに対してパーティションが弁別されて実行コンテキストで利用できるようになるまで、コンテナ認証を試みることはできない。

【００９１】

アイデンティドメインを利用するマルチテナントシステムにおいてサブジェクトを認証するための処理モデルは、従来の認証機能と同様である。よって、従来の認証システムおよび方法を、その他の特徴と組合わせて利用することにより、認証プロバイダがアイデンティティドメイン認識プロバイダであることを保証してもよい。認証プロバイダは、自身が認証できるアイデンティティドメインを認識しなければならず、その他のアイデンティティドメインの認証要求は無視しなければならない。たとえば、「Acme」アイデンティティドメインを認証するように構成されたプロバイダは、Acmeユーザの認証要求のみに応じその他のアイデンティドメインのユーザは無視する必要がある。典型的には、サポートされるアイデンティティドメインのリストはプロバイダの構成から得られるので、アドミニストレータは、システムから可視であるアイデンティティドメインを支配するが、これは、処理モデルの必要条件ではない。１つのアイデンティティドメインまたは多数のアイデンティティドメインについて１つのプロバイダが認証を行なってもよく、もしその機能がありそのように構成されているのであれば、アイデンティティドメインがないユーザを認証してもよい。各々が１つ以上のアイデンティティドメインを認証する複数の認証プロバイダを構成できる。

【００９２】

図７を参照して、たとえば、レルムＡ　７５０、レルムＢ　７６０、およびデフォルトレルム７０２は、それぞれの認証プロバイダを構成し得る。示されているように、レルムＡ　７５０は認証プロバイダＡＵＴＨ　Ａ　７５８を構成し、レルムＢ　７６０は認証プロバイダＡＵＴＨ　Ｂ　７６８を構成し、デフォルトレルム７０２は、認証プロバイダＡＵＴＨ　７０８を構成する。または、たとえば、アイデンティティドメイン情報を受けて用いることにより正しいアイデンティティドメインに対してユーザが確実に認証されるようにする、１つの認証プロバイダを構成してもよい。

【００９３】

ＡＵＴＨ　Ａ　７５８およびＡＵＴＨ　Ｂ　７６８等の認証プロバイダは、これらがサポートするアイデンティティドメインを特定する。認証プロバイダは、構成された認証プロバイダインスタンス１つ当たり１つのみのアイデンティティドメインをサポートする。ＳＳＰＩインターフェイス－アイデンティティドメインオーセンティケーター－は、各認証プロバイダが、それがサポートするアイデンティティドメインを構成／宣言できるようにする。アイデンティティドメインユーザを認証できる認証プロバイダは、インターフェイスを実現しアイデンティドメインコールバックを処理する。

【００９４】

認証プロバイダは、Ｊａｖａ認証および認可サービス（Java Authentication and Authorization Service：ＪＡＡＳ）処理モデルを利用して実現できる。ＪＡＡＳは２つの目的のために使用できる。１つの目的は、Ｊａｖａコードがアプリケーション、アプレット、ｂｅａｎ、またはサーブレットとして実行されているか否かに関わらず、このＪａｖａコードを現在実行しているのは誰かを、ユーザの認証によって確実にかつ安全に判断するためであり、もう１つの目的は、実行されるアクションを行なうのに必要なアクセス制御の権利（許可）をユーザが有することを、ユーザの認証によって保証するためである。ＪＡＡＳ認証は、プラガブルなやり方で行なわれる。これによって、Ｊａｖａアプリケーションを、基礎となる認証技術から独立した状態に保つことができる。新たなまたは更新された技術を、アプリケーションそのものを修正する必要なく、プラグすることができる。使用すべき特定の認証技術の実現を、実行時に判断する。この実現はログイン構成ファイルにおいて指定される。

【００９５】
認証プロバイダは、レルム上で構成された順序でコールされ、各認証プロバイダに対して設定されたＪＡＡＳ制御フラグに適合する。上記処理モデルの場合、アイデンティティドメインオーセンティケータにとって最も適切な構成は通常、各プロバイダが、ａ）アイデンティティドメインの離散的で重複しない組みを処理し、ｂ）「十分な」ＪＡＡＳ制御フラグを用いて構成されることである。その他の構成は、可能であるが、ＪＡＡＳ制御フラグと、ターゲットアイデンティティドメインに基づいて認証要求を処理するか無視するか選択できる認証プロバイダとの対話に照らして、慎重に検討しなければならない。
【００９６】
図８は、一実施形態に従い、複数のアイデンティティドメインを有するマルチテナント環境におけるログインモジュールによって実現される方法を示す。複数のアイデンティティドメインを有するマルチテナント環境における認証プロバイダのログインモジュールは、図８に示される方法を用いて、コールバックを処理しようと試みることにより認証要求を処理するか否か判断する。ステップ８１０において、レルム上に構成された第１の認証プロバイダにコールバックが与えられる。ステップ８１２において、認証プロバイダは、アイデンティティドメインユーザコールバックを処理しようと試みる。ステップ８１４において、アイデンティティドメインユーザコールバックが返された場合、認証プロバイダはそこからユーザのアイデンティティドメインを得る。ステップ８１６において、ユーザのアイデンティティドメインがこのオーセンティケータによって処理される場合、ユーザを認証する。パスワードコールバックを用いてユーザのパスワードを取得する。ステップ８１８において、ユーザのアイデンティティドメインがオーセンティケータによって処理されない場合は、「偽」を返して要求を無視する。ステップ８２０において、認証プロバイダが、アイデンティティドメインがないユーザを処理するように構成されていれば、認証プロバイダは、名前のコールバックを処理しようと試みる。ステップ８２２において、名前が発見されなければ、「偽」を返して要求を無視する。ステップ８２４において、名前が発見された場合は、ユーザを認証する。パスワードコールバックを用いてユーザのパスワードを取得する。ステップ８２６において、認証プロバイダがユーザを認証できなかった場合、システムは、レルム上に構成された次の認証プロバイダを（もしあれば）試す。第１のログインモジュールが要求を無視した場合、この要求は次にその要求を処理しようと試みる別のログインモジュールに渡される。認証プロバイダは、要求が処理されるまたは失敗に終わるまで、レルム上で構成された順序でコールされる。
【００９７】
アイデンティティアサーション（表明）プロバイダは、ユーザ名およびパスワード以外のトークンまたは資格証明、たとえば、ＳＡＭＬアサーション、ケルベロス（Kerberos）チケット、またはクッキーベースのセッショントークンに基づいてユーザを認証する認証プロバイダである。これらはコールバックを処理しない。代わりに、これらはユーザのトークンを検証しその内容を検査することによって、ユーザのアイデンティティ、たとえばユーザ名、グループメンバ構成等を求める。取得されたユーザ情報は、コールバックハンドラの形態で返され、これは最終的に、ユーザのサブジェクトに対する適切なプリンシパルを実装するためにログインモジュールチェーンに送られる。
【００９８】
その他の認証プロバイダと同様、パーティション分割された環境で使用されるアイデンティティアサーションプロバイダは、アイデンティティドメイン認識プロバイダとなるように構成され、そのアイデンティティドメインサポートを宣言するために適切なアイデンティティドメインオーセンティケータを実現する。これらは、構成からまたはアサートされたトークンから、ユーザに対して（かつ場合によってはアサートされたいずれかのグループに対して）どのようなアイデンティティドメインをアサートすべきかを判断することができる。アイデンティティドメインがあるユーザをアサートするとき、返されたコールバックハンドラは、アイデンティティドメインユーザコールバックを処理するように、また任意でアイデンティドメイングループコールバックを処理するように構成される。

【００９９】
ログインモジュールは、好ましくはプリンシパルファクトリを用いてプリンシパルを作成する。ファクトリは、アイデンティティドメイン情報を有するプリンシパルを作成する方法を提供する。プリンシパルを直接インスタンス化する場合は、アイデンティティドメインを、適切なコンストラクタ（constructor）またはセッター（setter）方法を用いて提供しなければならない。プリンシパル検証サービスは、認証時にすべてのプリンシパルに署名し、システムに入ってきたプリンシパルの署名を検証することにより、これらが最初にローカルドメインによって認証されたことを保証する。実際の署名／検証は、プリンシパルバリデータ（Validator）プロバイダに委任される。
【０１００】
プリンシパル検証は、レルムベースのサービスである。このことは、異なるレルムが異なるやり方でプリンシパルを検証できることを意味する。そうすることによって、あるレルムによって認証されたプリンシパルを、他のレルムによって認証されたプリンシパルから区別する。この挙動は、多くのシナリオにおいて場合によっては有用である可能性があるものの、これは必須ではない。たとえば、マルチテナントシステムは、ドメイン上で構成された１つの署名キー、すなわち「ドメイン資格証明」を用いることができ、したがって、範囲としてはドメイン全体に渡る。よって、プリンシパルは、どのレルムがこれらのプリンシパルを認証したかに関わらず、同一のキーで署名される。これに代わる実施形態においては、代わりに異なる署名キーをそれぞれ異なるレルムが使用してもよい。
【０１０１】
プリンシパルおよびアイデンティティアサーションキャッシュ。各レルムは、プリンシパルキャッシュとアイデンティティアサーションキャッシュを保持している。１番目は、同一のプリンシパルに繰返し署名するのを避けるために使用される署名付きプリンシパルのキャッシュである。なぜなら、署名は比較的コストが高い作業であるからである。キャッシュキーは、好ましくはユーザ名およびアイデンティティドメインであり、異なるアイデンティティからのプリンシパルが混乱しないようにする。２番目は、アイデンティティアサーションから生じたサブジェクトのキャッシュである。このキャッシュは、ＬＤＡＰ（またはその他のユーザ記憶域）に繰返し問合せることで同一のユーザアイデンティティが繰返しアサートされたときにユーザ情報（グループメンバ構成等）をフェッチするのを避けるために、実現される。プリンシパルキャッシュと同様、キャッシュキーは、好ましくはユーザ名およびユーザのアイデンティティドメインであり、異なるアイデンティティドメインからのプリンシパルが混乱しないようにする。
【０１０２】
認可
一実施形態において、マルチテナントサーバ環境は、ロールマッピングサービスおよび認可サービス双方で構成された認可サブシステムによってサポートされる、ロールベースのアクセス制御（role-based access control：ＲＢＡＣ）モデルを実現する。上記２つのサービスは概ね独立しているが、認可サービスは、ロールマッピングサービスをコールすることにより、特定の識別されたリソースについて認可の判断を下すときにユーザのロールを判断する。ロールマッピングサービスおよび認可サービスはいずれも、実際のロールマッピングおよび認可機能を提供するセキュリティサポートプロバイダに委任される。
【０１０３】
このシステムが提供するセキュリティサービスは、「レルムベース」または「非レルムベース」のサービスとして特徴付けることができる。レルムベースのサービスは、構成がセキュリティレルムオブジェクトによって表わされるセキュリティ「レルム」に対して構成されたサービスである。認可は、レルムオブジェクト上に構成されたレルムベースのセキュリティサービスである。
【０１０４】
セキュリティサポートプロバイダは、ロールマッピングおよび認可ルールに使用できる多数のポリシー「述部」を実現する。これらの述部は、特定の条件が満たされた場合に、

たとえば、サブジェクトにおけるプリンシパルが特定のユーザまたはグループ名と一致する場合または特定のＩＰアドレスから発生した要求と一致する場合、ロールを与えるかまたはリソースへのアクセスを直接与える。述部は、論理ＡＮＤ、ＯＲ、およびＮＯＴの意味論を用いて組合わせることができる。ＲＢＡＣに対するサポートは、ユーザが指定されたロールを有する場合にアクセスを与えるために認可ルールで使用されるロール述部によって与えられる。ＲＢＡＣモデルにおいて、アクセスの主要な決定要素は、ユーザが所与のロールを有するか有しないかである。

【０１０５】

ほとんどの場合ロールマッピングに使用される述部は、特定のグループ内のメンバ構成に基づいてロールを与えるグループ述部である。WebLogic管理ロールマッピングは、この述部を用いて、たとえば「管理」ロールを「アドミニストレータ」グループのメンバに与える。（特定のユーザ名と一致する）ユーザ述部を使用することはあまり一般的ではない。一般的には、その他の述部、たとえばＩＰアドレスおよび時刻を用いて、コンテキストに基づいてロールをさらに制限する。たとえば「銀行窓口」ロールを「窓口」グループのメンバに、銀行の営業時間中のみ、与えてもよい。

【０１０６】

パーティション分割された環境において、アクセス制御はさらに他の特質を示す。このとき、ユーザ、グループ、およびリソースは、パーティションに関連付けられ、アクセス判断は、ユーザのアイデンティティドメインと、リソースが属するパーティション双方を、ユーザのアイデンティティおよびロールに加えて考慮する。２つの新たな特徴／挙動によって、パーティション認識アクセス制御が可能になる。アイデンティティドメインを考慮してユーザおよびグループを比較する第１の新たな述部が与えられる。リソースのパーティション所有権を表わすために第２のアイデンティティドメインが使用される。このように、ユーザおよびリソース双方がアイデンティティドメインに関連付けられる。よって、認可の判断を、アイデンティティドメインに関連付けられたリソースにアクセスするために正しいアイデンティドメインにおける正しいロールをユーザが有するか否かを判断する認可プロバイダによって、行なうことができる。

【０１０７】

リソースは実際、たとえば図３に示されるアイデンティドメインではなく、パーティションに属する。しかしながら、リソース所有権をアイデンティティドメインとして表わすことは好都合である。なぜなら、それによって、認可の判断を下すときに、リソースアイデンティドメインとユーザ／グループアイデンティドメインとが直接比較されるからである。単純なストリング比較によって、２つのアイデンティドメインが一致するか否かを判断することができる。パーティションをアイデンティティドメインに対して比較することによって「一致」を判断する手法は、限定的な命名規則、マッピングテーブル、または双方が必要であろう。これはまた、実行時の効率が低いであろう。パーティションのためのアイデンティティドメインマッピングは、パーティションオブジェクトのプライマリアイデンティティドメイン属性上に構成される。例として図７のＰＩＤＤ　Ａ　７５４およびＰＩＤＤ　Ｂ　７６４参照。アイデンティドメインを用いてリソース所有権を表わす利点としてさらに他の利点は、モデルをパーティションに結び付けないことであり、よって、アイデンティティドメインを、パーティション分割されていない環境におけるリソース認可の制御に用いることもできる。

【０１０８】

以下の述部は、アイデンティティドメイン認識ユーザおよびグループマッチングをサポートする。これらの述部は、ロールマッピングにも認可ルールにも利用できるが、主として、WebLogicのＲＢＡＣモデルに適合するロールマッピングに使用されると予想される。

【０１０９】

【表１】

述部	ルール式	説明
IDDUser(user, idd)	User.name == user && User.idd == idd	指定されたアイデンティティドメインの指定されたユーザと一致。既存のユーザ述部と同様であるが、ユーザが指定されたアイデンティティドメインのユーザである場合に限り一致。
IDDGroup(group, idd)	Group.name == group && Group.idd == idd	指定されたアイデンティティドメインの指定されたグループと一致。既存のグループ述部と同様であるが、グループが指定されたアイデンティティドメインのグループである場合に限り一致。
OwnerIDDUser(user)	User.name == user && User.idd == Resource.idd	ユーザのアイデンティティドメインがリソース所有者のアイデンティティドメインと一致する場合、指定されたユーザと一致。
OwnerIDDGroup(group)	Group.name == group && Group.idd == Resource.idd	グループのアイデンティティドメインがリソース所有者のアイデンティティドメインと一致する場合、指定されたグループと一致。
AdminIDDUser(user)	User.name == user && User.idd == Admin.idd	ユーザのアイデンティティドメインが管理アイデンティティドメインと一致する場合、指定されたユーザと一致。
AdminIDDGroup(group)	Group.name == group && Group.idd == Admin.idd	グループのアイデンティティドメインが管理アイデンティティドメインと一致する場合、指定されたグループと一致。

【０１１０】

図９は、一実施形態に従い、複数のアイデンティティドメインを有するマルチテナント環境において実現される認可サブシステムを示す。図９に示されるように、アイデンティティドメイン９１２に関連付けられたユーザ９１０は、アイデンティティドメイン９１６に関連付けられたリソース９１４にアクセスしようと試みる。ラッパー（wrapper）クラスを用いてこれらの値を指定する。ラッパークラスは、必要であれば現在のパーティションを求め、与えられた所有権の値を、ロールマッピング／認可プロバイダによる使用のためにリソースアイデンティティドメインの値に変換する。ラッパーはまた、正しいリソースアイデンティティドメインの値が与えられることを保証する。リソースへのアクセスは、ロールマッピングサービス９２０と認可サービス９３０とを含む認可サブシステム９００によって制御される。ロールマッピングサービス９２０および認可サービス９３０は、実際のロールマッピングおよび認可機能を提供するセキュリティサポートプロバイダ９４０に委任される。セキュリティサポートプロバイダ９４０は、ユーザがコールされたリソースにアクセスするためのロールか否かの判断だけではなく、ユーザＩｄＤ９１２がリソースＩｄＤ９１６と一致するか否かの判断を可能にするルールを提供する述部９４２を実現する。リソースをコールするプロセスは、アクセスが要求されたリソースの所有権を特定するリソースアイデンティティドメイン値を提供する。認可プロバイダは、このようにして、ユーザアイデンティティ、ユーザアイデンティティドメインおよびユーザロールを

、リソースおよびリソースアイデンティティドメイン所有権と比較できるようにされて、アイデンティティドメイン認識述部を用いて認可判断を下す。

【０１１１】

パーティションセキュリティは、１つの新たな方法と、３つの新たなコンテキストハンドラ属性と、コンテキストハンドララッパークラスを、認可およびロールマッピングＡＰＩに加える。ロールマネージャインターフェイスに対する新たなUser In Role()法によって、呼び出し元は、ユーザが特定のロールか否かを直接判断することができ、リソースおよびコンテキストハンドラをクエリに対する入力として送る。３つの新たなコンテキストハンドラ属性によって、呼び出し元は、認可およびロールマッピングＡＰＩをコールするときにリソースのリソース所有権情報を指定することができる。RESOURCE_PARTITION（パーティション名）は、リソースを所有しているパーティションを示す。RESOURCE_OWNERSHIP_DEFAULTは、呼び出し元のコンテキストによって決められた現在のパーティションが、リソースを所有していることを、示す。RESOURCE_IDENTITY_DOMAINによって、呼び出し元は、パーティションアイデンティティドメインをリソース所有者として指定することができる。新たなコンテキストハンドララッパークラス、リソースＩＤＤコンテキストラッパーを、呼び出し元が用いて、既存のコンテキストハンドラインスタンスを、適切なアイデンティティドメイン所有権情報で修飾することができる。

【０１１２】

複数のレルムの説明で述べたように、コンテナは、実行時に正しいレルムに要求を向けるプロキシサービスを通してセキュリティサービスと対話する。認可マネージャおよびロールマネージャの場合、プロキシは、リソースタイプと所有権を考慮する論理を用いて正しいレルムを選択する。現在のパーティションが所有するリソース、たとえばアプリケーションリソースについては、「ローカルな」レルムが使用される。システムリソースについては、たとえばデフォルト／グローバルレルムが使用される。次に、リソースタイプに応じて、ローカルパーティションのレルムについてまたはデフォルト／グローバルレルムについて認可／ロールマッピングサービスをコールするか否かを判断する。

【０１１３】

アイデンティティドメイン認識述部を使用することにより、範囲指定されたロール、すなわち特定のアイデンティティドメインのコンテキストにおいてのみ有効なロールを可能にする。所有者ＩＤＤグループ述部を用いることによって、パーティション管理ロールを、アクセス要求されたリソースを所有しているアイデンティティドメインのパーティションアドミニストレータグループのメンバであるユーザに与えることができる。（リソースはロールクエリおよび認可クエリのパラメータである。）テナント管理ロールも同様に与えることができる。以下の新たなロールポリシーは、パーティション分割されたドメインのレルムに対してプロビジョニングされる。なお、テナントベースのロールマッピングは、それが必要な階層化されたコンポーネントによってプロビジョニングされることができる。

【０１１４】

この発明は、この開示の教示に従ってプログラミングされた１つ以上のプロセッサ、メモリおよび／またはコンピュータ読取可能記憶媒体を含む、１つ以上の従来の汎用または特化型デジタルコンピュータ、コンピューティング装置、マシン、またはマイクロプロセッサを使用して都合よく実現されてもよい。ソフトウェア技術の当業者には明らかであるように、この開示の教示に基づいて、適切なソフトウェアコーディングが、熟練したプログラマによって容易に準備され得る。

【０１１５】

実施形態によっては、本発明は、本発明のプロセスのうちいずれかを実行するためにコンピュータをプログラムするのに使用できる命令が格納された非一時的な記録媒体または（１つまたは複数の）コンピュータ読取可能な媒体であるコンピュータプログラムプロダクトを含む。この記録媒体は、フロッピー（登録商標）ディスク、光ディスク、ＤＶＤ、ＣＤ－ＲＯＭ、マイクロドライブ、および光磁気ディスクを含む、任意の種類のディスク

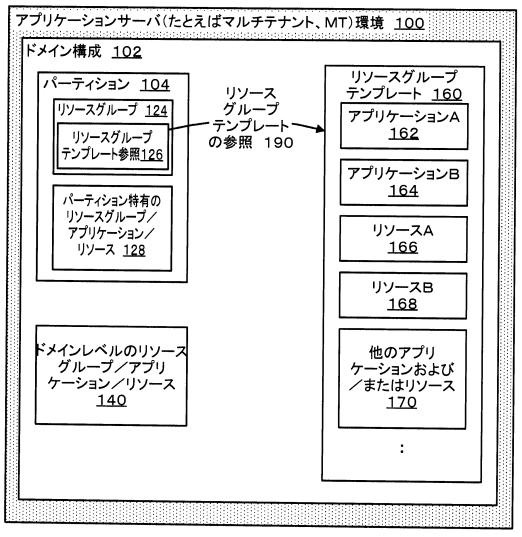
、ＲＯＭ、ＲＡＭ、ＥＰＲＯＭ、ＥＥＰＲＯＭ、ＤＲＡＭ、ＶＲＡＭ、フラッシュメモリデバイス、磁気もしくは光カード、ナノシステム（分子メモリＩＣを含む）、または、命令および／もしくはデータを格納するのに適した任意の種類の媒体もしくはデバイスを含み得るものの、これらに限定されない。

【０１１６】

本発明のこれまでの記載は例示および説明を目的として提供されている。すべてを網羅するまたは本発明を開示された形態そのものに限定することは意図されていない。当業者には数多くの変更および変形が明らかであろう。実施の形態は、本発明の原理およびその実際の応用を最もうまく説明することによって他の当業者がさまざまな実施の形態および意図している特定の用途に適したさまざまな変形を理解できるようにするために、選択され説明されている。本発明の範囲は添付の特許請求の範囲およびそれらの同等例によって規定されるものと意図されている。

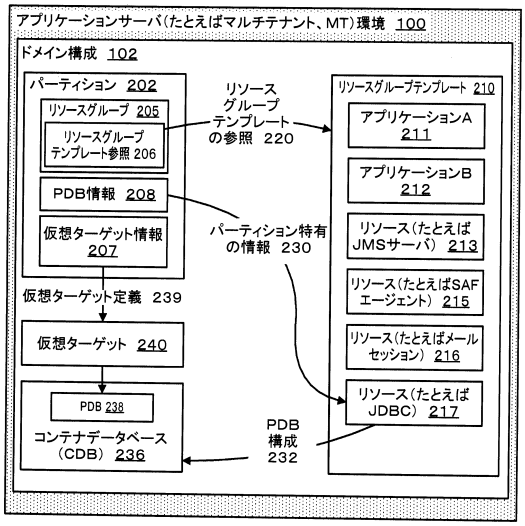
【図１】

Figure 1



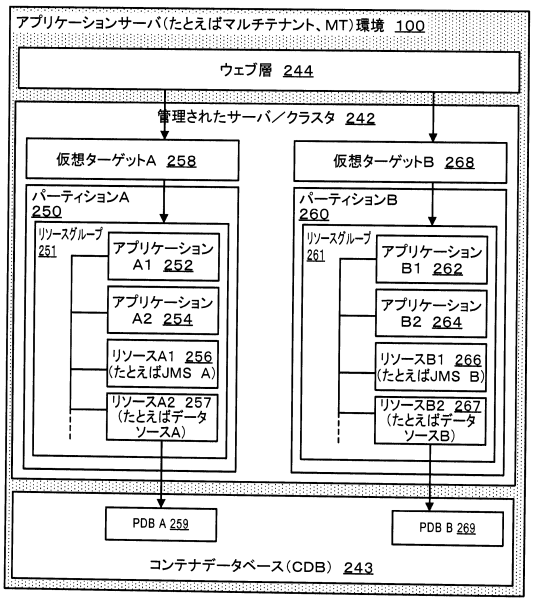
【図２】

Figure 2



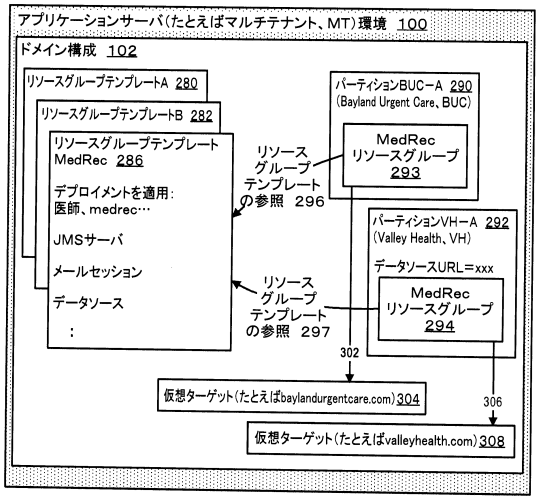
【図３】

Figure 3



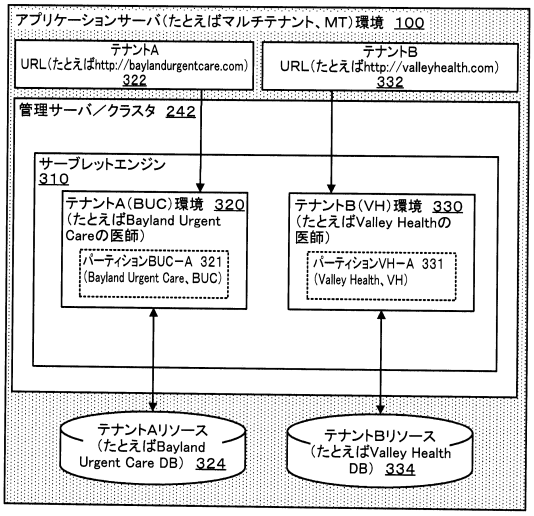
【図４】

Figure 4



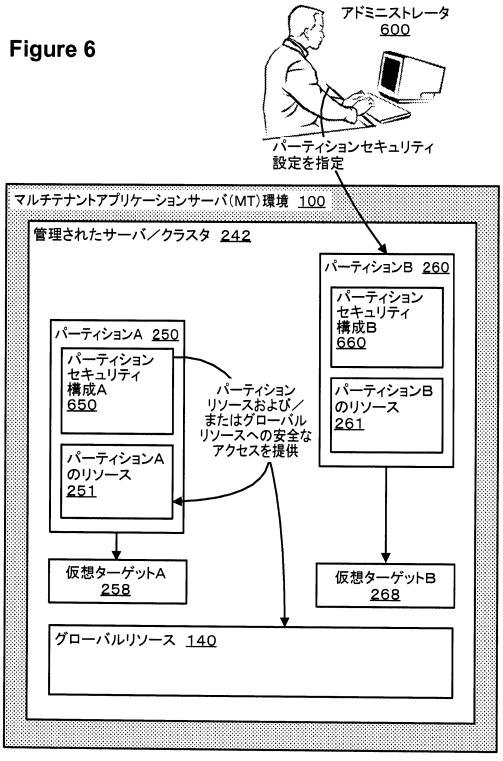
【図５】

Figure 5



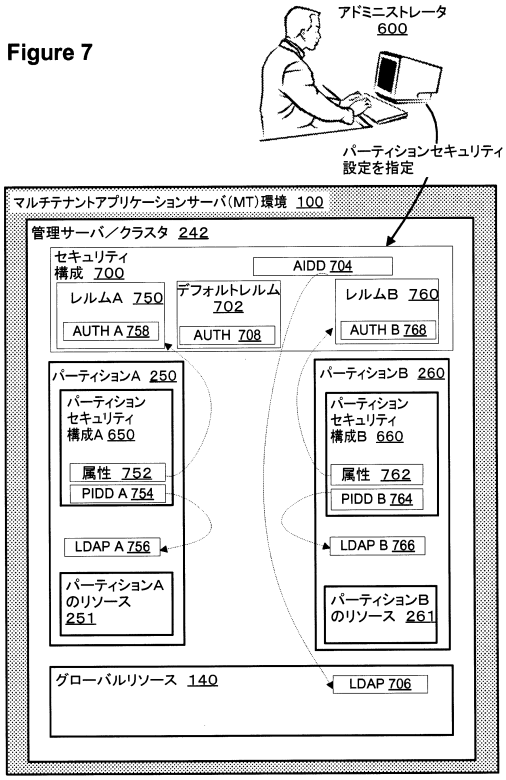
【図６】

Figure 6



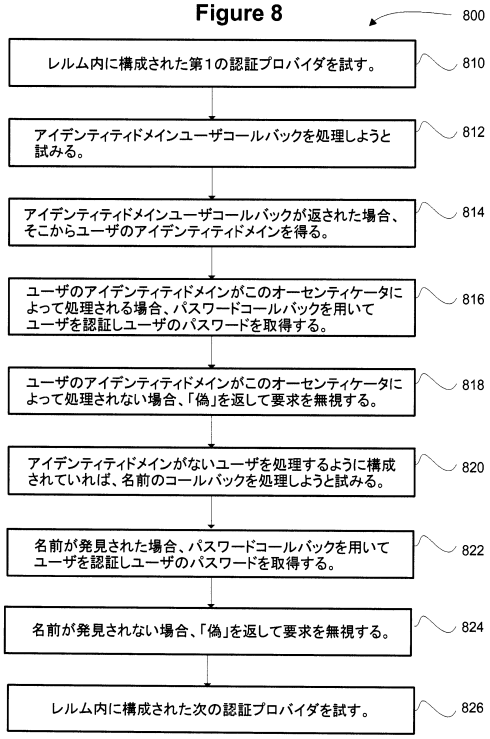
【図７】

Figure 7



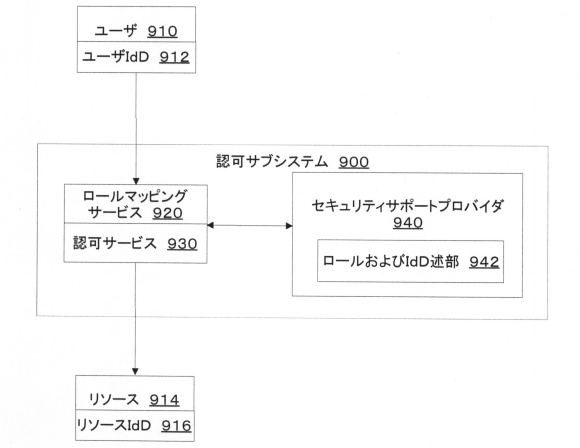
【図８】

Figure 8



【図９】

Figure 9



フロントページの続き

(72)発明者　ペレス，クレイグ
アメリカ合衆国、９４０６５　カリフォルニア州、レッドウッド・シティー、オラクル・パークウェイ、５００
(72)発明者　ガイ，デイビッド
アメリカ合衆国、９４０６５　カリフォルニア州、レッドウッド・シティー、オラクル・パークウェイ、５００
(72)発明者　バウアー，ピーター
アメリカ合衆国、０３０４９　ニュー・ハンプシャー州、ホリス、サウス・ゲート・ロード、２０
(72)発明者　リ，ジュアン
アメリカ合衆国、０２４９４　マサチューセッツ州、ニーダム、デイビッド・ロード、２２
(72)発明者　タンシル，ジェフ
アメリカ合衆国、９４０６５　カリフォルニア州、レッドウッド・シティー、オラクル・パークウェイ、５００
(72)発明者　スリラマデシカン，クリシュナ
アメリカ合衆国、９５０１４　カリフォルニア州、クパチーノ、パリッシュ・プレイス、１０１６３

審査官　上島　拓也

(56)参考文献　国際公開第２０１４／０３９７７２（WO，Ａ１）
国際公開第２０１４／０３９８８２（WO，Ａ１）
国際公開第２０１４／０２２３２３（WO，Ａ１）
米国特許出願公開第２０１３／０１１１５５８（ＵＳ，Ａ１）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　２１／６２
Ｇ０６Ｆ　２１／５３