Oracle® Application Server Single Sign-On

管理者ガイド

10*g*（10.1.4.0.1）

部品番号 : B31505-01

2006 年 9 月

ORACLE®

Oracle Application Server Single Sign-On 管理者ガイド , 10*g*（10.1.4.0.1）

部品番号 : B31505-01

原本名 : Oracle Application Server Single Sign-On Administrator's Guide, 10*g* (10.1.4.0.1)

原本部品番号 : B15988-01

原本協力者 : Nina Wishbow, Gaurav Bhatia, Kamalendu Biswas, Margaret Chou, Lee Cooper, Mike Hwa, Ganesh Kirti, Peifung Eric Lam, Andrew Maywah, Mark Nelson, Saurabh Shrivastava, Tim Willard

目次

2 基本的な管理

3 ディレクトリ対応 Single Sign-On

10 アプリケーション・サービス・プロバイダに対するサポートの有効化

11 Single Sign-On Server の監視

12 配置固有ページの作成

13 Oracle Identity Federation との統合

14 サード・パーティのアクセス管理システムとの統合

15 データのエクスポートとインポート

A OracleAS Single Sign-On のトラブルシューティング

図一覧

表一覧

はじめに

『Oracle Application Server Single Sign-On 管理者ガイド』では、Oracle Application Server（OracleAS）のユーザー認証を管理するための概要と手順について説明します。このマニュアルは、UNIX および Windows プラットフォームを対象にしています。

「はじめに」の項目は次のとおりです。

対象読者

『Oracle Application Server Single Sign-On 管理者ガイド』は、次のユーザーを対象にしています。

- OracleAS の認証の構成および管理を担当する管理者。
- 認証メカニズムに OracleAS Single Sign-On を使用した機能の開発者。特にそれらを mod_osso（Oracle HTTP Server 上の認証モジュール）と統合する開発者。
- OracleAS Single Sign-On を使用して Web アプリケーションへのアクセスを保護する方法に関心のあるユーザー。

このマニュアルの読者は、OracleAS の基礎知識があり、リリース 10.1.3 をインストールしている、またはインストールできることを前提にしています。

ドキュメントのアクセシビリティについて

オラクル社は、障害のあるお客様にもオラクル社の製品、サービスおよびサポート・ドキュメントを簡単にご利用いただけることを目標としています。オラクル社のドキュメントには、ユーザーが障害支援技術を使用して情報を利用できる機能が組み込まれています。HTML 形式のドキュメントで用意されており、障害のあるお客様が簡単にアクセスできるようにマークアップされています。標準規格は改善されつつあります。オラクル社はドキュメントをすべてのお客様がご利用できるように、市場をリードする他の技術ベンダーと積極的に連携して技術的な問題に対応しています。オラクル社のアクセシビリティについての詳細情報は、Oracle Accessibility Program の Web サイト `http://www.oracle.com/accessibility/` を参照してください。

ドキュメント内のサンプル・コードのアクセシビリティについて

スクリーン・リーダーは、ドキュメント内のサンプル・コードを正確に読めない場合があります。コード表記規則では閉じ括弧だけを行に記述する必要があります。しかし JAWS は括弧だけの行を読まない場合があります。

外部 Web サイトのドキュメントのアクセシビリティについて

このドキュメントにはオラクル社およびその関連会社が所有または管理しない Web サイトへのリンクが含まれている場合があります。オラクル社およびその関連会社は、それらの Web サイトのアクセシビリティに関しての評価や言及は行っておりません。

Oracle サポート・サービスへの TTY アクセス

アメリカ国内では、Oracle サポート・サービスへ 24 時間年中無休でテキスト電話（TTY）アクセスが提供されています。TTY サポートについては、(800)446-2398 にお電話ください。

関連ドキュメント

詳細は、次の Oracle ドキュメントを参照してください。

- 『Oracle Identity Management アプリケーション開発者ガイド』
- 『Oracle Internet Directory 管理者ガイド』

リリース・ノート、インストール関連ドキュメント、ホワイト・ペーパーまたはその他の関連ドキュメントは、OTN-J（Oracle Technology Network Japan）から、無償でダウンロードできます。OTN-J を使用するには、オンラインでの登録が必要です。登録は、次の Web サイトから無償で行えます。

`http://otn.oracle.co.jp/membership/`

すでに OTN-J のユーザー名およびパスワードを取得している場合は、次の URL で OTN-J Web サイトのドキュメントのセクションに直接接続できます。

`http://otn.oracle.co.jp/document/`

OracleAS Single Sign-On の開発に関する最新情報は、次のサイトから参照できます。

`http://www.oracle.com/technology/products/id_mgmt/osso/index.html`

表記規則

このマニュアルでは次の表記規則を使用します。

規則	意味
太字	太字は、操作に関連する Graphical User Interface 要素、または本文中で定義されている用語および用語集に記載されている用語を示します。
イタリック	イタリックは、ユーザーが特定の値を指定するプレースホルダ変数を示します。
固定幅フォント	固定幅フォントは、段落内のコマンド、URL、サンプル内のコード、画面に表示されるテキスト、または入力するテキストを示します。

サポートおよびサービス

次の各項に、各サービスに接続するための URL を記載します。

Oracle サポート・サービス

オラクル製品サポートの購入方法、および Oracle サポート・サービスへの連絡方法の詳細は、次の URL を参照してください。

`http://www.oracle.co.jp/support/`

製品マニュアル

製品のマニュアルは、次の URL にあります。

`http://otn.oracle.co.jp/document/`

研修およびトレーニング

研修に関する情報とスケジュールは、次の URL で入手できます。

`http://www.oracle.co.jp/education/`

その他の情報

オラクル製品やサービスに関するその他の情報については、次の URL から参照してください。

```
http://www.oracle.co.jp
http://otn.oracle.co.jp
```

注意： ドキュメント内に記載されている URL や参照ドキュメントには、Oracle Corporation が提供する英語の情報も含まれています。日本語版の情報については、前述の URL を参照してください。

OracleAS Single Sign-On の新機能

ここでは、OracleAS Single Sign-On 10*g*（10.1.4.0.1）の新機能について説明し、追加情報の参照先を示します。現在のリリースに移行するユーザーのために、以前のリリースからの情報も残しています。

このマニュアルに記載されている OracleAS Single Sign-On の新機能について、次の各項で説明します。

- フェデレーテッド認証
- カスタム（配置固有）ページの構成
- 起動用の構文の変更 OracleAS Single Sign-On
- Single Sign-On 管理グループの変更
- グローバリゼーション・サポート
- データベース・アクセス記述子（DAD）の廃止
- ロード・バランサがない場合の URL の保護
- 認証レベルに関する情報
- ログイン・ページのエラー・コード
- 認証 URL
- 複数のレルムに対する Single Sign-On Server の構成
- パートナ・アプリケーションに対する SSL の構成
- デバッグ・ログ・ファイル
- リソースを戻せない保護されたリソースへの URL
- mod_osso Cookie のセキュアな通信
- 廃止されたエラー・メッセージ

フェデレーテッド認証

- Oracle Application Server Single Sign-On と Oracle Identity Federation を使用して、フェデレーテッド認証を実装できます。フェデレーテッド・シングル・サインオンを使用すると、ユーザーは 1 つのサイトで認証されるだけで、複数の企業の Web サイトの情報にアクセスできます。Oracle Application Server Single Sign-On または Oracle Identity Federation によって保護されているリソースにアクセスするユーザーの認証メカニズムとなるように、これらの製品を構成することができます。

  関連項目： 第 13 章「Oracle Identity Federation との統合」

カスタム（配置固有）ページの構成

- シングル・サインオフ・ページでは、WWSSO_LS_CONFIGUATION_INFO$ 表が必要なくなりました。
- 外部アプリケーション用のカスタム・ログイン・ページを構成できます。

  関連項目： 12-12 ページの「配置固有ページのインストール」

起動用の構文の変更 OracleAS Single Sign-On

- OracleAS Single Sign-On を起動する構文が単純になりました。たとえば、管理ホーム・ページにアクセスするとき、以前は次の URL を入力していました。

  `http://host:port/pls/orasso`

  このかわりに、次の URL を使用できます。

  `http://host:port/sso`

  関連項目： 1-4 ページの「Single Sign-On Server へのアクセス」、2-9 ページの「管理ページへのアクセス」、4-8 ページの「パートナ・アプリケーション中間層での mod_osso の再登録」、第 7 章「SSL の有効化」、第 9 章「高度な配置オプション」、11-5 ページの「SSL 対応の Single Sign-On Server の監視」、第 12 章「配置固有ページの作成」

Single Sign-On 管理グループの変更

- この処理を実行するための手順が変更されました。

  関連項目： 2-4 ページの「Single Sign-On 管理グループの変更」

グローバリゼーション・サポート

- グローバリゼーション・サポートに関する情報が更新されています。追加情報への参照が追加されました。

  関連項目： 2-11 ページの「グローバリゼーション・サポートの構成」

データベース・アクセス記述子（DAD）の廃止

- この表は必要なくなり、削除されました。

  関連項目： 2-7 ページの「アクセス不能なサーバーのトラブルシューティング」

ロード・バランサがない場合の URL の保護

- この処理を実行するための構文が変更されました。

  関連項目： 7-5 ページの「ロード・バランシング・ルータがない場合の URL の保護」

認証レベルに関する情報

- 認証レベルに関する情報が増えました。

  関連項目： 6-3 ページの「認証レベル」

ログイン・ページのエラー・コード

- ログイン・ページのエラー・コードに関する情報が変更されました。

  関連項目： 12-8 ページの「ログイン・ページのエラー・コード」および 12-9 ページの「ログイン後のメッセージ」

認証 URL

- 認証 URL に関する情報が変更されました。

  関連項目： 12-2 ページの「Single Sign-On Server での配置固有ページの使用方法」

複数のレルムに対する Single Sign-On Server の構成

- この手順が変更されました。

  関連項目： 10-5 ページの「複数のレルムに対する Single Sign-On Server の構成」

パートナ・アプリケーションに対する SSL の構成

- パートナ・アプリケーション（OracleAS Single Sign-On を含む）に対する SSL の構成に関する情報が追加されました。

  関連項目： 7-6 ページの「SSL の構成に関する注意事項」および 7-2 ページの「自動 SSL 構成」

デバッグ・ログ・ファイル

- OC4J の実行中にデバッグ・ログ・ファイルを削除しないように注意が追加されました。

  関連項目： A-16 ページの「OracleAS Single Sign-On の障害の診断」

リソースを戻せない保護されたリソースへの URL

- URL の長さに関するブラウザの制限についての注意が追加されました。状況によっては、GET ディレクティブのかわりに POST メソッドを使用して mod_osso を構成することにより、この問題を回避できます。

  関連項目： A-2 ページの「URL が最大長を超えました」

mod_osso Cookie のセキュアな通信

- Cookie が HTTPS を使用して送信されるように、OssoSecureCookies ディレクティブの追加に関する項が追加されました。

  関連項目： 7-7 ページの「mod_osso Cookie のセキュアな通信」

廃止されたエラー・メッセージ

- OracleAS Single Sign-On の管理にアクセスしたときに許可されないエラーの項目が削除され、すべてタイプ 41400 エラーとなりました。

  関連項目： 付録 A「OracleAS Single Sign-On のトラブルシューティング」

1

OracleAS Single Sign-On

Oracle Application Server（OracleAS）Single Sign-On によって、ユーザーは、一組のユーザー名とパスワードおよびレルム ID（オプション）を使用して、他の Web アプリケーションのみでなく、OracleAS のすべての機能にログインできるようになります。

OracleAS Single Sign-On には、次の利点があります。

- 管理コストの削減

  Single Sign-On Server によって、複数のアカウントおよびパスワードをサポートする必要がなくなります。

- ログインの簡素化

  ユーザーは、アクセスするアプリケーションごとに異なるユーザー名とパスワードを使用する必要がなくなります。

- セキュリティの向上

  パスワードの入力が一度だけなので、ユーザーは、パスワードを簡単で覚えやすいものにしたり、書き留めておく必要がなくなります。

この章の項目は次のとおりです。

シングル・サインオン・システムの主要コンポーネント

OracleAS Single Sign-On が対話するコンポーネントを次の各項で説明します。

Single Sign-On Server

Single Sign-On Server は、経費報告、電子メール、福利厚生情報などのシングル・サインオン・アプリケーションに対する安全なログインを実現するプログラム・ロジックで構成されます。Single Sign-On Server のプログラム・ロジックは、Oracle Application Server データベース、Oracle HTTP Server および OC4J サーバーにあります。

Single Sign-On Server により、ユーザーは複数のアプリケーション（経費報告、電子メール、福利厚生情報など）に対して安全にログインできます。これらのアプリケーションには、パートナ・アプリケーションと外部アプリケーションの 2 つのフォームがあります。いずれの場合も、一度の認証で複数のアプリケーションにアクセスできます。

パートナ・アプリケーション

パートナ・アプリケーションとは、OracleAS Single Sign-On Server に認証機能を委譲する、Oracle Application Server アプリケーションまたは Oracle 以外のアプリケーションです。このようなアプリケーションでは、mod_osso という認証モジュールからヘッダーを受け取るので、ユーザーを再認証する必要がありません。

mod_osso モジュールにより、パートナ・アプリケーションは、ユーザーが一度 Single Sign-On Server にログインしていれば、ユーザー名とパスワードのかわりに認証済のユーザー情報を受け取ることができます。

パートナ・アプリケーションは、OracleAS Single Sign-On で認証されたユーザーにアプリケーションの使用を許可するかどうかを決定します。

パートナ・アプリケーションには、OracleAS Portal、OracleAS Discoverer および Oracle Delegated Administration Services が含まれます。

外部アプリケーション

外部アプリケーションでは、認証は OracleAS Single Sign-On Server に委譲されません。そのかわり、HTML ログイン・フォームが表示され、アプリケーションのユーザー名とパスワードが要求されます。外部アプリケーションでは、それぞれに一意のユーザー名とパスワードが要求される場合があります。HTML ログイン・フォームを使用する外部アプリケーションには、Yahoo! Mail などがあります。

ユーザーは、外部アプリケーションに最初にログインするときに、OracleAS Single Sign-On Server に自身の資格証明を取得させるように選択できます。資格証明を保存するには、最初のログイン時に、「このアプリケーションのログイン情報を保存する」チェック・ボックスを選択します。資格証明を保存すると、サーバーでは、Single Sign-On ユーザー名を使用して、アプリケーション名とパスワードを検索および取得し、認証用のユーザー情報を要求せずにユーザー・ログインを実行します。

Single Sign-On Server は、ユーザーが一度 Single Sign-On Server にログインすれば、ユーザーにかわってユーザー名とパスワードを外部アプリケーションに提供するように構成できます。また、アプリケーション用の資格証明を Single Sign-On データベースに格納するように選択することもできます。

mod_osso

mod_osso モジュールは、OracleAS アプリケーションに認証を提供する Oracle HTTP Server モジュールです。このモジュールは Oracle HTTP Server 上にあります。これにより、ユーザーが一度 OracleAS Single Sign-On Server にログインすると、OracleAS Single Sign-On で保護されるアプリケーションがユーザー名とパスワードのかわりに HTTP ヘッダーを受け取ることができます。これらのヘッダーの値は、mod_osso Cookie に格納されます。

mod_osso モジュールは、OracleAS Single Sign-On の以前のリリースでパートナ・アプリケーションを統合するために使用されていた Single Sign-On SDK にかわるものです。mod_osso は、アプリケーション・サーバーに配置すると、Single Sign-On Server の唯一のパートナ・アプリケーションとして機能して認証プロセスを単純化します。このようにして、mod_osso は、透過的な OracleAS アプリケーションの認証を実現します。結果として、OracleAS アプリケーションの管理者は、SDK との統合作業から解放されます。

ユーザーの認証後、アプリケーションでユーザーを認可するために必要に応じて使用される単純なヘッダー値が mod_osso によって送信されます。

- ユーザー名
- ユーザー GUID
- 言語および地域

Single Sign-On Server から mod_osso に渡される属性の詳細は、『Oracle Identity Management アプリケーション開発者ガイド』の mod_osso に関する章を参照してください。この章では、Single Sign-On に対応するアプリケーションを開発する方法について説明します。

mod_osso は、Oracle HTTP リスナーでのみ動作します。OracleAS Single Sign-On プラグインを使用すると、Sun One や IIS などのサード・パーティ製リスナーで動作するアプリケーションを保護できます。OracleAS Single Sign-On プラグインの使用方法は、『Oracle HTTP Server 管理者ガイド』のこのツールに関する付録を参照してください。

Oracle Internet Directory

Oracle Internet Directory は、分散ユーザーやネットワーク・リソースに関する情報の検索を可能にする、一般的な用途のディレクトリ・サービスです。Oracle Internet Directory は、すべての Single Sign-On ユーザー、すなわち管理者や非管理者のアカウントとパスワード用のリポジトリです。Single Sign-On Server は、このディレクトリ内のユーザー・エントリに基づいてユーザーを認証します。同時に、アプリケーションでのユーザー検証が可能となるユーザー属性をこのディレクトリから取得します。

Oracle Identity Management インフラストラクチャ

すべての企業識別情報および企業内の様々なアプリケーションへのアクセスを集中的かつ安全に管理できるようにするためのインフラストラクチャです。OracleAS Single Sign-On は、統合されたインフラストラクチャのリンクの 1 つで、このインフラストラクチャには Oracle Internet Directory、Oracle Directory Integration and Provisioning、Oracle Delegated Administration Services および OracleAS Certificate Authority も組み込まれています。Oracle Identity Management インフラストラクチャと呼ばれるこれらのコンポーネントは、連携して、ユーザーのセキュリティ・ライフ・サイクルとその他のネットワーク・エンティティを効率的かつ経済的に管理します。

Oracle Identity Management の利点は、『Oracle Identity Management Administrator's Guide』を参照してください。

Single Sign-On プロセス

この項では、次のプロセスについて説明します。

- Single Sign-On Server へのアクセス
- パートナ・アプリケーションへのアクセス
- 外部アプリケーションへのアクセス
- アプリケーションにアクセスする URL の制限
- シングル・サインオフ
- パスワードの変更
- グローバル・ユーザーの非アクティビティ・タイムアウト
- ワイヤレス・オプションによるサインオン

Single Sign-On Server へのアクセス

管理者以外のユーザーは、最初に OracleAS Portal などのパートナ・アプリケーションの URL を入力して、Single Sign-On Server にアクセスする必要があります。URL を入力すると、Single Sign-On ログイン画面が表示されます。正しいユーザー名とパスワードを一度入力すると、再度資格証明書を入力せずに、他のパートナ・アプリケーションや外部アプリケーションにアクセスできます。

管理ユーザーは、次のフォームの URL を入力すると、シングル・サインオンの管理ホームページにアクセスできます。

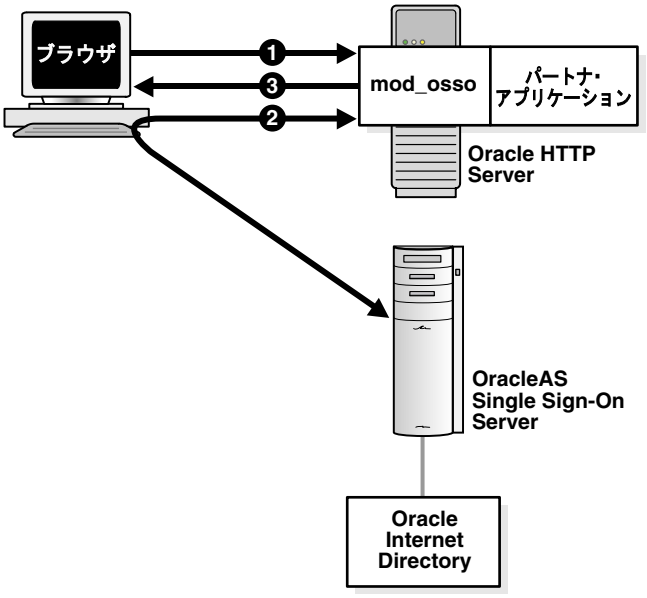
`http://`*`host`*`:`*`port`*`/sso`

host は、Single Sign-On Server が配置されているコンピュータの名前を示し、*port* は、サーバーのポート番号を示します。サーバーで SSL が有効になっている場合は、`http` を `https` に置き換える必要があります。ポート番号が `80` または `443`（SSL）の場合、URL から省略可能です。これらは、デフォルトのポート番号です。

パートナ・アプリケーションへのアクセス

図 1-1 は、ユーザーが mod_osso によって保護されているパートナ・アプリケーションの URL をリクエストした場合のプロセスを示しています。

図 1-1 mod_osso によるシングル・サインオン



1. ユーザーは、パートナ・アプリケーションにアクセスします。
2. ユーザーが、Single Sign-On Server へリダイレクトされます。サーバーで、ユーザーの資格証明がチェックされます。Oracle Internet Directory の資格証明の検証が完了すると、サーバーは SSO セッション Cookie を設定し、認証トークンをパートナ・アプリケーションに渡します。
3. パートナ・アプリケーションはリクエストされたコンテンツを提供します。

パートナ・アプリケーションに対する 2 回目以降の認証

パートナ・アプリケーションへのアクセスが要求されると、パートナ・アプリケーション・ログイン処理が開始されます。すでに Single Sign-On Server にログインしている場合、新しいパートナ・アプリケーションにアクセスすると次のように処理されます。

1. ユーザーは、パートナ・アプリケーションにアクセスします。
2. ユーザーが、Single Sign-On Server へリダイレクトされます。サーバーでは、ユーザーの認証資格証明はチェックされません。SSO セッション Cookie を使用して、ユーザー ID が検証されます。
3. サーバーがパートナ・アプリケーションに認証トークンを渡します。
4. パートナ・アプリケーションはリクエストされたコンテンツを提供します。

パートナ・アプリケーションからのログアウト

外部アプリケーションとは異なり、パートナ・アプリケーションではログアウト制御が Single Sign-On Server に渡されます。ユーザーが 1 つのパートナ・アプリケーションからログアウトすると、その他のパートナ・アプリケーションからも自動的にログアウトされます。

外部アプリケーションへのアクセス

外部アプリケーションは、Single Sign-On パートナ・アプリケーションの OracleAS Portal を通じて利用できます。

この項の項目は次のとおりです。

- OracleAS Portal の外部アプリケーション・ポートレットへのアクセス
- 外部アプリケーションに対する初めての認証
- 外部アプリケーションに対する 2 回目以降の認証
- 外部アプリケーションからのログアウト

OracleAS Portal の外部アプリケーション・ポートレットへのアクセス

外部アプリケーションにアクセスするには、OracleAS Portal ホームページで「外部アプリケーション」ポートレットを選択し、表示される外部アプリケーションのリストからアプリケーションを選択します。

外部アプリケーションに対する初めての認証

「外部アプリケーション」ポートレットからアプリケーションを選択すると、外部アプリケーションのログイン・プロシージャが開始されます。アプリケーションに初めてアクセスした場合は、次の手順が実行されます。

1. 外部アプリケーションのログイン・プロシージャは、シングル・サインオンのパスワード・ストアで資格証明をチェックします。資格証明が見つからない場合、Single Sign-On Server は資格証明の入力を求めるプロンプトを表示します。
2. ユーザー名とパスワードを入力します。アプリケーションのログイン画面で「このアプリケーションのログイン情報を保存する」チェック・ボックスを選択すると、これらの資格証明をパスワード・ストアに保存できます。
3. 資格証明をパスワード・ストアに保存する場合、Single Sign-On Server はこの資格証明を使用してログイン・フォームを作成し、アプリケーションのログイン処理ルーチンに送信します。このルーチンは、管理者によってあらかじめ構成されており、リクエストされたアプリケーションに関連付けられています。
4. Single Sign-On Server は、クライアント・ブラウザに対して、フォームと、そのフォームを外部アプリケーションにただちに送信する命令を送信します。
5. クライアントは、外部アプリケーションにフォームを送信してログインします。

資格証明をパスワード・ストアに保存しない場合は、ログインするたびにユーザー名とパスワードを入力する必要があります。

外部アプリケーションに対する 2 回目以降の認証

外部アプリケーションに初めてアクセスしたときに資格証明を保存した場合、Single Sign-On Server では以降のログイン時にその資格証明が使用されます。プロセスは次のとおりです。

1. ユーザーが、OracleAS Portal の「外部アプリケーション」ポートレットからリンクの 1 つをクリックします。
2. 外部アプリケーションのログイン・プロシージャは、パスワード・ストアで資格証明をチェックします。
3. Single Sign-On Server は資格証明を見つけます。この資格証明を使用してログイン・フォームを作成し、アプリケーションのログイン処理ルーチンに送信します。このルーチンは、管理者によってあらかじめ構成されており、リクエストされたアプリケーションに関連付けられています。
4. Single Sign-On Server は、クライアント・ブラウザに対して、フォームと、そのフォームを外部アプリケーションにただちに送信する命令を送信します。
5. クライアントは、外部アプリケーションにフォームを送信してログインします。

外部アプリケーションからのログアウト

パートナ・アプリケーションとは異なり、外部アプリケーションではログアウト制御が Single Sign-On Server に渡されません。ユーザーが、これらの各アプリケーションからログアウトすることになります。

アプリケーションにアクセスする URL の制限

一部のブラウザでは、ユーザーがリソースにアクセスしようとすると、リクエストされた元の URL がブラウザで許可されている最大長より短い場合でも、URL の最大長を超える場合があります。

シングル・サインオフ

実行しているアプリケーションからログアウトすることによって、シングル・サインオン・セッションを終了し、アクティブなすべてのパートナ・アプリケーションから同時にログアウトできます。パートナ・アプリケーションで「ログアウト」をクリックすると、「シングル・サインオフ」ページが表示され、そこでログアウトを実行できます。

サインオフに成功すると、「シングル・サインオフ」ページの各アプリケーション名の横にチェック・マークが表示されます。アプリケーション名の横に壊れたイメージが表示された場合、ログアウトに失敗したことを示しています。

1 つのセッションでアクティブ化されていたすべてのアプリケーション名にチェック・マークが表示されれば、「戻る」を選択して、ログアウトを開始したアプリケーションに戻ることができます。

パスワードの変更

パスワードの変更画面は、パスワードの期限切れが近く、猶予期間ログインの期間内である場合にのみ表示されます。パスワードがまだ有効な場合は、この画面で**「取消」**をクリックしてログインを続行できます。

その他の状況でパスワードを変更またはリセットするには、管理者でないユーザーは Oracle Delegated Administration Services へ移動する必要があります。これは、Oracle Internet Directory のサービスの 1 つで、ユーザーとグループの管理機能を実行します。

Oracle Delegated Administration Services ホームページは、次のフォームの URL によってアクセスできます。

`http://`*`host`*`:`*`port`*`/oiddas/`

`host` は、Oracle Delegated Administration Services が配置されているコンピュータの名前を示します。*`port`* は、このサーバーのポート番号を示します。Oracle Delegated Administration Services と OracleAS Single Sign-On は、通常は同じホスト名になります。Oracle Delegated Administration Services と OracleAS Single Sign-On をホスティングする Oracle HTTP Server が SSL に対応している場合は、`http` を `https` に置き換える必要があります。ポート番号は、デフォルトの `80` または `443`（SSL）の場合には省略できます。

パスワードには、&、{、}、<、>、"、'、(および) の各文字は使用できません。

注意： Single Sign-On ユーザー名とは異なり、Single Sign-On パスワードは、大 / 小文字を区別します。また、ユーザーの属している Oracle Internet Directory レルムに準拠します。

グローバル・ユーザーの非アクティビティ・タイムアウト

グローバル・ユーザーの非アクティビティ・タイムアウトは、あらかじめ構成されたアイドル時間を経過した場合、アプリケーションで再認証を要求できるようにする機能です。このタイムアウトは、シングル・サインオン・セッションのタイムアウトよりも短い非アクティビティ・タイムアウトが必要なセキュリティ重視のアプリケーションにとって有用な機能です。

グローバル・ユーザーの非アクティビティ・タイムアウトの制限時間の超過後にアプリケーションにアクセスとすると、通常の認証リクエストがアプリケーションから Single Sign-On Server に送信されます。非アクティビティ・タイムアウトの制限時間を超過していることが Single Sign-On Server で確認されると、ログインを要求するプロンプトが表示されます。制限時間を超過していない場合、ユーザーはセッション Cookie によって認証されます。

注意： シングル・サインオン・セッションが有効な場合でも、グローバル・タイムアウトの制限時間を超過している場合は、資格証明が要求されます。

関連項目： 2-11 ページの「グローバル・ユーザーの非アクティビティ・タイムアウトの構成」

ワイヤレス・オプションによるサインオン

OracleAS アプリケーションには、PDA や携帯電話、音声認識システムなどのモバイル・デバイスまたはワイヤレス・デバイスを使用してアクセスできます。PC ベース・システムの場合と同様に、認証メカニズムは OracleAS Single Sign-On です。ワイヤレス・オプションは、OracleAS のインストール時に選択できます。ワイヤレス・オプションを選択すると、モバイル・デバイス用のゲートウェイである Wireless deviceportal が Single Sign-On Server に自動的に登録されます。

OracleAS Wireless の詳細は、『Oracle Application Server Wireless 管理者ガイド』および『Oracle Application Server Wireless 開発者ガイド』を参照してください。

2

基本的な管理

この章では、シングル・サインオンの管理に必要な作業について説明します。この章の項目は次のとおりです。

Single Sign-On 管理者ロール

Single Sign-On Server への初回アクセス時には、Single Sign-On 管理者のみが存在します。この管理者は、orcladmin という名前の Oracle Application Server Single Sign-On スーパー・ユーザーです。Oracle Application Server Single Sign-On のインストール時に、インストールを行った人がこのユーザーのパスワードを設定します。パスワードには、&、{、}、<、>、"、'、(および) の各文字は使用できません。orcladmin アカウントは、シングル・サインオンの管理グループである iASAdmins のアカウントなど、他のアカウントを作成するために使用されます。

Single Sign-On 管理者は、管理ページを使用して次の作業を実行できます。

- サーバー設定の構成
- パートナ・アプリケーションの管理
- 外部アプリケーションの管理

管理権限の付与

Single Sign-On 管理者の権限を使用するには、管理グループ iASAdmins のメンバーになる必要があります。すなわち、このグループの既存メンバーが、新しい管理者をグループに追加する必要があります。

ユーザーを iASAdmins に割り当てるには、次の手順を実行します。

1. Oracle Directory Manager を起動します。このツールの起動方法は、『Oracle Internet Directory 管理者ガイド』を参照してください。
2. `cn=orcladmin` すなわちディレクトリ・スーパー・ユーザーとしてログインします。Oracle Internet Directory のインストール時は、このユーザーに割り当てたパスワードを使用してください。

   注意 : ディレクトリ・スーパー・ユーザー `cn=orcladmin` は、OracleAS スーパー・ユーザー `orcladmin` と同一ではありません。これらは、階層的に等しくない個別アカウントです。

3. 「システム・オブジェクト」フレームで、次のエントリを続けてクリックします。
   - Entry Management
   - dc=*default_identity_management_realm*
   - cn=OracleContext
   - cn=Groups
   - cn=iASAdmins

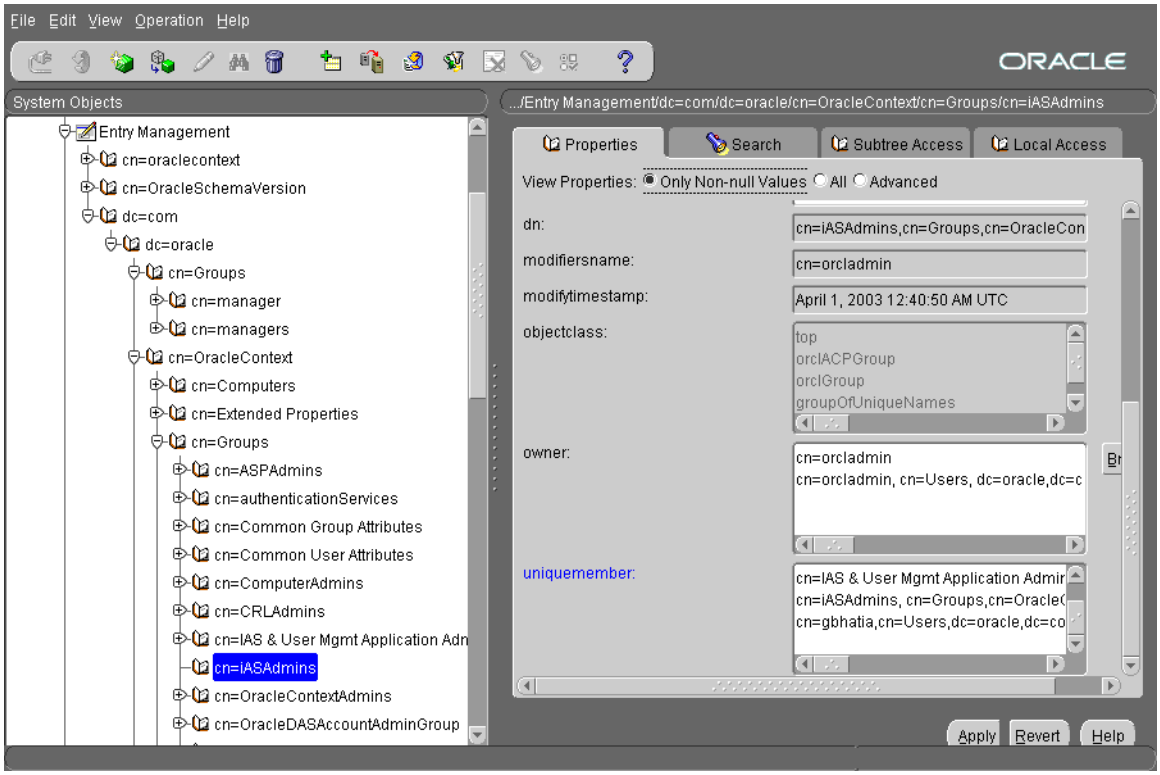
   例 :

   ```
   cn=iASAdmins,cn=Groups,cn=OracleContext,dc=oracle,dc=com
   ```

   `dc=oracle,dc=com` は、デフォルトの ID 管理レルムを示します。実際には、多くの場合、インストール先のドメイン名がデフォルトとして使用されます。

4. **iASAdmins** のタブの「**uniquemembers**」テキスト・ボックスに、そのユーザーのエントリを追加します。`uniquemembers` は、エントリ `iASAdmins` の属性の 1 つです。それ自体で、グループ `iASAdmins` のメンバーが定義されます。必ず、ユーザーの完全な DN を追加してください。2-3 ページの図 2-1 では、ユーザー `cn=gbhatia,cn=users,dc=oracle,dc=com` が追加されています。
5. 「適用」をクリックします。

図 2-1 Oracle Directory Manager の「iASAdmins」タブ



新規ユーザーを作成するには、Oracle Delegated Administration Services を使用します。詳細は、『Oracle Identity Management 委任管理ガイド』の Oracle Internet Directory セルフサービス・コンソールに関する章を参照してください。

Single Sign-On 管理グループの変更

デフォルトでは、Single Sign-On Server は、cn=iASAdmins,cn=Groups,cn=OracleContext, default_realm_dn ディレクトリ・エントリを使用して、ユーザーが OracleAS Single Sign-On の管理権限を持っているかどうかを判定します。次の手順を実行すると、グループ cn=sso_admins,dc=us,dc=acme,dc=com のすべてのユーザーは、Single Sign-On Server を管理できます。次の手順を実行して、グループをメンバーとして含めることも可能です。

1. ディレクトリでデフォルト・レルムの識別名に相対的な新規グループを作成します。たとえば、デフォルト・レルムが dc=us,dc=acme,dc=com であり、sso_admins というデフォルトの管理グループを作成する場合、次のエントリを作成します。

   ```
   cn=sso_admins,dc=us,dc=acme,dc=com
   ```

   Oracle Directory Manager または LDAP コマンドライン・ツールを使用して、このエントリを作成します。

2. *ORACLE_HOME*/sso/conf にある policy.properties ファイルを編集して、ディレクトリ内でグループ・エントリを識別するように ssoAdministratorGroupDN を更新します。

   ```
   ssoAdministratorGroupDN = cn=sso_admins
   ```

   デフォルト・レルム（この例では dc=us,dc=acme,dc=com）を含める必要はありません。Single Sign-On Server により、グループ・メンバーシップのチェック時に適切なレルムが付加されます。

3. 2-6 ページの「OC4J_SECURITY インスタンスの停止と起動」の手順に従って、OC4J_SECURITY インスタンスを再起動します。

policy.properties

policy.properties ファイルは、OracleAS Single Sign-On の構成ファイルです。このファイルには、Single Sign-On Server で必要とされる基本パラメータが組み込まれています。これらのパラメータのデフォルト値は、大部分のインストールに適合します。このファイルは変更しないでそのまま使用できます。

policy.properties は、マルチレベル認証などのシングル・サインオン拡張機能を実装するように構成できます。policy.properties ファイルは *ORACLE_HOME*/sso/conf にあります。付録 C「policy.properties」にこのファイルのコピーがあります。

注意： policy.properties を編集するときは、各行の末尾に空白を入れないでください。ファイルを編集したら、Single Sign-On Server を再起動します。詳細は、次の項を参照してください。

Single Sign-On コンポーネントの停止と起動

Single Sign-On コンポーネントの停止と起動には、コマンドラインまたは Oracle Enterprise Manager Application Server Control コンソールのどちらかを使用します。コンソールを使用すると、いくつかのコンポーネントを一度に停止または起動できます。コマンドラインの場合、この処理には複数のコマンドが必要です。

Application Server Control コンソールを使用する場合

コンソールを使用して Single Sign-On コンポーネントを停止および起動するには、次の手順を実行します。

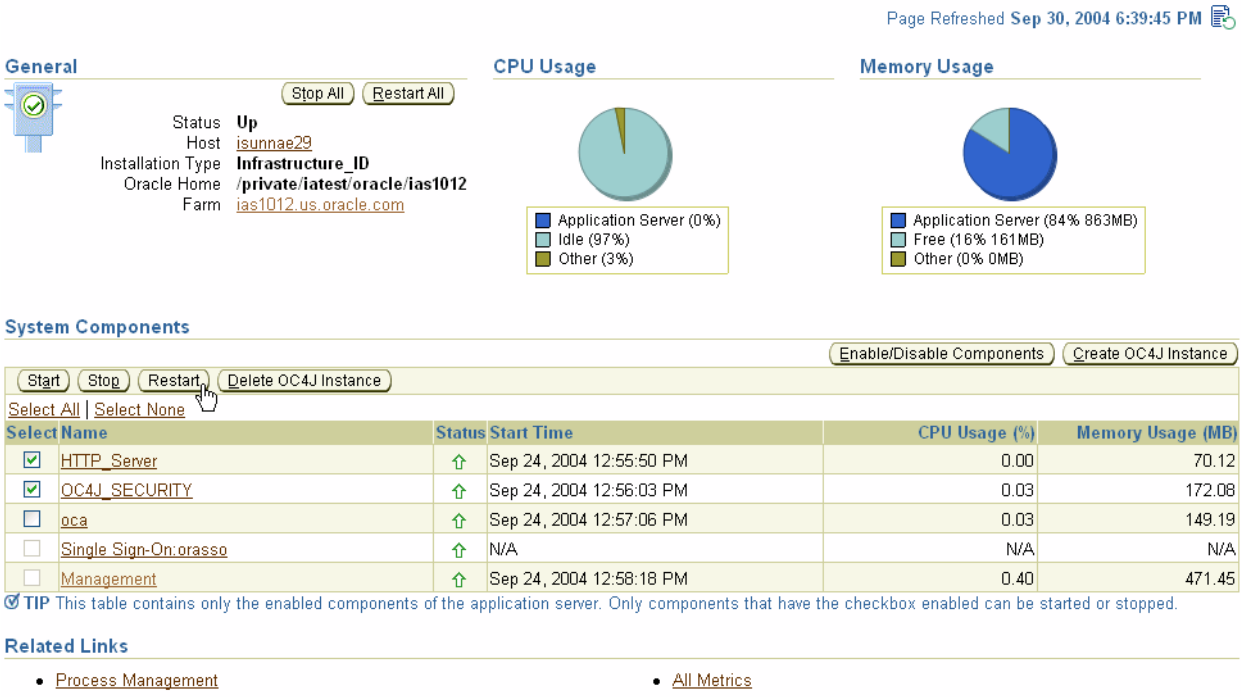
1. 管理対象の Oracle Enterprise Manager インフラストラクチャ・インスタンスのスタンドアロン・コンソールに移動します。スタンドアロン・コンソールに移動するには、OracleAS インスタンスをホスティングしているコンピュータのホスト名と Oracle Enterprise Manager のポート番号を入力します。デフォルトのポート番号は `1156` です。次のファイルから、インスタンス固有のポート番号を調べることができます。

   UNIX: *`ORACLE_HOME`*`/install/setupinfo.txt`

   Windows: *`ORACLE_HOME`*`¥install¥setupinfo.txt`

2. OracleAS 管理者の資格証明を使用してログインします。
3. 「ファーム」ページの**「スタンドアロン・インスタンス」**セクションで、適切な OracleAS インスタンスを選択します。
4. 「Application Server」ページの**「システム・コンポーネント」**リストで、停止、起動または再起動するコンポーネントのチェック・ボックスを選択し、リストの上の該当するボタンをクリックします。2-5 ページの図 2-2 では、シングル・サインオン中間層を再起動しています。ID 管理インフラストラクチャ全体を停止または再起動するには、ページ上段の**「すべてを停止」**または「すべてを再起動」をクリックします。

図 2-2 コンソールを使用したシングル・サインオン中間層の再起動



コマンドラインを使用する場合

コマンドを個別に発行することで、Oracle HTTP Server のみ、またはシングル・サインオン中間層全体を停止および起動できます。他のコマンドで、OC4J_SECURITY インスタンスのみを停止および起動することもできます。インフラストラクチャのすべてのコンポーネントを停止および起動するコマンドもあります。

Oracle HTTP Server の停止と起動

Oracle HTTP Server を停止して起動するには、次の 2 つのコマンドを発行します。

```
ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl startproc process-type=HTTP_Server
```

また、次のコマンドを発行して、Oracle HTTP Server を停止して起動することができます。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
```

OC4J_SECURITY インスタンスの停止と起動

OC4J_SECURITY インスタンスを停止して起動するには、次の 2 つのコマンドを発行します。

```
ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=OC4J_SECURITY
ORACLE_HOME/opmn/bin/opmnctl startproc process-type=OC4J_SECURITY
```

また、次のコマンドで OC4J_SECURITY インスタンスを停止して起動することもできます。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

シングル・サインオン中間層の停止と起動

シングル・サインオン中間層を停止して起動するには、Oracle HTTP Server と OC4J_SECURITY インスタンスの両方を停止して起動します。

```
ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl startproc process-type=HTTP_Server

ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=OC4J_SECURITY
ORACLE_HOME/opmn/bin/opmnctl startproc process-type=OC4J_SECURITY
```

また、次のコマンドでシングル・サインオン中間層を停止して起動することもできます。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

すべてのコンポーネントの停止と起動

Oracle HTTP Server、Single Sign-On Server、OC4J および Oracle Internet Directory を停止して起動するには、次のコマンドを発行します。

```
ORACLE_HOME/opmn/bin/opmnctl stopall
ORACLE_HOME/opmn/bin/opmnctl startall
```

このコマンドでは、インフラストラクチャ・コンポーネントがすべて同じ Oracle ホーム・ディレクトリに格納されていることを前提としています。

データベースの停止と起動

データベースをシャットダウンする必要がある場合は、次の手順に従ってデータベースおよび影響のあるコンポーネントを停止し、起動します。

1. Oracle HTTP Server、Single Sign-On Server、OC4J および Oracle Internet Directory を停止します。

   ORACLE_HOME/opmn/bin/opmnctl stopall

2. データベースをシャットダウンします。
3. データベースを起動します。
4. Oracle HTTP Server、Single Sign-On Server、OC4J および Oracle Internet Directory を起動します。

   ORACLE_HOME/opmn/bin/opmnctl startall

アクセス不能なサーバーのトラブルシューティング

Single Sign-On Server がアクセス不能であるために、OracleAS アプリケーションにアクセスできない場合があります。次の手順に従って、問題を診断してください。

1. 第 11 章の「監視用ページへのアクセス」の手順に従って、Application Server Control コンソールの「Application Server」ページにアクセスします。
2. 「Application Server」ページの「システム・コンポーネント」表で、Single Sign-On Server が実際に停止しているかどうかを確認します。赤の下向き矢印はサーバーが停止していることを示します。緑の上向き矢印は稼働中であることを示します。
3. 表をチェックして、Oracle HTTP Server が停止しているかどうかを確認します。
4. Oracle HTTP Server が停止している場合は、コマンドラインまたは Application Server Control コンソールを使用して再起動します。「Single Sign-On コンポーネントの停止と起動」を参照してください。
5. Oracle HTTP Server を起動できない場合は、サーバーのログ・ファイルをチェックして問題点を確認します。このファイルは、*ORACLE_HOME*/opmn/logs および *ORACLE_HOME*/Apache/Apache/logs/error_log にあります。
6. OracleAS のメタデータ・リポジトリのステータスをチェックします。
   a. Oracle Enterprise Manager Database Control コンソールを起動します。

      ORACLE_HOME/bin/emctl start dbconsole

   b. ブラウザで、Oracle Enterprise Manager Database Control の URL を入力します。

      http://*host_name.domain*:*port*/em

      URL で、*host_name* にはメタデータ・リポジトリがインストールされているコンピュータ名を指定します。*port* には、インストール時に Database Control 用に確保されたポート番号を指定します。ポート番号を調べるには、ファイル *ORACLE_HOME*/install/portlist.ini を確認します。次の行を調べます。

      Enterprise Manager Console HTTP Port(*database_name*) = *port_number*

      ポート番号は、5500 ～ 5519 の範囲の値です。
7. Database Control のページに SYS のアカウントでログインし、SYSDBA として接続します。
8. Database のホームページで、「一般」セクションのステータス・インジケータを調べます。データベースが稼働している場合は、手順 10 に進みます。データベースが停止している場合は、「起動」ボタン、または必要に応じて「リカバリの実行」ボタンをクリックします。

   データベースが起動したら、Application Server Control コンソール、Database Control コンソール、インフラストラクチャの中間層、およびメタデータ・リポジトリのインストールに関連する中間層を再起動します。詳細は、『Oracle Application Server 管理者ガイド』の起動と停止に関する章を参照してください。

データベースを起動できない場合は、『Oracle Database 管理者ガイド』を参照してください。

9. Application Server Control コンソールの「Application Server」ページで「システム・コンポーネント」表を調べて、OC4_SECURITY インスタンスが実行中かどうかを確認します。または、コマンドラインを使用して確認することもできます。

   ```
   opmnctl status
   ```

   OC4J_SECURITY インスタンスが停止している場合は、Application Server Control コンソールを使用して再起動します。OC4J_SECURITY が正常に起動したら、Single Sign-On Server にアクセス可能かどうかを確認します。OC4J_SECURITY が起動しない場合は、OC4J_Security のログでエラーをチェックします。これらのログの詳細は、付録 A の「ログ・ファイルの表示」を参照してください。

10. OC4J_SECURITY インスタンスのステータスをチェックした手順 10 と同じ方法で、Oracle Internet Directory のステータスをチェックします。必要に応じて、Application Server Control コンソールを使用してディレクトリを起動します。ディレクトリが起動しない場合は、ディレクトリのエラー・ログをチェックします。

OracleAS Single Sign-On 用ブラウザの作業環境の設定

OracleAS Single Sign-On でログインおよびログアウトするには、次のブラウザ設定を行う必要があります。

キャッシュ設定

正しくキャッシュを設定する手順は次のとおりです。

1. 次の順序でクリックして、キャッシュ設定のダイアログ・ボックスに移動します。
   - Internet Explorer:
     - ツール
     - インターネット オプション
     - 全般
     - 設定
   - Netscape Communicator:
     - 編集
     - 設定
     - 詳細
     - キャッシュ
2. Internet Explorer では「ページを表示するごとに確認する」を選択し、Netscape Communicator では「ページにアクセスするたび」を選択します。

イメージ設定

イメージを自動的にロードする手順は次のとおりです。

1. 次の順序でクリックします。
   - Internet Explorer:
     - ツール
     - インターネット オプション
     - 詳細設定
   - Netscape Communicator:
     - 編集
     - 設定
     - 詳細
2. Internet Explorer では「画像を表示する」を選択し、Netscape Communicator では「自動的に画像を読み込む」を選択します。

管理ページへのアクセス

シングル・サインオン UI の管理ページでは、シングル・サインオン・セッションの長さを設定したり、サーバーによる IP アドレスの検証を有効にすることができます。また、これらのページでは、パートナ・アプリケーションおよび外部アプリケーションを管理することもできます。

管理ページにアクセスするには、次の手順を実行します。

1. 次のフォームの URL を入力します。

   `http://`*`host`*`:`*`port`*`/sso`

   `host` にはサーバーが配置されているコンピュータ名を代入し、*`port`* にはこのサーバーのポート番号を代入します。サーバーで SSL が有効になっている場合は、`http` を `https` に置き換える必要があります。ポート番号は、デフォルトの `80` または `443`（SSL）の場合には省略できます。

   「パートナ・アプリケーションへのアクセス」ページが表示されます。
2. 「パートナ・アプリケーションへのアクセス」ページの右上にある「ログイン」をクリックします。

   「ログイン」ページが表示されます。
3. 管理者のユーザー名とパスワードを入力して、**「ログイン」**をクリックします。
4. ホームページが表示されます。管理機能を実行するには、「SSO Server 管理」をクリックします。

図 2-3 は、「**SSO Server 管理**」ページを示しています。

図 2-3「SSO Server 管理」ページ

Single Sign-On Server の編集ページを使用したサーバーの構成

シングル・サインオン・セッションの長さの修正と IP アドレスの検証を行うには、Single Sign-On Server の編集ページを使用します。Single Sign-On Server の編集ページにアクセスするには、「SSO Server 管理」ページで「**Single Sign-On Server 構成の編集**」をクリックします。

Single Sign-On Server の編集ページには、次のヘッダーとフィールドが表示されます。

表 2-1 Single Sign-On Server ポリシー

フィールド	説明
シングル・サインオン・セッションの持続期間	ユーザーがログインできるセッションの有効時間を入力します。デフォルトは 8 時間です。
Single Sign-On Server へのリクエストで使用される IP アドレスを検証します。	ブラウザの IP アドレスと認証リクエストに使用される IP アドレスが一致することを検証するときに選択します。このチェック・ボックスは、デフォルトでは選択されていません。

これらのパラメータのいずれかを変更した場合は、OC4J_SECURITY インスタンスを再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

グローバリゼーション・サポートの構成

OracleAS Single Sign-On 製品をインストールするとき、数十の言語がサポートされています。デフォルトの言語は英語ですが、ユーザーはブラウザをサポートされている任意の言語に設定できます。

サポートされている言語コードの詳細なリストは、次の URL にある『Oracle Application Server グローバリゼーション・サポート・ガイド』の付録 A を参照してください。

http://www.oracle.com/technology/documentation/index.html

この URL で表示されたページから、OracleAS Single Sign-On のマニュアルへのリンクをクリックし、適切なリリースのライブラリを表示する「View Library」リンクをクリックします。

グローバル・ユーザーの非アクティビティ・タイムアウトの構成

この項に進む前に、第 1 章「OracleAS Single Sign-On」の「グローバル・ユーザーの非アクティビティ・タイムアウト」の項を一読してください。

グローバル・ユーザーの非アクティビティ・タイムアウトは、1 つのドメインにのみ適用されます。つまり、タイムアウトを有効にしたコンピュータは、同じ Cookie ドメインに属している必要があります。これらのコンピュータ上のアプリケーションは、ドメインの Cookie を使用してユーザーのアクティビティを追跡します。たとえば、Single Sign-On Server に `login.acme.com` を使用している場合は、システム内の他のコンピュータでも、ホスト名に `.acme.com` が含まれている必要があります。たとえば、あるコンピュータは `host1.acme.com`、別のコンピュータは `host2.acme.com` というようになります。また、Single Sign-On Server を含め、これらすべてのコンピュータのクロックが 10 秒以内で同期化されている必要があります。

グローバル・ユーザーの非アクティビティ・タイムアウトは、デフォルトでは構成されていません。`ssogito.sql` スクリプトを実行して、有効にする必要があります。このスクリプトは、*`ORACLE_HOME`*`/sso/admin/plsql/sso` にあります。次の手順には、`ssogito.sql` の例も含まれています。

グローバル・ユーザーの非アクティビティ・タイムアウトを構成する手順は次のとおりです。

1. シングル・サインオンのスキーマ名とパスワードを使用して、SQL*Plus にログインします。デフォルトのスキーマ名は `orasso` です。パスワードの取得方法は、付録 B を参照してください。
2. 次のコマンドを入力して `ssogito.sql` を実行します。

   ```
   SQL> @ssogito.sql
   ```

   フィールドのリストが表示されます。
3. 「{timeout_cookie_domain} の値の入力」フィールドに、Single Sign-On Server に対応しているすべてのアプリケーションに共通のドメイン名を入力します。ドメイン名の前には必ずピリオドを付けてください。

   注意： このフィールドを空白にした場合、ドメイン名は、デフォルトで Single Sign-On Server のホスト名に設定されます。
4. 「inactivity period に値を入力してください」フィールドに、必要な非アクティブ期間を分単位で入力します。
5. この新規設定を有効にする場合は、[Return] キーまたは [Enter] キーを押します。このトランザクションを取り消す場合は、[Return] または [Enter] キーを 2 回押します。

トランザクションを完了すると、スクリプトによって、新しいタイムアウト設定のサマリーが提供されます。ssogito.sql の例は次のようになります。

```
SQL> @ssogito
==========================================
SSO Server Inactivity Timeout Configuration
==========================================
Timeout          : DISABLED
Cookie name      : OSSO_USER_CTX
Cookie domain    :
Inactivity period: 15 minutes
Encryption key   : 093D678526DAA66D
Note: timeout cookie domain will be defaulted
to the SSO Server hostname
------------------------------------------
To disable timeout set inactivity period to 0, (zero)
Press return key twice if you do not want
to change timeout configuration.

PL/SQL procedure successfully completed.

Enter value for timeout_cookie_domain: .oracle.com
Enter value for inactivity_period: 15
Timeout                  : ENABLED
New timeout cookie domain: .oracle.com
New inactivity period    : 15 minutes

PL/SQL procedure successfully completed.

No errors.
```

6. シングル・サインオン中間層を再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl startproc process-type=HTTP_Server

ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=OC4J_SECURITY
ORACLE_HOME/opmn/bin/opmnctl startproc process-type=OC4J_SECURITY
```

7. 非アクティビティ・タイムアウトを有効にするアプリケーション中間層で、mod_osso.conf ファイルを編集します。OssoIdleTimeout パラメータがあり、on に設定されていることを確認します。このファイルは *ORACLE_HOME*/Apache/Apache/conf にあります。正しい設定のファイルは次のようになります。

```
LoadModule osso_module libsexec/mod_osso.so
<IfModule mod_osso.c>
  OssoIpCheck off
  OssoIdleTimeout on
  OssoConfigFile /u01/oracleas10g/Apache/Apache/conf/osso/osso.conf
#
#Insert Protected Resources
#
.
.
.
</IfModule>
```

8. アプリケーション中間層で Oracle HTTP Server を再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
```

Oracle Delegated Administration Service と Single Sign-On Server が同じ中間層にあり、前者にグローバル・ユーザーの非アクティビティ・タイムアウトを適用する場合は、シングル・サインオン中間層で手順 7 と 8 を実行します。

サンプル・ファイルの取得

ipassample.jar ファイルには、証明書を使用したサインオンや配置固有のページなど、シングル・サインオン機能のサンプル・コードが組み込まれています。このファイルを抽出するには次のコマンドを使用します。

ORACLE_HOME/jdk/bin/jar -xvf *ORACLE_HOME*/sso/lib/ipassample.jar

3

ディレクトリ対応 Single Sign-On

この章では、Oracle Internet Directory に依存している OracleAS Single Sign-On の機能について説明します。Oracle Internet Directory は、すべての Single Sign-On ユーザー、すなわち管理者と非管理者のアカウントおよびパスワード用のリポジトリです。ユーザーとグループの管理機能はこのディレクトリですべて処理されます。

> **注意：** Oracle Internet Directory は、サード・パーティ・リポジトリに対する認証を受けるように構成できます。詳細は、『Oracle Identity Management 統合ガイド』を参照してください。

この章の項目は次のとおりです。

Oracle Internet Directory におけるユーザー管理

Single Sign-On ユーザーを管理するには、次のツールを使用します。

- Oracle Delegated Administration Services

  Oracle Delegated Administration Services は、管理者がユーザーとグループの管理に使用できるセルフサービス・アプリケーションです。たとえば、ユーザーの作成や削除、パスワードの変更を行うことができます。

  次のフォームの URL を入力すると、Oracle Delegated Administration Service にアクセスできます。

  `http://host:port/oiddas/`

  host には Oracle Delegated Administration Services サーバーが配置されているコンピュータ名を代入し、*port* にはこのサーバーのポート番号を代入します。インフラストラクチャの通常のインストールでは、Oracle Delegated Administration Service と OracleAS Single Sign-On は同じホスト名になります。

- Oracle Directory Manager

  Oracle Directory Manager は、Oracle Internet Directory のほとんどの機能を管理する Java ベースのツールです。このツールを使用して、パスワード・ポリシーを構成できます。

- LDAP コマンドライン・ツール

  `ldapmodify` などのコマンドライン・ツールは、Oracle Delegated Administration Services と Oracle Directory Manager のかわりに使用できます。これらのツールを使用して、テキスト・ファイルを操作できます。これらは、LDAP データ交換（LDIF）フォーマットを使用する引数を取ります。

パスワード・ポリシー

Single Sign-On ユーザーのパスワードは、Oracle Internet Directory にユーザー・エントリの属性として格納されます。ユーザーは、パスワードの有効期限が近い場合にのみ Single Sign-On UI でパスワードを変更できます。Oracle Delegated Administration Services は、いつでもこの目的で使用できます。ディレクトリ管理者は、Oracle Directory Manager を使用して、パスワードの有効期限に関する動作を企業ニーズに適合するように調整できます。

この項の項目は次のとおりです。

パスワード・ルール

Oracle Directory Manager のフィールドでは、パスワードに必要な最小文字数を指定できます。デフォルト値の詳細は、『Oracle Internet Directory 管理者ガイド』のパスワード・ポリシーに関する章を参照してください。

パスワードには、&、{、}、<、>、"、'、(および) の各文字は使用できません。

パスワードの有効期限の構成

Oracle Directory Manager または LDAP コマンドライン・ツールを使用すると、パスワードの有効期限の構成や、ユーザーにパスワードの変更を要求する時間を指定できます。ユーザーの猶予期間ログインを構成することもできます。これは、ユーザーのパスワードが有効期限切れになった後の期間を示します。ユーザーがこの期間内にパスワードを変更しなかった場合は、管理者がパスワードをリセットする必要があります。

パスワードの変更ページの動作

パスワードの有効期限が切れている場合や期限切れが近いときにユーザーがログインすると、サーバーは次のように動作します。

パスワードが失効している場合

パスワードの失効を表す画面が表示されます。ユーザーは、ディレクトリ管理者に連絡してパスワードのリセットを要求する必要があります。

パスワードが間もなく失効する場合

ログイン・ページにエラー・メッセージが表示されます。この場合、このページを取り消すか、パスワードを変更することができます。いずれの場合でも、パスワードの変更ページが表示されないときと同様に認証が行われます。

猶予期間ログインの実施

猶予期間ログインがディレクトリで構成されている場合は、パスワードの有効期限が切れるとパスワードの変更ページが表示されます。この場合、このページを取り消すか、パスワードを変更することができます。いずれの場合でも、認証の手順はユーザーのパスワードが有効であるときと同じです。

パスワードの変更の強制

この機能は、管理者によるパスワードのリセット後にパスワードの変更をユーザーに要求します。パスワードの変更の強制を有効にするには、ディレクトリ・エントリ `cn=pwdpolicyentry,cn=common,cn=products,cn=OracleContext, dc=`*`default_identity_management_realm`* の `pwdMustChange` 属性を設定します。これには、コマンドライン・ツール `ldapmodify` を使用します。`TRUE` に設定すると、パスワードの変更の強制機能が有効になります。`FALSE` に設定すると無効になります。ツールの実行方法は、『Oracle Internet Directory 管理者ガイド』のパスワード・ポリシーに関する章を参照してください。

アカウント・ロックアウトの構成

アカウント・ロックアウトは、ユーザーがアカウントとパスワードの間違った組合せを、Oracle Internet Directory で許可されている回数を超えて送信したときに発生します。一度ロックアウトされたユーザーは、どのワークステーションからも Single Sign-On Server にアクセスできません。デフォルトでは、11 回目のログイン試行でロックアウトが発生します。この制限回数に達すると、有効なユーザー名とパスワードの組合せを使用してもログインできなくなります。

Single Sign-On ユーザーのアカウントは Oracle Internet Directory で管理されているため、ディレクトリ管理者は、アカウント・ロックアウト・ポリシーを決めておく必要があります。Oracle Directory Manager のフィールドを使用して、ロックアウトの有効化と無効化を設定したり、ロックアウト期間を指定できます。

デフォルトのロックアウト期間は 1 日です。

ユーザーのロック解除

ユーザーのロックを解除する方法は、『Oracle Internet Directory 管理者ガイド』のパスワード・ポリシーに関する章を参照してください。

パスワード・ポリシーの構成

パスワード・ポリシーの構成方法は、『Oracle Internet Directory 管理者ガイド』のパスワード・ポリシーに関する章を参照してください。

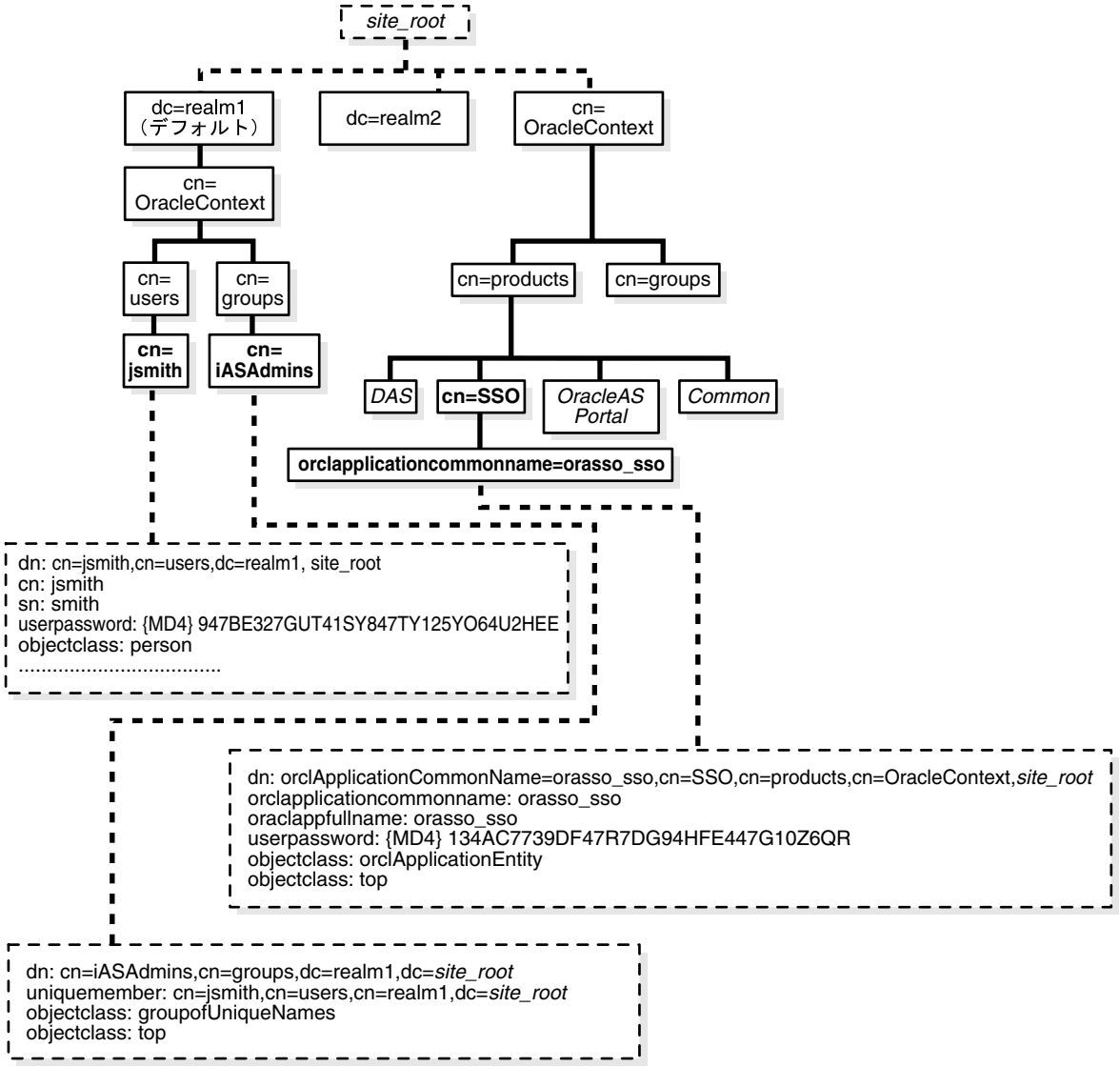
OracleAS Single Sign-On のディレクトリ・ツリー

他の OracleAS 補完コンポーネントと同様に、OracleAS Single Sign-On では、ディレクトリ情報ツリー（DIT）内に独自のコンテナがあります。このコンテナは、すべての Oracle 固有データのルートとしての役割を果すエントリである Oracle Context 内にあります。3-5 ページの図 3-1 に示す DIT の簡略図では、ルート Oracle Context とレルム固有 Oracle Context の両方が開かれています。ルート Oracle Context は、サイト全体の情報（すべての ID 管理レルムと製品に適用される情報）のリポジトリです。レルム固有の Oracle Context は、構造的にはルート・コンテキストのミラー・イメージですが、含まれる情報は特定のレルムのみに関連する情報です。これらのレルムには、特定のユーザー固有の構成情報とその他のネットワーク・エンティティが格納されます。レルムの詳細は、第 10 章「アプリケーション・サービス・プロバイダに対するサポートの有効化」を参照してください。

図 3-1 に示すように、Single Sign-On コンテナは、エントリ `cn=SSO` によって識別されます。このエントリには、1 つのエントリ `orclApplicationCommonName=orasso_sso` のみが含まれています。これは Single Sign-On Server のエントリです。図では、このエントリが開かれて、そのエントリを定義しているオブジェクト・クラスと属性が示されています。たとえば、`orclapplicationcommonname` 属性では、Single Sign-On Server のデフォルト名 `orasso` が指定されています。また、Single Sign-On Server には、`orclapplicationcommonname` に加えて、独自のパスワードがあることに注意してください。Single Sign-On Server がユーザー検索を実行するときに、ディレクトリ・サーバーはこのパスワードを使用して Single Sign-On Server を認証します。

コンテナ `Common` は、すべての OracleAS 製品に共通の情報リポジトリです。たとえば、製品がレルム検索ベースやノード、レルム・ニックネームを識別するための属性がこのコンテナに格納されています。この図には記載されていませんが、レルム固有の `Common` コンテナには、製品がレルム・サブツリー内でユーザーを検索するための属性が格納されています。図では、`SSO` コンテナの他に、管理者でもある OracleAS ユーザーのエントリも開かれています。

図 3-1 OracleAS Single Sign-On のディレクトリ情報ツリー



ディレクトリ・アクセス用 Single Sign-On Server の設定変更

ssooconf.sql スクリプトを使用して、ディレクトリ内の次の設定を変更できます。

- ディレクトリ・ホスト名
- ディレクトリ・ポート
- Single Sign-On Server 用のパスワード
- ディレクトリへの SSL 接続

> **注意：** ホスト名およびポート番号を変更できるのは、Oracle Internet Directory の新規インスタンスがレプリケートされたインスタンスである場合のみです。

Single Sign-On Server 用のディレクトリ設定を変更する手順は次のとおりです。

1. *ORACLE_HOME*/sso/admin/plsql/sso にあるスクリプトに移動します。
2. SQL*Plus にスキーマ orasso としてログインします。スキーマのパスワードの取得方法については、付録 B を参照してください。

   > **注意：** このスクリプトを実行できるのは、orasso のみです。

3. 次のコマンドを発行して ssooconf.sql を実行します。

   ```
   SQL> @ssooconf.sql
   ```

   次のプロンプトが表示されます。

   ```
   Enter value for new_oid_host
   ```

4. ディレクトリ・ホスト名に値を入力し、**[Return]** または **[Enter]** を押します。ディレクトリ・ホスト名を変更しない場合は、何も入力せずに **[Return]** または **[Enter]** を押し、次のプロンプトへ進みます。
5. 続く 3 つのプロンプト、Enter value for new_oid_port、Enter value for new_ssoserver_password および Enter value for new_ldapusessl に対し、手順 4 を繰り返します。最後のプロンプトには、Y（有効）または N（無効）のどちらかを入力します。

   > **注意：** Single Sign-On Server とディレクトリ間の SSL 接続は、デフォルトで存在します。

6. 最後にもう一度 **[Return]** または **[Enter]** を押して、変更内容を適用します。

   Single Sign-On Server の更新後の設定と更新前の設定が、スクリプトによって表示されます。

スクリプトの起動後に、変更を加えないことにした場合は、**[Return]** または **[Enter]** を押すと、既存の値のままになります。

ディレクトリ変更による Single Sign-On Server の更新

Single Sign-On Server では、Oracle Internet Directory DIT に関するメタデータがキャッシュされます。このメタデータには、ユーザー検索ベース、ユーザー・ニックネーム属性、レルム関連メタデータなどがあります。ディレクトリ DIT を変更した場合は、Single Sign-On Server のキャッシュを更新する必要があります。この作業を行うには、`ssoreoid.sql` スクリプトを実行します。

1. *ORACLE_HOME*`/sso/admin/plsql/sso` にあるスクリプトに移動します。
2. 次のように入力して、Single Sign-On スキーマにログインします。

   ```
   SQL> connect orasso/orasso_password
   ```

   `orasso` スキーマのパスワードの取得方法は、付録 B を参照してください。

   > **注意：** このスクリプトは、`sys` で実行することはできません。

3. 次のスクリプトを実行します。

   ```
   SQL> @ssoreoid.sql
   ```

4. Single Sign-On Server を再起動します。

   ```
   ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
   ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
   ```

スクリプトの実行を必要とする DIT 変更の一例を次に示します。

- デフォルトのレルム名またはレルム DN の変更あるいは両方の変更
- デフォルトのレルムの新規作成
- デフォルトのレルムのユーザー検索ベースまたはグループ検索ベースの変更あるいは両方の変更
- ユーザー・ニックネーム属性の変更

Oracle Internet Directory におけるレルム情報の変更方法は、『Oracle Internet Directory 管理者ガイド』を参照してください。

4

パートナ・アプリケーションの設定と管理

この章では、Single Sign-On でパートナ・アプリケーションを使用可能にする方法について説明します。このプロセスでは、認証モジュールの mod_osso を Single Sign-On Server に登録する必要があります。mod_osso およびパートナ・アプリケーションの詳細は、第 1 章を参照してください。

この章の項目は次のとおりです。

パートナ・アプリケーションの登録 : 登録方法

Single Sign-On パートナ・アプリケーションは mod_osso に統合され、mod_osso は OracleAS インストーラによって自動的に登録されます。したがって、パートナ・アプリケーションは mod_osso を介して登録されます。このモジュールを登録すると、そのエントリが ID 管理インフラストラクチャ・データベースとパートナ・アプリケーション・コンピュータの両方に作成されます。

mod_osso に統合されたアプリケーションは、プラットフォームが UNIX か Windows かによって、`ssoreg.sh` スクリプトまたは `ssoreg.bat` バッチ・ファイルで登録します。OracleAS Portal は、mod_osso ではなく SDK でシングル・サインオンを有効にするアプリケーションで、`ptlconfig` スクリプトで登録します。これら 3 つのツールは、すべてインストーラで起動します。ここでは、`ssoreg.sh` および `ssoreg.bat` について説明します。`ptlconfig` の使用方法は、『Oracle Application Server Portal 構成ガイド』のこのツールに関する付録を参照してください。

mod_osso の登録

特定の状況下では、シングル・サインオン登録ツールを使用して mod_osso を手動で登録する必要があります。たとえば、OracleAS のインストール後に、インフラストラクチャまたはアプリケーション層どちらかの Oracle HTTP Server のホスト名とポート番号が変更されている場合です。または、インストール後に SSL が有効にされているケースもこれに該当します。

シングル・サインオン登録ツールを実行すると、`osso.conf` の mod_osso 登録レコードが更新されます。このファイルは、登録ツールを実行するたびに作成されます。

この項の項目は次のとおりです。

ossoreg の構文とパラメータ

`ssoreg.sh` スクリプトと `ssoreg.bat` スクリプトは、同じパラメータを使用します。ここでは、2 つのコマンドの構文を記載します。スクリプトを実行する前に、`ORACLE_HOME` 環境変数を OracleAS がインストールされているディレクトリに設定します。詳細は「コマンド例」を参照してください。例では、ホーム・ディレクトリは `gitm1` という名前です。

次にコマンドを示します。

- UNIX:

```
$ORACLE_HOME/sso/bin/ssoreg.sh
-oracle_home_path orcl_home_path
-site_name site_name
-config_mod_osso TRUE
-mod_osso_url mod_osso_url
[-virtualhost]
[-update_mode CREATE | DELETE | MODIFY]
[-remote_midtier]
[-config_file config_file_path]
[-admin_info admin_info]
[-admin_id adminid]
```

- Windows:

```
%ORACLE_HOME%¥sso¥bin¥ssoreg.bat
-oracle_home_path orcl_home_path
-site_name site_name
-config_mod_osso TRUE
-mod_osso_url mod_osso_url
[-virtualhost]
[-update_mode CREATE | DELETE | MODIFY]
[-remote_midtier]
[-config_file config_file_path]
[-admin_info admin_info]
[-admin_id adminid]
```

oracle_home_path
Oracle ホームの絶対パスです。

site_name
サイト名です。通常は、パートナ・アプリケーションの有効なホスト名とポートが使用されます。たとえば、application.mydomain.com のように指定します。

config_mod_osso
TRUE に設定すると、登録するアプリケーションが mod_osso であることがこのパラメータで指定されます。osso.conf を生成するには、config_mod_osso を挿入する必要があります。

mod_osso_url
パートナ・アプリケーションの有効な URL です。この URL は、パートナ・アプリケーションへのアクセスに使用されます。値は、次の URL 形式で指定します。

http://*oracle_http_host*.*domain*:*port*

例 :

http://application.mydomain.com:7777

パートナの Oracle HTTP Server が HTTP のデフォルトのポート 80 または HTTPS のデフォルトのポート 443 をリスニングする場合は、ポート番号を省略します。

virtualhost
オプション。このパラメータは、Oracle HTTP 仮想ホストを Single Sign-On Server に登録する場合にのみ使用します。仮想ホストを登録しない場合は省略します。

HTTP 仮想ホストを作成する場合は、httpd.conf ファイルを使用して、保護された URL ごとに次のディレクティブを入力します。

```
<VirtualHost host_name>
  OssoConfigFile $ORACLE_HOME/Apache/Apache/conf/osso/host_name/osso.conf
  OssoIpCheck off
  #<Location /your_protectedORACLE_HOME_url>
  #  AuthType basic
     Require valid-user
  #</Location>
  #Other configuration information for the virtual host
</VirtualHost>
```

一方、HTTPS 仮想ホストを作成する場合は、ssl.conf ファイルを使用して同じディレクティブを入力します。コメント行は、アプリケーションを配置する前に削除する必要があります。httpd.conf と ssl.conf は、どちらも *ORACLE_HOME*/Apache/Apache/conf に格納されています。

仮想ホストを作成したら、次のコマンドを実行して、Distributed Cluster Management スキーマを更新します。

ORACLE_HOME/dcm/bin/dcmctl updateConfig -v -d

update_mode

オプション。パートナ登録レコードの作成、削除および変更を行います。CREATE はレコードを新規に作成します（デフォルト）。DELETE は既存のレコードを削除します。MODIFY は、既存のレコードを削除して新規にレコードを作成します。

remote_midtier

登録する mod_osso パートナ・アプリケーションがリモートの中間層にあることを指定します。このオプションは、構成する mod_osso パートナ・アプリケーションが別の ORACLE_HOME にあり、OracleAS Single Sign-On Server が現在の ORACLE_HOME でローカルに実行される場合のみ指定します。

config_file

osso.conf ファイルの場所です。パスは、通常、<ORACLE_HOME_PATH>/Apache/Apache/conf/osso/<filename>、またはその下のサブディレクトリになります。このパラメータは、-virtualhost または -remote_midtier が指定されている場合を除いてオプションになります。

- -virtualhost が指定されている場合は、構成する仮想ホストの osso.conf ファイルの場所になります。

  たとえば、次のように指定できます。

  ORACLE_HOME/Apache/Apache/conf/osso/*virtual_host_name*/osso.conf

  仮想ホストを登録する場合、このパラメータは必須です。config_file を省略した場合、非仮想ホストであるとみなされます。この場合、ssoreg によって、osso.conf という名前のファイルが *ORACLE_HOME*/Apache/Apache/conf/osso に作成されます。

- -remote_midtier が指定されている場合、パートナ・アプリケーションはリモートの中間層にあります。そのため、osso.conf 構成ファイルは、リモートの中間層にコピーまたは転送（FTP）されます。

admin_info

オプション。mod_osso 管理者のユーザー名です。このパラメータを省略すると、「パートナ・アプリケーションの編集」ページの「管理者の情報」フィールドが空白のままになります。

admin_id

オプション。電子メール・アドレスなどの管理者の追加情報です。このパラメータを省略すると、「パートナ・アプリケーションの編集」ページの「管理者の電子メール」フィールドが空白のままになります。

コマンド例

次のコマンド・シーケンスは、mod_osso インスタンスを Single Sign-On Server に登録します。例では、Oracle ホームは gitm1 という名前のディレクトリです。

- UNIX（csh および tcsh）：

```
setenv ORACLE_HOME /private/oracle/gitm1

$ORACLE_HOME/sso/bin/ssoreg.sh
-oracle_home_path $ORACLE_HOME
-site_name myhost.mydomain.com
-config_mod_osso TRUE
-mod_osso_url http://myhost.mydomain.com
```

- UNIX（Bourne および tcsh）：

```
ORACLE_HOME=/private/oracle/gitm1; export ORACLE_HOME

$ORACLE_HOME/sso/bin/ssoreg.sh
-oracle_home_path $ORACLE_HOME
-site_name myhost.mydomain.com
-config_mod_osso TRUE
-mod_osso_url http://myhost.mydomain.com
```

- Windows:

```
set ORACLE_HOME=c:¥private¥oracle¥gitm1

%ORACLE_HOME%¥sso¥bin¥ssoreg.bat
-oracle_home_path %ORACLE_HOME%
-site_name myhost.mydomain.com
-config_mod_osso TRUE
-mod_osso_url http://myhost.mydomain.com
```

Oracle HTTP Server の再起動

ssoreg を実行した後、Oracle HTTP Server を再起動します。

ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server

ロード・バランサを使用した複数のパートナ・アプリケーションの配置

可用性の高い配置で複数のパートナ・アプリケーション・インスタンスを構成するには、これらのインスタンスをインストールする前にロード・バランサを配置します。ロード・バランサによって、複数のパートナ・アプリケーションを 1 つのアドレスで公開でき、また実際にリクエストを処理するアプリケーション・サーバーの障害に備えることができます。HTTP ロード・バランサは、Oracle HTTP Server インスタンスの 1 つで発生した障害を検出し、別のインスタンスにリクエストをフェイルオーバーできます。

ここに示す使用例では、ロード・バランサを利用するパートナ・アプリケーションの構成に必要な手順を説明します。

使用例

この使用例では、次の架空の構成を前提としています。

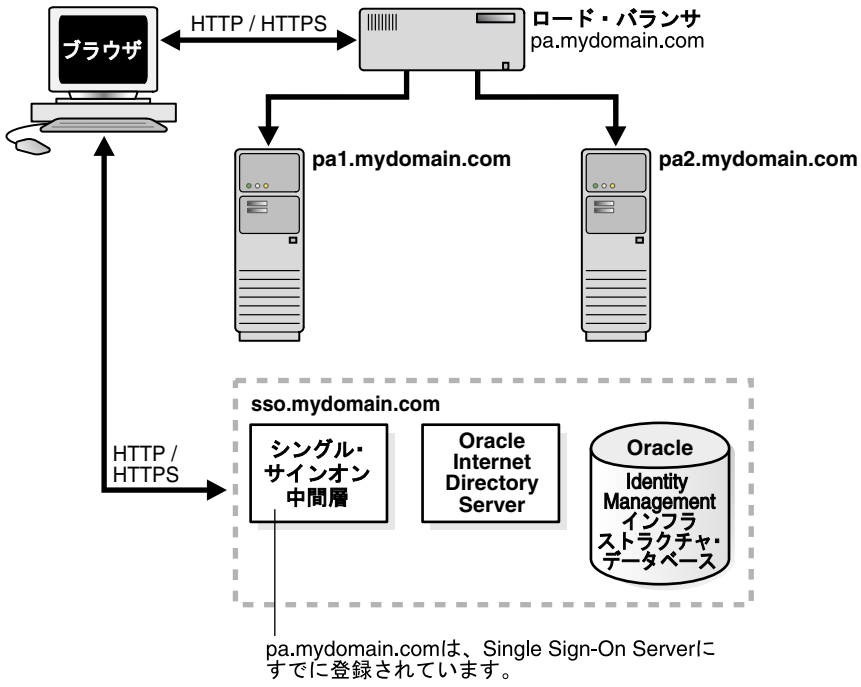
- pa1.mydomain.com と pa2.mydomain.com の 2 台のパートナ・アプリケーション・コンピュータがあります。これらのアプリケーション・サーバーは、非 SSL ポート 7777 をリスニングしています。
- これらのパートナ・アプリケーション・コンピュータは、sso.mydomain.com にある Single Sign-On Server を使用するように構成されています。
- ユーザーに公開されているパートナ・アプリケーションの有効なホスト名は pa.mydomain.com です。HTTP ロード・バランサは、このアドレス（ポート 80）をリスニングするように構成されています。HTTP ロード・バランサは負荷を分散し、ユーザーのリクエストのフェイルオーバーを pa1.mydomain.com と pa2.mydomain.com の間で行います。
- Single Sign-On Server、ディレクトリ・サーバーおよび ID 管理インフラストラクチャ・データベースは、sso.mydomain.com に配置されています。

注意：

- この使用例では、ロード・バランサが非 SSL ポート番号のポート 80 をリスニングします。ロード・バランサが SSL を使用してブラウザと対話するように構成されている場合は、別のポート番号を選択する必要があります。デフォルトの SSL ポート番号は 443 です。
- 2 台のパートナ・アプリケーション・コンピュータが配置されます。実際には、任意の数のコンピュータを使用できます。
- 使用されているホスト名は、あくまでも例です。これらの名前は、実際の実装では機能しない場合があります。実際のインストールでは、該当する値に置き換えてください。

図 4-1 は、この架空システムの構成を示しています。

図 4-1 複数のパートナ・アプリケーションで使用するロード・バランサ



構成手順

図 4-1 に示すシステムを設定するには、次の作業を行います。

- パートナ・アプリケーションのインストール
- パートナ・アプリケーション中間層での Oracle HTTP Server の構成
- HTTP ロード・バランサの構成
- パートナ・アプリケーション中間層での mod_osso の再登録

パートナ・アプリケーションのインストール

パートナ・アプリケーションを pa1.mydomain.com および pa2.mydomain.com にインストールします。インストーラがディレクトリの場所を要求したら、sso.mydomain.com に配置されたサーバーを選択します。

注意： ここで紹介するパートナ・アプリケーションは、任意の Web ベース・アプリケーションと置き換えて考えることができます。単純な例としては、Oracle HTTP Server と OC4J を含む OracleAS コア・インストールなどがあります。各アプリケーションのインストール・マニュアルを参照してください。

パートナ・アプリケーション中間層での Oracle HTTP Server の構成

OracleAS 中間層で、ユーザーと Oracle HTTP Server の間にロード・バランサを設置すると、パートナ・アプリケーションの有効な URL が変更されます。両方の中間層の構成ファイル httpd.conf を修正し、この変更を反映する必要があります。このファイルは *ORACLE_HOME*/Apache/Apache/conf にあります。

次の手順を実行します。

1. OracleAS 中間層で、Oracle HTTP Server を変更して、外部に公開する名前（使用例では pa.mydomain.com）をリスニングします。

   pa1.mydomain.com および pa2mydomain.com の httpd.conf ファイルに、次の行を追加します。

   ```
   ServerName pa.mydomain.com
   Port 80
   ```

   注意： httpd.conf に複数のポートが記述されている場合、変更されるポートは必ず最後のものになります。

   ---

2. ブラウザとロード・バランサの間に SSL を構成し、SSL 接続がロード・バランサで終了する場合は、pa1.mydomain.com と pa2.mydomain.com の両方に mod_certheaders を構成します。このモジュールによって、Oracle HTTP Server では、HTTP で受信するリクエストを SSL リクエストとして処理できるようになります。次の行を httpd.conf に追加します。この行は、ファイルの最後に追加できます。ファイル内での行の場所は重要ではありません。

   a. 次の行を入力し、モジュールをロードします。

      ```
      LoadModule certheaders_module libexec/mod_certheaders.so
      ```

   b. OracleAS Web Cache をロード・バランサとして使用する場合は、次の行を入力します。

      ```
      AddCertHeader HTTPS
      ```

      ハードウェア・ロード・バランサを使用する場合は、次の行を入力します。

      ```
      SimulateHttps on
      ```

HTTP ロード・バランサの構成

使用する HTTP ロード・バランサは、ハードウェアまたはソフトウェアのどちらでも構いません。ソフトウェアのロード・バランサを使用する場合は、OracleAS Web Cache を使用できます。

- ハードウェア・ロード・バランサ

  ハードウェアのロード・バランサを使用する場合は、実サーバーの 1 つのプールをアドレス pa1.mydomain.com および pa2.mydomain.com で構成します。1 つの仮想サーバーをアドレス pa.mydomain.com で構成します。この仮想サーバーは、ロード・バランサの外部インタフェースです。構成手順の詳細は、ロード・バランサのベンダーが提供するドキュメントを参照してください。

- ソフトウェア・ロード・バランサ

  接続リクエストのロード・バランスに OracleAS Web Cache を使用する場合は、次のドキュメントを参照してください。

  - 『Oracle Application Server Web Cache 管理者ガイド』の「Oracle Identity Management インフラストラクチャの強化」
  - 『Oracle Application Server Web Cache 管理者ガイド』の「Single Sign-On Server リクエストのルーティング」

注意： 最高のパフォーマンスを得るには、ハードウェア・ロード・バランサを使用してください。

パートナ・アプリケーション中間層での mod_osso の再登録

mod_osso を両方のパートナ・アプリケーション・インスタンスで、パートナ・アプリケーション pa.mydomain.com として登録します。

pa1.mydomain.com で mod_osso を登録するには、登録スクリプト ssoreg を実行します。ツールの実行方法は、この章の前半の「mod_osso の登録」の項を参照してください。この例では、スクリプトによって、pa.mydomain.com という名前のパートナ・アプリケーションが作成されます。

注意： パートナ・アプリケーション・コンピュータを Distributed Cluster Management 用に構成する場合は、以降の手順を省略してください。かわりに、pa1.mydomain.com で次のコマンドを実行します。

ORACLE_HOME/dcm/bin/dcmctl updateConfig -v -d

このコマンドにより、Distributed Cluster Management リポジトリに加えた変更が保存されるため、OracleAS 構成ファイルのバックアップ機能として利用できます。

---

mod_osso を pa2.mydomain.com に登録する手順は次のとおりです。

1. pa2.mydomain.com で、Single Sign-On 管理者としてシングル・サインオン管理ページにログインします。次の URL にログインしてください。

   http://sso.mydomain.com/sso

2. 「パートナ・アプリケーションの管理」ページを使用して、パートナ・アプリケーション pa2.mydomain.com の既存エントリを削除します。

3. pa1.mydomain.com から osso.conf ファイルをコピーします。ファイルを FTP で転送する場合は、バイナリ・モードを使用してください。このファイルのデフォルトのディレクトリは、*ORACLE_HOME*/Apache/Apache/conf/osso です。

4. Distributed Cluster Management リポジトリとコピーしたファイルを同期化します。これには、pa2.mydomain.com で次のコマンドを実行します。

   ORACLE_HOME/Apache/Apache/bin/ssotransfer
   ORACLE_HOME/Apache/Apache/conf/osso/osso.conf

   注意： ssotransfer コマンドは、Distributed Cluster Management リポジトリと仮想ホストに作成された mod_osso 構成ファイルとの同期化には使用しないでください。仮想ホストの mod_osso を登録する方法については、「仮想ホストでの mod_osso の構成（SSL および非 SSL）」を参照してください。

5. Oracle HTTP Server を再起動します。

   ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server

6. 次の有効な URL を使用して、パートナ・アプリケーションをテストします。

   http://pa.mydomain.com

パートナ・アプリケーションを mod_osso に統合する方法は、『Oracle Identity Management アプリケーション開発者ガイド』のシングル・サインオン対応のアプリケーションの開発に関する章を参照してください。

仮想ホストでの mod_osso の構成（SSL および非 SSL）

1 つの Oracle HTTP Server に複数の Web サイトを配置する必要がある場合があります。

次の例では、SSL 仮想ホストが mod_osso で保護されるように構成しています。この場合の仮想ホストは SSL ホストですが、この例はどの仮想ホストにも適用されます。

この例では、次の条件を前提としています。

- アプリケーション中間層のホスト名が app.mydomain.com であること。
- 中間層がすでに非 SSL パートナ・アプリケーションとして構成されていること。これは通常、アプリケーションを最初にインストールするときに、OracleAS インストーラによって行われます。
- アプリケーション中間層のデフォルトの SSL ポート番号が 4443 であること。

app.mydomain.com を SSL 仮想ホストとして構成するには、次の手順を実行します。

1. Oracle Identity Management のコンポーネント（特に、Oracle Internet Directory と Single Sign-On Server）が実行されていることを確認します。
2. app.mydomain.com が SSL 仮想ホストとして定義されていることを確認します。これは、OracleAS インストーラによって、ssl.conf ファイルの Virtual Host Context セクションに定義されています。このファイルは *ORACLE_HOME*/Apache/Apache/conf にあります。
3. SSL サイトにパートナ・アプリケーションを作成します。
   a. 中間層の Oracle ホームが正しいパスで設定されていることを確認します。詳細は、「mod_osso の登録」のコマンド例を参照してください。
   b. 中間層で次のコマンドを実行します。
      - UNIX:

```
$ORACLE_HOME/sso/bin/ssoreg.sh
-oracle_home_path $ORACLE_HOME
-site_name app.mydomain.com
-config_mod_osso TRUE
-mod_osso_url https://app.mydomain.com:4443
-virtualhost
-config_file $ORACLE_HOME/Apache/Apache/conf/osso/osso-https.conf
```

- Windows:

```
%ORACLE_HOME%¥sso¥bin¥ssoreg.bat
-oracle_home_path %ORACLE_HOME%
-site_name app.mydomain.com
-config_mod_osso TRUE
-mod_osso_url https://app.mydomain.com:4443
-virtualhost
-config_file $ORACLE_HOME/Apache/Apache/conf/osso/osso-https.conf
```

4. *ORACLE_HOME*/Apache/Apache/conf にある mod_osso.conf ファイルに移動します。このファイルを開いたら、次のディレクティブをコメントアウトします。

   - UNIX:

     ```
     LoadModule osso_module libexec/mod_osso.so
     ```

   - Windows:

     ```
     LoadModule osso_module modules¥ApacheModuleOsso.dll
     AddModule mod_osso.c
     ```

5. conf ディレクトリにある httpd.conf に、前の手順でコメントアウトしたディレクティブを追加します。デフォルトの設定の場合、LoadModule wchandshake_module libexec/mod_wchandshake.so の真下にディレクティブを追加します。

6. Oracle HTTP Server を再起動します。

   ```
   ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
   ```

7. 同じく conf ディレクトリにある ssl.conf で、VirtualHost に仮想ホストの osso.conf ファイルを追加します。デフォルトの osso.conf ファイルとの競合を避けるため、ファイルは osso-https.conf という名前で保存します。ファイル名が、登録スクリプトで使用されている名前と同じであることを確認します。

   ```
   <VirtualHost _default_:4443>
   .
   .
   .
   OssoConfigFile ORACLE_HOME/Apache/Apache/conf/osso/osso-https.conf
   OssoIpCheck off
   <Location /your_protected_url_for_the virtual site>
     AuthType basic
     Require valid-user
   </Location>
   .
   .
   .
   </VirtualHost>
   ```

8. Distributed Cluster Management リポジトリを更新します。

   ```
   ORACLE_HOME/dcm/bin/dcmctl updateConfig -v -d
   ```

9. アプリケーション中間層で Oracle HTTP Server を再起動します。

   ```
   ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
   ```

10. SSL と非 SSL サイトの両方をテストします。

5

外部アプリケーションの設定と管理

この章では、外部アプリケーションをシングル・サインオン対応に構成する方法について説明します。これらのアプリケーションは、Single Sign-On Server に認証を委譲するように変更されていない Web アプリケーションです。Web アプリケーションを外部アプリケーションとして構成すると、インタフェースを変更せずにそのアプリケーションのシングル・サインオンを有効にできます。これらのアプリケーションの詳細は、第 1 章の「外部アプリケーション」の項を参照してください。

この章の項目は次のとおりです。

インタフェースを使用した外部アプリケーションの配置と管理

外部アプリケーションを追加、編集または削除するには、「SSO Server 管理」ページのリンクから、「外部アプリケーションの管理」ページにアクセスします。追加した外部アプリケーションは、OracleAS Portal の「外部アプリケーション」ポートレットでアクセスできます。このポートレットは、OracleAS をインストールすると、「Portal」ページに追加できます。ページの表示およびカスタマイズの詳細は、『Oracle Application Server Portal ユーザーズ・ガイド』を参照してください。

この項の項目は次のとおりです。

外部アプリケーションの追加

「Single Sign-On Server 管理」ページで「外部アプリケーション管理」リンクをクリックし、「外部アプリケーションの追加」リンクをクリックすると、「外部アプリケーションの追加」ページが表示されます。このページには、次のヘッダーとフィールドが含まれています。

表 5-1 外部アプリケーション・ログイン

フィールド	説明
アプリケーション名	外部アプリケーションを識別する名前を入力します。この名前は、外部アプリケーションのデフォルト名になります。
ログイン URL	外部アプリケーションの HTML ログイン・ページを認証する送信先 URL を入力します。たとえば、Yahoo! Mail のログイン URL は、次のようになります。 `http://login.yahoo.com/config/login?6p4f5s403j3h0`
ユーザー名 /ID フィールド名	外部アプリケーションの HTML ログイン・フォームのユーザー名またはユーザー ID フィールドを識別する文字列を入力します。この文字列は、フォームの HTML ソースを表示するときに使用されます。(後続の手順に続く例を参照)。このフィールドは、Basic 認証を使用している場合は適用できません。
パスワード・フィールド名	アプリケーションの HTML ログイン・フォームのパスワード・フィールドを識別する文字列を入力します。この文字列は、フォームの HTML ソースを表示するときに使用されます。(後続の手順に続く例を参照)。このフィールドは、Basic 認証を使用している場合は適用できません。

表 5-2 認証方式

フィールド	説明
使用する認証タイプ	プルダウン・メニューから、アプリケーションで使用するフォーム送信方法を選択します。これによって、ブラウザからメッセージ・データを送信する方法が決まります。この文字列は、ログイン・フォームの HTML ソースを表示するときに使用されます。次の 3 つの方法の 1 つを選択します。 POST: Single Sign-On Server にデータを転送して、フォーム本体内のログイン資格証明を送信します。 GET: サーバーにページ・リクエストを提示して、ログイン URL の一部としてログイン証明書を送信します。 Basic 認証 : アプリケーション URL 内のログイン資格証明を送信します。この送信は、HTTP Basic 認証で保護されます。 **注意 :** ■ Basic 認証で使用するポップアップ・ウィンドウは、Windows XP のサービス・パック 2 ではデフォルトでブロックされます。このサービス・パックを使用する場合、シングル・サインオン用ログイン・ページのウィンドウを表示するよう、ブラウザを設定しなおしてください。これは、Internet Explorer の「ツール」メニューにある「ポップアップ・ブロック」で行います。 Mozilla など、他のブラウザおよびブラウザ・プラグインでも、ポップアップのブロックが可能です。該当するブラウザの場合、シングル・サインオンのログイン・ページがブロックされないようにしてください。 ■ Internet Explorer 5.0 以上を使用する場合、外部アプリケーションで Basic 認証が機能しないことがあります。このバージョンの Internet Explorer には、Microsoft MS04-004 Cumulative Security Update（832894）が含まれます。対処方法は、次のサイトを参照してください。 `http://support.microsoft.com`

表 5-3 追加フィールド

フィールド	説明
フィールド名	ログイン時にユーザー入力を要求するフィールドを HTML ログイン・フォームに追加した場合は、そのフィールドの名前を入力します。このフィールドは、Basic 認証を使用している場合は適用できません。
フィールド値	対応するフィールド名のデフォルト値を入力します（該当する場合）。このフィールドは、Basic 認証を使用している場合は適用できません。

外部アプリケーションを追加する手順 :

1. 「外部アプリケーションの管理」ページから、「外部アプリケーションの追加」を選択します。

   「外部アプリケーションの追加」ページが表示されます。

2. 「外部アプリケーション・ログイン」フィールドに、外部アプリケーション名と HTML ログイン・フォームの送信先 URL を入力します。Basic 認証を使用する場合は、保護された URL を入力します。

3. アプリケーションで HTTP POST 認証または HTTP GET 認証が使用されている場合は、「ユーザー名 /ID フィールド名」フィールドに、HTML ログイン・フォームのユーザー名またはユーザー ID フィールドを識別する文字列を入力します。

   この名前は、ログイン・フォームの HTML ソースを表示するときに使用されます。

   アプリケーションで Basic 認証方式が使用されている場合は、「ユーザー名 /ID フィールド名」フィールドを空にします。

4. アプリケーションで HTTP POST 認証または HTTP GET 認証が使用されている場合は、「パスワード・フィールド名」フィールドに、アプリケーションのパスワード・フィールドを識別する文字列を入力します。

   ログイン・フォームの HTML ソースを参照してください。

   アプリケーションで Basic 認証方式が使用されている場合は、「パスワード・フィールド名」フィールドを空にします。

5. ログイン時にユーザー入力を要求するフィールドを HTML ログイン・フォームに追加した場合は、「追加フィールド」フィールドに、そのフィールドの名前とデフォルト値を入力します。

   アプリケーションで Basic 認証方式が使用されている場合は、このフィールドを空にします。

6. HTML ログイン・フォームでユーザーが追加フィールドのデフォルト値を変更できるようにする場合は、「ユーザーに表示」チェック・ボックスを選択します。

7. 「OK」をクリックします。新しい外部アプリケーションが、「外部アプリケーションの管理」ページの「外部アプリケーションの編集 / 削除」ヘッダーの下に、その他の外部アプリケーションとともに表示されます。

8. アプリケーションのリンクをクリックして、ログインをテストします。

次の例は、Yahoo! Mail で使用される値のソースです。

```
<form method=post action="http://login.yahoo.com/config/login?6p4f5s403j3h0"
autocomplete=off name=a>
...
<td><input name=login size=20 maxlength=32></td>
....
<td><input name=passwd type=password size=20 maxlength=32></td>
...
<input type=checkbox name=".persistent" value="Y" >Remember my ID & password
...
</form>
```

このソースでは、次の要素の値を指定しています。

- ログイン URL:

  `http://login.yahoo.com/config/login?6p4f5s403j3h0`

- ユーザー名 /ID フィールド名 : `login`
- パスワード・フィールド名 : `passwd`
- 使用する認証タイプ : `POST`
- フィールド名 : `.persistent Y`
- フィールド値 : `[off]`

注意 : AS 中間層のホスト名を変更する場合、この層の外部アプリケーションについてログイン URL フィールドを手動で更新する必要があります。この変更は、次の項で説明する「外部アプリケーションの編集」ページで行います。

外部アプリケーションの編集

アプリケーションの横にある鉛筆アイコンをクリックすると、「外部アプリケーションの編集」ページが表示されます。ここで、アプリケーションを追加したときに入力した値を編集できます。編集を終了したら**「適用」**をクリックし、変更を入力して新しい値で再度ページを表示します。

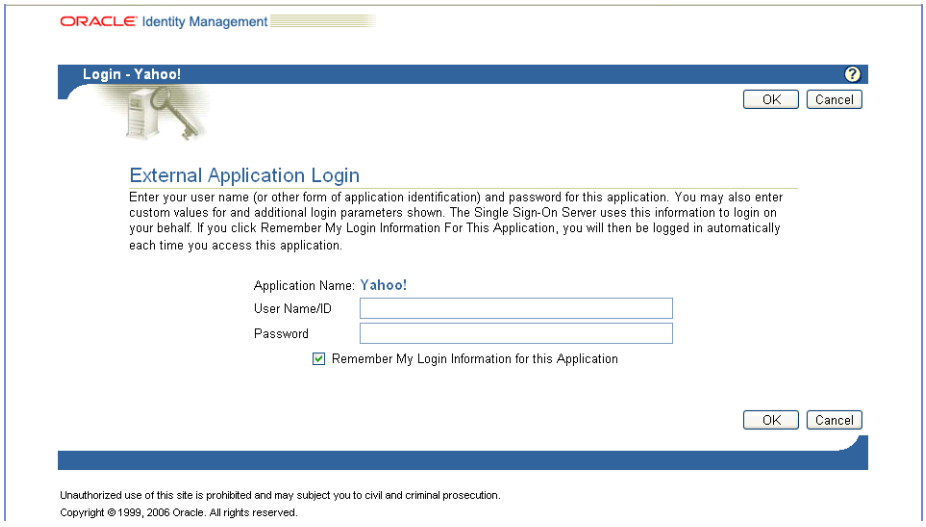
Single Sign-On データベースへの外部アプリケーション証明書の格納

ユーザーがアプリケーションにログインするたびに、それぞれの外部アプリケーションでは、ユーザー名とパスワードの受信を待機しています。これらのアプリケーションへのシングル・サインオンを有効にするには、ログイン時に、資格証明を Single Sign-On データベースに保存するように指定できます。

Single Sign-On ユーザーが初めて外部アプリケーションにログインすると、「外部アプリケーション・ログイン」ページが表示されます。資格証明書を入力した後、「このアプリケーションのログイン情報を保存する」チェック・ボックスを選択できます。このオプションを選択すると、次回アプリケーションにアクセスするときは、Single Sign-On Server がログインを代行します。

図 5-1 は、「外部アプリケーション・ログイン」ページを再現したものです。

図 5-1「外部アプリケーション・ログイン」ページ



注意：

- パスワードを変更した場合、「外部アプリケーション・ログイン」ページのパスワードも更新する必要があります。更新しないと、ログインしようとしたときに、このページからエラー・メッセージが返されます。
- パスワードには、&、{、}、<、>、"、'、（および）の各文字は使用できません。

Basic 認証アプリケーションのプロキシ認証

シングル・サインオン対応の外部アプリケーションには、SDK 対応パートナ・アプリケーションである OracleAS Portal の「外部アプリケーション」ポートレットを使用してアクセスするのが一般的です。この方法でアクセスするアプリケーションには、GET 認証、POST 認証または Basic 認証を構成できます。

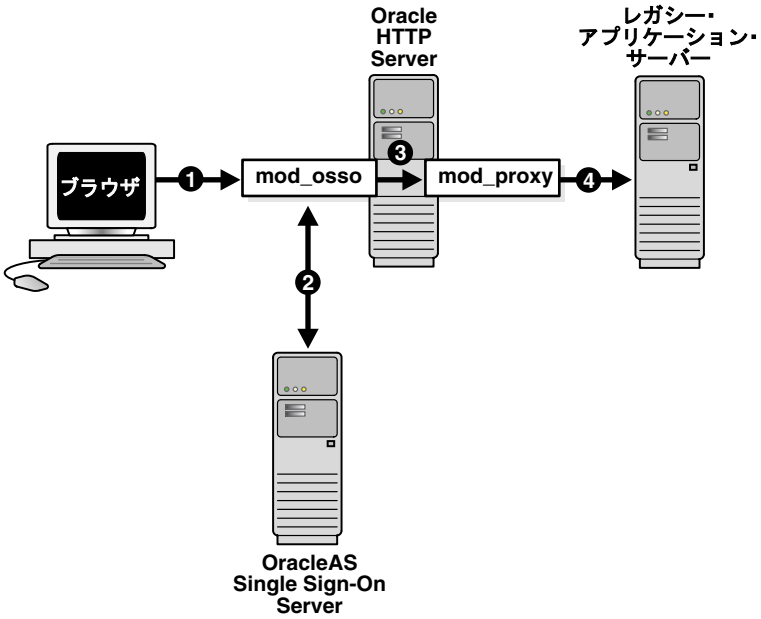
これにかわるものとして、別の Web サーバーにあるアプリケーションへのセキュアなプロキシとして Oracle HTTP Server を使用する方法があります。この方法では、モジュール mod_osso と mod_proxy を設定してシングル・サインオン対応の Basic 認証をサポートする必要があります。プロキシ認証の利点は、標準の方法で外部アプリケーションにアクセスしたときに発生する短時間の画面のちらつきがなくなることです。

この項の項目は次のとおりです。

Basic 認証のプロキシとしての Oracle HTTP Server の設定

適切に構成された mod_osso 対応の外部アプリケーションの認証は、パートナ・アプリケーションの認証と類似しています。すなわち、mod_osso は URL リクエストを取得して Single Sign-On Server にリダイレクトします。図 5-2 に、このプロセスを示します。

図 5-2 mod_osso/mod_proxy を使用した認証の流れ



1. Single Sign-On ユーザーは、ブックマークを選択するか仮想 URL を入力して外部アプリケーションをリクエストします。この仮想 URL によって、Oracle HTTP Server はリクエストを取得できます。
2. mod_osso は、取得したリクエストに認証ヘッダーを追加して、Single Sign-On Server からユーザーの資格証明を取得します。
3. mod_osso は、Single Sign-On Server から取得したユーザーの資格証明でヘッダー値を設定し、このヘッダー値を mod_proxy に渡します。
4. mod_proxy は、ユーザーの資格証明を Basic 認証ヘッダーのフォームで実 URL に渡します。この転送は、仮想 URL を実 URL にマップするディレクティブによって行われます。

構成の要件

Oracle HTTP Server でレガシー・アプリケーションの Basic 認証を構成するには、次の条件を満たす必要があります。

- プロキシ処理を使用するアプリケーションは、Basic 認証アプリケーションとして Single Sign-On Server に登録する必要があります。詳細は、「外部アプリケーションの追加」の項を参照してください。
- Oracle HTTP Server に mod_osso をインストールして有効にする必要があります。
- Oracle HTTP Server にデフォルトの mod_proxy をインストールして有効にする必要があります。
- 外部アプリケーションをホストする Web サーバーで Oracle HTTP Server がプロキシとして使用されている場合、その Web サーバーでは mod_osso を有効にできません。

構成手順

Oracle HTTP Server で外部アプリケーションの Basic 認証を構成するには、次の手順を実行します。

1. 次のセクションを、アプリケーション層の `mod_osso.conf` に追加します。このファイルは *ORACLE_HOME*`/Apache/Apache/conf` にあります。

```
<IfModule mod_proxy.c>
<Location /application_virtual_path>
   require valid-user
   AuthType Basic
   OssoLegacyApp on | off
</Location>

ProxyPass /application_virtual_path/ http://host:port/application_real_ path/
ProxyPassReverse /application_virtual_path/ http://host:port/application_real_path/
</IfModule>
```

   `OssoLegacyApp` ディレクティブは、保護された URL がレガシー・アプリケーションであるかどうかを示します。このディレクティブが見つからないか `off` に設定されている場合は、アプリケーションのユーザー名とパスワードを Single Sign-On データベースから取得するコードは実行されません。2 つの mod_proxy ディレクティブ `ProxyPass` と `ProxyPassReverse` によって、仮想 URL が実 URL にマップされます。

2. 次の行を `httpd.conf` に追加します。

```
Listen 5000
```

   このパラメータは、非 SSL ポート 5000 を使用して外部アプリケーションに関する情報にアクセスするように mod_osso に指定します。

3. Oracle HTTP Server を再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
```

4. Distributed Cluster Management スキーマを更新します。

```
ORACLE_HOME/dcm/bin/dcmctl updateConfig -v -d
```

注意：

- 仮想 URL のディレクトリを指定する必要はありません。便宜上、この URL はアプリケーション名のみで構成できます。
- SSL が有効な場合は、アプリケーションの実 URL の `http` を `https` に置き換えます。

6

マルチレベル認証

このドキュメントでは、各種のパートナ・アプリケーションに異なる認証レベルを割り当てるシングル・サインオン・システムの構成方法について説明します。このシステムでは、リクエストされたアプリケーションのセキュリティ・レベルに認証動作を合せることができます。

ドキュメントの項目は次のとおりです。

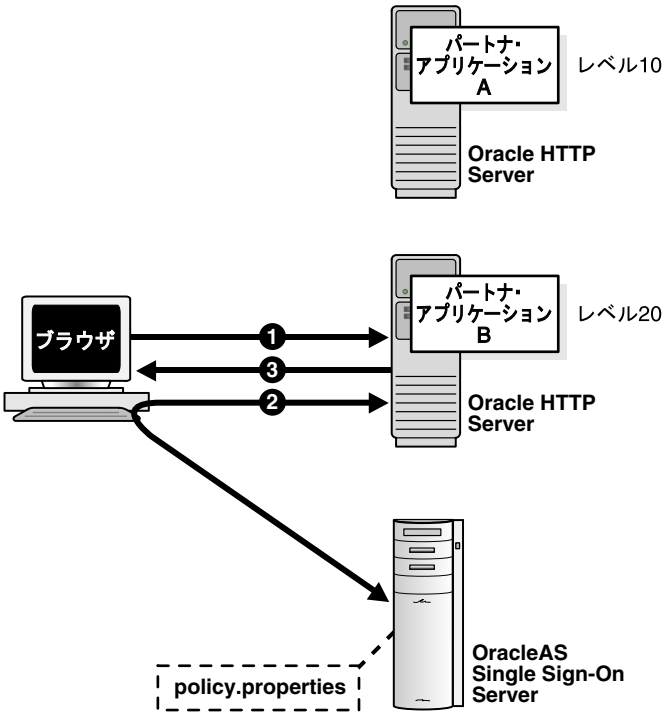
マルチレベル認証とは

OracleAS Single Sign-On では、保護するアプリケーションに異なる認証レベルを割り当てることができます。次に、これらの認証レベルを特定の認証プラグインにマップできます。たとえば、セキュリティが重視されるアプリケーションではユーザー証明書を要求するように構成し、セキュリティがさほど重要でないアプリケーションではユーザー名とパスワードを要求するように構成することもできます。

マルチレベル認証の仕組み

6-2 ページの図 6-1 にマルチレベル認証の仕組みを示します。

図 6-1 マルチレベル認証の流れ



1. ユーザーは、アプリケーション A ですでに認証されています。このユーザーは、次にアプリケーション B に移動します。
2. アプリケーション B は、このユーザーを Single Sign-On Server にリダイレクトします。
3. アプリケーション B の認証レベルはアプリケーション A よりも高いため、このユーザーは、より高いレベルの資格証明による再認証を Single Sign-On Server から要求されます。

注意： リリース 10.1.4 では、認証はパートナ・アプリケーションのルート・レベルで行われます。ルートの下の URL に認証レベルを割り当てることはできません。

マルチレベル・システムのコンポーネント

次の項目は、マルチレベル認証の仕組みを理解するためのキー・ポイントです。

- 認証レベル
- 認証プラグイン

認証レベル

認証レベルは、各アプリケーションに異なる認証動作を指定するためのパラメータです。*ORACLE_HOME*/sso/conf/policy.properties ファイルでは、6 つの認証レベルが定義されています。このファイルには認証レベルの名前と値が含まれます。付録 C にこのファイルのコピーがあります。

表 6-1 に認証レベルの例を示します。これらの認証レベルは、カスタマイズしたり新しく作成することができます。

表 6-1 デフォルトの認証レベル

認証レベル名	認証レベル値	説明
LowSecurity	20	デフォルト値は低レベルな認証に使用されます。
LowMediumSecurity	30	この値は、通常、カスタム認証モジュールに使用されます。
MediumSecurity	40	MediumSecurity のデフォルト値は、ユーザー名とパスワードによる認証です。
MediumHighSecurity	50	デフォルト値は、証明書による認証が必要であることを示します。
HighSecurity	60	この値は、通常、カスタム認証モジュールに使用されます。

各セキュリティ・レベルには、関連付けられた名前、実装されているセキュリティ・レベルに対するプラグイン・パラメータを含む Java クラス、およびこのセキュリティ・レベルによって保護される各アプリケーションのホスト名とポートが含まれます。

認証レベルには任意の名前を指定できます。ただし、一意である必要があり、名前を変更する場合は、policy.properties 内の関連するすべての場所で反映する必要があります。たとえば、NoSecurity=10 と NoSecurity=20 の両方指定することはできません。認証レベルの数値が低くなるほどセキュリティ・レベルも低くなります。値には正の整数を指定します。現在の認証レベルを比較し、認証レベルを高くする必要があるかどうか確認する場合にこれらの値を使用します。

セキュリティ・レベルによって認証方式が決まります。プラグイン・パラメータを含む Java クラスへの URL で表します。

たとえば、大部分のパートナ・アプリケーションではユーザー名とパスワードによる認証を実行し、特定のパートナ・アプリケーションでは証明書による認証を実行する場合、policy.properties に次の行を追加できます。

```
partner_application_host.example_company.com\:7777 = MediumHighSecurity
MediumHighSecurity=oracle.security.soo.auth.SSOX509CertAuth
```

高いレベルでログインし、低いレベルのアプリケーションにアクセスしようとする場合は、資格証明を再度要求されません。低いレベルのアプリケーションにログインし、高いレベルのアプリケーションにアクセスしようとすると、高いレベルに設定された認証方式が使用されます。たとえば、MediumSecurity でログインしたユーザーは LowSecurity を必要とするアプリケーションにアクセスできますが、LowSecurity でログインしたユーザーは、MediumSecurity を必要とするアプリケーションにアクセスするには認証が必要です。

認証プラグイン

認証プラグインは、特定の認証方式を実装したものです。この方式によってユーザーの資格証明が収集され、ユーザーが認証されます。

前項で説明した認証レベルのいずれかと、次の箇条書きに示す認証方式のいずれかを組み合せることができます。認証プラグインがマップする認証レベルは配置固有です。この組合せを実現するには `policy.properties` を使用します。

- パスワード認証

  デフォルトの標準方式です。

- デジタル証明書

  証明書による認証の詳細は、第 8 章を参照してください。

- Windows ネイティブ認証

  詳細は、『Oracle Identity Management 統合ガイド』の Microsoft Active Directory との統合に関する章を参照してください。

マルチレベル認証の構成

アプリケーションに対して認証レベルを構成しないと、`policy.properties` ファイルの `DefaultAuthLevel` パラメータにより認証に使用するデフォルト・レベルが決定されます。

認証に使用されるデフォルトのプラグインは、デフォルト・レベルに関連付けられたプラグインです。たとえば、`DefaultAuthLevel` の値が `MediumSecurity` の場合、アプリケーションは `policy.properties` ファイルの `MediumSecurity` の定義で指定されたプラグインを使用します。

使用例

この使用例では、架空の 2 つのパートナ・アプリケーションで異なる認証レベルとプラグインを使用するように構成する方法について説明します。ここでは、次の条件を前提としています。

- アプリケーション pa1 は、ホスト `pa1.mydomain.com` に配置されています。ポート `7777` をリスニングします。
- pa1 は、Single Sign-On Server にすでに登録されています。
- pa1 では、証明書による認証を使用する必要があります。
- アプリケーション pa2 は、ホスト `pa2.mydomain.com` に配置されています。ポート `7777` をリスニングします。
- pa2 は、Single Sign-On Server にすでに登録されています。
- pa2 では、パスワード認証を使用する必要があります。

構成手順

次の構成を使用して policy.properties を変更します。

1. 認証レベルの名前を policy.properties ファイルから選択します。必要な場合は、新しい認証レベルと対応する名前をこのファイルに追加します。
2. 2 つのパートナ・アプリケーションのルート URL に、認証レベルを次のように割り当てます。

   ```
   pa1.mydomain.com\:7777 = HighSecurity
   pa2.mydomain.com\:7777 = MediumSecurity
   ```

   注意 : ドメイン名の後にバックスラッシュを挿入してください。

3. 手順 1 で割り当てた認証レベル名に、認証プラグインを次のように割り当てます。

   ```
   HighSecurity_AuthPlugin = oracle.security.sso.server.auth.SSOX509CertAuth
   MediumSecurity_AuthPlugin = oracle.security.sso.server.auth.SSOServerAuth
   ```

   認証プラグイン名は、手順 1 で割り当てた認証レベル名と接頭辞 _AuthPlugin を連結したものです。

4. policy.properties を保存し、シングル・サインオン中間層を再起動します。

   ```
   ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
   ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
   ```

5. パートナ・アプリケーションをテストします。

7

SSL の有効化

この章では、Single Sign-On Server で Secure Sockets Layer（SSL）を有効化する方法について説明します。SSL では、秘密のセッション鍵が作成されるため、セキュアなチャネルを介した情報交換が可能になります。ユーザーがログインすると、Web サーバーによりブラウザにデジタル証明書が送信されます。ブラウザでは、Web サーバーから送信された公開鍵を使用して乱数を暗号化します。この暗号化されたデータは、秘密鍵の作成に使用されます。Single Sign-On Server で SSL を有効化すると、この形式の保護がサーバーのパートナ・アプリケーションに適用されます。このプロセスの使用により、OracleAS は高度なセキュリティを確保できます。

デフォルトでは、Single Sign-On Server は Oracle HTTP Server の HTTP ポートを使用します。ただし、インストール後に SSL を自動または手動で構成できます。

自動 SSL 構成

一般的なトポロジでは、SSL 構成ツールを使用して、Oracle HTTP Server のインストール後、SSL を有効化するために必要な手順を実行できます。このツールの詳細と実行方法は、『Oracle Application Server 管理者ガイド』の SSL 構成ツールの使用に関する項を参照してください。

サーバーを監視する場合は、Enterprise Manager エージェントのインストール・ディレクトリにあるビーコン認証局証明書ファイル（b64InternetCertificate.txt）に、インフラストラクチャ・サーバーの証明書が含まれている必要があります。詳細は、『Oracle Enterprise Manager アドバンスト構成』を参照してください。特に、「Oracle Enterprise Manager のセキュリティ」、HTTPS 経由で Web アプリケーションを監視するためのビーコンの構成に関する項を参照してください。

注意： SSL 構成ツールを使用するには、SSL ポートを構成するときの制限事項を理解する必要があります。詳細は、7-6 ページの「SSL の構成に関する注意事項」を参照してください。

手動 SSL 構成

SSL を手動で有効化するには、次の作業を順番に実行します。

- シングル・サインオン中間層での SSL の有効化
- Identity Management インフラストラクチャ・データベースの再構成
- シングル・サインオン URL の保護
- Oracle HTTP Server とシングル・サインオン中間層の再起動
- パートナ・アプリケーションの登録
- mod_osso Cookie のセキュアな通信

注意： Oracle HTTP Server が SSL 用に構成されている場合（最初の項目）、Single Sign-On Server も SSL 用に構成する必要があります（残りの項目）。そうしないと、ユーザーはシングル・サインオン URL にアクセスできません。この制限を回避するには、HTTP 経由でアクセスする URL に対して、SSL ディレクティブを無効にします。そのためには、*ORACLE_HOME*`/sso/conf/sso_apache.conf` を編集します。

シングル・サインオン中間層での SSL の有効化

次の手順では Oracle HTTP Server を構成します。実行する際は、次の点に留意してください。

- SSL は、シングル・サインオン中間層を実行しているコンピュータ、つまり Single Sign-On Server をホストするコンピュータで構成する必要があります。
- SSL サーバーを構成します。
- 簡易ネットワークの暗号化に対して SSL を有効にします。PKI 認証は必要ありません。ただし、有効な Wallet およびサーバー証明書を使用する必要があります。デフォルトの Wallet は、*ORACLE_HOME*`/Apache/Apache/conf/ssl.wlt/default` にあります。別の Wallet を使用する場合は、第 8 章の「Oracle HTTP Server」の項に記載されたガイドラインを参照してください。『Oracle Application Server 管理者ガイド』の Wallet および証明書の管理に関する章も参照してください。

Oracle HTTP Server で SSL をすぐに使用できるようにする手順は次のとおりです。

1. *ORACLE_HOME*/opmn/conf にある、opmn.xml ファイルをバックアップします。
2. opmn.xml で、start-mode パラメータの値を ssl-enabled に変更します。このパラメータは、次のコードで xml タグに囲まれ太字で示されています。

```
<ias-component id="HTTP_Server">
   <process-type id="HTTP_Server" module-id="OHS">
       <module-data>
          <category id="start-parameters">
              <data id="start-mode" value="ssl-enabled"/>
          </category>
       </module-data>
   <process-set id="HTTP_Server" numprocs="1"/>
   </process-type>
</ias-component>
```

3. 次のコマンドを実行して、Distributed Cluster Management データベースを変更内容で更新します。

```
ORACLE_HOME/dcm/bin/dcmctl updateconfig -ct opmn
```

4. 次のように指定して、変更した opmn 構成ファイルをリロードします。

```
ORACLE_HOME/opmn/bin/opmnctl reload
```

5. OracleAS Single Sign-On Server と通信する Oracle HTTP Server の非 SSL ポートをアクティブな状態で保持します。

   外部アプリケーション・ポートレットが非 SSL ポートを介して Single Sign-On Server と通信します。HTTP ポートはデフォルトで有効になっています。無効になっている場合を除き、ここでの操作は必要ありません。ログインにこのポートを使用することはできません。詳細は、7-6 ページの「SSL の構成に関する注意事項」を参照してください。

6. Oracle HTTP Server を再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl startproc process-type=HTTP_Server
```

7. SSL のシングル・サインオン中間層が有効になっていることを確認します。これには、OracleAS の最初のページで、https://*host*:*ssl_port* の書式を使用してアクセスします。

注意： インストール環境に複数の中間層がある場合は、「シングル・サインオン中間層での Oracle HTTP Server の構成」の手順 2 が完了していることを確認してください。この手順は、第 9 章に示されている配列例の 1 つ、「複数のシングル・サインオン中間層、1 つの Oracle Internet Directory」に含まれています。

Identity Management インフラストラクチャ・データベースの再構成

ID 管理インフラストラクチャ・データベースを再構成するには、次の手順を実行します。

1. ID 管理インフラストラクチャ・データベース内の、シングル・サインオン URL に含まれる `http` をすべて `https` に変更します。
2. データベース内のシングル・サインオン URL を変更するときは、シングル・サインオン中間層でも `targets.xml` ファイル内の同じ URL を変更する必要があります。`targets.xml` は、Oracle Enterprise Manager で監視される様々なターゲットの構成ファイルです。そのターゲットの 1 つが OracleAS Single Sign-On です。
3. Oracle Enterprise Manager のセキュリティを構成します。

この手順は、後の項で説明します。

シングル・サインオン URL の変更

シングル・サインオン中間層が配置されているコンピュータで慎重にコマンドを入力して、`ssocfg` スクリプトを実行します。使用する構文は次のとおりです。

- UNIX:

  ```
  $ORACLE_HOME/sso/bin/ssocfg.sh protocol host ssl_port
  ```

- Windows:

  ```
  %ORACLE_HOME%¥sso¥bin¥ssocfg.bat protocol host ssl_port
  ```

この場合、*protocol* は `https` です。（HTTP に戻す場合は、`http` を使用します。）パラメータ *host* は、Single Sign-On Server の Oracle HTTP リスナーのホスト名（サーバー名）です。

次に例を示します。

```
ssocfg.sh https login.acme.com 4443
```

正しいポート番号を確認するには、`ssl.conf` ファイルを調べます。OracleAS のインストール時にインストーラによって割り当てられるポート番号は `4443` です。

`ssocfg` が問題なく実行されると、ステータス `0` が返されます。スクリプトが成功したことを確認するために、OC4J_SECURITY インスタンスを再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

次に、SSL アドレスにある Single Sign-On Server にログインを試みます。

```
https://host:ssl_port/sso/
```

targets.xml の更新

`ssocfg` を実行したら、シングル・サインオン中間層で `targets.xml` ファイルを更新します。

`targets.xml` を更新するには、次の手順を実行します。

1. ファイルをバックアップします。

   ```
   cp ORACLE_HOME/sysman/emd/targets.xml ORACLE_HOME/sysman/emd/targets.xml.backup
   ```

2. ファイルを開いて、ターゲット・タイプ `oracle_sso_server` を検索します。このターゲット・タイプ内で、`ssocfg` に渡した次の 3 つの属性を検索し、編集します。

   - `HTTPMachine`: HTTP Server のホスト名
   - `HTTPPort`: Oracle HTTP Server の SSL ポート番号
   - `HTTPProtocol`: サーバーのプロトコル

   たとえば、次のように `ssocfg` を実行したとします。

   ```
   ORACLE_HOME/sso/bin/ssocfg.sh https sso.mydomain.com:4443
   ```

この場合は、次のように 3 つの属性を更新します。

```
<Property NAME="HTTPMachine" VALUE="sso.mydomain.com"/>
<Property NAME="HTTPPort" VALUE="4443"/>
<Property NAME="HTTPProtocol" VALUE="HTTPS"/>
```

3. ファイルを保存して閉じます。
4. OracleAS Console をリロードします。

```
ORACLE_HOME/bin/emctl reload
```

Oracle Enterprise Manager のセキュリティの構成

Single Sign-On Server で SSL を有効にする場合は、『Oracle Enterprise Manager アドバンスト構成』の Oracle Enterprise Manager のセキュリティに関する章で説明されているすべての構成手順に従う必要があります。特に、HTTPS を介して Web アプリケーションを監視するためのビーコンの構成方法に関する項には細心の注意を払ってください。Oracle ビーコンは Enterprise Manager のアプリケーション・サービス・レベル管理機能の一部で、アプリケーションのパフォーマンスの可用性とパフォーマンス監視機能を提供します。ビーコンは、HTTPS URL を使用する、SSL を介した URL の監視に使されます。

シングル・サインオン URL の保護

Single Sign-On Server で SSL を有効にしている場合は、HTTP を介してサーバーにアクセスするホストに HTTP アクセスを制限するように指定する必要があります。これは、OracleAS インストーラおよび OracleAS Portal をホストするコンピュータに特に当てはまります。

この項では、次の手順について説明します。

- ロード・バランシング・ルータがない場合の URL の保護
- ロード・バランシング・ルータがある場合の URL の保護

ロード・バランシング・ルータがない場合の URL の保護

Single Sign-On Server および OracleAS Portal のフロントエンドにロード・バランシング・ルータが配置されていない場合は、次の手順に従ってください。

ORACLE_HOME/sso/conf/sso_apache.conf で、次のディレクティブを検索してそのコメントを解除し、次に Allow from パラメータに値を指定します。

OracleAS Portal では、外部アプリケーションのリストを表示する URL に HTTP を介してアクセスする必要があります。このアクセスは、次のディレクティブで可能になります。<your_domain_name> を Portal の完全修飾ホスト名に置き換えた上で、ディレクティブのコメントを解除します。複数の Portal データベースがある場合は、それらのデータベースのドメイン名のみを入力します。

```
#<Location "/sso/eappslist">
#  Order deny,allow
#  Deny from all
#  Allow from <your_domain_name>
#</Location>
```

sso_apache.conf を編集してから、Distributed Cluster Management のリポジトリを更新します。

```
ORACLE_HOME/dcm/bin/dcmctl updateConfig -v -d
```

ロード・バランシング・ルータがある場合の URL の保護

Single Sign-On Server と OracleAS Portal のフロントエンドにロード・バランシング・ルータが配置されている構成では、ホストへのアクセスを制限するルールをロード・バランシング・ルータで直接設定する必要があります。この構成でホストへのアクセスを許可または拒否する場合は、そのルールを *ORACLE_HOME*/Apache/Apache/conf/sso_apache.conf ファイルに追加しないでください。

BigIP の場合の例を次に示します。

```
if (client_addr != <infrastructure db IP> netmask 255.255.255.0 and
    (http_uri starts_with
    "/sso/eappslist")) {
discard
}
else {
    use pool SSO
}
```

> **注意：** この例は、説明の目的でのみ示しています。実際に適用するアクセス・ルールは、使用しているロード・バランシング・ルータに合せるようにしてください。

Oracle HTTP Server とシングル・サインオン中間層の再起動

次の 2 つのコマンドを発行します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

SSL の構成に関する注意事項

SSL を構成するとき、次の 2 つのケースを考慮する必要があります。

- OracleAS Single Sign-On Server で SSL を構成した場合
- SSL 用にパートナ・アプリケーションを構成した場合

SSL 用に OracleAS Single Sign-On Server を構成した場合、OracleAS Single Sign-On Server のフロントエンドである Oracle HTTP Server で非 SSL ポートも保持する必要があります。このポートは、SSL が構成されているかどうかにかかわらず、OracleAS Single Sign-On の操作に必要です。

ただし、この場合、デフォルトではユーザーは Oracle HTTP Server にある OracleAS Single Sign-On で保護されたコンテンツに、その非 SSL ポートを使用してアクセスすることはできません。たとえば、Oracle HTTP Server 1 上の http://myhost.mydomain.com を保護するように OracleAS Single Sign-On を構成した場合、ブラウザでこの URL を入力するとエラーが返されます。かわりに、https を使用する必要があります。

SSL 用にパートナ・アプリケーションを構成した場合（OracleAS Single Sign-On の構成も含む）、ユーザーが SSL ポートと非 SSL ポートの両方を使用してコンテンツにアクセスできるようにするには、2 つのパートナ・アプリケーション・インスタンスを構成する必要があります。1 つのインスタンスは SSL ポートを使用するようにし、もう 1 つのインスタンスは非 SSL ポートを使用するようにします。詳細は、第 4 章「パートナ・アプリケーションの設定と管理」を参照してください。

> **注意：** OracleAS Single Sign-On Server のフロントエンドとなる Oracle HTTP Server で非 SSL ポートをブロックしないようにしてください。

パートナ・アプリケーションの登録

パートナ・アプリケーションで SSL を有効にしたら、mod_osso をシングル・サインオン中間層とアプリケーション中間層に登録します。この手順では、有効なシングル・サインオン URL を使用できるように mod_osso を構成します。詳細は、第 4 章の「仮想ホストでの mod_osso の構成（SSL および非 SSL）」の項を参照してください。Single Sign-On SDK と統合されたアプリケーションである OracleAS Portal を登録するには、`ptlconfig` ツールを使用します。`ptlconfig` の使用方法は、『Oracle Application Server Portal 構成ガイド』の付録 B を参照してください。

注意： パートナ・アプリケーションを登録すると、デフォルトではユーザーは保護されたコンテンツに非 SSL ポートを使用してアクセスすることはできません。詳細は、7-6 ページの「SSL の構成に関する注意事項」を参照してください。

---

mod_osso Cookie のセキュアな通信

OssoSecureCookies ディレクティブを追加すると、mod_osso で作成したすべての Cookie の Secure フラグをオンに設定することができます。オンに設定すると、ブラウザは HTTPS で保護された接続でのみこれらの Cookie を送信します。

ORACLE_HOME/Apache/Apache/conf/mod_osso.conf にある mod_osso 構成ファイルのこのディレクティブの例を次に示します。

```
<IfModule mod_osso.c>
OssoIpCheck off
OssoIdleTimeout off
OssoSecureCookies on
OssoConfigFile osso/osso.conf
<Location /j2ee/webapp>
require valid-user
AuthType Basic
</Location>
</IfModule>
```

8

デジタル証明書を使用したサインオン

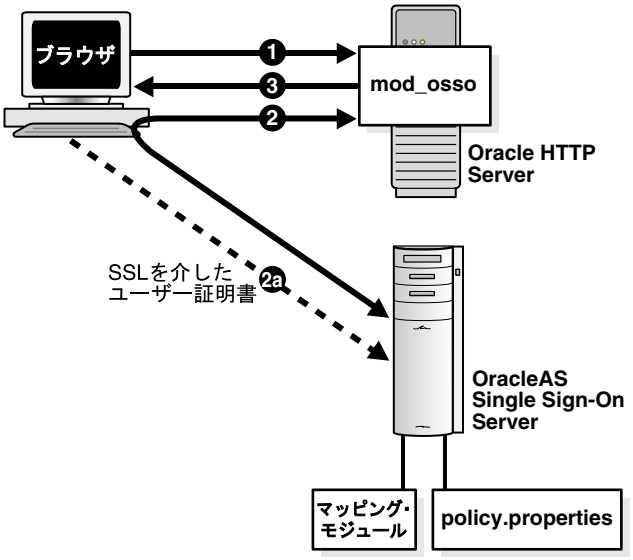
X.509 クライアント証明書を使用したシングル・サインオンでは、セキュリティ・レベルが簡易認証よりも強化されます。パートナ・アプリケーションでは、Single Sign-On Server が PKI に対応しているとき、デフォルトで PKI を使用できるという利点があります。

この章の項目は次のとおりです。

証明書を使用した認証の仕組み

図 8-1 に、証明書を使用したシングル・サインオンでの認証の流れを示します。

図 8-1 証明書を使用したシングル・サインオン



1. ユーザーは、パートナ・アプリケーションにアクセスします。
2. パートナ・アプリケーションは、認証のためにユーザーを Single Sign-On Server にリダイレクトします。このリダイレクションにおいて、ユーザーの証明書が Single Sign-On Server のログイン URL に送信されます（2a）。証明書が検証されると、Single Sign-On Server はリクエストされたアプリケーションをユーザーに返します。
3. アプリケーションはコンテンツを配信します。

注意： ブラウザが証明書ストアのパスワードを要求するように構成されている場合、その構成方法によっては、パスワードを一度入力するだけでよい場合があります。ログアウトしてからパートナ・アプリケーションにアクセスしようとすると、ユーザーの証明書がブラウザから Single Sign-On Server に自動的に転送されます。つまり、実際にはログアウトしていないことになります。正式にログアウトするには、ブラウザを閉じる必要があります。

システム要件

証明書を使用したシングル・サインオンを行うには、次の条件を満たしている必要があります。

- Single Sign-On Server と Oracle Internet Directory がインストールされていること
- Oracle HTTP Server に、有効なサーバー証明書がインストールされていること
- クライアント証明書の識別名が選択され、次の 2 つの条件のいずれかを満たしていること
  - ユーザー証明書の識別名が Oracle Internet Directory のユーザーの識別名と同じである。
  - ユーザー証明書の識別名にユーザー・ニックネームと、ユーザーが属するレルムの名前（オプション）が含まれている。
- クライアント証明書発行者の証明書が、信頼できる証明書として Single Sign-On Server にインストールされていること
- サーバー証明書発行者の証明書が、信頼できる証明書としてユーザーのブラウザにインストールされていること

証明書用のシングル・サインオン・システムの構成

証明書を使用したシングル・サインオンは OracleAS のデフォルト・オプションではないので、インストール後に構成する必要があります。証明書による認証を構成するには、事前にシングル・サインオン・システムで SSL を使用可能にする必要があります。第 7 章の作業を実行してからこの項に戻り、証明書用に次のコンポーネントを構成します。

- Oracle HTTP Server
- Single Sign-On Server
- Oracle Internet Directory

Oracle HTTP Server

Oracle HTTP Server を証明書用に構成するには、`ssl.conf` ファイルにパラメータを追加します。さらに、任意でサーバーおよびユーザー証明書を発行する認証局を選択します。

SSL パラメータの構成

必要な SSL パラメータを構成する手順は次のとおりです。

1. `ssl.conf` へ移動します。このファイルは *`ORACLE_HOME`*`/Apache/Apache/conf` にあります。

2. `ssl.conf` の `SSL Virtual Host Context` セクションで、表 8-1 に示すパラメータを追加または編集します。同時に、`SSLEngine` パラメータが `on` に構成されているのを確認します。これは、SSL 用の Oracle HTTP Server の構成で設定されているはずです。

表 8-1 証明書を使用したシングル・サインオンの構成時に使用する HTTP パラメータ

パラメータ	説明
`SSLWallet`	サーバーの Wallet の場所（パス）。デフォルトの場所は、*`ORACLE_HOME`*`/Apache/Apache/conf/ssl.wlt/default` です。 **注意：** Oracle ホームの実際の格納場所に変数を置き換えてください。 OracleAS Certificate Authority が OracleAS Single Sign-On と同じ Oracle ホームにインストールされ、この認証局を使用して証明書を発行する場合、Wallet の場所は *`ORACLE_HOME`*`/oca/wallet/ssl` になります。詳細は、「認証局の選択」を参照してください。
`SSLWalletPassword`	サーバーの Wallet のパスワード。
`SSLVerifyClient`	クライアント証明書の検証タイプ。次の 3 つのタイプがあります。 ■ `none`: 証明書のない SSL ■ `optional`: サーバー証明書、および任意でクライアント証明書 ■ `require`: サーバー証明書およびクライアント証明書 `optional` または `require` を選択する必要があります。

認証局の選択

OracleAS Certificate Authority がインストール済で、この認証局を使用して証明書を発行する場合は、目的の Oracle 認証局 Wallet を指定するように `ssl.conf` を編集します。Wallet は、表 8-1 に示す Oracle 認証局 Wallet を使用するか、Single Sign-On Server 専用の Wallet を Oracle 認証局で発行します。前者の場合は、Oracle 認証局の Wallet ディレクトリにある Wallet をデフォルトの Wallet ディレクトリにコピーします。後者の場合は、『Oracle Application Server Certificate Authority 管理者ガイド』の第 7 章の操作手順を参照してください。該当する項は「「サーバー / 下位 CA 証明書」タブ」です。この項は、エンド・ユーザー用のタブおよび処理に関する項に含まれている項です。Wallet を取得したら、その Wallet の場所を指すように `ssl.conf` を編集します。

第三者の認証局も使用することができます。この場合も、表 8-1 の説明のとおり `ssl.conf` を編集して Wallet の場所を指定する必要があります。

OracleAS Certificate Authority と OracleAS Single Sign-On を併用すると、証明書プロビジョニング・プロセスが単純になります。Oracle 認証局は、Oracle 認証局の UI の URL をシングル・サイン・オン・ユーザーにブロードキャストするように構成できます。ユーザーは、このリンクを使用してシングル・サインオンの証明書をリクエストできます。この証明書は、Oracle Internet Directory のユーザー・エントリに自動的にリンクされます。

Single Sign-On Server

証明書を受け入れるように Single Sign-On Server を構成するには、次の作業が必要です。

- デフォルトの認証プラグインによる policy.properties の構成
- 認証プラグインの構成ファイルの変更（オプション）
- ユーザー名マッピング・モジュールのカスタマイズ（オプション）
- シングル・サインオン中間層の再起動

少なくとも最初と最後の作業は行う必要があります。ユーザー名マッピング・モジュールをカスタマイズする場合は、間の 2 つの作業も行います。ユーザー名マッピングのデフォルト・モジュールでは、クライアント証明書の識別名（DN）が Oracle Internet Directory の Single Sign-On ユーザーと照合されます。デフォルトの実装では、ディレクトリ内にあるユーザーの識別名と証明書の識別名が同じであるものと想定しています。Oracle Internet Directory のユーザー名に証明書の識別名のフィールドをマップするモジュールも使用できます。識別名マッピング・モジュールのかわりにこのモジュールを使用する場合は、3 番目の作業の指示に従って `CertificateMappingModule` パラメータを変更します。

デフォルトの認証プラグインによる policy.properties の構成

適切な証明書サインオンの認証レベルで `policy.properties` ファイルの `DefaultAuthLevel` セクションを更新します。このファイルは、*`ORACLE_HOME`*`/sso/conf` にあります。デフォルトの認証レベルを次の値に設定します。

```
DefaultAuthLevel = MediumHighSecurity
```

次に、`Authentication plugins` セクションで、この認証レベルをデフォルトの認証プラグインに組み合せます。

```
MediumHighSecurity_AuthPlugin = oracle.security.sso.server.auth.SSOX509CertAuth
```

`policy.properties` の内容は、付録 C「policy.properties」に記載されています。

認証プラグインの構成ファイルの変更（オプション）

`X509CertAuth.properties` ファイルには次のパラメータがあります。このファイルは、`policy.properties` と同じディレクトリにあります。

注意： DN ベースのマッピング・モジュールを使用する場合は、この手順を省略してください。

CertificateMappingModule このパラメータは、ユーザー名マッピングを実行するクラス・ファイル名に構成します。このパラメータには、次の 2 つのデフォルト値のいずれかを指定します。

```
oracle.security.sso.server.auth.SSOCertMapperDn
```

または

```
oracle.security.sso.server.auth.SSOCertMapperNickname
```

最初のモジュールでは、ディレクトリ内にあるユーザーの識別名と証明書の識別名が同じであることを前提としています。このモジュールは、そのまま使用できるデフォルト設定です。2 番目のモジュールでは、証明書のユーザー DN の最初の属性が cn であることを前提としています。また、この属性が、Oracle Internet Directory のデフォルト・レルムにあるユーザー・ニックネームと同じであることも前提としています。たとえば、証明書のユーザー DN が `cn=john,cn=users,dc=acme,dc=com` の場合には、2 番目のモジュールを使用できます。逆に、DN が `e=john.smith@acme.com,cn=john,cn=users,dc=acme,dc=com` の場合、2 番目のモジュールは使用できません。ただし、この DN を使用するマッピング・モジュールを記述することができます。詳細は、「ユーザー名マッピング・モジュールのカスタマイズ（オプション）」を参照してください。独自のモジュールを記述する場合は、実装で使用するクラス・ファイル名を `CertificateMappingModule` に設定します。

CheckUserCertificate このパラメータは、ユーザー証明書を Oracle Internet Directory で検証するかどうかを指定します。デフォルト値は TRUE です。Oracle HTTP Server の SSL 保護で十分な場合は、このパラメータを FALSE に設定します。

CertificateAuthFailureUrl 証明書による認証に失敗すると、この URL にリダイレクトされ、エラー・メッセージが表示されます。

CertificateAuthFallback 有効な証明書なしにログインしようとしているユーザーに、パスワード認証を使用させる場合は、このパラメータを TRUE に設定します。このフォールバックは、デフォルトでは発生しません。手動で有効にする必要があります。このパラメータを FALSE に設定したり、空の状態にした場合、ユーザーには有効な証明書を要求するメッセージが表示されます。CertificateAuthFallback をファイルに追加する必要がある場合があります。その場合は、次のようにファイルの末尾に追加します。

```
#Allow authentication fallback
CertificateAuthFallback=true
```

> **注意：** CertificateAuthFallback を TRUE に設定した場合、マルチレベル認証は使用できません。

ユーザー名マッピング・モジュールのカスタマイズ（オプション）

ユーザー名マッピング・モジュールをカスタマイズするには、oracle.security.sso.ias904.toolkit.IPASUserMappingInterface に基づくマッピング・モジュールを実装します。このリリースに添付されているサンプル・マッピング・モジュールを参照してください。このサンプル・マッピング・モジュールには、SSOCertMapperDN.java と SSOCertMapperNickname.java があります。

> **注意：** 独自のマッピング・モジュールを記述しない場合は、この手順を省略してください。

サンプル・モジュールは次のクラスで構成されています。

- マッピング・モジュール・インタフェース

  このインタフェースには、次のメソッドが組み込まれています。

  ```
  public IPASUserInfo getUserInfo(
         javax.servlet.http.HttpServletRequest request)
         throws IPASException;
  ```

- ユーザー情報クラス

  このクラスには、ユーザー・ニックネームやユーザー識別情報などのユーザー情報が格納されています。パッケージ名は oracle.security.sso.ias904.toolkit.IPASUserInfo です。コンストラクタは次のとおりです。

  ```
  Public IPASUserInfo(
         String userNickName
         String realmNickname)

  Public IPASUserInfo(
         String userNickName,
         String userDN,
         String userGUID,
         String realmNickname,
         String realmDN,
         String realmGUID)
  ```

- 例外クラス

  ユーザー名マッピングで問題が発生すると、この例外が生成されます。クラス名は oracle.security.sso.ias904.toolkit.IPASException です。スーパー・クラスは java.lang.Exception です。コンストラクタは次のとおりです。

  ```
  public IPASException()
  public IPASException(String Message)
  ```

1. サンプル・モジュールを格納したファイル ipassample.jar を抽出します。

   ORACLE_HOME/jdk/bin/jar -xvf *ORACLE_HOME*/sso/lib/ipassample.jar

2. 次のインタフェースを実装する Java クラスを作成します。

   oracle.security.sso.ias904.toolkit.IPASUserMappingInterface

3. 次のカスタム実装をコンパイルします。

   ORACLE_HOME/jdk/bin/javac -classpath *ORACLE_HOME*/sso/lib/ipastoolkit.jar:*ORACLE_HOME*/lib/servlet.jar -d *class_directory java_file_name*

4. クラス・ファイルを JAR ファイル化して *ORACLE_HOME*/sso/plugin に配置します。

   ORACLE_HOME/jdk/bin/jar -cvf *ORACLE_HOME*/sso/plugin/CertMapImpl.jar *class_directory*

   この手順では、プラグイン・ディレクトリに個別のクラス・ファイルがないことを前提としています。ディレクトリに個別のファイルが存在する場合は、ファイルが重複する可能性があります。

5. x509CertAuth.properties を実装内容で更新します。「認証プラグインの構成ファイルの変更（オプション）」を参照してください。

シングル・サインオン中間層の再起動

サーバーを構成したら、中間層を再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

Oracle Internet Directory

証明書による認証を正しく行うには、ユーザー証明書が Oracle Internet Directory にあることが必要です。証明書が OracleAS Certificate Authority によって発行されている場合、その証明書は Oracle Internet Directory で自動的に公開されます。証明書が社内の認証局によって発行されている場合も同様です。証明書の発行者が第三者の認証局である場合は、セルフサービス・アプリケーションでこの機能を実現できます。また、ディレクトリ管理者は、コマンドライン・ツールの ldapmodify を使用して証明書を LDIF ファイルとしてディレクトリに追加できます。

注意：

- この項に記載された手順は、X509CertAuth.properties ファイルの CheckUserCertificate の値が true に設定されていることを前提としています。「認証プラグインの構成ファイルの変更（オプション）」を参照してください。
- usercertificate は証明書を格納しているバイナリ属性で、Oracle Internet Directory で検索できます。証明書検索の詳細は、『Oracle Internet Directory 管理者ガイド』の付録を参照してください。

---

ldapmodify を使用して証明書を公開する場合は、ツールを実行する前に、環境に合せて適切なグローバリゼーション・サポート変数を設定します。次に例を示します。

- UNIX:

```
setenv NLS_LANG AMERICAN_AMERICA.UTF8
```

- Windows:

```
set NLS_LANG=AMERICAN_AMERICA.UTF8
```

UNIX では、csh または tcsh 以外のシェルを使用している場合、別の手順でこの変数を設定する必要があります。

ldapmodify の構文は次のとおりです。

```
ORACLE_HOME/bin/ldapmodify
-h directory_host
-p directory_ssl_port
-D "directory_administrator"
-w administrator_password
-f file_name.ldif
```

次に示すサンプルの LDIF ファイルでは、ユーザー jsmith の証明書はディレクトリでのエントリの属性に従って示されています。属性のタイプは usercertificate です。属性値は長精度文字列型で、属性のタイプの後に続きます。

```
dn: cn=jsmith,cn=users,dc=realm1,dc=oracle,dc=com
changetype: modify
replace: usercertificate
usercertificate::MIIC3TCCAkYCAgP3MA0GCSqGSIb3DQEBBAUAMIG8MQswCQ
 NYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEXMBUGA1UEBxMOUmVkd29vZCBTaG9yZXMxGzAZBg
 VBAoTEk9yYWNsZSBDb3Jwb3JhdGlvbjEfMB0GA1UECxMWV2ViIFNpbmdsZSBTaWduLU9uLCBTVDEeMBwA
 1UEAxMVQ2VydGlmaWNhYoEHmF4gomtc4mxSKh/zAgMBAAEwDQYJKoZIhvcNAQEEBQADgYEAKwXoCLDRqm
 KY9LQtIjLnCaIJKUZmS1Qj+bhu/IHeZLGHg4TJg3O2XVA5u/VxwjLeGBqLXy2z7o3RujNKx2CVx6p/0Hk
 jnw4w6KVau2hcBgC9m4kzUGhHJ9b65v/zx7dIUKyJr4RF+lJhJg4/oYXxLrYHp5NAkHP4htT0gqCXiI=
```

属性値は ASCII 値ではないので、ここで示すように、証明書を BASE64 形式でエンコードする必要があります。他の属性とは異なり、BASE64 属性では区切り記号に二重コロン（::）を使用します。また、タブを使用すると、BASE64 属性を折り返すことができます。

証明書失効リストのメンテナンス

無効な証明書や期限切れの証明書を使用したユーザーがログインできないようにするため、管理者は Oracle HTTP Server の証明書失効リスト（CRL）を最新の状態に保つ必要があります。証明書を発行した認証局は、このリストを提供する必要があります。リストのメンテナンスには、ca-bundle.crl ファイルを使用できます。CRL ファイルのパスには *ORACLE_HOME*/Apache/Apache/conf を指定する必要があります。

認証にデジタル証明書を使用する OracleAS ユーザーについては、ディレクトリ・エントリの userCertificate 属性を更新できないようにする必要があります。これは、証明書の失効から CRL の更新までの時間が長くなるおそれがあるからです。無効なユーザーが CRL チェックを通過すると、ログインを検証するのは、userCertificate の設定のみとなります。ただし、Oracle Internet Directory では、デフォルトで userCertificate へのユーザー・アクセスが拒否されます。この属性の変更には、Single Sign-On 管理者、OracleAS Certificate Authority、第三者の認証局など、信頼できるエンティティのみを使用してください。

CRL の実装とメンテナンスの詳細は、ssl.conf ファイルの SSL Virtual Host Context セクションのコメントを参照してください。

9

高度な配置オプション

この章では、OracleAS Single Sign-On をデフォルト以外で使用する方法について説明します。ここでは、本番環境で起りえる使用例を示します。一部の使用例には、他の OracleAS コンポーネントと相互作用するコンポーネントの配置と構成が含まれます。

この章の項目は次のとおりです。

配置例

この項では、可用性を高めるための Single Sign-On Server の様々な配置例を示します。この項の項目は次のとおりです。

- 1 つのシングル・サインオン中間層、1 つの Oracle Internet Directory
- 複数のシングル・サインオン中間層、1 つの Oracle Internet Directory
- 地理的に分散している複数のシングル・サインオン・インスタンス
- その他の高可用性の配置

> **注意：** 以降の使用例で示す IP アドレスとホスト名は、単に例として示したものです。これらのアドレスと名前は、実際の実装では機能しない場合があります。実際のインストールでは、該当する値に置き換えてください。

1 つのシングル・サインオン中間層、1 つの Oracle Internet Directory

OracleAS Single Sign-On を配置する最も簡単で迅速な方法は、OracleAS Infrastructure のコンポーネントを同一のコンピュータにインストールすることです。この作業を行うには、インストール・タイプに OracleAS Infrastructure を選択し、インストール・オプションに Identity Management and OracleAS Metadata Repository を選択します。このインストール・タイプのコンポーネント・リストが表示されたら、デフォルトで選択されているコンポーネントを受け入れます。

または、「OracleAS Infrastructure」、「Identity Management」、「Single Sign-On」の順に選択し、シングル・サインオン中間層を別のコンピュータにインストールすることもできます。これは、最も簡単な分散構成です。

図 9-1 は、最初のインストールのタイプを示しています。図 9-2 は、2 番目のタイプを示しています。最初のタイプは、一般にテスト環境、ステージング環境または開発環境で使用されます。2 番目のタイプは、Single Sign-On と Oracle Internet Directory の各コンピュータ間にファイアウォールを設置する場合に適しています。これらのサーバーを別々のコンピュータに配置すると、パフォーマンスが向上するという付加的な利点があります。図 9-2 では、DMZ 内に Single Sign-On Server が配置されています。DMZ ではインターネット・トラフィックのフィルタ処理が行われます。この構成のディレクトリおよびデータベースには、イントラネット・ユーザーのみがアクセスできます。

図 9-1 デフォルトの Single Sign-On インストール : 1 台のコンピュータ

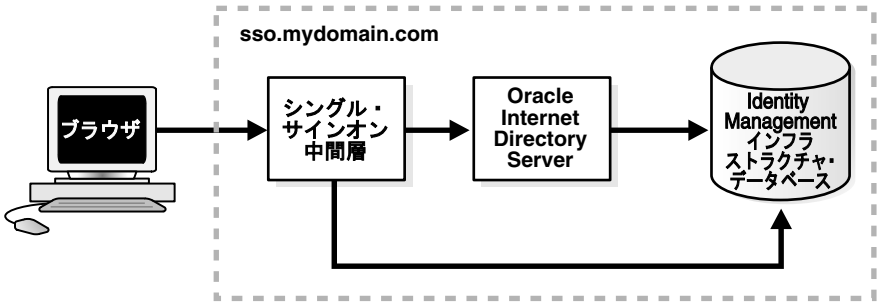
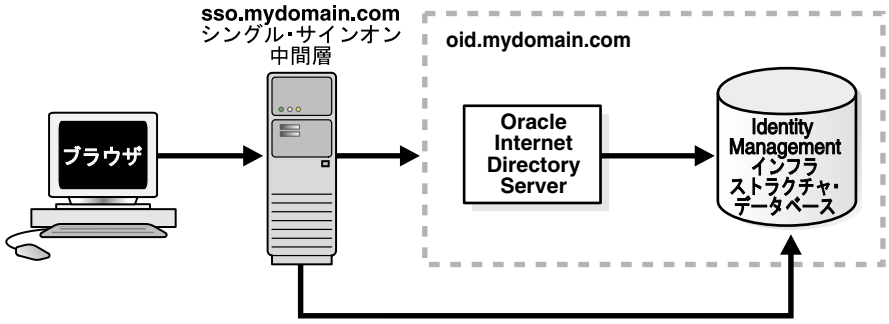


図 9-2 Single Sign-On インストール : 2 台のコンピュータ



複数のシングル・サインオン中間層、1 つの Oracle Internet Directory

最も単純な高可用性の使用例では、中間層のシングル・サインオン・インスタンス自体にフェイルオーバーの機能を組み込みます。複数の中間層を追加するとスループットが増加するため、Single Sign-On Server の可用性が向上します。

この構成では、複数の Oracle HTTP Server の前段に HTTP ロード・バランサを 1 つ配置します。バックエンドには、ディレクトリ・サーバーと ID 管理インフラストラクチャ・データベースを 1 つずつ配置します。ロード・バランサの目的は、複数の Single Sign-On パートナ・アプリケーションに単一のアドレスを公開するとともに、複数のシングル・サインオン中間層のファームを提供することです。ファーム内の中間層で実際にアプリケーションのリクエストが処理されます。HTTP ロード・バランサは、Oracle HTTP Server インスタンスの 1 つで発生した障害を検出し、別のインスタンスにリクエストをフェイルオーバーできます。

この項の項目は次のとおりです。

- クラスタ化の選択
- 使用例
- 構成手順

クラスタ化の選択

複数のシングル・サインオン中間層は、クラスタとして、または手動で配置できます。最初の方法は、インストールが容易である点で推奨されます。OracleAS インストーラは、シングル・サインオン・ノードを 1 つの Distributed Cluster Management（DCM）データベースの周囲に自動的にクラスタ化します。DCM は、ノードのいずれかでクラスタ構成情報に変更が発生した場合に、クラスタ構成情報をクラスタ内のすべてのノード間でレプリケートするコンポーネントです。この構成は、OracleAS Single Sign-On などの中間層コンポーネントがノード間でまったく同一にクラスタ化および構成されることから、OracleAS クラスタ（ID 管理）と呼ばれます。

ただし、DCM データベースに障害が発生すると、すべてのシングル・サインオン・ノードに障害が発生します。このような依存性を避ける場合は、各中間層とそれぞれの DCM データベースを手動で構成します。

手動による配置の構成手順は、次の項で説明します。

> **関連項目：** クラスタのインストール方法は、Oracle Application Server のインストレーション・ガイドの OracleAS クラスタ（ID 管理）に関する章を参照してください。具体的には、次の項を参照してください。
>
> - OracleAS クラスタ（ID 管理）の構成のインストールに関する項
> - OracleAS 分散クラスタ（ID 管理）の構成のインストールに関する項

使用例

この使用例では、次の架空の構成を想定しています。

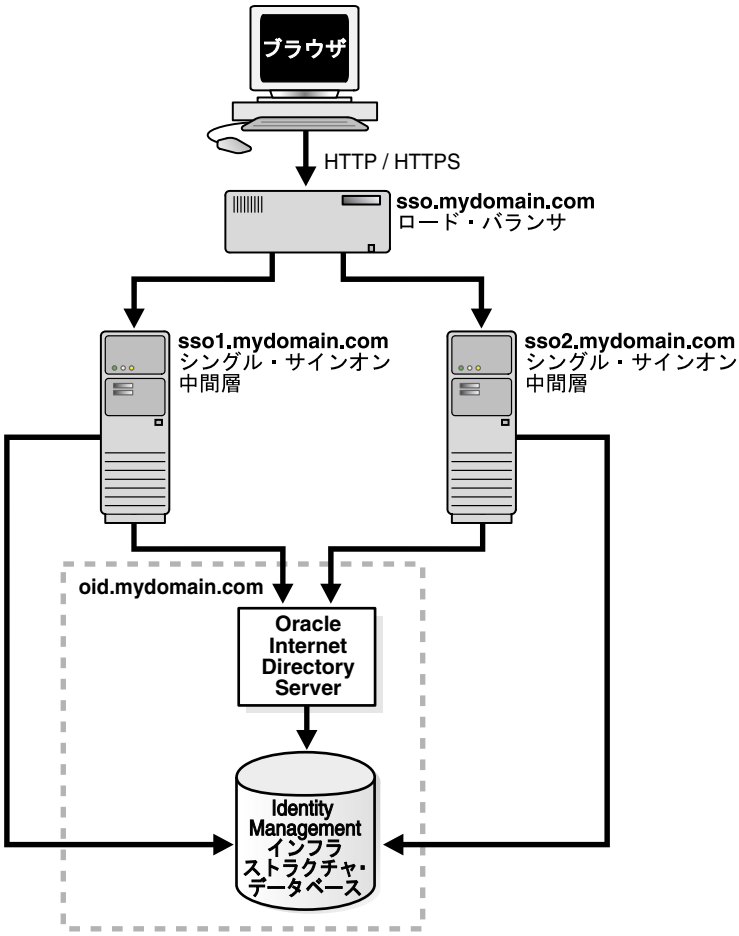
- ディレクトリ・サーバーと ID 管理インフラストラクチャ・データベースは、oid.mydomain.com に配置されています。
- 2 つのシングル・サインオン中間層があります。1 つはホスト sso1.mydomain.com（IP アドレス 138.1.34.172）にインストールされています。もう 1 つは sso2.mydomain.com（IP アドレス 138.1.34.173）にインストールされています。サーバーは、両方とも非 SSL ポート 7777 をリスニングしています。またこれらは、oid.mydomain.com に配置されているディレクトリと ID 管理インフラストラクチャ・データベースを使用するように構成されています。
- パートナ・アプリケーションに公開されている Single Sign-On Server の有効な URL は、sso.mydomain.com（IP アドレス 138.1.34.234）です。HTTP ロード・バランサは、sso.mydomain.com（ポート 80）をリスニングするように構成されています。HTTP ロード・バランサは、ユーザーのリクエストによる負荷を sso1.mydomain.com と sso2.mydomain.com の間で分散します。

注意：

- この使用例では、ロード・バランサが非 SSL ポート番号のポート 80 をリスニングします。
- ロード・バランサが SSL を使用してブラウザと対話するように構成されている場合は、別のポート番号を選択する必要があります。デフォルトの SSL ポート番号は 4443 です。
- この使用例とその次の使用例では、2 つのシングル・サインオン中間層が使用されています。実際には、任意の数の中間層を使用できます。

9-5 ページの図 9-3 は、Oracle Internet Directory のシングル・インスタンスを使用するように構成した 2 つのシングル・サインオン中間層を示しています。

図 9-3 2 つのシングル・サインオン中間層、1 つの Oracle Internet Directory



構成手順

図 9-3 に示すシングル・サインオン・システムを設定するには、次の作業を行います。

- ID 管理インフラストラクチャ・データベース、ディレクトリ・サーバーおよび Single Sign-On Server のインストール
- シングル・サインオン中間層での Oracle HTTP Server の構成
- HTTP ロード・バランサの構成
- ID 管理インフラストラクチャ・データベースの構成
- シングル・サインオン中間層での mod_osso の登録

ID 管理インフラストラクチャ・データベース、ディレクトリ・サーバーおよび Single Sign-On Server のインストール

1. パートナ・アプリケーションに公開する Single Sign-On Server 名を選択します。この名前はロード・バランサのアドレスにもなります。この配置例の場合、このアドレスは sso.mydomain.com です。
2. Identity Management and OracleAS Metadata Repository のオプションを選択して、OracleAS Infrastructure を `oid.mydomain.com` にインストールします。このインストール・タイプのコンポーネント・リストが表示されたら、Oracle Internet Directory のみを選択します。
3. 中間層 `sso1.mydomain.com` および `sso2.mydomain.com` に OracleAS Infrastructure をインストールし、オプションの「Identity Management」を選択します。このインストール・タイプのコンポーネント・リストが表示されたら、「OracleAS Single Sign-On」のみを選択します。Oracle Universal Installer で、これらのシングル・サインオン・インスタンスに関連付けられたディレクトリ・サーバーに名前を付けるように求められたら、`oid.mydomain.com` と入力します。

注意： デフォルトでは、OracleAS インストーラはある範囲の数値からポート番号を割り当てます。インストーラでコンポーネントに異なるポート番号を割り当てる場合は、Oracle Application Server のインストレーション・ガイドの第 4 章の「静的ポート番号」を参照してください。

シングル・サインオン中間層での Oracle HTTP Server の構成

ユーザーと Oracle HTTP Server の間にロード・バランサを配置すると、Single Sign-On Server の有効な URL が変更されます。両方のシングル・サインオン中間層の Oracle HTTP 構成ファイル `httpd.conf` を修正し、この変更を反映する必要があります。このファイルは `$ORACLE_HOME/Apache/Apache/conf` にあります。

1. `sso1.mydomain.com` および `sso2mydomain.com` で `httpd.conf` の次の行を編集します。

```
KeepAlive off
ServerName sso.mydomain.com
Port 80
```

注意： `httpd.conf` に複数のポートが記述されている場合、変更されるポートは必ず最後のものになります。

この手順では、シングル・サインオン中間層で Oracle HTTP Server を構成し、有効な URL をリスニングします（この使用例では `sso.mydomain.com`）。

2. ブラウザとロード・バランサの間に SSL を構成し、SSL 接続がロード・バランサで終了する場合は、`sso1.mydomain.com` と `sso2.mydomain.com` の両方に mod_certheaders を構成します。このモジュールによって、Oracle HTTP Server では、HTTP で受信するリクエストを SSL リクエストとして処理できるようになります。次の手順を追加します。これらは、httpd.conf の最後に追加できます。順序は重要ではありません。
   a. 両方の中間層の `httpd.conf` に、次の行を入力します。

      ```
      LoadModule certheaders_module libexec/mod_certheaders.so
      ```

   b. OracleAS Web Cache をロード・バランサとして使用する場合は、次の行を入力します。

      ```
      AddCertHeader HTTPS
      ```

      ハードウェア・ロード・バランサを使用する場合は、次の行を入力します。

      ```
      SimulateHttps on
      ```

3. 2 つの中間層のシステム・クロックを同期化します。

4. 次のコマンドを実行して、Distributed Configuration Management スキーマを変更内容で更新します。

```
$ORACLE_HOME/dcm/bin/dcmctl updateConfig -v -d
```

HTTP ロード・バランサの構成

HTTP ロード・バランサには、ハードウェア（BigIP、Alteon、Local Director など）またはソフトウェア（OracleAS Web Cache など）のどちらも使用できます。

- ハードウェア・ロード・バランサ

  ハードウェアのロード・バランサを使用する場合は、実サーバーの 1 つのプールをアドレス `138.1.34.172` および `138.1.34.173` で構成します。1 つの仮想サーバーをアドレス `138.1.34.234` で構成します。この仮想サーバーは、ロード・バランサの外部インタフェースです。構成手順の詳細は、ロード・バランサのベンダーが提供するドキュメントを参照してください。

- ソフトウェア・ロード・バランサ

  接続リクエストのロード・バランスに OracleAS Web Cache を使用する場合は、次のドキュメントを参照してください。

  - 『Oracle Application Server Web Cache 管理者ガイド』の「Oracle Identity Management インフラストラクチャの強化」
  - 『Oracle Application Server Web Cache 管理者ガイド』の「Single Sign-On Server リクエストのルーティング」

  注意： 最高のパフォーマンスを得るには、ハードウェア・ロード・バランサを使用してください。

ID 管理インフラストラクチャ・データベースの構成

シングル・サインオン中間層の 1 つで ssocfg スクリプトを実行します。このスクリプトによって、外部に公開された Single Sign-On Server のアドレスによる認証リクエストを受け入れるように Single Sign-On Server が構成されます。この例では、スクリプトは次のように実行します。

- UNIX:

  ```
  $ORACLE_HOME/sso/bin/ssocfg.sh http sso.mydomain.com 80
  ```

- Windows NT/2000:

  ```
  %ORACLE_HOME%¥sso¥bin¥ssocfg.bat http sso.mydomain.com 80
  ```

これらのコマンドの例では、ロード・バランサのリスナー・プロトコル、ホスト名、ポート番号が引数として指定されています。ロード・バランサのアドレスは、外部に公開されている Single Sign-On Server のアドレスであることを思い出してください。SSL を使用するようにロード・バランサが構成されている場合には、非 SSL ポート `80` を SSL ポート `4443` に置き換え、`http` を `https` に置き換えてください。

シングル・サインオン中間層での mod_osso の登録

両方の中間層コンピュータで、mod_osso をパートナ・アプリケーション sso.mydomain.com として登録します。

mod_osso を sso1.mydomain.com に登録する手順は次のとおりです。

1. 登録スクリプトを実行します。URL は、実際のインストール環境の該当する値に置き換えてください。このスクリプトでは、sso.mydomain.com という名前のパートナ・アプリケーションが作成されます。

   ```
   $ORACLE_HOME/sso/bin/ssoreg.sh
   -oracle_home_path orcl_home_path
   -site_name site_name
   -config_mod_osso TRUE
   -mod_osso_url mod_osso_url
   -u userid
   [-virtualhost]
   [-update_mode CREATE | DELETE | MODIFY]
   [-config_file config_file_path]
   [-admin_id adminid]
   [-admin_info admin_info]
   ```

   コマンド・パラメータの詳細は、第 4 章の「mod_osso の登録」の項を参照してください。

2. sso1.mydomain.com の中間層を再起動します。詳細は、第 2 章の「シングル・サインオン中間層の停止と起動」の項を参照してください。

mod_osso を sso2.mydomain.com に登録する手順は次のとおりです。

1. コンピュータ sso2.mydomain.com で、Single Sign-On 管理者としてシングル・サインオン管理ページにログインします。次の URL にログインしてください。

   ```
   http://sso.mydomain.com/sso
   ```

2. 「パートナ・アプリケーションの管理」ページを使用して、パートナ・アプリケーション sso2.mydomain.com の既存エントリを削除します。

3. コンピュータ sso1.mydomain.com から osso.conf ファイルをコピーします。ファイルを FTP で転送する場合は、バイナリ・モードを使用してください。このファイルを $ORACLE_HOME/Apache/Apache/conf/osso にコピーします。

4. Distributed Configuration Management リポジトリとコピーしたファイルを同期化します。これには、sso2.mydomain.com で次のコマンドを実行します。

   ```
   $ORACLE_HOME/Apache/Apache/bin/ssotransfer
   $ORACLE_HOME/Apache/Apache/conf/osso/osso.conf
   ```

   > **注意：** ssotransfer コマンドは、Distributed Configuration Management リポジトリと仮想ホストに作成された mod_osso 構成ファイルとの同期化には使用しないでください。仮想ホストの mod_osso を登録する方法は、第 4 章の「仮想ホストでの mod_osso の構成（SSL および非 SSL）」の項を参照してください。

5. sso2.mydomain.com の中間層を再起動します。詳細は、第 2 章の「シングル・サインオン中間層の停止と起動」の項を参照してください。

6. Oracle Delegated Administration Services がインストールされている場合は、そのベース URL を Oracle Directory Manager によって次の手順で変更します。

   a. ツールを起動します。

      ```
      $ORACLE_HOME/bin/oidadmin
      ```

   b. cn=orcladmin として Oracle Directory Manager にログインします。

c. 次のように指定して、orcldasurlbase 属性を含むエントリに移動します。

cn=OperationURLs,cn=DAS,cn=Products,cn=OracleContext,Entry Management

d. 属性を次の値に変更します。

http://sso.mydomain.com/

ホスト名の後にスラッシュを挿入してください。

e. 次の URL でパートナ・アプリケーション oiddas をテストします。

http://sso.mydomain.com/oiddas

7. 次の URL で Single Sign-On 管理アプリケーションをテストします。

http://sso.mydomain.com/sso

複数のシングル・サインオン中間層、レプリケートされた Oracle Internet Directory

通信量の多い Local Area Network（LAN）では、複数のシングル・サインオン中間層を Oracle Internet Directory のレプリケート・インスタンスで補強すると有効な場合があります。この配置では、中間層とディレクトリ・サーバーの両方でフェイルオーバーを実行できます。これにより、レプリカ・ノードを削除してメンテナンスを実行する間も他のノードでユーザーを処理できるため、ローリング・アップグレードを行う場合に役立ちます。

マルチマスター・レプリケーションを使用する Oracle Identity Management システムの配置方法は、『Oracle Application Server 高可用性ガイド』のこのトピックに関する章を参照してください。この章には、ID 管理インフラストラクチャの各コンポーネントを構成する方法も示されています。

地理的に分散している複数のシングル・サインオン・インスタンス

事業が地理的に広く分散している企業にとって、サーバーの可用性は非常に重要です。企業が 1 台のサーバーで Wide Area Network を介してリモート・ユーザーを認証している場合は、認証に長時間を要することがあります。ネットワーク・ラウンドトリップを短縮してアプリケーションへのアクセスを高速化するために、企業では、複数の Single Sign-On Server インスタンスを地理的に分散して実装できます。この配置では、アプリケーションの格納場所に関係なく、ユーザーはリモート・ロケーションに移動して、最も近いサーバーで認証を受けることができます。

この使用例では、Single Sign-On データベース表が Local Area Network（LAN）または Wide Area Network を介してレプリケートされます。Single Sign-On Server の有効なアドレスがユーザーの最寄りのシングル・サインオン・インスタンスに解決されるように、各シングル・サインオン中間層サイトに配置されている DNS サーバーを構成する必要があります。

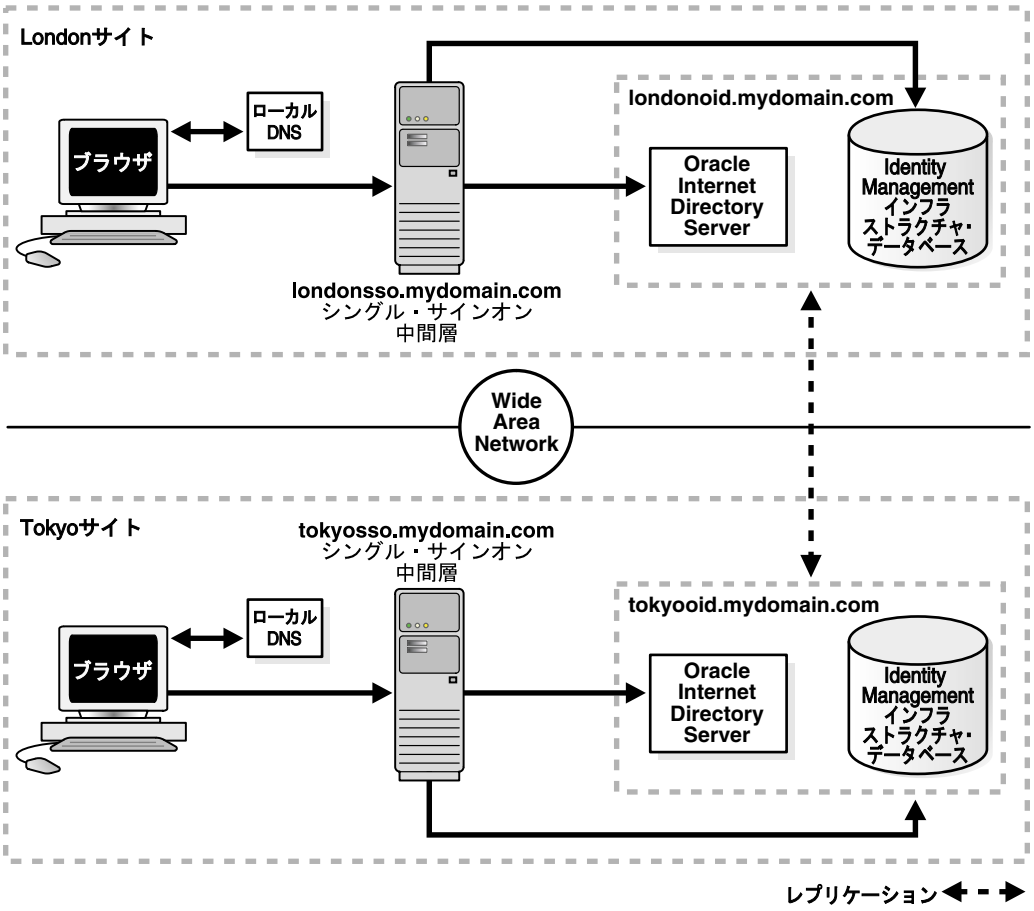
使用例

この使用例では、次の架空の構成を想定しています。

- londonsso.mydomain.com と tokyosso.mydomain.com の 2 つのシングル・サインオン中間層があります。Single Sign-On Server の有効なアドレスは sso.mydomain.com です。
- 2 つのシングル・サインオン中間層（londonoid.mydomain.com、tokyooid.mydomain.com）に 2 つのディレクトリ・サーバーと ID 管理インフラストラクチャ・データベースが関連付けられています。
- レプリケーションにおいては、londonoid.mydomain.com がマスター定義サイト（MDS）になります。このサイトではレプリケーション・スクリプトが実行され、データが最初にレプリケートされます。tokyooid.mydomain.com はリモート・マスター・サイト（RMS）になります。このサイトは、データのレプリケート先のサイトです。
- シングル・サインオン中間層と ID 管理インフラストラクチャ・データベースは、異なるコンピュータに配置されています。

9-10 ページの図 9-4 は、この地理的に分散しているシステムの配置後の構成を示しています。

図 9-4 地理的に分散している高可用性シングル・サインオン・システム



構成手順

図 9-5 に示す地理的に分散しているシングル・サインオン・システムは、「複数のシングル・サインオン中間層、1 つの Oracle Internet Directory」と「レプリケーション用の ID 管理データベースの構成」で示した手順を組み合せたものです。

1. MDS（`londonoid.mydomain.com`）と RMS（`tokyooid.mydomain`）に Oracle Internet Directory をインストールし、これらのサーバーをレプリケーション・グループとして設定します。詳細は、『Oracle Application Server 高可用性ガイド』のマルチマスター・レプリケーションを使用する Oracle Identity Management の配置方法に関する付録を参照してください。この手順には、インストールとレプリケーションの両方が含まれています。レプリケーションの概念は、『Oracle Internet Directory 管理者ガイド』も参照してください。

2. オプション「Identity Management」を選択して、OracleAS Infrastructure を中間層 `londonsso.mydomain.com` にインストールします。このインストール・タイプのコンポーネント・リストが表示されたら、「Single Sign-On」のみを選択します。Oracle Universal Installer で、このシングル・サインオン・インスタンスに名前を付けるように求められたら、`londonoid.mydomain.com` と入力します。

3. 中間層 `tokyosso.mydomain.com` で手順 2 を繰り返します。ここでは、`tokyooid.mydomain.com` にあるディレクトリ・サーバーを Single Sign-On Server に関連付ける必要があります。

4. Single Sign-On スキーマのパスワードを MDS と RMS のデータベース間で同期化します。この作業を行うには、「レプリケーション用の ID 管理データベースの構成」の手順 2 を実行します。

5. 2 つのシングル・サインオン・インスタンスは別々の場所で実行されていますが、パートナ・アプリケーションに公開されている有効なサーバー URL は 1 つのみです。この URL を使用できるように Single Sign-On Server を構成します。この使用例では、この URL を `sso.mydomain.com` と呼びます。詳細は、「シングル・サインオン中間層での Oracle HTTP Server の構成」を参照してください。

6. シングル・サインオン中間層を指す、DNS エイリアスの `sso.mydomain.com` を追加します。シングル・サインオン認証が必要なときに、ユーザーを最も近い中間層にルーティングするように DNS サーバーを構成します。たとえば、London ユーザーが `http://sso.mydomain.com` にリダイレクトされるとき、DNS サーバーはそのユーザーを `http://londonsso.mydomain.com` にルーティングする必要があります。同様に `http://sso.mydomain.com` にリダイレクトされる Tokyo ユーザーは、`http://tokyosso.mydomain.com` にルーティングする必要があります。

   高機能の DNS サーバー製品の中には、地理的位置に基づいてユーザーを最寄りのサーバーにルーティングできるものもあります。

その他の高可用性の配置

OracleAS は、シングル・サインオンやその他の OracleAS コンポーネントで、コールド・フェイルオーバー・クラスタ、障害時リカバリ、バックアップおよびリカバリをサポートしています。

OracleAS Cold Failover Cluster（インフラストラクチャ）

コールド・フェイルオーバー・クラスタは、ネットワーク・サービスの単一ビューを連携して提供する、疎結合のコンピュータのグループです。1 次ノードで障害が発生した場合は、クラスタ・ソフトウェアによって 1 次ノードの論理 IP アドレスと処理を 2 次ノードに移動できます。インフラストラクチャを実行しているノードはホットと呼ばれます。引き継ぎを待機しているノードはコールドと呼ばれます。このため、コールド・フェイルオーバーという用語が使用されます。

コールド・フェイルオーバー・クラスタの詳細は、『Oracle Application Server 高可用性ガイド』のインフラストラクチャの高可用性に関する章を参照してください。

障害時リカバリ

障害時リカバリの配置は、構成が同一の 2 つのサイト、プライマリ（本番）とセカンダリ（スタンバイ）で構成されます。2 つのサイトは、地理的に離れ、Wide Area Network で接続されていることもあります。障害のためにプライマリ・サイトが使用できない場合は、適切な時間内にセカンダリ・サイトを操作可能にできます。クライアントのリクエストは、本番の役割を担うサイトに常にルーティングされます。フェイルオーバーの発生後、クライアントのリクエストはセカンダリ・サイトにルーティングされ、その後セカンダリ・サイトは本番の役割を引き継ぎます。プライマリ・サイトとセカンダリ・サイトには同一の中間層サーバーが配置されており、これらのサーバーは 2 つのサイト間でも同一です。障害時のリカバリの詳細は、『Oracle Application Server 高可用性ガイド』のこのトピックに関する章を参照してください。

バックアップおよびリカバリ

バックアップおよびリカバリは、データ損失の防止と損失データの復元についての方針と手順を述べるときに使用する用語です。バックアップおよびリカバリの詳細は、『Oracle Application Server 管理者ガイド』のこのトピックに関する章を参照してください。

ID 管理データベースのレプリケート

この項では、ID 管理データベースを複数のインスタンス間でレプリケートする方法について説明します。OracleAS Single Sign-On と Oracle Internet Directory は、データベース表をレプリケートするスクリプトと手順を共有しています。次の事項を十分に理解してからこの項に進んでください。

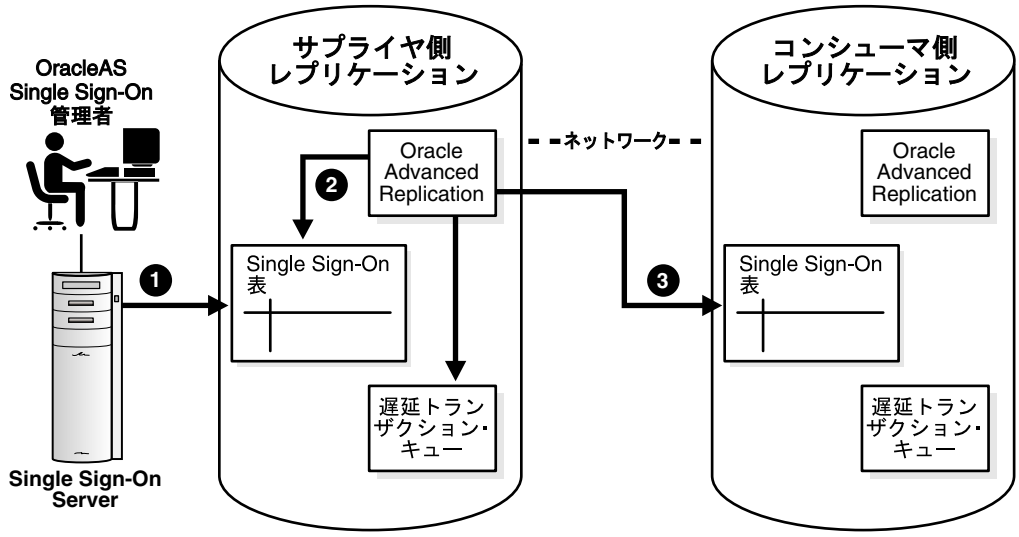
- 『Oracle Internet Directory 管理者ガイド』の「ディレクトリ・レプリケーションの概要」
- 『Oracle Internet Directory 管理者ガイド』の「Oracle ディレクトリ・レプリケーションの管理」
- 『Oracle Internet Directory 管理者ガイド』の「レプリケーション管理コマンドライン・ツールの構文」

この項の項目は次のとおりです。

レプリケーションのメカニズム

ID 管理インフラストラクチャでは、2 つのデータベース間の表のレプリケーションに Advanced Replication を使用します。この機能は、データの変更を複数のデータベースに非同期に伝播します。つまり、サプライヤは変更をシングル・サインオン表に書き込み、バッチされた変更をコンシューマに定期的に送信します。コンシューマは、このデータをレプリケートするサーバーです。地理的に分散している複数のシステムで、すべてのサーバーがデータを伝播または受信できます。この配置をマルチマスター・レプリケーションと呼びます。図 9-5 にそのプロセスを示します。

図 9-5 マルチマスター・レプリケーションのアーキテクチャ



1. Single Sign-On 管理者は Single Sign-On 管理アプリケーションを使用して Single Sign-On パートナ・アプリケーションまたは構成データを変更します。このプロセスでは、ID 管理インフラストラクチャ・データベースの対応する表エントリが変更されます。
2. Advanced Replication によって、遅延トランザクション・キューに変更がコピーされます。
3. Advanced Replication は、スケジュールされた間隔で、遅延トランザクション・キーのトランザクションをコンシューマ側のシングル・サインオン表にプッシュします。

レプリケーション用の ID 管理データベースの構成

『Oracle Internet Directory 管理者ガイド』でマルチマスター・レプリケーションの概念を理解してからこの項に進んでください。

また、「地理的に分散している複数のシングル・サインオン・インスタンス」で示した配置例について理解しておくこともお薦めします。この項では、シングル・サインオン・レプリケーションが発生する状況について説明します。

ID 管理データベースでレプリケーションを使用可能にする作業手順を次に示します。

1. マルチマスター・レプリケーション・グループをインストールおよび構成する場合は、『Oracle Application Server 高可用性ガイド』のマルチマスター・レプリケーションの設定に関する項を参照してください。シングル・サインオン表は、このプロセスにおいてレプリケートされます。
2. レプリケーション・スクリプトの実行後、管理者はスクリプトを実行してレプリケート・ノード間でスキーマのパスワードを同期化し、Single Sign-On Server とディレクトリ間の接続を確立する必要があります。詳細は、『Oracle Application Server 高可用性ガイド』の Oracle Application Server Single Sign-On スキーマのパスワードの同期化に関する項を参照してください。

   MDS で `ssoreplsetup.jar` ツールを実行し、Single Sign-On スキーマのパスワードを MDS と RMS のデータベース間で同期化します。この手順を RMS ごとに繰り返します。9-14 ページの表 9-1 は、このツールのパラメータの定義です。

   スクリプトを実行する手順は次のとおりです。

   a. *ORACLE_HOME*`/sso/lib` に移動します。
   b. ライブラリ・パスを次のように設定します。
      * UNIX（csh および tcsh）：

        ```
        setenv LD_LIBRARY_PATH $ORACLE_HOME/lib32:$LD_LIBRARY_PATH
        ```

      * Windows:

        ```
        set PATH=%ORACLE_HOME%¥bin:%PATH%
        ```

   c. 次のコマンドを発行します。

      ```
      ORACLE_HOME/jdk/bin/java -jar ssoreplsetup.jar
      [-prompt]
      mds_oid_host
      mds_oid_port
      mds_oid_admin
      mds_oid_password
      mds_ssl_enabled
      rms_oid_host
      rms_oid_port
      rms_oid_admin
      rms_oid_password
      rms_ssl_enabled
      rms_db_sys_password
      [-help]
      ```

表 9-1 ssoReplSetup のパラメータ

パラメータ	説明
mds_oid_host	MDS ディレクトリ・サーバーのホスト名。
mds_oid_port	MDS ディレクトリ・サーバーのポート番号。
mds_oid_admin	バインド DN: MDS ディレクトリ・サーバーへのユーザー認証。
mds_oid_password	MDS ディレクトリ・サーバーのバインド・パスワード。
mds_ssl_enabled	MDS が SSL 対応になっているかどうか。値は Y または N のいずれかです。このパラメータの値では、大文字と小文字は区別されません。 デフォルトでは、ディレクトリおよび Single Sign-On Server は SSL を介して通信するため、このパラメータは通常 Y に設定します。
rms_oid_host	RMS ディレクトリ・サーバーのホスト名。
rms_oid_port	RMS ディレクトリ・サーバーのポート番号。
rms_oid_admin	バインド DN: RMS ディレクトリ・サーバーへのユーザー認証。
rms_oid_password	RMS ディレクトリ・サーバーのバインド・パスワード。
rms_ssl_enabled	RMS が SSL 対応になっているかどうか。値は Y または N のいずれかです。このパラメータの値では、大文字と小文字は区別されません。 デフォルトでは、ディレクトリおよび Single Sign-On Server は SSL を介して通信するため、このパラメータは通常 Y に設定します。
rms_db_sys_password	RMS データベースの SYS パスワード。
-prompt	コンソールからすべての値をプロンプトに表示する場合に指定します。
-help	使用方法を表示する場合に指定します。

注意： 追加する RMS ノードごとに手順 2 を繰り返します。

レプリケーション・グループへのノードの追加

既存のシングル・サインオン・レプリケーション・グループにノードを追加する場合で、Oracle Internet Directory をこのノードにレプリケートしていないときは、『Oracle Internet Directory 管理者ガイド』の操作手順に従います。この新しいノードをシングル・サインオン用に構成するには、シングル・サインオン中間層をインストールして、「レプリケーション用の ID 管理データベースの構成」の手順 2 を繰り返します。

レプリケーション・グループからのノードの削除

シングル・サインオン・レプリケーション・グループからノードを削除するには、『Oracle Internet Directory 管理者ガイド』の操作手順に従います。

プロキシ・サーバーを使用する OracleAS Single Sign-On の配置

OracleAS Single Sign-On の前段にはリバース・プロキシを配置できます。プロキシは、次の様々な機能を備えています。

- Single Sign-On Server のホスト名を非表示にします。
- Single Sign-On Server ではなくプロキシで SSL 接続を終了します。
- ファイアウォールで公開するポート数を制限します。

Single Sign-On Server の前段で使用するプロキシには、以降の構成を適用します。これらの構成は、OracleAS Single Sign-On とプロキシ・サーバーがインストール済であることを前提としています。プロキシをインストールするには、プロキシのベンダーから提供されている操作手順に従ってください。

注意： 操作手順については仮想ホストも同様です。

IP チェックの無効化

Single Sign-On Server の前段で、特定の範囲にわたるプロキシ・アドレスを使用しているネットワーク構成では、シングル・サインオンの IP チェック機能をオフにする必要があります。IP チェックはデフォルトではオフになっていますが、これを確認するには「SSO Server の編集」ページに移動する必要があります。このページへのアクセス方法は、第 2 章の「管理ページへのアクセス」の項を参照してください。「SSO Server の編集」ページが表示されたら、**「SSO Server に出されたリクエストの IP アドレスを確認します」**ボックスの選択が解除されていることを確認します。

プロキシ・サーバーの有効化

プロキシ・サーバーを有効にする手順は次のとおりです。

1. シングル・サインオン中間層で `ssocfg` スクリプトを実行します。このスクリプトによって、Single Sign-On Server に格納されているホスト名がプロキシのホスト名に変更されます。次のコマンドの構文を使用して、プロキシ・サーバーのプロトコル、ホスト名およびポートの値を入力します。

   - UNIX:

     ```
     $ORACLE_HOME/sso/bin/ssocfg.sh http proxy_server_name proxy_port
     ```

   - Windows:

     ```
     %ORACLE_HOME%¥sso¥bin¥ssocfg.bat http proxy_server_name proxy_port
     ```

   サーバーが SSL 用に構成されている場合は、`http` を `https` に置き換え、非 SSL ポートを SSL ポートに置き換えます。

   `ssocfg` を実行したら、シングル・サインオン中間層で `targets.xml` ファイルを更新します。詳細は、7-4 ページの「targets.xml の更新」を参照してください。

2. シングル・サインオン中間層の `httpd.conf` ファイルに以降の行を追加します。このファイルは *ORACLE_HOME*`/Apache/Apache/conf` にあります。

   a. これらの行によって、ディレクティブ `ServerName` が実際のサーバー名からプロキシ名に変更されます。

      ```
      KeepAlive off
      ServerName proxy_host_name
      Port proxy_port
      ```

      SSL を使用している場合は、ポートに `4443` などの SSL ポートを指定する必要があります。

b. （SSL のみ）ブラウザとプロキシ・サーバー間で SSL 通信を構成している場合は、中間層に mod_certheaders を構成します。このモジュールによって、Oracle HTTP Server では、SSL リクエストとして受信する HTTP プロキシ・リクエストを処理できるようになります。次の行を `httpd.conf` に追加します。この行は、ファイルの最後に追加できます。ファイル内での行の場所は重要ではありません。

* 次の行を入力し、モジュールをロードします。

  UNIX:

  ```
  LoadModule certheaders_module libexec/mod_certheaders.so
  ```

  Windows:

  ```
  LoadModule certheaders_module modules/ApacheModuleCertHeaders.dll
  ```

* OracleAS Web Cache をプロキシとして使用している場合は、次の行を入力します。

  ```
  AddCertHeader HTTPS
  ```

  OracleAS Web Cache 以外のプロキシを使用している場合は、次の行を入力します。

  ```
  SimulateHttps on
  ```

3. シングル・サインオン中間層で mod_osso を登録します。この手順によって、実際のホスト名のかわりにプロキシのホスト名を使用するように mod_osso を構成します。登録ツールの実行方法は、第 4 章の「mod_osso の登録」の項を参照してください。

4. Distributed Configuration Management スキーマを更新します。

```
ORACLE_HOME/dcm/bin/dcmctl updateconfig
```

5. シングル・サインオン中間層を再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

6. 複数のシングル・サインオン中間層を配置する場合は、追加する中間層ごとに手順 2 ～ 4 を繰り返します。

7. 次のシングル・サインオン・ログイン URL を使用して、Single Sign-On Server にログインします。

```
http://proxy_host_name:proxy_port/sso/
```

この URL によって、Single Sign-On ホームページが表示されます。ログインできる場合は、プロキシが正しく構成されています。

ユーザー・ニックネームの変更におけるディレクトリ同期の設定

Single Sign-On データベースでは、外部アプリケーションのユーザー・データの格納と参照にユーザー・ニックネームを使用します。ニックネームの属性値が Oracle Internet Directory で変更されると、ユーザーは、新しいユーザー ID でログインするときに資格証明を再入力する必要があります。ユーザーの便宜を図るために、ディレクトリと Single Sign-On データベース間でユーザー名の変更を自動的に同期することができます。Oracle Directory Integration and Provisioning によるこの同期メカニズムは、ユーザーのエントリがディレクトリから削除されると、外部アプリケーションのデータを Single Sign-On データベースから削除します。

ディレクトリと Single Sign-On データベース間でニックネームの変更を同期化する手順は次のとおりです。

1. Oracle Directory Integration and Provisioning サーバーを起動します。詳細は、『Oracle Identity Management 統合ガイド』の管理ツールに関する章を参照してください。
2. 同期パッケージをロードします。最初に *ORACLE_HOME*/sso/admin/plsql/sso に移動し、次に Single Sign-On スキーマに接続します。

   ```
   sqlplus orasso/password
   ```

   orasso パスワードの取得方法は、付録 B を参照してください。
3. 次のパッケージを順番に実行します。

   ```
   SQL> @ssodip.sql
   SQL> @ssodip.pks
   SQL> @ssodip.pkb
   ```
4. シングル・サインオン・プロファイルを Oracle Internet Directory に登録します。この作業を行うには、次の構文でプロビジョニング・サブスクリプション・ツール（oidprovtool）を実行します。

   ```
   ORACLE_HOME/bin/oidprovtool
   operation=create
   ldap_host=oid_host
   ldap_port=oid_port
   ldap_user_dn=cn=orcladmin
   ldap_user_password=orcladmin_password
   schedule=synchronization_interval_in_seconds
   organization_dn=realm_DN
   application_dn=orclApplicationCommonName=ORASSO_SSOSERVER,cn=SSO,
   cn=Products,cn=OracleContext
   interface_name=LDAP_NTFY interface_type=PLSQL
   interface_connect_info=sso_database_host:sso_database_port:sso_
   database_SID:orasso:orasso_schema_password
   event_subscription=USER:user_search_base_for_realm:ADD(attribute_type)
   event_subscription=USER:user_search_base_for_realm:MODIFY(attribute_type)
   event_subscription=USER:user_search_base_for_realm:DELETE
   ```

   レルムを変更する場合は、プロファイルを登録します。ユーザー検索ベースは変更可能です。また、ニックネーム属性タイプも変更可能です。たとえば、uid 属性で cn 属性を置き換えることができます。

   oidprovtool の詳細は、『Oracle Internet Directory 管理者ガイド』の構文を記載した付録を参照してください。
5. Oracle Directory Integration and Provisioning サーバーの権限を orasso としてプロキシに付与します。これには、ディレクトリの orasso エントリを変更します。

   最初に、次の構文で LDIF ファイルを作成します。

   ```
   dn: orclApplicationCommonName=ORASSO_SSOSERVER,cn=SSO,cn=Products,
   cn=OracleContext
   changetype: modify
   add: orclaci
   orclaci: access to entry by group="cn=odisgroup,cn=odi,cn=oracle internet
   directory" (proxy)
   ```

6. スーパー・ユーザー `cn=orcladmin` として、LDIF ファイルをディレクトリにロードします。
7. Oracle Directory Integration and Provisioning サーバーが実行中であることを確認します。

同期のスケジュール状況によっては、ディレクトリに変更が発生した時刻と、その変更が Single Sign-On Server で同期される時刻の間にずれが生じることがあります。このずれのため、ユーザー ID の変更されたユーザーは、同期が最終的に完了するまで外部アプリケーションにアクセスできません。

10

アプリケーション・サービス・プロバイダに対するサポートの有効化

この章では、Oracle Identity Management インフラストラクチャの単一インスタンスで複数のレルムをサポートできるように Single Sign-On Server を有効にする方法を説明します。

この章の項目は次のとおりです。

アプリケーション・サービス・プロバイダ : 複数のレルムの配置に関する決定

アプリケーション・サービス・プロバイダとは、Oracle アプリケーションや Oracle 以外のアプリケーションをインストールしてメンテナンスし、通常は有料でこれらのアプリケーションを顧客が利用できるようにする企業です。このような企業は、同一のアプリケーション・インスタンスで複数のユーザーのグループにサービスを提供することで、規模拡大による収益率の向上を図ります。アプリケーション・サービス・プロバイダは、Oracle Identity Management インフラストラクチャの単一インスタンス内で異なるレルム（または異なるネームスペース）を使用して、別々の顧客に一意の Oracle 構成情報を設定し、格納する場合があります。

複数のレルムを配置するかどうかを決定する際にはユーザー ID が唯一の判断基準であり、ID 間で競合が存在しない場合は、単一のデフォルト・レルムでユーザーを管理することをお薦めします。アプリケーション・サービス・プロバイダのユーザーは、一意の電子メール ID でログインする場合もあります。ユーザー ID 間で競合がある場合は、レルムを別々にする必要があります。複数のレルムの配置は、Oracle10*g* の中間層コンポーネントと顧客アプリケーションの配置方法に影響する点にも注意してください。

注意 : Oracle Identity Management の詳細は、『Oracle Identity Management Administrator's Guide』を参照してください。

複数のレルムの設定と有効化

複数のレルムの設定作業では、OracleAS Single Sign-On を上回るリソースと管理オーバーヘッドが必要となる場合があります。このプロセスには他のコンポーネントが関係します。レルムの構成は実際に、次の 3 つから成るプロセスです。

- Oracle Internet Directory でのレルムの作成
- OracleAS Single Sign-On での複数のレルムの有効化
- パートナ・アプリケーションによる ID 管理レルムの認識

最初のプロセスについては、『Oracle Internet Directory 管理者ガイド』を参照してください。2 番目のプロセスは、この章で説明します。3 番目のプロセスは、製品関連のドキュメントを参照してください。

Single Sign-On Server による複数のレルムの認証の有効化

複数のレルムに対するシングル・サインオンの認証シーケンスは、単一のデフォルト・レルムでのシングル・サインオンの場合とほぼ同様です。ユーザーにとって唯一異なる点は、レルムの最初のタイプに属していたユーザーにログイン画面が表示されたときに（10-3 ページの図 10-1 を参照）、ユーザーはユーザー名とパスワードのみでなく、新しい資格証明（レルム・ニックネーム）も入力する必要がある点です。入力する値の大 / 小文字は、区別されません。

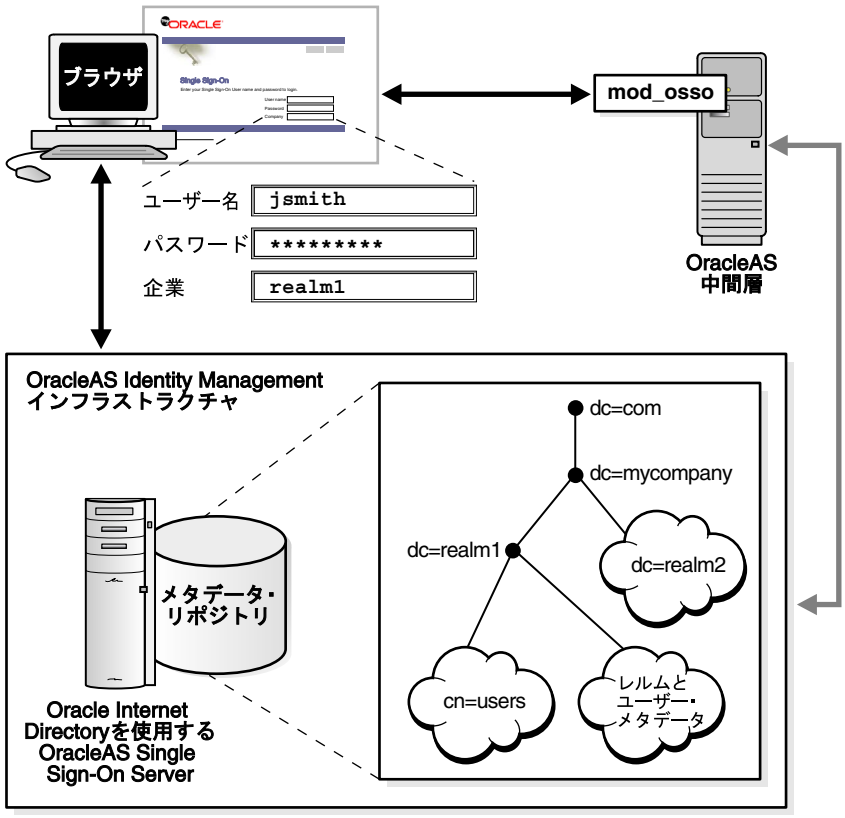
この項の項目は次のとおりです。

- Oracle Internet Directory でのレルムの検索
- パートナ・アプリケーションでのレルムに属するユーザーの検証

Oracle Internet Directory でのレルムの検索

ユーザーが資格証明を入力すると、レルム・ニックネームとユーザー名が Oracle Internet Directory 内のエントリにマップされます。具体的には、Single Sign-On Server がディレクトリ・メタデータを使用して、Oracle Internet Directory 内のレルムのエントリを検索します。このエントリを検出すると、Single Sign-On Server はレルム・メタデータを使用してユーザーを検索します。ユーザーのエントリが検出されると、パスワード（エントリの属性）が検証されます。パスワードの検証が完了すると、ユーザーは認証されます。

図 10-1 全体図 : 複数のレルムでのシングル・サインオン



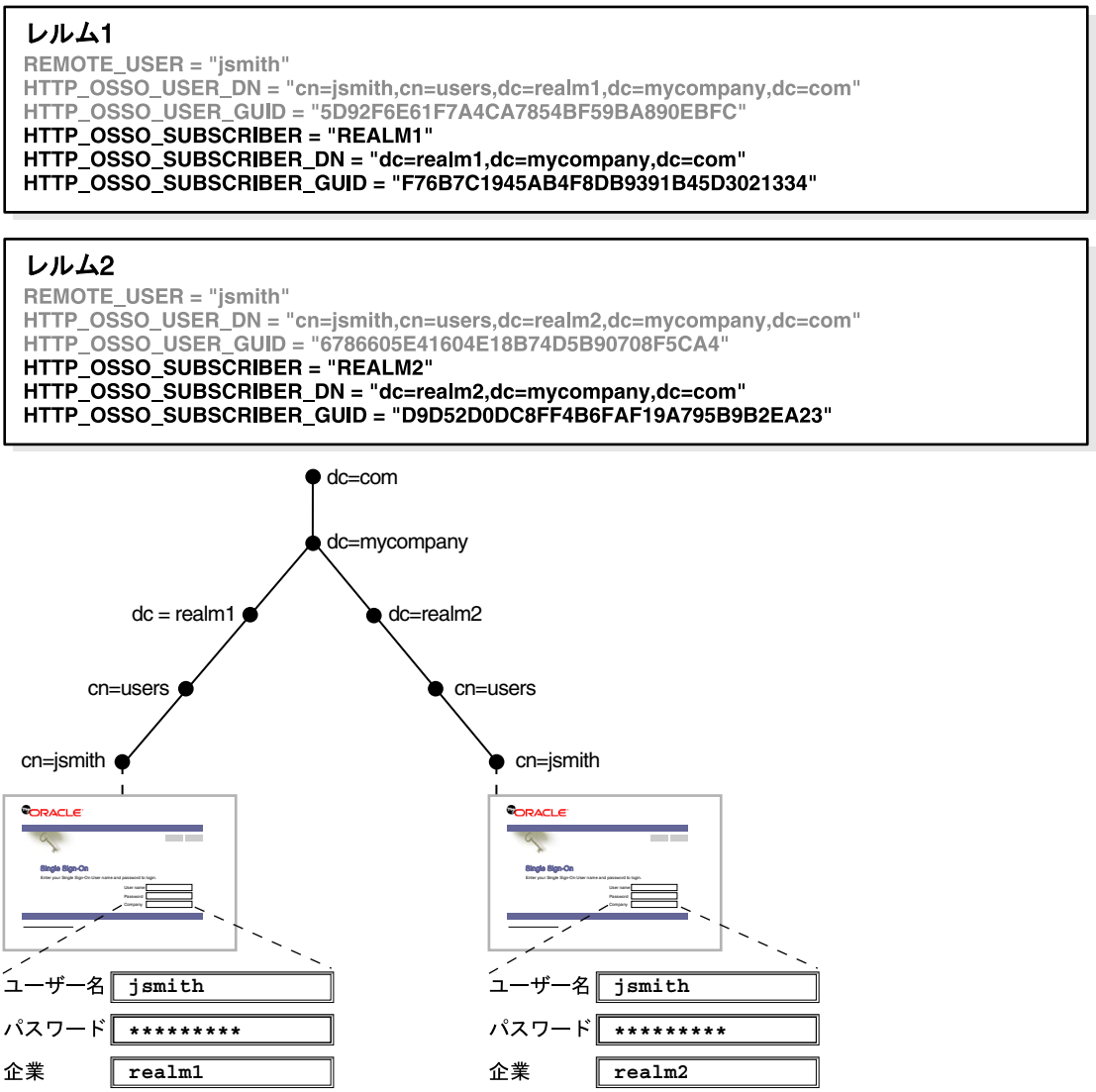
パートナ・アプリケーションでのレルムに属するユーザーの検証

同じニックネームを持ち、異なるレルムに属する 2 人のユーザーが存在する場合、パートナ・アプリケーションではこれらのユーザーを区別するメカニズムが必要になります。パートナ・アプリケーションではコンテンツ（株価ニュースと株式相場表を表示する OracleAS Portal ページなど）をリクエストするレルムのニーズに合うようにコンテンツを対応させる必要があるため、このようなメカニズムが必要となります。したがって、OracleAS リリース 9.0.4 では mod_osso に渡される属性として、レルム・ニックネーム、レルム DN、レルム GUID が追加されています。mod_osso によって Cookie が設定され、取得された属性が HTTP ヘッダーとして格納されます。提供するコンテンツを決定する場合、アプリケーションではファンクション・コールを使用して mod_osso ヘッダーからこれらの属性の 1 つを取得することもあります。

mod_osso ヘッダーと、mod_osso ヘッダーへのアクセスに使用するメソッドの詳細は、『Oracle Identity Management アプリケーション開発者ガイド』の mod_osso に関する章を参照してください。

10-4 ページの図 10-2 では、mod_osso で実行されるアプリケーションが、どのように 2 人のユーザーの HTTP ヘッダーを認識するかを示しています。2 人のユーザーは同じニックネームを持ち、異なるレルムに属しています。アプリケーションでは太字のヘッダーを使用して、2 人のユーザーを区別します。この場合のホスト（またはデフォルト・レルム）は mycompany.com です。

図 10-2 同じ名前を持つユーザーの mod_osso ヘッダー



複数のレルムに対する Single Sign-On Server の構成

複数のレルムに対して Single Sign-On Server を構成する場合は、Single Sign-On スキーマで各レルムのエントリを作成します。Oracle Internet Directory に作成する各レルムには、Single Sign-On スキーマの対応するエントリが必要です。

注意：

- Oracle Internet Directory でレルムを作成してから、Single Sign-On スキーマでレルムを作成します。
- 次の構成スクリプトは UNIX プラットフォームでのみ実行できます。Windows プラットフォームでは実行できません。

複数のレルムに対して Single Sign-On Server を構成する手順は次のとおりです。手順 1、2、5 は一度のみ実行してください。これらの手順によって、複数のレルムに対する Single Sign-On Server が有効になります。手順 3 と 4 は、レルムを追加するたびに実行する必要があります。

1. OracleAS Infrastructure と Single Sign-On Server がインストールされていることを確認します。
2. *ORACLE_HOME*/sso/admin/plsql/wwhost に移動します。

   次の構文を使用して、enblhstg.csh スクリプトを実行します。スクリプト・パラメータの詳細は、10-6 ページの表 10-1 を参照してください。

   ```
   enblhstg.csh -mode sso
                -sc sso_schema_connect_string
                -ss orasso
                -sw sso_schema_password
                -h oid_host_name
                -p oid_port
                -d "cn=orcladmin"
                -w oid_bind_password
   ```

   注意： Single Sign-On Server が分散配置の一部分である場合は、必ず OracleAS のメタデータ・リポジトリを含むコンピュータ上でスクリプトを実行してください。

   次に例を示します。

   ```
   enblhstg.csh -mode sso
                -sc webdbsvr2:1521:s901dev3
                -ss orasso
                -sw xyz
                -h dlsun670.us.oracle.com
                -p 389
                -d "cn=orcladmin"
                -w welcome123
   ```

3. レルムを見つけるか、Oracle Internet Directory にレルムを追加します。これを行うには、『Oracle Internet Directory 管理者ガイド』の操作手順に従います。

   SSO Server データベースで既存のレルムを検索するには、次の SQL 問合せを使用します。

   ```
   sqlplus orassso/password
   select subscriber_id from wwsub_model$;
   ```

4. Single Sign-On データベースでレルムのエントリを作成します。スクリプト *ORACLE_HOME*/sso/admin/plsql/wwhost/addsub.csh を使用します。Single Sign-On Server が分散配置の一部分である場合は、このときも必ず OracleAS のメタデータ・リポジトリを含むコンピュータ上でスクリプトを実行してください。

次の構文を使用して、スクリプトを実行します。

```
addsub.csh -name realm_nickname
           -id realm_ID
           -mode sso
           -sc sso_schema_connect_string
           -ss sso_schema_name
           -sw sso_schema_password
           -h oid_host_name
           -p oid_port
           -d oid_bind_dn
           -w oid_bind_dn_password
           -sp sys_schema_password
```

realm_nickname には OIDDAS を使用して作成されたレルムの名前を指定し、*realm_ID* には既存のその他のレルム値と競合しない整数値を指定します。

10-6 ページの表 10-1 に、enblhstg.csh と addsub.csh のパラメータ定義を示します。

表 10-1 enblhstg.csh と addsub.csh のパラメータ

パラメータ	説明
-mode	この値は sso にする必要があります。
-sc	Single Sign-On スキーマの接続文字列。*host*:*port*:*sid* の形式を使用します。
-ss	Single Sign-On スキーマの名前。このパラメータは orasso にする必要があります。
-sw	Single Sign-On スキーマのパスワード。取得方法については、付録 B を参照してください。
-h	Oracle Internet Directory Server のホスト名。
-p	Oracle Internet Directory Server のポート番号。
-d	Oracle Internet Directory Server のバインド DN。このパラメータの値は cn=orcladmin です。これはディレクトリ・スーパー・ユーザーです。
-w	Oracle Internet Directory スーパー・ユーザー（cn=orcladmin）のパスワード。
-name	レルム・ニックネーム。これはログイン・ページの「企業」フィールドに入力する値です。
-id	レルム ID。それまでに addsub.sh に渡されていない 2 以上の整数を選択します。（値 1 はデフォルト・レルム用に予約されています）。Single Sign-On Server の内部ではレルム ID が索引として使用されます。
-sp	sys スキーマのパスワード。このパスワードは、OracleAS のインストール時に選択されます。

注意：

- スクリプトで重複するサブスクライバ・エントリについて尋ねられた場合は、既存のエントリを使用するオプションを選択します。
- 1 レベルのレルムを作成する場合は、スクリプトに -sd *default_realm_id* および -type db パラメータを含めます。

5. シングル・サインオン中間層を停止して起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

複数のレルム用の管理権限の付与

Oracle Internet Directory では、レルムの作成時にデフォルト・レルムの DIT 構造がレルム間で伝播されます。ただし、デフォルト・レルムの DIT に存在するユーザー、グループおよび権限は、伝播されないので注意してください。ディレクトリ・スーパー・ユーザーまたはレルム管理者は、Oracle Directory Manager を使用して権限を割り当てる（再度割り当てる）必要があります。この用途でのツールの使用方法は、第 2 章の「管理権限の付与」の項を参照してください。

11

Single Sign-On Server の監視

この章では、Oracle システム管理コンソールである Oracle Enterprise Manager を使用して、Single Sign-On Server を監視する方法を説明します。

この章の項目は次のとおりです。

データベース監視パスワードの設定

OracleAS Single Sign-On の監視に使用するデータは Oracle データベースに格納されます。Oracle Enterprise Manager には監視用のユーザー・インタフェースがあります。Oracle Enterprise Manager はユーザー名とパスワードを使用して Oracle データベースに接続します。Single Sign-On の監視が適切に動作するように、Oracle Enterprise Manager でデータベース監視パスワードを設定する必要があります。

詳細は、『Oracle Enterprise Manager アドバンスト構成』の新規のターゲット監視資格証明の指定に関する項を参照してください。

監視用ページへのアクセス

スタンドアロン・コンソールでの Single Sing-On の監視 UI は、ホームページと「失敗ログインの詳細」ページの 2 つのページで構成されています。ホームページには、サーバーの負荷やユーザーの動作についての一般的な情報が表示されます。「失敗ログインの詳細」ページには、特定のユーザーについての失敗したログインの統計情報が表示されます。

Single Sign-On の監視用ホームページにアクセスするには、次の手順を実行します。

1. 管理対象の Oracle Enterprise Manager インスタンスのスタンドアロン・コンソールに移動します。

   OracleAS インスタンスをホスティングしているコンピュータのホスト名と Oracle Enterprise Manager のポート番号を入力します。デフォルトのポート番号は `1156` です。次のファイルから、インスタンス固有のポート番号を調べることができます。

   UNIX: *`ORACLE_HOME`*`/install/setupinfo.txt`

   Windows: *`ORACLE_HOME`*`¥install¥setupinfo.txt`

2. OracleAS 管理者の資格証明を使用してログインします。
3. 「ファーム」ページの**「スタンドアロン・インスタンス」**セクションで、適切な OracleAS インスタンスを選択します。
4. 「Application Server」ページの**「システム・コンポーネント」**リストで、Single Sign-On Server を選択します。

スタンドアロン・コンソールのホームページの解説と使用方法

ホームページ（11-3 ページの図 11-1 を参照）では、**「一般」**セクションに次の情報が表示されます。

- 状態

  緑の上向き矢印によって、Single Sign-On Server が実行中であることが示されます。赤の下向き矢印によって、サーバーが停止していることが示されます。

- 起動時間

  Single Sign-On スキーマを提供するデータベースの開始時刻。

- データベース

  Single Sign-On スキーマを提供するデータベースの SID/ インスタンス名。

- データベースのバージョン

  Single Sign-On スキーマを提供するデータベースのバージョン。

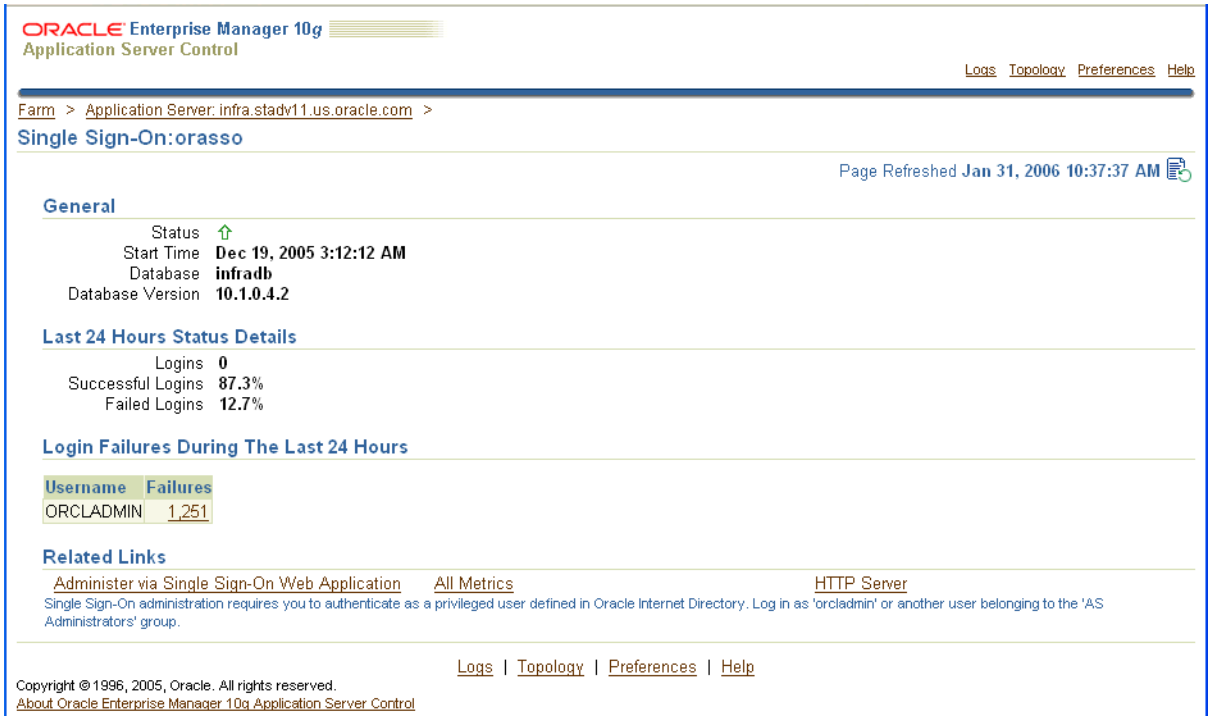
「過去 24 時間の状態の詳細」セクションには、次のメトリックがあります。

- ログイン
- 成功ログイン
- 失敗ログイン

セクション名のとおり、24 時間前から現在までの統計情報が表示されます。

「過去 24 時間の失敗ログイン」セクションでは、24 時間以内に発生したログインの失敗数を確認できます。「過去 24 時間の失敗ログイン」表で名前を選択します。**「失敗」**ヘッダーの下にある関連するリンクを選択します。このリンク先には、ユーザーのログインの失敗数が含まれています。リンクをクリックすると、「失敗ログインの詳細」ページが表示されます。

図 11-1 OracleAS Single Sign-On の監視用ホームページ



「関連リンク」セクションには、次のリンクがあります。

- HTTP Server

  Oracle HTTP Server の監視用ホームページが表示されます。

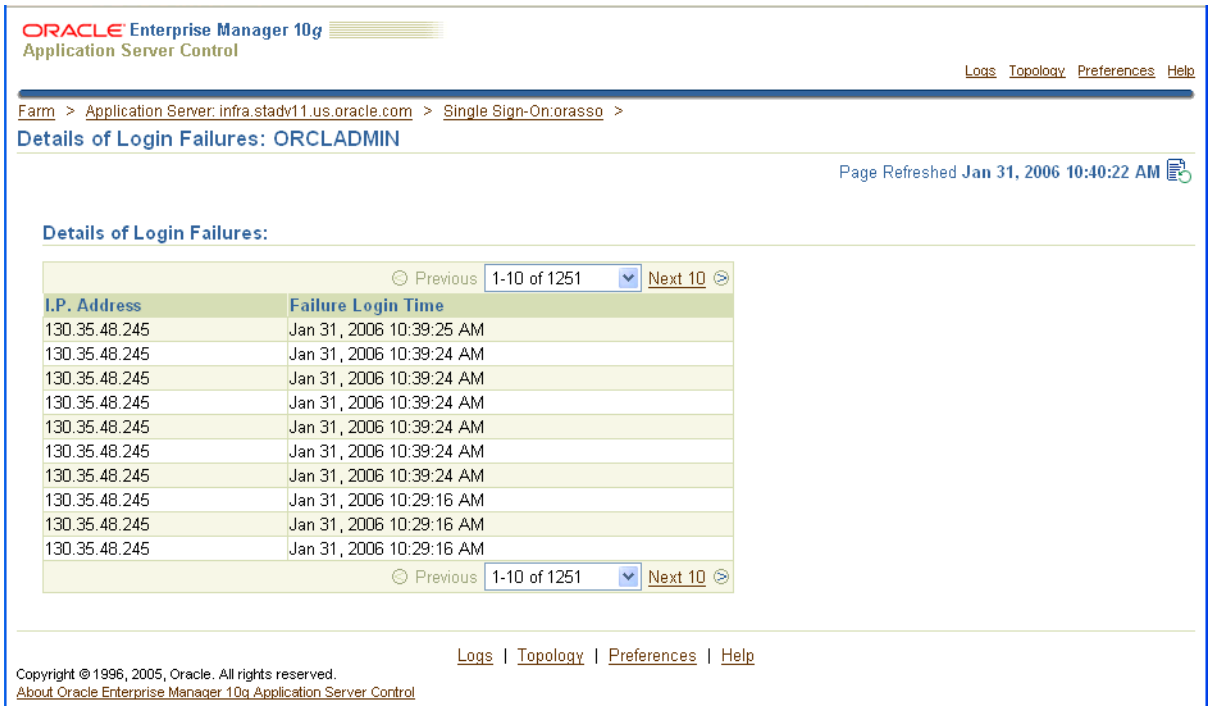
- Single Sign-On Web アプリケーション経由で管理

  Single Sign-On の管理用のホームページが表示されます。

「失敗ログインの詳細」ページの表示内容と使用方法

「過去 24 時間の失敗ログイン」表のリンクをクリックすると、「失敗ログインの詳細」ページ（図 11-2）が表示されます。このページの表には、特定のユーザーについてのログインに失敗した回数および関連 IP アドレスが表示されます。

図 11-2「失敗ログインの詳細」ページ



Single Sign-On の監視ターゲットのポート・プロパティの更新

Oracle HTTP Server のポート番号を変更した場合、そのサーバー上にある Single Sign-On の監視ターゲットのポート・プロパティも変更する必要があります。次の手順を実行して、変更を適用します。

1. `targets.xml` ファイルをバックアップします。

   ```
   cp ORACLE_HOME/sysman/emd/targets.xml ORACLE_HOME/sysman/emd/
   targets.xml.backup
   ```

   このファイルは、Oracle Enterprise Manager で監視される様々なターゲット（OracleAS Single Sign-On など）の構成ファイルです。

2. `targets.xml` でターゲット・タイプ `oracle_sso_server` を検索し、このターゲット・タイプに関連付けられた HTTP ポート値を見つけて編集します。

   ```
   <Property NAME="HTTPPort" VALUE="7777"/>
   ```

3. ファイルを保存して閉じます。

4. OracleAS Console をリロードします。

   ```
   ORACLE_HOME/bin/emctl reload
   ```

注意： ポート依存性の変更の詳細は、『Oracle Application Server 管理者ガイド』のポート番号に関する付録を参照してください。

OracleAS Web Cache インスタンスを使用したサーバーの監視

OracleAS Web Cache を複数のシングル・サインオン・インスタンスのロード・バランサとして使用している場合、OracleAS Web Cache コンピュータから Single Sign-On Server を監視できます。同時に、各シングル・サインオン・インスタンスの監視用ページからはそのインスタンスを監視できます。

OracleAS Web Cache インスタンスにシングル・サインオン・ターゲットを追加するには、次の手順を実行します。

1. OracleAS Web Cache インスタンス上にある、ファイル *ORACLE_HOME*/sysman/emd/targets.xml をバックアップします。
2. シングル・サインオン・インスタンスにある targets.xml ファイルから、シングル・サインオン・ターゲットの定義をコピーします。コピーした定義を、OracleAS Web Cache インスタンス上の targets.xml ファイルの末尾に貼り付けます。その際、終了タグ </Targets> の直前に挿入します。
3. Target TYPE タグ内にある、シングル・サインオン・インスタンスの oracle_ias ターゲットの名前を、OracleAS Web Cache インスタンスの oracle_ias ターゲットの名前に置き換えます。
4. シングル・サインオンの OracleHome 値を、OracleAS Web Cache の OracleHome 値に置き換えます。
5. HTTPMachine、HTTPPort および HTTPProtocol の値を、OracleAS Web Cache インスタンスの対応する値に変更します。
6. 次のコマンドを実行し、変更内容を適用します。

   ORACLE_HOME/bin/emctl reload

SSL 対応の Single Sign-On Server の監視

SSL 対応の Single Sign-On Server を監視する場合のガイドラインは、次のとおりです。Grid Control を使用して監視する場合も使用しない場合も同様です。

- シングル・サインオン中間層の *ORACLE_HOME*/sysman/emd/targets.xml ファイルの oracle_sso_server ターゲット・タイプが適切な HTTP 属性を持っている必要があります。

  たとえば、SSL 対応の Single Sign-On Server に次の URL で接続するとします。

  http://myhost.us.oracle.com:4443/sso

  この場合、HTTPPort は 4443 で、HTTPProtocol は https です。

  詳細は、第 7 章の「targets.xml の更新」の項を参照してください。

- Oracle Enterprise Manager の証明書構成ファイルに、インフラストラクチャ・サーバーの証明書が含まれている必要があります。

  構成ファイルに証明書を追加する方法は、『Oracle Enterprise Manager アドバンスト構成』の Oracle Enterprise Manager セキュリティに関する章を参照してください。特に、HTTPS を介して Web アプリケーションを監視するためのビーコンの構成方法に関する項を参照してください。

12

配置固有ページの作成

Oracle Application Server Single Sign-On のフレームワークでは、配置固有のログイン・ページ、パスワードの変更ページ、シングル・サインオフ・ページを Single Sign-On Server と統合できます。つまり、ユーザーは独自のルック・アンド・フィール要件やグローバリゼーション要件に合せてこれらのページをカスタマイズできます。

ただし、オラクル社では JavaServer Pages（JSP）ページの使用をお薦めします。他の Web テクノロジを使用すると、一貫性がなくなる場合があります。PL/SQL ページはサポートされていません。サンプル・ページが製品に付属しています。Oracle Application Server Single Sign-On 製品には、Oracle Application Server でのテスト用に設計されたサンプル・ページが付属しています。

この章の項目は次のとおりです。

Single Sign-On Server での配置固有ページの使用方法

シングル・サインオンのページを有効にするプロセスは、次のとおりです。

1. ユーザーはパートナ・アプリケーションをリクエストし、Single Sign-On Server にリダイレクトされます。
2. ユーザーが認証されない場合、Single Sign-On Server はユーザーをサンプル・ログイン・ページまたは配置固有ページにリダイレクトします。このリダイレクションの一環として、12-3 ページの表 12-1 に示すパラメータがサーバーからページに渡されます。
3. ユーザーはログイン・ページを送信します。これにより、12-3 ページの表 12-2 に示すパラメータが次の認証 URL に渡されます。

   `http://sso_host:sso_port/sso/auth`

   または

   `https://sso_host:sso_ssl_port/sso/auth`

   これらのパラメータのうち少なくとも 2 つ（`ssousername`、`password`）は変更可能なフィールドとしてページに表示されます。
4. ユーザー・パスワードの有効期限が切れるまでに十分な時間があり、Single Sign-On Server でユーザー名とパスワードが正しく検証されると、ユーザーはアプリケーションの成功 URL にリダイレクトされます。認証に失敗した場合、ユーザーはログイン・ページに再度リダイレクトされ、エラー・メッセージが表示されます。
5. ユーザー・パスワードの有効期限が近い場合は、ログイン・ページではなく、パスワードの変更ページが表示されます。また、配置固有のパスワードの変更ページを使用するようにサーバーが構成されている場合、ユーザーはこのページの URL にリダイレクトされ、12-4 ページの表 12-3 に示すパラメータがページに渡されます。

   > **注意：** 手順 5 では、ディレクトリ管理者がユーザーにパスワードの変更を強制した場合、パスワードの有効期限が過ぎているかどうかにかかわらず、前述の同じ条件が当てはまります。

   ユーザーは古いパスワード、新しいパスワード、確認用の新しいパスワードを入力して、パスワードの変更ページを送信します。このページからは、12-5 ページの表 12-4 に示すパラメータが次のパスワードの変更 URL に渡されます。

   `http://sso_host:sso_port/sso/ChangePwdServlet`

   または

   `https://sso_host:sso_ssl_port/sso/ChangePwdServlet`

   エラーが発生した場合、ユーザーはパスワードの変更ページにリダイレクトされ、エラー・メッセージが表示されます。エラーが発生する条件の詳細は、第 3 章の「パスワードの変更ページの動作」の項を参照してください。

   パスワードの変更に成功した場合、ユーザーは、認証リクエストをトリガーしたパートナ・アプリケーション URL にリダイレクトされます。
6. ユーザーがシングル・サインオン・セッションを終了するには、作業中のパートナ・アプリケーションで**「ログアウト」**をクリックします。これにより、アプリケーションのログアウト URL が同時にコールされ、ユーザーはアクセスしたすべてのアプリケーションからログアウトされ、シングル・サインオン・セッションが終了します。
7. ユーザーは、シングル・サインオフ・ページを表示する Single Sign-On Server にリダイレクトされます。配置固有ページを使用するようにサーバーが構成されている場合、ユーザーはこのページの URL にリダイレクトされ、12-6 ページの表 12-5 に示すパラメータがページに渡されます。
8. ユーザーはシングル・サインオフ・ページで「戻る」をクリックすると、ログアウトを開始したアプリケーションに戻ることができます。

注意： パスワードの変更ページを使用してパスワードを変更できるのは、パスワードの有効期限が近い場合のみです。Oracle Delegated Administration Services の UI を使用すると、いつでもパスワードを変更できます。このトピックの詳細は、第 3 章の「パスワードの変更ページの動作」の項を参照してください。

配置固有ページの記述方法

ログイン・ページ、パスワードの変更ページおよびシングル・サインオフ・ページの URL では、ページが適切に動作するために、以降の表に示すパラメータを受け入れる必要があります。

この項の項目は次のとおりです。

ログイン・ページのパラメータ

ログイン・ページの URL では、12-3 ページの表 12-1 に示すパラメータを受け入れる必要があります。

表 12-1 Single Sign-On Server によってページに送信されるログイン・ページのパラメータ

パラメータ	説明
site2pstoretoken	ログイン処理用の認証リクエスト・トークンが含まれます。
ssousername	ユーザー名が含まれます。
p_error_code	エラー・コードが文字列として含まれます。認証中にエラーが発生した場合に渡されます。
p_cancel_url	**「取消」**がクリックされたときにリダイレクトする URL が含まれます（ログイン・ページに「取消」ボタンがある場合）。この URL は、ログアウトを開始したパートナ・アプリケーションのホーム URL を指します。
locale	ユーザーの言語（オプション）。ISO 形式にする必要があります。たとえば、フランス語の場合は `fr-fr` です。このパラメータの詳細は、「グローバリゼーション・サポートの追加」を参照してください。

ログイン・ページでは、表 12-2 に示すパラメータを次の認証 URL に渡す必要があります。

`http://`*`sso_host`*`:`*`sso_port`*`/sso/auth`

表 12-2 ページから Single Sign-On Server に送信されるログイン・ページのパラメータ

パラメータ	説明
site2pstoretoken	ログイン処理のリダイレクト URL 情報が含まれます。
ssousername	ユーザー名が含まれます。UTF-8 形式でエンコードされている必要があります。
password	ユーザーによって入力されたパスワードが含まれます。UTF-8 形式でエンコードされている必要があります。

表 12-2 ページから Single Sign-On Server に送信されるログイン・ページのパラメータ（続き）

パラメータ	説明
subscribername	レルムが有効な場合のサブスクライバ・ニックネーム。UTF-8 形式でエンコードされている必要があります。 **注意：** このフィールドは、Single Sign-On Server で複数のレルムが有効な場合にのみ、ログイン・ページで必須になります。
locale	ユーザーの言語（オプション）。ISO 形式にする必要があります。たとえば、フランス語の場合は fr-fr です。このパラメータの詳細は、「グローバリゼーション・サポートの追加」を参照してください。
v	ページ・バージョンが含まれます。推奨されていますが、オプションです。パラメータが渡される場合、値は v1.4 にする必要があります。

ログイン・ページには少なくとも、パラメータ名が ssousername のテキスト・フィールドと、パラメータ名が password のパスワード・フィールドが必要です。これらの値は認証 URL に送信されます。ログイン・ページからは、site2pstoretoken も隠しパラメータとして送信する必要があります。ログイン・ページは、このパラメータをログイン URL に送信する必要があります。

ログイン・ページでは、これらのパラメータの送信に加え、p_error_code に指定された適切なエラー・メッセージの表示、「取消」がクリックされた場合の p_cancel_url へのリダイレクトが行われます。

パスワードを忘れた場合

ログイン・ページを作成する場合、ユーザーがパスワードをリセットするためのリンクを追加できます。この URL からは、Oracle Delegated Administration Service のホームページまたは Oracle Delegated Administration Service 内の「パスワードを忘れた場合」リンクに移動できます。「パスワードを忘れた場合」リンクをクリックしたユーザーには質問が用意されています。その質問に正確に答えないと、ユーザーはパスワードをリセットできません。

Oracle Delegated Administration Service は通常、OracleAS Single Sign-On と同じコンピュータで、次のフォームの URL によってアクセスできます。

http://*sso_host*:*sso_port*/oiddas/

「パスワードを忘れた場合」リンクを使用したパスワードのリセット方法は、『Oracle Identity Management 委任管理ガイド』の Oracle Internet Directory セルフサービス・コンソールに関する章を参照してください。

パスワードの変更ページのパラメータ

パスワードの変更ページの URL では、表 12-3 に示すパラメータを受け入れる必要があります。

表 12-3 パスワードの変更ページに送信されるパラメータ

パラメータ	説明
p_username	ページの上に表示されるユーザー名が含まれます。
p_subscribername	ホスティングが有効な場合のサブスクライバ・ニックネーム。 **注意：** このフィールドは、ログイン・ページに必須です。
p_error_code	前回のパスワード変更時にエラーが発生していた場合、文字列形式のエラー・コードが含まれます。
p_done_url	パスワードの保存後に戻る、ページの URL が含まれます。
site2pstoretoken	パスワードの有効期限が過ぎているか、近い場合に、/sso/auth ログイン URL から要求される site2pstoretoken が含まれます。

表 12-3 パスワードの変更ページに送信されるパラメータ（続き）

パラメータ	説明
p_pwd_is_exp	パスワードの有効期限が過ぎているか、近いことを示すフラグ値が含まれます。値は WARN または FORCE のどちらかです。関連するエラー・コードは表 12-8 を参照してください。
locale	ユーザーの言語（オプション）。ISO 形式にする必要があります。たとえば、フランス語の場合は fr-fr です。このパラメータの詳細は、「グローバリゼーション・サポートの追加」を参照してください。

パスワードの変更ページでは、表 12-4 に示すパラメータを次のパスワードの変更 URL に渡す必要があります。

http://*sso_host*:*sso_port*/sso/ChangePwdServlet

表 12-4 ページで送信されるパスワードの変更ページのパラメータ

パラメータ	説明
p_username	ページの上に表示されるユーザー名が含まれます。パスワードの変更ページから、隠しフィールドとして送信する必要があります。UTF-8 形式でエンコードされている必要があります。
p_old_password	古いパスワードが含まれます。UTF-8 形式でエンコードされている必要があります。
p_new_password	新しいパスワードが含まれます。UTF-8 形式でエンコードされている必要があります。
p_new_password_confirm	新しいパスワードの確認入力が含まれます。UTF-8 形式でエンコードされている必要があります。
p_done_url	パスワードの保存後に戻る、ページの URL が含まれます。
p_pwd_is_exp	パスワードの有効期限が過ぎているか、近いことを示すフラグ値が含まれます。値は WARN または FORCE のどちらかです。関連するエラー・コードは表 12-8 を参照してください。
site2pstoretoken	ログイン処理のリダイレクト URL 情報が含まれます。
p_action	変更をコミットします。値は OK（コミット）または CANCEL（無視）にする必要があります。
p_subscribername	ページの上に表示されるユーザー名が含まれます。
p_request	ユーザーがリクエストする、保護された URL。
locale	ユーザーの言語（オプション）。ISO 形式にする必要があります。たとえば、フランス語の場合は fr-fr です。 「グローバリゼーション・サポートの追加」を参照してください。

パスワードの変更ページには、少なくとも p_old_password、p_new_password、p_new_password_confirm の 3 つのパスワード・フィールドが必要です。このページでは、これらのフィールドをパスワードの変更 URL に送信する必要があります。

パスワードの変更ページからは、隠しパラメータとして p_done_url もパスワードの変更 URL に送信する必要があります。また、p_error_code の値に応じて、エラー・メッセージを表示する必要があります。

シングル・サインオフ・ページのパラメータ

シングル・サインオフ・ページの URL では、表 12-5 に示すパラメータを受け入れる必要があります。

表 12-5 シングル・サインオフ・ページに送信されるパラメータ

パラメータ	説明
p_app_name[1. . .n]	ページの上に表示されるアプリケーション名が含まれます。変数 n は、シングル・サインオフで管理するパートナ・アプリケーションの数です。
p_app_logout_url[1. . .n]	アプリケーションのログアウト URL が含まれます。変数 n は、シングル・サインオフで管理するパートナ・アプリケーションの数です。
p_done_url	戻るページの URL が含まれます。この URL により、ユーザーはログアウトを開始したアプリケーションに戻ります。
locale	ユーザーの言語（ISO 形式）。ログイン時にユーザーが同じ値を渡さない場合にのみ、送信されます。

外部アプリケーション・ログイン・ページのパラメータ

外部アプリケーション・ログイン・ページの URL では、表 12-6 に示すパラメータを受け入れる必要があります。

表 12-6 外部アプリケーション・ログイン・ページに送信されるパラメータ

パラメータ	説明
ID	外部アプリケーション ID。この ID は「外部アプリケーション管理」ページに表示されます。OracleAS Single Sign-On で構成された各外部アプリケーションに対して、それぞれ一意の ID が生成されます。この ID は、外部アプリケーションに対応する表の主キーになります。外部アプリケーション・ログイン・ページは、この ID をアプリケーションに返す必要があります。
p_app_name	ページの上に表示されるアプリケーション名が含まれます。これは、OracleAS Single Sign-On にアプリケーションが構成されたときに、外部アプリケーションに対して指定された名前です。
extappfieldname1..9	外部フィールド名。各外部アプリケーションは、最大で 9 つの追加フィールドを関連付けることができます。これらのフィールドは表示することも、表示しないこともできます。表示可能なフィールドは外部アプリケーション・ログイン・ページに表示され、ユーザーはこのフィールドのデフォルト値を変更できます。表示されないフィールドの値も外部アプリケーションに送信されますが、ユーザーはその値を変更できません。たとえば、ログイン・フィールド、パスワード・フィールド、ロケール・フィールドがあるアプリケーションに対して、値が FR である LO というフィールドを追加できます。詳細は、5-2 ページの「外部アプリケーションの追加」を参照してください。
extappfieldvalue1..9	外部フィールド値。
extappfielddisplay1..9	外部フィールドが表示可能かどうか（true または false）。フィールドが表示され変更できるのか（true）、または固定値を持つのか（false）を指定します。
mode	このパラメータはカスタム・ログイン・ページに渡される場合と、渡されない場合があります。ログイン・ページに渡す場合は、値を modify に設定して送信する必要があります。この場合、データベース内のユーザーの資格証明を更新するためにポータルから外部アプリケーション・ログイン・ページが呼ばれたことを、Single Sign-On アプリケーション・コントローラに示します。

表 12-6 外部アプリケーション・ログイン・ページに送信されるパラメータ（続き）

パラメータ	説明
p_error_code	前回のパスワード変更時にエラーが発生していた場合、文字列形式のエラー・コードが含まれます。
done	ユーザーの資格証明が更新された後にリダイレクトする必要があるレスポンスへの URL。これは、modify モードの場合に使用されます。

このページは、POST メソッドを使用して、表 12-7 に示すパラメータを外部アプリケーション・ログイン・コントローラに送信できる必要があります。

表 12-7 外部アプリケーション・ログイン・ページがアプリケーションに送信するパラメータ

パラメータ	説明
ID	外部アプリケーション ID。
p_app_username	アプリケーションにログインしているユーザーのユーザー名が含まれます。
p_app_pwd	ユーザーが送信するパスワード。
p_remember_credentials	アプリケーションのユーザー名およびパスワードをデータベースに保存する必要があるかどうかを、外部アプリケーション・ログイン・コントローラに示すフラグ。
extappfieldname1..9	外部フィールド名。詳細は、表 12-6 を参照してください。
extappfieldvalue1..9	外部フィールド値。
extappfielddisplay1..9	外部フィールドが表示可能かどうか（true または false）。詳細は、表 12-6 を参照してください。
p_change_password	モードが change password に設定されているかどうかを、外部アプリケーション・ログイン・コントローラに示すフラグ（true または false）。
mode	このパラメータはカスタム・ログイン・ページに渡される場合と、渡されない場合があります。ログイン・ページに渡す場合は、値を modify に設定して送信する必要があります。この場合、データベース内のユーザーの資格証明を更新するためにポータルから外部アプリケーション・ログイン・ページが呼ばれたことを、Single Sign-On アプリケーション・コントローラに示します。
done	ユーザーの資格証明が更新された後にリダイレクトする必要があるレスポンスへの URL。

ページのエラー・コード

ログイン・ページおよびパスワードの変更ページの URL では、ページが適切に動作するために、以降の表に示すプロセス・エラーを受け入れる必要があります。

ログイン・ページのエラー・コード

ログイン・ページでは、表 12-8 に示すエラー・コードを処理する必要があります。

表 12-8 ログイン・ページのエラー・コード

p_error_code の値	対応するメッセージと説明
acct_lock_err	説明 : ユーザーがログインに失敗した回数が多すぎます。 メッセージ : 「アカウントがロックされています。システム管理者に通知してください。」
pwd_exp_err	説明 : ユーザー・パスワードがすでに期限切れです。 メッセージ : 「パスワードが期限切れです。管理者に連絡して、パスワードをリセットしてください。」
null_uname_pwd_err	説明 : ユーザー名フィールドが空です。 メッセージ : 「有効なユーザー名を入力する必要があります」
auth_fail_exception	説明 : 認証に失敗しました。 メッセージ : 「認証に失敗しました。再試行してください。」
null_password_err	説明 : パスワード・フィールドが空です。 メッセージ : 「ログイン・パスワードを入力する必要があります」
sso_forced_auth	説明 : アプリケーションが認証を要求しています。 メッセージ : 「アクセスしようとしているアプリケーションに以前サインインしている場合でも、再度サインインする必要があります。」
unexpected_exception	説明 : 認証時に予期しないエラーが発生しました。 メッセージ : 「予期しないエラーが発生しました。再試行してください。」
unexp_err	説明 : 予期しないエラーが発生しました。 メッセージ : 「予期しないエラーが発生しました。管理者に通知してください」
internal_server_err	説明 : 内部サーバー・エラーが報告されました。 メッセージ : 「内部サーバー・エラーです。管理者に通知してください」
internal_server_try_again_err	説明 : 内部サーバー・エラーが報告され、再試行するように指示されました。 メッセージ : 「内部サーバー・エラーです。操作を再試行してください」
internal_server_try_later_err	説明 : 内部サーバー・エラーが報告され、後で再試行するように指示されました。 メッセージ : 「内部サーバー・エラーです。後で操作を実行してください。」

表 12-8 ログイン・ページのエラー・コード（続き）

p_error_code の値	対応するメッセージと説明
gito_err	説明 : 非アクティブのタイムアウトです。再度ログインする必要があります。 メッセージ : 「シングル・サインオン・セッションが期限切れになっています。セキュリティのため、一定の時間操作がないとセッションは期限切れになります。再度サインインしてください。」
cert_auth_err	説明 : 証明書によるサインオンに失敗しました。証明書が有効であることを確認し、有効でなければ管理者に連絡してください。 メッセージ : 「証明書ベースのサインインに失敗しました。証明書が有効であるか確認してください。有効でない場合はシステム管理者に連絡してください。」
session_exp_error	説明 : シングル・サインオン・セッションの時間制限に達しました。 メッセージ : 「シングル・サインオン・セッションが期限切れになっています。セキュリティのため、指定された時間が経過するとセッションは期限切れになります。再度サインインしてください。」
userid_mismatch	説明 : 強制認証中に入力されたユーザー ID が、現在のシングル・サインオン・セッションのユーザー ID と一致しません。 メッセージ : 「認証用に送信されたユーザー名は、既存のシングル・サインオン・セッションに存在するユーザー名と一致しません。」

ログイン後のメッセージ

ユーザーが認証された後に表示されるメッセージを表 12-9 に示します。これらのメッセージは、ログイン・ページで処理されますが、パスワード変更ページに表示されることもあります。

表 12-9 ログイン後のメッセージ

p_error_code の値	対応するメッセージと説明
pwd_expiry_warn_err	説明 : ユーザーのパスワードがもうすぐ期限切れになります。 メッセージ : 「パスワードがもうすぐ期限切れになります。変更してください。」
pwd_force_change_err	説明 : ユーザーのパスワードが期限切れです。変更する必要があります。 メッセージ : 「続行する前にパスワードを変更する必要があります。」
pwd_grace_login_err	説明 : ユーザーのパスワードが期限切れですが、再設定のための猶予期間中です。 メッセージ : 「パスワードが期限切れです。ログインの猶予期間です。パスワードを変更してください。」

パスワードの変更ページのエラー・コード

パスワードの変更ページでは、表 12-10 に示すエラー・コードを処理する必要があります。

表 12-10 パスワードの変更ページのエラー・コード

p_error_code の値	対応するエラー
confirm_pwd_fail_txt	古いパスワードと新しいパスワードが一致しません。
null_new_pwd_err	新しいパスワードが入力されていません。
null_old_pwd_err	古いパスワードが入力されていません。
pwd_expiry_warn_err	パスワードの有効期限が近づいています。
pwd_force_change_err	ユーザーは先に進む前に、パスワードを変更する必要があります。
pwd_grace_login_err	パスワードは期限切れですが、猶予期間ログインが許可されています。
account_deactivated_err	ユーザー・アカウントが無効です。
acct_lock_err	ユーザー・アカウントがロックされています。
pwd_illegal_value	パスワードに無効な値が含まれています。
pwd_in_history_err	パスワードがパスワード履歴に存在します。
pwd_min_length_err	パスワードが最小文字数の要件を満たしていません。
pwd_numeric	パスワードが数字の要件を満たしていません。

外部アプリケーション・ログインの変更ページのエラー・コード

外部アプリケーション・ログイン・ページでは、表 12-10 に示すエラー・コードを処理する必要があります。

表 12-11 外部アプリケーション・ログイン・ページのエラー・コード

p_error_code の値	対応するエラー
eapp_name_null	ユーザー ID が指定されていません。
eapp_pwd_null	パスワードが指定されていません。
ext_app_not_found	外部アプリケーションを識別できません。

グローバリゼーション・サポートの追加

OracleAS Single Sign-On のフレームワークでは、配置のニーズに合せて配置固有ページをグローバル化できます。表示されるページの言語を決定するための様々な方法があります。2 つの方法を後の項で説明します。

サポートされている言語コードの詳細なリストは、次の URL にある『Oracle Application Server グローバリゼーション・サポート・ガイド』の付録 A を参照してください。

http://www.oracle.com/technology/documentation/index.html

この URL で表示されたページから、OracleAS Single Sign-On のマニュアルへのリンクをクリックし、適切なリリースのライブラリを表示する「View Library」リンクをクリックします。

表示されるページの言語の決定

この項では、HTTP Accept-Language ヘッダーまたは配置ページのロジックを使用して、表示する言語を選択する方法について説明します。

Accept-Language ヘッダーを使用してページを決定する方法

ブラウザを使用すると、エンド・ユーザーは、Web コンテンツを表示する言語（ロケール）を決定できます。ブラウザでは、ユーザーが選択した言語が、HTTP Accept-Language ヘッダーとしてサーバーに送信されます。配置固有ページのロジックでは、このヘッダーを調べ、ページをレンダリングする必要があります。Single Sign-On Server では、このページを受け取ると、Accept-Language ヘッダーの値を読み取り、ユーザー ID の伝播時にその値をパートナ・アプリケーションに送信します。多くのパートナ・アプリケーションでは、ユーザーがこのヘッダーをオーバーライドできますが、シングル・サインオフ・ページはサインオン時に確立された言語で表示される点に注意してください。このため、すべてのパートナ・アプリケーションで同じセッション言語が使用されます。

Accept-Language ヘッダーは、言語を決定する際の推奨メカニズムです。この方法の主な利点は、エンド・ユーザーが他の Web サイトを参照している間に、言語をすでに設定している可能性が高いということです。そのため、これらのページとシングル・サインオンのページ間で参照の一貫性が保たれます。

ページのロジックを使用して言語を決定する方法

オラクル社では前述の方法をお薦めします。ただし、ブラウザで設定された言語を拡張（またはオーバーライド）するメカニズムに基づいてグローバリゼーションを実装することもできます。たとえば、次のいずれかの方法があります。

- ログイン・ページに言語一覧を表示し、ユーザーが選択できるようにします。ユーザーの便宜を考慮して、永続 Cookie を設定してこの選択を永続的なものにすることもできます。
- 言語を 1 つ設定して、ページをレンダリングします。複数のユーザーが 1 つの言語を使用する場合は、この方法が適しています。
- 集中管理されたアプリケーション・リポジトリまたはディレクトリから言語を取得します。ユーザー設定項目、システム設定項目、構成データの集中管理されたストアは、言語を格納するのに最適です。

ページのロジックを使用して言語を設定する場合、ページではこの情報を Single Sign-On Server に伝播する必要があります。Single Sign-On Server では、この情報をパートナ・アプリケーションに伝播する必要があります。最終的には、一貫性のあるグローバリゼーションが保たれます。ページでは、ログイン・フォームの `locale` パラメータ（表 12-2）を使用して、ISO-639 形式で言語を渡す必要があります。多くのサイトには、ISO-639 の 2 文字言語コードの全一覧があります。次のサイトにもこの一覧があります。

```
http://www.ics.uci.edu/pub/ietf/http/related/iso639.txt
```

次のサイトには、ISO-3166 の 2 文字国コードの全一覧があります。

```
http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html
```

注意： `locale` パラメータが Single Sign-On Server に渡されると（表 12-1）、パラメータ値が mod_osso に送信されます。mod_osso はこの値を HTTP Accept-Language ヘッダーの先頭に追加してから、ヘッダーをパートナ・アプリケーションに渡します。

ページのレンダリング

エンド・ユーザーのロケールが決定されると、配置固有ページでは対応する翻訳文字列を使用して、ページをレンダリングする必要があります。これらの文字列の格納方法および取得方法は、『Oracle Application Server グローバリゼーション・サポート・ガイド』のロケール認識に関する章を参照してください。Java 開発に関する標準的なドキュメントも参照してください。次に 2 つのリンクを紹介します。

- Java Internationalization Guide:

  `http://java.sun.com/j2se/1.4.2/docs/guide/intl/index.html`

- Java ドキュメントの一般的なリンク：

  `http://java.sun.com/j2se/1.4.2/docs`

配置固有ページに関するガイドライン

配置固有ページを実装する場合は、次のガイドラインに従ってください。

- ログイン・ページとパスワードの変更ページは、SSL で保護することをお薦めします。
- ログイン・ページとパスワードの変更ページでは、クロスサイト・スクリプティング攻撃に備えてコーディングする必要があります。
- ログイン・ページとパスワードの変更ページでは、自動埋込みとキャッシュを `off` に設定する必要があります。これによって、ユーザーの資格証明がブラウザに保存されたり、キャッシュされる恐れはなくなります。`AutoComplete` タグの例を次に示します。

  ```
  <FORM NAME="foo" AutoComplete="off" METHOD="POST" ACTION="bar">
  ```

- オラクル社では、無認可のアクセスに対して警告するバナーを表示するよう、ログイン・ページを構成することをお薦めします。たとえば、次のような文章を使用できます。

  ```
  Unauthorized use of this site is prohibited and may subject you to civil and
  criminal prosecution.
  ```

- Single Sign-On Server をホスティングするコンピュータに、ログイン・ページおよびパスワード変更ページを配置します。これにより、これらのページの不正なバージョンを簡単に検出できます。

配置固有ページのインストール

`policy.properties` ファイルを使用して、配置固有のログイン・ページとパスワードの変更ページをインストールします。

policy.properties ファイルを使用したログイン・ページ、シングル・サインオフ・ページおよびパスワードの変更ページのインストール

ログイン・ページ、シングル・サインオフ・ページおよびパスワードの変更ページを構成することができます。

独自のログイン・ページ、シングル・サインオフ・ページおよびパスワードの変更ページをインストールするには、次のようにします。

1. *ORACLE_HOME*`/sso/conf/policy.properties` のパラメータを編集します。ログイン・ページ、シングル・サインオフ・ページおよびパスワードの変更ページへのパスを、次の例に示す値に置き換えます。

   ```
   #Deployment login page link
   loginPageUrl = /sso/pages/login.jsp
   logoutPageUrl = /sso/pages/logout.jsp

   #Deployment change password page link
   chgPasswordPageUrl = /sso/pages/password.jsp
   ```

2. Single Sign-On Server を再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

policy.properties ファイルを使用したワイヤレスのログイン・ページとパスワードの変更ページのインストール

OracleAS Wireless のフレームワークでは、配置固有のワイヤレスのログイン・ページとパスワードの変更ページを統合できます。これらのページのインストール手順は、標準ページのインストール手順と同様です（前述の項を参照）。

ワイヤレスのログイン・ページとパスワードの変更ページをインストールするには、次のようにします。

1. *ORACLE_HOME*/sso/conf/policy.properties を開きます。
2. 次のパラメータを編集または追加します。

```
#Wireless login page link
wirelessLoginPageUrl = wireless_login_page_url
wirelessChgPasswordPageUrl = change_password_page_URL
```

3. Single Sign-On Server を再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

policy.properties ファイルを使用した外部アプリケーション・ログイン・ページのインストール

サード・パーティのアプリケーションにログインするときに表示されるログイン・ページをユーザーが構成することができます。

サード・パーティのアプリケーションのログイン・ページを構成するには、次のようにします。

1. ログイン・ページが 12-3 ページの「ログイン・ページのパラメータ」および 12-8 ページの「ページのエラー・コード」に示すページ・パラメータおよびページ・エラー・コードを受け入れることを確認します。
2. ログイン・ページを構成したら、*ORACLE_HOME*/sso/conf/policy.properties ファイルの extAppLoginPageUrl パラメータを編集し、ログイン・ページへのパスを次の例に示すパスに置き換えます。

```
extAppLoginPageUrl = /sso/pages/ealogin.jsp
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

3. 必要に応じて、アプリケーション・ページを構成できます。
4. Single Sign-On Server を再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

配置固有ページの例

ipassample.jar ファイルには、login-ex.jsp、password-ex.jsp、signoff-ex.jsp の各ファイルが含まれます。これらのファイルをカスタマイズして、配置に対応させることもできます。これらのファイルの使用方法は、第 2 章の「サンプル・ファイルの取得」を参照してください。

カスタム・クラスの使用

一般に、カスタマイズした配置固有のページは、OC4J_SECURITY によって使用されている現在のバージョンのコンポーネント・クラスで動作する必要があります。カスタム・アプリケーションで、特定のクラスの別のバージョンを使用する必要がある場合は、そのクラスを OC4J_SECURITY インスタンスではなく、別の OC4J インスタンスに配置する必要があります。

たとえば、OC4J_SECURITY で使用されるバージョンと競合するカスタム log4j クラスを使用する必要のある配置では、そのカスタム・クラスを含むローカルの log4j jar ファイルを使用する別の OC4J_SECURITY インスタンスを起動します。

警告：OC4J_SECURITY で使用されるクラスをカスタム・バージョンに置き換えると、OracleAS Single Sign-On などの Oracle Application Server コンポーネントが使用できなくなる場合があります。

13

Oracle Identity Federation との統合

この章では、Oracle Application Server Single Sign-On と Oracle Identity Federation を使用して、フェデレーテッド認証を実装する方法について説明します。フェデレーテッド・シングル・サインオンを使用すると、ユーザーは 1 つのサイトで認証されるだけで、複数の企業の Web サイトの情報にアクセスできます。Oracle Application Server Single Sign-On または Oracle Identity Federation によって保護されているリソースにアクセスするユーザーの認証メカニズムとなるように、これらの製品を構成することができます。

この章の項目は次のとおりです。

注意： この章では、Oracle Application Server Single Sign-On 製品の観点から Oracle Identity Federation へのシングル・サインオンを構成する方法についてのみ説明します。構成を完了するには、Oracle Identity Federation 製品でも設定を変更する必要があります。詳細は、『Oracle Secure Federation Services Administration Guide』を参照してください。

フェデレーテッド・シングル・サインオンの動作方法

ユーザーは、様々な企業の Web サイトで提供される複数のコンテンツ間を簡単に移動できる必要があります。一方、企業の Web サイトは、異なるセキュリティ製品を使用する様々なドメインから入るユーザーを認証および認可する方法が必要です。Oracle Identity Federation 製品はこのような問題に対処します。

ユーザーがリモート Web サイト上の保護されたリソースにアクセスしようとすると、ユーザーのサイトにある Oracle Identity Federation 製品は、ユーザー・リクエストの認証に使用するためにユーザー情報をリモート・サイトに転送します。次に例を示します。

- 航空会社からユーザーは、航空機を製造する会社のドキュメント・データベースにある技術資料にアクセスできます。
- 携帯電話会社の顧客は、サード・パーティ・サプライヤに外部委託された請求書支払いアプリケーションにアクセスできます。
- 福利厚生プロバイダに接続されている社内 HR ポータルから、社員が 401（K）アプリケーションにアクセスできます。

ユーザーは、自社の Web サイト上のリンクをクリックすることで、パートナの Web サイトにあるコンテンツへのアクセスを要求する場合があります。ユーザーが最初にアクセスを要求すると、ホーム・サイトのユーザー・データ・リポジトリに格納されたユーザー・プロファイル情報を使用して、自社のサイトで認証されます。ユーザーのホーム（ID プロバイダ）ドメインは、ユーザーのアクセス・リクエストを宛先（サービス・プロバイダ）サイトに転送します。このとき、宛先サイトでユーザー・リクエストの認証に必要な資格証明も一緒に転送されます。

OracleAS Single Sign-On と Oracle Identity Federation を統合すると、次のことが可能になります。

- ユーザーが保護されたリソースにアクセスしようとすると、リソースのプロバイダは問合せを Oracle Identity Federation に送信し、Oracle Identity Federation はその問合せを OracleAS Single Sign-On に転送できます。

  OracleAS Single Sign-On で認証されると、対象のリソースにアクセスできます。
- ユーザーが OracleAS Single Sign-On によって保護されたリソースにアクセスしようとすると、リクエストを Oracle Identity Federation に転送し、Oracle Identity Federation は認証を実行する適切な ID プロバイダを見つけることができます。

  ID プロバイダで認証されると、OracleAS Single Sign-On が保護するリソースにアクセスできます。

Oracle Identity Federation の詳細は、Oracle Technology Network の「Oracle Documentation」ページからアクセスできる『Oracle Secure Federation Services Administration Guide』を参照してください。URL は次のとおりです。

```
http://www.oracle.com/technology/documentation
```

ユーザーから見たフェデレーテッド・シングル・サインオン

この章の説明に従ってフェデレーテッド・シングル・サインオンを構成すると、サービス・プロバイダと ID プロバイダの両方に対して一度で認証を実行できます。

このようにして一括認証された後は、ユーザーは ID プロバイダにのみ資格証明を渡します。認証後、ユーザーはサービス・プロバイダにある保護リソースにアクセスできます。

サービス・プロバイダとしての Oracle Stack の構成

OracleAS Single Sign-On および Oracle Identity Federation がサービス・プロバイダの役割を果すとき、OracleAS Single Sign-On はユーザー認証を Oracle Identity Federation に委譲します。この場合、OracleAS Single Sign-On で保護されたリソースにユーザーがアクセスしようとするときに、Oracle Identity Federation が ID プロバイダを識別する中間層となるようにフェデレーテッド・シングル・サインオンを構成します。

デフォルトでは、OracleAS Single Sign-On と Oracle Identity Federation の間のシングル・サインオンに `MediumHighSecurity` 認証レベルが使用されます。この認証レベルを変更する場合は、OracleAS Single Sign-On のデフォルトの認証レベルか、それより上のレベルに設定することをお薦めします。低いレベルを使用すると、高いセキュリティ・レベルで保護されたアプリケーションにアクセスしようとしたときに、認証が必要になります。

次に示す作業概要はこの構成の手順をまとめたものです。詳細な手順は、作業概要の後に説明します。

注意： `policy.properties` ファイルを変更する前に OracleAS Single Sign-On Server を停止し、変更後に再起動する必要があります。

作業概要 : Oracle Identity Federation インスタンスへの認証の委譲

1. OracleAS Single Sign-On Server を停止します。
2. `policy.properties` ファイルで Oracle Identity Federation を認証メカニズムとして構成します。
3. Oracle Identity Federation で保護するアプリケーションを、`policy.properties` ファイルの保護アプリケーション・リストに追加します。
4. Oracle Application Server Single Sign-On Server を再起動します。
5. `policy.properties` ファイルに追加されたアプリケーションにアクセスしようとするユーザーを認証するように、Oracle Identity Federation を構成します。

   詳細は、『Oracle Secure Federation Services Administration Guide』を参照してください。

Oracle Application Server Single Sign-On Server を停止する手順

1. Oracle Enterprise Manager 10*g* Application Server Control コンソールから、停止するアプリケーション・サーバーのインスタンスをクリックします。
2. アプリケーション・サーバーの詳細ページから、OC4J_SECURITY を選択します。

   Oracle HTTP Server も停止する場合は、アプリケーション・サーバーの詳細ページで「HTTP サーバー」リンクをクリックします。
3. **「停止」**をクリックします。

   確認ページが表示されます。
4. 確認ページで**「はい」**をクリックします。

Oracle Identity Federation インスタンスに認証を委譲する手順

1. 次のファイルをテキスト・エディタで開きます。

   `OSSO_install_dir``/sso/conf/policy.properties`

   OSSO_install_dir は Oracle Application Server Single Sign-On がインストールされているディレクトリです。
2. 次の行のコメントを解除し、編集します。

   SASSOAuthnUrl: この行のコメントを解除し、Oracle Identity Federation のログイン URL を示すようにホスト名とポートを変更します。

SASSOLogoutUrl: この行のコメントを解除し、Oracle Identity Federation のログアウト URL を示すようにホスト名とポートを変更します。

コロン（:）は、バックスラッシュ（\）でエスケープする必要があります。次に例を示します。

```
SASSOAuthnUrl = http\://osfs_host.domain\:port/sso/authn

SASSOLogoutUrl =
http\://osfs_host.domain\:port/sso/jsp/sasso_logout_success.jsp
```

3. 次の行のコメントを解除し、Oracle Identity Federation のセキュリティ・レベルを設定します。

```
SASSOAuthLevel = MediumHighSecurity
```

4. policy.properties ファイルで、MediumHighSecurity 認証レベルのプラグインおよび監査レベルのコメントを解除します。

```
# MediumHighSecurity_AuthPlugin =
oracle.security.sso.server.auth.SASSOAuth
```

5. Oracle Identity Federation をホスティングしているサーバーのインストール・ディレクトリからキーストア・ファイル（ファイル名は keystore）を見つけます。

```
Oracle_Identity_Federation_installdir/sso/conf
```

キーストアを、policy.properties ファイルの SASSOConfigFile パラメータで指定した場所にコピーします。この場所は、OracleAS Single Sign-On Server のローカル・ホーム・ディレクトリからの相対パスです。
例:

```
SASSOConfigFile = /sso/conf/keystore
```

キーストアの生成の詳細は、『Oracle Secure Federation Services Administration Guide』を参照してください。

6. Oracle Enterprise Manager 10*g* Application Server Control コンソールから、変更する Oracle Enterprise Manager 10*g* のインスタンスをクリックします。

7. Oracle HTTP Server および OC4J_SECURITY を再起動します。

Oracle Identity Federation ポリシーによって保護されたアプリケーションを追加する手順

1. 13-3 ページの「Oracle Application Server Single Sign-On Server を停止する手順」の手順に従って、サーバーを停止します。

2. 次の場所にある policy.properties ファイルを編集します。

```
install_dir/sso/conf/policy.properties
```

install_dir は Oracle Application Server Single Sign-On がインストールされているディレクトリです。

3. policy.properties ファイルの Protected URL セクションで、保護する 1 つ以上のアプリケーションのホストとポートを設定します。たとえば、次のように設定します。

```
host\:port = MediumHighSecurity
```

host\:port は保護されるアプリケーションのホストとポートです。ホストおよびポートは、インストール中またはインストール後に構成されます。詳細は、中間層のドキュメントを参照してください。*MediumHighSecurity* は Oracle Identity Federation でシングル・サインオン用に構成されたセキュリティ・レベルです。（認証レベルの詳細は、第 6 章「マルチレベル認証」を参照してください。）

4. Oracle Enterprise Manager 10*g* Application Server Control コンソールから、起動するアプリケーション・サーバーのインスタンスをクリックします。

5. Oracle HTTP Server および OC4J_SECURITY を再起動します。

注意： 構成を完了するには、Oracle Identity Federation 製品でも設定を変更する必要があります。詳細は、『Oracle Secure Federation Services Administration Guide』を参照してください。

ID プロバイダとしての Oracle Stack の構成

OracleAS Single Sign-On および Oracle Identity Federation が ID プロバイダの役割を果すとき、Oracle Identity Federation はユーザー認証を OracleAS Single Sign-On に委譲します。この場合、リソースへのユーザー・リクエストを Oracle Identity Federation が OracleAS Single Sign-On に転送するようにフェデレーテッド・シングル・サインオンを構成します。このとき、OracleAS Single Sign-On が認証メカニズムとなります。

デフォルトでは、OracleAS Single Sign-On と Oracle Identity Federation の間のシングル・サインオンに `MediumHighSecurity` 認証レベルが使用されます。（詳細は、第 6 章「マルチレベル認証」を参照してください。）この認証レベルを変更する場合は、OracleAS Single Sign-On のデフォルトの認証レベルか、それより上のレベルに設定することをお薦めします。低いレベルを使用すると、高いセキュリティ・レベルで保護されたアプリケーションにアクセスしようとしたときに、認証が必要になります。

注意： 構成を完了するには、Oracle Identity Federation 製品でも設定を変更する必要があります。詳細は、『Oracle Secure Federation Services Administration Guide』を参照してください。

認証メカニズムとして OracleAS Single Sign-On を使用してフェデレーテッド認証を構成する手順

1. Oracle Enterprise Manager 10*g* Application Server Control コンソールへ移動します。
2. 停止するアプリケーション・サーバーのインスタンスをクリックします。
3. OC4J_SECURITY を停止するために、アプリケーション・サーバーの詳細ページから OC4J_SECURITY を選択します。
4. Oracle HTTP Server を停止するために、アプリケーション・サーバーの詳細ページで「HTTP サーバー」リンクをクリックします。
5. **「停止」**をクリックします。

   確認ページが表示されます。
6. 確認ページで**「はい」**をクリックします。
7. 次のファイルをテキスト・エディタで開きます。

   `OSSO_install_dir``/sso/conf/policy.properties`

   OSSO_install_dir は Oracle Application Server Single Sign-On がインストールされているディレクトリです。
8. 次の行のコメントを解除し、編集します。

   SASSOAuthnUrl: この行のコメントを解除し、Oracle Identity Federation のログイン URL を示すようにホスト名とポートを変更します。

   SASSOLogoutUrl: この行のコメントを解除し、Oracle Identity Federation のログアウト URL を示すようにホスト名とポートを変更します。

   コロン（:）は、バックスラッシュ（\）でエスケープする必要があります。次に例を示します。

   ```
   SASSOAuthnUrl = http\://osfs_host.domain\:port/sso/authn

   SASSOLogoutUrl =
   http\://osfs_host.domain\:port/sso/jsp/sasso_logout_success.jsp
   ```

9. Oracle Identity Federation をホスティングしているサーバーのインストール・ディレクトリからキーストア・ファイル（ファイル名は keystore）を見つけます。

   Oracle_Identity_Federation_install_dir/sso/conf

   キーストアを、policy.properties ファイルの SASSOConfigFile パラメータで指定した場所にコピーします。この場所は、OracleAS Single Sign-On Server のローカル・ホーム・ディレクトリからの相対パスです。
   例：

   ```
   SASSOConfigFile = /sso/conf/keystore
   ```

   キーストアの生成の詳細は、『Oracle Secure Federation Services Administration Guide』を参照してください。

10. OC4J_SECURITY および Oracle HTTP Server を再起動します。

Web Portal へのフェデレーテッド認証 URL の追加

Web Portal ページで、それぞれが異なる認証メカニズムを必要とする様々なリソースへのリンクを構成することができます。Oracle Identity Federation と OracleAS Single Sign-On の統合により、OracleAS Single Sign-On で保護された Web ページ上のリンクを次のように構成することが可能です。

- ユーザーがリンクをクリックしたときに Oracle Identity Federation のインスタンスを見つけるように OracleAS Single Sign-On に要求します。
- 特定の ID プロバイダからの認証を要求するように Oracle Identity Federation に指示します。

> **注意：** ID プロバイダの構成の詳細は、『Oracle Secure Federation Services Administration Guide』を参照してください。

Web Portal ページにフェデレーテッド認証リンクを構成する手順：

1. OracleAS Single Sign-On で保護されるようにリソースを設定します。
2. Portal ページの HTML コードに、次のリンクを指定します。

   ```
   <a href="http(s)://<rest-of-URL>?providerid=xxx">
   ```

   変数の説明

   - *http(s)* は使用するプロトコルです（http または https）。
   - *<rest-of-URL>* は保護されるリソースへのパスへの URL です。
   - providerid は、Oracle Identity Federation が ID プロバイダのために問い合せる必要がある OracleAS Single Sign-On に知らせるキーワードです。
   - *xxx* は、Oracle Identity Federation で構成された ID プロバイダ ID です。

14

サード・パーティのアクセス管理システムとの統合

この章では、OracleAS Single Sign-On とサード・パーティのアクセス管理製品を統合する方法について説明します。サード・パーティとの統合がどのように機能するかを説明し、統合 API を紹介します。この章の最後では、OracleAS Single Sign-On とサード・パーティのアクセス管理システムを統合する例を示します。

サード・パーティのシステムを設置している企業では、OracleAS Single Sign-On Server をサード・パーティのシステムと Oracle アプリケーション間の認証ゲートウェイとして動作させる認証アダプタを作成することによって、OracleAS スイートを利用できるようになります。

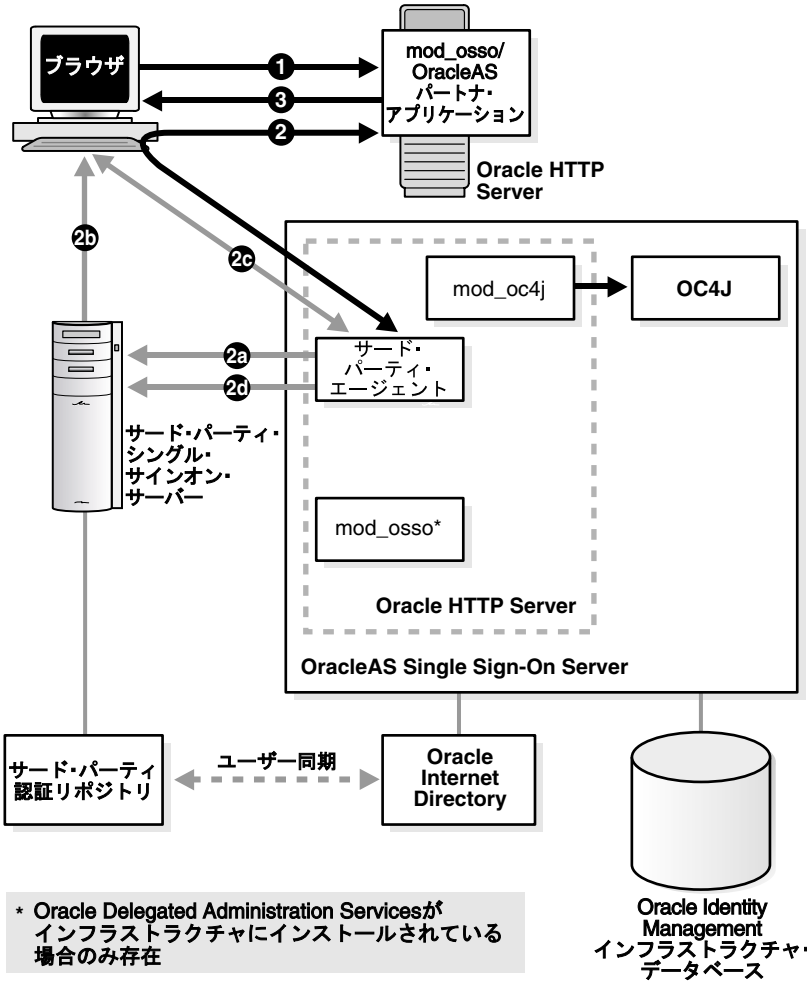
この章の項目は次のとおりです。

サード・パーティのアクセス管理の仕組み

サード・パーティのアクセス管理では、OracleAS Single Sign-On Server、サード・パーティのアクセス管理サーバーおよびパートナ・アプリケーションによって信頼の連鎖が形成されます。OracleAS Single Sign-On Server は、サード・パーティのアクセス管理サーバーに認証を委譲するため、それ自体が Single Sign-On 対応のアプリケーションになります。Oracle アプリケーションは OracleAS Single Sign-On Server とのみ連携し、サード・パーティのアクセス管理サーバーは認識しません。ただし、暗黙的にはサード・パーティ製のサーバーを信頼します。

OracleAS Single Sign-On がこのような配置の下でユーザーに認証トークンを発行できるようにするには、サード・パーティのアクセス管理サーバーが HTTP ヘッダーを設定するか、他のメカニズムを使用して、OracleAS Single Sign-On Server にユーザーの ID を渡す必要があります。ユーザーの ID を取得すると、OracleAS Single Sign-On Server はこれまでのように機能し、ユーザーを認証してパートナ・アプリケーションにリダイレクトします。14-2 ページの図 14-1 にそのプロセスを示します。

図 14-1 サード・パーティのサーバーを使用した Oracle パートナ・アプリケーションへのアクセス



この図から 2 つの使用例が考えられます。

使用例 1: ユーザーが、サード・パーティのサーバーに認証されていない場合

1. 認証されていないユーザーが、OracleAS Single Sign-On によって保護されているアプリケーションへのアクセスを試みます。
2. アプリケーションは、ユーザーを認証するために OracleAS Single Sign-On Server にユーザーをリダイレクトします。このリダイレクションの一環として、次の操作が行われます。
   a. OracleAS Single Sign-On Server は、ユーザーの認証をサード・パーティ・エージェント（通常は Oracle HTTP Server のモジュール）に任せます。このモジュールは、サード・パーティ・アダプタのインストール時にインストールします。このモジュールは、Single Sign-On Server にのみ配置することに注意してください。アプリケーション・サーバーには配置しません。

   注意： サード・パーティ・エージェントの詳細は、「統合事例 : SSOAcme」を参照してください。

   b. サード・パーティのサーバーは、ユーザーのブラウザにトークンを設定します。
   c. OracleAS Single Sign-On Server は、ブラウザからトークンを取得します。
   d. OracleAS Single Sign-On Server は、サード・パーティのサーバーでこのトークンを検証します。

   OracleAS Single Sign-On Server はトークンを検証した後、ユーザー（および認証されたユーザー情報）をリクエストされたアプリケーションに返します。
3. このアプリケーションによって、ユーザーの必要とするコンテンツが表示されます。

使用例 2: ユーザーが、サード・パーティのサーバーに認証されている場合

1. 認証されているユーザーが、OracleAS Single Sign-On によって保護されているアプリケーションへのアクセスを試みます。
2. アプリケーションは、ユーザーを認証するために OracleAS Single Sign-On Server にユーザーをリダイレクトします。このリダイレクションの一環として、次の操作が行われます。
   a. OracleAS Single Sign-On Server は、ブラウザからトークンを取得します（使用例 1 の手順 2c）。
   b. OracleAS Single Sign-On Server は、サード・パーティのサーバーでこのトークンを検証します（使用例 1 の手順 2d）。

   OracleAS Single Sign-On Server はトークンを検証した後、ユーザーをリクエストされたアプリケーションに返します。
3. このアプリケーションによって、ユーザーの必要とするコンテンツが表示されます。

   注意： 関連するシングル・サインオン・システムにすべての認可ユーザーがアクセスできるようにするには、ユーザー・リポジトリを 1 箇所で集中管理する必要があります。つまり、配置の前に、Oracle Internet Directory と外部リポジトリ間でユーザーを同期化する必要があります（このリポジトリが Oracle Internet Directory とは異なる場合）。

サード・パーティ・リポジトリと Oracle Internet Directory の同期化

前の手順で説明した認証プロセスでは、ユーザーのリポジトリが Oracle Internet Directory であるか、またはリポジトリがサード・パーティのディレクトリかデータベースであることを前提としています。リポジトリがサード・パーティのディレクトリかデータベースである場合には、ユーザー名情報を Oracle Internet Directory 内のユーザー・エントリと同期化する必要があります。同期化することにより、Single Sign-On Server は、シングル・サインオンが有効になったアプリケーションで要求されたユーザー属性を取得できます。

注意： 同期メカニズムが搭載されていない場合、サード・パーティのアクセス管理を統合できません。

サード・パーティのリポジトリと Oracle Internet Directory を同期化するには、Oracle Directory Integration and Provisioning または一括ロード・ツールのいずれかを使用します。詳細は、『Oracle Identity Management 統合ガイド』のディレクトリの同期化に関する章を参照してください。

サード・パーティ統合モジュール

サード・パーティとの統合を実現するには、2 通りの方法があります。ベンダーから提供された既存のパッケージを使用する方法と、Oracle から提供されたインタフェースを使用して、サード・パーティのアダプタを独自に構築する方法です。

ベンダーから提供されたパッケージを使用する場合

サード・パーティのアクセス管理ベンダーによっては、OracleAS Single Sign-On Server 用の認証アダプタを提供しているところがあります。これらの製品を使用すると、独自のコードを記述しなくても、サード・パーティのシステムを Oracle のシステムに統合できます。次のリンク先に、これらのベンダー製品に関する情報があります。リストされているすべてのベンダーの製品は、OracleAS Single Sign-On と組み合せて動作することが保証されています。ページの下の部分にある見出し「Documentation」の「Single Sign-On」を参照してください。これらの資料は、「Services」メニューには含まれないので注意してください。

```
http://www.oracle.com/technology/products/id_mgmt/partners/index.html
```

サード・パーティ・アダプタを使用する場合、アダプタを実装する手順の詳細は、このドキュメントではなくベンダーのドキュメントにあります。アダプタの実装についてサポートを必要とする場合は、担当のベンダーに直接連絡してください。連絡情報は、前述のリンク先に含まれます。

独自のパッケージを構築する場合

Java ツールキット oracle.security.sso.ias904.toolkit を使用して、サード・パーティとの統合用アダプタを独自に構築することができます。このツールキットには、認証の実行および配置固有の Cookie の設定に使用する 2 つのインタフェースが含まれています。OracleAS Single Sign-On Server では、認証時に前者のインタフェースが使用されます。Cookie 用のアダプタである後者のインタフェースは、ユーザー ID の確認に成功した後に使用されます。

この項では、oracle.security.sso.ias904.toolkit のインタフェースと関連するクラスについて説明します。この項の項目は次のとおりです。

- インタフェースの使用に関するガイドライン
- クラスとインタフェース
- 構成手順

インタフェースの使用に関するガイドライン

サード・パーティとの統合用アダプタを独自に構築する場合は、認証用のインタフェースを使用します。OracleAS Single Sign-On Server は、このインタフェースを実装することで、ユーザー ID を認証します。

Cookie インタフェースはオプションです。専用の Cookie が必要な環境またはアプリケーション用に提供されています。たとえば、一般的なユーザー・アプリケーションで設定された認証 Cookie をレプリケートする場合があります。または、Cookie を使用してユーザー・プリファレンスを設定することもあります。OracleAS Single Sign-On Server でユーザーが認証されたら、1 つ以上の Cookie を設定できます。Cookie インタフェースを使用する場合は、独自のアダプタにも、ベンダーから提供されたアダプタにも追加できます。

クラスとインタフェース

キット内のクラスとインタフェースは次の機能を実行します。

- トークンを使用した認証
- ユーザー情報コンストラクタ
- 外部 Cookie の設定
- 例外処理

トークンを使用した認証 IPASAuthInterface.java パッケージは、OracleAS Single Sign-On Server によって認証時に起動されます。トークンを使用した認証をサポートする場合、このインタフェースの実装者は、安全に設定された HTTP ヘッダーや安全な Cookie などから、安全な方法でユーザーの ID を取得して、ユーザー名を OracleAS Single Sign-On Server に戻す必要があります。次にインタフェースを示します。

IPASAuthInterface.java インタフェース

```
package oracle.security.sso.ias904.toolkit;

/**
 * Oracle Single Sign-On server authentication interface. This package can
 * be used to integrate with custom authentication mechanism or third-party
 * access management system.
 */
public interface IPASAuthInterface
{
  /**
   * This method returns IPASUserInfo object that contains either user name,
   * subscriber name, and requested URL, or full user and subscriber
   * attribute mappings, including DN, GUID, and requested URL.
   * @param  request - HTTP request object
   * @return IPASUserInfo object that contains
   * @authenticated user information
```

```
   * @see oracle.security.sso.ias904.toolkit.IPASUserInfo
   */
   IPASUserInfo authenticate(HttpServletRequest request)
       throws IPASAuthException, IPASInsufficientCredException;

  /**
   * This method returns a page URL - user will be redirected to
   * the page to enter login credentials
   * @param request - HTTP request object
   * @param message - Message to be displayed in the page
   * @return The page URL for collecting user login credential
   */
   java.net.URL getUserCredentialPage(HttpServletRequest request,
       String message);
}
```

ユーザー情報コンストラクタ ユーザー情報クラス IPASUserInfo.java は、ユーザー・ログインのコンストラクタを提供します。

IPASUserInfo.java クラス

```
package oracle.security.sso.ias904.toolkit;
/**
 * User information class
 */
public class IPASUserInfo
{
  /**
   * Constructor with user login name that belongs to default subscriber
   * @param userNickName - User login name
   */
   IPASUserInfo(String userNickName);


  /**
   * Constructor with user login name that belongs to default subscriber
   * @param userNickName - User login name
   * @param subscriberName - The user's subscriber name
   */
   IPASUserInfo(String userNickName, String subscriberName);


  /**
   * Constructor with user login name that belongs to default subscriber
   * @param userNickName - User login name
   * @param userDN - User directory distinguished name
   * @param userGUID - User directory GUID value
   * @param subscriberName - The subscriber name for this user
   * @param subscriberDN - The subscriber's directory distinguished name
   * for this user
   * @param subscriberGUID - The subscriber's directory GUID value
   * for this user
   */
   IPASUserInfo(String userNickName, String userDN, String userGUID,
       String subscriberName, String subscriberDN, String subscriberGUID);

  /**
   * This method returns subscriber distinguished name
   * @return subscriber distinguished name
   */
   String getSubscriberDN();
```

```
/**
 * This method sets subscriber distinguished name
 * @param subDn - subscriber distinguished name
 */
 void setSubscriberDN(String subDn);

/**
 * This method returns subscriber GUID value
 * @return subscriber GUID value
 */
 String getSubscriberGUID();

/**
 * This method sets subscriber GUID value
 * @param subGuid - subscriber GUID value
 */
 void setSubscriberGUID(String subGuid);

/**
 * This method returns subscriber name
 * @return subscriber name
 */
 String getSubscriberName();

/**
 * This method sets subscriber name
 * @param subscriber name
 */
 void setSubscriberName(String subName);

/**
 * This method returns user login name
 * @return user login name
 */
 String getUserName();

/**
 * This method sets user login name
 * @param user login name
 */
 void setUserName(String userName);

/**
 * This method returns user distinguished name
 * @return user distinguished name
 */
 String getUserDN();

/**
 * This method sets user distinguished name
 * @param user distinguished name
 */
 void setUserDN(String userDN);

/**
 * This method returns user GUID value
 * @return user GUID value
 */
 String getUserGUID();
```

```
  /**
   * This method sets user GUID value
   * @param user GUID value
   */
   void setUserGUID(String userGUID);
}
```

外部 Cookie の設定 サード・パーティのアプリケーションから Cookie を使用するために、OracleAS Single Sign-On Server に対して認証された後、ユーザーは IPASCustomCookieInterface.java インタフェースを使用して、1 つ以上の追加の Cookie を設定できます。たとえば、OracleAS Single Sign-On Server が管理する一連の Cookie に、別のシングル・サインオン・ベンダーによって設定された Cookie を追加できます。認証に成功し、Cookie アダプタが適切な認証レベルに対応する場合に、これらの Cookie が設定されます。

サード・パーティのアプリケーションによって設定された Cookie を実装するには、Cookie を定義する Java クラスを記述する必要があります。

IPASCustomCookieInterface.java インタフェース

```
package oracle.security.sso.ias904.toolkit;

/**
 * Oracle Single Sign-On server invokes this method to obtain
 * user-defined custom cookies which will be sent to the user
 * upon successful login.
 */
public interface IPASCustomCookieInterface
{
 /**
   * Oracle Single Sign-On server invokes this method to obtain
   * user-defined custom cookies which will be sent
   * to the user upon successful login.
   * @param user - IPASUserInfo object that contains
   * the authenticated user information
   * @param request -  HTTP request object
   * @return Array of javax.servlet.http.Cookie objects
   * @see javax.servlet.http.Cookie
   */
   public javax.servlet.http.Cookie[] getCustomCookie(IPASUserInfo user,
       HttpServletRequest request);
}
```

例外処理 この項では、oracle.security.sso.ias904.toolkit の例外処理クラスを示します。

IPASException.java クラス

```
package oracle.security.sso.ias904.toolkit;

/**
 * Generic exception class
 */
public class IPASException
   extends Exception
{
  /**
   * Default constructor with no error message
   */
   public IPASException()
   {
       super();
   }
```

```
  /**
   * Constructor with an error message
   */
   public IPASException(String message)
   {
       super(message);
   }
}
```

IPASAuthException.java クラス

```
package oracle.security.sso.ias904.toolkit;

/**
 * Authentication failure exception class
 */
public class IPASAuthException
   extends IPASException
{
  /**
   * Default constructor with no error message
   */
   public IPASAuthException()
   {
       super();
   }
  /**
   * Constructor with an error message
   */
   public IPASAuthException(String message)
   {
       super(message);
   }
}
```

IPASInsufficientCredException.java クラス

```
package oracle.security.sso.ias904.toolkit;

/**
 * Authentication failure exception due to insufficient user credentials
 */
public class IPASInsufficientCredException
   extends IPASException
{
  /**
   * Constructor with an error message
   */
   public IPASInsufficientCredException (String message)
   {
       super(message);
   }
}
```

構成手順

インタフェースを使用してサード・パーティとの統合用アダプタを独自に作成するには、次の手順を実行します。

1. サード・パーティのアクセス管理システムで、この URI を保護するルールを作成します。

   ```
   /sso/auth/*
   ```

2. 次のログアウト URI を、サード・パーティのアクセス管理システムに登録します。

   ```
   /sso/logout
   ```

   注意： この後の手順およびサンプル・パッケージでは、acme と SSOAcme の部分をすべて実際の会社名に置き換えてください。

3. パッケージ用の Java ファイルを作成します。詳細は、「統合事例 : SSOAcme」のサンプル・ファイルを参照してください。サンプル・ファイルは SSOAcmeAuth.java と呼ばれます。コンパイルの前に、次のパッケージ・ディレクティブをこのファイルに追加する必要があります。

   ```
   package acme.security.ssoplugin;
   ```

4. クラス・パス内の *ORACLE_HOME*/sso/lib/ipastoolkit.jar とともに、このファイルをコンパイルします。サンプル・ファイル SSOAcmeAuth.java は、次のようにコンパイルします。

   ```
   ORACLE_HOME/jdk/bin/javac -classpath ORACLE_HOME/sso/lib/ipastoolkit.jar:
   ORACLE_HOME/lib/servlet.jar -d ORACLE_HOME/sso/plugin SSOAcmeAuth.java
   ```

   このコマンドにより、SSOAcmeAuth.class が作成され、ディレクトリ *ORACLE_HOME*/sso/plugin/acme/security/ssoplugin に配置されます。

5. policy.properties ファイルで、簡易認証プラグインを手順 3 で作成したプラグインに置き換えます。簡易認証プラグインは、次のように記述されています。

   ```
   MediumSecurity_AuthPlugin = oracle.security.sso.server.auth.SSOServerAuth
   ```

   サンプルのプラグインは、次のようになります。

   ```
   MediumSecurity_AuthPlugin = acme.security.ssoplugin.SSOAcmeAuth
   ```

   注意： policy.properties を編集するときは、行の末尾に空白を入れないでください。

6. カスタム Cookie インタフェースを実装する場合は、各 Cookie の実装のクラス名、およびユーザーの認証が成功した場合にカスタム Cookie が設定される最低の認証レベルを policy.properties に追加します。

   ```
   # Custom Cookie Provider Class names
   # ----------------------------------
   # Sample custom cookie tester provider class

   CustomCookie_ProviderPlugin = class_name
   CustomCookieAuthLevel = authentication_level

   CustomCookie_ProviderPlugin1 = class_name1
   CustomCookieAuthLevel = authentication_level1

   CustomCookie_ProviderPlugin2 = class_name2
   CustomCookieAuthLevel = authentication_level2
   ```

   class_name は Cookie を実装する Java クラスの名前、*authentication_level* は NoSecurity, LowSecurity、LowMediumSecurity、MediumSecurity、MediumHighSecurity または HighSecurity です。

たとえば、3 つのプラグインを実装できます。1 つ目の CustomCookieProviderPlugin を test.custom.MyCookieProvider クラスに関連付け、2 つ目の CustomCookieProviderPlugin1 を com.custom.CustomCookieProvider クラスに関連付けることができます。

マルチレベル認証を使用せず、認証アダプタ・レベルのデフォルト設定を使用する場合は、この値を MediumSecurity に設定できます。

7. シングル・サインオン中間層を再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

8. 統合したシステムをテストします。

統合システムからのログアウト

この項に進む前に、シングル・サインオフの概念とその実装方法について確認する必要があります。第 1 章の「シングル・サインオフ」の項を参照してください。

サード・パーティ製品のログアウトには 2 つの方法があります。

- ユーザーはサード・パーティのアクセス管理システムを使用して、ログアウト・リクエストを開始します。

  この場合、ユーザーは、サード・パーティ・システムのログアウト・ハンドラを起動するログアウト・リンクをクリックします。サード・パーティのログアウト・フローにより、独自のセッションがクリーンアップされます。クリーンアップ後、サード・パーティ・システムは OracleAS Single Sign-On のログアウト・ハンドラを起動する必要があります。OracleAS Single Sign-On のログアウト・ハンドラを起動すると、OracleAS Single Sign-On Server で保護されるすべてのアプリケーションからユーザーがログアウトされます。シングル・サインオンのログアウトを実行するには、サード・パーティ・システムはユーザーを次の URL にリダイレクトする必要があります。

  `http://single_sign-on_host:single_sign-on_port/sso/logout`

  または、SSL が有効な場合、次の URL にリダイレクトします。

  `https://single_sign-on_host:single_sign-on_ssl_port/sso/logout`

  done_url は、ログアウト後にユーザーがリダイレクトされる URL です。

- ユーザーは OracleAS Single Sign-On システムを使用して、ログアウト・リクエストを開始します。

  この使用例では、Oracle パートナ・アプリケーションでログアウト・リンクをクリックします。これにより、OracleAS Single Sign-On のログアウト・ハンドラが起動されます。ログアウトが終了したら、ユーザーはサード・パーティ・システムからもログアウトする必要があります。サード・パーティ・システムに Oracle ログアウト・ハンドラ（前述の URL 内の `ls_logout`）を登録すると、同時ログアウトを実行できます。サード・パーティ・システムで Oracle ログアウト・ハンドラの起動が検出されると、サード・パーティのセッションがクリーンアップされます。

  サード・パーティのアダプタを使用している場合に、ログアウト処理を実際に実装する方法は、各ベンダーのドキュメントを参照してください。

Windows のネイティブ認証との統合

OracleAS Single Sign-On は Windows のネイティブ認証（WNA）と相互運用できます。詳細は、『Oracle Identity Management 統合ガイド』の Active Directory との統合に関する項を参照してください。

WNA を仮想ホストで使用する場合は、1 つのホスト名のみ構成できます。仮想ホストを使用するには、`/etc/hosts` ファイルで仮想ホストのサーバー IP アドレスを定義する必要があります。

注意： WNA 対応のシングル・サインオン・インストール環境のサーバー 1 台に対して、1 つの仮想ホストのみ配置できます。`/etc/hosts` ファイルで複数の仮想ホストを構成した場合でも、サーバー IP アドレスのリストにある最初の 1 つのみが WNA 対応のシングル・サインオン用に動作します。

統合事例 : SSOAcme

SSOAcme の事例を考えます。SSOAcme は、保護されたリソースに対するシングル・サインオン認証を提供する製品です。SSOAcme は、SSOAcme ポリシー・サーバーと SSOAcme エージェントの 2 つのコンポーネントで構成されています。SSOAcme ポリシー・サーバーはユーザー管理、セッション管理、認証、認可などの様々なサービスをユーザーに提供します。SSOAcme エージェントは Web サーバーおよび Web アプリケーション・サーバーに配置します。リソースのリクエストをチェックして、リソースが SSOAcme によって保護されているかどうかを確認します。

SSOAcme をすでにインストールしている顧客は、SSOAcme を使用して OracleAS アプリケーションにアクセスする場合があります。このようなアクセスを実現するには、SSOAcme が OracleAS Single Sign-On を経由して Oracle アプリケーションとデータをやり取りできるようにする API を使用します。

注意： SSOAcme は架空の製品です。ここでは、説明の目的でのみ使用しています。

この項の項目は次のとおりです。

サンプル統合パッケージ

ここに示す SSOAcme.java パッケージは、SSOAcme の既存の実装と OracleAS Single Sign-On との統合に使用できます。

```
/* Copyright (c) 2002, 2003, Oracle Corporation.  All rights reserved.  */

/*
   DESCRIPTION
     Sample class for SSOAcme integration with SSO Server

   PRIVATE CLASSES

   NOTES
     This class implements the SSOServerAuthInterface.
     To enable this integration, replace:
         oracle.security.sso.server.auth.SSOServerAuth
     with
         acme.security.ssoplugin.SSOAcmeAuth
     for the desired security level in policy.properties.
 */

import java.io.PrintWriter;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import oracle.security.sso.ias904.toolkit.IPASAuthInterface;
import oracle.security.sso.ias904.toolkit.IPASAuthException;
import oracle.security.sso.ias904.toolkit.IPASUserInfo;
import oracle.security.sso.ias904.toolkit.IPASInsufficientCredException;

public class SSOAcmeAuth implements IPASAuthInterface {

    private static String CLASS_NAME = "SSOAcmeAuth";
    private static String ACME_USER_HEADER = "ACME_USER";

    public SSOAcmeAuth() {

    }

    public IPASUserInfo authenticate(HttpServletRequest request)
      throws IPASAuthException, IPASInsufficientCredException {

      String AcmeUserName = null;

      try
       {
         AcmeUserName = request.getHeader(ACME_USER_HEADER);
       }
       catch (Exception e)
       {
         throw new IPASInsufficientCredException("No Acme Header");
       }

       if (AcmeUserName == null)
           throw new IPASInsufficientCredException("No Acme Header");

       IPASUserInfo authUser = new IPASUserInfo(AcmeUserName);

       return authUser;

    }
```

```
    public java.net.URL getUserCredentialPage(HttpServletRequest request,
        String msg) {

        // This function will never have been reached in the case of SSOAcme
        // because the SSOAcme Agent will intercept all requests
        return "http://error_url";

    }

}
```

リリース 9.0.2 のサンプル実装からリリース 10.1.3 への移行

リリース 9.0.2 の外部認証パッケージを使用してサード・パーティ製品での認証を実行していたユーザーは、この項を参照してください。リリース 9.0.2 のパッケージは PL/SQL で記述されていました。リリース 10.1.3 のパッケージは Java で記述されています。以降では、2 つのパッケージの関連するセクションを一緒に示します。

新しい認証インタフェース

リリース 10.1.3:

```
package acme.security.ssoplugin;

import java.io.PrintWriter;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import oracle.security.sso.server.util.SSODebug;
import oracle.security.sso.ias904.toolkit.IPASAuthInterface;
import oracle.security.sso.ias904.toolkit.IPASAuthException;
import oracle.security.sso.ias904.toolkit.IPASUserInfo;
import oracle.security.sso.ias904.toolkit.IPASInsufficientCredException;

public class SSOAcmeAuth implements IPASAuthInterface {

    private static String CLASS_NAME = "SSOAcmeAuth";
    private static String ACME_USER_HEADER = "ACME_USER";

public SSOAcmeAuth() {
}

public IPASUserInfo authenticate(HttpServletRequest request)
throws IPASAuthException, IPASInsufficientCredException {
```

リリース 9.0.2:

```
FUNCTION authenticate_user
  (
   p_user OUT VARCHAR2
  )
  return PLS_INTEGER
IS
 l_http_header varchar(1000);
 l_ssouser wwsec_person.user_name%type := NULL;
BEGIN
```

HTTP ヘッダーからのユーザー名の取得

リリース 10.1.3:

```
String AcmeUserName = null;

try
 {
  AcmeUserName = request.getHeader(ACME_USER_HEADER)
```

リリース 9.0.2:

```
l_http_header := owa_util.get_cgi_env('HTTP_Acme_USER');
debug_print('Acme ID : ' || l_http_header);
```

ユーザー名が存在しない場合のエラー処理

リリース 10.1.3:

```
}
catch (Exception e)
{
   DebugUtil.debug(SSODebug.ERROR, "exception: " + CLASS_NAME, e);
   throw new IPASInsufficientCredException("No Acme Header");
}

if (AcmeUserName == null)
throw new IPASInsufficientCredException("No Acme Header");
```

リリース 9.0.2:

```
IF ( (l_ssouser IS NULL) or
    ( INSTR(l_ssouser, GLOBAL_SEPARATOR) != 0) ) THEN
    debug_print('malformed user id: '
      || l_ssouser
      || ' returned by wwsso_auth_external.authenticate_user');
     RAISE EXT_AUTH_FAILURE_EXCEPTION;
ELSE
```

Single Sign-On Server に戻すユーザー名

リリース 10.1.3:

```
IPASUserInfo authUser = new IPASUserInfo(AcmeUserName);

return authUser;

}
```

リリース 9.0.2:

```
p_user := NLS_UPPER(l_ssouser);   -- p-user is the out parameter
return 0;                         -- SUCCESS error code
END IF;
```

15

データのエクスポートとインポート

この章では、複数の Single Sign-On Server 間でデータを移動する方法について説明します。データのエクスポートまたはインポートが必要になる様々な状況があります。たとえば、テスト・サーバーでデータを用意してから、本番サーバーにデータを移動することがあります。また、複数のサーバーを 1 つに統合することもあります。既存のサーバーをバックアップすることもあります。

この章の項目は次のとおりです。

エクスポートされるデータとインポートされるデータ

エクスポートとインポートのスクリプト（ssomig）によって、次の 3 つのカテゴリのデータが移動されます。

- 外部アプリケーションの定義とユーザー・データ
- パートナ・アプリケーションの登録 URL とトークン
- OracleAS Discoverer で様々なデータソースへのアクセスに使用する接続情報

ユーザー・アカウントを移動する必要がある場合は、LDAP コマンドライン・スクリプト（ldapsearch など）を使用してソース・ディレクトリからデータを抽出します。ldapadd または ldapmodify を使用して、ターゲット・ディレクトリにデータをロードします。これらのスクリプトの使用方法は、『Oracle Identity Management アプリケーション開発者ガイド』の構文に関する章を参照してください。

エクスポートとインポートのスクリプト : 構文とパラメータ

ssomig スクリプトでは、Perl、Oracle SQL*Plus、データベース・エクスポートおよびインポート・ツール（exp と imp）を使用して、リリース 10.1.3 の 2 つのサーバー間でデータを移動します。エクスポート・モードとインポート・モードは別々に実行する必要があります。ssomig は *ORACLE_HOME*/sso/bin にあります。

スクリプト構文

次の構文を使用して、ssomig を実行します。

```
ssomig -s sso_schema
       -p sso_password
       -c net_service_name
       -log_d log_dir
       {
         -export [-prompt]
                 [-noextappusrs]

         -import {-merge | -overwrite}
                 [-discoforce | -disconoforce]
       }
        [-log_f log_file]
        [-d dump_file_name]
        [-help]
```

スクリプト・パラメータ

表 15-1 に、ssomig に渡すパラメータの定義を示します。

表 15-1 ssomig に渡すパラメータ

パラメータ	説明	追加情報
-s	OracleAS Single Sign-On のデータベース・スキーマ名。	デフォルトは ORASSO です。
-p	OracleAS Single Sign-On のデータベース・スキーマ・パスワード。	OracleAS Infrastructure のインストール時にパスワードはランダム化されます。パスワードの取得方法は、付録 B を参照してください。
-c	OracleAS Single Sign-On データベースのネット・サービス名。	-
-log_d	ログ・ディレクトリ名。	このディレクトリは書込み可能にする必要があります。ログ・ファイル、エクスポート構成ファイルおよびダンプ・ファイルがこのディレクトリに書き込まれます。 スクリプトの実行時にはディレクトリの絶対パスを使用します。デフォルトは *ORACLE_HOME*/sso/log です。
-export	シングル・サインオン表からデータを抽出し、ダンプ・ファイルにロードします。	-
-prompt	パートナ・アプリケーションと外部アプリケーションを選択してエクスポートします。	export で使用します。
-noextappusrs	外部アプリケーションのユーザーがエクスポートされないように指定します。	export で使用します。 段階的なサーバーから本番サーバーにデータを移動するが、テスト・ユーザーは移動しない場合に、このモードを選択します。
-import	ダンプ・ファイルからデータを抽出し、シングル・サインオン表にロードします。	-
-merge	ターゲット・サーバーに存在しないパートナ・アプリケーションと外部アプリケーションのみをインポートします。	複数のサーバーから最初のサーバーをインポートした後、このモードを選択します。 import で使用します。
-overwrite	ターゲット・サーバーに存在するか否かにかかわらず、すべてのパートナ・アプリケーションと外部アプリケーションをインポートします。	複数のサーバーから最初のサーバーを移行する場合に、このモードを選択します。 import で使用します。
-discoforce	OracleAS Discoverer 情報をインポートして、ターゲット・サーバーの Discoverer 情報を置き換えます。	-
-disconoforce	ターゲット・サーバーに Discoverer データが存在しない場合にのみ、OracleAS Discoverer 情報をインポートします。	-
-log_f	ログ・ファイル名。	このファイルには、エクスポート結果と、SQL*Plus、exp、imp などのツールのランタイム・ステータスが含まれます。デフォルトのファイル名は ssomig.log です。
-d	ダンプ・ファイル名。	デフォルトは ssomig.dmp です。
-help	ssomig の構文とパラメータを記述します。	-

サーバー間でのデータのエクスポート

エクスポートとインポートのスクリプトが実行される際の使用例は、2 つのカテゴリに分けられます。単一サーバーからのエクスポートと、複数のサーバーからのエクスポートです。カテゴリに応じて、スクリプトは上書きモードまたはマージ・モードで実行されます。また、カテゴリに応じて、パートナ・アプリケーションと外部アプリケーションを選択してエクスポートするかどうかも指定します。この項では、単一サーバーでのエクスポートおよびインポートについて説明します。複数のサーバーのエクスポートおよびインポートについては、「複数のサーバーの統合」の項を参照してください。

この項の項目は次のとおりです。

エクスポートとインポートの使用例およびスクリプトの例

以降では、Single Sign-On Server 間でデータを移動する場合の使用例について説明します。各使用例では適切なコマンドを紹介します。

注意： 次の例は、UNIX を念頭において記述されていますが、これらは Windows でも動作します。ログ・ディレクトリ・パスで、フォワード・スラッシュを円記号（¥）に置き換えてください。

エクスポートの使用例

- パートナ・アプリケーションと外部アプリケーションをすべてエクスポートします。OracleAS Discoverer のデータ全体をエクスポートします。サーバーをバックアップする場合は、次のコマンドを使用します。

  ```
  ssomig -export -s orasso -p password -c net_service_name -log_d /tmp
  ```

- パートナ・アプリケーションと外部アプリケーションを選択してエクスポートします。OracleAS Discoverer のデータ全体をエクスポートします。段階的なデータを本番サーバーに移動する場合は、次のコマンドを実行します。

  ```
  ssomig -export -prompt -s orasso -p password -c net_service_name -log_d /tmp
  ```

- パートナ・アプリケーションを選択してエクスポートします。外部アプリケーションの定義を選択してエクスポートします。外部アプリケーションのユーザー・データはエクスポートしません。OracleAS Discoverer のデータ全体をエクスポートします。段階的なデータを本番サーバーに移動するが、テスト・ユーザーの外部アプリケーション情報を移動しない場合は、次のコマンドを実行します。

  ```
  ssomig -export -prompt -noextappusrs -s orasso -p password -c net_service_name
  -log_d /tmp
  ```

インポートの使用例

- パートナ・アプリケーションと外部アプリケーションをインポートします。インポートするエントリと同じエントリのみを上書きします。OracleAS Discoverer データは除外します。Discoverer を配置しない場合は、次のコマンドが便利です。

  ```
  ssomig -import -overwrite -s orasso -p password -c net_service_name -log_d /tmp
  ```

- パートナ・アプリケーション、外部アプリケーション、OracleAS Discoverer データをインポートします。インポートするエントリと同じエントリであるかどうかにかかわりなく、すべてのエントリを上書きします。ターゲット・サーバーでデータをリフレッシュする場合は、次のコマンドを実行します。

  ```
  ssomig -import -overwrite -s orasso -p password -c net_service_name -log_d /tmp
  -discoforce
  ```

- パートナ・アプリケーションと外部アプリケーションをインポートします。インポートするエントリと同じエントリであるかどうかにかかわりなく、すべてのエントリを上書きします。OracleAS Discoverer 情報は、ターゲット・サーバーに存在しない場合にのみインポートします。

  ```
  ssomig -import -overwrite -s orasso -p password -c net_service_name -log_d /tmp
  -disconoforce
  ```

スクリプトの実行

データをエクスポートする手順は、次のとおりです。

1. エクスポート元のコンピュータにログインします。
2. リリース 10.1.3 の Single Sign-On Server の Oracle ホームを指すように、Oracle ホームの環境変数 ORACLE_HOME を設定します。
3. スクリプトを実行します。(「エクスポートとインポートの使用例およびスクリプトの例」の項を参照)。

   これにより、ダンプ・ファイル ssomig.dmp、ログ・ファイル ssoconf.log、シングル・サインオンの構成ファイル ssoconf.log が作成されます。これら 3 つのファイルはログ・ディレクトリに作成されます。

   注意 : エクスポート・モードでプロンプト・オプションを使用して ssomig を実行すると、エクスポートから除外するアプリケーションを指定するよう要求されます。同時に、その選択が終了したら任意のキーを押すよう要求されます。ここでは、かわりに [Return] キーまたは [Enter] キーを押してください。このスクリプトでは、他のキーは無視されます。

データをインポートする手順は、次のとおりです。

1. インポート先のコンピュータにログインします。
2. リリース 10.1.3 の Single Sign-On Server の Oracle ホームを指すように、環境変数 ORACLE_HOME を設定します。
3. log_d パラメータは、エクスポートのログ・ファイルがあるログ・ディレクトリを指すようにします。インポート・モードで実行される場合、スクリプトは ssomig.dmp、ssoconf.log の各ファイルを参照する必要があります。エクスポート・サーバーがあるコンピュータから、これらのファイルをコピーする場合もあります。
4. import モードを選択して、スクリプトを実行します。(「エクスポートとインポートの使用例およびスクリプトの例」の項を参照)。

エクスポートとインポートの成功の確認

エクスポート操作とインポート操作が完了したら、ssomig.log を開いてエラーをチェックします。ファイル内のメッセージについて確認する場合は、「エラー・メッセージ」の項を参照してください。

複数のサーバーの統合

企業内の複数の部門でそれぞれ Single Sign-On Server を設置している場合は、この使用例が当てはまります。このようなサーバーを統合して、統合 ID 管理サービスを実現する場合があります。

次の方法で、複数のサーバーをエクスポートおよびインポートします。

1. ターゲット・サーバーを除き、すべての関連サーバーからデータをエクスポートします。スクリプトの実行方法については、「サーバー間でのデータのエクスポート」の項を参照してください。
2. 最初に移行する Single Sign-On Server に対して、`import` モード、`overwrite` オプションでスクリプトを実行します。詳細は、「インポートの使用例」の項を参照してください。
3. 以降のサーバーについては、`merge` モードでスクリプトを実行します。パートナ・アプリケーションと外部アプリケーションをターゲット・サーバーに 1 つずつインポートします。

   ```
   ssomig -import -merge -s orasso -p password -c net_service_name -log_d /tmp -d
   ssomig.dmp
   ```

   このコマンドでは、パートナ・アプリケーションと外部アプリケーションのみがマージされます。

> **注意：** 複数のサーバーをインポートする場合は、`overwrite` モードでスクリプトを実行して、前回の実行結果を取り消すことができます。

エラー・メッセージ

エクスポートおよびインポートの際には、次に示すエラー・メッセージが表示されることがあります。表 15-2 には、問題解決に役立つエラー・メッセージが定義されています。

表 15-2 エクスポートおよびインポートに関するエラー・コード

エラー	原因	処置
`SSO-80000: The operation was unsuccessful.`	1 つ以上のエラーによって、インポートまたはエクスポート（あるいは両方）が失敗しました。	ログ・ファイルまたはスクリーン画面の内容からエラーを判別します。
`SSO-80001: The environment variable ORACLE_HOME is not set.`	リリース 10.1.3 の Oracle ホームに変数が設定されていません。	「スクリプトの実行」の手順に従ってください。
`SSO-80002: Invalid ORACLE_HOME specified.`	`ORACLE_HOME` で示されるディレクトリは存在しないか、ディレクトリ内の必要なスクリプトを入手できません。	Oracle ホームを有効な Oracle インスタンスに設定します。
`SSO-80004: Invalid log directory. String is not writable.`	指定されたログ・ディレクトリに対する書込み権限がありません。	書込み権限が付与されているディレクトリを指定します。
`SSO-80005: Invalid log directory. String is not directory.`	指定されたログ・ディレクトリは存在しません。	有効なディレクトリを指定します。
`SSO-80008: Duplicate option string.`	コマンドライン・パラメータ文字列が繰り返されているか、相補的なオプションのセットが 2 つとも指定されています。	コマンドライン・パラメータ文字列は繰り返さないでください。 相補的なオプションのセット（`export` と `import` など）を 2 つとも指定しないでください。
`SSO-80009: Mandatory parameter missing: string.`	必須のコマンドライン・パラメータ文字列が指定されていません。	適切な値でパラメータ文字列を指定します。

表 15-2 エクスポートおよびインポートに関するエラー・コード（続き）

エラー	原因	処置
SSO-80010: Invalid SSO Server version detected.	スクリプトでは、ソース・サーバーまたはターゲット・サーバーのバージョンをサポートしません。	リリース 10.1.3 のサーバーを使用してエクスポート操作およびインポート操作を実行していることを確認します。
SSO-80011: Invalid option string.	パラメータ文字列が、認識されたコマンドライン・パラメータではありません。	help オプションを使用して有効なパラメータの一覧を取得します。
SSO-80012: Invalid SSO schema information.	スキーマ名、パスワードまたはネット・サービス名が無効です。	コマンドを再入力します。
SSO-80014: Invalid log file. String is not writable.	指定されたログ・ファイルに対する書込み権限がありません。	書込み権限が付与されているログ・ファイルを指定します。
SSO-80015: Failed to drop temporary tables.	必要なスクリプト・ファイルがなかったか、オペレーティング・システム・エラーまたはデータベース・エラーが発生しました。	ログ・ファイルで詳細を確認します。エラーがあれば修正します。
SSO-80050: Data export unsuccessful.	1 つ以上のエラーによって、エクスポート操作が失敗しました。	ログ・ファイルまたはスクリーン画面の内容からエラーを判別します。
SSO-80051: Copying data into the temporary tables failed.	スクリプト・ファイルがないか、オペレーティング・システム・エラーまたはデータベース・エラーが発生しました。	ログ・ファイルで詳細を確認します。エラーがあれば修正します。
SSO-80052: Invalid dump file. String not writable.	指定されたダンプ・ファイルに対する書込み権限がありません。	書込み権限が付与されているダンプ・ファイルを指定します。
SSO-80076: Cannot determine NLS information.	スクリプト・ファイルがないか、オペレーティング・システム・エラーまたはデータベース・エラーが発生しました。	ログ・ファイルで詳細を確認します。エラーがあれば修正します。
SSO-80077: The file string does not exist.	外部でファイル文字列が削除されたか、名前が変更されました。	スクリプト実行時には、外部でファイル文字列を編集または削除しないようにします。
SSO-80078: Creating the table that represents the config file failed.	スクリプト・ファイルがないか、オペレーティング・システム・エラーまたはデータベース・エラーが発生しました。	ログ・ファイルで詳細を確認します。エラーがあれば修正します。
SSO-80100: Data import unsuccessful.	1 つ以上のエラーによって、インポート操作が失敗しました。	ログ・ファイルまたはスクリーン画面の内容からエラーを判別します。エラーがあれば修正します。
SSO-80101: Cannot read the import dump file: string.	ダンプ・ファイル文字列に対する読取り権限がありません。	指定されたダンプ・ファイルに対する読取り権限を取得します。
SSO-80102: The dump file string is of size zero.	エクスポート時にエラーが発生しました。	ログ・ファイルを表示します。エラーがあれば修正します。
SSO-80103: Config file not found: string.	ダンプ・ファイルやログ・ファイルなどの必要な構成ファイルがインポート時に見つからない場合に、このエラーが発生します。	構成ファイルがログ・ディレクトリに存在することを確認します。
SSO-80104: Corrupted or invalid config file.	構成ファイルが変更されました。	ソースからターゲットへの送信時には、構成ファイルが変更されないようにします。
SSO-80150: Package loading into the SSO schema failed.	スクリプト・ファイルがないか、オペレーティング・システム・エラーまたはデータベース・エラーが発生しました。	ログ・ファイルで詳細を確認します。エラーがあれば修正します。

A

OracleAS Single Sign-On のトラブルシューティング

この付録では、Oracle Application Server Single Sign-On の使用時に発生しうる一般的な障害とその解決方法について説明します。また、ログ・ファイルの確認やデバッグの有効化など、OracleAS Single Sign-On 環境での問題の診断と解決に役立つ情報も提供します。

この付録の項目は次のとおりです。

Single Sign-On Server の一般的なエラーに関する障害と解決策

この項では、Single Sign-On Server の起動やアクセスに際して発生する一般的な障害と解決策について説明します。この項の項目は次のとおりです。

URL が最大長を超えました

一部のブラウザでは、保護されたリソースにユーザーがアクセスしようとすると、リクエストされた URL がブラウザで許可されている最大長より短い場合でも、URL の最大長を超える場合があります。この問題は、OracleAS Single Sign-On Server に情報を渡すときに、デフォルトでは mod_osso が GET ディレクティブを使用するために発生します。GET ディレクティブを使用すると、リクエスト処理の過程で、暗号キー、サイト ID、サイト・トークン、クライアントの IP アドレス、タイムスタンプなどが URL に追加されます。

障害

保護された URL にユーザーがアクセスしようとしても、OracleAS Single Sign-On Server からステータス・コード 302 が返され、リクエストしたリソースを取得できません。

解決策

OssoRedirectByForm ディレクティブを *$ORACLE_HOME*/Apache/Apache/conf ディレクトリにある mod_osso.conf ファイルに追加して、POST を使用するように mod_osso を構成します。

内部サーバー・エラー

OracleAS Single Sign-On にアクセスしたとき、Single Sign-On Server が正しく起動されなかった場合や、インフラストラクチャ・コンポーネントに接続できなかった場合に、「内部サーバー・エラー」というメッセージが表示されることがあります。

障害 1

Single Sign-On Server に接続したとき、次のエラー・メッセージが表示されます。

内部サーバー・エラーです。管理者に通知してください

このエラー・メッセージは、通常 Single Sign-On Server が正しく起動されない場合に表示されます。

解決策 1

次の手順で問題を解決します。

1. Single Sign-On Server が正しく起動されたかどうかを確認します。確認するには、起動ログ・ファイル *ORACLE_HOME*/opmn/logs/OC4J~OC4J_SECURITY~default_island~1 をチェックします。
2. ログ・ファイルにデータベース・エラーまたは Oracle Internet Directory のエラーが記録されている場合は、Single Sign-On Server の起動前にデータベースと Oracle Internet Directory が両方とも起動および稼働されていることを確認します。`SSOLoginServlet.init: SSO server started` というメッセージがあれば、Single Sign-On Server は正しく起動されています。
3. 次に、Single Sign-On Server のログ・ファイル（*ORACLE_HOME*/sso/log/ssoServer.log）をチェックします。
4. このログ・ファイルに `NumberFormatException or a specific configuration parameter not found` というエラー・メッセージが記録されている場合は、`policy.properties` で空白をチェックします。問題のありそうな構成が設定された行の行端にある空白を削除し、OC4J_SECURITY インスタンスを再起動します。

   ```
   ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
   ```

5. ファイル *ORACLE_HOME*/opmn/logs/OC4J~OC4J_SECURITY~default_island~1 にエラー・メッセージ `Orion Launcher SSO Server initialization failed` が記録されている場合は、次の手順を実行します。
   - データベースが使用可能であることを確認し、Single Sign-On Server を再起動します。

     ```
     ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
     ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
     ```

   - データベースが使用可能である場合、ディレクトリの接続に問題がある可能性があります。opmn ログをチェックします。次のエラー・メッセージがある場合、`ssooconf.sql` を実行して、Single Sign-On データベースでディレクトリ・アクセスが適切に構成されるようにします。

     ```
     java.lang.NumberFormatException: null
       at java.lang.Integer.parseInt(Integer.java:442)
       at java.lang.Integer.parseInt(Integer.java:524)
       at oracle.security.sso.server.conf.DatabaseConfigReader.
       setSSOServerConfig(DatabaseConfigReader.java:322)
     ```

6. `ssooconf.sql` の実行方法は、第 3 章の「ディレクトリ・アクセス用 Single Sign-On Server の設定変更」の項を参照してください。

障害 2

Single Sign-On Server に接続したとき、次のエラー・メッセージが表示されます。

内部サーバー・エラーです。後で操作を実行してください

インフラストラクチャ・データベースまたは Oracle Internet Directory が使用不可能であるか停止している場合に、このエラー・メッセージが表示されます。

解決策 2

ORACLE_HOME/sso/log/ssoServer.log でメッセージの詳細をチェックして、内容に応じてデータベースまたは Oracle Internet Directory の再起動を試みます。

予期しないエラー

Single Sign-On Server に接続したとき、次のエラー・メッセージが表示されます。

予期しないエラーが発生しました。管理者に通知してください

障害

このメッセージは、サーバー側のエラーを示している可能性があります。policy.properties ファイルが正しく構成されていないか、Java クラスがロードされていない可能性があります。また、パートナ・アプリケーションが正しく登録されていない場合もあります。

解決策

障害の原因を特定するには、次の手順を実行します。

1. 次のログ・ファイルでエラー・メッセージをチェックします。

   Single Sign-On Server のログ : *ORACLE_HOME*/sso/log/ssoServer.log

   Oracle HTTP Server のエラー・ログ : *ORACLE_HOME*/Apache/Apache/logs/error_log

2. OracleAS Single Sign-On の管理ページにログオンします。cn=orcladmin ではなく、orcladmin でログインします。

   http://*single_sign-on_host*:*single_sign-on_port*/sso

3. ログインできる場合は、Single Sign-On Server の問題ではなく、パートナ・アプリケーションの登録かアプリケーション自体に問題があります。

ファイルが見つからないエラー

Single Sign-On Server にアクセスしたとき、次のエラー・メッセージが表示されます。

ファイルが見つかりません

障害

Oracle HTTP Server のエラー・ログ（*ORACLE_HOME*/Apache/Apache/logs/error_log）をチェックします。ファイルが見つかりませんというメッセージがある場合、Oracle HTTP Server は認証リクエストを OC4J に委譲していません。

解決策

次のチェックを実行します。

1. mod_oc4j.conf で Single Sign-On アプリケーションのマッピングをチェックします。マウント構成 Oc4jMount/sso OC4J_SECURITY が設定されている必要があります。
2. default-web-access.log をチェックして、サーブレットで認証リクエストが受け取られたかどうかを確認します。

認証に失敗しました。

OracleAS Single Sign-On にログオンした後、「認証に失敗しました。」というエラーが表示される場合があります。

障害

ユーザーのパスワードが正しくないか、サーバーにユーザー認証に必要な権限がありません。

解決策

1. ユーザー DN が適切なレルムに対応することを確認して、ユーザーとしてディレクトリへのバインドを試みます。

   ```
   ORACLE_HOME/bin/ldapbind
   -h directory_server
   -p directory_ssl_port
   -D user_dn
   -w user_password
   -u 1
   ```

   バインドに失敗した場合は、ユーザーのパスワードが正しくありません。パスワードを再設定します。バインドに成功した場合は、手順 2 に進みます。

2. Single Sign-On Server としてディレクトリへのバインドを試みます。

   ```
   ORACLE_HOME/bin/ldapbind
   -h directory_server
   -p directory_ssl_port
   -D orclApplicationCommonName=ORASSO_
    SSOSERVER,cn=SSO,cn=Products,cn=OracleContext
   -w single_sign-on_server_password
   ```

   バインドに失敗した場合は、バインドで使用したサーバー・パスワードが正しくない可能性があります。正しいパスワードを設定するには、第 3 章の「ディレクトリ・アクセス用 Single Sign-On Server の設定変更」の説明に従って `ssooconf.sql` を実行します。バインドに成功した場合は、手順 3 に進みます。

3. Single Sign-On アプリケーションが SecurityAdmins グループのメンバーであるかどうかを確認します。このグループのメンバーでない場合は、ユーザーを認証できません。

   ```
   ORACLE_HOME/bin/ldapcompare
   -h directory_host
   -p directory_ssl_port
   -D orclApplicationCommonName=ORASSO_
    SSOSERVER,cn=SSO,cn=Products,cn=OracleContext
   -w orasso_password
   -b "cn=user_dn,cn=users,realm_dn"
   -a userpassword
   -v user_password
   ```

   メンバーでない場合は、SecurityAdmins グループにアプリケーションを追加し（`cn=OracleUserSecurityAdmins,cn=Groups,cn=OracleContext`）、ユーザーを再度認証します。アプリケーションがメンバーである場合は、問題の原因がディレクトリにある可能性があります。

認証用に送信されたユーザー名は、既存のシングル・サインオン・セッションに存在するユーザー名と一致しません。

障害

ユーザーがこのエラーに遭遇するのは、強制認証のリクエスト中のみです。最初の認証時と異なるユーザー ID、および場合によってはレルムを入力した場合に、このエラー・メッセージが表示されます。

解決策

強制認証中に入力するユーザー ID、および場合によってはレルムは、現在のシングル・サインオン・セッションのユーザー ID およびレルムと一致している必要があります。別の資格証明を使用してログインする場合は、その前に現在のセッションをログアウトしてください。

OracleAS Single Sign-On の管理にアクセスしたときに空白のページが表示されます

管理者が OracleAS Single Sign-On の管理ページにログオンしたときに、空白のページが表示されます。

障害 1

Oracle Internet Directory に `PUBLIC` ユーザー・エントリがないか、ディレクトリでユーザー・ニックネーム属性が変更されても新しい属性が `PUBLIC` エントリに追加されていません。

解決策 1

ディレクトリのユーザー検索ベースに `PUBLIC` ユーザー・エントリを追加します。ユーザー・ニックネーム属性が変更されている場合は、属性を `PUBLIC` ユーザー・エントリに追加します。

障害 2

ディレクトリに不適切な情報を使用して、Single Sign-On Server が構成されています。

解決策 2

`ssooconf.sql` を実行して、正しいディレクトリ情報で Single Sign-On Server を構成します。スクリプトの実行方法は、第 3 章の「ディレクトリ・アクセス用 Single Sign-On Server の設定変更」の項を参照してください。

障害 3

インストールに問題（Enabler エントリがない、または、不適切な SSL 登録）がある可能性があります。

解決策 3

`ssooconf.sql` を実行して、Enabler エントリで Single Sign-On Server を更新するか、SSL のシングル・サインオン URL を変更します。

障害 4

ディレクトリ DIT が変更されたが、この変更内容で Single Sign-On Server が更新されていません。

解決策 4

`ssoreoid.sql` を実行して、ディレクトリ DIT で Single Sign-On Server を更新します。

管理者の画面に OracleAS Single Sign-On の管理ページが表示されません

管理者が .../sso にログインしても、OracleAS Single Sign-On の管理ページが表示されません。

障害

管理者が iASAdmins グループのメンバーではありません。

```
cn=iASAdmin,cn=Groups,cn=OracleContext,realm_dn
```

解決策

ディレクトリで iASAdmins エントリの uniquemember 属性をチェックします。

```
ldapsearch -h directory_host
           -p directory_port
           -D orclApplicationCommonName=ORASSO_
              SSOSERVER,cn=SSO,cn=Products,cn=OracleContext
           -w orasso_password
           -b "cn=iasadmins,cn=groups,cn=oraclecontext,realm_dn"
              "uniquemember=cn=user,cn=users,realm_dn"
```

コマンド内の *user* が iASAdmins の一意のメンバーでない場合は、第 2 章の「管理権限の付与」の手順に従ってください。

「SSO Server 管理」リンクが OracleAS Single Sign-On の管理ページから失われています

障害

このリンクを確認できるのは、管理者のみです。リンクの表示されないユーザーは、エンド・ユーザーとしてログインしています。

解決策

次のエントリで、ユーザーが iASAdmins グループのメンバーであることを確認します。

```
cn=iASAdmins,cn=Groups,cn=OracleContext,dc=default_identity_management_realm
```

OracleAS Single Sign-On 管理グループを変更した場合、ユーザーがこのグループのメンバーであることを確認してください。

監査ログの挿入例外 : ORA-00018: 最大セッション数を超えました

Single Sign-On Server で過剰負荷が発生している場合に、このメッセージが表示されます。

障害

要求されたデータベース・セッション数が、init.ora ファイルで指定された数を超えています。

解決策

ID 管理インフラストラクチャ・データベースのプロパティを変更します。具体的には、processes パラメータと sessions パラメータの値を増加して、予測される負荷に対応するようにします。データベース固有の構成ファイル（init.ora など）を使用して変更を行います。init.ora は *ORACLE_HOME*/dbs にあります。

接続制限を超えました

障害

このメッセージは、「監査ログの挿入例外 : ORA-00018: 最大セッション数を超えました」というメッセージと関連性があります。

解決策

エンド・ユーザーが操作を再試行してください。または、管理者が接続制限を増やします。

システムがアイドル状態のときの、ログインが失敗したというメッセージ

OracleAS Single Sign-On がファイアウォールの内側で稼働し、しばらくアイドル状態になっているときに、ログインが失敗したというエラーが表示される場合があります。*ssoserver.log* に、次のテキストが記録されます。

```
AJPRequestHandler-ApplicationServerThread-11 DB connection error
java.sql.SQLException: Closed Connection
at oracle.jdbc.dbaccess.DBError.throwSqlException(DBError.java:189)
at oracle.jdbc.dbaccess.DBError.throwSqlException(DBError.java:231)
at oracle.jdbc.dbaccess.DBError.throwSqlException(DBError.java:294)
```

障害

一般的な本番環境では、ファイアウォールは OracleAS Single Sign-On と Oracle Internet Directory のサーバーを保護します。ファイアウォールは接続アクティビティを追跡し、ファイアウォールのタイムアウト値によって制御される期間が過ぎると、非アクティブなデータベース接続やディレクトリ接続を削除します。

データベース接続の削除が通知されていない場合、Single Sign-On Server が、失効した接続を使用して操作を実行しようとし、このエラーが発生する場合があります。

解決策

A-8 ページの「アイドル状態の LDAP またはデータベースの接続タイムアウトに起因するエラー」に示す解決策に従ってください。

アイドル状態の LDAP またはデータベースの接続タイムアウトに起因するエラー

システムに LDAP ファイアウォールが構成され、ファイアウォールがアイドル状態の LDAP 接続またはデータベース接続を削除する場合、OracleAS Single Sign-On Server へのログイン時に、内部サーバー・エラーが表示されることがあります。

障害

Oracle Application Server Single Sign-On と、LDAP またはデータベース・サーバーの間にファイアウォールがある場合、ファイアウォールが非アクティブな LDAP 接続またはデータベース接続を削除すると、ログイン・エラーが発生する場合があります。

解決策

OracleAS Single Sign-On Server の LDAP 接続時間またはデータベース接続時間を制御するパラメータを設定できます。このパラメータは `connectionIdleTimeout` と呼ばれ、`policy.properties` 構成ファイル内で値を分単位で指定します。このパラメータは、OracleAS Single Sign-On と、LDAP サーバーまたはデータベース・サーバーの間でファイアウォールを利用するデプロイで役立ちます。LDAP 接続またはデータベース接続がこのパラメータの値より長い期間アイドル状態になると、OracleAS Single Sign-On Server はプールからその接続を削除し、プールの新しい接続を使用しようとします。

LDAP またはデータベースの接続タイムアウトを設定するには、次の手順を実行します。

1. ORACLE_HOME/sso/conf/policy.properties ファイルを編集して、connectionIdleTimeout パラメータの値を追加または更新します。この値は、分を表す整数です。次の例では、アイドルのタイムアウト値を 120 分に設定しています。

   ```
   connectionIdleTimeout = 120
   ```

2. OC4J を再起動して OracleAS Single Sign-On を再起動します。

Portal へのログインの失敗

ユーザーが Portal、または OracleAS Single Sign-On によって保護されたアプリケーションにログインしようとすると、次のいずれかのエラーが表示されます。

- wwsec_app_priv.process_signon で予期しないエラーが発生（ユーザー定義例外）（WWC-41417）
- Oracle Internet Directory でエントリが見つかりませんでした（エラー・ステータス： -5）： 指定したユーザーは、ディレクトリ内に存在しません
- 詳細な操作： dbms_ldap_utl.get_group_membership（WWC-41745）

障害

Single Sign-On Server がデフォルトでユーザー・エントリをキャッシュすることが原因です。Oracle Internet Directory でユーザーを一度削除して再作成した場合、そのユーザー・エントリの orclGUID 属性が変更されるため、キャッシュとディレクトリのデータが同期しなくなります。その後、アプリケーションが Oracle Internet Directory のユーザー・エントリにアクセスしようとすると、Single Sign-On Server によって返される orclGUID 値とエントリの orclGUID は一致しません。

また、このエラーは、複数の検索ベースが構成されているレルムにユーザーを追加、およびそのようなレルムからユーザーを削除した場合にも発生します（cn=Common,cn=Products,cn=OracleContext エントリの orclcommonusersearchbase 属性）。たとえば、同じニックネームのユーザーが複数の検索ベースに存在する場合は、最初にリストされている検索ベースからユーザー・エントリが削除されます。そのため、キャッシュとディレクトリのデータの間に不一致が生じる場合があります。

解決策

キャッシュを無効にするには、次の手順を実行します。

1. /sso/conf/policy.properties のバックアップをとります。
2. policy.properties ファイルを編集して、cacheSize=1000 を cacheSize=-1 に変更します。
3. 次のコマンドで Single Sign-On Server を再起動します。

   ```
   opmnctl restartproc process-type=OC4J_SECURITY
   ```

証明書による認証のエラーの障害と解決策

証明書による認証に対して一般的なデバッグを行うには、次の手順を実行します。

1. `policy.properties` でデバッグ・レベルを `DEBUG` に設定し、シングル・サインオン中間層を再起動します。

   ```
   ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
   ORACLE_HOME/opmn/bin/opmnctl startproc process-type=OC4J_SECURITY
   ```

2. デバッグ時にブラウザの証明書情報を表示するには、*ORACLE_HOME*`/sso/lib/ipassample.jar` から `certinfo.jsp` ファイルを抽出します。
3. ファイルを *ORACLE_HOME*`/j2ee/applications/sso/web/jsp` に配置します。
4. 次の URL でファイルを表示します。

   `https://`*host*`:`*port*`/sso/certinfo.jsp`

OracleAS Single Sign-On で証明書による認証を使用すると、次の問題が発生する場合があります。

- ネットワーク・エラー : 接続が拒否されました
- Single Sign-On Server がユーザーの証明書を要求しません
- 証明書による認証が失敗し、ユーザーにログイン・ページが表示されます
- ユーザーのブラウザの証明書がありません
- マッピング・モジュールのクラス名が見つかりません
- マッピング・モジュールのインスタンスを作成できませんでした
- マッピング・モジュール・オブジェクトを作成できません
- マッピング・モジュールの作成中に例外が発生しました
- 証明書が一致しませんでした

ネットワーク・エラー : 接続が拒否されました

障害

ユーザーが SSL を介してパートナ・アプリケーションへのアクセスを試みると、このメッセージが表示されます。`httpd.conf` に `SSLEngine on` パラメータがないか、このパラメータが正しく入力されていない可能性があります。

解決策

第 8 章の「SSL パラメータの構成」の項の説明に従って、欠落しているパラメータを追加します。パラメータが存在し、正しく入力されている場合は、Oracle HTTP Server のログ・ファイルに問題が示されている可能性があります。

Single Sign-On Server がユーザーの証明書を要求しません

障害

オプション・パラメータ `SSLVerifyClient` が `httpd.conf` にないか、正しく入力されていません。

解決策

第 8 章の「SSL パラメータの構成」の項の説明に従って、欠落しているパラメータを追加します。パラメータが存在し、正しく入力されている場合は、Oracle HTTP Server のログ・ファイルに問題が示されている可能性があります。

証明書による認証が失敗し、ユーザーにログイン・ページが表示されます

障害

ユーザーの証明書がディレクトリにないか、正しく入力されていません。`ssoServer.log` で詳細を確認します。

解決策

ユーザーの証明書をディレクトリに再入力します。第 8 章の「Oracle Internet Directory」の手順を参照してください。

ユーザーのブラウザの証明書がありません

障害 1

ユーザーの証明書がブラウザにありません。

解決策 1

有効な証明書をユーザーのブラウザにインストールします。

障害 2

Oracle HTTP Server 上の SSL Wallet に、クライアント証明書を発行した認証局の信頼できる証明書が含まれていない可能性があります。

解決策 2

Oracle Wallet Manager を使用して、SSL Wallet に信頼できる証明書が含まれているかどうかを確認します。このツールの使用方法は、『Oracle Application Server 管理者ガイド』の Wallet および証明書の管理に関する章を参照してください。

マッピング・モジュールのクラス名が見つかりません

障害

`x509CertAuth.properties` にマッピング・モジュールのクラス名がないか、クラス名が正しくありません。

解決策

パラメータ `CertificateMappingModule` に値が割り当てられていることを確認します。割り当てられている場合は、この値が正しいことを確認します。

マッピング・モジュールのインスタンスを作成できませんでした

障害
カスタマイズしたマッピング・モジュールが正しく実装されていません。

解決策
カスタム・モジュールにデフォルト・コンストラクタがあることを確認します。

マッピング・モジュール・オブジェクトを作成できません

障害
カスタマイズしたマッピング・モジュールが正しく実装されていません。

解決策
第 8 章の「ユーザー名マッピング・モジュールのカスタマイズ（オプション）」に従って、指定されたインタフェースをカスタマイズしたモジュールで実装してください。

マッピング・モジュールの作成中に例外が発生しました

障害
カスタマイズしたマッピング・モジュールが正しく実装されていません。

解決策
第 8 章の「ユーザー名マッピング・モジュールのカスタマイズ（オプション）」に従って、指定されたインタフェースをカスタマイズしたモジュールで実装してください。

証明書が一致しませんでした

障害
ユーザーの証明書がディレクトリにないか、正しく入力されていません。`ssoServer.log` で詳細を確認します。

解決策
ユーザーの証明書をディレクトリに再入力します。第 8 章の「Oracle Internet Directory」の手順を参照してください。

Windows のネイティブ認証のエラーの障害と解決策

OracleAS Single Sign-On で Windows のネイティブ認証（WNA）を使用すると、次の問題が発生する場合があります。

- Windows での認証後にユーザーが URL にアクセスできない
- Windows で認証済のユーザーをブラウザで認証できない
- 資格証明が見つからないというエラーで Single Sign-On Server の起動に失敗する
- Single Sign-On Server に内部サーバー・エラーが表示される
- Single Sign-On ユーザーを KDC で認証できない
- パートナ・アプリケーションにアクセスしたとき、Windows のログイン・ダイアログが表示される

このような問題のほとんどは、外部認証プラグインや Microsoft Active Directory 用の同期プロファイルの構成が間違っていることに関連しています。Microsoft Active Directory の統合の問題に関する詳細なトラブルシューティング情報は、『Oracle Identity Management 統合ガイド』を参照してください。

また、OracleAS Single Sign-On と Windows のネイティブ認証の一般的なトラブルシューティングのヒントは、Oracle *MetaLink*（`http://metalink.oracle.com`）の Note 283268.1 も参照してください。

Windows での認証後にユーザーが URL にアクセスできない

Windows 環境で Microsoft Active Directory を使用した認証が可能なユーザーが、OracleAS Single Sign-On 経由で URL にアクセスできません。次のエラー・メッセージのいずれかが表示される場合があります。

アクセスは禁止されています

`HTTP` エラー・コード `403`

`Windows` のネイティブ認証に失敗しました。管理者に連絡してください。

障害

これは、次のいずれかの問題が原因と考えられます。

- Oracle Internet Directory に必要なユーザー・エントリが見つからないため、ユーザーは OracleAS Single Sign-On 経由で URL にアクセスできません。
- ユーザーがローカル・ドメインにのみログインしている場合（ユーザーが管理者としてローカル・マシンにログインした場合など）、OracleAS Single Sign-On はそのユーザーの社内 ID を認識しません。

解決策

次の手順を実行して、Microsoft Active Directory と Oracle Internet Directory の両方でユーザー ID が認識されるようにしてください。

1. Microsoft Active Directory で認識されるユーザー ID で、Windows のデスクトップ環境にログインします。このとき、実在のユーザーとしてログインし、（ローカル・マシンだけでなく）Microsoft Active Directory のドメインにログインするようにしてください。
2. ユーザー ID が Microsoft Active Directory のドメインで有効であることを確認したら、そのユーザー ID が Oracle Internet Directory に存在することを確認します。そのユーザーが存在しない場合は、`oditest` ユーティリティを使用して、Microsoft Active Directory の同期プロファイルに問題がないかトラブルシューティングを行います。

   `oditest` ユーティリティの詳細は、『Oracle Identity Management 統合ガイド』のトラブルシューティングに関する付録を参照してください。

3. そのユーザーが Oracle Internet Directory に存在する場合は、Microsoft Active Directory から Oracle Internet Directory に、そのユーザーの Kerberos プリンシパル属性が正しく同期化されているかどうかを確認します。`oditest` ユーティリティと `diptester` ユーティリティを使用すると、Microsoft Active Directory の同期プロファイルに問題がないかトラブルシューティングができます。

   `oditest` ユーティリティと `diptester` ユーティリティの詳細は、『Oracle Identity Management 統合ガイド』のトラブルシューティングに関する付録を参照してください。

Windows で認証済のユーザーをブラウザで認証できない

Windows 環境で Microsoft Active Directory によって認証済のユーザーが、ユーザーのブラウザで認証されません。

障害

ユーザーのブラウザが Windows の Kerberos 認証をサポートしていないか、正しく構成されていません。

解決策

ブラウザで Windows のネイティブ認証を使用するための構成手順については、『Oracle Identity Management 統合ガイド』を参照してください。

資格証明が見つからないというエラーで Single Sign-On Server の起動に失敗する

Single Sign-On Server の起動に失敗し、その起動ログ・ファイル `ORACLE_HOME/opmn/logs/OC4J~OC4J_SECURITY~default_island~1` に次のエラー・メッセージが記録されます。

資格証明が見つかりません。

障害

パラメータ `kerberos-servicename` が正しく構成されていない可能性があります。

解決策

詳細は、『Oracle Identity Management 統合ガイド』を参照してください。

Single Sign-On Server に内部サーバー・エラーが表示される

Windows のネイティブ認証を使用したとき、Single Sign-On Server に次のエラー・メッセージが表示されます。

内部サーバー・エラーです。管理者に連絡してください。

障害

OracleAS Single Sign-On 中間層に Windows のネイティブ認証が正しく構成されていません。

解決策

詳細は、『Oracle Identity Management 統合ガイド』を参照してください。

Single Sign-On ユーザーを KDC で認証できない

Windows のネイティブ認証を使用するユーザーに、次のエラーが表示されます。

KDC で認証できませんでした。

障害

Kerberos 構成ファイル krb5.conf 内で、レルム名が正しく構成されていません。

解決策

詳細は、『Oracle Identity Management 統合ガイド』を参照してください。

パートナ・アプリケーションにアクセスしたとき、Windows のログイン・ダイアログが表示される

Windows 環境で Microsoft Active Directory を使用して認証済のユーザーが OracleAS Single Sign-On パートナ・アプリケーションにアクセスしようとすると、Windows のログイン・ダイアログ（ユーザー名、パスワード、ドメインのプロンプト）が表示されます。

障害

対応するユーザー・エントリが Oracle Internet Directory に見つからないため、Single Sign-On Server は Kerberos トークンを認証できません。

解決策

Oracle Internet Directory にユーザー・エントリを追加します。これには、Microsoft Active Directory から Oracle Internet Directory にユーザー・エントリを同期化する方法をお薦めします。oditest ユーティリティと diptester ユーティリティを使用すると、Microsoft Active Directory の同期プロファイルに問題がないかトラブルシューティングができます。

oditest ユーティリティと diptester ユーティリティの詳細は、『Oracle Identity Management 統合ガイド』のトラブルシューティングに関する付録を参照してください。

パスワード・ポリシー・エラーの障害と解決策

パスワード・ポリシーに関連して、次の問題が発生する場合があります。

- 無効にされたユーザーがまだログインできる
- 無効にされたユーザーの画面に、アカウント無効ではなく認証失敗のメッセージが表示される
- ユーザーがログイン時にパスワードの期限切れのメッセージを受け取る
- コマンドライン・ツールでパスワードの期限切れメッセージが表示されない

無効にされたユーザーがまだログインできる

管理者が Oracle Internet Directory の orclIsEnabled 属性を使用してユーザーを無効にしましたが、ユーザーはまだログインできます。

障害

orclIsEnabled 属性が正しくありません。

解決策

コマンドラインでユーザーとして ldapbind を実行します。これによりアカウント無効のエラーが発生する場合は、属性値を再度入力します。

無効にされたユーザーの画面に、アカウント無効ではなく認証失敗のメッセージが表示される

障害

管理者が Oracle Internet Directory の `orclIsEnabled` 属性を使用してユーザーを無効にしましたが、ユーザーはアカウント無効のエラーではなく、認証失敗のエラーを受け取ります。

解決策

ありません。これは予期されている結果です。ユーザーのアカウントが無効になった場合、ユーザーは認証失敗のエラーを受け取ります。

ユーザーがログイン時にパスワードの期限切れのメッセージを受け取る

障害

ユーザー・パスワードの期限が切れています。

解決策

管理者はパスワードを再設定する必要があります。管理者はディレクトリでパスワードの期限切れ警告を有効にできます。このような警告によって、パスワードの期限切れ前にユーザーはパスワードを変更できます。

コマンドライン・ツールでパスワードの期限切れメッセージが表示されない

障害

ユーザーが Single Sign-On Server にログインしますが、パスワードの有効期限が近づいているため、パスワードの変更が要求されます。しかし、ユーザーがコマンドラインでバインドしようとすると、このメッセージは表示されず、バインドに成功します。

解決策

ありません。コマンドラインのツールを使用すると、特定の拡張ディレクトリ・メッセージが表示されません。LDAP のクライアント側 API でのみこれらのメッセージが表示されます。

OracleAS Single Sign-On の障害の診断

この項では、OracleAS Single Sign-On 環境で発生する障害の診断に役立つ情報を示します。この項の項目は次のとおりです。

- ログ・ファイルの表示
- デバッグ・ログ・レベルの引上げ
- Single Sign-On データベースでのデバッグ・オプションの有効化
- UI 操作に関する LDAP トレースの有効化

注意： OC4J の実行中に、ログ・ファイルを削除または編集しないでください。

ログ・ファイルの表示

次の OracleAS ログ・ファイルには、シングル・サインオン操作についてのデータが記録されます。

- Single Sign-On Server 用の一般ログ・ファイル :

  ORACLE_HOME/sso/log/ssoServer.log

  使用上の注意 :

  Single Sign-On Server は、すべてのエラーをこのファイルに書き込みます。デフォルトの保存場所を変更するには、*ORACLE_HOME*/sso/conf/policy.properties を編集します。このファイルでは、ロギング・レベルも変更できます。

- Single Sign-On Server の起動エラー・ログ :

  ORACLE_HOME/opmn/logs/OC4J~OC4J_SECURITY~default_island~1

  使用上の注意 :

  OC4J で生成されたこのファイルには、Single Sign-On Server 起動時のすべてのエラーが記録されます。OC4J_SECURITY インスタンスの起動時に opmnctl コマンドがハングアップしたり、コマンドラインでエラーがレポートされた場合には、このファイルでエラー・メッセージをチェックします。

- Web アプリケーション・ログ :

  ORACLE_HOME/j2ee/OC4J_SECURITY/application-deployments/sso/
  OC4J_SECURITY_default_island_1/application.log

  使用上の注意 :

  このファイルには、OC4J アプリケーションのランタイム・エラーが記録されます。

- OC4J サーブレット・アクセス・ログ :

  ORACLE_HOME/j2ee/OC4J_SECURITY/log/OC4J_SECURITY_default_island_1/
  default-web-access.log

  使用上の注意 :

  これも OC4J で生成されたファイルです。シングル・サインオンでのサーブレット・アクセス・ログが記録されます。ファイルをチェックして、認証サーブレットで認証リクエストが受け取られたかどうかを確認します。

- Oracle HTTP Server のエラー・ログ :

  ORACLE_HOME/Apache/Apache/logs/error_log

  使用上の注意 :

  Oracle HTTP Server が複数のログ・ファイルを切り替えながら使用するよう構成されている場合、それぞれのファイルにはタイムスタンプが追加されます。このタイムスタンプで、最新のログ・ファイルを判別してください。

- Oracle HTTP Server のアクセス・ログ :

  ORACLE_HOME/Apache/Apache/logs/access_lo

  使用上の注意 :

  Oracle HTTP Server が複数のログ・ファイルを切り替えながら使用するよう構成されている場合、それぞれのファイルにはタイムスタンプが追加されます。このタイムスタンプで、最新のログ・ファイルを判別してください。

デバッグ・ログ・レベルの引上げ

OracleAS Single Sign-On には 4 つのデバッグ・レベルがあります。デバッグ・レベル（昇順）と詳細を次に示します。

- ERROR: エラーのみのログ
- WARN: エラー、警告メッセージのログ
- INFO: 情報メッセージ（現在の日付、時間など）、エラー、警告メッセージのログ
- DEBUG: プログラム実行の詳細、エラー、警告メッセージ、情報メッセージのログ

デバッグの途中で、デバッグ・レベルを DEBUG などに上げる必要性が生じる場合があります。これには、ファイル *ORACLE_HOME*/sso/conf/policy.properties を変更します。

デバッグ・レベルを変更したら、OC4J_SECURITY インスタンスを再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

Single Sign-On データベースでのデバッグ・オプションの有効化

外部アプリケーションへのアクセスに使用する mod_plsql コードをデバッグする場合があります。この手順では、Single Sign-On データベースでデバッグを有効にして、詳細ログを表示する必要があります。パートナ・アプリケーションのデバッグにはこの手順を使用できないので注意してください。パートナ・アプリケーションのデバッグ情報は、*ORACLE_HOME*/sso/log/ssoServer.log にのみ格納されます。

mod_plsql デバッグを有効にするには、ORASSO スキーマにログインして、ssolsdbg.sql スクリプトを実行します。スキーマのパスワードの取得方法は、付録 B を参照してください。スクリプト内にあるコメント行をコメント解除してから、スクリプトを実行します。スクリプトのコピーは、*ORACLE_HOME*/sso/admin/plsql/sso にあります。

次にスクリプトを示します。

```
set scan off;
set feedback ON;
set verify ON;
set pages 50000;
set serveroutput ON;

CREATE OR replace PROCEDURE debug_print (str VARCHAR2) AS
BEGIN


   INSERT INTO wwsso_log$ VALUES (wwsso_log_pk_seq.nextval,
      substr(str, 1, 1000),
      sysdate, dbms_session.unique_session_id);

   commit;


END debug_print;
/

show errors;
```

デバッグ・ログを問い合せるには、次のコマンドを発行します。

```
SELECT * FROM WWSSO_LOG$ ORDER BY ID;
```

デバッグを無効にするには、ORASSO スキーマにログインして次の PL/SQL スクリプトを作成します。デバッグの終了時にはこの手順を実行する必要があります。この手順を実行しないと、データベース表に不必要なレコードが作成されます。スキーマのパスワードの取得方法は、付録 B を参照してください。

```
set scan off;
set feedback ON;
set verify ON;
set pages 50000;
set serveroutput ON;


CREATE OR replace PROCEDURE debug_print (str VARCHAR2) AS
--  PRAGMA autonomous_transaction;
BEGIN

    null;

END debug_print;
/

show errors;
```

UI 操作に関する LDAP トレースの有効化

シングル・サインオンの管理ページでは、DBMS_LDAP パッケージを使用してディレクトリ操作が実行されます。このような操作に関する詳細は、Single Sign-On データベースのデバッグ・ログで取得できます。ただしエラーを特定するには、クライアント側の LDAP トレースを有効にする必要があります。たとえば、管理者に対するシングル・サインオンのホームページに管理リンクが表示されない原因を調べる場合、LDAP のクライアント側 API によって、エラーが返される正確な位置を特定できます。その後、RDBMS トレース・ファイルでトレース結果を参照できます。

クライアント側のトレースを有効にするには、次の手順を実行します。

1. ORASSO スキーマに debugonldap.sql パッケージをロードして、トレースを有効にします。

   ```
   SQL> connect orasso/password
   ```

   スキーマのパスワードの取得方法は、付録 B を参照してください。

2. 次のスクリプトを実行します。

   ```
   SQL> @debugonldap.sql
   ```

   debugonldap.sql は次のとおりです。

   ```
   set scan off;
   set feedback ON;
   set verify ON;
   set pages 50000;
   set serveroutput ON;

   CREATE OR replace PROCEDURE debug_print (str VARCHAR2) AS
   BEGIN

      dbms_ldap.set_trace_level(65535);

      INSERT INTO wwsso_log$ VALUES
        (wwsso_log_pk_seq.nextval, substr(str, 1, 1000), sysdate,
        dbms_session.unique_session_id);

       commit;
   ```

```
END debug_print;
/

show errors;
```

3. エラーが発生したシングル・サインオン操作またはデバッグが必要なシングル・サインオン操作（ホームページへのログインなど）を、管理者として実行します。
4. RDBMS トレース・ディレクトリにある LDAP クライアント・ログをチェックします。

   このディレクトリの場所を確認するには、ID 管理インフラストラクチャ・データベースに SYS で接続し、次のコマンドを実行します。

   ```
   SQL> show parameter user_dump_dest
   ```

   返された値が、トレース・ファイルが格納されているディレクトリです。目的のディレクトリに移動したら、ファイルのタイムスタンプをチェックして該当するファイルを見つけます。

クライアント側のトレース・ファイルで問題が明らかにならない場合は、クライアント側のトレースを実行してから、サーバー側のトレースを有効にします。サーバー側のトレースを有効にする方法は、『Oracle Internet Directory 管理者ガイド』のロギング、監査および監視に関する章を参照してください。

トレースを無効にするには、次のパッケージをロードして実行します。

```
set scan OFF;
set feedback ON;
set verify ON;
set pages 50000;
set serveroutput ON;

CREATE OR replace PROCEDURE debug_print (str VARCHAR2) AS BEGIN
null;
END debug_print;
/
show errors;
```

OracleAS Single Sign-On のメンテナンス・タスク

この項では、OracleAS Single Sign-On の各種メンテナンス・タスクについて説明します。この項の項目は次のとおりです。

- シングル・サインオン監査レコードの管理
- LDAP 接続キャッシュのリフレッシュ
- Oracle Internet Directory 変更後の OC4J の再起動

シングル・サインオン監査レコードの管理

Single Sign-On Server では、認証の失敗と成功が Oracle Identity Management データベースに記録されます。やがて、監査表 ORASSO.WWSSO_AUDIT_LOG_TABLE_T の領域は一杯になります。このような場合、次のエラー・メッセージがデータベースの警告ログに出力されます。

```
ORA-1654: unable to extend index ORASSO.AUDIT_INDEX1 by 128 in tablespace IAS_META
```

さらに、後続の認証リクエストも失敗します。

ORASSO.WWSSO_AUDIT_LOG_TABLE_T を定期的に監視してください。この表が一杯になったら、バックアップを作成して空領域を作るか、領域を追加します。これは製品固有の内部表です。このため、表のクリーンアップには SQL*Plus を使用できますが、表に基づいたレポート・スクリプトまたは監視スクリプトの構築には、SQL*Plus および他のツールを使用できません。

LDAP 接続キャッシュのリフレッシュ

パフォーマンス上の理由から、Single Sign-On Server は Oracle Internet Directory への接続をキャッシュします。ディレクトリ・サーバーに、スケジューリングした停止またはスケジューリングしていない停止が設定されている場合、Single Sign-On Server では不適切なディレクトリ接続が維持され、ユーザーが外部アプリケーションにアクセスしようとするとディレクトリ設定エラーが発生することがあります。LDAP 接続キャッシュが無効な場合、Oracle HTTP Server を再起動する必要があります。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
```

LDAP 接続キャッシュをリフレッシュする必要があるかどうかを確認するには、次の手順を実行します。

1. Single Sign-On スキーマに接続します。スキーマのパスワードの取得方法は、付録 B を参照してください。
2. 次のコマンドを発行します。

   ```
   SELECT * FROM WWSSO_LOG$
   ```

3. ログ内に次のエラーがある場合は、Oracle HTTP Server を再起動します。

   ```
   'INVALID LDAP CONNECTION CACHE: RESTART ORACLE HTTP SERVER'
   ```

4. WWSSO_LOG$ からエラー・メッセージを削除します。

Oracle Internet Directory 変更後の OC4J の再起動

Oracle Internet Directory 内の値を変更した場合は、その変更内容で Single Sign-On Server を更新する必要があります。たとえば、ディレクトリ内でユーザー、サブスクライバ、またはグループ検索ベースを変更しても、Single Sign-On Server にその旨を通知しなかった場合、変更されたコンテナのユーザーはログインできなくなります。ssoreoid.sql スクリプトによって、ディレクトリの変更内容で Single Sign-On Server が更新されます。スクリプトの実行方法は、第 3 章の「ディレクトリ変更による Single Sign-On Server の更新」の項を参照してください。

スクリプトを実行したら、Single Sign-On Server を再起動する必要があります。

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

GET 以外の認証方式に関する注意事項

mod_osso で保護されたアプリケーションの最初のページには、GET 認証方式を使用する URL を使用してください。POST 方式を使用すると、ログイン時にユーザーが入力したデータが、Single Sign-On Server へのリダイレクト中に失われてしまいます。グローバル・ユーザーの非アクティブのタイムアウトを有効にするかどうかを決める際には、ユーザーはタイムアウトして再度ログインした後でリダイレクトされる点に注意してください。

その他の情報

その他の解決方法は、Oracle *MetaLink*（`http://metalink.oracle.com`）にあります。問題に対する解決方法が見つからない場合は、オラクル社カスタマ・サポート・センターに問い合せてください。

オラクル社カスタマ・サポート・センターによるトラブルシューティングを容易にするために、MetaLink の Note 248870.1 で説明されている手順を実行してください。

関連項目：

- Oracle Application Server のリリース・ノート（Oracle Technology Network（`http://www.oracle.com/technology/documentation/index.html`）で入手可能）

B

Single Sign-On スキーマのパスワードの取得

Single Sign-On スキーマのパスワードは、Oracle Application Server Infrastructure のインストール時にランダム化されます。パスワードを取得するには、コマンドライン・ツール `ldapsearch` または Oracle Directory Manager を使用します。

コマンドラインを使用する場合

ldapsearch でスキーマのパスワードを取得するには、次の構文を使用します。

```
ldapsearch  -h directory_host_name
             -p directory_ssl_port
             -D directory_bind_dn
             -w directory_bind_dn_password
             -b "orclReferenceName=infrastructure_database"
                "orclresourcename=ORASSO"
                orclpasswordattribute
             -u 1
```

次の表は、ldapsearch に渡されるパラメータの定義です。

パラメータ	説明
directory_host_name	ディレクトリ・サーバーのホスト名。
directory_ssl_port	ディレクトリ・サーバーのポート番号。
directory_bind_dn	ディレクトリに対するユーザー認証の識別名。
directory_bind_dn_password	ディレクトリに対するユーザー認証のパスワード。
infrastructure_database	パスワード属性（orclpasswordattribute）が定義されているディレクトリ・エントリの識別名。
-u	ディレクトリ・ポートを SSL ポートにグローバルに変更します。

次に例を示します。

```
ldapsearch -h oid.acme.com
           -p 636
           -D "cn=orcladmin"
           -w welcome1
           -b "orclReferenceName=disco.us.acme.com,cn=IAS Infrastructure
              Databases,cn=IAS,cn=Products,cn=oraclecontext"
              "orclresourcename=ORASSO"
              orclpasswordattribute
           -u 1
```

Oracle Directory Manager の使用方法

Oracle Directory Manager でスキーマのパスワードを取得するには、次の手順を実行します。

1. ツールを起動します。

   ORACLE_HOME/bin/oidadmin

2. **「システム・オブジェクト」**フレームで、次のエントリを続けて開きます。

   - Entry Management
   - cn=OracleContext
   - cn=Products
   - cn=IAS
   - cn=IAS Infrastructure Databases
   - orclReferenceName=*database_service_name_for_infrastructure_database*
   - OrclResourceName=ORASSO

OrclResourceName=ORASSO のタブの orclpasswordattribute のテキスト・ボックスに、スキーマのパスワードが表示されます。

C

policy.properties

この付録で示す policy.properties ファイルは、Single Sign-On Server で必要とされる基本パラメータを含む多目的な構成ファイルです。このファイルは、マルチレベル認証などの拡張機能の実装にも使用されます。

```
# SSO Server policy configurations

################################################################
# Authentication Levels
# ---------------------
# Set the auth levels from lower value to higher value.
# 10 being the lowest authentication level
# The auth level names (on the left hand side) can be changed to
# some other names if desired as long as the change is consistent
# in other usages within the policy file.

NoSecurity = 10
LowSecurity = 20
LowMediumSecurity = 30
MediumSecurity = 40
MediumHighSecurity = 50
HighSecurity = 60


#################################################################
# DefaultAuthLevel
# ----------------
# DefaultAuthLevel entry must have a value assigned.

DefaultAuthLevel = MediumSecurity


#####################################################################
# Authentication plugins
# ----------------------
# Assign a class name that implements SSOServerAuthInterface
# for each auth level referenced.
#
# The Authentication level name must be appended with
#  "_AuthPlugin" keyword.

MediumSecurity_AuthPlugin = oracle.security.sso.server.auth.SSOServerAuth
#####################################################################
# Custom Cookie Provider Class name
# ---------------------------------
# Sample custom cookie tester provider class
# CustomCookie_ProviderPlugin = oracle.security.sso.server.auth.CustomCookieTester
```

```
# Custom Cookie auth level
# ------------------------
# This is a mandatory attribute. If custom cookies are not needed it should
# be set to a higher value than any of the authentication levels used.

CustomCookieAuthLevel = HighSecurity


##########################################################################
# Protected URL configurations
# ----------------------------
# Assign a auth level to each protected (partner) application that is
# participating in SSO. If any of the partner apps are not listed with
# a specific auth level, then the DefaultAuthLevel will be used.
#
# Protected application URL configuration format:
# "Partner Application Root URL" = "AuthenticationLevel"
# host.company.com\:port = AuthLevelName
# NOTE: The required backslash(escape character) before the
# colon (:) character immediately preceding.
# There should be a corresponding auth plugin configured for the
# "AuthenticationLevel" used.
#
# Examples:
# The following example configures a SSO partner application hosted
# on host1.company.com:7777 machine using LowSecurity authentication level.
# This configuration will secure all URLs hosted on this host/port.
# host1.company.com\:7777 = LowSecurity
#
# The following example configures a SSO partner application hosted
# on host2.company.com:7777 machine using MediumSecurity authentication level
# This configuration will secure all URLs hosted on this host/port.
# host2.company.com\:7777 = MediumSecurity


#####################################################################
#SSO Server specfic configurations

# set the cache size in kbytes
#default is 250
cacheSize = -1

#set the minimum number of connections in the connection pool
#default is 5
minConnectionsInPool = 5

#set the maximum number of connections in the connection pool
#default is 150
maxConnectionsInPool = 150

#LDAP and database connection pool timeout in minutes
connectionIdleTimeout = 120

#Debug level {ERROR, WARN, INFO, DEBUG}
# default debug level is set to ERROR
debugLevel = ERROR

#Debug file location
#This is a mandatory property that needs to be passed
#the SSO server. A valid file location should be specified here
debugFile = /private/vshriram/infra1012/sso/log/ssoServer.log
```

```
#Deployment login page link
loginPageUrl = /sso/pages/login.jsp

#Deployment logout page link
logoutPageUrl = /sso/pages/logout.jsp

#Deployment external application login page link
extAppLoginPageUrl = /sso/pages/ealogin.jsp

#Deployment change password page link
chgPasswordPageUrl = /sso/pages/password.jsp

#Wireless login page link
wirelessLoginPageUrl = /wirelesssso/wirelesslogin.jsp
wirelessChgPasswordPageUrl = /wirelesssso/wirelesscpwd.jsp


SASSOAuthnUrl = http\://stads41.us.oracle.com\:/sso/authn
SASSOLogoutUrl = http\://stads41.us.oracle.com\:/sso/jsp/sasso_logout_success.jsp
SASSOAuthLevel = HighSecurity

#SASSO keyfile
SASSOConfigFile = %s_ssoLogOH%/sso/conf/keystore

#SASSO key rollover interval
ROLLOVER_INTERVAL = 600
```

用語集

3DES

「Triple Data Encryption Standard」を参照。

ACI

「アクセス制御項目」を参照。

ACL

「アクセス制御リスト」を参照。

ACP

「アクセス制御ポリシー・ポイント」を参照。

Advanced Encryption Standard（AES）

データ暗号化規格にかわる暗号標準として意図された対称型暗号アルゴリズム。商用および政府データの暗号化に対する、米国連邦情報処理標準（Federal Information Processing Standard: FIPS）となっている。

AES

「Advanced Encryption Standard」を参照。

API

「Application Program Interface」を参照。

Application Program Interface（API）

コンピュータ・アプリケーションと、下位レベルのサービスや機能（オペレーティング・システム、デバイス・ドライバ、他のソフトウェア・アプリケーションなど）との間のインタフェースとなる一連のソフトウェア・ルーチンおよび開発ツール。API は、プログラマがソフトウェア・アプリケーション間に連携を構築する際の基盤となる。たとえば、LDAP 対応のクライアントは、LDAP API で使用可能なプログラム・コールを通して、Oracle Internet Directory 情報にアクセスする。

ASN.1

Abstract Syntax Notation One（ASN.1）は、情報データの構文定義に使用される国際電気通信連合（International Telecommunication Union: ITU）の表記規約である。ASN.1 は、構造化された情報、特に通信メディアを介して送受信される情報の記述に使用される。ASN.1 は、インターネット・プロトコルの仕様において幅広く使用されている。

ASR

「Oracle Database アドバンスト・レプリケーション」を参照。

Basic 認証（basic authentication）

大半のブラウザによってサポートされる**認証**プロトコル。Web サーバーでは、データ通信に使用するエンコード済のユーザー名とパスワードを使用してエンティティが認証される。BASE64 エンコーディングは自由に利用できるデコーディング・ユーティリティを使用して誰でもデコードできるため、Basic 認証は平文認証と呼ばれることもある。エンコーディングと**暗号化**は異なることに注意すること。

BER

「**基本エンコーディング規則**」を参照。

Blowfish

DES にかわる暗号化を迅速に導入する目的で、Bruce Schneier 氏によって 1993 年に開発された**対称型暗号**アルゴリズム。Blowfish は、64 ビット・ブロックと最大 448 ビットの鍵を使用する**ブロック暗号**である。

CA

「**認証局**」を参照。

CA 証明書（CA certificate）

認証局は、自局が発行するすべての証明書を自身の**秘密鍵**で署名する。それに対応する認証局の**公開鍵**は、CA 証明書（またはルート証明書）と呼ばれる証明書の中に含まれる。ブラウザは、CA の秘密鍵によって署名されたメッセージを信頼するには、信頼できるルート証明書リストの中に CA 証明書を保持している必要がある。

CBC

「**暗号ブロック連鎖**」を参照。

Certificate Management Protocol（CMP）

Certificate Management Protocol（CMP）は、証明書の作成と管理に関連するあらゆる局面を取り扱う。CMP では、**認証局**、**登録局**、証明書が発行されたユーザーやアプリケーションなど、**公開鍵インフラストラクチャ**の構成要素間の対話がサポートされる。

Certificate Request Message Format（CRMF）

Certificate Request Message Format（CRMF）は、**X.509** 証明書のライフサイクル管理に関連したメッセージに使用されるフォーマットで、**RFC** 2511 仕様で規定されている。

CMP

「**Certificate Management Protocol**」を参照。

CMS

「**Cryptographic Message Syntax**」を参照。

configset

「**構成設定エントリ**」を参照。

CRL

「**証明書失効リスト**」を参照。

CRMF

「**Certificate Request Message Format**」を参照。

Cryptographic Message Syntax（CMS）

デジタル・メッセージの署名、ダイジェスト、認証および暗号化に使用される構文。**RFC** 3369 で定義される。

dads.conf

データベース・アクセス記述子の構成に使用される、Oracle HTTP Server の構成ファイル。

DAS

「**Oracle Delegated Administration Services**（DAS）」を参照。

Delegated Administration Services

「**Oracle Delegated Administration Services**」を参照。

DER

「**高度なエンコーディング規則**」を参照。

DES

「**データ暗号化規格**」を参照。

DIB

「**ディレクトリ情報ベース**」を参照。

Diffie-Hellman

保護されていないチャネルで通信を行う二者間で共有シークレットを構築することを可能にする公開鍵暗号プロトコル。Diffie-Hellman は 1976 年に公開され、使用可能になった最初の公開鍵暗号システムである。

「**対称型アルゴリズム**」も参照。

Digital Signature Algorithm（DSA）

Digital Signature Standard（DSS）の一部として使用されている**非対称型アルゴリズム**。DSA は暗号化には使用できず、デジタル署名にのみ適用される。このアルゴリズムは 1 組の大きな数を生成することで、署名者の認証と、結果として添付されているデータの整合性を保証する。DSA は、デジタル署名の生成と検証の両方に使用される。

「**Elliptic Curve Digital Signature Algorithm**」も参照。

Directory Integration and Provisioning Server

Oracle Directory Integration Platform 環境で、Oracle Internet Directory と**接続ディレクトリ**との間でデータの同期化を実行するサーバー。

Directory Manager

「**Oracle Directory Manager**」を参照。

DIS

「**Directory Integration and Provisioning Server**」を参照。

DIT

「**ディレクトリ情報ツリー**」を参照。

DN

「**識別名**」を参照。

Document Type Definition（DTD）

特定の **XML** ドキュメントで有効なタグおよびタグの順序に関する制約を指定するドキュメント。DTD は、XML の親言語である Simple Generalized Markup Language（SGML）の規則に準拠する。

DRG

「**ディレクトリ・レプリケーション・グループ**」を参照。

DSA

「**Digital Signature Algorithm**」または「**ディレクトリ・システム・エージェント**」を参照。

DSE

「**ディレクトリ固有のエントリ**」を参照。

DTD

「**Document Type Definition**」を参照。

ECC

「**Elliptic Curve Cryptography**」を参照。

ECDSA

「**Elliptic Curve Digital Signature Algorithm**」を参照。

EJB

「**Enterprise JavaBeans（EJB）**」を参照。

Elliptic Curve Cryptography（ECC）

RSA にかわる暗号化システムで、大きな数の因数分解よりも楕円曲線の離散対数問題の解決の方が困難であることに基づいている。ECC は、Certicom 社によって開発および商品化され、ワイヤレス・デバイスや PC カードのような、制限された演算能力の中で高速が要求される環境に特に適している。ECC は、同じ鍵サイズ（ビット単位）であれば、RSA よりもセキュリティが高い（鍵なしでの復号化がより困難である）。

Elliptic Curve Digital Signature Algorithm（ECDSA）

Elliptic Curve Digital Signature Algorithm（ECDSA）は、**Digital Signature Algorithm** 標準の楕円曲線版である。RSA 的な方式と比較した ECDSA の利点は、鍵の長さが短く、署名と復号化が高速なことである。たとえば、160 ビットの ECC 鍵は、1024 ビットの RSA 鍵に相当するセキュリティを実現する。210 ビットの ECC 鍵は 2048 ビットの RSA 鍵に相当し、セキュリティ・レベルが高くなるにつれて、この利点が顕著になる。

Enterprise JavaBeans（EJB）

Sun 社によって開発された Java API で、複数層からなるクライアント / サーバー・システムのコンポーネント・アーキテクチャを定義する。EJB システムは Java で記述されているため、プラットフォームに依存しない。オブジェクト指向に基づき、既存システムへの実装には、再コンパイルや構成をほとんどまたはまったく必要としない。

Enterprise Manager

「**Oracle Enterprise Manager**」を参照。

Federal Information Processing Standards（FIPS）

米国商務省の標準技術局（National Institute of Standards and Technology: NIST）によって発行されている情報処理標準。

FIM

「**フェデレーテッド ID 管理**」を参照。

FIPS

「**Federal Information Processing Standards**」を参照。

GET

ログイン資格証明がログイン URL の一部として送信される認証方式。

Global Unique Identifier（GUID）

エントリがディレクトリに追加されると、システムで生成され、エントリに挿入される識別子。マルチマスター・レプリケート環境で、DN ではなく GUID がエントリを一意に識別する。エントリの GUID をユーザーが変更することはできない。

GUID

「**Global Unique Identifier**」を参照。

Hashed Message Authentication Code（HMAC）

ハッシュ関数の 1 つの技法で、秘密のハッシュ関数結果の作成に使用される。HMAC は、MD5 や SHA などの既存のハッシュ関数を強化し、Transport Layer Security（TLS）で使用される。

HMAC

「**Hashed Message Authentication Code**」を参照。

HTTP

Hyper Text Transfer Protocol の略称。Web ブラウザとサーバー間で、ドキュメントのリクエストとその内容の転送に使用されるプロトコル。この仕様は、World Wide Web Consortium によって維持および開発される。

HTTP Server

「**Oracle HTTP Server**」を参照。

httpd.conf

Oracle HTTP Server の構成に使用されるファイル。

HTTP リダイレクト・プロファイル（HTTP Redirect Profile）

リクエストされたリソースが異なる URL にあることを示す**フェデレーション**・プロファイル。

iASAdmins

Oracle Application Server で、ユーザーとグループの管理に責任を持つ管理グループ。OracleAS Single Sign-On 管理者は、iASAdmins グループのメンバーである。

ID 管理（identity management）

組織でネットワーク・エンティティのセキュリティ・ライフ・サイクル全体を管理するプロセス。通常、組織のアプリケーション・ユーザーの管理を指す。セキュリティ・ライフ・サイクルの手順には、アカウント作成、一時停止、権限変更およびアカウント削除が含まれる。管理されるネットワーク・エンティティには、デバイス、プロセス、アプリケーション、またはネットワーク環境で対話する必要があるその他のすべてのものが含まれる。ID 管理プロセスで管理されるエンティティには、組織外のユーザー（顧客、取引先、Web サービスなど）も含まれる。

ID 管理インフラストラクチャ・データベース（identity management infrastructure database）

OracleAS Single Sign-On と Oracle Internet Directory のデータを保持するデータベース。

ID 管理レルム（identity management realm）

すべてが同じ管理ポリシーによって管理されている識別情報の集合。企業では、イントラネットへのアクセス権限を所有しているすべての従業員は 1 つのレルムに属し、企業の公開アプリケーションにアクセスするすべての外部ユーザーは別のレルムに属する。ID 管理レルムは、特別な**オブジェクト・クラス**が関連付けられた特定の**エントリ**でディレクトリ内に表される。

ID 管理レルム固有の Oracle コンテキスト（identity management realm-specific Oracle Context）

各 ID 管理レルムに含まれた Oracle コンテキスト。これには、次の情報が格納されている。

- ID 管理レルムのユーザー・ネーミング・ポリシー（ユーザーに名前を付け、配置する方法）
- 必須認証属性
- ID 管理レルム内のグループの位置
- ID 管理レルムに対する権限の割当て（レルムにユーザーを追加する権限の割当てなど）
- レルムに関するアプリケーション固有のデータ（認可など）

ID フェデレーション（identity federation）

ある**トラスト・サークル**内の 1 つ以上の ID プロバイダまたはサービス・プロバイダで**プリンシパル**が保持する可能性がある複数のアカウントのリンク。

このリンクがないと、取引がある隔離された複数のユーザー・アカウント（ローカル ID）を連携（フェデレート）するとき、2 つのエンティティ間の関係（多数のサービス・プロバイダと ID プロバイダを構成する関連性）を作成する必要がある。

「**ID プロバイダ**」および「**サービス・プロバイダ**」も参照。

ID プロバイダ（identity provider）

OSFS でサポートされている **ID フェデレーション**・プロトコルに定義された 3 つの主要な役割の 1 つ。残りの 2 つは、**サービス・プロバイダ**および**プリンシパル**。ID プロバイダは、ある**トラスト・サークル**内にある一連の ID を管理および認証する。

サービス・プロバイダは、プリンシパルの ID の ID プロバイダ認証に基づいて、サービスまたは商品をプリンシパルに提供する。

ID プロバイダは、他のサービス・プロバイダの提携を促すようなビジネス・サービスを提供するサービス・プロバイダ。通常、ID プロバイダがプリンシパルの ID を認証およびアサートする。

Internet Directory

「**Oracle Internet Directory**」を参照。

Internet Engineering Task Force（IETF）

新しいインターネット標準仕様の開発に従事する主要機関。インターネット・アーキテクチャおよびインターネットの円滑な操作の発展に関わるネットワーク設計者、運営者、ベンダーおよび研究者による国際的な団体である。

Internet Message Access Protocol（IMAP）

プロトコルの 1 種。クライアントは、このプロトコルを使用して、サーバー上の電子メール・メッセージに対するアクセスおよび操作を行う。リモートのメッセージ・フォルダ（メールボックスとも呼ばれる）を、ローカルのメールボックスと機能的に同じ方法で操作できる。

J2EE

「**Java 2 Platform, Enterprise Edition**」を参照。

Java 2 Platform, Enterprise Edition（J2EE）

Sun 社によって定義された、エンタープライズ・アプリケーションを開発および配置するための環境。J2EE プラットフォームは、複数層にわたる Web ベース・アプリケーションを開発する機能を提供するサービス、Application Program Interface（API）およびプロトコルのセットで構成される。

JavaServer Pages（JSP）

JavaServer Pages（JSP）はサーバー側のテクノロジで、Sun 社によって開発された Java サーブレット・テクノロジに対する拡張である。JSP には、HTML コードと連携して動作する動的なスクリプティング機能があり、それによってページ・ロジックが静的要素（ページの設計と表示）から分離される。Java ソース・コードとその拡張機能が HTML ページに埋め込まれることで、HTML がより機能的になり、データベースへの動的な問合せなどに使用される。

JSP

「**JavaServer Pages**」を参照。

LDAP

「**Lightweight Directory Access Protocol**」を参照。

LDAP Data Interchange Format（LDIF）

システム間でディレクトリ・データを交換するためのテキストベースの共通フォーマット。LDAP コマンドライン・ユーティリティに使用する入力ファイルをフォーマットするための一連の規格。

LDAP 接続キャッシュ（LDAP connection cache）

スループットを向上させるために、OracleAS Single Sign-On サーバーが、Oracle Internet Directory への接続をキャッシュして再利用すること。

LDIF

「**LDAP Data Interchange Format**」を参照。

Liberty Alliance

Liberty Alliance Project は、世界中の企業、非営利団体および非政府機関からなるコンソーシアムである。このコンソーシアムは、現在および将来のネットワーク・デバイスでサポートされる ID ベースの Web および**フェデレーテッド ID 管理**のオープン・スタンダードの開発に従事する。

Liberty ID-FF

Liberty Identity Federation Framework の略称。Web ベースの**シングル・サインオン**アーキテクチャにフェデレーテッド ID を提供する。

Lightweight Directory Access Protocol（LDAP）

ディレクトリ内の情報にアクセスするための 1 組のプロトコル。LDAP では、あらゆるタイプのインターネット・アクセスに必要な TCP/IP がサポートされる。また、その設計規則のフレームワークによって、Oracle Internet Directory などの業界標準のディレクトリ製品がサポートされる。これは **X.500** 標準の簡易版であるため、LDAP は X.500 Light と呼ばれることもある。

MAC

「**メッセージ認証コード**」を参照。

MD2

Message Digest Two の略称。メッセージ・ダイジェストを作成する**ハッシュ関数**。このアルゴリズムは入力テキストを処理し、元のメッセージに対して固有でデータの整合性検証に使用できる 128 ビットの**メッセージ・ダイジェスト**を作成する。MD2 は、RSA Security 社の Ron Rivest 氏によって開発され、スマート・カードなどの、メモリーが限られるシステムでの使用が意図されている。

MD4

Message Digest Four の略称。**MD2** と類似するが、ソフトウェアでの高速処理に特化して設計されている。

MD5

Message Digest Five の略称。メッセージ・ダイジェストを作成する**ハッシュ関数**。このアルゴリズムは入力テキストを処理し、元のメッセージに対して固有でデータの整合性検証に使用できる 128 ビットの**メッセージ・ダイジェスト**を作成する。MD5 は、**MD4** の潜在的な脆弱さが報告された後、Ron Rivest 氏によって開発された。MD5 は MD4 と類似するが、元のデータに対してより多くの処理を行うため速度は遅い。

MDS

「**マスター定義サイト**」を参照。

mod_osso

Oracle HTTP Server 上のモジュール。これにより、ユーザーが一度 OracleAS Single Sign-On Server にログインすると、OracleAS Single Sign-On で保護されるアプリケーションがユーザー名とパスワードのかわりに HTTP ヘッダーを受け取ることができる。これらのヘッダーの値は、**mod_osso Cookie** に格納される。

mod_osso Cookie

HTTP Server に格納されるユーザー・データ。Cookie はユーザーの認証時に作成される。同じユーザーが別のアプリケーションをリクエストした場合、Web サーバーは mod_osso Cookie の情報を使用してアプリケーションにユーザーをログインさせる。この機能によって、サーバーのレスポンス時間が短縮される。

mod_proxy

Oracle HTTP Server のモジュール。これにより、**mod_osso** を使用して、レガシー・アプリケーションまたは**外部アプリケーション**へのシングル・サインオンを有効にできる。

MTS

「**共有サーバー**」を参照。

Net Services

「**Oracle Net Services**」を参照。

OASIS

Organization for the Advancement of Structured Information Standards の略称。E-Business 標準の開発、策定および採用を推進する国際的な非営利組織。

OC4J

「**Oracle Containers for J2EE**」を参照。

OCA

「**Oracle Certificate Authority**」を参照。

OCI

「**Oracle Call Interface**」を参照。

OCSP

「**Online Certificate Status Protocol**」を参照。

OEM

「**Oracle Enterprise Manager**」を参照。

OID

「**Oracle Internet Directory**」を参照。

OID 制御ユーティリティ（OID Control Utility）

サーバーの起動と停止のコマンドを発行するコマンドライン・ツール。コマンドは、**OID モニター**のプロセスによって解析され、実行される。

OID データベース・パスワード・ユーティリティ（OID Database Password Utility）

Oracle Internet Directory が Oracle Database に接続するときのパスワードの変更に使用されるユーティリティ。

OID モニター（OID Monitor）

Oracle Internet Directory サーバー・プロセスの開始、監視および終了を実行する Oracle Internet Directory のコンポーネント。レプリケーション・サーバー（インストールされている場合）および Oracle Directory Integration Platform サーバーの制御も行う。

Online Certificate Status Protocol（OCSP）

デジタル証明書の有効性確認において、広く使用されている 2 つの方式のうちの 1 つ。もう 1 つの方式の**証明書失効リスト**は OCSP よりも古く、使用シナリオによっては OCSP に置き換えられている。OCSP 仕様は **RFC** 2560 で規定されている。

Oracle Application Server Single Sign-On

OracleAS Single Sign-On は、複数のアプリケーション（経費報告、電子メール、福利厚生情報など）に対する安全なログインを実現するプログラム・ロジックで構成される。これらのアプリケーションには、**パートナ・アプリケーション**と**外部アプリケーション**の 2 つのフォームがある。いずれの場合も、一度の認証で複数のアプリケーションにアクセスできる。

Oracle Call Interface（OCI）

Application Program Interface（API）の 1 つ。これにより、第三世代言語のネイティブ・プロシージャやファンクション・コールを使用して、Oracle Database サーバーにアクセスし、SQL 文の実行のすべての段階を制御するアプリケーションを作成できる。

Oracle Certificate Authority

Oracle Application Server 環境内で使用される**認証局**。OracleAS Certificate Authority では、証明書の格納リポジトリとして Oracle Internet Directory が使用される。OracleAS Certificate Authority を OracleAS Single Sign-On および Oracle Internet Directory と統合することで、これらに依存するアプリケーションに対する透過的な証明書プロビジョニング・メカニズムが実現される。Oracle Internet Directory にプロビジョニングされ OracleAS Single Sign-On で認証されるユーザーは、OracleAS Certificate Authority にデジタル証明書をリクエストできる。

Oracle CMS

IETF **Cryptographic Message Syntax** プロトコルのオラクル社による実装。CMS は、セキュアなメッセージ・エンベロープを実現するデータ保護方式を定義する。

Oracle Containers for J2EE（OC4J）

Java 2 Platform, Enterprise Edition 用の軽量でスケーラブルなコンテナ。

Oracle Crypto

コアな暗号化アルゴリズムを提供する Pure Java ライブラリ。

Oracle Database アドバンスト・レプリケーション（Oracle Database Advanced Replication）

2 つの Oracle データベース間で、データベースの表を継続的に同期化できる Oracle Database の機能。

Oracle Delegated Administration Services

Oracle Delegated Administration Services ユニットと呼ばれる個々の事前定義済サービスのセットで、ユーザーのかわりにディレクトリ操作を実行する。Oracle Internet Directory セルフ・サービス・コンソールによって、Oracle Internet Directory を使用する Oracle アプリケーションおよびサード・パーティ・アプリケーションの両方の管理ソリューションを容易に開発および配布できる。

Oracle Directory Integration and Provisioning

インタフェースとサービスの集合で、Oracle Internet Directory といくつかの関係するプラグインやコネクタを使用して複数のディレクトリを統合する。外部のユーザー・リポジトリが使用されている場合に、そこから Oracle 製品への認証を可能にする Oracle Internet Directory の機能。

Oracle Directory Integration and Provisioning Server

Oracle Directory Integration Platform 環境で、Oracle Internet Directory の変更イベントを監視し、**ディレクトリ統合プロファイル**の情報に基づいてアクションを行うデーモン・プロセス。

Oracle Directory Integration Platform

Oracle Internet Directory のコンポーネントの 1 つ。Oracle Internet Directory のような中央 LDAP ディレクトリの周囲のアプリケーションを統合するために開発されたフレームワーク。

Oracle Directory Manager

Oracle Internet Directory を管理するための、Graphical User Interface（GUI）を備えた Java ベースのツール。

Oracle Enterprise Manager

Oracle 製品の 1 つ。グラフィカルなコンソール、エージェント、標準的なサービスおよびツールを組合せ、Oracle 製品を管理するための統合された包括的なシステム管理プラットフォームを提供する。

Oracle HTTP Server

Hypertext Transfer Protocol（HTTP）を使用する、Web トランザクションを処理するソフトウェア。オラクル社では、Apache Group が開発した HTTP ソフトウェアを使用する。

Oracle Identity Management

すべての企業識別情報および企業内の様々なアプリケーションへのアクセスを集中的かつ安全に管理するための配置を可能にするインフラストラクチャ。

Oracle Internet Directory

分散ユーザーやネットワーク・リソースに関する情報の検索を可能にする、一般的な用途のディレクトリ・サービス。**Lightweight Directory Access Protocol** バージョン 3 と Oracle Database の高度のパフォーマンス、スケーラビリティ、耐久性および可用性を組み合せたもの。

Oracle Liberty SDK

Liberty Alliance Project 仕様のオラクル社による実装。サード・パーティ製の Liberty 準拠アプリケーション間で、フェデレーテッド・シングル・サインオンを可能にする。

Oracle Net Services

Oracle のネットワーク製品ファミリの基礎。Oracle Net Services を使用すると、サービスやアプリケーションを異なるコンピュータに配置して通信できる。Oracle Net Services の主な機能には、ネットワーク・セッションの確立およびクライアント・アプリケーションとサーバー間のデータ転送がある。Oracle Net Services は、ネットワーク上の各コンピュータに配置される。ネットワーク・セッションの確立後は、Oracle Net Services はクライアントとサーバーのためのデータ伝達手段として機能する。

Oracle PKI SDK

公開鍵インフラストラクチャ実装内で必要なセキュリティ・プロトコルのオラクル社による実装。

Oracle PKI 証明書使用条件（Oracle PKI certificate usages）

証明書でサポートされる Oracle アプリケーション・タイプを定義する。

Oracle S/MIME

Internet Engineering Task Force が発行するセキュアな電子メールに関する **Secure/Multipurpose Internet Mail Extension** 仕様のオラクル社による実装。

Oracle SAML

異種システムおよびアプリケーション間でセキュリティ資格証明を、XML ベースのフォーマットで交換するためのフレームワークを提供する。**Security Assertions Markup Language** に関する **OASIS** 仕様に基づく。

Oracle Security Engine

X.509 ベースの証明書管理機能を組み込んだ、Oracle Crypto の拡張。Oracle Security Engine は、Oracle Crypto のスーパーセットである。

Oracle Wallet Manager

セキュリティ管理者が、クライアントとサーバー上での公開鍵セキュリティ資格証明の管理に使用する Java ベースのアプリケーション。

『Oracle Advanced Security 管理者ガイド』も参照。

Oracle Web Services Security

既存のセキュリティ・テクノロジを使用して認証および認可を行うためのフレームワークを提供する。Web サービス・セキュリティに関する **OASIS** 仕様に基づく。

Oracle XML Security

XML 暗号化および XML 署名に関する W3C 仕様のオラクル社による実装。

OracleAS Portal

ファイル、イメージ、アプリケーションおよび Web サイトを統合するメカニズムを提供する、OracleAS Single Sign-On の**パートナ・アプリケーション**。「外部アプリケーション」ポートレットで、外部アプリケーションにアクセスできる。

Oracle コンテキスト（Oracle Context）

「**ID 管理レルム固有の Oracle コンテキスト**」および「**ルート Oracle コンテキスト**」を参照。

OWM

「**Oracle Wallet Manager**」を参照。

peer-to-peer レプリケーション（peer-to-peer replication）

マルチマスター・レプリケーションまたは n-way レプリケーションとも呼ばれる。同等に機能する複数サイトがレプリケートされたデータのグループを管理できるようにするレプリケーションのタイプ。このようなレプリケーション環境では、各ノードはサプライヤ・ノードであると同時にコンシューマ・ノードであり、各ノードでディレクトリ全体がレプリケートされる。

PKCS#1

公開鍵暗号規格（PKCS）とは、RSA Laboratories によって策定された仕様のこと。PKCS#1 は、RSA アルゴリズムをベースとした公開鍵暗号の実装に関する推奨事項を定めている。この内容には、暗号化の基本から、暗号化方式、署名方式、鍵の表記や各種方式の識別に使用する ASN.1 構文までが含まれる。

PKCS#10

公開鍵暗号規格（PKCS）とは、RSA Laboratories によって策定された仕様のこと。PKCS #10 は、公開鍵、名前および一連の可能な属性の証明リクエストに関する構文を定めている。

PKCS#12

公開鍵暗号規格（PKCS）とは、RSA Laboratories によって策定された仕様のこと。PKCS #12 は、個人の識別情報（秘密鍵、証明書、その他の秘密情報、拡張項目など）の伝送に関する構文を定めている。この標準をサポートするシステム（ブラウザやオペレーティング・システムなど）を使用するユーザーは、主に **Wallet** と呼ばれるフォーマットによって、1 組の個人用識別情報をインポート、エクスポートおよび利用できる。

PKCS#5

公開鍵暗号規格（PKCS）とは、RSA Laboratories によって策定された仕様のこと。PKCS#5 は、パスワードをベースとした暗号化の実装に関する推奨事項を定めている。

PKCS#7

公開鍵暗号規格（PKCS）とは、RSA Laboratories によって策定された仕様のこと。PKCS #7 は、デジタル署名やデジタル・エンベロープなどの、暗号化が適用されるデータに関する汎用構文を定めている。

PKCS#8

公開鍵暗号規格（PKCS）とは、RSA Laboratories によって策定された仕様のこと。PKCS #8 は、公開鍵と秘密鍵の対応付けアルゴリズムや一連の属性などの、秘密鍵情報に関する構文を定めている。この標準は、暗号化された秘密鍵に関する構文も定めている。

PKI

「**公開鍵インフラストラクチャ**」を参照。

point-to-point レプリケーション（point-to-point replication）

ファンアウト・レプリケーション（fan-out replication）とも呼ばれる。サプライヤがコンシューマに直接レプリケートするレプリケーションのタイプ。コンシューマは 1 つ以上の他のコンシューマにレプリケートできる。レプリケーションには、完全レプリケーションと部分レプリケーションがある。

policy.properties

シングル・サインオン・サーバーに必要な基本パラメータが含まれる、Oracle Application Server Single Sign-On の多目的構成ファイル。OracleAS Single Sign-On が持つマルチレベル認証などの高度な機能を構成する際にも使用される。

POSIX

Portable Operating System Interface for UNIX の略称。アプリケーションのソース・コードの記述方法を決めることで、異なるオペレーティング・システム間でのアプリケーションの移植を可能にするプログラミング・インタフェース標準のセット。この標準セットは、**Internet Engineering Task Force** によって開発されている。

POST プロファイル（POST Profile）

ログイン資格証明がログイン・フォーム本体内で送信される**認証**方式。

Project Liberty

「**Liberty Alliance**」を参照。

RC2

Rivest Cipher Two の略称。RSA Security 社の Ronald Rivest 氏によって開発された 64 ビット**ブロック暗号**で、**データ暗号化規格**に置き換える目的で設計された。

RC4

Rivest Cipher Four の略称。RSA Security 社の Ronald Rivest 氏によって開発された**ストリーム暗号**。RC4 では、最大 1024 ビットの可変長の鍵を使用できる。RC4 は、**Secure Sockets Layer** プロトコルを使用する Web サイト間で通信を暗号化することによってデータ伝送を保護する際に、最も幅広く使用されている。

RDN

「**相対識別名**」を参照。

RFC

Internet Request For Comments とも呼ばれる。インターネット関連のプロトコルとポリシーの定義が記述されたドキュメント。Internet Engineering Task Force（IETF）が、新しい標準の討議、開発および構築を進めている。標準は、RFC という頭字語とリファレンス番号を使用して公開される。たとえば、電子メールの公式標準は RFC 822 である。

RSA

公開鍵暗号アルゴリズムの名前で、その考案者（Rivest、Shamir および Adelman の 3 氏）から名付けられた。RSA アルゴリズムは、最も幅広く使用されている暗号化 / 認証アルゴリズムで、Netscape 社および Microsoft 社の Web ブラウザや、他の多くの製品の一部として組み込まれている。

RSAES-OAEP

RSA Encryption Scheme - Optimal Asymmetric Encryption Padding の略称。**RSA** アルゴリズムと OAEP 方式を組み合せた公開鍵暗号化方式。Optimal Asymmetric Encryption Padding（OAEP）は、Mihir Bellare と Phil Rogaway の 2 氏によって開発されたメッセージ・エンコーディング方式である。

S/MIME

「**Secure/Multipurpose Internet Mail Extension**」を参照。

SAML

「**Security Assertions Markup Language**」を参照。

SASL

「**Simple Authentication and Security Layer**」を参照。

Secure Hash Algorithm（SHA）

入力に基づいて 160 ビットの**メッセージ・ダイジェスト**を生成する**ハッシュ関数**アルゴリズム。このアルゴリズムは、Digital Signature Standard（DSS）で使用されている。128、192、256 ビットの 3 通りの鍵サイズを提供する Advanced Encryption Standard（AES）の導入によって、それらに対応する同レベルのセキュリティを持つハッシュ・アルゴリズムが必要となっている。より新しい SHA-256、SHA-284 および SHA-512 ハッシュ・アルゴリズムは、これらの拡張要件を満たしている。

Secure Sockets Layer（SSL）

ネットワーク（インターネットなど）経由での暗号化および認証された通信を実現するために、Netscape 社によって設計されたプロトコル。SSL では、RSA 社の**公開鍵暗号**システムが使用され、デジタル証明書の使用も組み込まれている。SSL では、セキュアな通信の 3 要素である**機密保護**、**認証**および**整合性**が実現される。

SSL は **Transport Layer Security** へと発展している。TLS と SSL は相互運用できないが、SSL 対応クライアントは TLS で送信されたメッセージを処理できる。

Secure/Multipurpose Internet Mail Extension（S/MIME）

デジタル署名と**暗号化**を使用した MIME データの保護に関する Internet Engineering Task Force（IETF）の標準。

Security Assertions Markup Language（SAML）

サブジェクトに関するセキュリティ情報を交換するためのメカニズムを定義する **XML** ベースのフレームワーク。これは、アクセス制御の決定に使用するサブジェクトについてのアサーションを作成することによって行う。SAML は、他の方法では相互運用できない可能性がある ID プロバイダとサービス・プロバイダの間で、**認証**および**認可**情報の交換を可能にする。

SAML 2.0 は、SAML 1.1 標準のバージョンアップ版で、Shibboleth および **Liberty ID-FF** の両方の仕様が取り込まれている。SAML 2.0 の重要な点は、ユーザー ID をそのユーザーの協力により 2 つのサイトが確立、保守できる点である。また、プライバシ・メカニズムおよびグローバル・ログアウトのサポートも含まれる。

SGA

「**システム・グローバル領域**」を参照。

SHA

「**Secure Hash Algorithm**」を参照。

Signed Public Key And Challenge（SPKAC）

Netscape Navigator ブラウザが証明書のリクエストに使用する固有のプロトコル。

Simple Authentication and Security Layer（SASL）

アプリケーション・プロトコルに**認証**および**認可**機能を追加する方法。SASL により、プロトコルとコネクションの間にセキュリティ・レイヤーが提供され、ユーザーがサーバーに対して認証可能となる。後続のプロトコル対話を保護するようにセキュリティ・レイヤーを規定することもできる。

Simple Object Access Protocol（SOAP）

Simple Object Access Protocol の略称。**XML** ベースのプロトコルで、インターネットを介したシステム間でメッセージを交換するためのフレームワークを定義する。SOAP は、Web サービスで広く使用されるプロトコルであり、HTTP、FTP などの転送プロトコルで使用される。SOAP メッセージは 3 つの部分から成る。1 つは、メッセージとその処理方法が記述されたエンベロープで、残りは、アプリケーションで定義されたデータ型のインスタンスを表現するための 1 組のエンコーディング規則と、リモート・プロシージャ・コールおよびレスポンスの表記規則である。

Single Sign-On SDK

OracleAS Single Sign-On パートナ・アプリケーションをシングル・サインオン対応にするためのレガシー API。SDK は、PL/SQL API および Java API、さらにこれらの API を実装する方法を実証するサンプル・コードで構成される。この SDK は現在は使用不可で、かわりに **mod_osso** が使用される。

Single Sign-On Server

Single Sign-On アプリケーション（経費報告、電子メール、福利厚生情報など）に安全にログインできるようにするプログラム・ロジック。

SLAPD

スタンドアロンの LDAP デーモン。レプリケーションを除くディレクトリの大半の機能を担当する LDAP ディレクトリ・サーバー・サービス。

SOAP

「**Simple Object Access Protocol**」を参照。

SPKAC

「**Signed Public Key And Challenge**」を参照。

SSL

「**Secure Sockets Layer**」を参照。

SSO

「**シングル・サインオン**」を参照。

subACLSubentry

アクセス制御リスト情報が含まれた特定のタイプの**サブエントリ**。

subSchemaSubentry

スキーマ情報が含まれた特定のタイプの**サブエントリ**。

Time Stamp Protocol（TSP）

RFC 3161 で規定されるプロトコル。デジタル・メッセージのタイムスタンプに関する参加エンティティ、メッセージ形式および転送プロトコルが定義される。TSP システムでは、信頼できる第三者機関である時刻認証局（TSA）によって、メッセージのタイムスタンプが発行される。

TLS

「**Transport Layer Security**」を参照。

Transport Layer Security（TLS）

インターネット上の通信プライバシを提供するプロトコル。このプロトコルによって、クライアント / サーバー・アプリケーションは、通信時の盗聴、改ざんまたはメッセージの偽造を防止できる。

Triple Data Encryption Standard（3DES）

IBM 社によって 1974 年に開発された**データ暗号化規格**に基づく暗号化アルゴリズム。1977 年に米国の連邦標準として採用されている。3DES では、64 ビットの鍵が 3 つ使用される（鍵の長さは全体で 192 ビットになるが、実際の鍵長は 56 ビットである）。データは、第一の鍵で暗号化され、第二の鍵で復号化され、さらに第三の鍵で再度暗号化される。結果として、3DES は標準的な DES よりも 3 倍低速になるが、3 倍セキュアになる。

TSP

「**Time Stamp Protocol**」を参照。

Unicode

汎用キャラクタ・セットのタイプ。16 ビットの領域にエンコードされた 64K 個の文字の集合。既存のほとんどすべてのキャラクタ・セット規格の文字をすべてエンコードする。世界中で使用されているほとんどの記述法を含む。Unicode は Unicode Inc. によって所有および定義される。Unicode は標準的なエンコーディングであり、異なるロケールで値を伝達できることを意味する。しかし、Unicode とすべての Oracle キャラクタ・セットとの間で、情報の損失なしにラウンドトリップ変換が行われることは保証されない。

UNIX Crypt

UNIX 暗号化アルゴリズム。

URI

Uniform Resource Identifier の略称。Web 上にある、あらゆるコンテンツ（テキスト・ページ、ビデオ・クリップ、サウンド・クリップ、静止画、動画、プログラムなど）の位置を識別する手段。最も一般的な URI は Web ページ・アドレスで、**URL** と呼ばれる、URI の特別な形式つまりサブセットで構成される。

URL

Uniform Resource Locator の略称。インターネット上にあるアクセス可能なファイルのアドレス。テキスト・ファイル、HTML ページ、画像ファイル、プログラムなど、HTTP でサポートされるすべてのファイルが対象となる。URL には、リソースへのアクセスに必要なプロトコルの名前、インターネット上の特定のコンピュータを識別するドメイン名、およびコンピュータ上のファイル場所の階層的な記述が含まれる。

URLC トークン（URLC token）

認証されたユーザー情報を**パートナ・アプリケーション**に渡す OracleAS Single Sign-On コード。パートナ・アプリケーションはこの情報を使用してセッション Cookie を作成する。

UTC（Coordinated Universal Time）

世界中のあらゆる場所で共通の標準時間。以前から現在に至るまで広くグリニッジ時（GMT）または世界時と呼ばれており、UTC は名目上は地球の本初子午線に関する平均太陽時を表す。UTC 形式である場合、値の最後に z が示される（例 : 200011281010z）。

UTF-16

Unicode の 16 ビット・エンコーディング。Latin-1 文字は、この規格の最初の 256 コード・ポイントである。

UTF-8

文字ごとに連続した 1、2、3 または 4 バイトを使用する **Unicode** の可変幅 8 ビット・エンコーディング。0 ～ 127 の文字（7 ビット ASCII 文字）は 1 バイトでエンコードされ、128 ～ 2047 の文字では 2 バイト、2048 ～ 65535 の文字では 3 バイト、65536 以上の文字は 4 バイトを必要とする。このための Oracle キャラクタ・セット名は AL32UTF8（Unicode 3.1 規格用）となる。

Wallet

個々のエンティティに対するセキュリティ資格証明の格納と管理に使用される抽象的な概念。様々な暗号化サービスで使用するために、資格証明の格納と取出しを実現する。Wallet Resource Locator（WRL）は、Wallet の位置を特定するために必要な情報をすべて提供する。

Wallet Manager

「**Oracle Wallet Manager**」を参照。

Web Services Description Language（WSDL）

XML を使用して Web サービスを定義するための標準形式。WSDL 定義には、Web サービスへのアクセス方法と、それを使用して実行できる操作が記述される。

Web サービス（Web service）

HTTP、**XML**、**SOAP** などの標準的なインターネット・プロトコルを使用してアクセスできるアプリケーションまたはビジネス・ロジック。Web サービスでは、コンポーネントベース開発と World Wide Web の両者が持つ優れた利点が結び付けられている。Web サービスは、コンポーネントと同様、サービスの実装方法を知らなくても使用または再利用できるブラックボックス機能を実現する。

WS-Federation

Web Services Federation Language のこと。Microsoft、IBM、BEA、VeriSign および RSA Security 社によって開発された仕様。WS-Federation は、異種または同種の方式を使用するエンティティ間で、**フェデレーション**を構築可能にするメカニズムを定義する。これは、公開されている **Web サービス**間で、識別情報、属性および認証の信頼性を確立および仲介することで実現される。

「**Liberty Alliance**」も参照。

WSDL

「**Web Services Description Language**」を参照。

X.500

グローバル・ディレクトリの構造化方法を定義する、国際電気通信連合（ITU）の標準。X.500 ディレクトリは、国、都道府県、市町村などの情報カテゴリごとに異なるレベルを持つ階層である。

X.509

デジタル証明書の定義において、最も幅広く使用されている標準。これは、認証サービスが備わった階層型ディレクトリに関する国際電気通信連合（ITU）の標準で、多くの**公開鍵インフラストラクチャ**実装で使用されている。

XML

eXtensible Markup Language の略称。World Wide Web Consortium（W3C）によって開発された仕様。XML は、Standard Generalized Mark-Up Language（SGML）の縮小版で、Web ドキュメントに特化して設計されている。XML はメタ言語（タグ・セットを定義する方法）で、開発者はこれによって、様々な種類のドキュメントに対して独自にカスタマイズしたマークアップ言語を定義できる。

XML 正規化（XML canonicalization: XML C14N）

論理的に同等な 2 つの XML ドキュメントを同じ物理表現に解決するプロセス。署名はデータが最初に計算処理されたときの物理表現に対してのみ検証可能なため、XML 正規化はデジタル証明において重要となる。詳細は、W3C の XML 正規化仕様を参照。

アーティファクト・プロファイル（artifact profile）

アサーション全体を送信するのではなく、アーティファクトという**アサーション**への小さなリファレンスを使用してデータを送信する**認証**メカニズム。この**プロファイル**は、処理できる文字数が限定されたブラウザに対応する。

アカウント・ロックアウト（account lockout）

指定された時間内にログオン試行に繰り返し失敗した場合に、セキュリティ・ポリシーの設定に基づいてユーザー・アカウントをロックするセキュリティ機能。OracleAS Single Sign-On では、ユーザーがアカウントとパスワードの組合せを、任意の数のワークステーションから Oracle Internet Directory によって許可されている回数を超えて発行するとアカウント・ロックアウトが適用される。デフォルトのロックアウト時間は 24 時間である。

アクセス制御項目（access control item: ACI）

ACI とは、様々なエンティティまたは対象がディレクトリ内の指定されたオブジェクトに対して操作を行う必要がある権限を表す。この情報は、ユーザーによる変更が可能な操作**属性**として Oracle Internet Directory に格納され、各属性はアクセス制御項目（ACI）と呼ばれる。ACI により、ディレクトリ・データに対するユーザー・アクセス権限が決定される。ACI は、エントリ（構造型アクセス項目）と属性（コンテンツ・アクセス項目）に対するアクセスを制御する 1 組の規則で構成される。両方のアクセス項目に対するアクセス権限を、1 つ以上のユーザーまたはグループに付与できる。

アクセス制御ポリシー・ポイント（access control policy point: ACP）

ディレクトリ情報ツリー内のすべての下位エントリに適用されるアクセス制御ポリシー情報を含むディレクトリ・エントリ。この情報は、エントリ自体とその下位エントリすべてに影響を与える。Oracle Internet Directory では、ACP を作成することで、ディレクトリ内の**サブツリー**全体にアクセス制御ポリシーを適用できる。

アクセス制御リスト（access control list : ACL）

コンピュータ・システム内のリソースと、それらのリソースへのアクセスを許可されたユーザーのユーザー名から成るリスト。Oracle Internet Directory では、ACL は、ディレクトリ・オブジェクトに関連付けられた**アクセス制御項目**の**属性値**のリストである。このリストの属性値は、様々なディレクトリ・ユーザー・エンティティ（対象）が各オブジェクトに対して所有している権限を表す。

アサーション（assertion）

リソースにアクセスしようとしている対象についての情報を交換するために、セキュリティ・ドメインでプロバイダが使用する文。ID プロバイダとサービス・プロバイダは ID に関するアサーションを交換して**認証**および**認可**を決定し、リソースを保護するセキュリティ・ポリシーを有効にする。

アドバンスト対称型レプリケーション（advanced symmetric replication: ASR）

「**Oracle Database アドバンスト・レプリケーション**」を参照。

アドバンスト・レプリケーション（advanced replication）

「**Oracle Database アドバンスト・レプリケーション**」を参照。

アプリケーション・サービス・プロバイダ（application service provider）

ソフトウェアベースのサービスやソリューションの管理および配信を、中央のデータ・センターから Wide Area Network を通じて顧客に提供するサード・パーティ・エンティティ。つまり、アプリケーション・サービス・プロバイダ（ASP）は、企業が必要とする情報技術の一部またはすべてをアウトソースする手段となる。

暗号（cipher）

「**暗号化アルゴリズム**」を参照。

暗号化（cryptography）

情報を判読できない形式に変換することによって保護する処理。情報は、データを判読不能にする**鍵**を使用して暗号化され、情報が再度必要になったときに復号化される。「**公開鍵暗号**」と「**対称型暗号**」も参照。

暗号化（encryption）

暗号化アルゴリズムを適用することで、平文を暗号文に変換する処理。

暗号化アルゴリズム（cryptographic algorithm）

判読可能なデータ（平文）を判読できないデータ（暗号文）に変換する、またはその逆を行うために定義された一連の処理手順。これらの変換には特別なシークレット情報が必要で、通常、それらは**鍵**に含まれる。暗号化アルゴリズムには、**DES**、**AES**、**Blowfish**、**RSA** などがある。

暗号化証明書（encryption certificate）

電子メッセージ、ファイル、ドキュメントまたはデータ伝送の暗号化や、同じ目的のセッション鍵の交換または確立に使用される**公開鍵**を含む**証明書**。

暗号スイート（cipher suite）

Secure Sockets Layer において、ネットワークのノード間でメッセージ交換に使用される認証、暗号化およびデータ整合性アルゴリズムのセット。SSL ハンドシェイク時に、2 つのノード間で折衝し、メッセージを送受信するときに使用する暗号スイートを確認する。

暗号ブロック連鎖（cipher block chaining: CBC）

ブロック暗号の操作モードの 1 つ。CBC では、初期設定ベクトル（IV）と呼ばれる特定長のベクトル値が使用される。CBC の最も重要な特性の 1 つは、連鎖メカニズムによって、特定の暗号文ブロックの復号化が、それよりも前のすべての暗号文ブロックに依存することにある。結果として、1 つ前の暗号文ブロックには、それよりも前のすべてのブロックの全体としての妥当性が含まれることになる。

暗号文（ciphertext）

判読可能なデータ（平文）に**暗号化アルゴリズム**を適用することで、適切な**鍵**の所有者以外は誰も判読できないデータに変換したもの。

一方向関数（one-way function）

一方向への計算は容易だが、逆の計算（反対方向への計算）は非常に難しい関数。

一方向ハッシュ関数（one-way hash function）

可変サイズの入力を取得して、固定サイズの出力を作成する**一方向関数**。

「**ハッシュ関数**」も参照。

一致規則（matching rule）

検索または比較操作における、検索対象の属性値と格納されている属性値との間の等価性の判断。たとえば、telephoneNumber 属性に関連付けられた一致規則では、(650) 123-4567 を (650) 123-4567 または 6501234567 のいずれか、あるいはその両方と一致させることができる。**属性**の作成時に、その属性を一致規則と対応付けることができる。

委任管理者（delegated administrator）

ホスティングされた環境では、アプリケーション・サービス・プロバイダなどの 1 企業が、他の複数の企業に Oracle コンポーネントを使用可能にして、その情報を格納する。この種の環境では、グローバル管理者はディレクトリ全体にまたがるアクティビティを実行する。委任管理者と呼ばれる他の管理者は、特定の ID 管理レルムでのロール、または特定のアプリケーションに対してのロールを持つ。

インスタンス（instance）

「**ディレクトリ・サーバー・インスタンス**」を参照。

インフラストラクチャ層（infrastructure tier）

ID 管理を担当する Oracle Application Server コンポーネント。OracleAS Single Sign-On、Oracle Delegated Administration Services および Oracle Internet Directory が、このコンポーネントに該当する。

インポート・エージェント（import agent）

Oracle Directory Integration Platform 環境で、Oracle Internet Directory にデータをインポートするエージェント。

インポート・データ・ファイル（import data file）

Oracle Directory Integration Platform 環境で、**インポート・エージェント**によってインポートされたデータを格納するファイル。

エクスポート・エージェント（export agent）

Oracle Directory Integration Platform 環境で、Oracle Internet Directory からデータをエクスポートするエージェント。

エクスポート・データ・ファイル（export data file）

Oracle Directory Integration Platform 環境で、**エクスポート・エージェント**によってエクスポートされたデータを格納するファイル。

エクスポート・ファイル（export file）

「**エクスポート・データ・ファイル**」を参照。

エンドツーエンド・セキュリティ（end-to-end security）

メッセージレベルのセキュリティによって実現される特性。ビジネス・エンティティ内およびビジネス・エンティティ間の複数のアプリケーションでメッセージが伝送処理されるときに、それらのあらゆる経路でメッセージがセキュアである場合に確立される。

エントリ（entry）

ユーザーなどのオブジェクトを表す、ディレクトリ内の一意なレコード。エントリは**属性**とそれに関連付けられた**属性値**で構成され、それらはエントリ・オブジェクトを定義する**オブジェクト・クラス**によって規定される。LDAP ディレクトリ構造内のすべてのエントリは、その**識別名**によって一意に識別される。

オブジェクト・クラス（object class）

LDAP において、情報のグループ化に使用される。通常、オブジェクト・クラスは、従業員やサーバーなどの実社会の事物をモデル化する。各ディレクトリ・エントリは、1 つ以上のオブジェクト・クラスに属する。オブジェクト・クラスは、エントリを構成する属性を決定する。オブジェクト・クラスは別のオブジェクト・クラスから導出でき、結果として、他のクラスの特性が継承される。

介在者（man-in-the-middle）

第三者によるメッセージの不正傍受などのセキュリティ攻撃。第三者、つまり介在者は、メッセージを復号化して再暗号化し（元のメッセージを変更する場合と変更しない場合がある）、元のメッセージの宛先である受信者に転送する。これらの処理はすべて、正当な送受信者が気付かないうちに行われる。この種のセキュリティ攻撃は、**認証**が行われていない場合にのみ発生する。

外部アプリケーション（external application）

OracleAS Single Sign-On サーバーに認証を委任しないアプリケーション。そのかわり、HTML ログイン・フォームが表示され、アプリケーションのユーザー名とパスワードが要求される。ユーザーは、最初のログイン時に OracleAS Single Sign-On サーバーで自身の資格証明を取得するように選択できる。その後、ユーザーは外部アプリケーションに透過的にログインできるようになる。

外部エージェント（external agent）

Oracle Directory Integration Platform サーバーに依存しないディレクトリ統合エージェント。Oracle Directory Integration Platform サーバーは外部エージェントに対して、スケジューリング、マッピングまたはエラー処理の各サービスを提供しない。外部エージェントは、通常、サード・パーティのメタディレクトリ・ソリューションを Oracle Directory Integration Platform に統合するときに使用する。

鍵（key）

特定のデータ・ブロックの暗号化と復号化の成功に必要なシークレット情報を含むデータ構造。鍵のサイズが大きくなるにつれて、暗号化されたデータ・ブロックのクラッキングは困難になる。たとえば、256 ビットの鍵は 128 ビットの鍵よりも安全である。

鍵のペア（key pair）

公開鍵とそれに対応する**秘密鍵**のペア。

「**公開鍵と秘密鍵のペア**」も参照。

仮想 IP アドレス（virtual IP address）

Oracle Application Server Cold Failover Cluster（Identity Management）では、各物理ノードに独自の物理 IP アドレスと物理ホスト名がある。単一のシステムであるというイメージを外部に示すために、クラスタは、クラスタ内のどの物理ノードにも変更できる動的 IP アドレスを使用する。これは、仮想 IP アドレスと呼ばれる。

仮想ホスト（virtual host）

1 つ以上の Web サイトまたはドメインをホスティングする 1 台の物理的な Web サーバー・マシン、または他のマシンに対するプロキシ（受信リクエストを受け取り、それらを適切なサーバーにルーティングする）としての機能を持つサーバー。

OracleAS Single Sign-On では、仮想ホストは、2 つ以上の OracleAS Single Sign-On サーバー間でのロード・バランシングに使用される。また、仮想ホストはセキュリティの追加層を提供する。

仮想ホスト名（virtual host name）

Oracle Application Server Cold Failover Cluster（Identity Management）で、特定の仮想 IP アドレスに対応するホスト名。

簡易認証（simple authentication）

ネットワークでの送信時に暗号化されない識別名とパスワードを使用して、クライアントがサーバーに対して自己認証を行うプロセス。簡易認証オプションでは、クライアントが送信した識別名とパスワードと、ディレクトリに格納されている識別名とパスワードが一致していることをサーバーが検証する。

管理領域（administrative area）

ディレクトリ・サーバー上の 1 つの**サブツリー**。そのエントリは、1 つの管理認可レベルで制御される。指定された管理者が、管理領域内の各**エントリ**に加えて、ディレクトリ・**スキーマ**、**アクセス制御リスト**およびそれらのエントリの**属性**を制御する。

基本エンコーディング規則（Basic Encoding Rules: BER）

ASN.1 に示されているデータ・ユニットをエンコーディングするための標準規則。BER は ASN.1 と間違って組み合されることがある。ASN.1 は抽象的な構文定義言語で、エンコーディング規則には適用できない。

機密保護（confidentiality）

暗号化では、機密保護（またはプライバシ保護）は、認可されていないエンティティがデータを読み取ることを防止する機能を意味する。通常、これは**暗号化**によって実現される。

キャッシュ（cache）

通常は、コンピュータ内部にある、高速にアクセス可能な一定量のメモリー領域のことを指す。ただし、Web では、ブラウザがダウンロードしたファイルや画像を格納するコンピュータ上の場所を指すことが多い。

競合（contention）

リソースの競合。

強制認証（forced authentication）

ユーザーが事前に設定された時間アイドル状態であった場合に、再認証をユーザーに強制する動作。Oracle Application Server Single Sign-On では、グローバル・ユーザーの非アクティブ・タイムアウトを指定できる。この機能はセキュリティ重視のアプリケーションがインストールされている場合に使用する。

兄弟関係（sibling）

1 つ以上の他のエントリと同じ親を持ったエントリ。

共有サーバー（shared server）

多数のユーザー・プロセスが、非常に少数のサーバー・プロセスを共有できるように構成されたサーバー。これにより、サポートされるユーザー数が増える。共有サーバー構成では、多数のユーザー・プロセスがディスパッチャに接続する。ディスパッチャは、複数の着信ネットワーク・セッション・リクエストを共通キューに送る。複数のサーバー・プロセスの共有プールの中で、あるアイドル状態の共有サーバー・プロセスが共通キューからリクエストを取り出す。これは、サーバー・プロセスの小規模プールで大量のクライアントを処理できることを意味する。専用サーバーと対比。

クライアント SSL 証明書（client SSL certificates）

Secure Sockets Layer で、サーバーに対するクライアント・マシンの身元確認（クライアント認証）に使用される**証明書**。

クラスタ（cluster）

単一のコンピューティング・リソースとして使用される、相互接続された使用可能なすべてのコンピュータの集合。ハードウェア・クラスタによって、高可用性およびスケーラビリティが実現する。

グループ検索ベース（group search base）

Oracle Internet Directory のデフォルトの**ディレクトリ情報ツリー**で、すべてのグループを検索できる ID 管理レルムのノード。

グローバリゼーション・サポート（globalization support）

Graphical User Interface（GUI）に対する複数言語サポート。Oracle Application Server Single Sign-On では、29 言語がサポートされる。

グローバル管理者（global administrator）

ホスティングされた環境では、アプリケーション・サービス・プロバイダなどの 1 企業が、他の複数の企業に Oracle コンポーネントを使用可能にして、その情報を格納する。この種の環境では、グローバル管理者はディレクトリ全体にまたがるアクティビティを実行する。

グローバルな一意のユーザー ID（globally unique user ID）

ユーザーを一意に識別する数値。ユーザー名、パスワード、識別名は変更または追加できるが、グローバルな一意のユーザー ID は常に同じである。

グローバル・ユーザーの非アクティビティ・タイムアウト（global user inactivity timeout）

Oracle Application Server Single Sign-On のオプション機能。事前定義された一定時間アイドル状態が続いた場合に、ユーザーに再認証を強制する。グローバル・ユーザーの非アクティビティ・タイムアウトは、シングル・サインオン・セッションのタイムアウトよりかなり短い。

継承（inherit）

オブジェクト・クラスが別のクラスから導出されたときに、導出元のオブジェクト・クラスの多数の特性も導出（継承）されること。同様に、属性のサブタイプも、そのスーパータイプの特性を継承する。

ゲスト・ユーザー（guest user）

匿名ユーザーではなく、特定のユーザー・エントリも持っていないユーザー。

検証（verification）

署名とその署名の意図的な適用先となるデータ・ブロックを作成する目的で、**公開鍵**とそれに対応する**秘密鍵**が与えられているときに、特定の**デジタル署名**が有効であることを確認するプロセス。

コード署名証明書（code signing certificates）

Java プログラム、JavaScript またはその他の署名ファイルに署名したエンティティの身元確認に使用される**証明書**。

コールド・バックアップ（cold backup）

Oracle Internet Directory では、データベース・コピー・プロシージャを使用して、新規**ディレクトリ・システム・エージェント**ノードを既存のレプリケート・システムに追加する手順を表す。

公開鍵（public key）

公開鍵暗号で使用される**公開鍵と秘密鍵のペア**において、一般に公開される鍵。エンティティは、公開鍵を使用してデータを暗号化する。そのデータは、公開鍵の所有者のみが、対応する**秘密鍵**を使用して復号化できる。公開鍵は、対応する秘密鍵で作成されたデジタル署名の検証にも使用できる。

公開鍵暗号（public key cryptography）

公開鍵暗号（非対称型暗号とも呼ばれる）では、公開鍵と秘密鍵の 2 つの鍵が使用される。これらの鍵は、鍵のペアと呼ばれる。秘密鍵は秘密にしておく必要があるが、公開鍵は任意のパーティに送信できる。秘密鍵と公開鍵は、数学的に関連付けられている。秘密鍵によって署名されたメッセージは、対応する公開鍵によって検証できる。同様に、公開鍵によって暗号化されたメッセージは、対応する秘密鍵によって復号化できる。秘密鍵の所有者のみがメッセージを復号化できるため、この方式によって機密保護が保証される。

公開鍵暗号（public-key encryption）

メッセージの送信側が、受信側の公開鍵でメッセージを暗号化するプロセス。配信されたメッセージは、受信側の秘密鍵で復号化される。

公開鍵インフラストラクチャ（public key infrastructure: PKI）

公開鍵と**秘密鍵**の発行、配布および証明を管理するシステム。通常、PKI は次のコンポーネントによって構成される。

- **認証局**：デジタル証明書の生成、発行、公開および失効に責任を持つ。
- **登録局**：CA に対する証明書リクエストに記載されている情報の検証に責任を持つ。
- ディレクトリ・サービス：CA によって**証明書**または**証明書失効リスト**が公開される場所。このシステムに依存する第三者は、ここでそれらを取得できる。
- 依存する第三者：**デジタル署名**の検証とデータの暗号化に、CA によって発行された証明書と、それに含まれる**公開鍵**を使用するエンティティ。

公開鍵証明書（public key certificate）

「**証明書**」を参照。

公開鍵と秘密鍵のペア（public/private key pair）

数学的に関連付けられた 2 つの数字のセット。1 つは秘密鍵、もう 1 つは公開鍵と呼ばれる。公開鍵は通常広く使用可能であるのに対して、秘密鍵はその所有者のみ使用可能である。公開鍵で暗号化されたデータは、それに関連付けられた秘密鍵でのみ復号化でき、秘密鍵で暗号化されたデータは、それに関連付けられた公開鍵でのみ復号化できる。公開鍵で暗号化されたデータを、同じ公開鍵で復号化することはできない。

構成設定エントリ（configuration set entry）

ディレクトリ・サーバーの特定インスタンスに関する構成パラメータを保持している Oracle Internet Directory エントリ。複数の構成設定エントリを格納でき、実行時に参照できる。構成設定エントリは、**ディレクトリ固有のエントリ**の `subConfigsubEntry` 属性で指定されているサブツリー内でメンテナンスされる。DSE 自体は、サーバーの起動対象である関連の**ディレクトリ情報ベース**に常駐している。

高度なエンコーディング規則（Distinguished Encoding Rules: DER）

ASN.1 オブジェクトをバイト・シーケンスにエンコーディングするための 1 組の規則。DER は**基本エンコーディング規則**の特殊な形式である。

コンシューマ（consumer）

レプリケーション更新の宛先となるディレクトリ・サーバー。スレーブと呼ばれることもある。

コンテキスト接頭辞（context prefix）

ネーミング・コンテキストのルートの**識別名**。

サード・パーティのアクセス管理システム（third-party access management system）

OracleAS Single Sign-On を使用して Oracle Application Server アプリケーションにアクセスできるように変更できる、Oracle 以外のシングル・サインオン・システム。

サーバー証明書（server certificate）

セキュアな Web サーバーを使用してデータを提供する組織の ID が真正であることを証明する**証明書**。サーバー証明書は、相互に信頼できる**認証局**によって発行された**公開鍵と秘密鍵のペア**に関連付けられる必要がある。サーバー証明書は、ブラウザと Web サーバー間のセキュアな通信に必須である。

サービス時間（service time）

リクエストの開始から、そのリクエストに対するレスポンスの完了までの時間。

サービス・プロバイダ（service provider）

OSFS でサポートされている **ID フェデレーション**・プロトコルに定義された 3 つの主要な役割の 1 つ。残りの 2 つは、**ID プロバイダ**および**プリンシパル**。

サービス・プロバイダは、SAML の依存者であり、ID プロバイダにプリンシパルの ID の認証を依頼しながら、サービスまたは商品をプリンシパルに提供する。

サブエントリ（subentry）

サブツリー内のエントリ・グループに適用可能な情報が含まれているエントリのタイプ。情報には次の 3 つのタイプがある。

- アクセス制御ポリシー・ポイント
- スキーマ規則
- 共通属性

サブエントリは、管理領域のルートのすぐ下に位置している。

サブクラス（subclass）

別のオブジェクト・クラスから導出されたオブジェクト・クラス。導出元のオブジェクト・クラスは、その**スーパークラス**と呼ばれる。

サブスキーマ DN（subschema DN）

独立した**スキーマ**定義を持つ**ディレクトリ情報ツリー**領域のリスト。

サブタイプ（subtype）

オプションを持たない同じ属性に対して、1 つ以上のオプションを持つ属性。たとえば、American English をオプションとして持つ commonName（cn）属性は、そのオプションを持たない commonName（cn）属性のサブタイプである。逆に、オプションを持たない commonName（cn）属性は、オプションを持つ同じ属性の**スーパータイプ**である。

サブツリー（subtree）

ディレクトリ階層（**ディレクトリ情報ツリー**とも呼ばれる）の中の 1 つのセクション。通常、サブツリーは特定のディレクトリ・ノードから始まり、ディレクトリ階層内でそのノードよりも下位にあるすべてのサブディレクトリとオブジェクトが含まれる。

サプライヤ（supplier）

レプリケーションにおいて、**ネーミング・コンテキスト**のマスター・コピーを保持しているサーバー。マスター・コピーから**コンシューマ**・サーバーに更新を供給する。

参照（referral）

ディレクトリ・サーバーがクライアントに提供する情報。リクエストする情報を見つけるためにクライアントが接続する必要がある他のサーバーを示す。

「**ナレッジ参照**」も参照。

識別名（distinguished name: DN）

X.500 識別名（DN）は、ディレクトリ・ツリー内のノードの一意名である。DN は、ユーザーまたはそれ以外のディレクトリ・エントリの一意名の作成に使用される。DN は、ルート・ノードから特定のエントリのノードまでのパス上にある、ツリー内の各ノードから選択された**属性**の連結である。たとえば、LDAP 表記規則では、米国のオラクル社に勤務する John Smith という名前のユーザーの DN は、cn=John Smith, ou=People, o=Oracle, c=us となる。

思考時間（think time）

ユーザーが実際にプロセッサを使用していない時間。

システム・グローバル領域（System Global Area: SGA）

共有メモリー構造の 1 グループ。1 つの Oracle データベース・インスタンスに関するデータと制御情報が含まれている。複数のユーザーが同じインスタンスに同時に接続した場合、そのインスタンスの SGA 内のデータはユーザー間で共有される。したがって、SGA は共有グローバル領域と呼ばれることもある。バックグラウンド・プロセスとメモリー・バッファの組合せは、Oracle インスタンスと呼ばれる。

システム固有のエージェント（native agent）

Oracle Directory Integration Platform 環境において、**Directory Integration and Provisioning Server** の制御下で実行されるエージェント。**外部エージェント**と対比。

システム操作属性（system operational attribute）

ディレクトリ自体の操作に関係する情報を保持する属性。一部の操作情報は、サーバーを制御するためにディレクトリによって指定される（例 : エントリのタイムスタンプ）。アクセス情報などのその他の操作情報は、管理者が定義し、ディレクトリ・プログラムの処理時に、そのプログラムによって使用される。

従属 CA（subordinate CA）

従属**認証局**のこと。階層構造を持つ**公開鍵インフラストラクチャ**において、その証明書署名鍵が別の CA によって証明され、その役割が他の CA によって制約される CA。

従属参照（subordinate reference）

エントリのすぐ下から始まる**ネーミング・コンテキスト**の参照位置を、**ディレクトリ情報ツリー**内の下位方向に指し示す**ナレッジ参照**。

上位参照（superior reference）

ディレクトリ情報ツリー内で、参照先の**ディレクトリ・システム・エージェント**が保持しているすべてのネーミング・コンテキストより上位のネーミング・コンテキストを保持している DSA を上位方向に指し示す**ナレッジ参照**。

条件（predicates）

Oracle Application Server Certificate Authority（OCA）において、ポリシーに適用可能な論理式のこと。ポリシー条件式は、受信する証明書リクエストまたは証明書失効化に対してポリシーを適用する方法を制限する。たとえば、次の条件式は、DN に ou=sales,o=acme,c=us が含まれるクライアントからのリクエストまたは失効化には、表示されているポリシーが異なる効力を持つことを指定している。

```
Type=="client" AND DN=="ou=sales,o=acme,c=us"
```

証明書（certificate）

公開鍵とその所有者の識別情報を関連付ける特別な形式のデータ構造。証明書は、**認証局**によって発行される。証明書には、特定のエンティティの名前、シリアル番号、有効期限および公開鍵が含まれる。証明書は、それが本物であることを受信側が検証できるように、発行元の CA によってデジタル署名される。大半のデジタル証明書は、**X.509** 標準に準拠する。

証明書失効リスト（Certificate Fevocation List: CRL）

発行元の**認証局**によって失効されたデジタル**証明書**のリスト。

証明連鎖（certificate chain）

ユーザー**証明書**とそれに関連付けられた **CA 証明書**の 1 つ以上のペアを含む、順序付けられた証明書のリスト。

シングル・サインオフ（single sign-off）

OracleAS Single Sign-On セッションを終了して、すべてのアクティブなパートナ・アプリケーションから同時にログアウトするプロセス。作業中のアプリケーションからログアウトすると、シングル・サインオフを実行できる。

シングル・サインオン（single sign-on: SSO）

フェデレーテッド（連携）環境では、シングル・サインオンにより、ユーザーは ID プロバイダとサービス・プロバイダから構成されるフェデレーテッド・グループのメンバーに一度のみサインオンすれば、後から各メンバーのリソースを使用する際に再びサインオンする必要がない。

申告（claim）

エンティティによって行われる宣言内容（名前、ID、鍵、グループなど）。

信頼できる証明書（trusted certificate）

一定の信頼度を有すると認定された第三者の識別情報。信頼できる証明書は、識別情報の内容がそのエンティティと一致していることを検証するときに使用される。通常、信頼できる証明書は、ユーザー証明書の発行業務を行う、信頼された**認証局**によって発行される。

スーパークラス（superclass）

別のオブジェクト・クラスの導出元の**オブジェクト・クラス**。たとえば、オブジェクト・クラス `person` は、オブジェクト・クラス `organizationalPerson` のスーパークラスである。後者の `organizationalPerson` は、`person` の**サブクラス**であり、`person` に含まれている属性を継承する。

スーパータイプ（supertype）

1 つ以上のオプションを持つ同じ属性に対して、オプションを持たない属性。たとえば、オプションを持たない `commonName`（`cn`）属性は、オプションを持つ同じ属性のスーパータイプである。逆に、American English をオプションとして持つ `commonName`（`cn`）属性は、そのオプションを持たない `commonName`（`cn`）属性の**サブタイプ**である。

スーパーユーザー（super user）

一般的には、ディレクトリ情報へのあらゆるアクセスが可能な特別なディレクトリ管理者。

スキーマ（schema）

属性、**オブジェクト・クラス**およびそれらに対応する**一致規則**の集合。

スケーラビリティ（scalability）

使用可能なハードウェア・リソースに応じて、そのハードウェア・リソースによってのみ制限されるシステムの機能。

ストリーム暗号（stream cipher）

対称型アルゴリズムの一種。ストリーム暗号では、一度に 1 ビットや 1 バイトという小さな単位で暗号化され、特定形式のフィードバック・メカニズムの実装によって鍵が絶えず変更される。**RC4** はストリーム暗号の例である。

「**ブロック暗号**」も参照。

スポンサ・ノード（sponsor node）

レプリケーションにおいて、新規ノードに初期データを設定するために使用されるノード。

スマート・ナレッジ参照（smart knowledge reference）

ナレッジ参照エントリが検索の有効範囲内にあるときに戻される**ナレッジ参照**。リクエストされた情報を格納しているサーバーを示す。

スループット（throughput）

Oracle Internet Directory が単位時間ごとに処理するリクエストの数。通常、「操作 / 秒」（1 秒当たりの操作件数）で表される。

スレーブ（slave）

「**コンシューマ**」を参照。

成功 URL（success URL）

Oracle Application Server Single Sign-On の使用時に、アプリケーションとのセッションおよびセッション Cookie を構築する役割を持つルーチンへの URL。

整合性（integrity）

暗号化では、権限のないエンティティによってデータが変更されていないかどうかを検出する機能を表す。

セカンダリ・ノード（secondary node）

Oracle Application Server Cold Failover Cluster（Identity Management）で、フェイルオーバー中にアプリケーションの移動先となるクラスタ・ノード。

「**プライマリ・ノード**」も参照。

セキュリティ・トークン（security token）

Liberty プロトコルで、申告を表し、実証する一連のセキュリティ情報を示す。

セッション鍵（session key）

メッセージまたは通信の 1 セッション期間内でのみ使用される**秘密鍵**。

接続記述子（connect descriptor）

特別にフォーマットされた、ネットワーク接続の接続先の説明。接続記述子には、宛先サービスとネットワーク・ルート情報が含まれる。

宛先サービスを示すには、その Oracle Database に対応するサービス名、あるいは Oracle リリース 8.0 またはバージョン 7 のデータベースに対応する Oracle システム識別子（SID）を使用する。ネットワーク・ルートは、少なくとも、ネットワーク・アドレスによってリスナーの位置を提供する。

接続ディレクトリ（connected directory）

Oracle Directory Integration Platform 環境で、それ自体（たとえば、Oracle Human Resource データベース）と Oracle Internet Directory との間で完全なデータの同期が必要な情報リポジトリ。

相対識別名（relative distinguished name: RDN）

ローカルの最下位レベルのエントリ名。エントリのアドレスを一意に識別するために使用される他の修飾エントリ名は含まれない。たとえば、`cn=Smith,o=acme,c=US` では、`cn=Smith` が相対識別名である。

属性（attribute）

ディレクトリの属性には、名前、電話番号、役職名などの具体的なデータ要素が保持される。各**エントリ**は 1 組の属性から構成され、それぞれが**オブジェクト・クラス**に所属する。さらに、各属性にはタイプと値があり、タイプは属性の情報の種類を説明するものであり、値には実際のデータが格納されている。

属性一意性（attribute uniqueness）

指定した 2 つの**属性**に同じ値が含まれていないようにする Oracle Internet Directory 機能。企業ディレクトリと同期しているアプリケーションで、属性を一意キーとして使用することを可能にする。

属性構成ファイル（attribute configuration file）

Oracle Directory Integration Platform 環境で、接続ディレクトリに関係のある属性を指定するファイル。

属性値（attribute value）

特定の**エントリ**の**属性**内に保持される実際のデータ。たとえば、属性の型が `email` であれば、属性値は `sally.jones@oracle.com` のようになる。

属性の型（attribute type）

属性の型は、データ型、最大長、単一値か複数値かなど、データ要素に関する情報を指定する。属性の型は、名前や電子メール・アドレスなど、値が実社会で持つ意味を表し、特定のデータ断片の作成と格納に適用するルールを指定する。

その他の情報リポジトリ（other information repository）

Oracle Internet Directory 以外のすべての情報リポジトリ。Oracle Directory Integration Platform 環境では、Oracle Internet Directory が**中央ディレクトリ**として機能する。

待機時間（latency）

指定したディレクトリ操作が完了するまでのクライアントの待機時間。待機時間は、空費時間として定義される場合がある。ネットワーク通信では、待機時間は、ソースから宛先へパケットが移動する時間として定義される。

待機時間（wait time）

リクエストの発行からレスポンスの開始までの時間。

ダイジェスト（digest）

「**メッセージ・ダイジェスト**」を参照。

対称鍵（symmetric key）

「**秘密鍵**」を参照。

対称型アルゴリズム（symmetric algorithm）

暗号化と復号化に同じ鍵を使用する暗号アルゴリズム。主要な対称型（秘密鍵）アルゴリズムには、**ストリーム暗号**と**ブロック暗号**の 2 つのタイプがある。

対称型暗号（symmetric cryptography）

共有秘密暗号とも呼ばれる、データの暗号化と復号化に同じ鍵を使用するシステム。対称型暗号の課題は、送信者と受信者が秘密鍵を合意する手段の安全性を保証することである。転送中の秘密鍵が第三者によって傍受された場合、傍受者はその秘密鍵を使用することで、その鍵によって暗号化されたすべてのデータを復号できるようになる。通常、対称型暗号は非対称型暗号りも高速で、大量のデータ交換が必要なときに使用されることが多い。対称型暗号のアルゴリズムには、**DES**、**RC2**、**RC4** などがある。

単一鍵ペア Wallet（single key-pair wallet）

単一のユーザー**証明書**とその関連する**秘密鍵**が含まれる **PKCS#12** 形式の Wallet。**公開鍵**は証明書に埋め込まれている。

中央ディレクトリ（central directory）

Oracle Directory Integration Platform 環境で、中央リポジトリとして機能するディレクトリ。Oracle Directory Integration Platform 環境では、Oracle Internet Directory が中央ディレクトリになる。

中間層（middle tier）

Oracle HTTP Server と OC4J で構成される、OracleAS Single Sign-On インスタンスの一部。OracleAS Single Sign-On の中間層は、ID 管理インフラストラクチャ・データベースとクライアントの間にある。

データ暗号化規格（Data Encryption Standard: DES）

幅広く使用されている**対称型暗号**アルゴリズムで、1974 年に IBM 社によって開発された。DES では、64 ビットのデータ・ブロックごとに 56 ビットの鍵が適用される。DES および 3DES は、主に **S/MIME** の暗号化アルゴリズムとして使用される。

データ整合性（data integrity）

受信メッセージの内容が、送信時の元のメッセージの内容から変更されていないことを保証すること。

「**整合性**」も参照。

データベース・アクセス記述子（database access descriptor: DAD）

特定の Oracle Application Server コンポーネント（OracleAS Single Sign-On スキーマなど）のデータベース接続情報。

ディレクトリ（directory）

「**Oracle Internet Directory**」、「**Lightweight Directory Access Protocol**」および「**X.500**」を参照。

ディレクトリ固有のエントリ（directory-specific entry: DSE）

ディレクトリ・サーバー固有のエントリ。異なるディレクトリ・サーバーに同じ**ディレクトリ情報ツリー**名を保持できるが、内容は異なる必要がある。つまり、DSE を保持しているディレクトリに固有の内容を保持できる。DSE は、それを保持しているディレクトリ・サーバーに固有の内容を含むエントリである。

ディレクトリ・サーバー・インスタンス（directory server instance）

ディレクトリ・サーバーの個々の起動のこと。異なるディレクトリ・サーバーの起動（それぞれ、同じまたは異なる構成設定エントリと起動フラグで起動）は、異なるディレクトリ・サーバー・インスタンスと呼ばれる。

ディレクトリ・システム・エージェント（directory system agent: DSA）

ディレクトリ・サーバーを表す **X.500** の用語。

ディレクトリ情報ツリー（directory information tree: DIT）

エントリの **DN** で構成されるツリー形式の階層構造。

ディレクトリ情報ベース（directory information base: DIB）

ディレクトリに保持されているすべての情報の完全なセット。DIB は、**ディレクトリ情報ツリー**内で、階層的に相互に関連するエントリで構成されている。

ディレクトリ同期プロファイル（directory synchronization profile）

Oracle Internet Directory と外部システム間の同期の実現方法を記述した特殊な**ディレクトリ統合プロファイル**。

ディレクトリ統合プロファイル（directory integration profile）

Oracle Directory Integration Platform 環境での、Oracle Directory Integration Platform による外部システムとの通信方法および通信内容を示す Oracle Internet Directory のエントリ。

ディレクトリ・ネーミング・コンテキスト（directory naming context）

「**ネーミング・コンテキスト**」を参照。

ディレクトリ・プロビジョニング・プロファイル（directory provisioning profile）

Oracle Directory Integration Platform がディレクトリ対応アプリケーションに送信するプロビジョニング関連通知の性質を記述した特殊な**ディレクトリ統合プロファイル**。

ディレクトリ・ユーザー・エージェント（directory user agent: DUA）

ディレクトリ・ユーザーのかわりにディレクトリ・サービスにアクセスするソフトウェア。ディレクトリ・ユーザーは人の場合もあれば、別のソフトウェア・コンポーネントの場合もある。

ディレクトリ・レプリケーション・グループ（directory replication group: DRG）

レプリケーション承諾のメンバーであるディレクトリ・サーバーの集合。

デジタル証明書（digital certificate）

「**証明書**」を参照。

デジタル署名（digital signature）

デジタル署名は、特定のデータ・ブロックに対して 2 ステップのプロセスを適用して得られる。最初に、データに**ハッシュ関数**を適用して結果を生成する。次に、その結果を署名者の**秘密鍵**を使用して暗号化する。デジタル署名は、データの整合性、メッセージ認証および否認防止を保証する目的で使用できる。デジタル署名アルゴリズムには、**DSA**、**RSA**、**ECDSA** などがある。

デフェデレーション（defederation）

ID プロバイダまたは**サービス・プロバイダ**からユーザーのアカウントのリンクを解除する行為。

デフォルト ID 管理レルム（default identity management realm）

ホスティングされた環境では、アプリケーション・サービス・プロバイダなどの 1 企業が、他の複数の企業に Oracle コンポーネントを使用可能にして、その情報を格納する。このようなホスティングされた環境では、ホスティングしている企業はデフォルト ID 管理レルムと呼ばれ、ホスティングされている企業はそれぞれ**ディレクトリ情報ツリー**内のその企業独自の ID 管理レルムに関連付けれる。

デフォルト・ナレッジ参照（default knowledge reference）

ベース・オブジェクトがディレクトリになく、操作がサーバーによってローカルに保持されていない**ネーミング・コンテキスト**で実行されたときに戻される**ナレッジ参照**。デフォルト・ナレッジ参照は、一般的にディレクトリ・パーティション化対策についてより多くのナレッジを持つサーバーに送信する。

デフォルト・レルム位置（default realm location）

デフォルト ID 管理レルムのルートを識別する**ルート Oracle コンテキスト**での属性。

同時クライアント数（concurrent clients）

Oracle Internet Directory とのセッションを確立しているクライアントの総数。

同時実行性（concurrency）

複数のリクエストを同時に処理できる機能。同時実行性メカニズムの例には、スレッドおよびプロセスなどがある。

同時操作数（concurrent operations）

すべての**同時クライアント数**のリクエストに基づいて Oracle Internet Directory で実行されている操作の数。一部のクライアントではセッションがアイドル状態の可能性があるため、この数は同時クライアントの数と必ずしも同じではない。

登録局（Registration Authority: RA）

登録局（RA）は、**認証局**によって証明書が発行される前のユーザーの検証と登録に責任を持つ。RA は、各申請者に対して、新しく適用される証明書の相対識別値または相対識別名を割り当てる。RA は、証明書の署名および発行は行わない。

特定管理領域（specific administrative area）

次の 3 つの側面を制御する管理領域。

- サブスキーマ管理
- アクセス制御管理
- 共通属性管理

特定管理領域では、この 3 つの管理の側面のうち 1 つが制御される。特定管理領域は、自律型管理領域の一部である。

匿名認証（anonymous authentication）

ディレクトリがユーザー名とパスワードの組合せを要求せずにユーザーを認証するプロセス。各匿名ユーザーは、匿名ユーザー用に指定された権限を行使する。

ドメイン（domain）

ドメインには、あるプリンシパルがリソースを利用できる Web サイトおよびアプリケーションが含まれる。フェデレーテッド・サイトは ID プロバイダ（ソース・ドメイン）、サービス・プロバイダ（ターゲット・ドメイン）またはその両方として機能する。

ドメイン・コンポーネント属性（domain component attribute）

ドメイン・コンポーネント（dc）属性は、ドメイン名から識別名を構築する際に使用できる。たとえば、oracle.com などのドメイン名が使用されている場合は、dc=oracle, dc=com で始まる DN を構築して、この DN をディレクトリ情報の該当サブツリーのルートとして使用する。

トラスト・サークル（circle of trust）

ID プロバイダとサービス・プロバイダのグループ内の信頼関係。このグループ内では、プリンシパルによりプロバイダとの商取引において 1 つのフェデレーテッド ID およびシングル・サインオンの使用が認められている。

企業は、Liberty 対応のテクノロジおよび企業間の信頼関係を定義する運用協定に基づいて、トラスト・サークルに参加、加入する。

「フェデレーテッド ID 管理」および「Liberty Alliance」も参照。

トラスト・ポイント（trustpoint）

「信頼できる証明書」を参照。

名前識別子プロファイル（name identifier profile）

フェデレーション・プロファイル。これにより、プロバイダは一般ユーザーの 1 つの名前識別子を割り当てたり、更新するときに、その関係者に通知することができる。

ナレッジ参照（knowledge reference）

リモートディレクトリ・システム・エージェントに関するアクセス情報（名前とアドレス）およびそのリモート DSA が保持しているディレクトリ情報ツリーのサブツリーの名前。ナレッジ参照は、参照とも呼ばれる。

ニックネーム属性（nickname attribute）

ディレクトリ全体のユーザーを一意に識別するために使用する属性。この属性のデフォルト値は `uid`。アプリケーションでは、この属性を使用して単純なユーザー名が完全な識別名に変換される。ユーザー・ニックネーム属性を複数値にはできない。つまり、ユーザーは同じ属性名で格納される複数のニックネームを所有できない。

認可（authorization）

サービスまたはネットワーク・リソースへのアクセスを許可または拒否するプロセス。大半のセキュリティ・システムは、2 ステップのプロセスを基本としている。最初のステップは認証で、ここでユーザーは自身の ID を証明する。2 番目のステップは認可で、ここでユーザーは、各自の ID と定義済の認可ポリシーに基づいて各種リソースへのアクセスが許可される。

認可ポリシー（authorization policy）

認可ポリシーは、保護されたリソースに対するアクセスを制御する方法を決定する。ポリシーによって、ID およびオブジェクトが、特定のシステム・モデルに従って一連の権限に対応付けられる。たとえば、認可ポリシーによって、営業部に属しているユーザーのみが販売レポートにアクセスできるなどが規定される。

認証（authentication）

エンティティが主張している ID を、その資格証明に基づいて検証するプロセス。ユーザーの認証は、一般的に、ユーザーが知っているか所持しているもの（パスワードや証明書など）に基づいて行われる。

電子メッセージの認証の場合は、特定のシステム（**公開鍵暗号**など）を使用して、ファイルまたはメッセージが主張しているとおりの個人または企業から間違いなく発信されたものであることを検証するプロセスや、メッセージの内容に基づくチェックを使用して、メッセージが配信中に変更されていないことを検証するプロセスが含まれる。

認証局（Certificate Authority: CA）

デジタル**証明書**の発行、更新および失効を行う、信頼できる第三者機関。CA の基本的な役割はエンティティの識別情報を保証することで、申請者の検証を**登録局**に委任する場合もある。広く一般に知られている認証局（CA）には、Digital Signature Trust、Thawte、VeriSign などがある。

認証プラグイン（authentication plugin）

特定の認証方式の実装。OracleAS Single Sign-On には、パスワード認証、デジタル証明書、Windows ネイティブ認証、サード・パーティのアクセス管理用に Java プラグインが用意されている。

認証レベル（authentication level）

アプリケーションに特定の認証動作を指定できる、OracleAS Single Sign-On のパラメータ。このパラメータと特定の**認証プラグイン**をリンクできる。

ネーミング・コンテキスト（naming context）

完全に 1 つのサーバーに常駐しているサブツリー。サブツリーは連続している必要がある。つまり、サブツリーの最上位の役割を果すエントリから始まり、下位方向にリーフ・エントリまたは従属ネーミング・コンテキストへの**ナレッジ参照**（参照とも呼ばれる）のいずれかまでを範囲とする必要がある。単一のエントリから**ディレクトリ情報ツリー**全体までをその範囲とすることができる。

ネーミング属性（naming attribute）

Oracle Delegated Administration Services または Oracle Internet Directory Java API を使用して作成した新規ユーザー・エントリの相対識別名を構成するために使用する属性。この属性のデフォルト値は cn。

ネット・サービス名（net service name）

接続記述子に変換されるサービスの単純な名前。ユーザーは、接続するサービスに対する接続文字列内のネット・サービス名に従ってユーザー名とパスワードを渡すことによって、接続リクエストを開始する。次に例を示す。

```
CONNECT username/password@net_service_name
```

必要に応じて、ネット・サービス名は次のような様々な場所に格納できる。

- 各クライアントのローカル構成ファイル（tnsnames.ora）
- ディレクトリ・サーバー
- Oracle Names Server
- NDS、NIS、CDS などの外部ネーミング・サービス

パーティション（partition）

一意の重複していないディレクトリ・ネーミング・コンテキスト。1 つのディレクトリ・サーバーに格納されている。

パートナ・アプリケーション（partner application）

OracleAS Single Sign-On Server に認証機能を委譲する、Oracle Application Server アプリケーションまたは Oracle 以外のアプリケーション。このようなアプリケーションでは、**mod_osso** ヘッダーを受け取るので、ユーザーを再認証する必要がない。

バインド（binding）

ネットワークの場合は、通信エンティティ間の論理的な接続の確立を意味する。

Oracle Internet Directory では、バインドはディレクトリに対して認証を行うプロセスを表す。

SOAP メッセージを、相互に交換する目的で他のプロトコル（基礎となるプロトコル）内またはその上で伝送する、一定の形式に従った規則の組合せもバインドと呼ばれる。

ハッシュ（hash）

アルゴリズムを使用してテキスト文字列から生成される数値。ハッシュ値は、テキスト文字列より大幅に短くなる。ハッシュの数値は、セキュリティの目的とデータに対する高速アクセスの目的で使用する。

「**ハッシュ関数**」も参照。

ハッシュ関数（hash function）

暗号化におけるハッシュ関数または一方向ハッシュ関数は、特定のデータ・ブロックに適用されるアルゴリズムを意味する。ハッシュ関数の結果は、特定のデータ・ブロックの整合性を保証する目的で使用できる。ハッシュ関数が安全であるためには、既知のデータ・ブロックと既知の結果を与えられたときに、同じ結果となる別のデータ・ブロックを作成することがきわめて困難である必要がある。

判読可能データ（readable data）

暗号化を使用するときに暗号文に変換される前のデータ、または復号化を使用するときに暗号文から変換された結果のデータ。

ハンドシェイク（handshake）

2 台のコンピュータが通信セッションを開始するために使用するプロトコル。

非対称型アルゴリズム（asymmetric algorithm）

暗号化と**復号化**に異なる**鍵**を使用する**暗号化アルゴリズム**。

「**公開鍵暗号**」も参照。

非対称型暗号化（asymmetric cryptography）

「**公開鍵暗号**」を参照。

否認防止（non-repudiation）

暗号化において、特定の**デジタル署名**が特定のエンティティの**秘密鍵**によって生成されていることと、メッセージが特定の時点で改ざんされずに伝送されていることを保証する機能。

秘密鍵（private key）

公開鍵暗号で使用される**公開鍵と秘密鍵のペア**において、秘密にされる鍵。エンティティは自身の秘密鍵を使用して、**公開鍵**によって暗号化されたデータを復号化する。また、エンティティは秘密鍵を使用して、**デジタル署名**を作成することもできる。エンティティの公開鍵によって暗号化されたデータと、秘密鍵によって作成された署名のセキュリティは、秘密鍵の秘密が維持されていることに依存する。

秘密鍵（secret key）

対称型アルゴリズムで使用される**鍵**。秘密鍵は暗号化と復号化の両方に使用されるため、暗号文を相互に送受信するパーティ間で共有される必要があるが、許可されていないすべてのエンティティに対しては秘密が維持される必要がある。

秘密鍵暗号（secret key cryptography）

「**対称型暗号**」を参照。

平文（plaintext）

暗号化によって暗号文に変換される前の判読可能データ、または復号化によって暗号文から変換された結果の判読可能データ。

ファンアウト・レプリケーション（fan-out replication）

point-to-point レプリケーションとも呼ばれる。サプライヤがコンシューマに直接レプリケートするレプリケーションのタイプ。コンシューマは 1 つ以上の他のコンシューマにレプリケートできる。レプリケーションには、完全レプリケーションと部分レプリケーションがある。

フィルタ（filter）

ディレクトリに対するリクエストまたは検索結果として返されるエントリを定義する式。フィルタは、多くの場合、`cn=susie smith,o=acme,c=us` のような識別名で表される。

フェイルオーバー（failover）

障害を認識し、リカバリする処理。Oracle Application Server Cold Failover Cluster（Identity Management）で、1 つのクラスタ・ノード上で実行されているアプリケーションは、他のクラスタ・ノードに透過的に移行される。この移行時に、クラスタ上のサービスにアクセスするクライアントは一時的に接続できず、フェイルオーバーが完了した後、再接続する必要がある場合がある。

フェデレーション（federation）

「**ID フェデレーション**」を参照。

フェデレーテッド ID 管理（federated identity management: FIM）

自律型ドメイン内で ID と資格をポータブルにするための協定、標準およびテクノロジ。FIM によって、複数ドメイン全体でユーザー認証が認識可能になり、認証済ユーザーが複数ドメイン内のパーソナライズされたサービスに参加可能になる。FIM は、複数のアカウント間で ID 情報をリンク可能にすることで、個人情報が一箇所に格納されることの危険性を回避する。フェデレーテッド ID には、トラストと標準という 2 つの主要なコンポーネントが必要である。フェデレーテッド ID 管理のトラスト・モデルは、**トラスト・サークル**に基づく。標準は、**Liberty Alliance** Project によって定義される。

復号化（decryption）

暗号化されたメッセージ（暗号文）の内容を、元の可読書式（平文）に変換する処理。

プライマリ・ノード（primary node）

Oracle Application Server Cold Failover Cluster（Identity Management）で、指定した時間にアプリケーションが実行されるクラスタ・ノード。

「**セカンダリ・ノード**」も参照。

プリンシパル（principal）

OSFS でサポートされている **ID フェデレーション**・プロトコルに定義された 3 つの主要な役割の 1 つ。残りの 2 つは、**ID プロバイダ**および**サービス・プロバイダ**。

プリンシパルは、サービスを使用し、フェデレーテッド ID を取得できるエンティティである。通常、プリンシパルは人またはユーザー、あるいはその ID が認証可能なシステム・エンティティである。

プロキシ・サーバー（proxy server）

Web ブラウザなどのクライアント・アプリケーションと実サーバーの間にあるサーバー。プロキシ・サーバーは、実サーバーに対するすべてのリクエストを代理受信して、自分がそのリクエストを処理できるかどうかを調べる。処理できない場合、リクエストは実サーバーに転送される。OracleAS Single Sign-On では、プロキシは、ロード・バランシング目的とセキュリティ対策用の追加層として使用される。

「**ロード・バランサ**」も参照。

プロキシ・ユーザー（proxy user）

通常、ファイアウォールなどの中間層を備えた環境で利用されるユーザー。このような環境では、エンド・ユーザーは中間層に対して認証を行う。この結果、中間層はエンド・ユーザーにかわってディレクトリにログインする。プロキシ・ユーザーには ID を切り替える権限があり、一度ディレクトリにログインすると、エンド・ユーザーの ID に切り替える。次に、その特定のエンド・ユーザーに付与されている認可を使用して、エンド・ユーザーのかわりに操作を実行する。

ブロック暗号（block cipher）

対称型アルゴリズムの一種。ブロック暗号では、メッセージを固定サイズのブロック（一般的には 64 ビット）に分割し、各ブロックを鍵によって暗号化する方法でメッセージが暗号化される。広く一般に知られているブロック暗号には、**Blowfish**、**DES** および **AES** などがある。

「**ストリーム暗号**」も参照。

プロビジョニング（provisioning）

エンタープライズ環境で使用可能なアプリケーションおよびその他のリソースへのアクセスをユーザーに付与するプロセス。

プロビジョニング・アプリケーション（provisioned applications）

ユーザーおよびグループの情報が Oracle Internet Directory に一元化される環境にあるアプリケーション。これらのアプリケーションは、一般的に Oracle Internet Directory 内の該当する情報に対する変更に関心がある。

プロビジョニング・エージェント（provisioning agent）

Oracle 固有のプロビジョニング・イベントを外部またはサード・パーティのアプリケーション固有のイベントに変換するアプリケーションまたはプロセス。

プロビジョニング統合プロファイル（provisioning integration profile）

Oracle Directory Integration Platform がディレクトリ対応アプリケーションに送信するプロビジョニング関連通知の性質を記述した特殊な**ディレクトリ統合プロファイル**。

プロファイル（profile）

「**ディレクトリ統合プロファイル**」を参照。

変更ログ（change logs）

ディレクトリ・サーバーに加えられた変更を記録するデータベース。

ポリシーの優先順位（policy precedence）

Oracle Application Server Certificate Authority（OCA）では、メイン・ポリシー・ページに表示されている順番で、ポリシーが受信リクエストに適用される。OCA ポリシー・プロセッサ・モジュールがポリシーを解析する際、ポリシー・リストの上部にあるポリシーが最初にリクエストに適用される。ポリシー・リストの下部にあるポリシーは最後に適用され、他のポリシーよりも優先される。有効なポリシーのみが受信リクエストに適用される。

マスター・サイト（master site）

レプリケーションにおいて、**マスター定義サイト**以外のサイトで、LDAP レプリケーションのメンバーであるサイト。

マスター定義サイト（master definition site: MDS）

レプリケーションにおいて、管理者が構成スクリプトを実行する Oracle Internet Directory のデータベース。

マッピング・ルール・ファイル（mapping rules file）

Oracle Directory Integration Platform 環境で、Oracle Internet Directory 属性と**接続ディレクトリ**の属性との間のマッピングを指定するファイル。

マルチマスター・レプリケーション（multimaster replication）

peer-to-peer または n-way レプリケーションとも呼ばれる。同等に機能する複数のサイトがレプリケートされたデータのグループを管理できるようにするレプリケーションのタイプ。マルチマスター・レプリケーション環境では、各ノードはサプライヤ・ノードであると同時にコンシューマ・ノードであり、各ノードでディレクトリ全体がレプリケートされる。

メタディレクトリ（metadirectory）

企業のすべてのディレクトリ間で情報を共有するディレクトリ・ソリューション。すべてのディレクトリを 1 つの仮想ディレクトリに統合する。集中的に管理できるため、管理コストを削減できる。ディレクトリ間でデータが同期化されるため、企業内のデータに一貫性があり最新であることが保証される。

メッセージ・ダイジェスト（message digest）

ハッシュ関数の結果。

「ハッシュ」も参照。

メッセージ認証（message authentication）

特定のメッセージが特定のエンティティから発信されたことを検証するプロセス。

「**認証**」も参照。

メッセージ認証コード（message authentication code: MAC）

メッセージ認証コード（MAC）は、特定のデータ・ブロックに対して 2 ステップのプロセスを適用して得られる。最初に、**ハッシュ関数**の結果を取得する。次に、その結果を**秘密鍵**を使用して暗号化する。MAC は、特定のデータ・ブロックのソースの認証に使用できる。

ユーザー検索ベース（user search base）

Oracle Internet Directory のデフォルトの**ディレクトリ情報ツリー**で、すべてのユーザーが配置される ID 管理レルムのノード。

ユーザー名マッピング・モジュール（user name mapping module）

ユーザー**証明書**とユーザーのニックネームにマップする OracleAS Single Sign-On の Java モジュール。マップ後、ニックネームは認証モジュールに渡される。認証モジュールはこのニックネームを使用して、ディレクトリからユーザーの証明書を取得する。

猶予期間ログイン（grace login）

パスワード期限切れ前の指定された期間内に行われるログイン。

リモート・マスター・サイト（remote master site: RMS）

レプリケート環境における**マスター定義サイト**以外のサイトで、**Oracle Database アドバンスト・レプリケーション**のメンバーであるサイト。

リレーショナル・データベース（relational database）

構造化されたデータの集合。同一の列のセットを持つ 1 つ以上の行で構成される表にデータが格納される。Oracle では、複数の表のデータを容易にリンクできる。このため、Oracle はリレーショナル・データベース管理システム、すなわち RDBMS と呼ばれる。Oracle はデータを複数の表に格納し、さらに表間の関係を定義できる。このリンクは両方の表に共通の、1 つ以上のフィールドに基づいて行われる。

ルート CA（root CA）

ルート**認証局**のこと。階層構造を持つ**公開鍵インフラストラクチャ**において、その**公開鍵**がセキュリティ・ドメイン全体の最も信頼できるデータとして機能する CA。

ルート DSE（root DSE）

「**ルート・ディレクトリ固有のエントリ**」を参照。

ルート Oracle コンテキスト（root Oracle Context）

Oracle Identity Management インフラストラクチャでは、ルート Oracle コンテキストは、インフラストラクチャのデフォルト ID 管理レルムへのポインタを含む Oracle Internet Directory のエントリである。単純な名前を指定して ID 管理レルムの位置を特定する方法の詳細も含まれる。

ルート・ディレクトリ固有のエントリ（root directory specific entry: root DSE）

ディレクトリに関する操作情報を格納するエントリ。情報は複数の属性に格納されている。

レガシー・アプリケーション（legacy application）

認証を OracleAS Single Sign-On サーバーに委任するように変更できない古いアプリケーション。**外部アプリケーション**と呼ばれることもある。

レジストリ・エントリ（registry entry）

Oracle Internet Directory サーバーの起動（**ディレクトリ・サーバー・インスタンス**と呼ばれる）に関連する実行時情報が含まれているエントリ。レジストリ・エントリはディレクトリ自体に格納され、対応するディレクトリ・サーバー・インスタンスが停止するまで保持される。

レスポンス時間（response time）

リクエストの発行からレスポンスの完了までの時間。

レプリカ（replica）

ネーミング・コンテキストの個々のコピー。1 つのサーバー内に格納されている。

レプリケーション承諾（replication agreement）

ディレクトリ・レプリケーション・グループ内のディレクトリ・サーバー間におけるレプリケーションの関係を記述する特別なディレクトリ・エントリ。

レルム（realm）

「**ID 管理レルム**」を参照。

レルム検索ベース（realm search base）

すべての **ID 管理レルム**を含む**ディレクトリ情報ツリー**内のエントリを識別する**ルート Oracle コンテキスト**での属性。この属性は、単純なレルム名をディレクトリ内の対応するエントリにマッピングする際に使用される。

ロード・バランサ（load balancer）

過剰負荷またはフェイルオーバーにより、複数のサーバー間で接続リクエストを振り分ける、ハードウェア・デバイスおよびソフトウェア。BigIP、Alteon、Local Director などは、一般的なハードウェア・デバイスである。ロード・バランシング・ソフトウェアには、Oracle Application Server Web Cache がある。

論理ホスト（logical host）

Oracle Application Server Cold Failover Cluster（Identity Management）で、1 つ以上のディスク・グループおよびホスト名と IP アドレスのペア。論理ホストは、クラスタ内の物理ホストにマップされる。この物理ホストは、論理ホストのホスト名と IP アドレスを使用する。

索引

P

S

T

U

W

X

あ

え

か

は

ふ

ほ

ま

ゆ

り

ろ