# Oracle® Application Server Certificate Authority

管理者ガイド

10*g*（9.0.4）

**部品番号 : B12374-02**

2004 年 6 月

ORACLE®

Oracle Application Server Certificate Authority 管理者ガイド , 10*g*（9.0.4）

部品番号 : B12374-02

原本名 : Oracle Application Server Certificate Authority Administrator's Guide, 10*g* (9.0.4)

原本部品番号 : B10663-02

原本著者 : Jeffrey E. Levinger

原本協力者 : Lakshmi Kethana, Mode Nalini, Paul Needham, Shreedhar Patwari, Deepak Ramakrishnan, Gary Truong, Miranda Zhai

# 目次


**はじめに** ..... xiii

対象読者 ..... xiv
Oracle Identity Management ..... xiv
このマニュアルの構成 ..... xv
関連ドキュメント ..... xvii
表記規則 ..... xviii

**1　公開鍵インフラストラクチャと OracleAS**

**PKI とは** ..... 1-2
鍵のペア ..... 1-2
認証局（CA）およびデジタル証明書 ..... 1-3
CA 署名 ..... 1-3
信頼のレベル ..... 1-3
デジタル証明書の内容および使用方法 ..... 1-4
PKI 資格証明のコンテナ ..... 1-5
登録局（RA） ..... 1-6
**PKI の利点** ..... 1-6
**OracleAS PKI の概要** ..... 1-7
以前のコストおよび問題 ..... 1-7
OracleAS PKI の利点 ..... 1-7
OracleAS PKI のコンポーネント ..... 1-8
コンテナ、Oracle Wallet および Oracle Wallet Manager（OWM） ..... 1-8
Secure Sockets Layer（SSL） ..... 1-9
Oracle Internet Directory および Single Sign-On（SSO） ..... 1-9
Oracle Application Server Certificate Authority ..... 1-9

## 2 識別情報管理および OracleAS Certificate Authority の機能


**識別情報管理の構成要素とアーキテクチャ** ..... 2-2
Oracle Identity Management ..... 2-3
企業での Oracle Identity Management の使用 ..... 2-4
Oracle セキュリティ・アーキテクチャでの Oracle Identity Management の役割 ..... 2-5
Oracle Identity Management での OracleAS Certificate Authority の役割 ..... 2-6
SSO 統合を介した簡易プロビジョニング ..... 2-6
**Oracle Application Server Certificate Authority の主要機能** ..... 2-7
オープン規格に対するサポート ..... 2-7
柔軟なポリシー ..... 2-7
管理者およびエンド・ユーザーにとっての使いやすさ ..... 2-8
OCA 画面での National Language Support（NLS） ..... 2-8
スケーラビリティ、パフォーマンスおよび高可用性 ..... 2-9
**自動または従来型のプロビジョニング** ..... 2-9
OracleAS Single Sign-On 認証 ..... 2-10
Secure Sockets Layer（SSL）を使用した証明書ベースの認証 ..... 2-10
手動による承認 ..... 2-10
**階層的な認証局のサポート** ..... 2-11
**配置およびインストール** ..... 2-12


## 3 OCA および証明書の管理の概要


**Oracle Application Server Certificate Authority の起動および停止** ..... 3-2
**管理者の証明書の要求** ..... 3-3
**管理者の証明書の置換** ..... 3-7
**OracleAS Certificate Authority 管理インタフェースの概要** ..... 3-8
「認証管理」タブ ..... 3-10
**証明書の管理** ..... 3-11
証明書要求の承認または拒否 ..... 3-12
証明書要求の承認方法 ..... 3-12
証明書要求の拒否方法 ..... 3-12
証明書の詳細の表示 ..... 3-12
証明書の失効 ..... 3-13
証明書の更新 ..... 3-14
単一の証明書要求または発行済証明書の表示 ..... 3-14
拡張検索の使用方法 ..... 3-15

要求ステータスを使用した証明書要求の検索 ..... 3-16
識別名（DN）を使用した検索 ..... 3-16
拡張 DN を使用した検索 ..... 3-17
シリアル番号の範囲を使用した検索 ..... 3-17
証明書のステータスを使用した検索 ..... 3-17
**証明書失効リスト（CRL）の更新** ..... 3-18
**Single Sign-On（SSO）および OracleAS Certificate Authority（OCA）** ..... 3-19
SSO 認証済ユーザーへの OCA 証明書要求 URL のブロードキャスト ..... 3-20
OCA 証明書要求 URL への SSO 認証済ユーザーのアクセス ..... 3-20
ユーザー証明書と SSO の使用 ..... 3-22
**OracleAS Certificate Authority のインストールのデフォルト値** ..... 3-23
SSO および OCA での PKI 認証の有効化 ..... 3-25

## 4 Oracle Application Server Certificate Authority の構成

**管理インタフェースの構成** ..... 4-2
**「構成管理」タブ** ..... 4-3
構成タスクの概要 ..... 4-4
「通知」サブタブ ..... 4-5
メール詳細 ..... 4-5
アラート ..... 4-6
スケジュールされたジョブ ..... 4-6
電子メールのテンプレート ..... 4-7
トークンの値 ..... 4-8
「一般」サブタブ ..... 4-10
証明書の公開 ..... 4-10
SSL 認証および SSO 認証 ..... 4-10
ロギングおよびトレース ..... 4-11
デフォルトのベース DN コンポーネント ..... 4-11
データベースの設定 ..... 4-12
ディレクトリの設定 ..... 4-12
**「ログの表示」タブ** ..... 4-13
**認証局運用規程の作成および更新** ..... 4-14

## 5 Oracle Application Server Certificate Authority でのポリシー管理

**定義** ..... 5-2
**ポリシー管理の概要** ..... 5-2

**Oracle Application Server Certificate Authority のポリシー** ..... 5-4
RSAKeyConstraints ..... 5-4
ValidityRule ..... 5-6
UniqueCertificateConstraint ..... 5-8
RevocationConstraints ..... 5-10
RenewalRequestConstraint ..... 5-11
**Oracle Application Server Certificate Authority の「ポリシー」サブタブ** ..... 5-14
製品に付属の証明書要求ポリシー ..... 5-16
製品に付属の証明書失効ポリシー ..... 5-16
製品に付属の証明書更新ポリシー ..... 5-16
ポリシー操作 ..... 5-16
編集 ..... 5-17
有効化または無効化 ..... 5-17
削除 ..... 5-17
ポリシーの並び替え ..... 5-18
ポリシーの追加 ..... 5-20
**ポリシー・ルールの条件** ..... 5-21
複数の条件による評価 ..... 5-25
複数の条件による評価の例 ..... 5-25
複数の条件による評価の例 2 ..... 5-26
条件の並び替え ..... 5-26
条件の追加 ..... 5-28
**カスタム・ポリシー・プラグインの開発** ..... 5-30
ポリシーにより実行される処理について ..... 5-30
新しいポリシー・プラグインを作成する手順 ..... 5-31
カスタム・ポリシー・プラグインの例 ..... 5-33
汎用エラー・メッセージ ..... 5-35

## 6 OracleAS Certificate Authority の管理 : 高度なトピック

**OracleAS Certificate Authority の Wallet 操作** ..... 6-2
CA 署名 Wallet の再生成 ..... 6-2
CA SSL Wallet および CA SMIME Wallet の再生成 ..... 6-3
CA SMIME Wallet ..... 6-3
重要な Wallet の更新 ..... 6-4
パスワードの変更 ..... 6-5
**OracleAS Certificate Authority の構成操作** ..... 6-5

第三者の SSL Wallet を使用するための Oracle HTTP Server の構成 ..... 6-6
認証局の証明書の失効 ..... 6-7
OCA Web 管理者の証明書の失効 ..... 6-8
OCA 画面での National Language Support（NLS）の構成 ..... 6-9
**カスタマイズのサポート** ..... 6-10
**Oracle Application Server Certificate Authority での OCA アクションのログまたはトレース** ..... 6-13
OracleAS Certificate Authority のログ情報またはトレース情報の消去 ..... 6-14
**OCA が使用するインフラストラクチャ・サービスの変更** ..... 6-15
OCA が使用する Identity Management（IM）サービス（SSO/OID）の変更 ..... 6-15
OCA が使用する Metadata Repository（MR）サービスの変更 ..... 6-17
OCA の接続情報の格納場所および表示場所 ..... 6-17
**OracleAS Certificate Authority および高可用性機能** ..... 6-17
コールド・フェイルオーバーを使用した OracleAS Certificate Authority の配置 ..... 6-18
Real Application Clusters を使用した OracleAS Certificate Authority の配置 ..... 6-18
**OracleAS Certificate Authority のバックアップおよびリカバリでの考慮事項** ..... 6-19
**証明書公開レルムの制限** ..... 6-22
**CA の置換および OracleAS Certificate Authority の削除** ..... 6-23

## 7　Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース

**ユーザー・インタフェースへのアクセス** ..... 7-2
**エンド・ユーザー用のタブおよび処理** ..... 7-3
「ユーザー証明書」タブ ..... 7-4
Single Sign-On（SSO）認証 ..... 7-5
OracleAS Certificate Authority が信頼されるブラウザの構成 ..... 7-7
Internet Explorer での証明書発行元への信頼 ..... 7-7
Netscape での証明書発行元への信頼 ..... 7-7
Secure Sockets Layer（SSL）認証 ..... 7-9
手動認証 ..... 7-10
証明書の検索、更新および失効 ..... 7-11
証明書の取得 ..... 7-11
証明書の更新 ..... 7-11
証明書の失効 ..... 7-11
「サーバー / 下位 CA 証明書」タブ ..... 7-12
下位 CA 証明書 ..... 7-12
**CA 証明書のダウンロード** ..... 7-13

**ブラウザへの証明書失効リスト（CRL）のインポート** ..... 7-13
Netscape の場合 ..... 7-14
Internet Explorer（IE）の場合 ..... 7-14
**ファイル・システムへの証明書失効リスト（CRL）のダウンロード** ..... 7-15
**ブラウザへの新規発行の証明書のインポート** ..... 7-15
**ブラウザからの Wallet のエクスポート（バックアップ）** ..... 7-16
**ファイル・システムからの証明書のインポート** ..... 7-18

## A コマンドライン管理

**コマンドライン・ツール** ..... A-2
Convertwallet の例 ..... A-7
**Oracle Certificate Authority Server の起動** ..... A-8
**Oracle Application Server Certificate Authority Server の停止** ..... A-8
**Oracle Application Server Certificate Authority サービスの状態の検索** ..... A-9
**権限付きパスワードの変更** ..... A-9
**ルート認証局の証明書の再生成** ..... A-11
**認証局の SSL 証明書および Wallet の再生成** ..... A-12
**ルート CA 証明書の失効** ..... A-12
**CA SSL サーバー Wallet の SSO 形式への変換** ..... A-14
**Oracle Application Server Certificate Authority からの下位 CA Wallet の生成** ..... A-14
**下位 CA Wallet のインストール / インポート** ..... A-15
**下位 CA 用の CA SSL Wallet の生成** ..... A-16
**ログまたはトレース記憶域の消去** ..... A-17
**OCA リポジトリ接続情報の更新** ..... A-17
**SSO 認証の設定（linksso および unlinksso コマンド）** ..... A-18
**ログ / トレース・オプションの設定** ..... A-18

## B CA の階層の設定

**下位 CA Wallet の生成** ..... B-2
**新しい下位 CA Wallet のインストールおよび使用** ..... B-3
別の CA の下位 CA にするための OCA インスタンスの構成 ..... B-5
下位 CA 用の CA SSL Wallet および CA SMIME Wallet の生成 ..... B-6

## C トラブルシューティングの既知のヒント

**1. 基礎的な問題および警告** ..... C-3
a. 問題 : 証明書要求で鍵のペアが生成されない（Windows）。 ..... C-3

b. 問題 : 通常のユーザーでログインした後、管理者でログインできない。 ..... C-3
c. 問題 : パスワードの変更には、OCA のコマンドライン・ツール **ocactl** を使用する必要がある。 ..... C-3
**2. ブラウザの問題** ..... C-4
a. 問題 : CA SSL サーバーの CN がマシン名と一致しない場合に、ブラウザが警告を表示する。 ..... C-4
b. 問題 : 最初（最も右側）の CN 構成要素しか使用されない。 ..... C-4
c. Netscape の場合 ..... C-4
i. 問題 : 複数の証明書が使用可能であるにもかかわらず、ポップアップ・ウィンドウに証明書が 1 つしか表示されない。 ..... C-4
ii. 問題 : CA 証明書が信頼できるかどうかの質問が引き続き表示される。 ..... C-4
iii. 問題 : 証明書の有効期限が切れたという警告が表示される。 ..... C-5
iv. 問題 : 下位 CA と CA SSL 両方のクライアント証明書が表示される。 ..... C-5
d. Internet Explorer（IE）の場合 ..... C-5
i. 問題 :「ページを表示できません」というメッセージが表示される。 ..... C-5
ii. 問題 : ブラウザに CRL をインポートできない。 ..... C-5
iii. 問題 : セキュアな情報とセキュアでない情報の両方がページに含まれているというメッセージが表示される。 ..... C-5
iv. 問題 : オンライン・ヘルプを開くと、セキュリティ・アラートが生成される。 ..... C-6
**3. ネットワークの問題** ..... C-6
a. 問題 : SSO ユーザー名 / パスワードを使用して OCA にログインすると、エラー・メッセージが表示される。 ..... C-6
b. 問題 : ネットワーク・エラーのメッセージが表示される。 ..... C-7
c. 問題 : OCA が動作しなくなる。あるいはネットワークまたはサーバーのメッセージが表示される。 ..... C-7
**4. 証明書の問題** ..... C-8
a. 問題 : ユーザー証明書をインポートしても CA 証明書がインポートされない（Netscape）。 ..... C-8
b. 問題 :「認証管理」タブへのアクセスまたは使用ができない。 ..... C-8
c. 問題 : 管理者が別のマシンから作業する必要がある。 ..... C-8
**5. シングル・サインオン（SSO）の問題** ..... C-9
a. 問題 : SSO の証明書に表示される名前が「USER」になる。 ..... C-9
b. 問題 : 鍵の生成中に VB スクリプトのエラー・メッセージが表示される。 ..... C-9
c. 問題 :「ページを表示できません」というメッセージが表示される（Internet Explorer）。 ..... C-9
d. 問題 : Internet Explorer で SSO ログイン・ページに進むと、セキュリティ警告ダイアログが表示される。 ..... C-10
**6. 検索の問題** ..... C-10
a. 問題 : 検索画面で [Enter] を押すと内部エラーが発生する。 ..... C-10

**7. バックアップの保護の問題** ..... C-10
a. 問題 : OCA の内部リポジトリのリカバリ可能性を保証する。 ..... C-10
**8. 一般的な問題** ..... C-11
a. 問題 : ページのロードに時間がかかりすぎる、またはハングアップする。 ..... C-11
b. 問題 : 新しい Web 管理者を登録すると JAZN エラーが生じる。 ..... C-11
c. 問題 : Outlook Express に SMIME 署名証明書が表示されない。 ..... C-11
d. 問題 : CA SSL サーバーの CN について、警告が表示される。 ..... C-11

## D 拡張領域

## E SSO での SSL および PKI の有効化

**SSO での SSL の有効化** ..... E-2
**SSO での PKI の有効化** ..... E-4
**SSL を有効化した SSO への、OCA の仮想ホストの再登録** ..... E-5
OCA の再登録の例 ..... E-6

## F 用語集

## 索引

# 表リスト


3-1 管理者の証明書の DN 情報 ..... 3-4
3-2 検索要素 ..... 3-16
3-3 証明書のシリアル番号の検索範囲を指定する要素 ..... 3-17
3-4 Wallets、CRL および OHS ポートに対するインストールの値（注記 1 を参照） ..... 3-24
4-1 「構成管理」の「通知」サブタブのタスクおよび説明 ..... 4-4
4-2 「構成管理」の「一般」サブタブのタスクおよび説明 ..... 4-4
4-3 「構成管理」の「ポリシー」サブタブのタスクおよび説明 ..... 4-4
4-4 通知、テンプレート、および電子メールのカスタマイズに使用できるトークン ..... 4-7
4-5 通知およびテンプレートのカスタマイズに使用できるトークンの値 ..... 4-8
5-1 OracleAS Certificate Authority でのポリシーの概念、用語および定義 ..... 5-2
5-2 制約固有のデフォルトのポリシー・ルール ..... 5-4
5-3 RSAKeyConstraints ポリシー・ルールのパラメータ ..... 5-4
5-4 ValidityRule ポリシーのパラメータ ..... 5-7
5-5 UniqueCertificateConstraint ポリシー・ルールのパラメータ ..... 5-9
5-6 RevocationConstraints ポリシー・ルールのパラメータ ..... 5-11
5-7 RenewalConstraints ポリシー・ルールのパラメータ ..... 5-11
5-8 論理演算子 ..... 5-22
5-9 条件の属性 ..... 5-23
5-10 カスタム・ポリシー・プラグインの処理手順 ..... 5-31
6-1 Single Sign-On のポップアップ画面のカスタマイズ ..... 6-12
6-2 OCA のログ・データおよびトレース・データの格納場所 ..... 6-13
6-3 バックアップ / リカバリの使用例 ..... 6-20
6-4 Backup and Recovery Tool ..... 6-21
7-1 証明書のタイプおよび使用方法 ..... 7-3
7-2 認証タイプ ..... 7-4
A-1 コマンドおよび構成操作へのリンク ..... A-2
A-2 OracleAS Certificate Authority（OCA）ocactl ツールの操作およびパラメータ ..... A-3
A-3 パスワードのタイプおよび使用方法 ..... A-9
A-4 権限付きロールおよび setpasswd コマンド ..... A-10
A-5 `revokecert` コマンドで使用する失効理由 ..... A-13
F-1 OracleAS Certificate Authority で使用される用語の定義 ..... F-1

# 図リスト


1-1 Oracle Application Server Certificate Authority が発行する証明書 ..... 1-5
2-1 企業識別管理ソリューションのモデル ..... 2-2
2-2 企業統合型の識別情報管理 ..... 2-4
2-3 Oracle Identity Management のセキュリティ・モデル ..... 2-5
2-4 Oracle Application Server Certificate Authority のデフォルトのインストール ..... 2-13
2-5 OracleAS Certificate Authority の推奨される本番インストール ..... 2-14

# はじめに

Oracle Application Server Certificate Authority（OCA）では、PKI（公開鍵インフラストラクチャ）テクノロジに基づいて、デジタル証明を発行および管理できます。Oracle Application Server Certificate Authority が提供する簡単な管理方法によってこの証明を行うと、セキュリティが向上し、ユーザー認証にかかる時間とリソースが削減されます。

Oracle Application Server Certificate Authority を使用すると、エンド・エンティティ（ユーザーおよびサーバー）は自分自身を認証できます。この認証では、SSO、SSL またはその他の既存の認証方式に基づいて OCA が発行する証明書を使用します。これらの証明書を使用すると、証明書を識別することによって認証をより速く、より安全に処理できます。各証明書は、発行時に OID に公開され、期限切れまたは失効時に削除されます。ユーザーは、OCA の Web インタフェースにアクセスして、ユーザー自身の証明書の発行、失効または更新をリクエストできます。エンド・ユーザーが OCA の Web インタフェースにアクセスするために特別な権限は必要ありません。ただし、証明書の発行、失効または更新を行うには、OCA から以前に発行された証明書を使用して、SSO または SSL による認証を済ませておく必要があります。これが済んでいないと、OCA の管理者が手動で認証することが必要になります。

このマニュアルでは、公開鍵証明書の管理方法について説明します。

ここでは、次の項目について説明します。


- 対象読者
- Oracle Identity Management
- このマニュアルの構成
- 関連ドキュメント
- 表記規則

# 対象読者

このマニュアルは、証明書リクエストおよび証明書関連の操作を管理する Oracle Application Server Certificate Authority の管理者を対象としています。

# Oracle Identity Management

Oracle Application Server Certificate Authority（OCA）は、Oracle Identity Management のコンポーネントです。Oracle Identity Management は統合されたインフラストラクチャであり、Oracle 製品および他のエンタープライズ・アプリケーションに対して分散セキュリティ・サービスを提供します。Oracle Identity Management インフラストラクチャには、次のコンポーネントおよび機能が含まれます。

- **Oracle Internet Directory**（OID）。Oracle Database に実装されている、スケーラブルで強力な LDAP V3 準拠のディレクトリ・サービスです。
- **Oracle Directory Integration and Provisioning**。Oracle Internet Directory の一部で、Oracle Internet Directory などのディレクトリとユーザー・リポジトリの同期を可能にします。また、Oracle のコンポーネントおよびアプリケーションに対して自動プロビジョニング・サービスを提供します。標準インタフェースを使用して、サード・パーティのアプリケーションに対しても同様のサービスを提供します。
- **Oracle Delegated Administration Services**。Oracle Internet Directory の一部で、ユーザーおよびアプリケーション管理者が信頼できる、プロキシ・ベースのディレクトリ情報管理を提供します。
- **Oracle Application Server Single Sign-On**（SSO）。Oracle およびサード・パーティの Web アプリケーションに対して、シングル・サインオン・アクセスを提供します。
- **Oracle Application Server Certificate Authority**。X.509 バージョン 3（以降 v3）PKI 証明書を生成して公開し、厳密な認証方式、保護メッセージなどをサポートします。

  SSL、OC4J および HTTP Server を使用することに加えて、Oracle Application Server Certificate Authority は、SSO および OID に依存するように構成されています。OCA は、使用中の DN の OID エントリにすべての有効な証明書を公開し、Netscape、Internet Explorer または Mozilla による証明書の登録およびダウンロードをサポートします。OCA では、失効した証明書は即座に、期限切れの証明書は定期的に OID から削除されるため、SSO などのコンポーネントはこれらの OID エントリに依存できます。また、管理者も、SSO を使用して URL を公開できるように OCA を構成することができます。この構成を選択すると、証明書を所有していないすべての SSO 認証済ユーザーが、証明書リクエスト用の OCA ページを表示できます。OCA 証明書は、すべての Oracle コンポーネントの認証、または SSO 対応のすべてのアプリケーションの使用の認可に使用できます。

通常のエンタープライズ・アプリケーションの配置では、1 つの Oracle Identity Management インフラストラクチャが複数のサーバーおよびコンポーネント・インスタンスで構成されて配置されます。この構成には、高可用性、情報のローカライズ、委任コンポーネント管理などのメリットがあります。企業に配置された各追加アプリケーションは、識別情報管理サービスに共有インフラストラクチャを使用します。この配置モデルには次のメリットがあります。

- **一時的な作業 :** Identity Management インフラストラクチャの計画および実装は、エンタープライズ・アプリケーションを配置するたびに行う必要はなく、一時的な作業になります。その結果、ポータル、J2EE アプリケーション、E-Business アプリケーションなどの新しいアプリケーションを、迅速に配置できます。
- **集中管理 :** 複数の場所で管理されている場合でも、識別情報を集中管理します。変更は、すべてのエンタープライズ・アプリケーションですぐに有効になります。
- **ユーザーのシングル・サインオン :** 集中セキュリティ・インフラストラクチャを使用すると、エンタープライズ・アプリケーション全体でユーザーのシングル・サインオンを実現できます。
- **単一の統合ポイント :** Identity Management インフラストラクチャを一元化すると、企業の Oracle 環境と他の識別情報管理システムの間に単一の統合ポイントが提供されます。これによって、複数の「ポイントツーポイント」のカスタム統合ソリューションを使用する必要がなくなります。

Oracle Identity Management インフラストラクチャの計画、配置および使用の詳細は、『Oracle Identity Management 概要および配置プランニング・ガイド』を参照してください。

OCA のデフォルトの配置構成は、Oracle Application Server 10*g* のインストレーション・ガイドの第 6.20 項を参照してください。推奨の配置構成およびインストール手順は、そのマニュアルの第 11.9 項を参照してください。

# このマニュアルの構成

このマニュアルは、7 つの章と 5 つの付録で構成されています。

### 第 1 章「公開鍵インフラストラクチャと OracleAS」

この章では、公開鍵インフラストラクチャおよびその Oracle での実装の概要について説明します。

### 第 2 章「識別情報管理および OracleAS Certificate Authority の機能」

この章では、業界標準の証明書の管理、LDAP ディレクトリと Single Sign-On との統合およびポリシーの適用に使用する主要機能およびインタフェース（スケーラブル、Web ブラウザ）について説明します。

### 第 3 章「OCA および証明書の管理の概要」

この章では、Web 管理者のインタフェースを使用して OCA および証明書を管理する方法について説明します。

### 第 4 章「Oracle Application Server Certificate Authority の構成」

この章では、証明書のリクエスト、更新または失効に使用する OCA ユーザー・インタフェースについて説明します。

### 第 5 章「Oracle Application Server Certificate Authority でのポリシー管理」

この章では、OCA で配信されたポリシーを管理または変更する方法、および証明書の発行、更新または失効のリクエストを処理するために新しいポリシーを作成および管理する方法について説明します。管理者は、Web インタフェースを使用してポリシーを変更できます。

### 第 6 章「OracleAS Certificate Authority の管理 : 高度なトピック」

この章では、Oracle Application Server Certificate Authority の要件、および Oracle Application Server の高可用性機能およびバックアップとリカバリの手順との相互運用性について説明します。

### 第 7 章「Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース」

この章では、エンド・ユーザーが証明書のリクエスト、取得、更新または取消しに使用する Web インタフェースについて説明します。

### 付録 A「コマンドライン管理」

この付録では、証明書の管理を含む管理用 `ocactl` コマンドライン・ツールのすべての構文および使用例を示します。

### 付録 B「CA の階層の設定」

この付録では、下位認証局の取得方法およびインポート方法について説明します。下位認証局とは、証明書が上位の CA 機関で署名される CA です。

### 付録 C「トラブルシューティングの既知のヒント」

この付録では、Oracle Application Server Certificate Authority のインストール、管理または使用中に発生する可能性がある問題またはエラー・メッセージに対する解決策およびその他の提案を示します。

### 付録 D「拡張領域」

この付録では、X.509 v3 および IETF の PKIX 標準拡張領域について説明します。Oracle Application Server Certificate Authority は、これらの拡張領域に準拠しています。

**付録 F「用語集」**
この付録では、OracleAS Certificate Authority に関連する、主要な用語や概念の定義を示します。

# 関連ドキュメント

- Oracle Application Server 10*g* のインストレーション・ガイド
- 『Oracle Application Server 10*g* 管理者ガイド』
- 『Oracle Application Server 10*g* セキュリティ・ガイド』
- 『Oracle Application Server Single Sign-On 管理者ガイド』
- 『Oracle Application Server 10*g* 高可用性ガイド』
- 『Oracle Database バックアップおよびリカバリ・アドバンスト・ユーザーズ・ガイド』
- 『Oracle Internet Directory 管理者ガイド』
- 『Oracle Advanced Security 管理者ガイド』

このマニュアルの多くの例で、Oracle インストール時にデフォルトでインストールされるシード・データベースのサンプル・スキーマを使用しています。これらのスキーマの作成方法および使用方法は、『Oracle Database サンプル・スキーマ』を参照してください。

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/`

# 表記規則

この項では、このマニュアルの本文およびコード例で使用される表記規則について説明します。この項の内容は次のとおりです。


- 本文の表記規則
- コード例の表記規則


## 本文の表記規則

本文では、特定の項目が一目でわかるように、次の表記規則を使用します。次の表に、その規則と使用例を示します。

| 規則 | 意味 | 例 |
|---|---|---|
| **太字** | 太字は、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。 | この句を指定すると、**索引構成表**が作成されます。 |
| 固定幅フォントの大文字 | 固定幅フォントの大文字は、システム指定の要素を示します。このような要素には、パラメータ、権限、データ型、Recovery Manager キーワード、SQL キーワード、SQL*Plus またはユーティリティ・コマンド、パッケージおよびメソッドがあります。また、システム指定の列名、データベース・オブジェクト、データベース構造、ユーザー名およびロールも含まれます。 | `NUMBER` 列に対してのみ、この句を指定できます。<br>`BACKUP` コマンドを使用して、データベースのバックアップを作成できます。<br>`USER_TABLES` データ・ディクショナリ・ビュー内の `TABLE_NAME` 列を問い合せます。<br>`DBMS_STATS.GENERATE_STATS` プロシージャを使用します。 |
| 固定幅フォントの小文字 | 固定幅フォントの小文字は、実行可能ファイル、ファイル名、ディレクトリ名およびユーザーが指定する要素のサンプルを示します。このような要素には、コンピュータ名およびデータベース名、ネット・サービス名および接続識別子があります。また、ユーザーが指定するデータベース・オブジェクトとデータベース構造、列名、パッケージとクラス、ユーザー名とロール、プログラム・ユニットおよびパラメータ値も含まれます。 | `sqlplus` と入力して、SQL*Plus をオープンします。<br>パスワードは、`orapwd` ファイルで指定します。<br>`/disk1/oracle/dbs` ディレクトリ内のデータ・ファイルおよび制御ファイルのバックアップを作成します。<br>`hr.departments` 表には、`department_id`、`department_name` および `location_id` 列があります。<br>`QUERY_REWRITE_ENABLED` 初期化パラメータを `true` に設定します。<br>`oe` ユーザーとして接続します。 |

| 規則 | 意味 | 例 |
|---|---|---|
| | **注意：**プログラム要素には、大文字と小文字を組み合せて使用するものもあります。これらの要素は、記載されているとおりに入力してください。 | JRepUtil クラスが次のメソッドを実装します。 |
| *固定幅フォントの小文字のイタリック* | 固定幅フォントの小文字のイタリックは、プレースホルダまたは変数を示します。 | *parallel_clause* を指定できます。<br>U*old_release*.SQL を実行します。ここで、*old_release* とはアップグレード前にインストールしたリリースを示します。 |

## コード例の表記規則

コード例は、SQL、PL/SQL、SQL*Plus または他のコマンドライン文の例です。次のように固定幅フォントで表示され、通常のテキストと区別されます。

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

次の表に、コード例で使用される表記規則とその使用例を示します。

| 規則 | 意味 | 例 |
|---|---|---|
| [ ] | 大カッコは、カッコ内の項目を任意に選択することを表します。大カッコは、入力しないでください。 | DECIMAL (*digits* [ , *precision* ]) |
| { } | 中カッコは、カッコ内の項目のうち、1 つが必須であることを表します。中カッコは、入力しないでください。 | {ENABLE \| DISABLE} |
| \| | 縦線は、大カッコまたは中カッコ内の複数の選択項目の区切りに使用します。項目のうちの 1 つを入力します。縦線は、入力しないでください。 | {ENABLE \| DISABLE}<br>[COMPRESS \| NOCOMPRESS] |
| ... | 水平の省略記号は、次のいずれかを示します。 | |
| | 例に直接関連しないコードの一部が省略されている。 | CREATE TABLE ... AS *subquery*; |
| | コードの一部を繰り返すことができる。 | SELECT *col1*, *col2*, ... , *coln* FROM employees; |

| 規則 | 意味 | 例 |
|---|---|---|
| .<br>.<br>. | 垂直の省略記号は、例に直接関連しない複数の行が省略されていることを示します。 | SQL> SELECT NAME FROM V$DATAFILE;<br>NAME<br>----------<br>/fsl/dbs/tbs_01.dbf<br>/fs1/dbs/tbs_02.dbf<br>.<br>.<br>.<br>/fsl/dbs/tbs_09.dbf<br>9 rows selected. |
| その他の記号 | 大カッコ、中カッコ、縦線および省略記号以外の記号は、記載されているとおりに入力する必要があります。 | acctbal NUMBER(11,2);<br>acct CONSTANT NUMBER(4) := 3; |
| *イタリック体* | イタリック体は、特定の値を指定する必要があるプレースホルダや変数を示します。 | CONNECT SYSTEM/*system_password*<br>DB_NAME = *database_name* |
| 大文字 | 大文字は、システム指定の要素を示します。これらの要素は、ユーザー定義の要素と区別するために大文字で示されます。大カッコ内にないかぎり、表示されているとおりの順序および綴りで入力します。ただし、大 / 小文字が区別されないため、小文字でも入力できます。 | SELECT last_name, employee_id FROM employees;<br>SELECT * FROM USER_TABLES;<br>DROP TABLE hr.employees; |
| 小文字 | 小文字は、ユーザー指定のプログラム要素を示します。たとえば、表名、列名またはファイル名などです。 | SELECT last_name, employee_id FROM employees;<br>sqlplus hr/hr |
| | **注意 :** プログラム要素には、大文字と小文字を組み合せて使用するものもあります。これらの要素は、記載されているとおりに入力してください。 | CREATE USER mjones IDENTIFIED BY ty3MU9; |

# 1

# 公開鍵インフラストラクチャと OracleAS

公開鍵インフラストラクチャ（PKI）は、パブリック・ネットワークおよびプライベート・ネットワーク上でセキュアな通信ができるように設計されています。さらに、PKI では、電子メールの保護、否認防止用のデジタル署名、データ整合性という重要な機能が実現されます。過去 25 年以上の間 PKI で問題となっていることの 1 つに、PKI に関連する必要なインフラストラクチャを配置できないことがあります。実際、このインフラストラクチャのコストおよび複雑さが、PKI が広範囲に使用されない主な原因となっています。

Oracle Identity Management インフラストラクチャは、高可用性、スケーラビリティ、ディレクトリ・サービス、シングル・サインオン、委任管理サービスおよびディレクトリ統合サービスを組み合せて、PKI に理想的な環境を提供します。このインフラストラクチャは、これらのメリットによって、Oracle Application Server Certificate Authority が常駐する理想的な場所になります。その結果、Oracle Application Server Certificate Authority は、Oracle Identity Management インフラストラクチャの一部となり、このインフラストラクチャが持つ集中管理機能とスケーラビリティによって、PKI を配置するコストと複雑さが軽減されます。

この章では、PKI について詳しく説明します。説明する内容は次のとおりです。


- PKI とは
- PKI の利点
- OracleAS PKI の概要

# PKI とは

PKI は次の要素を統合します。

- データの転送および格納を保護する暗号化アルゴリズム
- 異なるユーザーに対して一意の暗号化を実行できる暗号鍵
- 広範囲なネットワークで暗号をセキュアに使用でき、なおかつ、適切な受信者のみがセキュアに復号化できる鍵の配布方法
- 鍵とその正当な所有者の関係を保証する信頼できるエンティティ

こうした要素が一体になることで、この章で説明しているとおり、イントラネット、エクストラネットおよび E-Commerce アプリケーションに、高度なセキュリティが実現します。安全で信頼できるユーザー認証、データ整合性、署名されたメッセージの否認防止、転送または格納された情報への不正アクセスの防止などの利点があります。

この項では、PKI の主要機能について説明します。説明する内容は次のとおりです。

- 鍵のペア
- 認証局（CA）およびデジタル証明書
- 登録局（RA）

## 鍵のペア

暗号化は、データをわかりにくくして不正アクセスまたは改ざんから保護することを意味します。ただし、認可された受信者が元のデータに復元できる方法を使用します。元のデータを暗号化または置換する方法には、送信者と受信者のみが知っている鍵と呼ばれるテキストまたは数値を使用します。送信者と受信者の両方が同じ鍵を使用する場合、その暗号化方法は「対称型」と呼ばれます。対称型方式で暗号化する場合は、必要な機密性を損なわずに、傍受者が取得できないようにその鍵を送信者と受信者の両方に配布する方法が問題となります。また、送信者と受信者のペアごとに個別の鍵が必要となるため、各通信者が多くの鍵（受信者ごとに 1 つ）を保持する必要があるという問題もあります。

PKI の根幹は、秘密鍵と公開鍵のペアを使用することです。このペアは、公開鍵と秘密鍵が異なるため「非対称型」と呼ばれます。各ユーザーは、通信相手のユーザーの数に関係なく、鍵のペアを 1 つしか所有しません。

PKI の各鍵は、2 進数で構成されています。通常は 512 ～ 2048 ビットです。512 は弱い暗号、1024 は強い暗号で、2048 は軍事用です。アルゴリズムは、これらの鍵ビットとデータ・ビットを組み合せてデータを暗号化します。

鍵のペアの各所有者は、公開鍵は公開しますが、秘密鍵は公開しません。他のユーザーは、公開鍵を使用して、鍵のペアの所有者に送信するプライベート・メッセージを暗号化できます。それに対して、鍵のペアの所有者は、秘密鍵を使用してそのメッセージを復号化するか、または重要な送信メッセージに署名します。この方式の有効性は、公開鍵は簡単かつ安全に配布でき、復号化に必要な秘密鍵は共有しないという考えに基づいています。

# 認証局（CA）およびデジタル証明書

認証局は、公開鍵の所有者の識別情報を保証する信頼できる第三者機関です。このマニュアルで説明している Oracle Application Server Certificate Authority は、こうしたエンティティの 1 つです。他には VeriSign 社、Thawte 社などがあります。認証局は、デジタル証明書を作成して特定のユーザーへの公開鍵のリンクを検証します。このデジタル証明書には、公開鍵、および鍵の所有者と署名を行う認証局についての情報が含まれます。PKI の証明書を使用してユーザーの識別情報を認証することは、運転免許証やパスポートによる身元確認と類似しています。これらの証明書を忘れたり、変更することはほとんどあり得ないためです。

この項は次のトピックで構成されています。

- CA 署名
- 信頼のレベル
- デジタル証明書の内容および使用方法
- PKI 資格証明のコンテナ

## CA 署名

CA は、秘密鍵を使用してデジタル証明に署名します。この署名によって、すべてのユーザーが、CA の公開鍵を使用して、署名が認証されていて証明書が有効であることを検証できます。証明書が検証されると、証明書の所有者の公開鍵は、所有者に対するメッセージの暗号化またはメッセージに残した所有者の署名の検証に使用できます。

## 信頼のレベル

様々なレベルの CA が存在します。各 CA がより信頼できるソース（上位レベルの CA）から証明書を受信すると、信頼の階層が確立されます。ルート CA から下位 CA を経由して、下位レベルのトラスト・ポイントに至る信頼できる各リンク行は、高信頼パスと呼ばれます。

トップ・レベルの CA はルート CA と呼ばれ、信頼関係の原点となっています。ルート CA の下の CA は、下位 CA と呼ばれます。同じルート CA を共有するすべてのエンド・ユーザーは、最終的に同じ認証ソースを信頼しているため、信頼できる方法で相互に通信できます。

公開鍵とリンクされた識別情報が検証済であることを正式に表す証明書を信頼することは、証明書を発行した機関（CA）を信頼することを意味します。これに対して CA は、証明書の要求時に提供される情報の検証を、登録局（RA）という別のエンティティに任せています。

## デジタル証明書の内容および使用方法

Oracle Application Server Certificate Authority が発行するデジタル証明書は、ISO 規格の X.509 v3、および Internet Engineering Task Force（`http://www.ietf.org/`）の PKIX ワーキング・グループが公開している RFC 2459 に準拠します。

X.509 v3 規格では、SSL、暗号化およびデジタル署名用に別々の証明書を有効にする拡張領域が導入されました。X.509 v3 証明書には、次のユーザー情報が含まれます。

- 証明書所有者の識別名（DN）
- 証明書を発行した認証局の DN

---

**注意：**

DN の DC コンポーネントおよび EMAIL コンポーネントでは、印刷可能な（ASCII）文字のみを使用する必要があります。

この制限は、マルチバイト・キャラクタ・セットを使用するロケールでも、識別名の DC コンポーネントおよび EMAIL コンポーネントには ASCII 文字を使用する必要があるという意味です。

---

- 証明書所有者の公開鍵
- 証明書発行者のデジタル署名
- 証明書の有効期間
- 証明書のシリアル番号

図 1-1 に、これらのすべての要素を含む、新規発行の証明書を示します。

**図 1-1 Oracle Application Server Certificate Authority が発行する証明書**


Oracle Application Server Certificate Authority は、X.509 証明書を発行および処理します。また、複数の証明書タイプをサポートするので、X.509 CRL（証明書失効リスト）の発行および処理も行います。

## PKI 資格証明のコンテナ

コンテナは、メッセージの署名や検証などの PKI 操作に使用する様々な関連資格証明の保持に使用します。このようなコンテナのデータ構造に、ユーザーの秘密鍵、証明書およびユーザーが信頼するルート証明書のリストが安全に格納されます。SSL 接続でのピアの識別情報または受信された署名の検証には、信頼できる証明書を使用します。Netscape や Internet Explorer などのブラウザでは、証明書のコンテナを「証明書データベース」や「証明書キャッシュ」と呼ぶことがあります。Oracle Identity Management インフラストラクチャでは、このコンテナを「Oracle Wallet」と呼びます。

## 登録局（RA）

RA はオプションのシステムであり、エンド・エンティティ識別情報の検証や認証など、一部の管理機能を CA から委任されます。これは、CA とユーザー間のインタフェースとして動作します。RA は、新しい証明書の発行、期限切れの証明書の更新および証明書の失効の要求を受信します。RA は、要求を行ったユーザーが指定した識別情報を評価して、そのユーザーが、本人であるかどうかを検証します。既存の証明書の場合、RA は、要求を行ったユーザーと指定した識別情報および公開鍵との関連を検証し、承認された要求を CA に送信します。

**注意：** OracleAS では、RA の機能は、Oracle Application Server Certificate Authority 自体が実行します。

# PKI の利点

PKI には次の利点があります。

- セキュアで信頼できるユーザー認証

  信頼できる認証は 2 つの要素に依存しています。1 つ目の要素は、公開鍵 / 秘密鍵のペアの秘密鍵部分を所有していることの証明です。これは、公開鍵を使用する自動処理によって検証されます。2 つ目の要素は、認証局による、公開鍵が特定の識別情報に属することの検証です。PKI ベースのデジタル証明書によって、鍵のペアに基づいた識別情報の接続が検証されます。

- データ整合性

  確立された公開鍵 / 秘密鍵のペアの秘密鍵を使用してデジタル・トランザクションに署名すると、送信中にデータを変更することが難しくなります。ユーザー X によるこのデジタル署名は、ユーザー X の秘密鍵で暗号化された元のメッセージのコード化されたダイジェストです。受信者は、ユーザー X の対応する公開鍵を使用して、メッセージが変更されていないか、およびそのメッセージが実際に、X によって送信されたものかを検証できます。メッセージまたはダイジェストが変更されていると、公開鍵を使用した検証に失敗し、メッセージまたはダイジェストが信頼できないものであることが受信者にわかります。

- 否認防止

  デジタル署名により、メッセージ発信者は、メッセージを作成したことを否認することも難しくなります。

- 送信または格納された情報への不正アクセスの防止

  公開鍵から秘密鍵を導出するために必要な時間と手間を考慮すると、鍵のペアの所有者以外のユーザーがメッセージを復号化することはまずできません。

# OracleAS PKI の概要

この項では、OracleAS PKI の概要について説明します。説明する内容は次のとおりです。


- 以前のコストおよび問題
- OracleAS PKI の利点
- OracleAS PKI のコンポーネント


## 以前のコストおよび問題

OracleAS PKI を導入するまでは、認証に使用する証明書の取得に多くの手順と時間が必要でした。適切なフォームを取得して必要事項を正確に入力し、適切な登録局に配信する必要がありました。登録局によって識別情報が検証され、承認済フォームがユーザーに戻されると、ユーザーはそのフォームを認証局に配信する必要がありました。認証局はこの承認済フォームを処理し、実際の証明書を発行しました。この配信を行うために、承認済の要求の内容を別のフォームにカット・アンド・ペーストすることも、たびたび必要になりました。認証局がこの新しいフォームを受信した後、実際の証明書が届くまで数日または数週間かかることもありました。

## OracleAS PKI の利点

OracleAS PKI によって、多くのコストと時間を必要とした以前の手順のほとんどが不要になり効率化されます。OracleAS PKI は、認証機能、ユーザー・リポジトリおよびアプリケーションを緊密に統合します。また、第三者機関に証明書を要求し、手動でその証明書をアプリケーションおよび中央ディレクトリに送信するというユーザーの負担を軽減します。

OracleAS PKI の主要部分である Oracle Application Server Certificate Authority は、わかりやすいワンストップ・ソリューションで、使いやすい Web インタフェースが用意されており、登録局（RA）は CA に統合されています。ユーザーは、要求をオンラインで送信し、認証情報を提供して自動的に証明書を取得します。この証明書は、Oracle Internet Directory 内のユーザーのエントリに自動的にリンクされ、シングル・サインオンを有効にして、対応するディレクトリのエントリに対してチェックを行い、ユーザーを認証します。この Identity Management インフラストラクチャおよび Oracle Application Server Certificate Authority は、データベースや Oracle Collaboration Suite など、他の多くの Oracle コンポーネントで使用されます。

証明書は、ユーザーに対して発行された後、シングル・サインオン資格証明のかわりに使用できます。その結果、認証要件が PKI ほど厳しくないシングル・サインオン・アプリケーションだけでなく、PKI 用に構成されたすべてのシングル・サインオン・アプリケーションに即座にアクセスできます。前述のとおり、ユーザーの鍵のペアを使用してデジタル署名を有効にでき、整合性および否認防止が保証されます。

# OracleAS PKI のコンポーネント

OracleAS PKI は業界標準仕様に準拠し、次のコンポーネントを使用します。

- コンテナ、Oracle Wallet および Oracle Wallet Manager（OWM）
- Secure Sockets Layer（SSL）
- Oracle Internet Directory および Single Sign-On（SSO）
- Oracle Application Server Certificate Authority

## コンテナ、Oracle Wallet および Oracle Wallet Manager（OWM）

証明書の形式と内容および証明書のコンテナは、いくつかの国際規格で定義されています。X.509 v3 規格で、これらの証明書の仕様が定義されています（「デジタル証明書の内容および使用方法」の項を参照）。PKCS #12（個人情報交換構文）規格で、コンテナの仕様が定義されています。

既存の標準 PKI 資格証明を所有するユーザーは、その資格証明を PKCS #12 形式でエクスポートし、Netscape Communicator、Microsoft Internet Explorer などのブラウザまたは Oracle Wallet Manager にインポートできます。PKCS #12 規格によって、相互運用性が高まり、組織にかかる PKI 配置のコストが軽減されます。

> **関連項目：** 次の第 7 章の項
>
> 「ブラウザへの新規発行の証明書のインポート」
>
> 「ブラウザからの Wallet のエクスポート（バックアップ）」
>
> 「ファイル・システムからの証明書のインポート」

Oracle Wallet Manager を使用すると、このような証明書の取得、使用および格納を簡単に実行できます。Oracle Wallet Manager には、証明書およびそのコンテナを使用して行う通常の操作、または証明書およびそのコンテナに対して行う通常の操作を標準化したグラフィカル・ユーザー・インタフェース（GUI）が用意されています。この GUI を、OracleAS では Oracle Wallet と呼びます。

サーバー管理者は、OWM を使用して PKCS #10 証明書要求を作成できます。完了した要求が OWM で生成された後、管理者は、その要求をファイル・システムに保存したり、Oracle Application Server Certificate Authority のサーバー / 下位 CA 証明書フォームにコピー・アンド・ペーストして、OCA 証明書を要求できます。前述の「関連項目」の最後のリンクを参照してください。

これらの Wallet は PKCS #12 規格に準拠し、Oracle Application Server Certificate Authority で使用されるコンテナです。Netscape Communicator、Microsoft Internet Explorer などのサード・パーティ・アプリケーションと相互運用性があるため、オペレーティング・システム間に貴重な移植性が実現します。

### Secure Sockets Layer（SSL）

Secure Sockets Layer（SSL）は、インターネットの保護に最も広く使用されているプロトコルです。公開鍵を暗号化して認証、暗号化、データ整合性を有効にします。また SSL では、これらのツールを使用して、サーバーとクライアントの両方で使用する一意のワンタイム・セッション・パスワードを暗号化することにより、セッション鍵の管理をセキュアにできます。このパスワードは、セキュアに送受信された後、サーバーとクライアント間で行われる後続の通信すべての暗号化に使用されます。この暗号化によって、他のユーザーはそれらのメッセージを解読できなくなります。Oracle HTTP Server、Web Cache、Oracle Internet Directory、Oracle データベースなど、すべてのサーバー・コンポーネントで、通信の保護に SSL が使用されます。

### Oracle Internet Directory および Single Sign-On（SSO）

Oracle Internet Directory は、LDAP バージョン 3 のディレクトリです。LDAP は、Lightweight Directory Access Protocol の略称です。このディレクトリによって、Oracle Application Server Certificate Authority が発行した証明書の公開などの認証資格証明用に中央リポジトリが提供されるため、PKI ベースのシングル・サインオンが可能になります。Oracle Internet Directory によって、アクセスが属性レベルで制御されるため、特定のユーザーによる特定の属性の読取り、書込みまたは更新権限が制限されます。また、SSL を使用して、ディレクトリに対する問合せとレスポンスを保護および認証します。

### Oracle Application Server Certificate Authority

OracleAS 製品スイートに新しく追加された Oracle Application Server Certificate Authority を使用すると、証明書のライフ・サイクル全体を管理できます。このライフ・サイクルには、新しい証明書に対する要求の記録および処理、ユーザー資格証明の検証、これらの証明書の発行、更新または失効が含まれます。以前、これらの処理には、単調でエラーが発生しやすい、記録保持操作およびカット・アンド・ペースト操作が別々に必要でした。

Oracle Application Server Certificate Authority では、数回のクリックで証明書の生成、送信および格納を実行できます。これによって、資格証明の検証および認証が単純かつ高速になりました。

Oracle Application Server Certificate Authority は、Oracle Application Server のインフラストラクチャ・コンポーネントです（必須ではありません）。

# 2

# 識別情報管理および OracleAS Certificate Authority の機能

Oracle Application Server Certificate Authority（OCA）はセキュアなメカニズムで、クライアントおよびサーバーに対して X.509 v3 デジタル証明書の作成および署名を行います。OCA は、管理者が選択または作成したポリシーを適用します（第 5 章を参照）。また、その管理者によって、スケーラブルな Web ベースのインタフェースを使用して制御されます（第 4 章を参照）。OCA は、Web ベースのユーザー・インタフェース（第 7 章を参照）を含む、このような証明書をサポートおよび管理するための安全なインフラストラクチャを提供します。

この章では、Oracle Application Server Certificate Authority の機能および処理を有効にするアーキテクチャについて説明します。内容は次のとおりです。

- 識別情報管理の構成要素とアーキテクチャ
- Oracle Application Server Certificate Authority の主要機能
- 自動または従来型のプロビジョニング
- 階層的な認証局のサポート
- 配置およびインストール

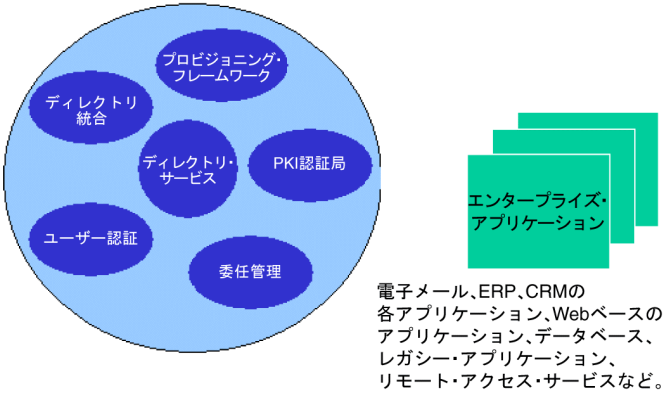
# 識別情報管理の構成要素とアーキテクチャ

完全な識別情報管理ソリューションには、次のコンポーネントが含まれます。

- ユーザー情報を格納および管理するための、スケーラブルで安全な、規格準拠のディレクトリ・サービス
- エンタープライズ・プロビジョニング・システム（HR アプリケーションなど）にリンク可能、またはスタンドアロンで動作可能なユーザー・プロビジョニング・フレームワーク
- 識別情報管理システムの管理者が、個々のアプリケーションの管理者または直接エンド・ユーザーにアクセス権を選択して委任できる委任管理モデルおよびアプリケーション
- 様々な要件に対応する適切なセキュリティ・モデルおよびユーザー・インタフェース・モデル
- 企業が、識別情報管理ディレクトリを、レガシー・ディレクトリまたはアプリケーション固有のディレクトリに接続できるディレクトリ統合プラットフォーム
- ユーザー認証用のランタイム・モデルおよびアプリケーション
- PKI 証明書を作成および管理するシステム

図 2-1 に、企業識別情報管理ソリューションのモデルを示します。

**図 2-1 企業識別管理ソリューションのモデル**



次の項で、Oracle Identity Management インフラストラクチャの詳細を説明します。

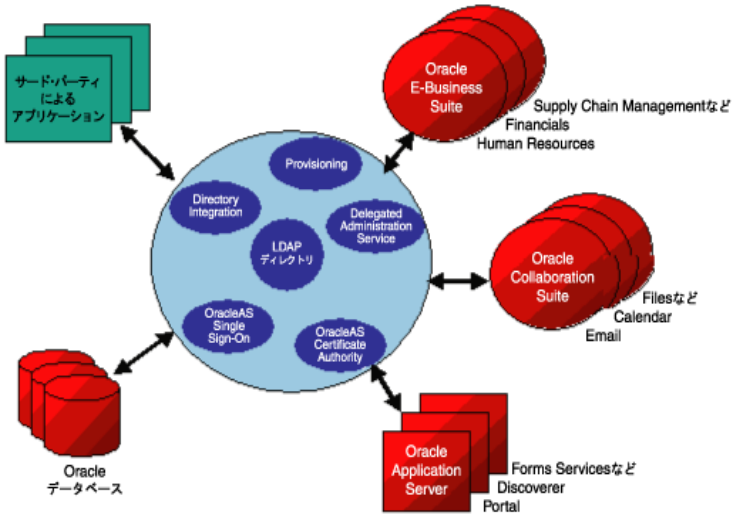

- Oracle Identity Management
- 企業での Oracle Identity Management の使用
- Oracle セキュリティ・アーキテクチャでの Oracle Identity Management の役割
- Oracle Identity Management での OracleAS Certificate Authority の役割
- SSO 統合を介した簡易プロビジョニング


## Oracle Identity Management

Oracle Identity Management は、企業全体でユーザーおよびアプリケーションを保護する場合に Oracle 製品が依存する統合インフラストラクチャです。Oracle Identity Management の代表的なリリース手段は Oracle Application Server ですが、これは、他の Oracle 製品でもインフラストラクチャの一部として付属しています。Oracle Identity Management インフラストラクチャには、次のコンポーネントが含まれます。

- Oracle Internet Directory。Oracle Database に実装されている、スケーラブルで堅牢な LDAP V3 準拠のディレクトリ・サービスです。
- Oracle Directory Integration and Provisioning。Oracle Internet Directory と他のディレクトリの同期を可能にし、Oracle コンポーネントおよびアプリケーションに対して自動プロビジョニング・サービスを提供します。また、標準インタフェースを使用して、サード・パーティのアプリケーションに対しても、自動プロビジョニングを提供します。
- Oracle Delegated Administration Services。ユーザーおよびアプリケーション管理者によるディレクトリ情報を、プロキシ・ベースで信頼できる形で管理できます。このコンポーネントは、ポータルや電子メールなどのアプリケーションで活用できます。
- OracleAS Single Sign-On。エンド・ユーザーはこれにより、Oracle およびサード・パーティの Web アプリケーションに対してシングル・サインオン・アクセスができます。
- Oracle Application Server Certificate Authority。X.509 v3 証明書を生成および公開して、PKI ベースの強固な認証方式をサポートします。

**図 2-2 企業統合型の識別情報管理**




## 企業での Oracle Identity Management の使用

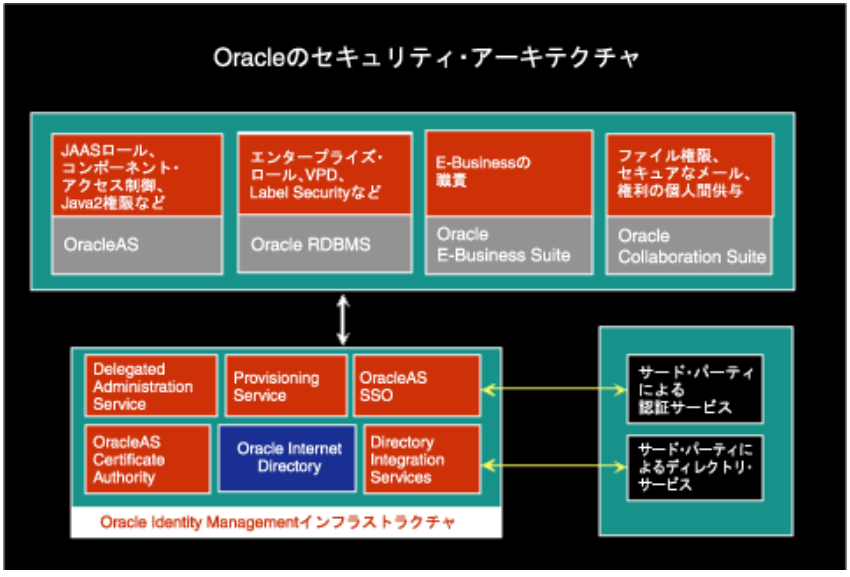
Oracle Identity Management は、Oracle 製品にエンタープライズ・インフラストラクチャを提供するように設計されていますが、カスタムおよびサード・パーティのエンタープライズ・アプリケーション、ハードウェアおよび企業のネットワーク・オペレーティング・システム用の、堅牢でスケーラブルな識別情報管理ソリューションとしても機能します。

また、オラクル社は、サード・パーティのアプリケーション・ベンダーと提携しているので、Oracle Identity Management をインストールした後、サード・パーティのアプリケーションですぐに使用できるようになっています。

## Oracle セキュリティ・アーキテクチャでの Oracle Identity Management の役割

各 Oracle テクノロジ・スタック（RDBMS、Application Server、E-business Suite および Collaboration Suite）では、それぞれの設計の核として適切なセキュリティ・モデルをサポートします。それでも、これらのすべてで、それぞれのセキュリティ・モデルおよび機能の実装に、Oracle Identity Management インフラストラクチャを使用しています。図 2-3 に、このアーキテクチャを示します。

**図 2-3 Oracle Identity Management のセキュリティ・モデル**




OracleAS は、Java Authentication and Authorization Service（JAAS）と呼ばれる J2EE 準拠のセキュリティ・サービスをサポートします。JAAS は、Oracle Internet Directory で定義したユーザーおよびロールを利用できるように構成できます。同様に、データベースのセキュリティ機能である、エンタープライズ・ユーザーおよび Oracle Label Security という手段によって、Oracle Internet Directory で定義したユーザーおよびロールを活用できます。つまり、これら両方のプラットフォームがあることで、それぞれ固有のセキュリティ機能を使用して開発されたアプリケーションで、基盤となる Identity Management インフラストラクチャを透過的に活用できるようになります。

Oracle Collaboration Suite および Oracle E-Business Suite は、RDBMS および OracleAS プラットフォーム上に階層化されたアプリケーション・スタックです。前述のとおり、この階層化によって、Oracle Identity Management インフラストラクチャとの間接統合があるレベルで行われます。また、これらの製品には、Oracle Identity Management に依存した独立機能もあります。たとえば、Oracle E-mail や Oracle Voicemail & Fax などの Collaboration Suite コンポーネントは、Oracle Internet Directory を使用して、製品固有のユーザー設定項目、ユーザーの連絡先およびアドレス帳を管理する必要があります。これらのコンポーネントは、電子メールを保護するために、Oracle Application Server Certificate Authority に依存します。

これらの Oracle テクノロジ製品では、Provisioning Integration Service を活用して、ユーザー・アカウントとユーザー権限のプロビジョニングおよびプロビジョニング解除を、自動的に行います。Delegated Administration Services は、ユーザーの設定項目および連絡先のセルフ・サービス管理に広く使用されています。また、これらの製品のセキュリティ管理インタフェースは、サービス・ユニットと呼ばれるユーザー管理およびグループ管理の基本単位を活用します。

## Oracle Identity Management での OracleAS Certificate Authority の役割

Oracle Application Server Certificate Authority は、Oracle Internet Directory および Single Sign-On を介して、Oracle Identity Management インフラストラクチャを使用します。Oracle Internet Directory によって、証明書を発行時に公開し、その情報をすべての接続データベースに伝播できます。Single Sign-On は、アプリケーションなどの Oracle コンポーネント（Oracle Collaboration Suite のエンタープライズ・ユーザーおよび電子メール保護機能など）が依存する標準インタフェースを提供します。Oracle Application Server Certificate Authority が発行した証明書は、単純かつ高速で、整合性のある識別情報管理に求められるセキュアな認証をサポートします。

## SSO 統合を介した簡易プロビジョニング

OracleAS Single Sign-On（SSO）Server に対して認証を行うアプリケーション・ユーザーは、透過的に証明書を取得でき、技術教育や PKI の理解は必要ありません。その後は、アプリケーションで、新しく発行された証明書を使用して SSO によるこのアプリケーション・ユーザーの認証が透過的に行われ、セキュリティが強化されます。発行された PKI 証明書は、Oracle Internet Directory（OID）に自動的に公開されます。この強力な機能を提供することにより、Oracle Database、Oracle Internet Directory および OracleAS Single Sign-On Server のセキュリティ、高可用性およびスケーラビリティが高まります。

Oracle Application Server Certificate Authority（OCA）管理者は、オプションで OCA を構成して、SSO を介して URL をブロードキャストできます。これを実行すると、SSO を介して認証を行うユーザーは、OCA の簡単なグラフィカル・インタフェースを使用して証明書を要求できます。この証明書があれば、その後の SSO 認証がさらに簡単になります。これは、SSO で、OID を使用してユーザーのブラウザで自動的に提供された証明書を検証できるためです。OCA は、失効した証明書および期限切れの証明書を定期的に OID から自動的に削除するため、SSO は、OID の情報を信頼して使用できます。

# Oracle Application Server Certificate Authority の主要機能

Oracle Application Server Certificate Authority の主要機能は、スケーラブルな Web ブラウザ・インタフェースを介して使用できます。これらの機能は、業界標準の証明書の管理、LDAP ディレクトリとの統合およびポリシーの適用をサポートします。説明する内容は次のとおりです。

- オープン規格に対するサポート
- 柔軟なポリシー
- 管理者およびエンド・ユーザーにとっての使いやすさ
- OCA 画面での National Language Support（NLS）
- スケーラビリティ、パフォーマンスおよび高可用性

## オープン規格に対するサポート

Oracle Application Server Certificate Authority は、オープン規格をサポートするので、異機種間コンピューティング環境での通信ができます。Oracle Application Server Certificate Authority は、次の規格をサポートします。

- X.509 v3 証明書および証明書失効リスト（CRL）
- IETF PKIX 規格
- 最長 4096 ビットの署名鍵（RSA）
- スマートカード
- Microsoft Internet Explorer および Netscape Communicator を使用した証明書要求
- 様々な PKCS 規格（5、7、8、10、12 など）
- 証明書要求の複数の登録プロトコル（証明書要求の Signed Public Key and Challenge（SPKAC）や Public Key Cryptography Standard（PKCS）#10 など）

## 柔軟なポリシー

ポリシーは、ルールと制限のセットで、ユーザーが使用を認められるアクション、アクセスまたは認証を制限します。Oracle Application Server Certificate Authority には、構成可能なポリシー・ルールのセットが用意されており、このセットを使用して、ユーザー（またはユーザーのグループ）が取得できる証明書のプロパティを制限できます。サイトでは、これらのルールをカスタマイズして、特定の PKI 要件を満たすように、Oracle Application Server Certificate Authority を構成できます。デフォルトのポリシー・ルールがいくつか用意されていますが、独自のポリシー・ルールも適用できます。

## 管理者およびエンド・ユーザーにとっての使いやすさ

Oracle Application Server Certificate Authority の Web ベースの管理インタフェースには、「認証管理」および「構成管理」という 2 つの主要タブがあります。これらを使用する場合、管理者は、最初のエントリ時にフォームに必要事項を入力し、その後証明書をインポートして登録する必要があります。

「認証管理」タブを使用すると、管理者は、証明書要求の承認または拒否、および証明書失効リスト（CRL）の生成または更新ができます。また、様々な理由（セキュリティが損なわれた場合など）によって、発行された証明書を失効させることもできます（OCA を停止および起動する場合、管理者は、コマンドライン・ツール `ocactl` を使用する必要があります。このツールには管理者のパスワードが必要です）。

Oracle Application Server Certificate Authority の Web ベースのエンド・ユーザー・インタフェースにも、「ユーザー証明書」および「サーバー / 下位 CA 証明書」という 2 つのタブがあります。「ユーザー証明書」タブをクリックすると、ユーザーは、OracleAS Single Sign-On 名およびパスワードを使用してユーザー自身を認証できます。SSO 認証を選択し「送信」をクリックすると、SSO ウィンドウが表示され、SSO 用のユーザー名およびパスワードを入力できます。

「ユーザー証明書」ページには、すべての証明書要求やそのステータス（保留、認可済、拒否済）などが表示されます。新しい証明書の要求、証明書失効リスト（CRL）のダウンロードまたは認証方式の変更を実行できます。

「サーバー / 下位 CA 証明書」タブをクリックすると、新しいサーバー / 下位 CA 証明書の要求、CRL のダウンロードまたは CA 証明書のダウンロードを実行できます。また、ID/ シリアル番号または一般名で、特定の証明書または証明書要求を検索できます。

## OCA 画面での National Language Support（NLS）

特定の前提を満たす場合は、OracleAS Certificate Authority の管理者用画面およびユーザー用画面を、クライアントまたはサーバーの言語で表示できます。それには、データベースのキャラクタ・セットが UTF8 であると同時に、必要な言語が OCA によりサポートされている必要があります。この前提が満たされない場合は、英語が使用されます。OCA の管理コマンドライン・ツールの `ocactl` で使用できるのは英語によるコマンドだけですが、メッセージ（情報メッセージやエラー・メッセージなど）は、サポートされている場合にかぎり、サーバー・ロケールの言語で表示されます。サポートされていない場合は、英語が使用されます。

**関連項目：** 第 6 章「OracleAS Certificate Authority の管理 : 高度なトピック」の「OCA 画面での National Language Support（NLS）の構成」の項

## スケーラビリティ、パフォーマンスおよび高可用性

Oracle Application Server Certificate Authority は、OracleAS をアプリケーション・サーバーとして統合し、Oracle Database を次の情報のリポジトリとして統合することによって、自動的にこれらの機能を利用できます。

- ユーザー、ロールおよび権限
- 保留中および承認済の証明書要求
- 発行された証明書
- ユーザー・アクティビティのロギング情報および JAZN 認証情報

# 自動または従来型のプロビジョニング

従来型プロビジョニングでは、管理者がユーザーに証明書を発行します。Oracle Application Server Certificate Authority が SSO および SSL を使用して提供する自動プロビジョニングでは、PKI のサポートに使用してきた従来の方法でのコストおよび遅延を削減できます。

SSO 認証の場合、Oracle Application Server Certificate Authority は、mod_osso および OracleAS Single Sign-On Server を使用します。SSO で自動的に認証されているユーザーに Oracle Application Server Certificate Authority から証明書を発行する場合にこれらの方式を使用すると、証明書の管理が簡単になります。

以前に X.509 v3 証明書を発行されているユーザーは、Oracle 認証局に対する認証手段として、その証明書を HTTPS 経由で送信できます。証明書が同じ Oracle 認証局で発行され、まだ失効になっていない場合、証明書要求は自動的に承認されます。承認が迅速に行われるため、ユーザーは、管理者またはセキュリティ担当者が要求を承認するまで待たずに、暗号化または署名の追加証明書を取得できます。

また、OCA は、Netscape と Internet Explorer を統合することによってスマートカードをサポートし、ブラウザのローカル設定で指定されている言語でそのフォームを表示できます。

Oracle Application Server Certificate Authority は、次の認証方式をサポートします。次の項で、その内容を説明します。

- OracleAS Single Sign-On 認証
- Secure Sockets Layer（SSL）を使用した証明書ベースの認証
- 手動による承認

## OracleAS Single Sign-On 認証

OracleAS デフォルトのユーザー管理および認証プラットフォームは、Single Sign-On Server および Oracle Internet Directory で構成されます。Oracle 認証局は、Oracle Internet Directory を証明書の格納リポジトリとして使用します。このアーキテクチャでは、証明書を集中管理できるため、証明書のプロビジョニングおよび失効が簡素化されます。

Oracle Application Server Certificate Authority を OracleAS Single Sign-On Server および Oracle Internet Directory と統合することによって、これらに依存するアプリケーション用に、透過的な証明書プロビジョニング・メカニズムが提供されます。Oracle Internet Directory でプロビジョニングされ、OracleAS Single Sign-On Server に対して認証されたユーザーは、Oracle 認証局にデジタル証明書を要求できます。OracleAS Single Sign-On Server では、「SSO 統合を介した簡易プロビジョニング」の項で説明しているとおりに OCA が構成されている場合、「証明書の取得」ポップアップ・ページが表示され、この操作が容易になります。このユーザーは、ユーザー名 / パスワードまたは既存の SSL 証明書（あるいはその両方）を使用して認証できます。「証明書の要求」ボタンをクリックするだけで、証明書はすぐに Oracle Internet Directory で自動的にプロビジョニングされます。

この方式では、OracleAS Single Sign-On Server の機能を活用してユーザーが識別され、Oracle Internet Directory のデータを使用して、証明書要求の必要なフィールドに必要事項が入力されます。同様に、Oracle 認証局の管理者または証明書の所有者は、リアルタイムで証明書を失効させることができます。その場合、証明書は Oracle Internet Directory から自動的に削除されます。その後は、失効にした証明書を SSO 認証に使用しようとしても失敗します。

## Secure Sockets Layer（SSL）を使用した証明書ベースの認証

Oracle Application Server Certificate Authority は、証明書ベースの認証をサポートするため、ユーザーは、以前から持っていて失効になっていない X.509 v3 証明書によって、HTTPS 経由で Oracle Application Server Certificate Authority により認証されます。この方式でユーザーを認証した場合、Oracle Application Server Certificate Authority は、SSL や署名などの目的で、新しい証明書を遅延なく自動的に発行できます。

## 手動による承認

組織のセキュリティ・ポリシーでは、自動処理で証明書を発行するかわりに、手動で証明書要求を承認するように指示できます。この操作を選択すると、承認および認証に従来型の手動モードが使用され、Single Sign-On モードと SSL モードはオフになります。Oracle 認証局は、このような承認処理を適用して、要求を行ったユーザーの識別情報を手動で検証するように、管理者またはセキュリティ担当者に要求できます。

手動で認証を承認する場合、Oracle Application Server Certificate Authority が受け入れる証明書要求には、すべての CA に必要な基本入力フィールドを使用します。この手動処理では、ユーザーが、名前、電子メール・アドレス、場所などの個人情報の入力が求められます(ユーザーは、オプションとして、ドメイン・コンポーネントなどの詳細な DN 属性を指定することで、証明書要求をカスタマイズできます)。手動方式は、OracleAS Single Sign-On 認証や Secure Socket Layer 認証より複雑です。ただし、この方式では、既存の証明書を表示およびダウンロードする追加オプションを使用できます。サーバーおよび下位 CA でも、この手動処理を実行して証明書を要求できます。

# 階層的な認証局のサポート

Oracle Application Server Certificate Authority は、認証局の階層をサポートします。階層的な PKI では、セキュリティ・ドメインのルート CA は、最終的にすべてのユーザーによって信頼される、唯一の基点となる CA です。その識別情報は、信頼できるパスの先頭に使用されます。

Oracle Application Server Certificate Authority は、ルート CA として動作できます。また、別の CA の証明書を検証して下位 CA を作成することもできます。そのかわりに、下位のインストール環境の署名および SSL 証明書は、別にインストールした Oracle 認証局など、規格準拠の認証局からも取得できます。この下位 CA は、さらに下位レベルの CA に対しても証明書を発行できます。各認可レベルの証明書は上位レベルの CA によって署名されているため、ユーザーは、認証局のパスを信頼できる認可レベルまたはルート CA までトレースすることによって、証明連鎖を検証できます。

別々の認証局からの下位 CA 証明書の取得は、PKI インフラストラクチャがすでに整備されている場合に有効です。階層的な CA のサポートは、地理的に分散している組織で有効です。

**関連項目：** 付録 B「CA の階層の設定」

階層的な CA の使用には、コストおよび安全性の面でも重要な利点があります。この場合、通常の操作は下位 CA に担当させ、ルート CA は特別に保護できます。こうした保護には、高度にセキュアな場所でのオフライン化も含まれます。この方法であれば、オンラインの下位 CA が危殆化した場合でも、それを失効させ、新しい下位 CA を作成して置換することができます。それ以前のすべての操作では、発行済の証明書を引き続き使用できます。ただし、ルート CA が危殆化した場合は、まったく新しいインフラストラクチャを構築して、元のルート CA に依存するアプリケーションをすべて更新する必要があります。

# 配置およびインストール

Oracle Application Server Certificate Authority（OCA）では、OCA に必要な次のコンポーネントについて様々な配置方法をとることができます。

- Oracle HTTP Server（OHS）（OCA と同じマシン上に配置する必要があります）
- OCA 用の OC4J（OCA と同じマシン上に配置する必要があります）
- Infrastructure Metadata Repository
- Oracle Internet Directory（OID）
- Oracle Single Sign-On Server（SSO）（オプション）

---

**注意：** OCA は、デフォルトの選択として自動的にインストールされるわけではありません。OCA をインストールするには、インストールの対象として選択する必要があります。

---

デフォルトの配置では、これらすべてのコンポーネントは同じマシン上および同じ Oracle ホームに配置されます（図 2-4 を参照）。この構成は、開発および非本番環境に適しており、デフォルトのインストール構成です。OCA のこのデフォルトの配置構成のインストール手順は、Oracle Application Server 10*g* のインストレーション・ガイドの第 6.14 項を参照してください。

---

**注意：** 1 つのリポジトリでの OracleAS Certificate Authority のスキーマは、1 つの OCA とのみ併用できます。

別の OracleAS Certificate Authority をインストールする場合、先行する OCA のインストールに使用したリポジトリは選択できません。同じリポジトリを選択すると、OCA 構成ツールが正常に実行されません。

それによって、インストール処理を途中で終了し、インストール全体をやり直すことが必要になります。

---

---

**注意：** OracleAS Certificate Authority をインストールする際、zh ロケールと zh_TW ロケールでは、OCA のインストールおよび起動を実行できません。これらのロケールのかわりに、次のロケールのいずれかを使用します。

中国語（簡体字）用には zh_CN.GBK

中国語（繁体字）用には zh_TW.BIG5

---

推奨される本番配置では、OHS、OC4J、OCA および Infrastructure Metadata Repository は、1 つのマシン上、そして 1 つの Oracle ホーム内に配置されます。SSO、OID などの残りのコンポーネントは、別のマシン上の別の Oracle ホーム内に配置されます。このように物理的に分離することによって、個別の位置のセキュリティを強化し、OCA を安全な位置で保護できます。OCA は信頼できる証明連鎖の最上位にあるため、本番環境ではこれらの追加の保護が必要になります（図 2-5 を参照）。同様に、Oracle Application Server Certificate Authority のセキュリティ上の理由から、これらのコンポーネントの起動または停止に Enterprise Manager は使用しないことをお薦めします。

この推奨される配置構成のインストール手順は、Oracle Application Server 10*g* のインストレーション・ガイドの第 6.20 項を参照してください。

**図 2-4 Oracle Application Server Certificate Authority のデフォルトのインストール**

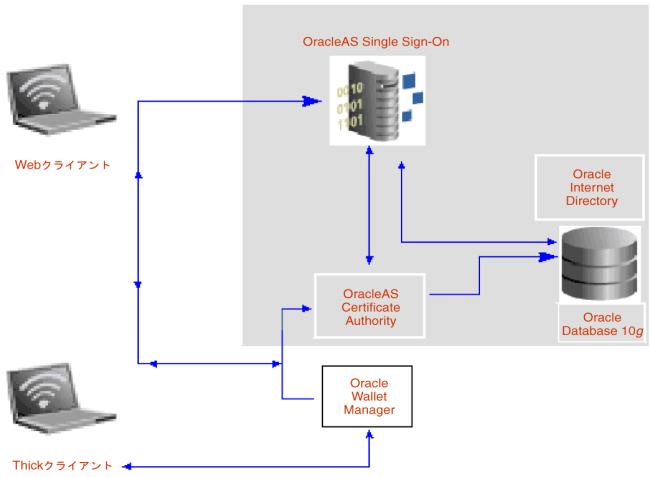
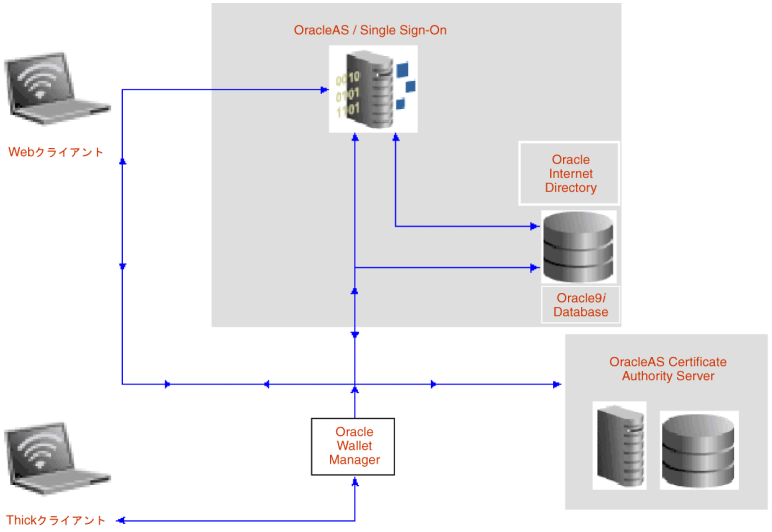


**図 2-5 OracleAS Certificate Authority の推奨される本番インストール**



# 3

# OCA および証明書の管理の概要

Oracle Application Server Certificate Authority の Web ベースの管理インタフェースは、次に示す 3 つの大きな事項を対象としています。各事項は、ホームページのタブからアクセスできます。

- 証明書管理に関する事項 : 証明書の発行、失効または更新の要求、すでに発行されている証明書、および証明書失効リスト（CRL）
- 構成管理に関する事項 : OracleAS Certificate Authority のアクション用パラメータおよび証明書のセキュリティ・ポリシーを実装するためのパラメータ
- Oracle Application Server Certificate Authority アクティビティのログの表示

この章では、前述の 1 つ目の事項（証明書の管理）について説明します。他の 2 つの事項は、第 4 章「Oracle Application Server Certificate Authority の構成」を参照してください。

管理操作には、付録 A「コマンドライン管理」で説明するコマンドライン・インタフェースが必要な場合があります。これらの操作のうち 2 つは、Oracle Application Server Certificate Authority の起動および停止です。詳細は、後の項を参照してください。管理者の証明書の要求または置換とともに説明します。

エンド・ユーザーが Oracle Application Server Certificate Authority と対話する際には、Web ベースの独立したインタフェースを使用できます。このインタフェースにより、個人的な証明書関連の操作を実行可能なフォームが提供されます。詳細は、第 7 章「Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース」を参照してください。

この章の内容は、次のとおりです。


- Oracle Application Server Certificate Authority の起動および停止
- 管理者の証明書の要求
- 管理者の証明書の置換
- OracleAS Certificate Authority 管理インタフェースの概要
- 証明書の管理

- 証明書失効リスト（CRL）の更新
- Single Sign-On（SSO）および OracleAS Certificate Authority（OCA）
- OracleAS Certificate Authority のインストールのデフォルト値


# Oracle Application Server Certificate Authority の起動および停止

セキュリティ上の理由から、OCA の起動および停止の操作は、コマンドライン・ツール `ocactl` を使用しないと実行できません。このツールには管理者のパスワードが必要です。これらの操作の使用例は、3-7 ページの「管理者の証明書の置換」を参照してください。このツールの詳細は、付録 A「コマンドライン管理」を参照してください。

OracleAS Certificate Authority を起動するには、次の 5 つのコンポーネントが動作中または使用可能である必要があります。

- Infrastructure Metadata Repository
- Oracle Internet Directory
- Oracle Single Sign-On Server（SSO）（オプション）
- Oracle HTTP Server（OHS）
- OCA 用の OC4J

OCA が、他のインフラストラクチャ・コンポーネントとは異なる `$ORACLE_HOME` にインストールされている場合は、リポジトリの後に、OHS および OCA 用の OC4J を別々に起動する必要があります。このコマンドは次のように、OCA の `$ORACLE_HOME` で使用します。

```
$ORACLE_HOME/opmn/bin/opmnctl startall
```

OCA を含むすべてのインフラストラクチャ・コンポーネントが 1 つの `$ORACLE_HOME` にインストールされている場合は、OHS および OC4J はすでに起動されています。

Oracle Application Server Certificate Authority を起動、停止または再起動するには、コマンドラインで、次に示すコマンドを入力します。

1. Oracle Application Server Certificate Authority を停止するには、次のコマンドを使用します。

   ```
   $ORACLE_HOME/oca/bin/ocactl stop
   ```

2. Oracle Application Server Certificate Authority を起動（または再起動）するには、次のコマンドを使用します。

   ```
   $ORACLE_HOME/oca/bin/ocactl start
   ```

3. Oracle Application Server Certificate Authority のステータスを取得するには、次のコマンドを使用します。

```
$ORACLE_HOME/oca/bin/ocactl status
```

# 管理者の証明書の要求

Web ベースのインタフェースで Oracle Application Server Certificate Authority の管理オプションおよび制御を使用するには、管理者の証明書が必要です。インストール中に管理者のパスワードを作成しておくと、この証明書は簡単に取得できます。他のタスクを実行する前に、最初にこの証明書を取得する必要があります。

他のシステムでは、管理者の PKI 証明書の要求、取得およびインストールに、コマンドライン、フロッピィ・ディスクおよびカット・アンド・ペースト操作が必要です。

ただし、Oracle Application Server Certificate Authority を使用すると、この処理は単純で簡単になります。

ユーザー認証用に管理者の証明書を要求するには、Oracle Application Server Certificate Authority を初めて起動した後に表示されるフォームに、必要事項を入力して送信するだけです。管理者として使用するコンピュータから Oracle Application Server Certificate Authority にアクセスしている必要があります。「認証管理」タブをクリックすると、「ようこそ」ページが表示された後、識別情報データの入力を求めるフォームが表示されます。

このフォームには、一般名、組織およびインストール中に作成した Certificate Authority 管理者のパスワードを入力する必要があります。電子メール・アドレス、組織単位、地域、州および国など、他の DN 情報も指定できます。

証明書の鍵のサイズ（デフォルトは 1024）および有効期間（デフォルトは 1 年）を選択できます。

管理者の証明書が発行されたら、それをブラウザにインポートします。ブラウザでこの証明書を使用すると、管理および構成インタフェースで Certificate Authority の機能にアクセスして、証明書の要求、証明書の失効または更新、およびポリシーの管理ができます。

この簡単な処理（簡単な要求フォームに必要事項を記入して実行する簡単なインポート）は、PKI 証明書の取得や使用のために（Oracle Application Server Certificate Authority より前で）実行する必要のあったすべての操作のかわりに実行できます。

証明書を要求するには、次の手順を実行します。

1. Oracle Application Server Certificate Authority の管理インタフェースにアクセスします。

   Web ブラウザを起動して、インストールの最後に表示されたとおりに URL および管理サーバーのポート番号を入力します。たとえば、次のように入力します。

```
https://Oracle_HTTP_host:ssl_port/oca/admin
```

oracle_HTTP_HOST は、OCA のインストール先ホストです。

ssl_port は、$ORACLE_HOME/install/portlist.ini の Oracle Certificate Authority SSL Server Authentication port に記載されています。Windows の場合、このパスは $ORACLE_HOME¥install¥portlist.ini です。

> **注意：** インストール後にポートが変更されていると、portlist.ini に最新情報が保管されていません。この場合は、Oracle Enterprise Manager Control にサインオンし、OCA がインストールされているインスタンスをクリックします。次に、「ポート」リンクをクリックし、「タイプ」列で「OCA Server Authentication (SSL)」というエントリを探して、その横の「使用中のポート」という見出しの列に表示されている数値を使用します。

画面に、「ようこそ」ページが表示されます。このページに表示されたリンクをクリックすると、管理者の証明書を要求するためのフォームが表示されます。

1. このフォームに DN、パスワードおよび証明書情報を入力して、証明書を要求します。
   - **DN 情報 :** 管理者を証明書の認証済所有者として識別する識別名（DN）用のデータを入力します。

**表 3-1 管理者の証明書の DN 情報**

| フィールド名 | 入力情報 |
|---|---|
| 一般名 | 証明書に記載する名前 |
| 電子メール・アドレス | 管理者の電子メール・アドレス |
| 組織単位 | 管理者が属する組織単位または部門の名前 |
| 組織 | 管理者が属する企業または組織の名前 |
| 市 / 地域 | 管理者の所在地 |
| 都道府県 | 管理者が所在する都道府県 |
| 国 | 管理者の国を表す 2 文字のコード |

**注意 :**

DN の DC コンポーネントおよび EMAIL コンポーネントでは、印刷可能な（ASCII）文字のみを使用する必要があります。

この制限は、マルチバイト・キャラクタ・セットを使用するロケールでも、識別名の DC コンポーネントおよび EMAIL コンポーネントには ASCII 文字を使用する必要があるという意味です。

- **Certificate Authority 管理者パスワード :** 証明書および構成の管理を実行できるのは、Oracle Application Server Certificate Authority の管理者だけです。この管理者は、OCA のインストール時に「OCA 管理者パスワードの指定」画面で入力したパスワードを、このセクションに入力すると初期認証されます。

  パスワードには、次の制限が適用されます。

  * 先頭文字には、データベースのキャラクタ・セットに含まれるアルファベットを使用します。
  * 8 文字以上の長さにする必要があります。
  * アルファベットとアルファベット以外の文字（数値または特殊文字）を
  * それぞれ 1 文字以上使用します。
  * ASCII キャラクタ・セットに含まれる文字のみを使用します。
  * Oracle の予約語は使用できません。
  * データベースのキャラクタ・セットに含まれる英数字のみを使用します。必要に応じて、アンダースコア（_）、ドル記号（$）または番号記号（#）を使用できますが、オラクル社では、$ と # の使用は避けるよう強くお薦めします。

  したがって、インストール時に選択する OCA 管理者のパスワードは、これらの制限に従う必要があります。

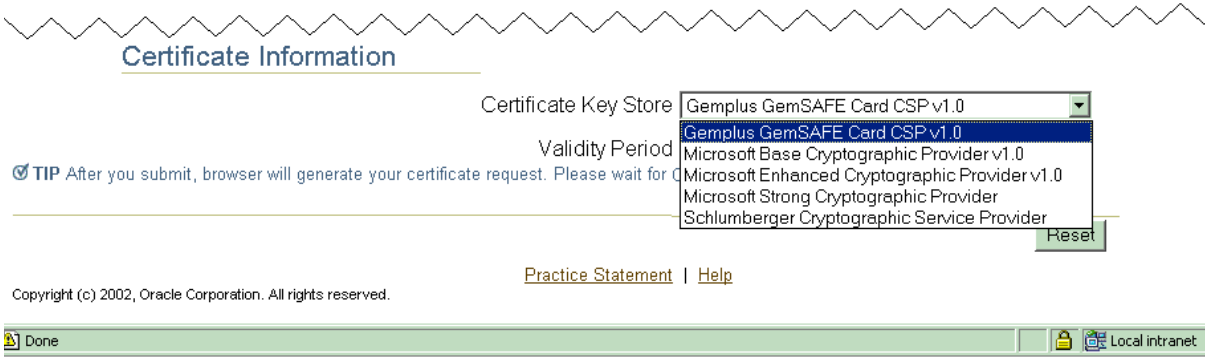
  パスワードの複雑さを検証する Oracle のルーチン（PL/SQL スクリプト UTLPWDMG.SQL によって指定）をデータベースで使用する場合は、パスワードは次の要件（またはそのスクリプトに追加する要件）も満たす必要があります。

  * 4 文字以上の長さにする必要があります。
  * ユーザー名と同じものは使用できません。
  * アルファベット、数字および句読記号をそれぞれ 1 文字以上使用します。
  * welcome、account、database、user など、単純明快な語は使用できません。
  * 一度設定したパスワードを後から変更する場合は、元のパスワードを 3 文字以上変更する必要があります。

- **証明書情報 :** 新しい証明書の作成に必須の 2 つの要素は、証明書の鍵のサイズおよび有効期間（または満了日）です。フォームのこのセクションで、これらのパラメータを選択します。

  * Netscape の場合は、512、1024 など、生成される鍵のペアのサイズ（ビット単位）を示す鍵サイズが表示されます。サイトに適したサイズを選択します。1024 は標準のデフォルトで、高いセキュリティが実現します。高い数値を選択すると、パフォーマンスは低下しますが、セキュリティは向上します。

  * Internet Explorer の場合は、暗号化サービス用に選択可能なプロバイダを示すキーストアが表示されます。標準の選択肢には、「Microsoft Base Cryptographic Provider」、「Microsoft Enhanced Cryptographic Provider」および「Microsoft Strong Cryptographic Provider」があり、鍵のサイズは、それぞれ 512 ビット、1024 ビット、2048 ビットに固定されています。また、スマートカードを使用する場合の Gemplus など、その他の選択肢が表示される場合もあります。フォームのこのセクションは、次のように表示されます。




Oracle Application Server Certificate Authority では、管理者の証明書に対して Microsoft Enhanced Cryptographic Provider を使用することをお薦めします。ただし、Gemplus などのスマートカード・リーダーが使用可能な場合は、それらを使用してください。リーダーがインストールされていない場合に、Gemplus や Schlumberger などのスマートカード・サプライヤを選択するとエラーになります。

- **有効期間 :** 証明書の有効期間。標準的なデフォルト値である 1 年が表示されますが、目的に合った期間も選択できます。

1. 最初からやりなおす必要がある場合は、「回復」ボタンをクリックします。
2. 管理者の証明書に対する要求を送信するには、「送信」ボタンをクリックします（ブラウザのセキュリティ・パスワードの指定が必要になる場合もあります）。
3. 鍵のペアの生成時には、ブラウザに表示される手順に従います。この処理は、選択した鍵のサイズおよびプロセッサ / メモリーの制限によって、数分かかる場合があります。

4. 「証明書のインポート」をクリックします（ブラウザのセキュリティ・パスワードの指定が必要になる場合もあります）。

   指定した一般名にクライアント認証の証明書が格納されます。

   これで、Oracle Application Server Certificate Authority の Web ベースのインタフェースを介して実行可能なタスクを、すべて実行できるようになりました（第 4 章「Oracle Application Server Certificate Authority の構成」を参照）。

# 管理者の証明書の置換

管理者の証明書を置換する必要が発生する場合があります。その原因には、秘密鍵のパスワードの紛失、秘密鍵の危殆化または盗難、新しい人間への管理者ロールの付与などがあります。

管理者の証明書を置換するには、サーバーを停止し、現行の管理者証明書を失効させて、サーバーを再起動する必要があります。これらのタスクは、コマンドライン・ツール `ocactl` を使用して実行します。このツールには OCA 管理者のパスワードが必要です。セキュリティ上の理由から、これらのコマンドはコマンドラインでのみ使用可能です。グラフィカル・ユーザー・インタフェース（GUI）では使用できません。

次に、管理者は、Oracle Application Server Certificate Authority の Web ページにナビゲートし、「Web 管理者登録」に表示されるフォームに必要事項を入力します（「管理者の証明書の要求」の項のここまでの記述を参照）。

次に、3 つの関連コマンドライン・タスクを示します。

1. Oracle Application Server Certificate Authority サーバーを停止するには、コマンドラインで次のコマンドを入力します。

   ```
   $ORACLE_HOME/oca/bin/ocactl stop
   ```

2. 管理者の証明書を失効させるには、次のコマンドを入力します。

   ```
   $ORACLE_HOME/oca/bin/ocactl revokecert -type WEBADMIN -reason <REASON_CODE>
   ```

   注意 : 次の（| で区切られた）いずれかの理由コードを選択できます。

   ```
   {KEY_COMPROMISE | CA_COMPROMISE | AFFILIATION_CHANGE | SUPERSEDED | CESSATION_
   OF_OPERATION | CERTIFICATE_HOLD | REMOVE_FROM_CRL | UNSPECIFIED}
   ```

3. 管理パスワードの変更もできます。詳細は、付録 A「コマンドライン管理」の「権限付きパスワードの変更」の項を参照してください。

4. コマンドラインで、次のいずれかのコマンドを入力して、Oracle Application Server Certificate Authority のサービスを起動します。

   UNIX の場合 : `$ORACLE_HOME/oca/bin/ocactl start`

   Windows の場合 : `%ORACLE_HOME%¥oca¥bin¥ocactl start`

この時点で、3-3 ページの「管理者の証明書の要求」の項の手順に従って、管理者の証明書を取得し、すべての管理機能を使用可能にします。

# OracleAS Certificate Authority 管理インタフェースの概要

管理タスクを実行するには、有効な管理者の証明書を所有している必要があります。最初のサインインを、管理者としてではなく一般ユーザーとして行うと、付録 C「トラブルシューティングの既知のヒント」の第 1 項「基礎的な問題および警告」の項目 a「問題 : 証明書要求で鍵のペアが生成されない（Windows）。」で説明されているエラー・メッセージが表示される場合があります。

Oracle Application Server Certificate Authority 管理インタフェースにアクセスするには、Web ブラウザを起動します。インストールの最後に表示されたとおりに、URL および管理サーバーのポート番号を入力します。

```
https://Oracle_HTTP_host:ssl_port/oca/admin
```

`oracle_HTTP_HOST` は、OCA のインストール先ホストです。

`ssl_port` は、$ORACLE_HOME/install/portlist.ini の

Oracle Certificate Authority SSL Server Authentication port に記載されています。Windows の場合、このパスは $ORACLE_HOME¥install¥portlist.ini です。

---

**注意 :** インストール後にポートが変更されていると、portlist.ini に最新情報が保管されていません。この場合は、Oracle Enterprise Manager Control にサインオンし、OCA がインストールされているインスタンスをクリックします。次に、「ポート」リンクをクリックし、「タイプ」列で「OCA Server Authentication (SSL)」というエントリを探して、その横の「使用中のポート」という見出しの列に表示されている数値を使用します。

---

OCA の起動コマンドを発行すると、次の図に示すように、3 つのサブタブが追加された Oracle Application Server Certificate Authority のホームページが表示されます。


これらの 3 つのサブタブを使用して、証明書または認証局の構成を管理する特定のタスクを実行できます。

- 「認証管理」タブ（説明はこの章）
- 「構成管理」タブ（説明は第 4 章）
- 「ログの表示」タブ（説明は第 4 章「Oracle Application Server Certificate Authority の構成」）

## 「認証管理」タブ

「認証管理」タブには、証明書の保留要求がすべて表示されます。表示されるページは次のようになります。


管理者は、このページを使用して、この後の各項で説明するタスクを選択できます。

# 証明書の管理

Oracle Application Server Certificate Authority では、すべての証明書要求およびそれらの現行ステータス（保留、拒否済または認証済）のマスター・リストが保持されます。「認証管理」タブをクリックすると、アクションの必要な証明書要求（保留）がすべて表示されます。管理者には、このような要求の承認または拒否、必要に応じて証明書の失効または更新、および証明書失効リスト（CRL）生成の管理を実行する役割があります。

これらのタスクを管理者として実行する場合は、証明書または証明書要求のマスター・リストを名前または番号で検索した後、特定の証明書または所定のリクエストを検証できます。

その後、次のタスクを実行できます。

- 個々の証明書要求の承認または拒否
- 発行された特定の証明書の失効（退職したユーザーが所有しているなどの理由で、証明書が危殆化されたか、適切でなくなっている場合）または有効期限前後の短期間での既存の証明書の更新

  **関連項目：** この更新期間の時間枠を指定できます。詳細は第 5 章「Oracle Application Server Certificate Authority でのポリシー管理」の、次の項を参照してください。

  - 「Oracle Application Server Certificate Authority の「ポリシー」サブタブ」の項の「製品に付属の証明書更新ポリシー」
  - 「ポリシー操作」の項の「編集」

次の項で、これらのすべての証明書管理タスクについて説明します。

- 証明書要求の承認または拒否
- 証明書の詳細の表示
- 証明書の失効
- 証明書の更新
- 単一の証明書要求または発行済証明書の表示
- 拡張検索の使用方法

# 証明書要求の承認または拒否

「認証管理」タブの開始画面には、証明書の保留要求すべてのリストが表示されます。要求を承認または拒否するには、それぞれの処理に対応する次の手順に従います。

## 証明書要求の承認方法

1. 隣のラジオ・ボタンをクリックして、目的の証明書要求を選択します。
2. 「詳細表示」をクリックします。
   「証明書要求の詳細」画面が表示され、選択した証明書の情報が示されます。要求を行ったユーザーの連絡先が表示されます。ユーザーに電子メールを送信するか、電話するなどして、組織で定めたユーザー認証手順に従う必要があります。
3. 有効期間を確認し、必要に応じて変更します。
4. 下位 CA 証明書を発行する場合は、(信頼できる認証局を表示する) デフォルトのパス長は 2 と表示されます (この値は、必要に応じて変更できます)。
5. 「承認」をクリックします。
   証明書要求が承認されたことを示すメッセージが表示されます。
   証明書要求の所有者が証明書をインポートできるように、その所有者に通知してください。

## 証明書要求の拒否方法

1. 隣のラジオ・ボタンをクリックして、目的の証明書要求を選択します。要求を行ったユーザーを確認できない場合または証明書のプロパティに誤りがある場合は、証明書要求を拒否する必要があります。
2. 「詳細表示」をクリックします。
   「証明書要求の詳細」画面が表示され、選択した証明書の情報が示されます。
3. 「拒否」をクリックします。
   選択した証明書要求が拒否されたことを示すメッセージが表示されます。要求を行ったユーザーに拒否を通知してください。

# 証明書の詳細の表示

「認証管理」タブで、証明書を選択して詳細を表示できます。

単一の証明書を選択する方法は、3-14 ページの「単一の証明書要求または発行済証明書の表示」の項を参照してください。

証明書のリストを表示する方法は、3-15 ページの「拡張検索の使用方法」の項を参照してください。

検索結果から、表示する証明書を選択して「詳細表示」をクリックします。「証明書」ページが表示され、証明書の詳細が示されます (このページを使用して、選択した証明書の失効、更新またはインポートも実行できます)。

## 証明書の失効

管理者は、証明書を失効させることができます。次のいずれかの場合は、失効させる必要があります。

- 証明書の所有者がステータスを変更し、証明書を使用する権限を持たなくなった場合
- 証明書の所有者の秘密鍵が危殆化された場合

ターゲットの証明書を検索する方法は、3-14 ページの「単一の証明書要求または発行済証明書の表示」の項または 3-15 ページの「拡張検索の使用方法」の項に記載された手順を参照してください。正しい証明書を選択した後、「詳細表示」をクリックして詳細を表示するか、次の手順で証明書を失効させます。

1. 失効要求を送信するには、「失効」ボタンをクリックします。
「失効確認」画面が表示されます。この画面で、「キー危殆化」、「所属変更」、「CA 危殆化」、「証明書保留」、「運用停止」、「CRL から削除」、「破棄」または「未指定」の 8 つから、いずれかの失効理由を選択する必要があります。
2. その後、「取消」をクリックして証明書を有効にしておくか、「OK」をクリックして証明書を失効させることができます。

**関連項目：** SSO または SSL の認証を使用しているエンド・ユーザーは、自分自身の証明書を失効させることもできます。詳細は、第 7 章「Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース」の「証明書の失効」の項を参照してください。

---

**注意：**

- 管理者およびルート CA の証明書は、Web ベースのインタフェースを介して失効させることはできません。失効は、`ocactl` コマンドライン・ツール以外ではできません。
- ルート CA 証明書の失効は、影響が大きい操作です。インストールした Oracle Application Server Certificate Authority が機能しなくなり、すでに発行されている証明書が無効になります。この失効操作を実行するのは、付録 A「コマンドライン管理」の「ルート CA 証明書の失効」の項で説明するとおり、CA 鍵が危殆化された場合だけです。
- 管理者の証明書の失効が必要になるのは、鍵の危殆化または盗難、または新しい人間に管理者ロールが付与された場合などです。第 6 章「OracleAS Certificate Authority の管理 : 高度なトピック」の「OCA Web 管理者の証明書の失効」の項を参照してください。

---

## 証明書の更新

管理者は、ユーザーの証明書を中断することなく継続して使用できるように、期限切れ前後の 10 日間（デフォルトのポリシー）に更新できます（管理者は、期限切れ前後の許容日数を変更できます）。期限切れの証明書は、満了日の前後に設定した許容日数中に更新できます。証明書が期限切れになり、この許容日数中に更新しなかった場合、その証明書は使用できなくなります。このため、新しい証明書要求を送信して承認を得ることによって置換する必要があります。

証明書を更新する場合、管理者は、証明書を選択し（表示および検索に関する項を参照）、「詳細表示」をクリックして「証明書」ページを表示した後、「更新」をクリックします。日付が、証明書の満了日の前後に設定された時間枠（デフォルトでは前後 10 日間）内の場合は、証明書を更新できます。時間枠外の場合は、設定した時間枠に関するエラー・メッセージが表示されます。

SSO または SSL の認証済更新要求の場合は、ユーザーの証明書更新を制御するポリシーと同じポリシー（RenewalCertificateRequestConstraints）が自動的に適用されます。Oracle Application Server Certificate Authority で、エンド・エンティティからの更新要求が処理されると、このポリシーによって、更新された証明書に対して新しい有効期間が設定されます。

## 単一の証明書要求または発行済証明書の表示

Web ベースのユーザー・インタフェースの最初のページで、Oracle Application Server Certificate Authority 管理インタフェースを使用して、特定の証明書または証明書要求を表示できます（指定した条件を満たす証明書または要求のリストを生成する方法は、「拡張検索の使用方法」の項を参照。）

特定の証明書または証明書要求を検索するには、次の手順を実行します。

1. 「検索」プルダウン・メニューを使用します。
   - 証明書の保留要求をすべて表示するには、「すべての保留要求」を選択します。
   - 発行済の特定の証明書を表示するには、「証明書」を選択します。
   - 特定の証明書要求を表示するには、「証明書要求」を選択します。
   - 特定の要求の ID またはシリアル番号を検索するには、「ID/ シリアル番号」を選択します。
   - 特定の一般名を検索するには、「一般名」を選択します。
2. 「検索」条件フィールドに、検索要求に適した値を入力します。
   - 「すべての保留要求」には、値を指定する必要はありません。
   - 「ID/ シリアル番号」には、目的の証明書または要求のシリアル番号または要求 ID を入力します。
   - 「一般名」には、目的の一般名を入力します。

3. 「実行」をクリックします（「実行」のかわりに [Enter] を押しても動作しません）。

- 単一の証明書要求が正常に検索されると、その証明書要求を表す行が表示されます。「詳細表示」をクリックすると、要求に関する情報（連絡先、要求を行ったユーザー、有効期間など）が、「承認」および「拒否」のラベルが付いたボタンとともに表示されます。いずれのボタンをクリックしても、その要求と対応するステータスが関連付けられます。このステータスは、今後検索結果として表示した場合、常に、この証明書要求とともに表示されます。
- 証明書の保留要求がすべて正常に検索されると、それらがリストに表示されます。25 件を超える場合は、25 件ずつ表示されます。要求を識別する番号をクリックすると、詳細が表示され、要求の承認または拒否を実行できます。
- 単一の発行済証明書が正常に検索されると、「詳細表示」ボタンとともに、その証明書を表す行が表示されます。「詳細表示」をクリックすると、「失効」、「更新」または「ブラウザへのインポート」ボタンとともに証明書のデータが表示されます。「失効」ボタンを使用すると、証明書が無効になり、データベース内で「失効」というタグが付けられます。今後、「証明書失効リスト（CRL）の更新」ボタンを選択したときに、失効済証明書を持つエンティティが認証されないように、失効済証明書の最新のデータベース・リストがブラウザに格納されます。

**関連項目：**「証明書失効リスト（CRL）の更新」

## 拡張検索の使用方法

「拡張検索」機能を使用すると、次のように、より複雑な検索条件を指定して複数の証明書または証明書要求を検索および表示できます。

- 証明書要求の場合は、個別の検索で、保留要求、拒否済要求または認証済要求をすべて表示できます。
- 要求または発行済証明書の場合は、電子メール・アドレス、拡張 DN、シリアル番号または範囲、あるいは DN 内の特定のエントリ（名前、組織、州、国など）で検索できます。これらの構成要素は、連続した文字列として指定する必要があります。たとえば、cn=lakshmi, ou=st, o=oracle が所有する証明書は、検索条件として cn=lakshmi, o=oracle を指定しても、選択も検出もされません。この指定では、ou=st が指定されていないため、検索文字列が連続していません。

管理者は、検索結果から次のことを実行できます。

- 証明書検索で検索された単一の証明書のいずれかを選択し、詳細を表示した後、更新または失効（あるいはブラウザへのインポート）を実行できます。
- 証明書要求の検索で検出された単一の証明書要求のいずれかを選択し、詳細を表示した後、証明書発行の承認または拒否のいずれかを実行できます。

各タイプの検索で、検索パラメータを指定した後、「実行」ボタンをクリックします。Oracle Application Server Certificate Authority では、一度に 25 件のレコードが表示されます。

証明書要求または発行済証明書に対して拡張検索を実行するには、次の手順を実行します。

1. 「認証管理」ページの「拡張検索」をクリックします。

   結果ページは、次のセクションで構成されています。これらのセクションから特定の検索タイプを選択できます。

   - 要求ステータスを使用した証明書要求の検索（「保留」、「拒否済」または「認証済」）
   - 識別名（DN）を使用した検索（証明書または証明書要求）
   - 拡張 DN を使用した検索（証明書または証明書要求）
   - シリアル番号の範囲を使用した検索または要求 ID の範囲（証明書または証明書要求）
   - 証明書のステータスを使用した検索（有効、失効済または満了の証明書）

2. 検索タイプを指定した後、「実行」ボタンをクリックして結果のリストを表示します。

   すべての検索結果に対して、Oracle Application Server Certificate Authority では、一度に 25 件のレコードが表示されます。残りのレコードを表示するには、「前へ」および「次へ」ボタンを使用してナビゲートします。

## 要求ステータスを使用した証明書要求の検索

「拡張検索」ページのこのセクションを使用して、ステータス別に証明書要求を表示します。ドロップダウン・メニューから、「保留」、「拒否済」または「認証済」を選択して「実行」をクリックします。選択したステータスと一致する証明書要求のリストに、レコードが 25 件ずつ表示されます。

## 識別名（DN）を使用した検索

「拡張検索」ページのこのセクションを使用して、特定の所有者別の証明書を表示します。所有者にはサーバーまたはエンド・ユーザーを指定できます。発行済証明書別または要求された証明書別に検索できます。

**表 3-2 検索要素**

| 検索要素 | 範囲指定要素の意味 / 内容 |
|---|---|
| 一般名 | 検索する証明書上の名前 |
| 電子メール・アドレス | DN の一部である電子メール・アドレス |
| 組織単位 | 所有者が属する企業または組織の名前 |
| 市 / 地域 | 所有者の所在地 |
| 都道府県 | 所有者が所在する都道府県 |

**表 3-2 検索要素（続き）**

| 検索要素 | 範囲指定要素の意味 / 内容 |
|---|---|
| 国 | 所有者の国を表す 2 文字のコード |

**注意： DN および拡張 DN を使用した検索について**

DN および拡張 DN を使用した検索では、順序が連続した検索が必要になります。複数のフィールドを選択する場合または拡張 DN を使用する場合は、連続した文字列を形成する必要があります。たとえば、cn=johnDoe, ou=st, o=oracle, c=us, ou=st という有効な証明書では、o=oracle は検索文字列として有効ですが、ou=st, c=us は有効ではありません。

## 拡張 DN を使用した検索

「拡張検索」ページのこのセクションを使用して、所有者の識別名別に、発行済証明書（「証明書」）または要求された証明書（「証明書要求」）を検索します。各 RDN 文字列に対する値を入力するかわりに、完全な DN 文字列を入力できます。

**関連項目：** 付録 F「用語集」の「ドメイン・コンポーネント属性」

## シリアル番号の範囲を使用した検索

「拡張検索」ページのこのセクションを使用して、発行済証明書または要求された証明書をすべて、シリアル番号の範囲内で検索します。発行済証明書別または要求された証明書別に検索できます。いずれかを選択し、対象となる最小および最大のシリアル番号を指定して「実行」をクリックします。

**表 3-3 証明書のシリアル番号の検索範囲を指定する要素**

| 範囲指定要素 | 範囲指定要素の意味 / 内容 |
|---|---|
| 最小シリアル番号 | 範囲内の最小シリアル番号を入力します。 |
| 最大シリアル番号 | 範囲内の最大シリアル番号を入力します。 |

## 証明書のステータスを使用した検索

「拡張検索」ページのこのセクションを使用して、有効、失効済または満了の証明書をすべて検索します。これら 3 つのいずれかを選択して「実行」をクリックします。

# 証明書失効リスト（CRL）の更新

証明書を失効にすると、ユーザーの環境で使用できなくなります。失効したことを公開して、失効済証明書が誤って使用されないようにします。証明書失効リスト（CRL）と呼ばれる、失効済証明書のリストを公開すると、認証を行うエンティティは、最初にこのリストを確認できるため、誤使用を防止できます。たとえば、信頼できる環境のすべてのアプリケーションで、CRL を使用して失効済証明書の認証を防止できます。

更新済の CRL を生成するには、次の手順を実行します。

1. 「認証管理」のメイン・ページで、「証明書失効リスト（CRL）の更新」ボタンをクリックします。
   「証明書失効リストの更新」フォームが表示されます。
2. 「CRL の有効期間」に、次の更新までの日数を数値で指定します。
3. 「署名アルゴリズム」に、「RSA 付き MD5」や「RSA 付き SHA1」などをドロップダウン・メニューから選択します。

フォームに必要事項を入力した後、「送信」ボタンをクリックします。これによって CRL が生成されます。

この CRL は、「CRL のダウンロード」を選択した後、「ブラウザへのインポート」または「ローカル ディスクにダウンロード」を選択して、表示用または保存用に取得できます。

**関連項目：** 第 7 章「Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース」の「ファイル・システムへの証明書失効リスト（CRL）のダウンロード」の項

Oracle HTTP Server は、このリストを使用して、受信した SSL 証明書の妥当性を確認し、証明書が CRL にあるエンド・エンティティとの SSL 接続を拒否します。システムでこのようなサーバーが複数使用されている場合は、それらのサーバーが使用する適切なパスまたはファイル名に、サーバーの CRL としてその CRL をコピーする必要があります。各サーバーの CRL を設定するには、サーバーごとに決められた手順を実行します。

同様に、ブラウザおよび電子メールのクライアントは、これらの CRL を使用して受信した S/MIME 電子メールを検証し、接続しているサーバーを検証できます。

# Single Sign-On（SSO）および OracleAS Certificate Authority（OCA）

OCA と SSO は相互補完の関係にあり、ユーザー証明書のプロビジョニングを簡単にし、SSO を使用するすべてのアプリケーションに対する PKI 認証を、それらの証明書を使用して有効にします。この項で説明する 2 つの構成オプションを選択すると、この連携がより容易になります。

- SSO 認証済ユーザーへの OCA 証明書要求 URL のブロードキャスト
- OCA 証明書要求 URL への SSO 認証済ユーザーのアクセス

最初の構成オプションであるブロードキャストを使用すると、SSO ユーザーは、デフォルトの OCA 構成を使用するよりも簡単に、証明書要求を配信できるようになります。OCA のデフォルトでは、SSO 認証済ユーザーが証明書要求を配信すると証明書を提供するように構成されていますが、それにはいくつかの手順が必要になります。このプロセスは、第 7 章「Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース」の「Single Sign-On（SSO）認証」の項で説明します。

ブロードキャスト・オプションでは、送信可能なリンクがすべてのユーザーに提供され、ユーザーは SSO/OCA 証明書を直接要求できるようになるため、要求はさらに容易になります。

2 番目の構成オプションは、最初のオプションの説明に続いて、「OCA 証明書要求 URL への SSO 認証済ユーザーのアクセス」の項で説明します。そこでは、SSO 構成を簡略化することで、構成プロセスを大幅に短縮する OCA 構成コマンドについて説明します。SSO のデフォルト配置では、PKI 認証に必要な SSL が自動的に使用されるわけではありません。このため、OCA 提供のユーザー証明書を実行時に SSO で使用するには、SSL および証明書を使用できるように SSO を構成する必要があります。この 2 番目の構成オプションを説明する次の項では、通常のデフォルト構成を活用して、このプロセスをさらに簡略化する方法について説明します。

2 番目の構成オプションの説明には、次の 2 つの項があります。

- SSO および OCA での PKI 認証の有効化
- ユーザー証明書と SSO の使用

ここでは、OCA および SSO を使用する PKI 認証に必要なすべての手順と、Single Sign-On の認証プロセスについて説明します。

## SSO 認証済ユーザーへの OCA 証明書要求 URL のブロードキャスト

SSO ユーザーが OCA 証明書の取得に使用する URL は、埋込みの HTML リンクとして電子メールで送信できます。または、エンタープライズ・ポータルのリンクとして公開できます。これらの方法を使用することで、証明書を必要とするユーザーに対して、より柔軟にこの機能を公開できます。

SSO 証明書要求の URL は、次のとおりです。

```
https://<Oracle_HTTP_host>:<oca_ssl_port>/oca/sso_oca_link
```

もちろん、電子メールを送る際には、<Oracle_HTTP_host> をホストの Web または IP アドレスに、<oca_ssl_port> を Oracle 認証局 SSL サーバーの認証ポート番号に置換する必要があります。

oracle_HTTP_HOST は、OCA のインストール先ホストです。

oca_ssl_port は、$ORACLE_HOME/install/portlist.ini の Oracle Certificate Authority SSL Server Authentication port に記載されています。Windows の場合、このパスは $ORACLE_HOME¥install¥portlist.ini です。

これでユーザーは、このリンクをクリックし、次の項の「OCA 証明書要求 URL への SSO 認証済ユーザーのアクセス」で説明する手順と同様の手順を実行できます。

---

**注意：** インストール後にポートが変更されていると、portlist.ini に最新情報が保管されていません。この場合は、Oracle Enterprise Manager Control にサインオンし、OCA がインストールされているインスタンスをクリックします。次に、「ポート」リンクをクリックし、「タイプ」列で「OCA Server Authentication (SSL)」というエントリを探して、その横の「使用中のポート」という見出しの列に表示されている数値を使用します。

---

## OCA 証明書要求 URL への SSO 認証済ユーザーのアクセス

OCA は、SSO 認証を実行するようにデフォルトで構成されますが、いくつかの手順があります。ユーザーは、OCA ユーザー・インタフェースにアクセスし、SSO 認証を選択した後、証明書を要求する必要があります（詳細は、第 7 章「Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース」の「Single Sign-On（SSO）認証」の項を参照）。一部のユーザーには、この処理の実行が難しい場合があります。

そのため、Oracle Application Server Certificate Authority には、SSO Server による認証後、OCA 証明書要求 URL にユーザーを直接送ってユーザー・インタフェースを簡単にするメカニズムがあります。

Oracle Application Server Certificate Authority では、この URL を SSO Server に通知して、SSO がユーザー認証に証明書を使用しないときに必ずこの URL が表示されるように構成できます。SSO は、証明書を使用しないでユーザーを認証すると、そのユーザーが証明書を要求できる OCA 画面を表示します。証明書を作成し、ユーザーのブラウザにインポートすると、その後の認証では、その証明書が自動的に使用されるだけです（ただし、このポップアップ画面は、ユーザーの興味の有無に関係なくすべてのユーザーに対して表示されるため、無関係のユーザーには不便な場合もあります）。

この方法で OCA を構成する場合、管理者は（管理者用パスワードで）、ocactl コマンドライン・ツールを使用して、次のコマンドを発行します。

```
ocactl linksso
```

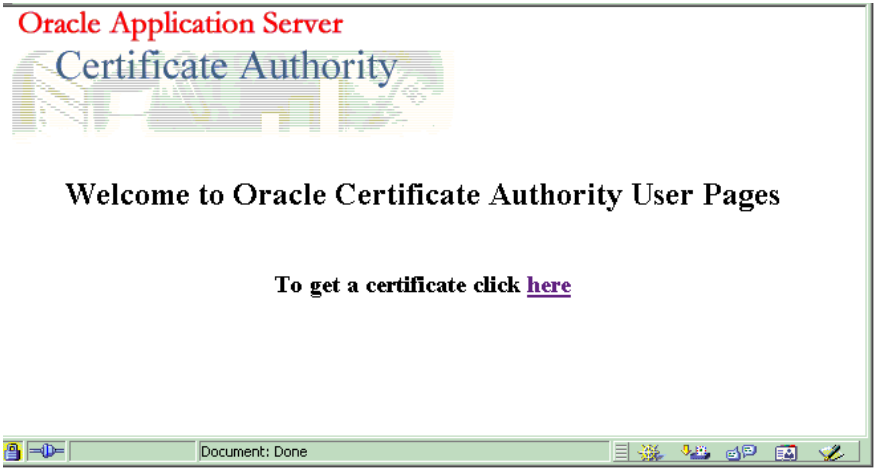
また、管理者は（管理者用パスワードで）、ocactl コマンドライン・ツールを使用して、SSO Server 経由でこの URL の使用を取り消すことができます。この処理には、次のコマンドを発行します。

```
ocactl unlinksso
```

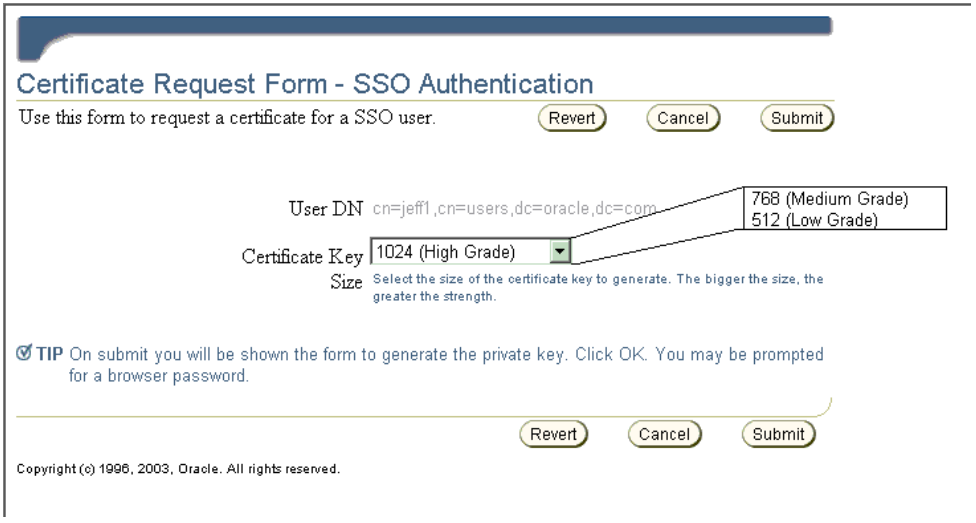
これらのコマンドを使用する場合、OCA サービスを停止する必要はありません。ただし、SSO Server の ORACLE_HOME で次のコマンドを使用し、SSO Server を再起動して有効にする必要があります。

```
$ORACLE_HOME/opmn/bin/opmnctl stopproc type=oc4j instancename=oca
$ORACLE_HOME/opmn/bin/opmnctl startproc type=oc4j instancename=oca
```

ocactl linksso コマンドを実行し、SSO Server を再起動すると、SSO が証明書を使用しないでユーザーを認証している場合は、常に、OCA の「ようこそ」ページが表示されます。次のようなページが表示されます。



SSO ユーザーが「ここ」リンクをクリックすると、次に示す OCA 証明書要求のページが表示されます。




この合成図では、SSO ユーザーが鍵のサイズを選択し、その選択したサイズが指定どおり設定された後、「送信」をクリックする必要があることを示しています（「回復」をクリックすると、選択内容がデフォルトに戻ります）。要求の送信後、この証明書の鍵が自動的に生成されます（この処理には数分かかる場合があります）。その後、この証明書は、Oracle Internet Directory にインポートされ、ユーザーに表示されます。ユーザーが証明書の情報を確認し、「ブラウザへのインポート」をクリックすると、証明書はユーザーのブラウザにインポートされ、自動使用できるようになります。

## ユーザー証明書と SSO の使用

Single Sign-On Server への OCA の再登録が完了したら、Single Sign-On を使用して OCA への認証を行っていたユーザーは、以前と同様に証明書を使用できます。

新規ユーザーは、前述の項の説明に従い、SSO 用 OCA 証明書要求 URL を使用して証明書をプロビジョニングできます。

SSO がユーザーを証明書で認識できるようになると、ユーザーは、ユーザー名 / パスワード・ログインまたは証明書のいずれかを使用して、OCA などのアプリケーションにアクセスできます。

つまり、ユーザーは、ユーザー名 / パスワードを使用してログインし、手順に従って証明書を作成してブラウザにインポートした後、PKI を介して SSO に対して自己認証を実行できます。

ユーザーのブラウザに、一部のアプリケーションの使用に認証を求める、SSO に対する証明書が表示されると、SSO は、ディレクトリを参照してその証明書を確認します。ユーザーのニックネーム（場合によってはサブスクライバ名も）で格納されている証明書とブラウザに表示されている証明書が一致している場合、認証は正常に実行されています。

Single Sign-On Server は、要求された URL にユーザーをリダイレクトできるように、ユーザー情報を含む URLC トークンをアプリケーションに提供します。この後、要求された内容を配信できます。

# OracleAS Certificate Authority のインストールのデフォルト値

表 3-4「Wallets、CRL および OHS ポートに対するインストールの値（注記 1 を参照）」に、インストールのデフォルト値と、一部の重要な Wallet のデフォルト・ロケーションや有効期間などの情報を示します。

下位 CA の深度、つまりパス長を変更する場合は、コマンドラインを使用して CA 署名 Wallet を再生成する必要があります。付録 A「コマンドライン管理」の「Oracle Application Server Certificate Authority からの下位 CA Wallet の生成」の項の説明に従い、`ocactl` を使用してください。

ただし、CA を再生成すると、以前に発行された証明書はすべて無効になります。そのため、パス長の値を変更する場合は、インストール後に CA 署名 Wallet をただちに再生成する必要があります。SSL Wallet など、依存する Wallet もすべて同様です。

---

**注意：** 1 つのリポジトリでの OracleAS Certificate Authority のスキーマは、1 つの OCA とのみ併用できます。

別の OracleAS Certificate Authority をインストールする場合、先行する OCA のインストールに使用したリポジトリは選択できません。同じリポジトリを選択すると、OCA 構成ツールが正常に実行されません。

それによって、インストール処理を途中で終了し、インストール全体をやり直すことが必要になります。

---

**表 3-4 Wallets、CRL および OHS ポートに対するインストールの値（注記 1 を参照）**

| Wallet または値のタイプ | デフォルト DN | デフォルトの鍵のサイズ | デフォルトの有効期間 | その他の値 | この Wallet または値のロケーション |
|---|---|---|---|---|---|
| CA 署名 Wallet | この DN はインストール時に入力される（注記 2 を参照） | 2048（注記 2 と 3 を参照） | 3560 日 | デフォルトのパス長 = 3 | データベース |
| CA SSL Wallet | cn=<hostname> + CA の DN（CA の CN を除く） | 1024（注記 4 を参照） | 730 日 | | $OH/oca/wallet/ssl（注記 5 を参照） |
| OCA 仮想ホストの OHS ポート | -- | -- | -- | 4400 および 4401（注記 6 を参照） | $OH/Apache/Apache/conf/ocm_apache.conf（注記 7 を参照） |
| 証明書失効リスト | -- | -- | 1 日 | -- | -- |

表 3-4「Wallets、CRL および OHS ポートに対するインストールの値（注記 1 を参照）」の注記

1. 複数のプロパティに ocactl を使用します。
2. 証明書の署名に CA 署名 Wallet を使用する場合、インストール時に変更できるのは DN および鍵のサイズのみです。

---

**注意：**

DN の DC コンポーネントおよび EMAIL コンポーネントでは、印刷可能な（ASCII）文字のみを使用する必要があります。

この制限は、マルチバイト・キャラクタ・セットを使用するロケールでも、識別名の DC コンポーネントおよび EMAIL コンポーネントには ASCII 文字を使用する必要があるという意味です。

---

3. CA 署名 Wallet の場合は、`ocactl generatewallet -type CA` を実行して CA 署名 Wallet を再生成することで、インストール後にすべての要素を変更できます。また、デフォルトの有効期間は、新しい有効期間で証明書を更新することで変更できます。

4. Certificate Authority をホストする HTTP Server で使用されます。すべての CA SSL Wallet 値は、`ocactl generatewallet -type CASSL` を実行することで変更できます。CA SSL Wallet は、コマンドライン・オプションを使用して、いつでも再生成できます（期限切れの後も含む）。また、VeriSign などの異なる CA の SSL Wallet と置換することもできます。これは、最初に OCA に接続したときに、「CA 証明書が信頼できません」という警告を回避する目的で実行できます。
5. $OH は $ORACLE_HOME を意味します。したがって、完全なロケーションは $ORACLE_HOME/oca/wallet/ssl です。
6. OCA を複数インストールする場合などは、4402 から 4419 までのポートを使用できます。
7. $OH は $ORACLE_HOME を意味します。したがって、完全なロケーションは $ORACLE_HOME/Apache/conf/ocm_apache.conf です。

---

**注意：** ocm_apache.conf ファイルには、OCA のリスナー・ポートが 2 つ定義されています。

2 つ必要な理由は、証明書を必要としない機能と必要とする機能があるためです。

Apache で ClientCertificate オプション・ディレクティブを使用した場合、証明書に関連するダイアログが常に表示されることになるため、この方法より、リスナー・ポートを 2 つ使用する方法のほうが適切です。

---

## SSO および OCA での PKI 認証の有効化

証明書を使用するように SSO を構成するには、いくつかの手順を実行する必要があります。付録 E にこれらの手順のすべてを示していますが、詳細なコンテキストと説明については、『Oracle Application Server Single Sign-On 管理者ガイド』に記載しているため、そちらのガイドも参照してください。一般的な実行手順の概要を次に示します。

1. 前述のガイドの第 9 章の説明に従い、SSL を有効化します。Java と PL/SQL で内容が異なりますが、Java の項の手順に従ってください。
2. 同じガイドの第 7 章の説明に従い、証明書用に SSO を構成します。
3. 付録 E の「SSL を有効化した SSO への、OCA の仮想ホストの再登録」の項の説明に従い、OCA の仮想ホストを Single Sign-On Server に再登録します。

PKI を有効にすると、SSO Server は、ユーザー名およびパスワードを要求するかわりに、証明書を使用してアプリケーションのユーザーを認証できます。OracleAS Single Sign-On のパートナ・アプリケーションのユーザーが SSO 認証を選択すると、そのアプリケーションへのログインに使用する証明書を選択するよう、ブラウザから指示されます。使用する証明書は、以前にブラウザにインポートした証明書です。目的の証明書を選択すると、SSO Server サーバーはその証明書を使用してユーザーを認証し、ユーザーが要求したパートナ・アプリケーションにユーザーをリダイレクトします。

この処理では、次のことが問題になります。

- ユーザーは OCA にログオンして自分の証明書を取得する必要があります。
- OCA も OracleAS Single Sign-On 認証サービスを使用するため、証明書のないユーザーは OCA にログオンできません。

この問題は、OracleAS Single Sign-On Server で複数の認証レベルを使用することによって解決されます。PKI が有効化されると、どのパートナ・アプリケーションでも、中 - 高レベルのセキュリティ（証明書による認証）が使用されますが、OCA では、ユーザー名 / パスワードまたは Windows のネイティブ認証による中レベルのセキュリティを使用できます。それによって、OCA は証明書を発行する前にパスワードを使用してユーザーを認証できるようになり、その一方で、それ以外の SSO 対応アプリケーションには、証明書を使用した認証が強制されます。

ユーザー名 / パスワードによる中レベルのセキュリティを OCA で使用するための構成手順など、すべての手順は、付録 E を参照してください。セキュリティ・レベル固有の手順は、付録 E の「SSO での PKI の有効化」に記載しています。

同様に、Windows のネイティブ認証など、他の認証メカニズムを使用するように OCA を構成することもできます。目的の認証メカニズムを実装するプラグインにセキュリティ・レベルを割り当て、（「SSO での PKI の有効化」の）手順 3 で説明しているように、そのセキュリティ・レベルを使用するように OCA URL を割り当てます。

> **関連項目：** 詳細は、『Oracle Application Server Single Sign-On 管理者ガイド』の第 6 章「マルチレベル認証」を参照してください。

# 4

# Oracle Application Server Certificate Authority の構成

Oracle Application Server Certificate Authority の Web ベースの管理インタフェースは、次に示す 3 つの大きな事項を対象としています。各事項は、ホームページのタブからアクセスできます。

- 発行済証明書に関する事項 : 証明書の発行、失効および更新の要求、および証明書失効リスト（CRL）
- 構成に関する事項 : Oracle Application Server Certificate Authority のアクション用パラメータおよび証明書のセキュリティ・ポリシーを実装するためのパラメータ
- Oracle Application Server Certificate Authority アクティビティのログの表示

この章では、認証局運用規程に指定する内容とあわせて、前述の 2 つ目と 3 つ目の事項（構成管理およびログの表示）について説明します。

この章の内容は、次のとおりです。


- 管理インタフェースの構成
- 「構成管理」タブ
- 「ログの表示」タブ
- 認証局運用規程の作成および更新

# 管理インタフェースの構成

次の図に示すように、Oracle Application Server Certificate Authority のグラフィカル・ユーザー・インタフェース（GUI）のホームページには、この他に 3 つのタブがあります。


これらの 3 つのサブタブを使用して、証明書または認証局の構成を管理する特定のタスクを実行できます。

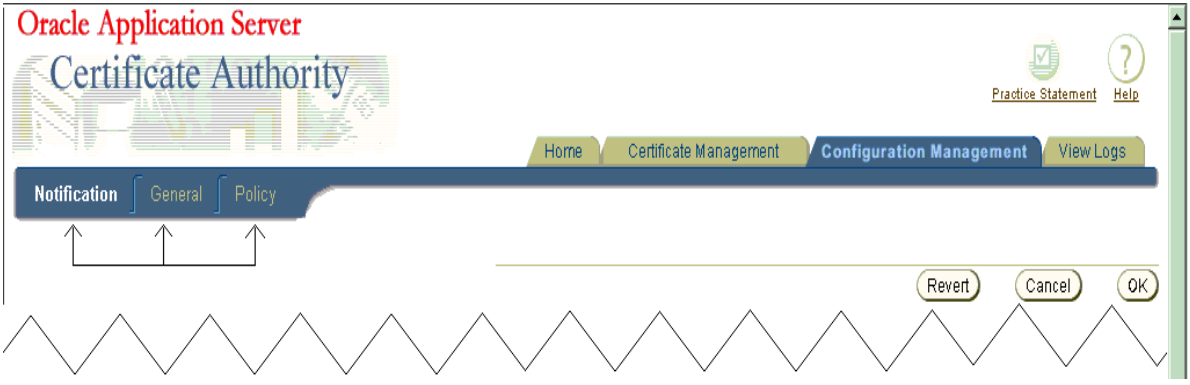
- 「認証管理」タブについては、第 3 章で説明しています。特に「証明書の管理」の項を参照してください。
- 「構成管理」タブについては、この章で説明しています。
- 「ログの表示」タブについては、この章で説明しています。

# 「構成管理」タブ

「構成管理」タブは、Oracle Application Server Certificate Authority の Web 環境に初めてアクセスする際に選択可能な 4 つの項目のうちの 1 つです。ホームページで「構成管理」タブをクリックすると、1 つ目のサブタブが表示されます。各サブタブでは、Oracle Application Server Certificate Authority の構成管理機能が分類されています。

次の項で、これらのサブタブの内容および使用方法について説明します。

- 構成タスクの概要
- 「通知」サブタブ
- 「一般」サブタブ
- Oracle Application Server Certificate Authority の「ポリシー」サブタブおよびポリシー操作については、第 5 章「Oracle Application Server Certificate Authority でのポリシー管理」で説明しています。



# 構成タスクの概要

表 4-1、表 4-2 および表 4-3 に、「構成管理」の「通知」、「一般」および「ポリシー」という各サブタブで実行するタスクを示します。

**表 4-1 「構成管理」の「通知」サブタブのタスクおよび説明**

| 「通知」サブタブのタスクおよびデータ | 参照先 |
|---|---|
| アラートと通知の宛先となるサーバー名および電子メール・アドレスを指定する。 | ■ メール詳細 |
| 表示するアラート・タイプを指定する。 | ■ アラート |
| CRL の生成間隔、CRL の検証間隔およびディレクトリの同期間隔を指定する。 | ■ スケジュールされたジョブ |

**表 4-2 「構成管理」の「一般」サブタブのタスクおよび説明**

| 「一般」サブタブのタスクおよびデータ | 参照先 |
|---|---|
| Oracle Internet Directory とあわせて、SSL 通信チャネルまたは非 SSL 通信チャネルを、証明書の公開に使用することを指定する。 | ■ 証明書の公開 |
| 証明書の管理のために、エンド・ユーザーが SSL 認証および SSO 認証を使用できることを指定する。 | ■ SSL 認証および SSO 認証 |
| ロギングまたはトレース（両方を実行すること、あるいはどちらも実行しないこと）を指定する。 | ■ ロギングおよびトレース |
| 登録情報に示される DN コンポーネントのデフォルト値を指定する。 | ■ デフォルトのベース DN コンポーネント |
| データベースおよびディレクトリの構成パラメータを表示する。 | ■ データベースの設定 , ディレクトリの設定 |

**表 4-3 「構成管理」の「ポリシー」サブタブのタスクおよび説明**

| Oracle Application Server Certificate Authority の「ポリシー」サブタブのタスクおよびデータ（第 5 章） | 参照先 |
|---|---|
| 使用可能な操作（証明書の要求、失効、更新など）に適用可能なポリシーを参照する。 | ■ 製品に付属の証明書要求ポリシー<br>■ 製品に付属の証明書失効ポリシー<br>■ 製品に付属の証明書更新ポリシー<br>■ ポリシー操作 |
| ポリシーの編集、有効化、無効化、削除、追加および並び替え。 | |

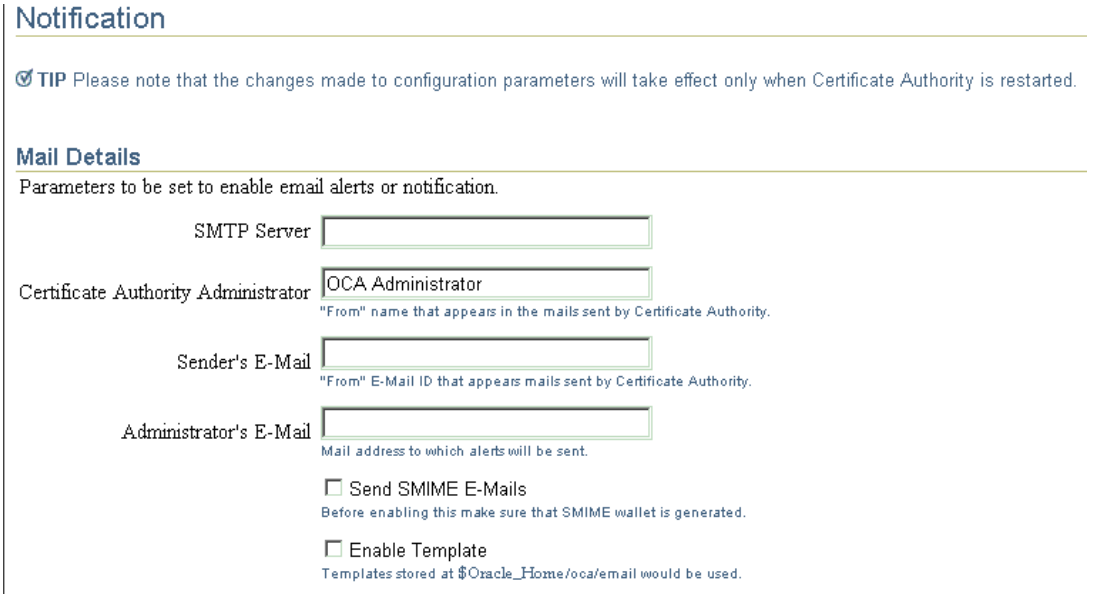
# 「通知」サブタブ

「通知」パラメータでは、管理者への通知電子メールをトリガーするイベント、通知電子メールの生成方法およびこれらのイベントを検出する頻度を制御します。

「通知」構成パラメータの変更を有効にするには、Oracle Application Server Certificate Authority を再起動する必要があります。

## メール詳細

「メール」パラメータを使用すると、管理者として指定した電子メール・アドレスおよびOCA ユーザー（必要に応じて判断される）に、暗号または平文による電子メール通知を送信できるようになります。その際、指定したサーバー、送信者およびテンプレートが使用されます。「通知」サブタブ画面で、次の項目を指定します。




インストール後は、「テンプレートの有効化」の下のヒントに、テンプレート・ディレクトリへの正確なパスが表示されます。たとえば、インストール時に $Oracle_Home を /private/sitename/username に定義した場合は、このヒントには「/private/sitename/username/oca/email に格納されているテンプレートが使用されます。」と表示されます。

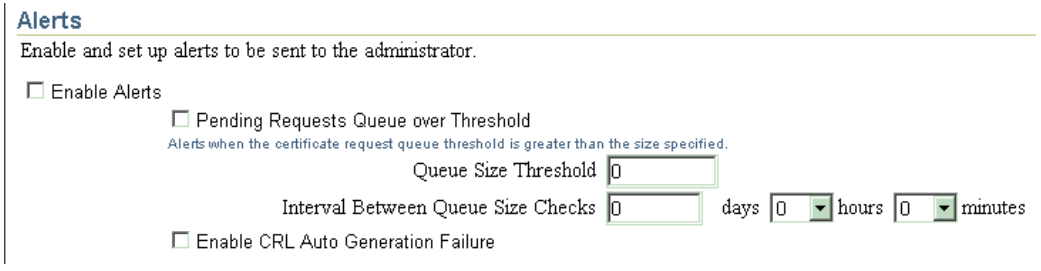
**関連項目：** 第 6 章「OracleAS Certificate Authority の管理 : 高度なトピック」の「CA SSL Wallet および CA SMIME Wallet の再生成」の項

## アラート

「アラート」パラメータを使用すると、次の場合にアラートを受信するかどうかを指定できます。

- 証明書の保留要求の数が、ここで指定したキューのしきい値を超え、指定したスケジュールで検証が行われる場合。
- CRL の自動生成に失敗した場合。たとえば、データベースまたは Oracle Internet Directory が一時的に使用できない場合、CRL の自動生成に失敗します。他にも、メモリー、入出力または接続性に関連する、予期しないランタイム・エラーまたは構成エラーが発生した場合も、CRL の自動生成に失敗します。

「通知」サブタブ画面で、次の項目を指定します。

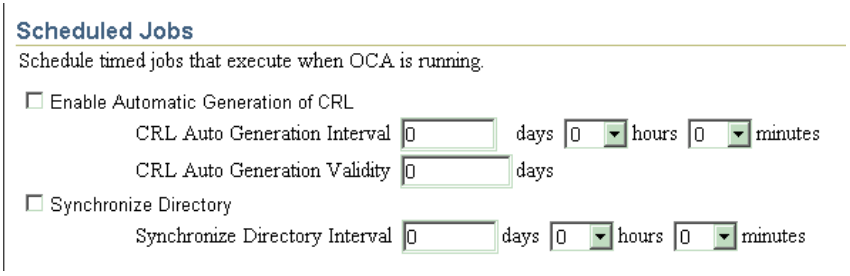



## スケジュールされたジョブ

「スケジュールされたジョブ」パラメータを使用すると、自動ジョブについて次の項目を選択できます。

- CRL を自動生成するかどうかと、その頻度。この機能によって、失効または期限切れになった証明書の検出に CRL を使用するアプリケーションをサポートする処理を、定期的かつ確実に実行できます。
- ディレクトリを同期化するかどうかと、その頻度。この機能によって、Oracle Internet Directory に格納されている証明書の情報が、適切なタイミングで定期的に更新されます。ディレクトリが一時的に停止している間に証明書を発行（または失効や期限切れ）した場合でも、同期化中は公開（または削除）されます。

「通知」サブタブ画面で、次の項目を指定します。




# 電子メールのテンプレート

管理者は「通知」サブタブの「メール詳細」で該当のチェック・ボックスを選択することによって、テンプレートを有効化できます。その後、電子メールのアラートおよび通知の本文をテンプレートとして指定し、カスタマイズできます。これらのテンプレートは次のディレクトリに格納されています。

$ORACLE_HOME/oca/templates/email

次に示すトークンを使用して電子メールの書式を設定することで、特定の情報を提供できます。これらのトークンは電子メールの送信前に置換されます。表 4-4 に、通知、電子メールの書式のファイル名、およびサポートされているトークンを示します。

**表 4-4 通知、テンプレート、および電子メールのカスタマイズに使用できるトークン**

| 通知 | テンプレートのファイル名 | サポートされているトークン |
|---|---|---|
| CertificateRequestNotify | reqacc.txt | #NAME#、#REQUESTID#、#SUBJECTDN#、#PHONE#、#EMAIL# |
| RequestApprovalNotify | reqapp.txt | #NAME#、#REQUESTID#、#SUBJECTDN#、#SERIALNUM#、#OCAURL#、#PHONE#、#EMAIL#、#VALIDITY# |
| RequestRejectionNotify | reqrej.txt | #NAME#、#REQUESTID#、#SUBJECTDN#、#PHONE#、#EMAIL# |
| PendingRequestsAlert | pendreq.txt | #NAME#、#NUMBERREQUESTS# |
| CRLAutoGenFailureAlert | crlfail.txt | #NAME# |

**注意：** 「通知」画面の「構成管理」で「テンプレートを使用する」チェック・ボックスを選択していないと、テンプレートは使用されません。すべてのアラートおよび通知のテキストは事前に定義されており、変更はできません。

## トークンの値

表 4-5 に、アラートまたは通知の送信前に各トークンと置換される値を示します。

**表 4-5 通知およびテンプレートのカスタマイズに使用できるトークンの値**

| 通知およびテンプレートのファイル名 | サポートされているトークンおよびそれと置換されるデータ |
|---|---|
| CertificateRequestNotify<br>テンプレート =reqacc.txt | #NAME#: 証明書要求に指定された連絡先データの名前に置換されます。<br>#REQUESTID#: この要求に対して OCA が発行する要求 ID に置換されます。<br>#SUBJECTDN#: 証明書要求の DN に置換されます。<br>#PHONE#: 証明書要求の連絡先データの電話番号に置換されます。<br>#EMAIL#: 証明書要求の連絡先データの電子メール・アドレスに置換されます。 |
| RequestApprovalNotify<br>テンプレート =reqapp.txt | #NAME#: 証明書要求に指定された連絡先データの名前に置換されます。<br>#REQUESTID#: この要求に対して OCA が発行する要求 ID に置換されます。<br>#SUBJECTDN#: 証明書要求の DN に置換されます。<br>#SERIALNUM#: 証明書のシリアル番号に置換されます。<br>#OCAURL#: ユーザーのホームページの URL に置換されます。<br>#PHONE#: 証明書要求の連絡先データの電話番号に置換されます。<br>#EMAIL#: 証明書要求の連絡先データの電子メール・アドレスに置換されます。<br>#VALIDITY#: 管理者によるその証明書要求の承認の有効期間に置換されます。 |

**表 4-5 通知およびテンプレートのカスタマイズに使用できるトークンの値（続き）**

| 通知およびテンプレートのファイル名 | サポートされているトークンおよびそれと置換されるデータ |
|---|---|
| RequestRejectionNotify<br>テンプレート reqrej.txt | #NAME#: 証明書要求の連絡先データの名前に置換されます。<br>#REQUESTID#: この要求に対して OCA が発行する要求 ID に置換されます。<br>#SUBJECTDN#: 証明書要求の DN に置換されます。<br>#PHONE#: 証明書要求の連絡先データの電話番号に置換されます。<br>#EMAIL#: 証明書要求の連絡先データの電子メール・アドレスに置換されます。 |
| PendingRequestsAlert<br>テンプレート =pendreq.txt | #NAME#: 「通知」画面の「構成管理」にある「OracleAS Certificate Authority 管理者」フィールドに指定されている値に置換されます。<br>#NUMBERREQUESTS#: OCA リポジトリ内の保留要求の数に置換されます。 |
| CRLAutoGenFailureAlert<br>テンプレート =crlfail.txt | #NAME#: 「通知」画面の「構成管理」にある「OracleAS Certificate Authority 管理者」フィールドに指定されている値に置換されます。 |

**注意：**

これらのテンプレートの編集に使用した言語が最終的な結果でも使用されるため、編集にはサーバーの言語を使用することをお薦めします。メッセージ本文はサーバーのロケールの言語でエンコードされます。

テンプレートを使用しない場合、アラートと通知はすべて、サーバーのロケールの言語で表示されます。

# 「一般」サブタブ

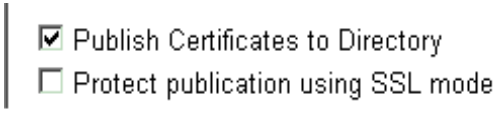
このサブタブを使用すると、次のタスクを制御するパラメータを設定できます。

- 証明書の公開
- SSL 認証および SSO 認証
- ロギングおよびトレース
- デフォルトのベース DN コンポーネント
- データベースの設定
- ディレクトリの設定

「一般」構成パラメータの変更を有効にするには、Oracle Application Server Certificate Authority を再起動する必要があります。

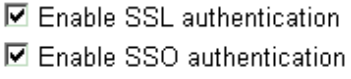
## 証明書の公開

この項の手順を実行することで、ディレクトリへの証明書を公開できます。OCA は常に、SSL ポートを使用して Oracle Internet Directory に接続しているため、ここに表示される 2 番目のチェック・ボックス（「SSL モードを使用した発行の保護」）は不要になります。Diffie-Hellman による SSL 直接接続は認証を必要としないため、OCA は、セキュアになった SSL 接続でユーザー名とパスワードを送信することによって、ディレクトリ・サーバーに対して自己認証します。



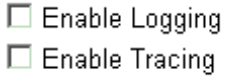

## SSL 認証および SSO 認証

ここでは、SSL ユーザーまたは SSO ユーザーが自動認識可能かどうかを指定できます。この指定によって、ユーザーの既存の証明書（または SSO 認証）が、ユーザーの識別情報を認証しているものとして受け入れられます。これらの項目は、デフォルトでは有効になっており、管理者が介入しなくても、Oracle Application Server Certificate Authority によって、新しい証明書がユーザーに発行されます。



## ロギングおよびトレース

ここでは、すべてのユーザー・アクティビティのログ・ファイルを作成するかどうか、またはすべてのエラー詳細のトレース・ファイルを作成するかどうか、あるいはその両方を作成するかを指定できます。




ログは OCA リポジトリに格納されます。「ログの表示」タブから、これらのログを参照できます。トレースは、ファイル・システムの $ORACLE_HOME/oca/logs/oca.trc ファイルに格納されます。

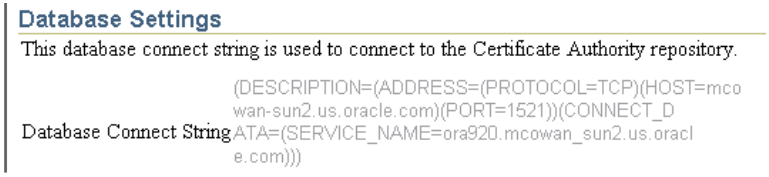
## デフォルトのベース DN コンポーネント

ここで入力した値は、手動の登録情報申請フォームにあるいくつかの識別名要素の事前入力に使用され、証明書要求の送信に使用されます。


この機能はユーザーの利便性のみを目的としており、一般的なフィールドを補うためのものです。ここで入力した値は、必要に応じて変更できます。

## データベースの設定

ここでは、Oracle Application Server Certificate Authority リポジトリへの接続に使用されているデータベース接続文字列が表示されます。

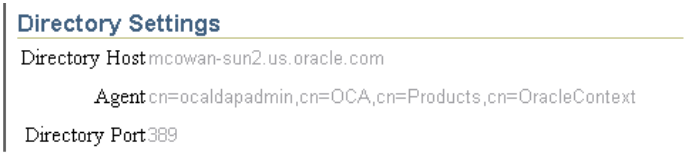



この設定は、OracleAS Certificate Authority のリポジトリが新しい場所に移動された場合、または接続文字列が変更された場合にのみ変更されます。これには、接続に使用するノードやポートの変更などがあります。この場合、`ocactl updateconnection` コマンドを使用して、リポジトリの接続設定を更新できます。その後、OCA を再起動すると、新しい接続情報が使用されます。

> **関連項目：** 表 A-2「OracleAS Certificate Authority（OCA）ocactl ツールの操作およびパラメータ」にある `updateconnection`

## ディレクトリの設定

ここでは、Oracle Internet Directory との接続に使用されているホスト、エージェントおよびポートが表示されます。接続文字列が変更された場合、`ocactl updateconnection` コマンドを使用して、リポジトリの接続設定を更新できます。その後、OCA を再起動すると、新しい接続情報が使用されます。



# 「ログの表示」タブ

この構成管理ページでは、Oracle Application Server Certificate Authority の使用中に発生するトランザクションまたはエラーに関して、メッセージを記録したログを参照できます。次のような画面が表示されます。


これらのログの各行は、ログの ID 番号、クライアントのアクティビティが開始された IP アドレスおよび操作の実行日で始まる 6 つの要素で構成されています。各行には、ログのエントリ・タイプ、エントリを生成した Oracle Application Server Certificate Authority のコンポーネントおよびアクティビティに関するコンポーネントのメッセージも含まれます。

# 認証局運用規程の作成および更新

認証局運用規程は、サイトと認証局（後述）に適用されるポリシーおよび手続きについて定義しています。通常、次のような情報が含まれます。

- 法律上の注意事項、義務および責任
- 証明書の使用についての警告
- 公開鍵インフラストラクチャを使用する際に必要な知識
- 使用されている規格またはプロトコル
- 証明書固有のデータ：
  - ライフ・サイクルの詳細
  - 制限
  - 主要な長所およびセキュリティに関する関連事項
- サイトにおける認証局の階層
- 提供されるサービス
- 証明書の入手、失効または更新方法
- 連絡先

$ORACLE_HOME/oca/help/Help/oca_cps.html ファイルを編集すると、認証局運用規程（CPS）を追加または変更できます。

Oracle Application Server Certificate Authority を再起動した後、各ページに表示される「運用規定」アイコンをクリックすると、「運用」ページに変更内容が表示されます。

---

**注意：** 前述の手順に従って OCA 管理者が作成する認証局運用規程は、外国語に対応するものではありません。このことは、OCA サーバーの言語とは異なる言語のクライアントでは、サーバーの言語でのみ認証局運用規程を表示できることを意味します。

---

# 5

# Oracle Application Server Certificate Authority でのポリシー管理

Oracle Application Server Certificate Authority は、組織で指定したポリシーを自動的に施行して、証明書の発行、失効または更新の要求に適用します。Oracle Application Server Certificate Authority で提供されるポリシー・ルールは、標準的な必要事項をサポートします。ただし、Oracle Application Server Certificate Authority の Web ベースのインタフェースの「構成管理」タブを使用したり、サイトの必要性に応じたカスタム・ポリシー・プラグインを追加することによって、管理者が構成することもできます。管理者は、必要に応じてこれらのポリシーを無効にして回避することもできます。

この章では、カスタム・ポリシー・プラグインを開発するツールなど、Oracle Application Server Certificate Authority のポリシー管理コンポーネントについて説明します。

この章の内容は、次のとおりです。


- 定義
- ポリシー管理の概要
- Oracle Application Server Certificate Authority のポリシー
- Oracle Application Server Certificate Authority の「ポリシー」サブタブ
- ポリシー・ルールの条件
- カスタム・ポリシー・プラグインの開発

# 定義

**表 5-1 OracleAS Certificate Authority でのポリシーの概念、用語および定義**

| 概念または用語 | 定義 |
|---|---|
| ポリシー・ルールまたはポリシー | Oracle Application Server Certificate Authority におけるポリシー・ルールとは、証明書や要求などに適用されるパラメータ値のデフォルトおよび範囲のセットです。たとえば、有効期間のポリシー・ルールには、365 日（最小有効期間）、730 日（デフォルト）および 3650 日（最大有効期間）を指定できます。<br><br>ポリシー・ルールには、ルールの用途を制限または変更する条件を含めることもできます。条件を指定しない場合、更新などの特定の操作へのポリシー・ルールが、すべての要求に適用されます。 |
| 条件 | Oracle Application Server Certificate Authority における条件とは、証明書または証明書要求のタイプを識別するために作成する式および対応する値です。証明書または証明書要求のタイプが条件式と一致すると、要求の妥当性を評価するために、ポリシーのデフォルト値のかわりに、これらの対応する値が使用されます。<br><br>条件が使用できるのは OCA のデフォルト・ポリシーのみで、「カスタム・ポリシー・プラグインの開発」の項で説明するカスタム・ポリシーでは使用できません。<br><br>例 : Type=="client"、Type=="server" または Type=="*" |
| プラグイン | ポリシー・ルールを実装する Java クラス |

# ポリシー管理の概要

ポリシー管理とは、組織的な制約を施行するために Oracle Application Server Certificate Authority 管理者が選択したポリシー（ルールのセット）の定式化および適用を意味します。制約には、鍵のアルゴリズム、鍵のサイズおよび有効期間をユーザーが選択するための項目などがあります。

管理者は、Oracle Application Server Certificate Authority で提供されたポリシーを使用して、次の操作を定義できます。

- Oracle Application Server Certificate Authority（OCA）が、受信した要求（証明書の発行、失効および更新）を評価する方法
- CA が証明書のパラメータ（有効期間や鍵の長さなど）に適用する制限、またはサブジェクト名および使用方法が同じ証明書を複数発行する際に適用する制限

Oracle Application Server Certificate Authority の Web ベースのインタフェースで、「構成管理」タブの編集機能を使用して、ポリシー・ルールの有効化、無効化または変更を行うことができます。「Oracle Application Server Certificate Authority の「ポリシー」サブタブ」の項を参照してください。

新しいルールを作成したり、ルールを具体化するポリシー・プラグインを開発することもできます。各ルールは、管理者が選択した評価または制限を実装するポリシー・プラグイン（Java クラス）に具体化されます。ポリシー・ルールとポリシー・プラグインは、1 対 1 でマッピングされます。Oracle Application Server Certificate Authority のデフォルトのプラグインは、一般的に必要なほぼすべてのポリシー構成を対象としています。ポリシー・プラグインを作成する場合、5-30 ページの「カスタム・ポリシー・プラグインの開発」で説明するように、管理者は適切なプログラミング手法に従い、Oracle Application Server Certificate Authority パッケージが提供する API を使用する必要があります。

サイト固有のポリシーを定義する新しいプラグインを開発した後、同じ「ポリシー」サブタブを使用して、そのプラグインに名前を付け、Oracle Application Server Certificate Authority に登録することができます。プラグインを有効にすると、Oracle Application Server Certificate Authority は、プラグインに定義されているとおりに新しいルールを施行します。

ポリシー・ルールは、Oracle Application Server Certificate Authority エンジンのポリシー・プロセッサ・モジュールによって施行されます。このプロセッサ・モジュールは、すべての有効なルールを順次施行します。有効化されていないルール、または無効化されたルールは施行されません。「ポリシー」サブタブの各操作の「ポリシー・ルール」ページで指定した順序が使用されます。つまり、プロセッサ・モジュールは、各操作の「ポリシー・ルール」ページで指定した順に、ポリシー・プラグインをコールします。受信したあらゆる要求は、操作のタイプ（要求、更新、失効など）に対応する、適切で有効なすべてのポリシー・ルールの対象となります。ルールが有効で、受信した要求と条件が一致しない場合、その要求は拒否されます。

各ポリシー・ルールは、証明書の発行、失効または更新の要求のうち、1 つ以上の属性に関係します。たとえば、ある属性は、RSA アルゴリズムで使用する鍵の最小サイズおよび最大サイズに関係します。関係するデフォルト・ポリシーは、このようなすべての属性が、アルゴリズムの有効範囲内にあることを検証します。

ポリシーは、管理インタフェースの「ポリシー」サブタブを使用した Web ベースのインタフェースを介して管理されます。

条件を含むポリシー処理の詳細は、「ポリシー・ルールの条件」の項を参照してください。

# Oracle Application Server Certificate Authority のポリシー

Oracle Application Server Certificate Authority では、制約固有のポリシー・ルールを提供しています。このポリシー・ルールは、証明書の登録、失効または更新の受信要求をポリシー・プロセッサが評価する際に使用します。各ルールの範囲内で、Oracle Application Server Certificate Authority を構成して、特定の属性について、受信した要求を検証できます。また、これらの属性を受け入れたり、変更したり、要求を拒否することもできます。

ポリシー・ルールが有効な場合、Oracle 認証局サーバーによって、処理中の証明書要求にルールが適用されます。

表 5-2 に、制約固有のデフォルトのポリシー・ルールを示します。最初の列に、各ポリシー・ルールについての参照先を示します。

**表 5-2 制約固有のデフォルトのポリシー・ルール**

| ポリシー・ルール名 | 機能 | デフォルトの状態 |
|---|---|---|
| RSAKeyConstraints | 鍵の長さに制約を施行する | 有効 |
| ValidityRule | 指定した有効期間を証明書に施行する | 有効 |
| UniqueCertificateConstraint | 同じ使用方法で同じ名前のサブジェクトに複数の証明書を発行することを禁止する | 有効 |
| RevocationConstraints | 期限切れの証明書の失効要求を許可または拒否する | 有効 |
| RenewalRequestConstraint | 期限切れの証明書の更新要求を許可または拒否する | 有効 |

## RSAKeyConstraints

RSAKeyConstraints ポリシー・ルールは、RSA の公開鍵 / 秘密鍵に使用する鍵の最小サイズおよび最大サイズの制約を適用します。

表 5-3 に、RSA 鍵の制約モジュールのパラメータを示します。

**表 5-3 RSAKeyConstraints ポリシー・ルールのパラメータ**

| パラメータ | 説明 |
|---|---|
| Status（有効または無効）<br><br>デフォルト : 有効 | 「ポリシー・ルール」ページで、ルールが有効か無効かを指定します。<br><br>ルールを有効にして他のパラメータを適切に設定すると、Oracle Certificate Manager は、条件式で指定した証明書にルールを適用します。<br><br>ルールを無効にすると、Oracle Certificate Manager は、512 ～ 4096 で 16 の倍数の RSA 鍵のサイズを許可します。 |

**表 5-3 RSAKeyConstraints ポリシー・ルールのパラメータ（続き）**

| パラメータ | 説明 |
|---|---|
| predicate<br><br>デフォルト : "*" | このルールの条件式を指定して、ルールを適用する証明書のタイプを制限します。証明書要求にルールを適用する場合は、このフィールドに「*」を入力します。<br><br>例 : Type=="client"<br>Type=="*"<br><br>「ポリシー・ルールの条件」の項を参照してください。 |
| minSize<br><br>デフォルト : 512 | RSA 鍵の最小の長さ（ビット単位のモジュールの長さ）を指定します。maxSize パラメータで指定した値以下の値を設定する必要があります。<br><br>有効値 : 512、1024、2048 または 4096 ビット |
| maxSize<br><br>デフォルト : 2048 | RSA 鍵の最大の長さ（ビット単位のモジュールの長さ）を指定します。minSize パラメータで指定した値以上の値を設定する必要があります。<br><br>有効値 : 512、1024、2048 または 4096 ビット |

管理者は、複雑な条件式を使用して、複数の組合せで predicate、minSize および maxSize を指定できます。

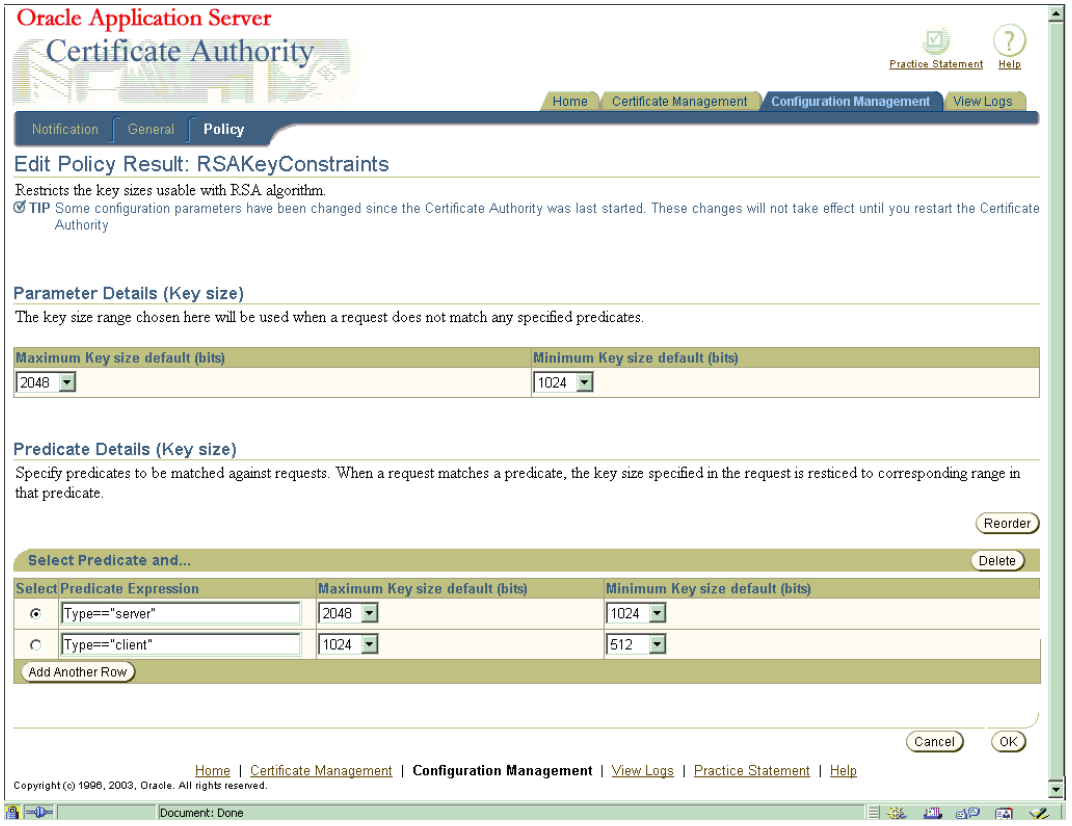
たとえば、ある組織では、最小サイズと最大サイズが、Sales 部門では 512 と 1024、Marketing 部門では 1024 と 2048 に設定する必要があるとします。複数の条件式および値のセットを使用して、この要件を指定できます。

条件 1: dn=="ou=Sales"

minSize は 512、maxSize は 1024 に指定します。

条件 2: dn=="ou=Marketing"

minSize は 1024、maxSize は 2048 に指定します。




## ValidityRule

ValidityRule ポリシー・ルールは、証明書要求の有効期間が適切かどうかを判断し、次の方法で最小および最大の有効期間を施行します。

- （SSO 認証または SSL 認証による）自動認証ユーザーの証明書要求では、このルールにより有効期間が設定されます。
- 手動認証ユーザーの証明書またはサーバーの証明書に対する要求がポリシーと一致しない場合、その要求は拒否されます。

表 5-4 に、発行有効期間の制約モジュールのパラメータを示します。この項の最後に、Web ベースのインタフェースでの表示を示します。

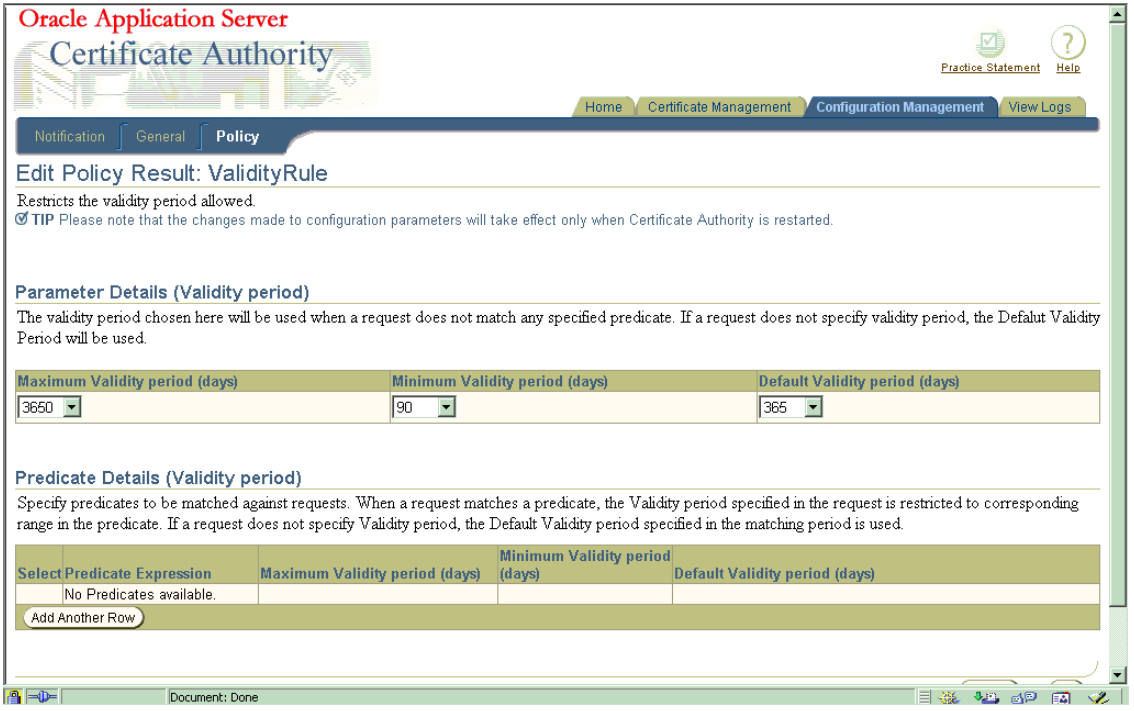
**表 5-4 ValidityRule ポリシーのパラメータ**

| パラメータ | 説明 |
|---|---|
| Status（有効または無効）<br><br>デフォルト : 有効 | 「ポリシー・ルール」ページで、ルールが有効か無効かを指定します。<br><br>ルールを有効にして他のパラメータを適切に設定すると、predicate パラメータに指定した、構成済の証明書の有効期間が Oracle Application Server Certificate Authority によって検証されます。<br><br>ルールを無効にすると、Oracle Application Server Certificate Authority が、構成済の証明書の有効期間を検証する際に、ルールに指定した有効期間は使用されません。かわりに、リクエストに指定した有効期間が使用されます。 |
| Minimum Validity<br><br>デフォルトの最小期間 : 90 日 | 証明書の最小有効期間（日数）を指定します。<br><br>有効値 : 0（ゼロ）より大きく、Maximum Validity パラメータで指定した値より小さい整数。 |
| Maximum Validity<br><br>デフォルトの最大期間 : 3650 日 | 証明書の最大有効期間（日数）を指定します。<br><br>有効値 : 0（ゼロ）より大きく、Minimum Validity パラメータで指定した値より大きい整数。<br><br>デフォルトの有効期間は Default Maximum: 3650 日。 |
| validityPeriod<br><br>デフォルト : 365 日 | SSO ユーザーおよび SSL ユーザーの有効期間を指定します。最小有効期間と最大有効期間の間の値を指定する必要があります。<br><br>値は 365 日に設定されています。 |
| predicate | このルールの条件式を指定して、ルールを適用する証明書のタイプを制限します。ルールをすべての証明書要求に適用する場合は、フィールドに「*」を入力します。<br><br>例 : Type=="client"<br>Type=="*"<br><br>「ポリシー・ルールの条件」の項を参照してください。 |

このルールを無効にすると、Oracle Application Server Certificate Authority によって、証明書要求に指定した有効期間の証明書が発行されます。その期間は、CA の証明書の有効期間以内に設定されます。

自動認証のクライアント・ユーザー（SSO 認証ユーザーや SSL 認証ユーザーなど）の場合、有効期間は、ポリシーに指定した条件と一致するデフォルトの有効期間を使用して、自動的に設定されます。その他のユーザーの場合、有効期間は、証明書要求に指定したとおりになります。この機能によって、自動認証ユーザーが取得する有効期間は管理者が正確に指定できるため、これらのユーザーがこの値を入力する必要はなくなります。

認証局には、5 ～ 10 年の有効期間を適用できます。CA に対する有効期間を長めに設定すると、更新や置換の必要がなく、発行した証明書は長期にわたって有効となります。Oracle Application Server Certificate Authority のインストール処理では、ルート CA に対してデフォルト値である 10 年が使用されます。次の図に、ValidityRule パラメータを示します。




## UniqueCertificateConstraint

UniqueCertificateConstraint ポリシー・ルールは、OCA が、同じ使用方法で、同じサブジェクト名に複数の証明書を発行することを禁止します。有効に設定すると、ポリシーのパラメータがこのような複数の証明書を禁止するように設定されている場合に、この要求を拒否することができます。

ポリシーは、受信した証明書要求と同じサブジェクト DN を持つ証明書が Oracle Application Server Certificate Authority リポジトリにないか確認します。サブジェクト DN が同じ証明書が見つかった場合は、次に、証明書の使用方法（SSL、署名など）を確認します。要求している DN を持つ証明書が存在し、同じ使用方法が指定されている場合は、ポリシーに複数の証明書を拒否するように設定していると、その要求は拒否されます。

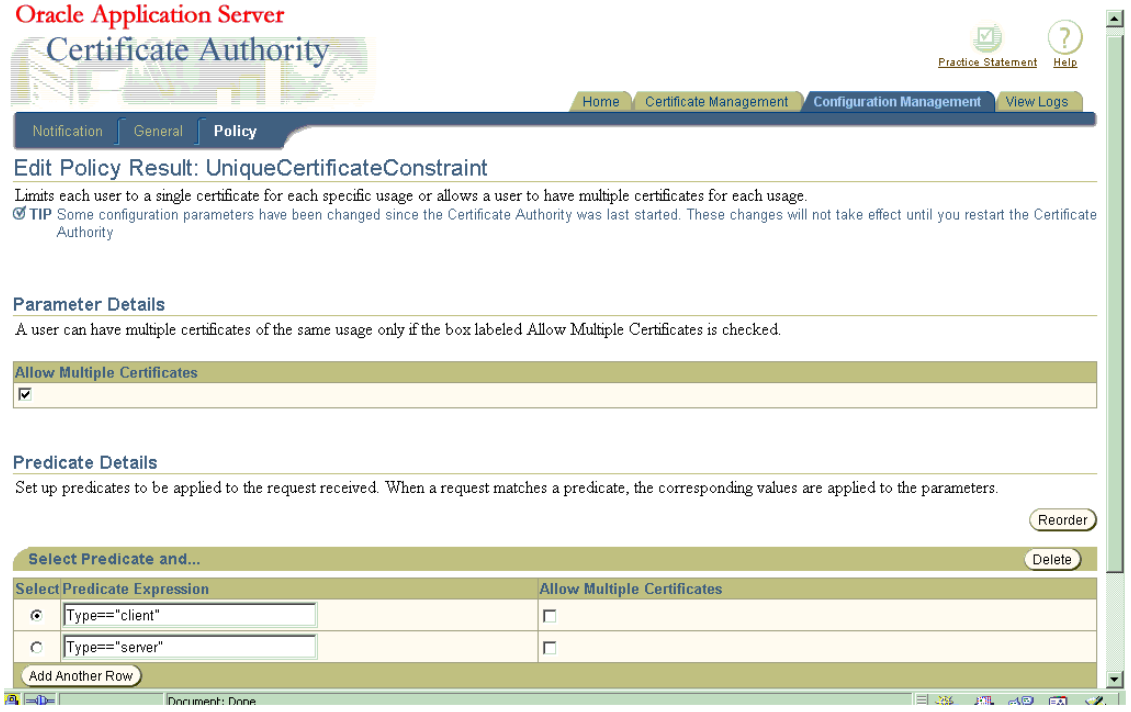



表 5-5 に、UniqueCertificateConstraint モジュールのパラメータを示します。

**表 5-5 UniqueCertificateConstraint ポリシー・ルールのパラメータ**

| パラメータ | 説明 |
|---|---|
| Status（有効または無効）<br><br>デフォルト：有効 | 「ポリシー・ルール」ページで、ルールが有効か無効かを指定します。<br><br>ルールが有効な場合は、使用方法が同じ複数の証明書を許可するチェック・ボックスを使用します。サブジェクト名および使用方法が同じ複数の証明書を禁止する場合は、要求は拒否されます。<br><br>ルールを無効にすると、OCA は、サブジェクト名および使用方法が同じ複数の証明書要求を許可します。 |

**表 5-5 UniqueCertificateConstraint ポリシー・ルールのパラメータ（続き）**

| パラメータ | 説明 |
|---|---|
| 同じ DN と同じ使用方法を持つ複数の証明書を許可するチェック・ボックス | 選択すると、OCA は、使用方法が同じでも、すでに証明書を持っている DN にも新しい証明書を発行できます。<br>選択を解除すると、OCA は、新旧の証明書の使用方法が同じ場合に、すでに証明書を持っている DN に新しい証明書を発行できません。 |
| デフォルト：選択 | |

## RevocationConstraints

OCA 管理者は、このポリシーをユーザーからの証明書失効要求に適用することにより、期限切れの証明書の失効を制限できます。このポリシーが有効な場合は、満了後でも期限切れの証明書を失効させることができます。PKI の設定で期限切れの証明書の失効を許可しない場合は、このポリシーを使用して、期限切れの証明書を失効させないように Oracle Application Server Certificate Authority を構成できます。

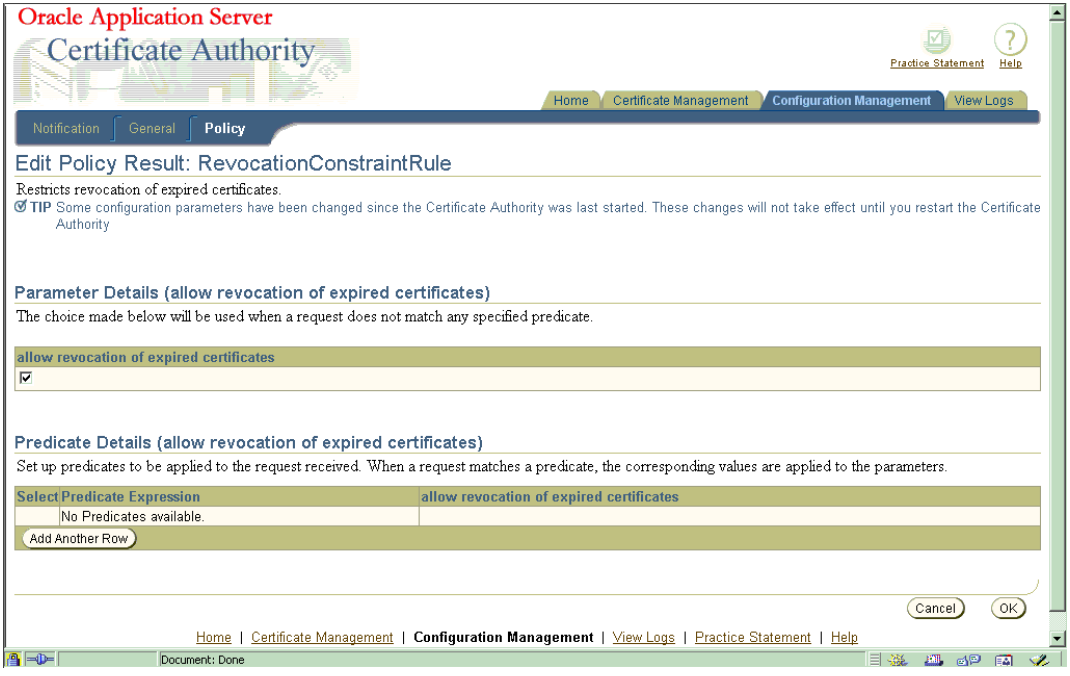


表 5-6 に、失効制約モジュールのパラメータを示します。

**表 5-6 RevocationConstraints ポリシー・ルールのパラメータ**

| パラメータ | 説明 |
|---|---|
| Status（有効または無効）<br><br>デフォルト : 有効 | 「ポリシー・ルール」ページで、ルールが有効か無効かを指定します。<br><br>ルールを有効にして他のパラメータを適切に設定すると、Oracle Application Server Certificate Authority は、失効させる証明書の有効期間、および allowExpiredCerts パラメータに割り当てられる値を確認し、それに従って失効要求を許可または拒否します。<br><br>ルールを無効にすると、OCA は、失効させる証明書の有効期間および期限が切れているかどうかの確認は行いません。証明書は失効されるだけです。 |
| allowExpiredCerts<br><br>デフォルト : TRUE | 期限切れの証明書の失効を許可するか（TRUE）、拒否するか（FALSE）を指定します。 |

## RenewalRequestConstraint

OCA 管理者は、このポリシーを証明書の更新要求に適用することによって、証明書の更新（管理者の証明書の更新も含む）が可能な時間枠を制限できます。このポリシーが有効な場合、ユーザーは、満了日の前後に指定した日数以外では証明書を更新できません。このポリシーを構成することによって、PKI の設定で期限切れの証明書の更新を除外または制約することができます。

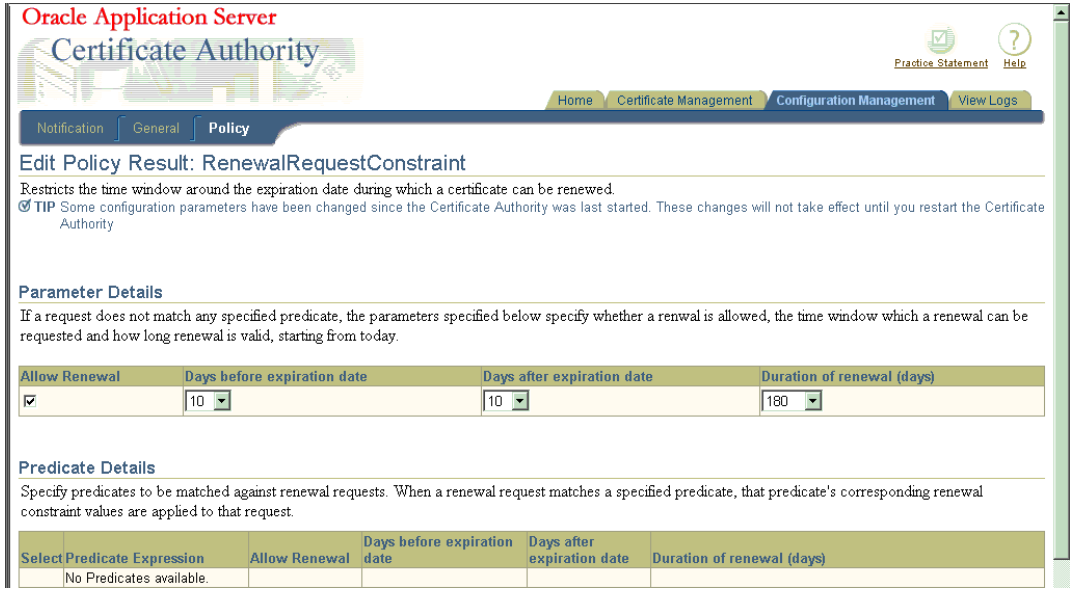
表 5-7 に、更新を制約するポリシー・ルールのパラメータを示します。

**表 5-7 RenewalConstraints ポリシー・ルールのパラメータ**

| パラメータ | 説明 |
|---|---|
| Status（有効または無効）<br><br>デフォルト : 有効 | 「ポリシー・ルール」ページで、ルールが有効か無効かを指定します。<br><br>ルールを有効にして他のパラメータを適切に設定すると、Oracle Application Server Certificate Authority は renewalNotBefore パラメータおよび renewalNotAfter パラメータを確認し、満了日の前後に指定した日数内に要求が行われたかどうかを検証します。確認が正常に終了すると、有効期間が validityPeriod パラメータに指定した値に設定されます。<br><br>ルールを無効にすると、OCA によって、証明書の更新が要求された日付を確認せずに、そのまま更新し、有効期間を 365 日に設定します。 |

**表 5-7 RenewalConstraints ポリシー・ルールのパラメータ（続き）**

| パラメータ | 説明 |
|---|---|
| predicate<br><br>（デフォルトはなし） | このルールの条件式を指定します。ルールをすべての証明書要求に適用する場合は、このフィールドに「*」を入力します（デフォルト）。自動認証ユーザーの場合、クライアントのタイプは常に「ocmcert」であるため、「DN=="ou=ST,o=Oracle,c=US"」のような、タイプの条件式は必須ではありません（DN エントリは、連続して指定する必要があります。「C=」エントリまで完全に指定する必要がありますが、「CN」で始める必要はありません）。<br><br>「ポリシー・ルールの条件」の項を参照してください。 |
| allowRenewal<br><br>デフォルト : TRUE | 証明書の更新を許可するか（TRUE）、拒否するか（FALSE）を指定します。 |
| renewalNotBefore<br><br>デフォルト : 10 | 満了日の何日前まで証明書の更新が可能かを指定します。<br><br>有効な値は、10、15、20、25 または 30 です。 |
| renewalNotAfter<br><br>デフォルト : 10 | 満了日の何日後まで証明書の更新が可能かを指定します。<br><br>有効な値は、10、15、20、25 または 30 です。 |
| validityPeriod<br><br>デフォルト : 365 日 | 証明書の更新有効期間（日数）を指定します。有効値 : 数値（期間は無制限） |




Oracle Application Server Certificate Authority のすべてのポリシーは、OCA 管理者が、Web ベースの管理インタフェースの「ポリシー」サブタブを使用して管理します。

# Oracle Application Server Certificate Authority の「ポリシー」サブタブ

「ポリシー」サブタブを最初に選択すると、証明書要求に適用可能なすべてのポリシー・ルールが Oracle Application Server Certificate Authority に表示されます。


「ポリシーの表示」のラベルが付いたドロップダウン・ボックスから「失効」と「更新」のどちらかを選択することによって、表示するポリシー・ルールを、失効に適用するものと更新に適用するものに切り替えることができます。その後、Oracle Application Server Certificate Authority に、これらのポリシーが表示されます。次の項で、Oracle Application Server Certificate Authority に付属のポリシー、および管理者が使用できるアクションについて説明します。

- 製品に付属の証明書要求ポリシー
- 製品に付属の証明書失効ポリシー
- 製品に付属の証明書更新ポリシー
- ポリシー操作

ポリシーは、証明書要求の評価に使用するルール、および発行された証明書の更新または失効に使用するルールを指定します。要求、失効または更新のポリシーを追加することができます。また、複数のポリシーが存在する場合は、ポリシーを並び替えて、適用順序を変更することもできます。指定したタイプの各ポリシーで、パラメータと条件を参照および編集したり、そのポリシーを有効化または無効化することができます。OCA のデフォルト・ポリシーは削除できませんが、カスタム・ポリシーは削除できます。

ポリシーを追加するには、名前と説明を指定し、さらに `$ORACLE_HOME/oca/policy` ディレクトリ（Windows の場合は、`$ORACLE_HOME¥oca¥policy`）に jar として事前に追加したクラスを指定する必要があります。

**関連項目：**「カスタム・ポリシー・プラグインの開発」

管理者は、すべてのポリシーを無効にすることができます。ポリシーを無効にしても削除されないため、今後使用できなくなるわけではありません。ただし、OCA リポジトリのエントリは、後で再度有効にできるためリセットされます。ポリシーを削除すると、別のポリシーとして追加しないかぎり、永続的に使用できません。

ポリシーは、OCA リポジトリのエントリによって有効になります。無効なポリシー（または OCA リポジトリには指定されていたが有効ではないポリシー）を有効にすると、そのポリシーのパラメータおよび条件が再び有効になります。

通常、ポリシーのパラメータは、デフォルトの制限または範囲を指定します。証明書要求は、この制限または範囲に従う必要があり、違反すると自動的に拒否されます。機能や制約を有効または無効にするだけのパラメータもあります。パラメータは、条件に指定された場合を除き、すべての場合に適用されます。

ポリシーの条件は、ポリシーのパラメータの制限、範囲または制約が、他のすべての証明書または要求のデフォルトとは異なるように指定されている、特定の証明書または要求のタイプを識別します。

「ポリシー」構成パラメータの変更を有効にするには、Oracle Application Server Certificate Authority を再起動する必要があります。手順は第 3 章の「Oracle Application Server Certificate Authority の起動および停止」を参照してください。

次の項で説明するとおり、Oracle Application Server Certificate Authority には、証明書の要求、失効および更新に適用するポリシーが用意されています。

証明書を発行するときに「ポリシーを適用」チェック・ボックスの選択を解除すると、管理者は、ポリシーを上書きすることができます。

## 製品に付属の証明書要求ポリシー

証明書要求は、セキュリティ上重要な 4 つの要因を制限するポリシーのパラメータおよび条件を満たしている必要があります。製品に付属のポリシーで、次の問題に関係するパラメータおよび条件を調整できます。

- 鍵サイズの範囲の調整および RSA の公開鍵 / 秘密鍵のデフォルトの設定
- 有効期間の範囲の調整およびデフォルトの設定
- ユーザーが、使用方法（SSL、CA または SMIME を個別に設計した署名、鍵の暗号化またはデータの暗号化）ごとに、複数の証明書を持つことに対する許可または禁止
- 信頼できる証明書の DN を、証明書の申請者または所有者として使用することに対する許可または禁止

## 製品に付属の証明書失効ポリシー

RevocationConstraintRule は、Oracle Application Server Certificate Authority に付属する OCA のデフォルト・ポリシーです。期限切れの証明書の失効を許可または禁止するなど、このポリシーのパラメータおよび条件は、必要に応じて設定できます。

## 製品に付属の証明書更新ポリシー

RenewalRequestConstraint ポリシーのパラメータおよびデフォルトを設定できます。このポリシーは、証明書の更新可否、更新するタイミングおよびその期間を設定します。設定された満了日前後の日数を設定して、更新が許可されている範囲内で時間枠を指定します。デフォルトは、満了日の前後 10 日です。デフォルトの更新期間は 365 日ですが、変更もできます。

## ポリシー操作

「ポリシー・ルール」画面には、実行可能な操作のボタンが表示されます。それぞれのボタンについては、「編集」、「有効化または無効化」、「削除」、「ポリシーの並び替え」、および「ポリシーの追加」の各項で説明します。

## 編集

ポリシーを選択して「編集」をクリックすると、ポリシーの画面が表示され、現在設定されているパラメータおよび条件が示されます。たとえば、鍵の制約ポリシーの画面では、鍵の最大サイズおよび最小サイズのデフォルトが表示されます。また、特定の証明書タイプでこれらのデフォルトを変更する条件も示されます。

どのページにおいても、デフォルトのパラメータまたは既存の条件に関連付けられた特定の値とは異なる値を選択できます。標準的なポリシーでは、「述語式」テキスト・ボックスに入力してこれらの条件を変更したり、「並び替え」ボタンを使用して条件の順序を変更したり、「追加」ボタンを使用して条件を追加することもできます（「条件の並び替え」および「条件の追加」を参照）。

カスタム・ポリシーを選択して「編集」をクリックすると、「カスタム・ポリシー」編集画面が表示されます。通常の編集画面には、デフォルト・ポリシーしか表示されません。

## 有効化または無効化

ポリシーを作成すると、そのポリシーを有効にできます。これによって、指定した操作（要求、失効または更新）にポリシーが適用されます。有効にしない場合、またはポリシーを作成した際に無効にした場合、ポリシーのパラメータ、デフォルトおよび条件は、どの操作（要求、失効または更新）にも適用されません。

ただし、ポリシーを無効にしても、データベースには使用可能な状態で残ります。後でこのポリシーを有効にすることもできます。

これに対して、ポリシーを削除すると、データベースからも削除されます。まったく別のポリシーとして再度入力しないかぎり、永続的に使用できなくなります。

## 削除

「ポリシー・ルール」ページでは、Oracle Application Server Certificate Authority のデフォルト・ポリシーを削除できません。削除できるのはカスタム・ポリシーだけです。追加したカスタム・ポリシーがリストに表示されるので、ポリシーを選択して「削除」をクリックできます。

特定のルールに対する「編集」ページで、条件を選択して「削除」をクリックできます。すぐにその条件が削除され、削除したことを示す情報メッセージが表示されます。

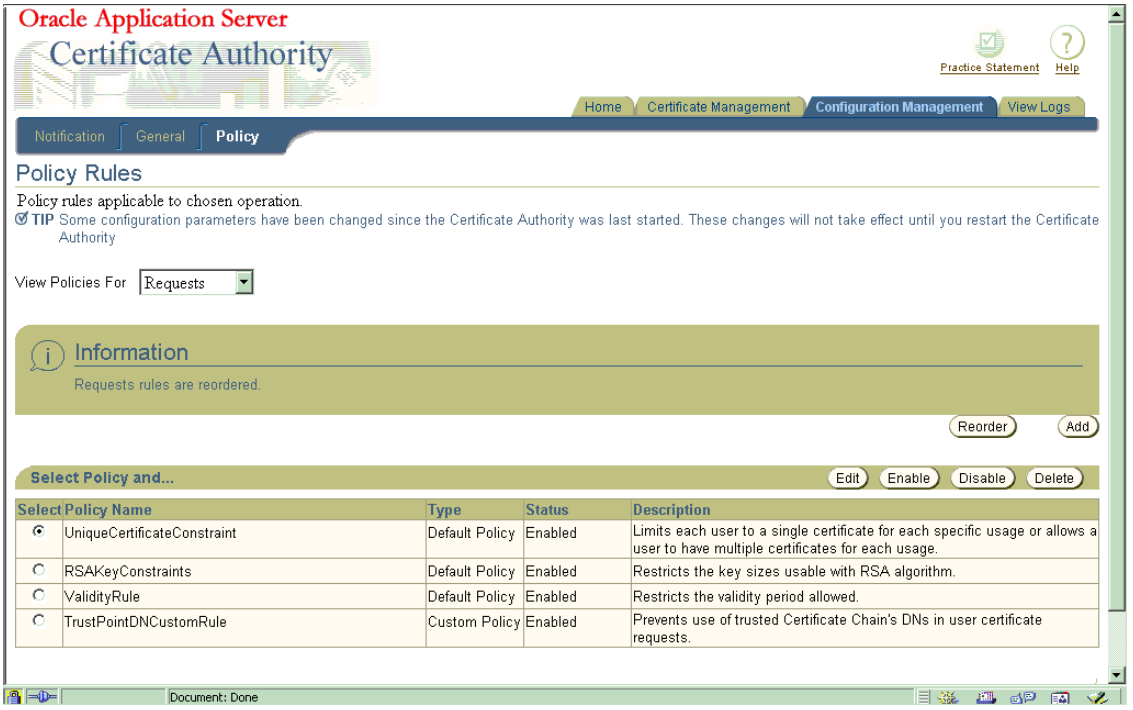
## ポリシーの並び替え

管理者は、ポリシーの適用順序を変更できます。たとえば、証明書要求のデフォルト・ポリシーが、次に示す順序で表示されているとします。


「並び替え」をクリックすると、既存のポリシーのリストが表示されます。次の画面で、ポリシーを選択して並び替え、目的の順序になるように変更します。


UniqueCertificateConstraint ポリシーを 2 つ上に移動するには、このポリシーをクリックして選択し、上矢印のボタンを 2 回クリックします。そうすると、次のような画面になります。


「OK」をクリックすると、次の画面に示すとおり、そのポリシーは、以前の場所ではなく一番上に表示されます。




Oracle Application Server Certificate Authority から、変更に対するアラートを示す情報メッセージが表示されます。

ポリシー・ルールに含まれる条件も、同じ方法で並び替えることができます。「条件の並び替え」の項を参照してください。

## ポリシーの追加

「ポリシー・ルール」では、「追加」ボタンをクリックして、操作（要求、失効または更新）に新しいポリシーを追加できます。$ORACLE_HOME¥oca¥policy ディレクトリに jar として定義し、使用可能にしていたオブジェクト・クラスに具体化されているカスタム・ポリシーのみ追加できます。新しいポリシーの名前、説明およびオブジェクト・クラスを入力し、有効にするかどうかを指定するフォームが表示されます。カスタム・ポリシーの開発の詳細は、「カスタム・ポリシー・プラグインの開発」の項を参照してください。


また、ポリシー・ルールに含まれる条件を、そのポリシーの「編集」ページに表示される任意のデフォルト・ポリシーに追加できます（カスタム・ポリシーには条件を追加できません）。「条件の追加」の項を参照してください。

# ポリシー・ルールの条件

「ポリシー管理の概要」の項に説明するように、ポリシー・ルールは、特定の規則に従って指定および施行されます。この項では、ポリシー・ルールの条件の使用について説明します。また、次の項で、例を示します。

- 複数の条件による評価
  - 複数の条件による評価の例
  - 複数の条件による評価の例 2
  - 条件の並び替え
  - 条件の追加

---

**注意：** ポリシー・ルールは、複数のタイプの要求（証明書の発行、失効または更新の要求）で共有することはできません。

---

条件には、受信した証明書要求の検証に使用する特定の値および式を指定します。指定した値は、条件式が証明書要求の対応する要素と一致する場合に、ポリシーのデフォルトのかわりに使用されます。一致する場合は、その条件式に関連付けられた値を使用して、要求の妥当性が評価され、ポリシーのデフォルト値のかわりにパラメータが設定されます。

条件の指定は任意です。また、条件はカスタム・ポリシーには使用できません。

デフォルト・ポリシーに含まれるルールに対しては、Web ベースのインタフェースから条件を指定できます。条件の指定後は、ポリシーを適用する特定の証明書操作（要求、失効、更新など）のすべての受信要求に対して、指定した条件が照合されます。

受信した証明書または証明書要求がどの条件式とも一致しない場合、またはルールに条件がない場合は、ポリシーに指定されたデフォルトの値、範囲またはアクションを使用して、要求が評価されます。たとえば、要求の値は、ポリシーに指定された適切なデフォルトの範囲内にあるかどうかが検証されます。範囲内の場合、要求は許可されます。値が、指定されたデフォルトと一致しない場合、または指定された範囲内にない場合は、内容を記したエラー・メッセージが表示され、要求が拒否されます。

受信した証明書または証明書要求が、条件で指定したタイプと一致した場合、ルールのデフォルトまたは範囲は、その証明書または証明書要求に適用されません。適用できるのは、その条件に対応するものとして指定した値だけです。

このように、管理者は、デフォルト・ポリシーのルールを拡張し、様々なユーザー用に構成することができます。たとえば、Sales 部門に設定した有効期間より長い有効期間を、Development 部門に設定することができます。

条件式は論理式です。変数および関係演算子を使用して式を作成します。たとえば、条件を設定し、様々なグループのユーザーの証明書に対して、異なる有効期間を設定することができます。

次に、有効な条件式の例を示します。

```
Type==client AND DN=="ou=Sales,o=oracle,c=us"
Type==server AND DN=="o=Oracle,c=us"
```

表 5-8 に、条件式に使用する論理演算子を示します。

**表 5-8 論理演算子**

| 演算子 | 説明 |
|---|---|
| == | 等しい |
| != | 等しくない |
| AND | 論理演算子 AND |

次のルールでは、デリミタ「:=」を使用して、ポリシーの式の名前とその有効な構文を区切ります。ポリシーの式を構成する際に有効な構文を示します。

```
Predicate expression := Expression | AndExpression

AndExpression := Expression AND Expression

Expression := Attribute op Value

Attribute :=  <attrib_name>

op:    ==  or    !=

Value :=  a string
```

Oracle Application Server Certificate Authority では、OR、<、> などの演算子はサポートしていません。条件を複数の条件に分割して同じ値を指定することにより、OR 論理式を実装できます（ポリシー・プラグインおよび API は、複数の条件をサポートします）。条件では、二重引用符で囲んだ文字列を値に指定できます。属性は、常に <attrib_name> として指定します。すべての条件式および文字列の値では、大文字と小文字が区別されません。式の値に「"*"」を設定して、対象となるすべての属性と照合することができます。たとえば、「type=="*"」と指定すると、すべての証明書タイプが一致します。ただし、「"*"」を他の文字列とともに使用した文字列の部分一致はサポートされていません。

表 5-9 に、属性および指定可能な値を示します。

**表 5-9 条件の属性**

| 属性 | 変数名 | 説明 |
|---|---|---|
| type | type | 証明書タイプを指定します。指定可能な値は、次のとおりです。<br>■ type=="client"<br>■ type=="server"<br>■ type=="ca " |
| usage | usage | 証明書の使用方法のタイプを指定します。指定可能な値は、次のとおりです。<br>■ usage=="ssl"<br>■ usage=="smime_enc"<br>■ usage=="smime_sign"<br>■ usage=="code_sign"<br>■ usage=="ca_sign" |

**表 5-9 条件の属性（続き）**

| 属性 | 変数名 | 説明 |
|---|---|---|
| DN | DN | 識別名を指定します。有効なパラメータは、有効な部分 DN または完全 DN です（DN エントリは、連続して指定する必要があります。「C=」エントリまで完全に指定する必要がありますが、「CN」で始める必要はありません）。 |

RFC1779 に指定されるように、Oracle Application Server Certificate Authority では、最大の構成要素を末尾に指定した DN を使用します。たとえば、適切な書式で記述された DN「cn=user31415,ou=security,ou=ST,o=Oracle,c=US」では、「cn」は最小の構成要素で、「c」は最大の構成要素です。

RDN とは、相対識別名（Relative Distinguished Name）の略語で、エントリを一意に指定するうえでこれ以上の修飾を必要としない、最も詳細なローカル・エントリ名を意味します。RDN が複数回出現する場合は、最初に指定された最小の RDN が、次に出現する RDN の子として判断されます。前述の例では、「ou=security」が「ou=ST」の前に出現するため、「security」は「ST」に従属する単位と判断されます。

条件で指定する DN は、どの RDN からでも始めることができますが、ルートまで完全に指定する必要があります。たとえば、「ou=ST,o=Oracle,c=US」は場所を指定する有効な部分 DN ですが、「ou=ST,o=Oracle」は「o=Oracle」で終わり、ルート（たとえば「c=US」）が含まれていないため、無効な部分 DN です。

ビッグ・エンディアン（最大の構成要素を最初に指定）の順序をサポートするために、OCA は、ポリシーの評価のためだけに、DN を照合する前に内部的にリトル・エンディアンの順序に変換します。

DN の構成要素を、条件式に指定した DN の式と照合する場合、次のルールが適用されます。

条件は、条件全体が DN の最後の部分と一致するかどうかを照合します。

たとえば、次の条件式があるとします。

DN=="ou=ST,o=Oracle,c=US"

この条件式は、次のすべての DN と一致します。

"cn=user31415,ou=ST,o=Oracle,c=US"

"cn=quser2787,ou=security,ou=ST,o=Oracle,c=US"

"cn=kuser987,ou=security, ou=DAS,ou=ST,o=Oracle,c=US"

前述の条件式は、次の DN とは一致しません。

"cn=user31415,ou=DAS,ou=ST,o=Oracle,c=IN"

"cn=quser2787,ou=ST, ou=pki, o=Oracle,c=US"

"cn=kuser987,ou=ST,o=Oracle, st=CA,c=US"

## 複数の条件による評価

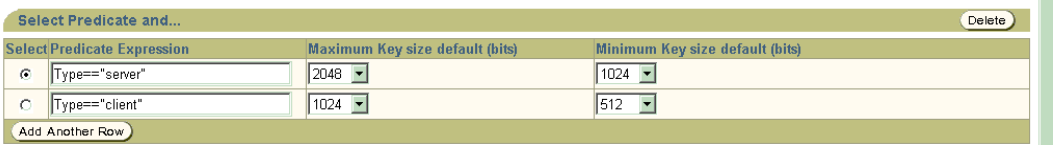
ポリシー・ルールには、複数の条件を指定できます。ポリシー・ルールに複数の条件が指定されている場合は、最初の条件式と受信した証明書要求のオブジェクトの比較から、評価が開始されます。一致する場合は、ルールが適用されます。一致しない場合は、次の条件式と要求を比較して評価されます。この処理は、条件が証明書要求オブジェクトと一致するまで、または照合する条件がなくなるまで続行され、照合する条件がない場合は、ポリシー・ルールのデフォルト値が適用されます。

最適な一致の検出は試行されず、最初に一致したものが使用されます。管理者は、組織に最適な順序で条件を指定する必要があります。

ある基準をルールの最上位に指定し、その条件式が特定の照合および最小の RDN を対象とする場合は、その条件式が最初に評価されます。

### 複数の条件による評価の例

次の例では、ルールによって、複数の条件がどのように評価されるかを示します。この例は、RSA ルールが使用する鍵のサイズのポリシー・ルールです。ルールには、サーバー証明書およびクライアント証明書についての 2 つの条件式が含まれます。条件式には、対応する最小および最大の鍵のサイズが指定されています。受信したサーバー証明書要求またはクライアント証明書要求で指定された鍵のサイズが、対応する条件で指定した範囲外の場合、その要求はルールによって拒否されます。




どちらの条件式も、受信した証明書要求と一致しない場合は、要求された鍵のサイズが、デフォルトとして指定されている最小および最大の鍵のサイズと比較されます。要求された鍵のサイズが範囲外の場合は、要求は拒否されます。範囲内の場合は承認されます。

## 複数の条件による評価の例 2

複数の条件を使用すると、微細な評価ができます。条件は、Oracle Application Server Certificate Authority の Web ベースのインタフェースの「構成管理」タブにある「編集」ページのリストに従って、上から順に適用されます。順序は重要です。

ポリシーの最初の条件に「`Type=="client"`」、「`OU=="Oracle"`」および「`CN=="Clay"`」と指定され、鍵の長さが 2048 に設定されているとします。

次に、同じポリシーの後続の条件に「`Type=="client"`」および「`OU=="Oracle"`」と指定され、鍵の長さが 512 に設定されているとします。

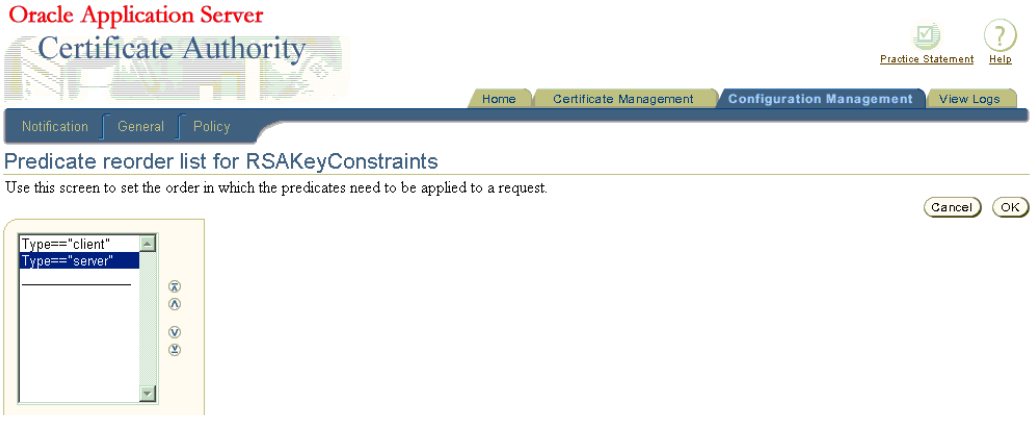
この場合、鍵の長さが 2048 に設定されるのは、Clay からのクライアント要求だけで、他のすべての Oracle クライアント要求の鍵の長さは 512 に設定されます。

ただし、順序が逆の場合はポリシー内で一般的な条件が先になるので、Clay の鍵の長さも 512 に設定されます。順序の中で 1 つ上の条件（より一般的な条件）のみがこのポリシーの対象となるため、より限定的な条件は適用されません。

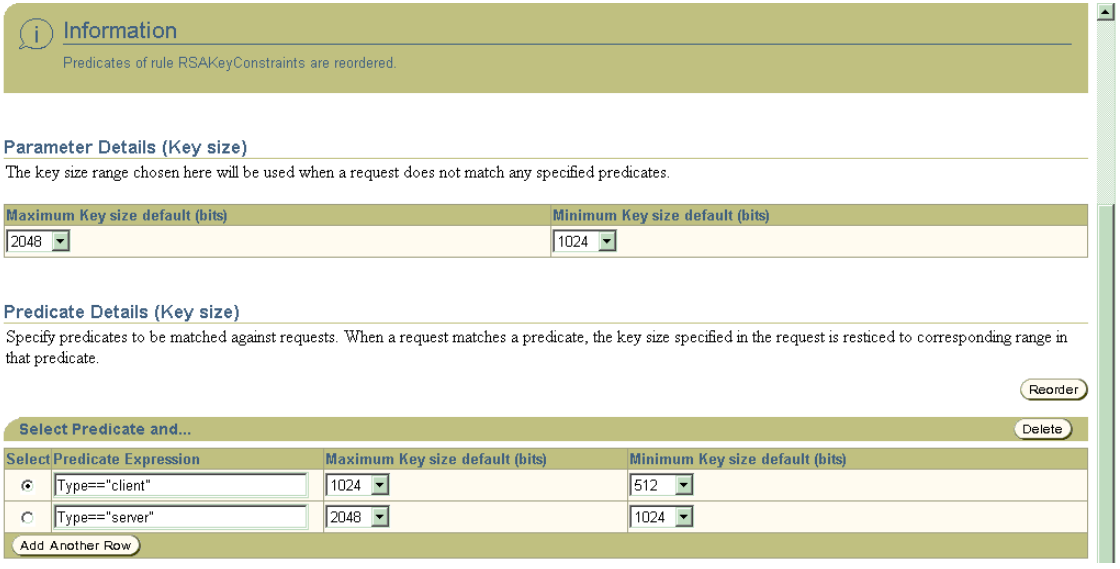
## 条件の並び替え

条件の並び替えは、ポリシーの並び替えと同じ方法で変更できます（5-18 ページの「ポリシーの並び替え」の項を参照）。「複数の条件による評価の例」の項に登場するページのように、条件を表示するページで「並び替え」をクリックすると、次のような画面が表示されます。


移動する条件をクリックして選択し、いずれかの矢印ボタンをクリックします。条件は、選択した方向に移動します。たとえば、「複数の条件による評価の例」の条件の順序を逆にすると、次のような画面が表示されます。



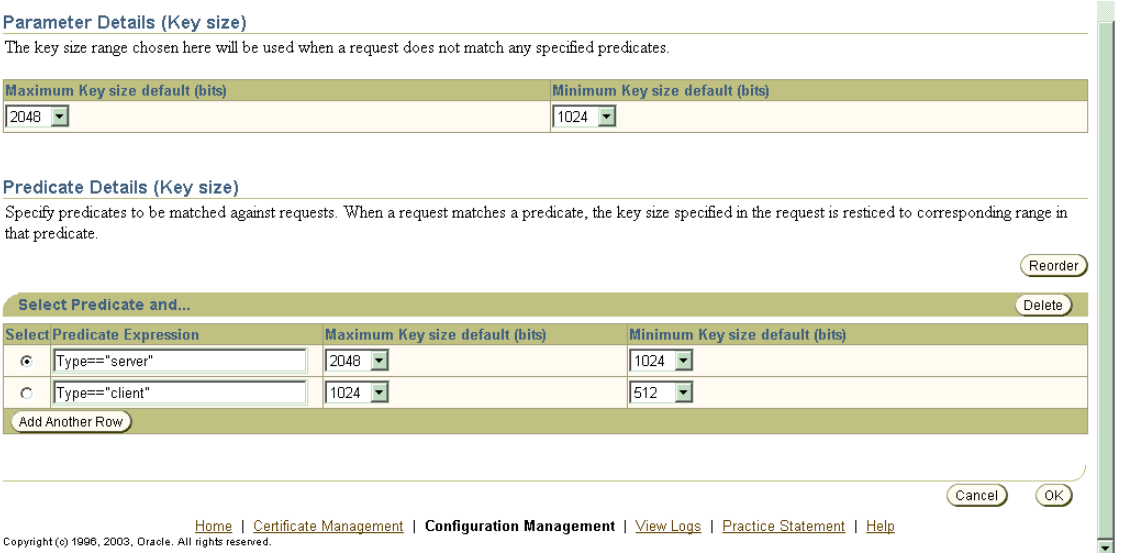

「OK」をクリックすると、次に示すとおり、ルールに対する条件の順序が決定します。



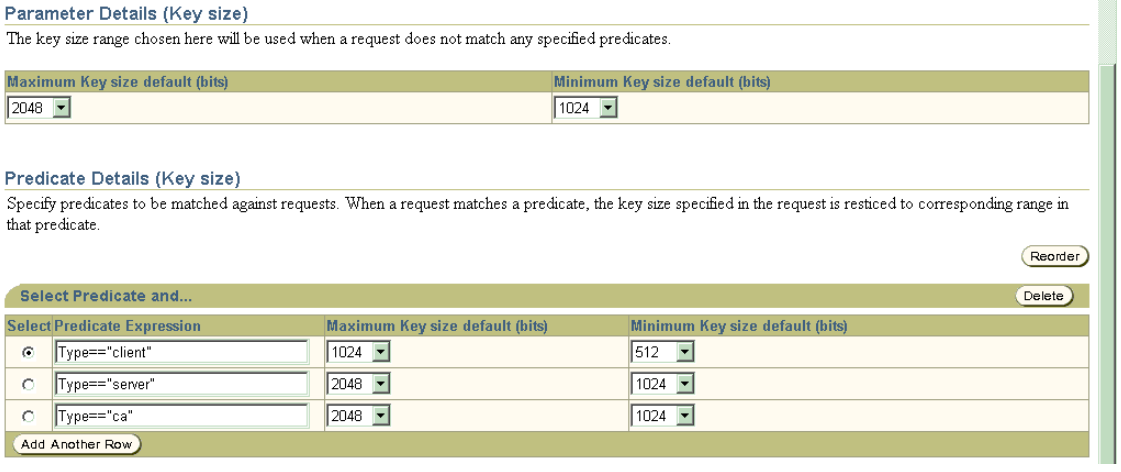

Oracle Application Server Certificate Authority から、順序が変更されたことを知らせる情報メッセージが表示されます。

## 条件の追加

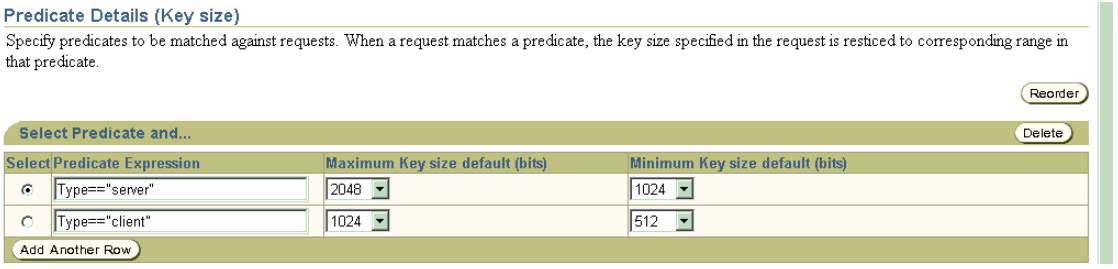
条件を表示するページで「1 行追加」をクリックすると、条件を追加できます。空の入力行が表示されます。



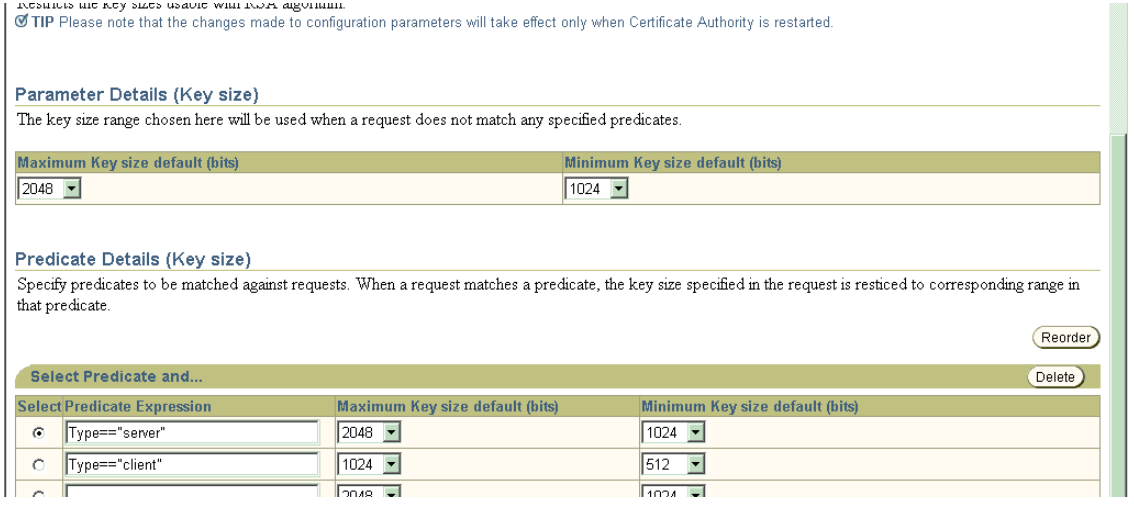

次に示すように、空の行に有効な式を入力して「OK」を押すと、その式が受け入れられ、ポリシーのメイン・ページに戻ります。



特定のポリシーの「編集」ページで「1 行追加」をクリックすると、条件を追加できます。次の画面に示す条件の例では、サーバー証明書の要求には、エンド・ユーザーの証明書要求よりも、大きいサイズの鍵を使用する必要があります。




「1 行追加」をクリックすると、条件を指定する空の行が追加表示されます。この行の「述語式」ボックスに新しい条件を入力できます。条件が一致するときに使用する機能またはデフォルトのパラメータ範囲も指定します。




無効な条件またはこのルールにすでに存在するルールを指定すると、エラー・メッセージが表示されます。

必要な条件の指定が完了したら、「OK」をクリックします。このルールの元のページが、新しい条件の式が一番下に追加された状態で表示されます。

# カスタム・ポリシー・プラグインの開発

OCA に付属のデフォルトのポリシー・プラグインは汎用のプラグインです。組織の具体的な要件に合せてポリシーの構造を拡張するために、管理者は、OCA が提供するフレームワークを使用してカスタム・プラグインを作成できます。このフレームワークには、証明書と証明書要求の情報を取得するための API、およびいくつかの汎用的な機能が含まれます。カスタム・プラグインを実装するために、管理者は、Java クラスを作成し、OCA に登録する必要があります。これを「ポリシーの追加」と呼びます。

次のような状況に対処する場合は、カスタム・プラグインを開発することをお薦めします。

- 企業の追加アカウント・リポジトリを使用して、ユーザーの要求を検証する場合
- 他のユーザー・リポジトリに基づいて追加のフィールドを設定する場合

OCA が提供する API を使用すると、管理者のカスタム・プラグインで、要求のパラメータ、および証明書と証明書要求の属性を取得できます。

> **関連項目：** Oracle Application Server Certificate Authority に付属の Javadoc

次の項では、管理者がカスタム・プラグインを開発する際に役立つツールおよび例について説明します。


- ポリシーにより実行される処理について
- 新しいポリシー・プラグインを作成する手順
- カスタム・ポリシー・プラグインの例
- 汎用エラー・メッセージ


## ポリシーにより実行される処理について

カスタム・プラグインは、OCACustomPolicyPlugin インタフェースを実装することで記述できます。このインタフェースの enforce メソッドに渡される OCAPolicyRequest オブジェクトには、（証明書および証明書要求の）重要な属性とその値セットがすべて含まれています。カスタム・プラグインは、このオブジェクトを読み取り、証明書要求または証明書の属性を取得または設定できます。

カスタム・ポリシー・プラグインでは、次の手順で処理が実行されます。

**表 5-10 カスタム・ポリシー・プラグインの処理手順**

| 手順 | 結果 |
|---|---|
| OCA カスタム・プラグインの enforce メソッドが、ポリシー・プロセッサから OCAPolicyRequest を受け取る。 | 登録、更新または失効の要求で設定された実際のパラメータ値の取得に必要なオブジェクトが、自動的に取得されます。これらのパラメータには、DN、有効期間、シリアル番号などがあります。 |
| OCAPolicyRequest から取得したパラメータと、プラグインの想定するパラメータ値をプラグインが検証する。 | ポリシーが正常に検証された場合は、setPluginResult メソッドを使用してプラグインの結果が設定され、ポリシー・プロセッサに TRUE が返されます。正常に検証できなかった場合は、setError() を使用してエラーが設定され、ポリシー・プロセッサに FALSE が返されます。 |

## 新しいポリシー・プラグインを作成する手順

新しいポリシー・プラグインを作成するには、次の 4 つの手順を実行します。

1. 次の項に示すサンプル実装を参考にして、OCACustomPlugin インタフェースを実装する Java クラスを記述します。
2. 手順 1 で実装した Java クラスを保存します。次に、Java の CLASSPATH に $ORACLE_HOME/oca/lib/oca-1_3.jar を追加して、クラス・ファイルを取得してから、保存したクラスをコンパイルします。
3. jar ユーティリティを使用して、クラス・ファイルから jar ファイルを生成します。
   a. たとえば、前の項のコードから jar ファイルが生成され、example.jar に格納されます。

      クラスから jar ファイルを生成するには、$ORACLE_HOME/jdk/bin ディレクトリにある jar ユーティリティを使用します。

      * example.jar を作成するには、次を実行します。

        ```
        $ORACLE_HOME/jdk/bin/jar cvf example.jar oca
        ```

      * ここで、example.jar は jar ファイルの名前で、oca は custom/policy/plugin/examplePlugin.class を格納するパッケージ・ディレクトリです。

   b. jar tvf example.jar を実行すると、oca/custom/policy/plugin というディレクトリ構造の下に、examplePlugin.class ファイルが置かれます。
4. この jar ファイルを $ORACLE_HOME/oca/policy ディレクトリに配置します（Windows プラットフォームの場合は、次のように、スラッシュが円記号になります。

   $ORACLE_HOME¥oca¥policy¥

   このディレクトリは、Oracle Application Server Certificate Authority によって事前に作成されています。

5. OCA、OCA の OC4J および OHS を停止します。ORACLE_HOME で次のコマンドを使用します。

```
$ORACLE_HOME/oca/bin/ocactl stop
$ORACLE_HOME/opmn/bin/opmnctl stopproc type=oc4j instancename=oca
$ORACLE_HOME/opmn/bin/opmnctl stopproc type=ohs
```

6. OHS、OCA の OC4J および OCA を、この順序で起動します。同種のコマンドを使用します。

```
$ORACLE_HOME/opmn/bin/opmnctl startproc type= oc4j instancename=oca
$ORACLE_HOME/opmn/bin/opmnctl startproc type=ohs
$ORACLE_HOME/oca/bin/ocactl start
```

7. OCA 管理者の Web ベースのインタフェースを使用して、カスタム・ポリシーを追加し、新しいルールを定義します。

   「構成管理」の「ポリシー」サブタブ内にある「ポリシー・ルール」ページに移動して、「追加」ボタンをクリックし、フィールドに必要事項を入力します。カスタム・ポリシーの名前、説明およびクラスを指定します。「有効化」チェック・ボックスを選択してポリシーを有効にし、「OK」をクリックします。


8. 第 3 章「OCA および証明書の管理の概要」の「Oracle Application Server Certificate Authority の起動および停止」の項の説明に従い、OCA を再起動します。新しい jar ファイルが検出および認識され、そのルールが有効になるには、OCA を再起動する必要があります。この手順が完了したら、プラグインの追加により変更された項目に応じて、証明書、更新または失効の要求にカスタム・ポリシーが適用されます。

## カスタム・ポリシー・プラグインの例

カスタム・ポリシー・プラグインを記述するには、OCACustomPolicyplugin インタフェースを実装する必要があります。

独自のポリシー・プラグインを用意するための最初の手順は、新しい Java クラスの作成です。

この項では、証明書要求の国コードが US でないことを確認するカスタム・ポリシー・プラグインの例を紹介します。

```
1:  package oca.custom.policy.plugin;
2:  import oracle.security.oca.exception.OCMException;
3:  import oracle.security.oca.policy.custom.OCACustomPolicyplugin;
4:  import oracle.security.oca.policy.OCAPolicyRequest;
5:  import oracle.security.oca.policy.OCMPolicyConstants;
6:  public class PolicyCustomPlugin implements OCACustomPolicyPlugin
7:  {                          // ends at line 35
8:     public boolean enforce (OCAPolicyRequest policyRequest)
9:     {                       // ends at line 34
10:      // Add the functionality here.
11:     // Assume the plug-in should reject all requests with country code as US.
12:          if (!policyRequest.getCountry().equals("US"))
13:          {
14:               //Plug-in check succeeded.Country ID in request is not US.
15:                //Hence return true.
16:               return true;
17:          }
18:          else
19:          {              // ends at line 33
20:               //Plug-in check failed:Country ID in request is US.Set error and return false.
21:  try
22:  {
```

```
23:
policyRequest.setError("PolicyCustomPlugin",OCMPolicyConstants.POLICY_ERROR,
                                "Country ID cannot be US.");
24:                  //The first parameter is the plug-in name.
25:                 //The second parameter is the status, which is an ERROR.
26:                //The third parameter is the Message to be displayed.
27:   }
28:  catch(OCMException e)
29: {
30:   //enter exception handling here
31: }
32:                   return false;
33:            }
34:     }
35:  }
```

この例で、行 1 は、このカスタム・ポリシー・プラグインが入っているパッケージです。カスタム・ポリシー・プラグインは、頭文字が oracle.security.oca 以外のパッケージに入れることができます。

行 2 ～ 5 は、必要なクラス・ファイルをインポートします。これらのファイルの詳細は、Javadoc API のドキュメントを参照してください。

行 6 は、OCACustomPolicyPlugin インタフェースを実装します。このカスタム・ポリシー・インタフェースは、すべてのカスタム・プラグインで実装する必要があります。OCA が提供するインタフェースは、パッケージ oracle.security.oca.policy.custom に入っていて、$ORACLE_HOME/oca/lib/oca-1_3.jar にあります。

行 8 は、このプラグインの機能を格納するメソッドを実装します。ポリシー・プロセッサがこのプラグインを起動すると、enforce メソッドが起動されます。

行 9 ～ 28 は、このプラグインインの機能を開始します。

行 12 は、国コードが US でないか確認します。policyRequest で使用できるメソッドの詳細は、OCA に付属する API ドキュメントを参照してください。

行 16 は、ポリシー・プロセッサに成功を返します。

行 18 は、要求の国コードが US のときに発生するエラー状態の処理を定めます。

行 23 は、policyRequest にエラー・コードを設定します。このエラー・コードはポリシー・プロセッサにより読み取られ、画面にレンダリングされます。新しい SSO ユーザー証明書を取得し、すぐに証明書の更新を試みると、同じようなエラーが表示されます。更新用のプラグインにより、エラーがスローされます。

行 30 は、例外を処理するコードに置き換えてください。

行 32 は、ポリシー・プロセッサにエラー・ステータスを返し、要求がポリシー・チェックに失敗したため処理されないことを示します。

## 汎用エラー・メッセージ

次に示すのは、汎用エラー・メッセージとそれに関連付けられた定数です。これは、ポリシーの適用中にエラーが検出された場合に設定できます。これらのメッセージは、OCA がサポートする各言語に変換され、次のような 3 つの内容が表されます。

- 有効期間が無効
  - "OCA_POLICY_INVALID_VALIDITY"
- 要求された有効期間が CA 証明書の有効期間を超過
  - "OCA_POLICY_INVALID_VALIDITY_CA"
- 識別名が無効
  - "OCA_POLICY_INVALID_DN"

たとえば、前述のカスタム・ポリシーの例で、行 13 を次のように変更したとします。

```
13:                 policyRequest.setError("examplePlug-in",OCMPolicyConstants.POLICY_
ERROR,
                                    OCAPolicyMessage.OCA_POLICY_INVALID_DN);
```

この場合、「識別名が無効」というエラーが出力に表示されます。

**関連項目：** OCA カスタム・プラグインに用意されているクラスとメソッド、および使用できる定数の説明は、他のドキュメントに付属する Javadoc を参照してください。

---

**注意：** OCA がサポートする汎用エラー・メッセージは、OCA がサポートする各言語に変換されるので、カスタム・プラグインでも使用できます。これらの定数を使用すると、OCA がサポートする言語であれば、エラー・メッセージをレンダリングできます。

これらのメッセージを使用しない場合は、あらゆる有効な java 文字列を使用できます。ただし、これらの java 文字列は他の言語に変換されないので、指定された文字列そのままでレンダリングされます。

---

# 6

# OracleAS Certificate Authority の管理 : 高度なトピック

この章では、Oracle Application Server Certificate Authority の管理機能、高可用性機能およびバックアップとリカバリの手順について、追加のコンテキストおよび詳細を説明します。内容は、次のとおりです。

- OracleAS Certificate Authority の Wallet 操作
- OracleAS Certificate Authority の構成操作
- カスタマイズのサポート
- Oracle Application Server Certificate Authority での OCA アクションのログまたはトレース
- OCA が使用するインフラストラクチャ・サービスの変更
- OracleAS Certificate Authority および高可用性機能
- OracleAS Certificate Authority のバックアップおよびリカバリでの考慮事項
- 証明書公開レルムの制限
- CA の置換および OracleAS Certificate Authority の削除

# OracleAS Certificate Authority の Wallet 操作

Wallet は、証明書および信頼できる認証局の証明書のコンテナです。Oracle Application Server Certificate Authority は、これらの必須要素について、セキュアな格納およびアクセスに Wallet を使用します。証明書、信頼できる認証局またはパスワードを変更する場合、管理者は整合性とセキュリティを維持しながらそれらを使用できるようにアクションを実行する必要があります。次の項で、そのようなアクションについて説明します。


- CA 署名 Wallet の再生成
- CA SSL Wallet および CA SMIME Wallet の再生成
- 重要な Wallet の更新
- パスワードの変更


## CA 署名 Wallet の再生成

**注意：** この操作を試行する際は、注意する必要があります。この操作では CA 署名証明書が再生成されるため、既存の CA 証明書が置換され、既存の CA が発行した証明書がすべて無効になります。

Oracle Application Server Certificate Authority をルート認証局（CA）としてインストールすると、CA 署名証明書、CA SSL Wallet および CA SMIME Wallet が作成されます。CA 鍵が危殆化した場合は、次の項の説明に従い、`ocactl` 管理コマンドライン・ツールを使用してこれらの Wallet を再生成できます。

新しい CA 証明書および秘密鍵は OCA リポジトリに格納されます。この秘密鍵は、生成中に要求されたパスワードによって暗号化されます。以前の CA 署名証明書のエントリ、および以前の CA 署名証明書が発行したその他のすべての証明書は無効になります。CA SSL や CA SMIME などの重要な Wallet は、再生成する必要があります。CA Wallet を再生成した後、以前の CA が発行した CRL は無効になります。

CA Wallet を生成するコマンドの例を示します。

```
ocactl generatewallet -type CA
```

このコマンドを実行するには、OCA を停止する必要があります。また、この処理には数分かかる場合があります。OCA の再起動の手順は、第 3 章「OCA および証明書の管理の概要」の「Oracle Application Server Certificate Authority の起動および停止」の項を参照してください。

# CA SSL Wallet および CA SMIME Wallet の再生成

CA SSL Wallet はインストール時に生成され、これを使用すると、Oracle Application Server Certificate Authority エンジンが HTTPS モードでリスニングできるようになります。状況によっては、OCA サーバーとのセキュアな通信を確立するために、CA SSL Wallet および CA SMIME Wallet の再生成が必要になります。こうした状況には、Wallet が危殆化または破壊された場合、CA Wallet が再生成された場合、新しい下位 CA 証明書がインポートされた場合などがあります。

CA SSL Wallet を生成するコマンドの例を示します。

```
ocactl generatewallet -type CASSL
```

このコマンドを実行する場合は、OCA、OCA の OC4J および OHS をすべて停止する必要があります。このコマンドを実行した後で、OHS、OCA の OC4J および OCA を、この順序で再起動します。

この Wallet は、ディレクトリ $ORACLE_HOME/oca/wallet/ssl に ewallet.p12（PKCS #12）として格納され、生成時に指定したパスワードによって暗号化されます。また、このコマンドでは SSO 形式の CA SSL Wallet も生成され、$ORACLE_HOME/oca/wallet/ssl に cwallet.sso として格納されます。

cwallet.sso を使用するメリットは、Oracle HTTP Server の管理者が、Wallet のパスワードを指定しなくても、HTTP Server を SSL モードで起動できることです。通常、このパスワードは、PKCS #12 Wallet を使用して HTTP Server を SSL モードで起動する際に要求されます。

SSO 形式の Wallet は暗号化されており、ユーザーは、ファイルを開いたり鍵を抽出することができません。ただし、この Wallet は所有者権限のみで作成されるため、Wallet を保護するには、オペレーティング・システムのファイル権限が必要です。次回、OPMN で OCA インスタンスを起動したときに、SSL サーバー認証でこの Wallet が使用されます。

## CA SMIME Wallet

CA SMIME Wallet は、Oracle Application Server Certificate Authority でアラートおよび通知メッセージに署名するために使用されます。この Wallet が使用されるのは、OCA 管理ページの「構成管理」の「通知」ページで、「SMIME 電子メールの送信」を有効にした場合だけです。

> **関連項目：** 第 4 章「Oracle Application Server Certificate Authority の構成」の「メール詳細」の項

この SMIME Wallet が危殆化または破壊された場合、または CA Wallet が再生成されたときは、CA SMIME Wallet を再生成する必要があります。この Wallet は、Wallet の生成中に指定したパスワードによって暗号化されます。

CA SMIME Wallet を生成するコマンドの例を示します。

```
ocactl generatewallet -type CASMIME
```

次の手順に従って、CA SMIME Wallet を生成および使用します。

1. OCA を停止します。次のコマンドを使用します。

   ```
   $ORACLE_HOME/oca/bin/ocactl stop
   ```

2. 前述のコマンドを使用して、CA SMIME Wallet を生成します。
3. 「SMIME 電子メールの送信」を有効にしていない場合は、OCA の Web ベースのインタフェースで「構成管理」の「通知」ページに移動し、「SMIME 電子メールの送信」を有効にします。このオプションは、生成された CA SMIME Wallet を使用して、アラートおよび通知に署名します。
4. OCA を起動します。次のコマンドを使用します。

   ```
   $ORACLE_HOME/oca/bin/ocactl start
   ```

CA SMIME Wallet を再生成した後、以前の CA SMIME は無効になります。新しい CA SMIME Wallet を使用して、アラート・メッセージおよび通知メッセージに署名します。

## 重要な Wallet の更新

証明書の満了日が近づくと、Wallet の更新が必要になります。CA Wallet、CA SSL Wallet および CA SMIME Wallet は、`ocactl` 管理コマンドライン・ツールを使用して更新できます。`renewcert` コマンドの実行中、`ocactl` から新しい有効期間の入力が求められます。入力値は証明書の更新期間（日数）です。

CA 署名証明書を更新すると、新しい有効期間が設定された新しい証明書が作成され、OCA のメタデータ・リポジトリに格納されます。

CA SSL Wallet を更新すると、$ORACLE_HOME/oca/wallet/ssl/ に格納されている以前の Wallet ewallet.p12 は、更新された Wallet で上書きされます。また、CA SSL Wallet の更新によって、$ORACLE_HOME/oca/wallet/ssl/ の cwallet.sso も上書きされます。

CA SMIME Wallet を更新すると、$ORACLE_HOME/oca/wallet/email に格納されている以前の CA SMIME Wallet は、新しい Wallet で上書きされます。

CA Wallet を更新するコマンドの例を示します。

```
ocactl renewcert -type CA
```

更新した Wallet を有効にするには、第 3 章「OCA および証明書の管理の概要」の「Oracle Application Server Certificate Authority の起動および停止」の項の説明に従い、OHS、OCA の OC4J および OCA を、この順序で再起動する必要があります。

## パスワードの変更

インストール後は、CA、CA SSL、CA SMIME およびデータベース（DB）のパスワードを変更できます。データベース以外のパスワードはすべて、Oracle Application Server Certificate Authority が動作中の場合でも変更できます。既存の DB パスワードを使用して CA への接続をアクティブにすると、Oracle Application Server Certificate Authority を停止するまで DB パスワードは変更できません。

---

**注意：** OCA スキーマのパスワードは、ocactl setpasswd -type DB コマンドを実行しないと変更できません。sqlplus などを使用して、データベースに直接アクセスしても変更できません。

---

これらのコマンドを実行して変更された内容は、次回、Oracle Application Server Certificate Authority を起動したときに有効になります。`ocactl` を使用するたびに、OCA 管理者のパスワードが必要になります。このパスワードが認証されると、コマンドで指定したロール・タイプの新しいパスワードの入力が要求され、パスワード・ストアのパスワードと置換されます。この結果は、OCA 管理者の最新のパスワードを使用して再度暗号化されます。

OCA リポジトリ・パスワードを変更するコマンドの例を示します。

```
ocactl setpasswd -type DB
```

---

**注意：** CA SSL Wallet のパスワードを変更した場合は、OHS、OCA の OC4J および OCA を、この順序で再起動する必要があります。

---

# OracleAS Certificate Authority の構成操作

OracleAS Certificate Authority の管理者は、OracleAS Certificate Authority を使用する現場のニーズを満たすように、OracleAS Certificate Authority を構成する必要があります。これらの操作の一部は、Web ベースのインタフェースを介して行います。その他の操作では、`ocactl` など、OracleAS Certificate Authority が依存するコンポーネントを制御する OracleAS Certificate Authority 管理コマンドライン・ツールを使用する必要があります。次の項で、これらの構成操作および、管理者が実行する必要のあるアクションについて説明します。

- 第三者の SSL Wallet を使用するための Oracle HTTP Server の構成
- 認証局の証明書の失効
- OCA Web 管理者の証明書の失効
- OCA 画面での National Language Support（NLS）の構成

## 第三者の SSL Wallet を使用するための Oracle HTTP Server の構成

OCA をインストールすると、自動的に SSL モードで構成されます。明示的に CA をインポートするか、CA エントリを追加編集して CA 証明書をインポートするまでは、このサイトは信頼できないという警告が表示されます。この警告が表示されないように、OCA 管理者は VeriSign などの既知の CA から OCA サーバーの SSL 証明書を取得できます。

convertwallet コマンドを使用して、SSL サーバーの Wallet（PKCS #12 形式の ewallet.p12）を、SSO 形式の Wallet（ファイル名 cwallet.sso）に変換します。cwallet.sso を使用するメリットは、Wallet のパスワードを指定しなくても、HTTP Server を SSL モードで起動できることです。通常、このパスワードは、PKCS #12 Wallet を使用して HTTP Server を SSL モードで起動する際に要求されます。SSO 形式の Wallet は暗号化されており、ユーザーはファイルを開いたり鍵を抽出することができません。ただし、この Wallet は所有者権限のみで作成されるため、Wallet を保護するには、オペレーティング・システムのファイル権限が必要です。つまり、convertwallet コマンドを使用すると、Wallet パスワードを要求することなく、SSO（シングル・サインオン）で自動的に Web サーバーを SSL モードで起動できます。

既知の CA から Wallet をインポートするには、管理者は次の手順を実行します。

1. OCA、OCA の OC4J および OHS を停止します。
2. $ORACLE_HOME/oca/wallet/ssl に Wallet をバックアップします。
3. Oracle Wallet Manager を使用して、完全な SSL サーバー Wallet を作成します。
   a. SSL 証明書を要求します。
   b. サーバー証明書を発行した第三者 CA の証明書をインポートします。
   c. 要求したサーバー証明書をインポートします。
4. OWM を使用して、現行の OCA CA の証明書をトラスト・ポイントとしてこの Wallet にインポートします。

   **関連項目：**『Oracle Application Server 10*g* セキュリティ・ガイド』。特に、付録「Oracle Wallet Manager での PKI 資格証明の管理」。

5. $ORACLE_HOME/oca/wallet/ssl に Wallet を保存します。

   これで、OCA が発行した証明書は、CA SSL サーバーの証明書であるこの Wallet に対して、クライアントの証明書として信頼されます。
6. 第三者の CA（PKCS #12 形式）から作成された Wallet を、$ORACLE_HOME/oca/wallet/ssl/ewallet.p12 にコピーします。
7. `convertwallet -format SSO` を実行します。
8. OCA、OCA の OC4J および OHS を、この順序で再起動します。

## 認証局の証明書の失効

CA 署名証明書の失効は、OCA のインストールが機能しなくなり、すでに発行されている証明書が無効になるため、非常に影響が大きい操作です。この失効操作は、CA 鍵が危殆化された場合以外は実行しないでください。この操作を実行すると、新しい認証局をインストールできます。

下位 CA を使用すると、こうしたリスクとコストが軽減されます。階層的な CA 構造では、通常の操作は下位 CA で実行でき、ルート CA は高度にセキュアな場所でオフライン化されるなど特別に保護されます。この方法であれば、オンラインの下位 CA が危殆化した場合でも、それを失効させ、新しい下位 CA を作成して置換することができます。それ以前のすべての操作では、発行済の証明書を引き続き使用できます。

ただし、ルート CA が危殆化した場合は、まったく新しいインフラストラクチャを構築して、元のルート CA に依存するアプリケーションをすべて更新する必要があります。

こうした理由により、CA を階層化して、ルート CA を特別に保護することをお薦めします。

revokecert コマンドを使用すると、ルート認証局または OCA 管理者の証明書を失効させることができます。このコマンドが使用できるのは、OCA サービスが実行されていない場合に限られます。ルート認証局の証明書の失効は、実行中の OCA 操作用に新しい CA 署名証明書をインストールする前に実行する必要があります。

新しい CA をインストールする場合、最初に、既存の CA が発行したすべての証明書を失効させ、証明書失効リストを更新します。新しい CA 署名証明書が生成されるまで、以前の CA が署名したすべての証明書が OCA リポジトリで「無効」のマークが付けられるため、この手順は必須です。

その後、`revokecert` を使用して、パラメータで理由コードを指定し、以前の CA Wallet を失効させます。CA 署名証明書が失効になると、CA が発行したすべての証明書を失効させていない場合は、それらは一貫性のない状態になります。

OCA 管理者の証明書を失効させると、新しい証明書を取得するまで、管理者は、Web 上の管理機能にいっさいアクセスできません。管理者が管理ホームページを開くと、新しい管理者の証明書を取得するために、新規登録が要求されます。

鍵が危殆化された場合に、CA 証明書を失効させるコマンドの例を示します。

```
ocactl revokecert -type CA -reason KEY_COMPROMISE
```

CA 証明書を失効させ、OCA を再起動するには、次の手順を実行します。

1. OCA を停止します。次のコマンドを使用します。

   ```
   $ORACLE_HOME/oca/bin/ocactl stop
   ```

2. 前述のコマンドを使用して、CA Wallet を失効させます。
3. CA SSL Wallet を再生成します。

4. OCA を起動します。次のコマンドを使用します。

```
$ORACLE_HOME/oca/bin/ocactl start
```

## OCA Web 管理者の証明書の失効

管理者の証明書を置換する必要が発生する場合があります。その原因には、秘密鍵のパスワードの紛失、秘密鍵の危殆化または盗難、新しい人間への管理者ロールの付与などがあります。この失効操作を実行するのは、Web 管理者の鍵が危殆化された場合だけです。この操作によって新しい OCA Web 管理者を登録できます。

管理者の証明書を置換するには、サーバーを停止し、現行の管理者証明書を失効させて、サーバーを再起動する必要があります。これらのタスクは、コマンドライン・ツール `ocactl` を使用して実行します。このツールには OCA 管理者のパスワードが必要です。

OCA 管理者の証明書を失効させると、新しい証明書を取得するまで、管理者は、Web 上の管理機能にいっさいアクセスできません。管理者が管理ホームページを開くと、新しい管理者の証明書を取得するために、新規登録が要求されます。

次に、管理者は、Oracle Application Server Certificate Authority の Web ページにナビゲートし、OCA 管理フォームに必要事項を入力します。

鍵が危殆化された場合に、Web 管理者の Wallet を失効させるコマンドの例を示します。

```
ocactl revokecert -type WEBADMIN -reason KEY_COMPROMISE
```

Web 管理者の証明書を失効させ、OCA を再起動するには、次の手順を実行します。

1. OCA を停止します。次のコマンドを使用します。

```
$ORACLE_HOME/oca/bin/ocactl stop
```

2. OCA の OC4J を停止します。`ORACLE_HOME` で次のコマンドを使用します。

```
$ORACLE_HOME/opmn/bin/opmnctl startproc type=oc4j instancename=oca
```

3. 前述のコマンドを使用して、Web 管理者の証明書を失効させます。
4. OCA を起動します。次のコマンドを使用します。

```
$ORACLE_HOME/oca/bin/ocactl start
```

## OCA 画面での National Language Support（NLS）の構成

次の条件を満たす場合は、OracleAS Certificate Authority の管理者用画面およびユーザー用画面を、クライアントまたはサーバーの言語で表示できます。

1. データベースのキャラクタ・セットは UTF8 である必要があります。
2. OracleAS Certificate Authority の UI およびオンライン・ヘルプは、クライアントのロケールでレンダリングされます。クライアントのロケールがサポートされない場合は、画面はサーバーのロケールでレンダリングされます。サーバーのロケール言語も OCA によりサポートされない場合は、英語が使用されます。
3. 認証局運用規程は、管理者が認証局運用規程の編集に使用したロケールでレンダリングされ、クライアントのロケールには左右されません。
4. ocactl での NLS のサポートは、サーバーのロケールによって決まります。サーバーのロケールが OCA のサポート対象言語でない場合、表示は英語になります。
5. どのロケールでも、実際の ocactl コマンド自体は英語です。
6. アラート、通知、エラー・メッセージなどの情報メッセージは、サーバーのロケールの言語で表示されます。サーバーとクライアントでロケールの言語が異なる場合でも、クライアントの言語では表示されません。たとえば、OCA のインストール先サーバーのロケールの言語が英語のときに、クライアントから日本語で要求が送信された場合、通知は英語で出力されます。

   アラートまたは通知のカスタマイズにテンプレートを使用する場合（これについては次の項で説明）、テンプレートの編集に使用した言語が使用されます。メッセージの本文はサーバーのロケールの言語でエンコードされるため、テンプレートの編集はサーバーの言語で行うことをお薦めします。

   テンプレートを使用しない場合、アラートと通知はすべて、サーバーのロケールの言語で表示されます。

   > **注意：** クライアントのロケールの言語がアラビア語の場合、画面は英語でレンダリングされます。ただし、レンダリングは右から左の順序で行われます（不具合の番号は 3382624 および 3384940）。

7. CA 証明書および CA SSL 証明書の DN には、マルチバイト・キャラクタを使用できません。CA の DN にマルチバイト・キャラクタが使用されている場合は、インストールに失敗します（不具合 : 2991110）。
8. OracleAS Certificate Authority をインストールする際、zh ロケールと zh_TW ロケールでは、いずれも OCA のインストールおよび起動を実行できません。これらのロケールのかわりに、次のロケールのいずれかを使用します。

   中国語（簡体字）用には zh_CN.GBK

   中国語（繁体字）用には zh_TW.BIG5

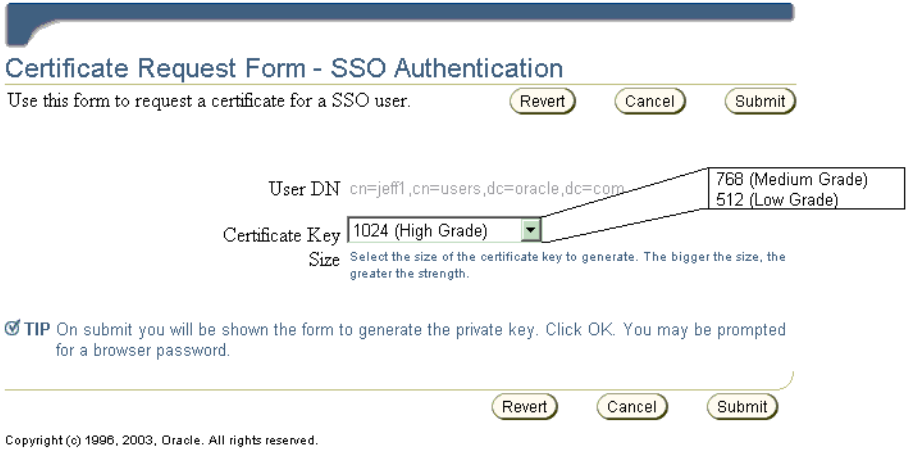
# カスタマイズのサポート

OCA では、次の 3 つのプロビジョニング・ページに独自のヘッダーおよびフッターを指定することで、SSO と OCA 間のインタフェースをカスタマイズできます。

1. 「ようこそ」画面


2. 「登録」画面

3. 「証明書のインポート」画面




**関連項目：** 第 3 章「OCA および証明書の管理の概要」の「Single Sign-On（SSO）および OracleAS Certificate Authority（OCA）」の項

デフォルトでは、OCA の既存画面はカスタマイズなしでレンダリングされますが、OCA 管理者は固有のヘッダーまたはフッターを使用して、これら 3 つのいずれの画面もカスタマイズできます。OCA 管理者は、これらの各画面にカスタム HTML ファイルを指定することで、対応するデフォルト画面ではなくカスタマイズした画面を使用するように OCA に通知します。これらのカスタム HTML ファイルには、静的な HTML コンテンツを含めることができます。カスタマイズされた HTML ファイルがない場合、またはそのサイズが 0（ゼロ）の場合は、デフォルト画面が使用されます。

こうしたカスタム HTML ファイルの作成に使用できるテンプレートは、$ORACLE_HOME/oca/templates/screens という名前のディレクトリにあります。管理者は、このコンテンツのルック・アンド・フィールを制御します。

サイズが 0 でない画面カスタマイズ用の HTML ファイルがある場合は、デフォルト画面のかわりに、そのコンテンツが使用されます。

**注意：**

これらの HTML ファイルのいずれかを変更した後は、変更内容を有効にするために、OCA を再起動する必要があります。

画面、メッセージ、アラート、通知など、カスタマイズされたものの内容、翻訳およびアクセス可能性に関しては、OCA は何も行いません。カスタマイズされた内容はそのままレンダリングされます。

**表 6-1 Single Sign-On のポップアップ画面のカスタマイズ**

| 画面名 | テキストを置換できる位置 | 置換目的のテキストを含むファイル[1] |
|---|---|---|
| 「ようこそ」画面 | ヘッダーの場合：「OracleAS Certificate Authority へようこそ」と書かれた OCA 行と「証明書を取得するにはここをクリック」と書かれた OCA 行の間 | $ORACLE_HOME/oca/templates/screens/homeheader.html |
| | フッターの場合：「証明書を取得するにはここをクリック」と書かれた行の下 | $ORACLE_HOME/oca/templates/screens/homefooter.html |
| 「登録」画面 | ヘッダーの場合：OCA の最上部にある青いバーと、「ユーザー DN」と書かれた行の間 | $ORACLE_HOME/oca/templates/screens/enrollheader.html |
| | フッターの場合：最下部にある「キー・サイズ」と書かれた行と「SKI」と書かれた行の下 | $ORACLE_HOME/oca/templates/screens/enrollfooter.html |
| 「証明書のインポート」画面 | ヘッダーの場合：「証明書の表示」と書かれた OCA 行と「証明書詳細」と書かれた OCA 行の間 | $ORACLE_HOME/oca/templates/screens/importheader.html |
| | フッターの場合：最下部にある「証明書詳細の後」と書かれた OCA 行と「SKI」と書かれた OCA 行の間 | $ORACLE_HOME/oca/templates/screens/importfooter.html |

[1] この列に示すいずれかのファイルが 0 以外のサイズで存在する場合は、対応するヘッダーまたはフッターが、そのファイルの静的 HTML で置換される。

# Oracle Application Server Certificate Authority での OCA アクションのログまたはトレース

ocactl 設定コマンドを使用すると、ログおよびトレースが有効になり、ログおよびトレースのストレージに記録された OCA 操作および管理操作を参照できます。

**表 6-2 OCA のログ・データおよびトレース・データの格納場所**

| データ型 | 格納の形態 | 場所 |
|---|---|---|
| LOG | OCA リポジトリ | OCA リポジトリ |
| TRACE | ファイル : oca.trc | $ORACLE_HOME/oca/logs |
| ADMIN LOG | ファイル : admin.log | $ORACLE_HOME/oca/logs |
| ADMIN TRACE | ファイル : admin.trc | $ORACLE_HOME/oca/logs |

設定コマンドの形式は次のとおりです。

```
ocactl set  -type {LOG | TRACE} -mode {OCA|ADMIN} -state {ON|OFF}
```

例 :

1. `ocactl set -type LOG -mode OCA -state ON`
   OCA リポジトリへのログ・メッセージの格納を有効にします。
2. `ocactl set -type TRACE -mode OCA -state ON`
   oca.trc ファイルへのトレース・メッセージの格納を有効にします。
3. `ocactl set -type LOG -mode ADMIN -state ON`
   admin.log ファイルへのログ・メッセージの格納を有効にします。
4. `ocactl set -type TRACE -mode ADMIN -state ON`
   admin.trc ファイルへのトレース・メッセージの格納を有効にします。
5. `ocactl set -type TRACE -state OFF`
   トレースをオフにします。トレース・データは格納されません。

## OracleAS Certificate Authority のログ情報またはトレース情報の消去

ocactl 管理コマンドライン・ツールを使用すると、管理者の選択で既存のログ・ストレージまたはトレース・ストレージを消去できます。OCA LOG は OCA リポジトリに格納されます。また、OCA TRACE は $ORACLE_HOME/oca/logs の oca.trc に格納されます。ADMIN LOG は $ORACLE_HOME/oca/log の admin.log に、ADMIN TRACE は $ORACLE_HOME/oca/logs の admin.trc に格納されます。

有効な形式およびモードで消去コマンドを実行すると、ログまたはストレージの古い情報は消去されます。ファイル oca.trc、admin.trc および admin.log は、OS の削除コマンドを使用してファイル・システムから削除できます。

消去コマンドの形式は次のとおりです。

```
ocactl clear -type {LOG |TRACE}  -mode {OCA|ADMIN}
```

例 :

1. `ocactl clear -type LOG -mode ADMIN`

   $ORACLE_HOME/oca/logs から ADMIN LOG ファイル admin.log を削除します。

2. `ocactl clear -type TRACE -mode ADMIN`

   $ORACLE_HOME/oca/logs から ADMIN TRACE ファイル admin.trc を削除します。

3. `ocactl clear -type LOG -mode OCA`

   OCA リポジトリのログ・メッセージを削除します。

4. `ocactl clear -type TRACE -mode OCA`

   $ORACLE_HOME/oca/logs から OCA TRACE ファイル oca.trc を削除します。

# OCA が使用するインフラストラクチャ・サービスの変更

新しいポートやホストを使用するなど、OracleAS Single Sign-On（SSO）および Oracle Internet Directory（OID）に対する変更は、次に示す状況など、様々な形で発生します。

- バックアップ後の操作のリストア
- LDAP（ディレクトリ）または Oracle Database の構成の変更
- パイロット・シナリオから本番環境への移行

**関連項目：**『Oracle Application Server 10*g* 管理者ガイド』

OCA は、Oracle Identity Management（IM）インフラストラクチャのコンポーネントとしてインストールされ、OID、SSO および Metadata Repository のサービスを使用します。これらのコンポーネントのいずれかが置換またはリストアされた場合は、これらの新しいサービスを使用するように OCA を構成できます。OCA は、これら 3 つのコンポーネントの既存バージョンを使用するか、新しい OID、SSO および Metadata Repository を使用して動作できます。

OracleAS では、次の 2 つのタイプのインフラストラクチャの変更がサポートされます。

- OCA が使用する Identity Management（IM）サービス（SSO/OID）の変更
- OCA が使用する Metadata Repository（MR）サービスの変更

次の項では、これらのサービスに関するデータの表示について説明します。

- OCA の接続情報の格納場所および表示場所

## OCA が使用する Identity Management（IM）サービス（SSO/OID）の変更

新しい SSO または OID のインストール後に、OCA が使用する IM サービスを変更するには、次の 2 つの手順を実行する必要があります。

- 新しい IM をインストールし既存データを移行します。
- 新しくインストールした IM サービスを使用できるように、OCA を構成します。

OracleAS には、新しい IM（SSO/OID）がインストールされたという前提で、ある IM インスタンスから別のインスタンスにデータを移行するスクリプトが用意されています。ただし、Application Server Control コンソールの「Infrastructure」ページの識別管理の変更ウィザードは、OCA には使用できません。OCA 自体がインフラストラクチャ・コンポーネントのためです。そのため、OCA では、OCA が使用する IM サービスの変更は、OCA 管理コマンドライン・ツール `ocactl` の changesecurity コマンドによりサポートされます。

**関連項目：**

- OCA 管理コマンドライン・ツールの詳細は、付録 A「コマンドライン管理」を参照してください。
- スクリプトを含む、Identity Management インフラストラクチャの IM サービスおよびメタデータ・サービスの変更の詳細は、『Oracle Application Server 10*g* 管理者ガイド』を参照してください。

OCA に新しい IM サービスを構築するには、次の手順を実行します。

1. マシン 1 に、Identity Management および Metadata Repository をインストールします。
2. マシン 2 に、Identity Management をインストールします。
3. OracleAS に用意されているスクリプトを使用して、マシン 1 からマシン 2 に IM データを移行します。
4. OCA が稼動するマシン（マシン 1）で、OCA、OCA の OC4J および OHS を停止します。次のコマンドを使用します。

   ```
   $ORACLE_HOME/oca/bin/ocactl stop
   $ORACLE_HOME/opmn/bin/opmnctl stopall
   ```

5. マシン 1 で、ias.properties ファイルを編集し、`$ORACLE_HOME/config` ディレクトリの下にある OIDhost パラメータおよび OIDport パラメータが、新しい IM（マシン 2）を指すようにします。
6. マシン 1 で、次のコマンドを実行します。

   ```
   $ORACLE_HOME/oca/bin/ocactl changesecurity -server_auth_port portno
   ```

   このコマンドは、次の 2 つのアクションを実行します。

   IM サービスの新しいマシン（マシン 2）を指すよう、$ORACLE_HOME/oca/conf にある oca.conf ファイルを更新します。

   新しい SSO サーバー（マシン 2）に OCA を登録します。

---

**注意：** Identity Management（IM）の再関連付けを行うことで、スケーラビリティやフェイルオーバーの確保を目的とした SSO サービスまたは OID サービスの構成変更や、パイロット IM から本番 IM への移行に対応できます。

このような再関連付けの詳細は、『Oracle Application Server 10*g* 管理者ガイド』を参照してください。

---

## OCA が使用する Metadata Repository（MR）サービスの変更

元の物理データベースから別の物理データベースへの OCA のメタデータ・サービスの変更は、サポートされていません。ただし、リスナーやポートの変更など、接続文字列の変更は、updateconnection コマンドを使用することで対応できます（付録 A を参照）。

## OCA の接続情報の格納場所および表示場所

OCA リポジトリおよびディレクトリへの接続を定義する情報（証明書の公開に使用）は、Oracle Internet Directory（OID）に格納されます。この接続情報は、最初に、OracleAS のインストール時に OID に書き込まれます。同時にこの接続情報は OID からフェッチされ、Oracle Application Server Certificate Authority の構成ファイル `$ORACLE_HOME/oca/conf/oca.conf` に書き込まれます。

この接続情報は、Oracle Application Server Certificate Authority 管理者用の Web ベースのインタフェースの「一般」サブタブにある設定セクションに表示されます。

**関連項目：** 付録 A「コマンドライン管理」の表 A-2 にある updateconnection

# OracleAS Certificate Authority および高可用性機能

Oracle Application Server の高可用性機能は、『Oracle Application Server 10*g* 高可用性ガイド』で詳細に説明されています。ここでの説明は、これらの機能を紹介するための概要です。

Oracle Application Server Certificate Authority を使用すると、実際の高可用性システムで、迅速かつ容易に証明書を使用できます。次の項で、Oracle Application Server の Cold Failover Clusters および Real Application Clusters（RAC）の高可用性機能をサポートするリンク、手順、表記規則および準備事項について説明します。

- コールド・フェイルオーバーを使用した OracleAS Certificate Authority の配置
- Real Application Clusters を使用した OracleAS Certificate Authority の配置

**関連項目：** 『Oracle Application Server 10*g* 高可用性ガイド』

## コールド・フェイルオーバーを使用した OracleAS Certificate Authority の配置

コールド・フェイルオーバー構成では、複数の物理ホストから共有ディスク上の共通のストアにアクセスでき、各物理ノードは、同時に 1 つ以上の論理ホストをホスティングできます。Oracle Application Server Cold Failover Cluster を使用すると、OracleAS インスタンスを障害ノードからバックアップに透過的にフェイルオーバーできます。また、メンテナンスのためにフェイルオーバーを手動で開始することもできます。

この例では、ソフトウェアとデータベースをそれぞれ 1 つしかインストールしていません。また、2 つの物理ホストが、OCA/OracleAS のソフトウェアとデータベースがインストールされたディスクへのアクセスを共有します。OracleAS のハードウェアがマシンのクラスタとして構成される場合は、インストーラはそのノードをクラスタの一部と認識し、仮想ホストの名前の入力を要求します。物理ホスト 1 に障害が発生した場合、またはメンテナンスのためにオフラインになっている場合、その論理ホスト名（仮想ホスト A）は他の物理ホストに移行されます。ベンダー固有のスクリプトとハードウェア・クラスタ・ソフトウェアを使用すると、必要なデータベース、リスナーおよび OCA/OracleAS プロセスを起動して、透過的フェイルオーバーを有効にできます。クライアントは、最小限のサービス停止で、以前と同じ論理ホストへの接続を継続できます。HTTP Server および OC4J を起動した後は、`ocactl start` コマンドを使用して、Oracle Application Server Certificate Authority も再起動する必要があります。

**関連項目：**『Oracle Application Server 10*g* 高可用性ガイド』

## Real Application Clusters を使用した OracleAS Certificate Authority の配置

Oracle Real Application Clusters は、OracleAS Infrastructure で堅牢なクラスタ・アーキテクチャを構築できるので、Cold Failover Clusters よりもさらに透過的な高可用性ソリューションが実現します。Real Application Clusters ソリューションではすべてのノードがアクティブなため、あるノードから別のノードへのフェイルオーバーが高速で、手動での介入を必要としません。また、この相互にアクティブなセットアップにより、この上に配置されている OracleAS Infrastructure のスケーラビリティも向上します。

データベース・ファイルは、すべてのノードからアクセスできる共有ストレージにインストールされます。データベースは、データベース・インスタンスにより、読取り / 書込み操作に対して同時に開きます。インフラストラクチャ構成ファイルはデータベースにではなく、各ノードのローカルなファイル・システムにあり、ノード固有ではない同一の構成情報が格納されます。クラスタは 3 つ以上のノードで構成でき、それらのすべてが中間層からの要求をアクティブに受け入れます。

クラスタのフロントエンドにはロード・バランサ機器が配置されます。ロード・バランサは、それ自体に障害が発生する場合に備えて、可用性を維持できるようなフォルト・トレラントなモードで配置することをお薦めします。このロード・バランサは、HTTP、HTTPS、LDAP（ディレクトリ）要求など、Oracle Net でない通信を、中間層から Infrastructure に転送します。ロード・バランサの構成は、中間層からどのアクティブな Infrastructure ノードにも要求を転送できるように設定します。

OCA では、RAC は完全にサポートされません。OCA では、Oracle Internet Directory、Oracle Database、OracleAS Single Sign-On など、RAC 構成の他のインフラストラクチャ・コンポーネントを使用できますが、OCA 自体は RAC モードにできません。

**関連項目：** これらのコンポーネントを RAC モードでインストールする方法は、『Oracle Application Server 10*g* 高可用性ガイド』を参照してください。

---

**注意：** OracleAS Certificate Authority 10*g*（9.0.4）では、RAC は Windows でサポートされていません。

---

# OracleAS Certificate Authority のバックアップおよびリカバリでの考慮事項

バックアップおよびリカバリという言葉は、データの消失に対する防御とデータを消失した場合の再構築の両方に際して行う様々な戦略と手順を指しています。OracleAS Backup and Recovery Tool は、障害が発生した場合に、OracleAS 環境のバックアップとリカバリを支援します。

**関連項目：** Backup and Recovery Tool の詳細な説明と手順は、次のドキュメントを参照してください。

- 使用可能な各種バックアップおよびリカバリ方法、OracleAS Backup and Recovery Tool のインストールおよび構成、コンポーネント単位のバックアップおよびリカバリのそれぞれの詳細は、『Oracle Application Server 10*g* 管理者ガイド』にあるバックアップとリカバリの説明を参照してください。
- データベースのバックアップは、『Oracle Database バックアップおよびリカバリ・アドバンスト・ユーザーズ・ガイド』に示すバックアップおよびリカバリに関する Oracle のガイドラインを参照してください。
- Oracle Internet Directory のバックアップは、『Oracle Internet Directory 管理者ガイド』を参照してください。

この後の説明はあくまで概要です。詳細な情報は、前述のドキュメントを参照してください。

次のような状況で、バックアップ / リカバリ技法を使用してデータをリカバリできます。

**表 6-3 バックアップ/ リカバリの使用例**

| 状況 | 対応策 |
|---|---|
| ホストの消失 | ホスト名および IP アドレスが同じ新しいホストにリストアできます。<br>あるいは、ホスト名および IP アドレスが異なる新しいホストにリストアすることもできます。 |
| Oracle ソフトウェア / バイナリの消失または破損 | Oracle バイナリが破損または消失した場合は、インフラストラクチャ全体をリカバリする必要があります。 |
| データベース・インスタンスのクラッシュなど、Metadata Repository インスタンスの障害 | データベース・インスタンスのリカバリ方法を使用して、Metadata Repository インスタンスをリカバリします。 |
| Metadata Repository データベースの障害（Metadata Repository のみが破損し、インフラストラクチャの Oracle ホームにある他のファイルは破損していない場合） | B/R スクリプトを使用して Metadata Repository をバックアップし、OracleAS Backup and Recovery Tool を使用してデータベースをリカバリします。 |
| OracleAS コンポーネントのランタイム構成ファイルの削除または破損 | B/R スクリプトを使用して、構成ファイルをリストアします。 |
| Metadata Repository リスナーの障害 | リスナー・プロセスを停止して再起動します。 |

メンテナンスが必要な場合やサービスが予期せず消失した場合は、様々なバックアップおよびリカバリ手順によって、Oracle Application Server Certificate Authority（OCA）の情報と機能が保護および保持されます。

バックアップ方法および対応するリカバリ方法は、次の Backup and Recovery Tool によりサポートされます。

**表 6-4 Backup and Recovery Tool**

| ツール名 | 機能 |
|---|---|
| コールド・バックアップ / リカバリ | OracleAS Infrastructure の全プロセスおよび Metadata Repository のクリーンな通常停止の完了後にバックアップされた Oracle ホーム、構成ファイル、データベース・ファイルなど、OracleAS Infrastructure インスタンス全体をリストアすることを指します。 |
| 部分オンライン（ホット）バックアップ / リカバリ | OracleAS のインスタンスおよび Metadata Repository の適切なオンライン・バックアップの完了後に、バックアップされた OracleAS Infrastructure の構成ファイルおよびデータベース・ファイルをリストアすることを指します。 |
| 構成ファイルの 増分バックアップ / リカバリ | オンライン・バックアップから取得した OracleAS Infrastructure 構成ファイルのみをリストアすることを指します。 |

OCA は、Oracle Database をプライマリ・リポジトリとして使用しているため、データベースに格納されている OCA の情報は、データベースがバックアップされる際に自動的にバックアップされます。同様に OCA は、証明書および特定の SSO 操作の公開に Oracle Internet Directory（OID）を使用します。前述の 3 つのドキュメントには、関連するバックアップ操作およびリカバリ操作がすべて詳細に説明されています。

データベースおよびディレクトリに格納されている情報に加え、Oracle Application Server Certificate Authority は多くの重要なオペレーティング・システム・ファイルも作成します。これらのファイルは、通常のバックアップ処理の一部としてバックアップする必要があります。次のようなファイルがあります（$ORACLE_HOME は OCA がインストールされているホーム・ディレクトリを表します）。

- $ORACLE_HOME/oca/conf/oca.conf
- $ORACLE_HOME/oca/pwdstore/ocmpassword.p12
- $ORACLE_HOME/oca/wallet/ssl/ewallet.p12
- $ORACLE_HOME/oca/wallet/ssl/cwallet.sso
- $ORACLE_HOME/Apache/Apache/conf/httpd.conf
- $ORACLE_HOME/Apache/Apache/conf/ocm_apache.conf
- $ORACLE_HOME/Apache/Apache/conf/osso/oca/osso.conf

# 証明書公開レルムの制限

各部門が地理的に分散している大規模な組織では、より効率的なローカル管理のために、部門ごとに別々の認証局を確立できます。これらの部門は、Wyoming や New York など複数の州、または米国や英国など複数の国に存在する場合もあります。OCA の様々なインスタンスは、下位 CA や相互に信頼する独立した CA です。

デフォルトでは、Oracle Application Server Certificate Authority（OCA）のインスタンスを特定のマシンにインストールすると、エントリは Oracle Internet Directory に格納されます。インストールされた OCA インスタンスは、次の識別名で表されます。

```
cn=ocaN,cn=OCA,cn=PKI,cn=Products,cn=OraclContext
```

（N は 1、2 ...n）

現行の OCA に対応するエントリを確認するには、「管理」ページに進み、さらに「構成管理」タブの「一般」サブタブを参照します。「エージェントのディレクトリ設定エントリ」の DN に、現行の Oracle Internet Directory での現行の OCA が示されます。

デフォルトでは、このような各 CA は最上位の Oracle コンテキストであるグループ cn=PKIAdmins,cn=Groups,cn=OracleContext のメンバーです。

このような CA がユーザー証明書を公開すると、その証明書は自動的に Oracle Internet Directory 内の対応するサブスクライバ・レルムのユーザーの DN エントリに格納されます。デフォルトでは、すべての CA が信頼され、ディレクトリ全体のあらゆるユーザー・エントリに公開できます。たとえば、US レルムのユーザーは UK OCA から証明書を受信でき、ユーザー証明書は US レルムのそのユーザーの DN エントリに格納されます。

ただし、OCA インスタンスの公開権限を制限して、特定のサブスクライバ・レルム以外には公開しないようにできます。たとえば、UK OCA を UK サブスクライバ・レルムにのみ公開するように制限できます。このように制限した場合、ユーザーの通常のレルムは UK OCA にアクセスできないため、UK OCA が US ユーザーに発行した証明書は公開できません。

OCA を特定のレルムに制限する場合は、最上位のグループ（cn=PKIAdmins,cn=Groups,cn=OracleContext）から削除し、指定したグループにその OCA エントリを追加する必要があります。たとえば、OCA2 の公開をサブスクライバ dc=com,dc=acme のみに制限する場合は、次の 2 つのコマンドを使用します。

```
-remove cn=oca2,cn=cn=OCA,cn=PKI,cn=Products,cn=OraclContext from group
cn=PKIAdmins,cn=Groups,cn=OracleContext

-add cn=oca2,cn=cn=OCA,cn=PKI,cn=Products,cn=OraclContext to group
cn=PKIAdmins,cn=Groups,cn=OracleContext,dc=acme,dc=com
```

さらには、CA を制限するカスタム・プラグインを記述することで、特定の一連の DN からの証明書のみを管理できます。たとえば、第 5 章「Oracle Application Server Certificate Authority でのポリシー管理」で開発したカスタム・プラグインでは、米国以外のドメインからの証明書のみを発行するように CA を制限しています。

この制限は、第 5 章の「カスタム・ポリシー・プラグインの例」の項に示した例の 7 行目、すなわち次の行で記述されています。

```
7:              if (!policyRequest.getCountry().equals("US"))
```

「!」を削除して「US」を目的のレルムに変更するように少し変更し、さらに後続の数行を修正すると、これに依存する行によって、証明書の発行が、指定したレルムに制限されます。

# CA の置換および OracleAS Certificate Authority の削除

秘密鍵が危殆化したなどの理由で、まれにルート CA を抜本的に置換する必要が生じることがあります。このような場合は、OCA をいったん削除してから再インストールします。OCA を削除すると、元々インストールされていたデータベースおよび Oracle Internet Directory エントリのトレースはすべて削除されます。

このような削除を実行するには、Oracle Application Server 10*g* のインストレーション・ガイドの説明に従ってください。

# 7

# Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース

「エンド・ユーザー」は、ユーザーのみでなく、サーバーとアプリケーション間の認証を容易にするために証明書を取得するサーバー・エンティティも表します。

エンド・ユーザーおよび管理者が Oracle Application Server Certificate Authority サーバーと対話する際には、別々の HTML インタフェースを使用できます。エンド・ユーザーは、これらの HTML インタフェースを使用して、証明書に関連する個人的な操作を実行でき、管理者は証明書を管理できます。

この章の内容は、次のとおりです。


- ユーザー・インタフェースへのアクセス
- エンド・ユーザー用のタブおよび処理
  - 「ユーザー証明書」タブ
  - 証明書の検索、更新および失効
  - 「サーバー / 下位 CA 証明書」タブ
  - 下位 CA 証明書
- CA 証明書のダウンロード
- ブラウザへの証明書失効リスト（CRL）のインポート
- ファイル・システムへの証明書失効リスト（CRL）のダウンロード
- ブラウザへの新規発行の証明書のインポート
- ブラウザからの Wallet のエクスポート（バックアップ）
- ファイル・システムからの証明書のインポート


Netscape および Internet Explorer の両方がサポートされます。

# ユーザー・インタフェースへのアクセス

Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェースのホームページにアクセスするには、Web ブラウザを起動して、インストールの最後に表示された管理サーバーの URL およびポート番号を入力します。たとえば、次のように入力します。

```
https://Oracle_HTTP_host:ssl_port/oca/user
```

次のような、Oracle Application Server Certificate Authority のユーザー・ホームページが表示されます。


このページに説明されているとおり、この Web ベースのインタフェースを使用して、証明書の要求や、証明書または証明書要求の更新、失効または検索を行うことができます。これらの機能を使用するには、「ユーザー証明書」タブと「サーバー / 下位 CA 証明書」タブのいずれかをクリックします。

また、「ここをクリック」リンクを使用して、認証局の証明書または最新の証明書失効リスト（CRL）をブラウザにインポートすることもできます。

同様に、管理者は、管理者用の「ここをクリック」リンクを使用して、追加使用の目的で、ファイル・システムに CA 証明書または CRL をダウンロードできます。

# エンド・ユーザー用のタブおよび処理

Oracle Application Server Certificate Authority の Web ベースのインタフェースでは、次に示すとおり、エンド・ユーザーは認証局と 2 つのタイプの対話を実行できます。

「ユーザー証明書」タブでは、次の処理を行うことができます。

- Oracle Application Server Certificate Authority に対して、ユーザー自身で認証すること。この処理を実行するには、既存の Single Sign-On（SSO）または SSL 証明書を使用するか、管理者による手動の認証を要求します。
- OCA 管理者が手動で承認できるように、新しい証明書要求を作成すること（エンド・ユーザーまたはサーバーの場合）。
- 証明書の自動的な要求および受信（SSL ユーザーおよび SSO ユーザーの場合）。
- 証明書のインポート、表示、失効または更新。
- 認証方式の変更。
- CA 証明書のダウンロード。
- 最新の証明書失効リスト（CRL）のダウンロード。

表 7-1 に、Oracle Application Server Certificate Authority がサポートしている証明書のタイプと、それぞれについての簡単な説明を示します。

**表 7-1 証明書のタイプおよび使用方法**

| 証明書のタイプ | 意味および使用方法 |
|---|---|
| 暗号化 | 他のユーザーが公開鍵で暗号化したメッセージを送信できるようにし、それによって、自分だけが秘密鍵を使用してメッセージを解読できるようにします。 |
| 署名 | 秘密鍵でメッセージ・ダイジェストに署名することで、他のユーザーは公開鍵を使用して、このユーザーがメッセージを送信したこと、およびその内容が変更されていないことを検証できます。 |
| コード署名 | 秘密鍵でソフトウェアに署名することで、クライアントは公開鍵を使用して、このユーザーがソフトウェアの配布元であることを検証できます。 |
| 証明書への署名 | 発行する証明書に秘密鍵を使用して署名することで、受信者は公開鍵を使用して、証明書が送信者本人によって署名されていることを検証できます。 |
| SSL | 証明書を SSL 認証で使用します。 |

「サーバー / 下位 CA 証明書」タブでは、次の処理を行うことができます。

- ID、シリアル番号、一般名などによる、証明書および証明書要求の検索
- サーバー証明書および下位 CA 証明書の要求
- CA 証明書または証明書失効リスト（CRL）のインポート

## 「ユーザー証明書」タブ

このタブを初めて表示するときに、「認証」ページが表示されます。このページでは、Oracle Application Server Certificate Authority に対する、ユーザー自身の認証方法を選択できます。

表 7-2 に、使用可能な認証タイプおよび認証方式を示します。

**表 7-2 認証タイプ**

| 認証タイプ | 説明 | 方式の概要（詳細は次の項を参照） |
|---|---|---|
| Single Sign-On（SSO） | 認証は、Single Sign-On Server に基づいて自動化されます。通常は、パスワード・ベースです。 | 「OracleAS Single Sign-On 名およびパスワードの使用」というラベルが付いたラジオ・ボタンをクリックし、「送信」をクリックします。 |
| Secure Sockets Layer（SSL） | 認証は、事前に発行された SSL 証明書に基づいて自動化されます。 | 「既存の証明書の使用」というラベルが付いたラジオ・ボタンをクリックし、「送信」をクリックします。 |
| 手動 | 認証は自動化されません。「証明書要求」フォームに必要事項を入力して送信し、管理者からの承認を待ちます。 | 「手動承認 / 認証を使用します」というラベルが付いたラジオ・ボタンをクリックし、「送信」をクリックします。 |

**関連項目：** 認証については、第 2 章を参照してください。

次の項で、これらの認証タイプおよび認証方式について詳しく説明します。

- Single Sign-On（SSO）認証
- OracleAS Certificate Authority が信頼されるブラウザの構成
- Secure Sockets Layer（SSL）認証
- 手動認証
- 証明書の検索、更新および失効
- 「サーバー / 下位 CA 証明書」タブ
- 下位 CA 証明書

**注意：** Netscape では、エンド・ユーザーおよび管理者のいずれの場合にも、鍵のサイズ（512 または 1024）を選択できます。

Internet Explorer（IE）では、ハードコードされた基本的な 512 ビットが使用されます。これは通常、リストの最初に表示される選択肢で、次に拡張選択肢の 1024 ビット、その後に他の選択肢が続きます。ただし、IE のバージョンによっては、Gemplus がリストの最初に表示される場合もあります。コンピュータにスマートカードのカード・リーダーが装備されていない場合は、Gemplus を選択すると、鍵のサイズの解決法が見つからないためにエラーが表示されます。カード・リーダーが装備されている場合は、Gemplus スマートカードによって鍵のサイズが判断されます。

## Single Sign-On（SSO）認証

次の手順を実行して、必須の SSO 認証情報（ユーザー名やパスワードなど）を提供することにより、SSO ユーザーは自動的に証明書を取得したり、証明書を管理できます。

1. 「認証」フォームで、「OracleAS Single Sign-On 名およびパスワードの使用」というラベルの付いたオプションを選択して、「送信」をクリックします。
   SSO のログイン・ページにリダイレクトされます。
2. SSO ユーザー名およびパスワードを入力します。有効な証明書を示す「ユーザー証明書 - SSO」フォームが表示され、次の作業を行うことができます。
   - 証明書の取得
   - 選択した証明書の詳細の表示
   - 現行の証明書の更新
   - 現行の証明書の失効

証明書を取得するには、手順 3 ～ 5 を実行します。

3. 「ユーザー証明書 - SSO」フォームの「証明書の要求」をクリックして、「証明書要求」フォームを表示します。
4. 「証明書要求」フォームで、適切な情報を入力して、フォームを送信します。使用しているブラウザが Netscape の場合と Internet Explorer の場合では、表示される選択肢が少し異なります。
   - Netscape の場合は、512、1024 など、生成される鍵のペアのサイズ（ビット単位）を示す鍵サイズが表示されます。

- Internet Explorer の場合は、暗号化サービス用に選択可能なプロバイダを示すキーストアが表示されます。標準の選択肢には、「Microsoft Base Cryptographic Provider」、「Microsoft Enhanced Cryptographic Provider」および「Microsoft Strong Cryptographic Provider」があり、鍵のサイズは、それぞれ 512 ビット、1024 ビット、2048 ビットに固定されています。また、スマートカードを使用する場合の Gemplus など、その他の選択肢が表示される場合もあります。要件に応じて、サイズを選択します。1024 ビット（拡張選択肢）の使用をお薦めします。
- **有効期間 :** 証明書の有効期間を日数で指定します。ただし、ValidityRule ポリシーの「デフォルトの有効期間」に指定されている数値を使用して、Oracle Application Server Certificate Authority が自動的に設定するため、SSO ユーザーが有効期間に関する情報を入力する必要はありません。

必要事項を入力したフォームを送信すると、「証明書」フォームが表示され、証明書に記録された情報が示されます。

5. 情報が正しいことを確認した後、証明書の署名者の名前を書き留めます。この名前は後で必要です。次に、「ブラウザへのインポート」ボタンをクリックして、ブラウザに証明書をインポートします。次に示すとおり、Netscape と Internet Explorer では、インポートが成功したことを伝える方法は異なります。

---

**注意 :** 「ブラウザへのインポート」のかわりに「OK」をクリックすると、証明書が作成されて OCA リポジトリに格納され、Oracle Internet Directory に公開されます。ただし、インポートするまで、証明書はブラウザからサーバーに渡されません。「ブラウザへの新規発行の証明書のインポート」の項を参照してください。

---

- Netscape の場合は、証明書がインポートされると、ブラウザの左下のステータス・バーに「ドキュメント : 完了。」と表示されます。この時点で「OK」をクリックします。カーソルが砂時計のままでも、処理は完了しています。対応する CA（署名者）の証明書も、インポートされています。

---

**注意 :** 信頼できる証明書にするには、CA 証明書の使用方法を編集して、その認証局が発行した証明書を、ネットワーク・サイト、電子メール・ユーザー、ソフトウェア開発者のいずれかまたはすべてに対して信頼すると指定します。これらの選択肢のチェック・ボックスは、Netscape のメニュー・バーの「セキュリティ」から表示できます。「Netscape での証明書発行元への信頼」の項を参照してください。

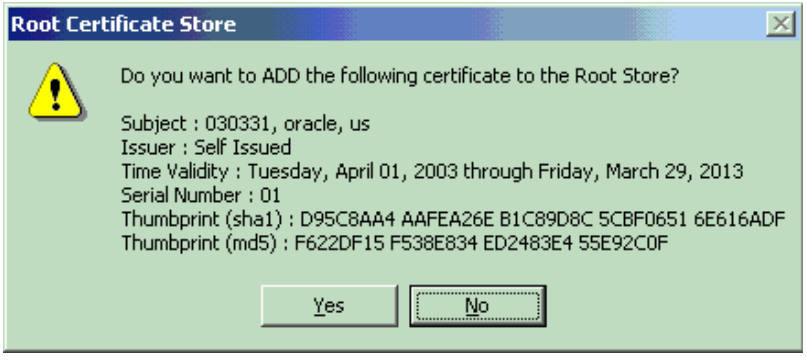
---

- Internet Explorer の場合は、証明書がインポートされると、証明書が正しくインポートされたというメッセージが表示されます。また、CA の詳細が表示されたウィンドウで、署名者の証明書をインポートするかどうかを確認するメッセージが表示されます。「OK」をクリックして、署名者の証明書もインポートすることを確認します。Internet Explorer では、これらの証明書は信頼できると自動的に判断されます。

## OracleAS Certificate Authority が信頼されるブラウザの構成

この処理は、Netscape と Internet Explorer で少し異なります。

**Internet Explorer での証明書発行元への信頼** Internet Explorer を使用して証明書をインポートする場合、その証明書をルート・ストアに追加するかどうかを確認するメッセージが表示されます。

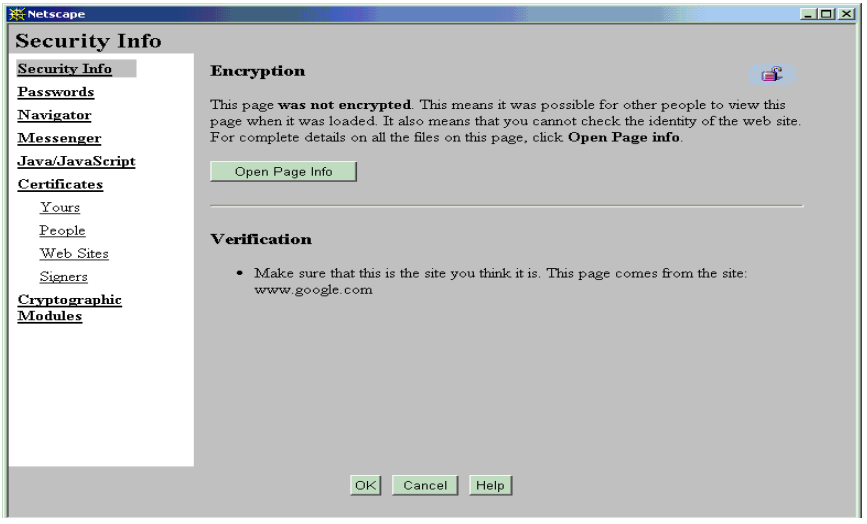



「はい」をクリックすると、証明書がインポートされ、発行元が「信頼できる」と設定されます。証明書は、メニューから「ツール」→「インターネット オプション」→「コンテンツ」→「証明書」を選択して表示できます。次に、4 つのタブが表示されます。それぞれのタブでは、ユーザー自身の証明書、他のユーザーを認証するために提供された他人の証明書、証明書を提供した中間認証局、信頼できると設定したルート認証局を参照できます。

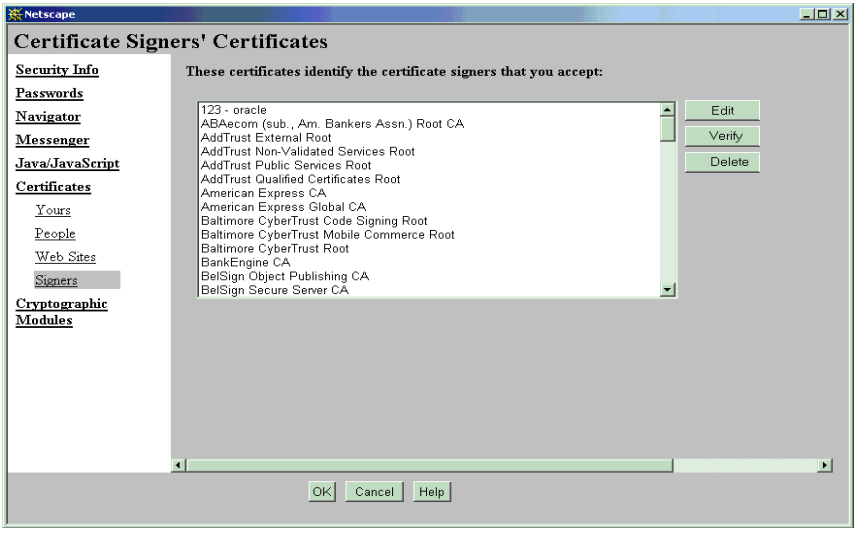
**Netscape での証明書発行元への信頼** Netscape を使用して証明書をインポートする場合、要求した証明書と、新しい CA 証明書に署名して発行した認証局を示す証明書の両方がインポートされます。通知されるのは、ブラウザの左下のステータス・バー領域に表示される「ドキュメント : 完了。」というメッセージのみです。ただし、新しい証明書が信頼できるようになるのは、署名者の証明書を信頼させるアクティビティを Netscape に対して明示的に指定した時点です。

証明書を信頼させるアクティビティを指定するには、次の手順を実行します。

1. Netscape の左下のステータス・バーにあるロック・アイコンをクリックして、「セキュリティ情報」ページを開きます（または、メニュー・バーから「Communicator」→「ツール」→「セキュリティ情報」を選択します）。次のようなページが表示されます。




2. 「署名者」リンクをクリックします。次のようなページが表示されます。



3. 証明書の詳細を参照したときに書き留めておいた署名者の名前をクリックし、「編集」をクリックします。次のようなページが表示されます。


4. 前述の画面例で選択状態になっている 3 つのチェック・ボックスを選択して、「OK」をクリックします。

これで、CA 証明書は信頼され、ブラウザで接続するネットワーク・サイト、署名または暗号化された受信メッセージ、あるいは署名されたソフトウェアの証明書を検証できます。

## Secure Sockets Layer（SSL）認証

認証局から SSL 証明書をすでに取得している場合は、現行の SSL 証明書を ID として使用して、今後の認証で使用するために、Oracle Application Server Certificate Authority 証明書を取得できます。手順は、次のとおりです。

1. 「認証」フォームの「既存の証明書の使用」オプションを選択して、「送信」をクリックします。「ユーザー証明書 - SSL」フォームが表示され、次の作業を行うことができます。

   - 証明書の取得
   - 選択した証明書の詳細の表示
   - 現行の証明書の更新
   - 現行の証明書の失効

証明書を取得するには、手順 2 ～ 5 を実行します。

2. 「ユーザー証明書 - SSL」フォームの「証明書の要求」をクリックして、「証明書要求」フォームを表示します。
3. 「証明書要求」フォームで、適切な情報を入力して、フォームを送信します。前述の「Single Sign-On（SSO）認証」の項で説明しているとおり、インタフェースは、Netscape と Internet Explorer で少し異なります。

   必要事項を入力したフォームを送信すると、「証明書」フォームが表示され、証明書に記録された情報が示されます。
4. 情報が正しいことを確認した後、「ブラウザへのインポート」ボタンをクリックして、ブラウザに証明書をインポートします。
5. 「OK」をクリックして戻ります。

## 手動認証

手動認証を使用して証明書を取得するには、次の手順を実行します。

1. 「認証」フォームの「手動承認 / 認証を使用します」を選択して、「送信」をクリックします。「ユーザー証明書」フォームが表示され、DN および連絡先情報の指定や、鍵のサイズ、使用方法および要求する証明書の有効期間の選択を行うことができます。
2. 「ユーザー証明書」フォームの「証明書の要求」をクリックして、「証明書要求」フォームを表示します。
3. 「証明書要求」フォームで、適切な DN および連絡先情報を入力した後、登録フォームのドロップダウン・リストを使用して、鍵のサイズと、SSL 証明書、暗号化証明書または署名証明書のいずれかを選択し、Oracle 認証局の管理者にフォームを送信します。

このユーザー要求に固有の要求 ID が割り当てられます。証明書が承認されると、その検索にはこの ID を使用します。

管理者の承認を受信するまでは、証明書は使用可能になりません。

証明書が承認されたことを管理者から伝えられた後、証明書の取得フォームに移動して、要求 ID または DN を使用して証明書を検索してインポートします。

# 証明書の検索、更新および失効

証明書要求が承認された後、発行された証明書を取得して、確認およびインポートできます。証明書を要求したときと同じマシンとブラウザを使用します。

SSO 証明書または SSL 証明書は、一定の期間使用した後、満了日の前後に設定した期間内に更新できます。

発行された証明書は、確認時に、対象のユーザーまたはアクティビティに対してなんらかの理由で不適切または無効な場合は、失効させることができます。

次の項で、これらの証明書の操作について説明します。

- 証明書の取得
- 証明書の更新
- 証明書の失効

## 証明書の取得

手動認証による証明書要求の承認が通知された後、証明書を確認してインポートする必要があります。証明書は、通知されたシリアル番号を「ユーザー証明書」ページの「検索」フィールドに入力して検索できます。検索後にシリアル番号の横のラジオ・ボタンをクリックして選択してから「詳細表示」をクリックすると、生成時に使用されたデータを確認できます。次に、「ブラウザへの新規発行の証明書のインポート」の項の説明に従って証明書をインポートできます。

特定の証明書に対してこれらのデータが不適切な場合は、新しい証明書を申請して、不適切な証明書を失効させ、新しい証明書に置換する必要があります。

## 証明書の更新

SSO 証明書および SSL 証明書のユーザーは、証明書を更新することができます。

ユーザーは、証明書の満了予定日の前後に設定された一定日数の間、証明書を更新できます。デフォルトでは、この期間は、証明書の満了日の前後、それぞれ 10 日間です。ただし、管理者は、Web ベースの管理インタフェースの構成タブを使用して、この期間を変更することができます。ユーザーは、証明書を選択して「詳細表示」をクリックして、証明書を更新できます。

## 証明書の失効

SSO 証明書および SSL 証明書のユーザーは、証明書を失効させることができます。

証明書にエラーまたは問題が見つかった場合や秘密鍵が盗まれた場合などには、証明書を失効させる必要があります。ユーザーは、新しい証明書に正しい情報を提供します。新しい証明書を使用すると、以前の証明書に関連する問題はすべてなくなります。

証明書を失効させると、OCA リポジトリで失効済のマークが付けられ、次回 CRL が生成されるときに CRL に追加されます。ただし、失効した証明書は、ブラウザのデータベースから自動的には削除されません。手動で削除する必要があります。Netscape の場合、ブラウザのセキュリティ・アイコンをクリックして、「証明書」の下の「本人」をクリックし、表示されたリストから失効した証明書を選択して「削除」をクリックします。

## 「サーバー / 下位 CA 証明書」タブ

どのサーバーの管理者も、サーバー証明書を取得して、他のサーバーまたはユーザーに対するそのサーバーの PKI 認証を有効にできます。そのためには、Oracle Wallet Manager（またはサード・パーティによる同等ツール）を使用して生成される PKCS #10 要求フォームが必要です。『Oracle Application Server 10*g* セキュリティ・ガイド』の Oracle Wallet Manager に関する章を参照してください。

「サーバー証明書」タブ・ページで、次の手順を実行します。

1. ホームページで「サーバー / 下位 CA 証明書」タブを選択して、「サーバー証明書」フォームを表示します。
2. 「証明書の要求」ボタンをクリックします。
3. 「サーバー / 下位 CA 証明書要求」フォームに、Oracle Wallet Manager が以前生成した、必要事項が入力された PKCS #10 要求フォームを貼り付けて、必要な証明書のタイプを選択します。SSL、暗号化、署名、コード署名または CA 署名のサーバー証明書を要求できます。下位 CA として使用するには、登録フォームの証明書の使用方法に「CA 署名」を指定します。表示されるドロップダウン・リストの選択肢から、要求した証明書の有効期間も選択します。
4. 適切な情報を入力し、管理者にフォームを送信します。

管理者がこの要求を承認するまで、サーバーの管理者は認証を取得できません。

## 下位 CA 証明書

1 つの企業内で異なる大陸に部門が存在するなど、単一の CA では実用的でない場合は、PKI 構造内に複数の CA を保持することができます。階層的な PKI では、ルート CA は、すべてのユーザーによって信頼されている単一の CA です。ルート CA の公開鍵は、セキュリティ・ドメインに対する信頼できるパスの開始位置として機能します。

Oracle Application Server Certificate Authority は、ルート CA となる場合と、第三者 CA から下位 CA 証明書を取得する場合があります。Oracle Application Server Certificate Authority が、別の CA の証明書署名を認証することで、下位 CA を作成することもできます。下位 CA がさらに下位レベルの CA に対して証明書を発行する場合、いわゆる証明連鎖が生まれます。いずれかの下位 CA が署名した個々の証明書には、ルート CA までのすべての CA の証明書が表示されている必要があります。それぞれの認証局の証明書は上位の CA によって署名されているため、特定の証明書の妥当性を検証するには、認証局のパスをルート CA までトレースします。

下位 CA 証明書を取得するには、次の手順を実行します。

1. ホームページで「サーバー / 下位 CA 証明書」タブを選択して、「下位 CA 証明書」フォームを表示します。
2. 「証明書の要求」ボタンをクリックします。
3. 「下位 CA 証明書要求」フォームで適切な情報を入力した後、証明書の使用方法のタイプに「CA 署名」を選択して、管理者にフォームを送信します。

要求者は、管理者がこの要求を承認するまで、証明書を取得できません。

# CA 証明書のダウンロード

Netscape の場合、「証明書の要求」をクリックすると、Oracle Application Server Certificate Authority によって、一連のダイアログ・ボックスが表示されます。これらのダイアログ・ボックスには、OCA 証明書の受付けに必要な操作の説明が表示されます。表示されるそれぞれのダイアログ・ボックスで「次へ」をクリックし、最後のダイアログ・ボックスでは「終了」をクリックします。CA 証明書がブラウザに自動的にダウンロードされます。

Internet Explorer の場合は、CA 証明書のインポートを承認するか拒否するかを確認するメッセージが表示されるだけです。証明書を取得しない場合でも、この CA が証明書を発行しているサーバーを信頼するためだけに承認できます。ブラウザによって、証明書を保存するかどうか、または現在の位置から開くかどうかを確認するメッセージが表示されます。ブラウザに CA 証明書をインポートする場合は、「このファイルを上記の場所から開く」を選択して「OK」をクリックします。次に表示されるウィンドウで「証明書のインストール」を選択し、証明書のインポートを承認して、ブラウザのリポジトリに CA 証明書を格納します。

# ブラウザへの証明書失効リスト（CRL）のインポート

ブラウザに証明書失効リスト（CRL）をインポートすると、個人または企業から提供された証明書が失効している場合に警告を表示できます。失効した証明書を使用すると、偽装の問題がある可能性、あるいは提供または使用している製品に問題がある可能性が示される場合があります。警告が表示されることによって、不適切である可能性がある操作を回避できます。

この操作は、ブラウザによって異なる手順が必要になります。

- Netscape の場合
- Internet Explorer（IE）の場合

## Netscape の場合

Oracle Application Server Certificate Authority の「ユーザー証明書」タブから、次の操作を実行します。

1. 「CRL のダウンロード」ボタンをクリックします。
   「CRL のダウンロード」フォームが表示されます。
2. 「ブラウザへの CRL のインポート」をクリックします。
   CRL がインポートされます。

CRL は、「セキュリティ」→「署名者」→「CRL の表示 / 編集」で参照できます。

すでに CRL を取得していて、その CRL がダウンロード中の CRL 以降まで有効の場合は、ダウンロードしようとしている CRL が、ブラウザにすでに存在する CRL より有効期間が短いことを示す小さなダイアログ・ボックスが表示されます。

また、次の操作を実行することもできます。

- 「CRL のバイナリでのダウンロード」というラベルが付いたボタンをクリックして、格納するディレクトリを選択し、CRL のバイナリ・コピーをダウンロードする。
- 「Base64 形式での CRL のダウンロード」というラベルが付いたボタンをクリックして、ダウンロード先のディレクトリを選択し、BASE64 形式でコピーをダウンロードする。

## Internet Explorer（IE）の場合

IE の場合、CRL はブラウザに直接インポートされません。CA 証明書をインポートする場合と同様、IE では、「ディスクに保存する」または「このファイルを上記の場所から開く」のどちらを選択するかを確認するメッセージが表示されます。後者を選択すると、CRL はインポートされません。「ディスクに保存する」を選択した場合は、次の操作を実行します。

1. 「ツール」メニューから、「インターネット オプション」→「コンテンツ」→「証明書」を選択します。
2. 「インポート」を選択します。
3. CRL を選択します。

CRL がインポートされたというメッセージが表示されます。

# ファイル・システムへの証明書失効リスト（CRL）のダウンロード

ファイル・システムに証明書失効リスト（CRL）をダウンロードすると、他のプログラムはそれを使用して、個人または企業から提出された証明書のうち、失効しているものや期限切れのものを検出できます。こうした証明書の使用を回避することで、不適切なユーザーや不正なユーザーから、リソースおよびアプリケーションを保護できます。

CRL をダウンロードするには、次の手順を実行します。

1. Oracle Application Server Certificate Authority の「ユーザー証明書」ページに進みます。
2. 「CRL のダウンロード」をクリックします。
3. 「CRL のバイナリでのダウンロード」または「Base64 形式での CRL のダウンロード」をクリックします。
4. 選択したディレクトリに CRL を保存します。
5. $ORACLE_HOME/apache/apache/conf にある http.conf ファイルを変更し、SSLCARevocationFilePath パラメータを入れ、新しい CRL ファイルの入っているディレクトリをそのパラメータが指すようにします。

# ブラウザへの新規発行の証明書のインポート

証明書に対する要求が承認された後、Oracle Application Server Certificate Authority では、新しいウィンドウに詳細が表示されます。これにより、詳細が意図する内容と一致しているかどうかを確認できます。証明書の名前や有効期間などの属性が適切かどうかを確認します。詳細に重大なエラーがある場合は、この証明書を失効させて、要求フォームに正しい情報を指定して新しい証明書を申請します。

確認後、「証明書のインポート」ボタンをクリックして、ブラウザに証明書のコピーをインポートします。ブラウザの左下のステータス・バー領域に、「ドキュメント : 完了。」というメッセージが表示されます。次に「OK」をクリックします。

「証明書のインポート」をクリックしないで「OK」のみをクリックすると、サーバーには証明書のコピーが保持されますが、ブラウザには保持されません。このため、アプリケーション、ディレクトリまたは別のサーバーに対する認証が必要な場合は、証明書を提供できません。

証明書のインポート処理では、ルート CA までの CA の連鎖もインポートされます。ただし、Netscape では、ユーザー証明書とともにインポートされた CA 証明書は自動的には信頼されません。次の手順を実行して、信頼を確立する必要があります。

- Netscape の場合

  1. セキュリティ・アイコンをクリックします。
  2. 「署名者」リポジトリを選択します。
  3. 適切な CA 証明書の名前を選択します（リポジトリ・パスワードを入力するように求められる場合があります）。
  4. CA 証明書を編集します。
  5. 適切なチェック・ボックスを選択します。
  6. 「OK」を選択します。

  この手順では適切な信頼関係が確立されるため、SSL セッションを確立するときに、インポートする証明書が発行した証明書が Netscape ブラウザで信頼されます。

# ブラウザからの Wallet のエクスポート（バックアップ）

システムまたはブラウザが破壊された場合にリストアできるように、Wallet をエクスポートしてファイル・システムに保管してください。Wallet には、証明書、秘密鍵、および証明書を発行した信頼できる認証局に対する証明書の連鎖が含まれます。

Netscape 4.7x または 4.8 の場合は、次の手順で証明書をエクスポートします。

1. メニュー・バーでセキュリティ・アイコンを選択します。

   ウィンドウが表示され、セキュリティ情報を確認するための選択肢が表示されます。

2. 「証明書」の下の「本人」をクリックします。

   従属するウィンドウが表示され、証明書の名前が表示されます。

3. エクスポートする証明書をクリックします。
4. 従属ウィンドウの右側にある「エクスポート」ボタンをクリックします。
5. メッセージが表示されたら、パスワードを入力して、この証明書のセキュリティおよび整合性を保持します。パスワードは、2 回入力するように求められます。入力した内容が一致している必要があります。

   通常どおり、このパスワードは、この証明書を検索または再インストールするために書き留めておきます。パスワードがないと、証明書は使用できません。

6. メッセージが表示されたら、暗号化された証明書の格納先のファイル・システム、パス名およびファイル名を入力します。

   証明書が正常にエクスポートされたというメッセージが表示されます。

Netscape 7.x または Mozilla 1.5 の場合は、次の手順で証明書をエクスポートします。

1. 「編集」メニューの「設定」をクリックします。「設定」ウィンドウが表示されます。
2. 「設定」ウィンドウで「プライバシーとセキュリティ」オプションを開き、「証明書」をクリックします。
3. 「証明書の管理」（右側画面）をクリックして、「証明書マネージャ」ウィンドウを表示します。
4. エクスポートが必要な証明書を選択して「バックアップ」をクリックします。
5. PKCS #12 Wallet のファイル名を入力して「保存」をクリックします。
6. Netscape リポジトリのパスワードを入力して「OK」をクリックします。

   ウィンドウが開き、「セキュリティ デバイスのマスター パスワードを入力してください」というプロンプトが表示されます。正しいパスワード（ブラウザのリポジトリのパスワード）を入力すると、新しいウィンドウが表示されます。
7. この「証明書バックアップ パスワードの選択」ウィンドウで、PKCS #12 Wallet の暗号化に使用するパスワードを入力します。確認のため、同じパスワードをもう一度入力する必要があります。このウィンドウには、パスワードの質に関する情報を示すパスワード品質メーターがあります。
8. 「OK」をクリックします。アラートが表示され、バックアップが正常に終了したことが伝えられます。

Internet Explorer の場合は、次の手順で証明書をエクスポートします。

1. 「ツール」メニューから、「インターネット オプション」を選択します。

   ウィンドウが表示され、選択可能な 6 つのタブが示されます。
2. 「コンテンツ」タブを選択して、「証明書」ボタンをクリックします。

   「証明書マネージャ」ウィンドウが表示され、4 つのタブが示されます。これらのタブで、ユーザー自身の証明書、他人の証明書、信頼できる中間証明機関の名前および有効期限を参照できます。
3. 「個人」タブでは、エクスポートする特定の証明書をクリックします。
4. ウィンドウの下の「エクスポート」ボタンをクリックします。
5. 「証明書のエクスポート ウィザード」で「次へ」をクリックします。
6. 秘密鍵をエクスポートする場合は、「はい」ラジオ・ボタンをクリックします（秘密鍵をエクスポートしない場合は、「いいえ」ラジオ・ボタンをクリックします）。「はい」をクリックすると、秘密鍵も格納されます。
7. 「次へ」をクリックします。

8. PKCS #12 を選択し、その下にある 2 つのチェック・ボックスを選択して、「次へ」をクリックします。
9. メッセージが表示されたら、パスワードを入力して、秘密鍵のセキュリティを保持します。パスワードは、2 回入力するように求められます。入力した内容が一致している必要があります。

   通常どおり、このパスワードは、この秘密鍵を検索または再利用するために記録しておきます。パスワードがないと、秘密鍵は使用できません。
10. メッセージが表示されたら、暗号化された証明書および鍵の格納先のファイル・システム、パス名およびファイル名を入力します。
11. 新しいウィンドウに、選択した項目が表示されます。この情報を確認した後、「完了」をクリックします。

    「エクスポートが完了しました。」というメッセージが表示されます。
12. 「OK」→「閉じる」→「OK」をクリックして、この処理で使用したウィンドウを終了します。

# ファイル・システムからの証明書のインポート

ファイル・システムに格納されているファイルから、ブラウザに証明書をインポートすることができます。ファイルのタイプは、拡張子が .p12 の PKCS #12 である必要があります。ブラウザに証明書をインポートするには、Wallet の暗号化に使用したパスワードが必要です。手順は、次のとおりです。

Netscape 4.7x または 4.8 の場合は、次の手順で証明書をインポートします。

1. メニュー・バー（または、下部のステータス・バー）でセキュリティ・アイコンを選択します。
2. 「証明書」の下の、「本人」リンクをクリックします。リストおよびいくつかのボタンが表示されます。
3. 「証明書のインポート ...」をクリックします。
4. Wallet および目的の証明書が含まれるディレクトリに移動して、.p12 ファイルをダブルクリックします。

   Wallet のパスワードの入力を求めるダイアログ・ボックスが表示されます。
5. パスワードを入力します（入力したパスワードが不適切な場合は、復号化に失敗したため、ファイルが破壊されているか、または無効であるというメッセージが表示されます）。証明書がすでにインポートされている場合は、Netscape によって通知され、この要求の処理は続行されません。
6. 証明書が正常にインポートされた後、「OK」をクリックします。

Netscape 7x または Mozilla 1.5 の場合は、次の手順で PKCS #12 Wallet から証明書をインポートします。

1. ブラウザの「編集」メニューの「設定」をクリックします。「設定」ウィンドウが表示されます。
2. 「設定」ウィンドウで「プライバシーとセキュリティ」オプションを開き、「証明書」をクリックします。
3. 「証明書の管理 ...」（右側画面）をクリックして、「証明書マネージャ」ウィンドウを表示します。
4. 「インポート」をクリックします。
5. インポートする証明書と鍵が入った PKCS #12 Wallet を選択して、「開く」をクリックします。
6. 表示されるポップアップ・ウィンドウで、Netscape リポジトリのパスワードを入力して「OK」をクリックします。

   ウィンドウが開き、「セキュリティ デバイスのマスター パスワードを入力してください」というプロンプトが表示されます。パスワードを入力すると、「パスワード入力ダイアログ」という新しいウィンドウが表示されます。
7. この新しいウィンドウで、PKCS #12 Wallet の復号化に使用するパスワードを入力して「OK」をクリックします。
8. アラートが表示され、証明書と秘密鍵のリストアが正常に終了したことが伝えられます。

Internet Explorer（IE）の場合は、次の手順で PKCS #12 Wallet から証明書をインポートします。

1. 「ツール」メニューから、「インターネット オプション」を選択します。

   ウィンドウが表示され、選択可能な 6 つのタブが示されます。
2. 「コンテンツ」タブを選択して、「証明書」ボタンをクリックします。「個人」タブには、ユーザー自身の証明書が表示されます。
3. 「インポート」をクリックします。「証明書のインポート ウィザード」ウィンドウが表示されます。
4. 「次へ」をクリックした後「参照」をクリックして、目的の証明書が含まれているディレクトリを選択します。
5. ダブルクリックしてフルパスをウィザードに入力し、「次へ」をクリックします。
6. 選択した Wallet のパスワードを入力します。
7. 「次へ」をクリックします。

8. Internet Explorer では、証明書のタイプに基づいて自動的に証明書ストアを選択することも、その他のラジオ・ボタンをクリックして、証明書ストアへのパスを入力し、証明書の格納先を指定することもできます。
9. 「次へ」をクリックします。
10. 「完了」をクリックします。

    IE によって使用される証明書ストアに、証明書を発行した CA の証明書が含まれていない場合は、証明書をストアに追加するかどうかを確認するダイアログ・ボックスが表示されます。
11. 「はい」をクリックします。この証明書があると、この CA（または同じ信頼連鎖の他の認証局）によって証明書が発行されているその他のサーバーまたはユーザーを認証できます。

    ダイアログ・ボックスが表示され、インポートが正常に終了したことが伝えられます。
12. 「閉じる」および「OK」をクリックして、IE の証明書およびセキュリティ領域を終了します。

# A

# コマンドライン管理

この付録は、Oracle Application Server Certificate Authority コマンドライン・ツール `ocactl` から使用可能なコマンドおよびオプションへのクイック・ヘルプ・リファレンスです。これらのコマンドの詳細な使用方法については、ユースケースとあわせて第 6 章を参照してください。

この付録では、管理コマンドライン・ツール `ocactl` を使用した Oracle Application Server Certificate Authority 管理タスクの実行方法、および Oracle Application Server Certificate Authority をホストするコンピュータを介した操作方法について説明します。

この付録の内容は、次のとおりです。

**表 A-1 コマンドおよび構成操作へのリンク**

| 一般的な操作 | 参照先 |
|---|---|
| 基本的な管理：<br>コマンドおよび操作 | ■ コマンドライン・ツール<br>■ Oracle Certificate Authority Server の起動<br>■ Oracle Application Server Certificate Authority Server の停止<br>■ Oracle Application Server Certificate Authority サービスの状態の検索<br>■ 権限付きパスワードの変更<br>■ OCA リポジトリ接続情報の更新 |
| ルート証明操作 | ■ ルート認証局の証明書の再生成<br>■ ルート CA 証明書の失効 |
| SSL/SSO 操作 | ■ CA SSL サーバー Wallet の SSO 形式への変換<br>■ 認証局の SSL 証明書および Wallet の再生成<br>■ SSO 認証の設定（linksso および unlinksso コマンド） |
| 下位 CA 操作 | ■ Oracle Application Server Certificate Authority からの下位 CA Wallet の生成<br>■ 下位 CA Wallet のインストール / インポート<br>■ 下位 CA 用の CA SSL Wallet の生成 |
| ログ / トレース操作 | ■ ログ / トレース・オプションの設定<br>■ ログまたはトレース記憶域の消去 |

# コマンドライン・ツール

OCA 管理者は、コマンドライン・ツール `ocactl` を使用して、Oracle Application Server Certificate Authority の様々な操作に必要なパラメータを指定します（パスに `oca/bin` を追加することが必要になる場合もあります）。このツールを起動するたびに、OCA 管理者のパスワードの入力が求められます。このパスワードは CA 署名のパスワードと常に同一です（パスワードの入力時に、低速な telnet/rlogin セッションおよび [Back Space] を使用した場合は、パスワードの一部が表示されます）。

このコマンドの一般的な形式は次のとおりです。

```
ocactl < 操作名 > -type < 存在する場合は関連パラメータ名 >
```

たとえば、Oracle Application Server Certificate Authority を起動する場合は、次のコマンドを入力します。

```
ocactl start
```

別の例として、Oracle Application Server Certificate Authority と Oracle Internet Directory 間で相互認証を行う証明書を公開する場合に、CASSL 操作の証明書および Wallet を生成するには、次のコマンドを入力します。

```
ocactl generatewallet -type CASSL
```

パラメータを含まないコマンドがあることに注意してください。パラメータを使用しないコマンドでは、キーワード「-type」も使用しません。

パラメータを必要とするコマンドでは、パラメータの前にキーワード `-type` を使用する必要があります。

唯一の例外は `convertwallet` コマンドで、表 A-2 に示すように、特別な構文になっています。

表 A-2 に、主な操作（アルファベット順）および関連パラメータを示します。その表の後で、`convertwallet` コマンドの追加パラメータについて説明します。

次の操作は、表に直接リンクされています。

changeschema、changesecurity、clear、generatewallet、help、importwallet、linksso、renewcert、revokecert、set、setpasswd、start、stop、unlinksso、updateconnection

**表 A-2 OracleAS Certificate Authority（OCA）ocactl ツールの操作およびパラメータ**

| 操作 | パラメータ | 意味 |
|---|---|---|
| changeschema | `-host hostname`<br>`-service service` | データベース全体を別のデータベースに変更し、そのデータを新しいデータベースに移行するときに使用します。<br>`hostname` は、新しいマシンの名前です。<br>`service` は、そのマシン上のサービスの名前です。 |
| changesecurity | -server_auth_port port | OCA が使用する Identity Management サービス（OID/SSO）を、新しい OID および SSO Server に変更します。<br>新しい IM マシンおよびポート番号で oca.conf を更新し、OCA を新しい SSO に登録するときに指定されたポートを使用します。 |
| clear | LOG、TRACE<br>OCA または ADMIN | 選択したログまたはトレース・データのタイプ（OCA または ADMIN）に対し、前述の `set` コマンドで指定した格納場所（ファイルまたはデータベース表）を消去します（OCA を実行中でない場合は、該当するデータがすべて消去されます）。<br>各コマンドの例は、第 6 章「OracleAS Certificate Authority の管理 : 高度なトピック」の「Oracle Application Server Certificate Authority での OCA アクションのログまたはトレース」の項を参照してください。 |
| convertwallet | 後述の説明を参照 | この表の後の「Convertwallet の例」の項を参照。 |

**表 A-2 OracleAS Certificate Authority（OCA）ocactl ツールの操作およびパラメータ（続き）**

| 操作 | パラメータ | 意味 |
|---|---|---|
| generatewallet | CA、<br>CASSL、<br>または<br>CASMIME | 指定されたタイプの証明書および Wallet を生成します。このタイプには、認証局署名証明書と認証局 SSL 証明書があります。<br>generatewallet コマンドの例を示します。<br>`ocactl generatewallet -type CASSL`<br>次のタイプの Wallet が、指定された場所に格納されます。<br>CA　Oracle Application Server Certificate Authority リポジトリ<br>CASSL　`$ORACLE_HOME/oca/wallet/ssl`<br>CASMIME　Oracle Application Server Certificate Authority リポジトリ |
| help | < コマンド名 > | コマンド名を指定するとコマンドの構文が表示されます。<br>help コマンドの例を示します。<br>`ocactl help setconfig` |
| importwallet | SUBCA | Wallet を格納するディレクトリおよび管理者のパスワードの入力を要求した後、このコマンドは ewallet.p12 という Wallet をインストールします。これは、下位 CA サーバーの Wallet です。<br>importwallet コマンドの例を示します。<br>`ocactl importwallet -type SUBCA` |
| linksso | < なし > | OCA を SSO に登録し、証明書を所有していない SSO ユーザーが証明書を要求できるように OCA 証明書の登録フォームを表示します。<br>（このコマンドでは OCA サービスを停止する必要はありません。ただし、SSO Server を再起動するまでこのサービスは有効になりません。） |
| renewcert | CA、<br>CASSL、<br>CASMIME | OCA が実行中でない場合にこのコマンドを使用すると、新しい有効期間（日数）を入力するプロンプトが表示され、管理者は指定した証明書を更新できます。<br>renewcert コマンドの例を示します。<br>`ocactl renewcert -type CA` |
| revokecert<br>（CA を失効させると、インストールしてある OCA が動作しなくなります。） | CA<br>WEBADMIN<br>（この操作には、注意と確認が必要です。） | OCA が動作中でない場合しか使用できません。ルート CA 証明書を失効させます。CA パラメータで指定可能なその他の理由コードについては、「ルート CA 証明書の失効」の項を参照してください。<br>revokecert コマンドの例を示します。<br>`ocactl revokecert -type CA -reason SUPERSEDED`<br>失効理由の詳細は、表 A-5 を参照してください。 |

**表 A-2 OracleAS Certificate Authority（OCA）ocactl ツールの操作およびパラメータ（続き）**

| 操作 | パラメータ | 意味 |
|---|---|---|
| set | LOG または TRACE<br>ON または OFF<br>OCA または ADMIN | OCA 構成を設定して、LOG または TRACE の後に指定した状態（ON または OFF）あるいはモード（OCA または ADMIN）の追加パラメータを使用します。手順は次のとおりです。<br>各コマンドの例は、第 6 章「OracleAS Certificate Authority の管理 : 高度なトピック」の「Oracle Application Server Certificate Authority での OCA アクションのログまたはトレース」の項を参照してください。 |
| setpasswd | CA、<br>DB、<br>CASSL、<br>または<br>CASMIME | 指定したロール（管理者、データベース管理者、ディレクトリ、OCA ユーザーまたは認証局 SSL サーバー）のパスワードを要求および再設定します。証明書の生成および使用方法に関するパスワードの使用、設定および格納については、このマニュアルの該当する項を参照してください。<br>setpasswd コマンドの例を示します。<br>`ocactl setpasswd -type DB` |
| start | < パラメータなし > | Oracle Application Server Certificate Authority サービスを起動します。（OCA を起動するには、OC4J、OHS およびデータベースが実行されている必要があります。OC4J および OHS は、コマンドライン・ツール opmn で制御します。）<br>start コマンドの例を示します。<br>`ocactl start` |
| status | < パラメータなし > | Oracle Application Server Certificate Authority サービスの状態を表示します。<br>status コマンドの例を示します。<br>`ocactl status` |

**表 A-2 OracleAS Certificate Authority（OCA）ocactl ツールの操作およびパラメータ（続き）**

| 操作 | パラメータ | 意味 |
|---|---|---|
| stop | < パラメータなし > | Oracle Application Server Certificate Authority サービスを停止します。<br>（データベース、Web サーバー、OracleAS は停止されません。データベース接続プールが停止され、ログ出力、トレース出力および構成データ・ファイルが閉じられます。）<br>stop コマンドの例を示します。<br>`ocactl stop` |
| unlinksso | < なし > | SSO から OCA を登録解除します。「ようこそ」画面および登録フォーム画面は表示されなくなります。<br>（このコマンドでは OCA サービスを停止する必要はありません。ただし、SSO Server を再起動するまでこのサービスは有効になりません。） |
| updateconnection | < パラメータなし > | Oracle Internet Directory（OID）に格納されている接続情報を、OCA 構成ファイル $ORACLE_HOME/oca/conf/oca.conf に書き込みます。<br>これらの文字列は、OCA リポジトリへの接続および（証明書の公開に使用する）ディレクトリへの接続に使用されます。<br>（この接続情報は、Oracle Application Server Certificate Authority 管理者用の Web ベースのインタフェースの「一般」サブタブにある設定セクションに表示されます。）<br>OCA の接続情報は、最初に OracleAS のインストール時に OID に書き込まれます。この接続情報は OID からフェッチされ、oca.conf にも書き込まれます。OCA が別のデータベースに移動したり、構成情報が変更されると、この接続情報は変更されます。その例として、RAC 対応データベースでの RAC ノードの追加や削除など、接続文字列内のノードやポートの変更があります（データの移行は必要ありません。ポートを変更する場合は、『Oracle Application Server 10*g* 管理者ガイド』の「インフラストラクチャ・ポートの変更」で説明している手順に従ってください）。<br>注意 : 構成の設定を変更したら、その後で ocactl updateconnection を実行する必要があります。また、このコマンドの使用後に、次のコマンドを発行して OCA を再起動する必要があります。<br>$ORACLE_HOME/oca/bin/ocactl stop<br>$ORACLE_HOME/oca/bin/ocactl start |

## Convertwallet の例

表 A-2 では、ocactl を使用して発行できる大部分のコマンドの例を示しています。ただし、convertwallet コマンドは異なる構文を使用します。構文については、この項で例を示して説明します。

convertwallet コマンドを使用して、SSL サーバーの Wallet（PKCS #12 形式の ewallet.p12）を、SSO 形式の Wallet（ファイル名 cwallet.sso）に変換します。

cwallet.sso を使用するメリットは、Wallet のパスワードを指定しなくても、HTTP Server を SSL モードで起動できることです。通常、このパスワードは、PKCS #12 Wallet を使用して HTTP Server を SSL モードで起動する際に要求されます。

SSO 形式の Wallet は暗号化されており、ユーザーはファイルを開いたり鍵を抽出することができません。ただし、この Wallet は所有者権限のみで作成されるため、Wallet を保護するには、オペレーティング・システムのファイル権限が必要です。

つまり、convertwallet コマンドを使用すると、Wallet パスワードを要求することなく、SSO（シングル・サインオン）で自動的に Web サーバーを SSL モードで起動できます。convertwallet の構文は次のとおりです。

```
convertwallet -format SSO [-walletwrl <wallet-location>]
```

次に例を示します。

```
convertwallet -format SSO -walletwrl $ORACLE_HOME/wallets
```

オプションのパラメータ -walletwrl は、CA SSL PKCS #12 Wallet が、ファイル名 ewallet.p12 で格納されているディレクトリを指定する次のパラメータを識別します。

-walletwrl を指定すると、ocactl は、OCA によって作成されていない（OCA 以外から取得された）CA SSL Wallet を管理者が変換しようとしているとみなします。OCA のパスワード・ストアにはパスワードが含まれないため、管理者は元の CA SSL Wallet のパスワードを指定して、指定したディレクトリの Wallet を読み取る必要があります。Wallet が開かれた後、証明書は .sso 形式に変換され、-walletwrl <wallet-location> で指定した同一の場所に戻されます。

-walletwrl を指定しない場合、ocactl は、この Wallet を OCA のインストール時に OCA によって生成された CA SSL Wallet であるとみなします。そのため、このコマンドは、ocactl コマンドの使用を有効にする際に指定した OCA 管理者のパスワードを使用して、CA SSL パスワードを含む内部パスワード・ストアを開きます。その後、このパスワードを使用し、CA SSL Wallet（$ORACLE_HOME/oca/wallet/ssl ディレクトリに格納）を開いて変換します。

Wallet の格納先を示す <wlt-location> を指定しない場合、デフォルトでこの Wallet は $ORACLE_HOME/oca/wallet/ssl（またはインストール時に指定した場所）に格納されます。

Oracle Application Server Certificate Authority は、OHS、OCA の OC4J および Oracle Application Server Certificate Authority がこの順序で再起動された後にのみ、$ORACLE_HOME/oca/wallet/ssl/ に格納されている新しい SSO Wallet を使用します（必要なインフラストラクチャを起動する手順は、『Oracle Application Server 10*g* 管理者ガイド』の第 4.1 項を参照してください。OHS、OC4J などの中間層コンポーネントを起動する方法については、第 4.2 項を参照してください）。

# Oracle Certificate Authority Server の起動

OC4J、OHS およびデータベースを起動した後、管理者およびユーザーがアクセスするフォームをサポートする Oracle Application Server Certificate Authority サービスを起動できます。OC4J および OHS を起動するには、コマンドライン・ツール opmnctl を次のように使用します。

```
$ORACLE_HOME/opmn/bin/opmnctl startproc type=oc4j instancename=oca
$ORACLE_HOME/opmn/bin/opmnctl startproc type=ohs
```

Oracle Application Server Certificate Authority を起動するには、次のコマンドを発行します。

```
ocactl start
```

このコマンドには管理者のパスワードが必要です。コマンドを実行すると、Oracle Application Server Certificate Authority エンジンが起動して、データベース接続プール、ログ出力ファイル、トレース出力ファイルおよび XML 構成ファイルが作成されます。

# Oracle Application Server Certificate Authority Server の停止

stop コマンドを実行すると、Oracle Application Server Certificate Authority サービスが停止します。その他のサービスには影響はありません。つまり、データベース、OracleAS および Web サーバーは変更されません。

Oracle Application Server Certificate Authority サービスを停止するには、次のコマンドを発行します。

```
ocactl stop
```

## Oracle Application Server Certificate Authority サービスの状態の検索

status コマンドを発行すると、Oracle Application Server Certificate Authority サービスの状態を表示できます。このコマンドには管理者のパスワードが必要です。コマンドを実行すると、Oracle Application Server Certificate Authority エンジンに問合せが行われます。レスポンスには、データベース接続プール、ログ出力、トレース出力、XML の各ファイルおよびパスワード・ストアの開閉状態が表示されます。

Oracle Application Server Certificate Authority サービスの状態を取得するには、次のコマンドを発行します。

```
ocactl status
```

## 権限付きパスワードの変更

インストール時に、Oracle Application Server Certificate Authority のパスワード・ストアが作成されます。これには、OCA 操作に必要な初期パスワードが格納されています。

**表 A-3 パスワードのタイプおよび使用方法**

| パスワードのタイプ | パスワードの使用方法 |
|---|---|
| OCA データベース・ユーザー | OCA 情報を含むデータベース表にアクセスできます。 |
| CA SSL | 認証局が SSL を使用して通信できます。また、Oracle Wallet Manager がこの Wallet にアクセスして、トラスト・ポイントを追加することもできます。インストール時に、ランダムに選択されたパスワードを使用して Wallet を暗号化します。このパスワードを使用して、Wallet に既知のパスワードを設定すると、この Wallet を Oracle Wallet Manager で開くことができます。 |

このパスワード・ストアの情報は、CA 署名パスワードでもある OCA 管理者のパスワードを使用して暗号化されます。OCA 管理者のパスワードは、パスワード・ストアに格納されません。

インストール後、様々な管理者の権限付き操作のパスワードを変更できます。次に示すように、ocactl ツールで setpasswd コマンドを使用します。

**表 A-4 権限付きロールおよび setpasswd コマンド**

| 権限付きロール | ロールのパスワードの変更コマンド | 新しいパスワードのその他の使用方法 |
|---|---|---|
| Oracle Application Server Certificate Authority 管理者 | `ocactl setpasswd -type CA` | 認証局署名パスワード |
| CA SSL サーバー | `ocactl setpasswd -type CASSL` | CA SSL Wallet パスワード |
| Oracle Application Server Certificate Authority データベース管理者 | `ocactl setpasswd -type DB` | OCA がデータベースへのログインに使用する、データベース内の DB 用パスワード |
| 管理者の署名通知メール | `ocactl setpasswd -type CASMIME` | パスワード・ストア内の CASMIME パスワード |

データベース（DB）パスワード以外のパスワードはすべて、Oracle Application Server Certificate Authority が動作中の場合でも変更できます。既存の DB パスワードを使用して CA への接続をアクティブにすると、Oracle Application Server Certificate Authority が停止するまで DB パスワードを変更できません。Oracle Application Server Certificate Authority の停止中に DB パスワードを変更した後、Oracle Application Server Certificate Authority を起動するために新しいパスワードが必要になります。このパスワードは操作中にも使用します。

これらのコマンドを実行して変更された内容は、次回、Oracle Application Server Certificate Authority を起動したときに有効になります。Certificate Authority が再起動されるまで、新しいパスワードは使用されません。以前のパスワード情報はメモリーに格納されているため、参照された Wallet（古いパスワードで暗号化）は使用可能です。Oracle Application Server Certificate Authority を再起動した後、新しいパスワードが有効になります。

`ocactl` を使用するたびに、OCA 管理者のパスワードが必要になります。このパスワードが認証されると、コマンドで指定したロール・タイプの新しいパスワードの入力が要求され、パスワード・ストアのパスワードと置換されます。この結果は、OCA 管理者の最新のパスワードを使用して再度暗号化されます。

# ルート認証局の証明書の再生成

Oracle Application Server Certificate Authority をルート認証局（CA）としてインストールすると、ルート CA 証明書および Wallet が作成されます。CA 鍵が危殆化した場合は、`ocactl` 管理コマンドライン・ツールを使用して、この証明書を再生成できます。新しい CA 証明書および秘密鍵は OCA リポジトリに格納されます。この秘密鍵は、生成中に要求されたパスワードによって暗号化されます。

以前の CA 署名証明書のエントリ、および以前の CA 署名証明書が発行したその他のすべての証明書は無効になります。CA SSL、CA SMIME などの重要な Wallet は、再生成する必要があります。CA Wallet を再生成した後、以前の CA が発行した CRL は無効になります。

> **関連項目：** 第 6 章「OracleAS Certificate Authority の管理 : 高度なトピック」。特に、「CA 署名 Wallet の再生成」および「CA SSL Wallet および CA SMIME Wallet の再生成」の項。

この操作が実行できるのは、OCA が正常にインストールされた後で、OCA サービスが実行されていない場合だけです。

ルート CA Wallet を生成できるのは、OCA を実行していない場合だけです。OCA を実行中の場合は、OCA を停止し、次のコマンドを使用してルート CA Wallet を再生成します。

```
ocactl generatewallet -type CA
```

この証明書はバイナリ・ファイルとして、ディレクトリ $ORACLE_HOME/oca/wallet/ca に格納されます。

署名鍵は、ディレクトリ $ORACLE_HOME/oca/wallet/ca に格納され、OCA 管理者のパスワードによって暗号化されます。

パスワード・ストアは、管理者のパスワードで暗号化されて、ディレクトリ $ORACLE_HOME/oca/pwdstore に保持されます。DB パスワードは、最初は管理者のパスワードと同じです。OCA ではリポジトリにアクセスする際に DB パスワードを使用します。管理者は次のコマンドを使用して DB パスワードを変更できます。

ocactl setpasswd -type DB

# 認証局の SSL 証明書および Wallet の再生成

CA SSL 証明書および Wallet はインストール時に生成され、Oracle Application Server Certificate Authority のエンジンが HTTPS モードでリスニングするために使用されます。これらが危殆化または破壊された場合、あるいは CA Wallet が再生成された場合は、セキュアな通信を再度確立するために、これらを再生成する必要があります。

CA SSL Wallet を生成できるのは、OCA を実行していない場合だけです。OCA を実行中の場合は、OCA を停止し、次のコマンドを使用して CA SSL 証明書および Wallet を再生成します。

```
ocactl generatewallet -type CASSL
```

この Wallet は、ディレクトリ $ORACLE_HOME/oca/wallet/ssl に格納され、生成中に指定したパスワードによって暗号化されます。

また、このコマンドでは SSO 形式の CA SSL Wallet も生成され、$ORACLE_HOME/oca/wallet/ssl に cwallet.sso として格納されます。

# ルート CA 証明書の失効

ルート CA 証明書の失効は、影響が大きい操作です。インストールした OCA が機能しなくなり、すでに発行されている証明書が無効になります。この失効操作は、CA 鍵が危殆化された場合以外は実行しないでください。この操作を実行すると、新しい認証局をインストールできます。

`revokecert` コマンドを使用すると、ルート認証局または OCA 管理者の証明書を失効させることができます。このコマンドが使用できるのは、OCA が実行されていない場合だけです。

ルート認証局の証明書の失効は、実行中の OCA 操作用に新しいルート CA をインストールする前に実行する必要があります。

新しい CA をインストールする場合、最初に、`revokecert` を使用して、パラメータで理由コードを指定し、以前の CA Wallet を失効させます。ルート証明書が失効すると、CA が発行したすべての証明書は整合性のない状態になります。そのため、ルート CA 証明書を失効させる前に、最初に、既存の CA が発行した証明書をすべて失効させ、証明書失効リストを更新します。この操作を実行しない場合、新しい CA 署名証明書の生成時に、以前の CA が署名した古い証明書すべてが、OCA リポジトリで「無効」のマークが付けられます。

OCA 管理者の証明書を失効させると、新しい証明書を取得するまで、管理者は、Web 上の管理機能にいっさいアクセスできません。管理者が管理ホームページを開くと、新しい管理者の証明書を取得するために、新規登録が要求されます。

ルート認証局の証明書を失効させるには、最初に、OCA を停止する必要があります。その後、次のコマンドを発行します。

```
ocactl revokecert -type CA -reason < 失効理由 >
```

CA 証明書を失効させる主な理由は鍵の危殆化なので、実際のコマンドは次のようになります。

```
ocactl revokecert -type CA -reason KEY_COMPROMISE
```

それ以外の状況で失効が必要な場合は、< 失効理由 > エントリを、次の 8 つの句のうち、最も適したものに置換することができます。

**表 A-5 `revokecert` コマンドで使用する失効理由**

| 失効理由 | 説明 |
|---|---|
| AFFILIATION_CHANGE | 組織が、別の CA の使用を決定した。 |
| CA_COMPROMISE | なんらかの理由でルート CA を信頼できないため、新しい CA が必要になった。 |
| CERTIFICATE_HOLD | なんらかの疑惑があるために証明書が保留されている。 |
| CESSATION_OF_OPERATION | 現在のルート CA が稼動を中止したため、新しい CA が必要になった。 |
| KEY_COMPROMISE | ルート CA の鍵が危殆化したため、そのルート CA に基づく証明書が信頼できない。 |
| REMOVE_FROM_CRL | 証明書の状態は REVOKED であるが、この失効した証明書が CRL に追加されない。 |
| SUPERSEDED | ルート CA の証明書が置換された。以前の証明書は削除し、新しい証明書をインストールする必要がある。 |
| UNSPECIFIED | 使用可能な理由がない、または指定されていない。これがデフォルトの理由コード。 |

# CA SSL サーバー Wallet の SSO 形式への変換

管理コマンドライン・ツールの convertwallet コマンドを使用すると、CA SSL サーバー Wallet を Oracle Single Sign-On（SSO）形式に変換できます。このコマンドは、他のディレクトリを指定しないかぎり、現行の CA SSL Wallet の場所を使用します。

CA SSL サーバー Wallet を SSO 形式に変換する場合、ルート・ユーザーとして <wlt-location> にディレクトリを指定して、次のコマンドを発行します。

```
ocactl convertwallet -format SSO [-walletwrl <wlt-location>]
```

オプションのパラメータ -walletwrl は、CA SSL PKCS #12 Wallet が格納されているソースの場所を指定する次のパラメータを識別します。この Wallet は、ここで指定するディレクトリに、ewallet.p12 というファイル名で格納する必要があります。-walletwrl を指定すると、ocactl は、OCA によって作成されていない（OCA 以外から取得された）CA SSL Wallet を管理者が変換しようとしているとみなします。OCA のパスワード・ストアにはパスワードが含まれないため、管理者は元の CA SSL Wallet のパスワードを指定して、指定したディレクトリの Wallet を読み取る必要があります。Wallet が開かれた後、証明書は .sso 形式に変換され、-walletwrl <wlt-location> で指定した同一の場所に戻されます。

このソースの場所を指定しない場合、ocactl は、この Wallet を、OCA のインストール時に OCA によって生成された CA SSL Wallet であるとみなします。そのため、このコマンドは、ocactl コマンドの使用を有効にする際に指定した OCA 管理者のパスワードを使用して、CA SSL パスワードを含む内部パスワード・ストアを開きます。その後、このパスワードを使用し、CA SSL Wallet（$ORACLE_HOME/oca/wallet/ssl ディレクトリに格納）を開いて変換します。

Wallet の格納先を示す <wlt-location> を指定しない場合、デフォルトでこの Wallet は $ORACLE_HOME/oca/wallet/ssl（またはインストール時に指定した場所）に格納されます。

# Oracle Application Server Certificate Authority からの下位 CA Wallet の生成

次の手順で、Oracle Application Server Certificate Authority から下位 CA Wallet を生成できます。

1. 新しい Wallet を作成し、Oracle Wallet Manager を使用して証明書要求を生成します。
2. サーバー / 下位 CA 登録フォームを使用して PKCS #10 要求を送信し、証明書の使用方法に「CA 署名」を選択します。
3. Oracle Application Server Certificate Authority の管理フォームを使用して、下位 CA 証明書を発行します。パス長（下位 CA が所有できるレベルの数）を指定します。
4. サーバー / 下位 CA 登録フォームに移動します。「CA 証明書のダウンロード」をクリックすると、CA 証明書が表示されます。上位 CA が存在する場合は、その内容も記述されます。

5. CA の BASE64 証明書を画面からコピーして、信頼できる証明書として Oracle Wallet Manager にインポートします。CA とともにトラスト・ポイントが存在する場合は、「信頼できる証明書のインポート」オプションを使用して、1 つずつ Oracle Wallet Manager にコピーします。
6. サーバー / 下位 CA 登録フォームを使用して、下位 CA のシリアル番号または一般名を指定して、証明書の詳細を取得します。「詳細表示」をクリックして、下位 CA 証明書を BASE64 形式で表示します。
7. BASE64 形式の下位 CA 証明書をコピーして、ユーザー証明書として Oracle Wallet Manager にインポートします。
8. Oracle Wallet Manager を使用して、下位 CA Wallet を保存します。Wallet は ewallet.p12 として格納されます。

# 下位 CA Wallet のインストール / インポート

この項の手順を実行すると、下位 CA Wallet をインストールおよび使用して、CA の階層を作成できます。この Wallet は、「Oracle Application Server Certificate Authority からの下位 CA Wallet の生成」の項に示すとおり、Oracle Application Server Certificate Authority から生成できます。また、CMS などの X.509 v3 準拠の CA からも生成できます。

---

**注意：** X.509 v3 CA から SSL Wallet をインポートする場合、『Oracle Application Server 10*g* セキュリティ・ガイド』に示す Oracle HTTP Server の構成手順に従ってください。また、『Oracle Advanced Security 管理者ガイド』の Oracle Wallet Manager の説明も参照してください。

---

下位 CA Wallet をインポートする前に、Oracle Application Server Certificate Authority を正常にインストールしておく必要があります。インストールしてあれば、リポジトリ、パスワード・ストア、ルート CA Wallet および CA SSL Wallet が作成されます。その後、次の手順を実行します。

1. OC4J および OHS が実行されている場合は、次のコマンドを使用して停止します。

```
$ORACLE_HOME/opmn/bin/opmnctl stopproc type=oc4j instancename=oca
$ORACLE_HOME/opmn/bin/opmnctl stopproc type=ohs
```

2. ocactl importwallet コマンドを使用して、下位 CA Wallet をインストールします。コマンドは次のとおりです。

```
importwallet -type SUBCA
```

このコマンドを実行すると、管理者のパスワード、新しい下位 CA（ewallet.p12）の Wallet が格納されているディレクトリ、およびその Wallet のパスワードを入力するように要求されます。その後、その Wallet から新しい CA の証明書および秘密鍵がフェッチされ、OCA リポジトリに格納されます。コマンドからの要求に対して入力する、新しい CA の Wallet に使用されるパスワードは、新しい CA の署名パスワードです。このパスワードは、OCA 管理者のパスワードになります。

詳細は、付録 B「CA の階層の設定」を参照してください。

以前のルート CA 証明書のエントリ、および以前のルート CA が署名したその他のすべての証明書は無効になります。Oracle Application Server Certificate Authority リポジトリに格納されている以前の CA 証明書および CA 鍵は、それぞれ新しい下位 CA 証明書および下位 CA 鍵によって上書きされます。新しい下位 CA が発行する証明書は、以前の下位 CA 証明書よりもシリアル番号が大きいため、新しい下位 CA 証明書のエントリおよびシリアル番号がリポジトリに追加されます。また、以前の CA が発行した管理者証明書はパスワード・ストアから削除されます。下位 CA Wallet のインポート中に、ocactl は、BasicConstraintsExtension と KeyUsageExtensions が DIGITAL_SIGNATURE、KEY_CERT_SIGN、CRL_SIGN および NON_REPUDIATION になるように、正しいビット数が設定されていることを確認します。これらの拡張子が設定されていない場合、この Wallet は下位 CA Wallet として受け入れられない場合があります。

# 下位 CA 用の CA SSL Wallet の生成

「認証局の SSL 証明書および Wallet の再生成」の項に示すとおり、CA SSL 証明書および Wallet はインストール時に生成されます。これらを使用すると、Oracle Application Server Certificate Authority は、HTTPS モードでリスニングできるようになります。また、これらが危殆化または破壊された場合は、セキュアな通信を確立するために再生成できます。

下位 CA SSL Wallet は、OCA が実行されていない場合に次のコマンドで生成することもできます。

```
ocactl generatewallet -type CASSL
```

この Wallet は下位 CA によって署名され、Wallet の生成中に指定したパスワードで暗号化されて、ディレクトリ $ORACLE_HOME/oca/wallet/ssl に格納されます。

ルート・ユーザーは、次のコマンドを使用して、この Wallet を SSO 形式に変換できます。

```
ocactl convertwallet -format SSO
```

下位 CA をインストールすると、SSL 証明書を発行した以前の CA は存在しなくなります。OCA に接続中のクライアントは、現行の CA 証明書を信頼します。以前の CA が発行した CA SSL は信頼できないため、下位 CA をインポートした後、または CA SSL Wallet が破壊または危殆化された後、CA SSL 証明書を再生成する必要があります。

この CA SSL 証明書および Wallet を生成した後、次の手順を実行します。

1. HTTP Server を起動します。
2. OC4J を起動します。
3. Oracle Application Server Certificate Authority を起動します。

Oracle Application Server Certificate Authority は、署名証明書の要求に下位 CA 証明書を使用するようになります。

# ログまたはトレース記憶域の消去

次の管理コマンドライン・ツールを使用すると、既存のログ・ファイルまたはトレース・ファイルを管理者が選択して消去できます。消去コマンドの形式は次のとおりです。

```
ocactl clear -type {LOG |TRACE} -mode {OCA | ADMIN}
```

コマンドは次のとおりです。

- `ocactl clear -type LOG -mode ADMIN`
- `ocactl clear -type TRACE -mode ADMIN`
- `ocactl clear -type LOG -mode OCA`
- `ocactl clear -type TRACE -mode OCA`

各コマンドを実行すると、対応するログ・データまたはトレース・データが消去されます。ログ・データを消去すると、OCA リポジトリからデータが削除され、トレース・データを消去すると、$ORACLE_HOME/oca/logs からファイル oca.trc が削除されます。

# OCA リポジトリ接続情報の更新

証明書の公開に使用されるこの接続情報は、Oracle Application Server Certificate Authority 管理者用の Web ベースのインタフェースの「一般」サブタブにある設定セクションに表示されます。この情報には、OCA がリポジトリや Oracle Internet Directory（OID）に接続するときに使用する接続文字列などがあります。

`ocactl` コマンド `updateconnection` は、OCA 構成ファイル $ORACLE_HOME/oca/conf/oca.conf に接続情報を書き込みます。

---

**注意：** changeschema、changesecurity、clear、generatewallet、help、importwallet、linksso、renewcert、revokecert、set、setpasswd、start、stop、unlinksso および updateconnection については、A-3 ページの表 A-2「OracleAS Certificate Authority（OCA）ocactl ツールの操作およびパラメータ」を参照してください。

---

OCA の接続情報は、最初に OracleAS のインストール時に OID に書き込まれます。この接続情報は OID からフェッチされ、oca.conf にも書き込まれます。OCA が別のデータベースに移動した場合、この接続情報は変更されます。

# SSO 認証の設定（linksso および unlinksso コマンド）

Single Sign-On 認証を使用すると、リソースおよびアプリケーションへのアクセス速度が上がります。また、ユーザー名とパスワードのかわりに証明書を使用すると、より迅速かつ効果的にアクセスできるようになります。

Oracle Application Server Certificate Authority には優先プロセスがあり、それによって、SSO 認証済ユーザーはこのような証明書を要求および受信できます。

OCA 管理者が `ocactl linksso` コマンドを実行すると、OCA は SSO に登録され、OCA 証明書の登録フォームが、証明書を持たない SSO ユーザーに表示されます。このような高速処理機能を使用すると、SSO ユーザーは証明書を要求でき、その証明書を OCA が発行した後、今後の認証で使用するためにその証明書をブラウザにインポートできます。

この処理機能の詳細は、第 3 章「OCA および証明書の管理の概要」の「Single Sign-On（SSO）および OracleAS Certificate Authority（OCA）」の項を参照してください。概要は、第 6 章「OracleAS Certificate Authority の管理 : 高度なトピック」の「OracleAS Certificate Authority および高可用性機能」の項を参照してください。

`ocactl linksso` コマンドでは OCA サービスを停止する必要はありません。ただし、SSO Server を再起動するまでこのサービスは有効になりません。

# ログ / トレース・オプションの設定

管理者は、`ocactl set` コマンドでデータのタイプおよびそのデータ生成のオン / オフを指定して、ロギング操作およびトレース操作を開始できます。このコマンドの形式は次のとおりです。

```
ocactl set -type LOG -state ON
ocactl set -type TRACE -state ON
ocactl set -type LOG -state OFF
ocactl set -type TRACE -state OFF
```

前述の最初の 2 つのコマンドで生成されたデータは、次の場所に格納されます。

- ログ・データの場合は OCA リポジトリ
- トレース・データの場合はオペレーティング・システム・ファイル oca.trc

OFF コマンドを使用すると、ログ・データまたはトレース・データの生成処理が停止します。すでに収集されたデータは、`ocactl clear` コマンドを発行するか、Oracle Application Server Certificate Authority が操作を停止するまで、指定された場所に格納されています。

# B

# CA の階層の設定

この付録では、下位認証局の取得方法およびインポート方法について説明します。下位認証局とは、証明書が上位の CA 機関で署名される CA です。下位 CA は、遠隔地の部門で使用するために、企業の本社にインストールされている元の Oracle Application Server Certificate Authority によって認可することができます。また、新しい下位 CA は、OCA とは階層やルートが異なる、まったく違う認証局によって認可（署名）することもできます。

次に、取得およびインポート処理の概要について説明します。

OracleAS Certificate Authority の管理者は、Oracle Wallet Manager（OWM）や第三者機関による同様の方式を使用して、下位 CA Wallet および証明書を取得します。通常、PKCS #10 証明書要求を生成する最初の手順は、フォームへの必要事項の入力です。OWM は、入力の完了したフォームを使用して要求を作成します。この要求は、要求元のエンティティの認証に必要な情報がすべて記載されたテキストを暗号化した本文となります。

次に、管理者は、OWM インタフェースからこの要求フォームをコピーして、第三者機関の証明書発行インタフェースに貼り付け、証明書の要求 ID を受信します。この ID は、BASE64 形式の証明書が発行時にフェッチおよび表示するために使用できます。その他の CA については、CA 固有の手順を実行します。CA によっては、証明書がユーザーのメール ID に送信される場合もあります。

証明書を受信した後、OWM を使用してユーザー証明書としてインポートし、証明書の発行元である CA をトラスト・ポイントとして追加します。証明書が承認された後、OWM によって PKCS #12 形式の Wallet に格納されます。この Wallet は、下位 CA Wallet として使用できます。

OCA の管理ツールにはインポート・オプションが用意されているので、管理者は、格納された下位 CA Wallet および証明書を、下位 CA として実行中の OCA インスタンスにインポートできます。インポート操作では、OCA の標準的な操作に適応するように、暗号化および格納場所の自動的な変更も行われます。この付録では、次のトピックでこれらの手順をすべて説明します。

- 下位 CA Wallet の生成
- 新しい下位 CA Wallet のインストールおよび使用
- 下位 CA 用の CA SSL Wallet および CA SMIME Wallet の生成

# 下位 CA Wallet の生成

この後の手順では、OCA の管理者向けに、Oracle CA から下位 CA Wallet を生成する方法について説明します。

1. Oracle Wallet Manager またはサード・パーティによるツールを使用して、PKCS #10 要求を生成します。
2. OCA のサーバー / 下位 CA 登録フォームを使用して PKCS #10 要求を送信し、証明書の使用方法に「CA 署名」を選択します。

   **関連項目：** 第 7 章「Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース」の「「サーバー / 下位 CA 証明書」タブ」の項

3. OCA 管理フォームを使用して、下位 CA 証明書を発行します（第三者機関への登録を使用した場合は、証明書が通知されるのを待ちます）。

   **関連項目：** 第 3 章「OCA および証明書の管理の概要」の「証明書要求の承認または拒否」の項

4. 証明書が承認されたら（第三者機関の発行元を使用した場合は、第三者機関から承認通知を受信したら）、サーバー / 下位 CA 登録フォームに移動して、「CA 証明書のダウンロード」をクリックします。「拡張」ボタンが表示されます。「拡張」をクリックすると、CA 証明書が PKCS #7 形式の CA 連鎖の下に表示されます。トラスト・ポイントが存在する場合は、トラスト・ポイントも表示されます。
5. CA の BASE64 証明書を画面からコピーして、Oracle Wallet Manager に移動し、信頼できる証明書として OWM にインポートします。CA とともにトラスト・ポイントが存在する場合は、OWM の「信頼できる証明書のインポート」オプションを使用して、1 つずつ Oracle Wallet Manager にコピーします。

   **関連項目：**『Oracle Application Server 10*g* セキュリティ・ガイド』の Oracle Wallet Manager に関する章

6. サーバー / 下位 CA 登録フォームを使用して、下位 CA のシリアル番号または一般名を指定して、証明書の詳細を取得します。「詳細表示」をクリックして、下位 CA 証明書を BASE64 形式で表示します。
7. BASE64 形式の下位 CA 証明書をコピーして、ユーザー証明書として OWM にインポートします。
8. OWM を使用して、指定したファイル格納先に下位 CA Wallet を保存します。

# 新しい下位 CA Wallet のインストールおよび使用

この項の手順を実行することで、CA の階層を作成できます。新しい下位 CA の Wallet は、OCA または X.509 v3 準拠の CA で生成できます。下位 CA の Wallet は、それをインストールした後、証明書が発行される前に、Oracle Wallet Manager でただちに作成する必要があります。その時点で作成しておかないと、新しい下位 CA のインストール後に、こうした証明書は無効になります。第三者機関の発行元には、iPlanet Certificate Management System（CMS）や VeriSign 社などがあります。第三者機関の証明書を使用するには、証明書が、付録 D「拡張領域」で説明する OCA の拡張領域要件を満たしている必要があります。

**関連項目：** 第 7 章「Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース」の「下位 CA 証明書」の項

1. Oracle Application Server Certificate Authority をインストールすると、OCA リポジトリ、パスワード・ストア、ルート CA Wallet および CA SSL Wallet が作成されます。

   ---

   **注意：** 1 つのリポジトリでの OracleAS Certificate Authority のスキーマは、1 つの OCA とのみ併用できます。

   別の OracleAS Certificate Authority をインストールする場合、先行する OCA のインストールに使用したリポジトリは選択できません。同じリポジトリを選択すると、OCA 構成ツールが正常に実行されません。

   それによって、インストール処理を途中で終了し、インストール全体をやり直すことが必要になります。

   ---

2. OC4J および Oracle HTTP Server（Apache）が実行されている場合は、次のコマンドを使用して停止します。

   ```
   $ORACLE_HOME/opmn/bin/opmnctl stopproc type=oc4j instancename=oca
   $ORACLE_HOME/opmn/bin/opmnctl stopproc type=ohs
   ```

3. 次のコマンドを実行して、下位 CA Wallet をインストールします。

   ```
   ocactl importwallet -type SUBCA
   ```

   **関連項目：** 詳細は、付録 A「コマンドライン管理」を参照してください。たとえば、下位 CA Wallet をインポートするとき、ocactl では、適切な拡張領域に正しいビットが設定されている必要があります。Wallet が下位 CA Wallet として機能できるのは、正しいビットが設定されている場合だけです。BasicConstraintsExtension には、DIGITAL_SIGNATURE が設定されている必要があります。KeyUsageExtensions では、KEY_CERT_SIGN（証明書署名）、CRL_SIGN および NON_REPUDIATION の 3 つのビットがすべて設定されている必要があります。

**注意：** importwallet を実行してエラー・メッセージが表示された場合は、ブラウザに証明書をインポートし、その詳細を表示してエラーを確認してください。Internet Explorer では、BasicConstraintsExtension と KeyUsageExtensions のどちらかに、適切な設定がされていないことが示されます。

下位 CA Wallet をインストールすると、次の処理が行われます。

**a.** 既存の管理者のパスワード、新しい下位 CA の Wallet（ewallet.p12）が格納されているディレクトリおよびその Wallet のパスワードを入力するように要求されます。

コマンドからの要求に対して入力する、新しい CA の Wallet に使用されるパスワードは、新しい CA の署名パスワードです。このパスワードは、OCA 管理者のパスワードになります。

**b.** その Wallet から新しい下位 CA の証明書、秘密鍵およびシリアル番号がフェッチされ、OCA リポジトリに格納されます。

この操作により、以前から OCA リポジトリ内にあるレコードのうち、対応するものが上書きされます。したがって、以前のルート CA 証明書、鍵および署名証明書のパスワードは、新しい下位 CA 証明書、鍵およびパスワードにそれぞれ置換されます。

**c.** この下位 CA が発行する証明書のシリアル番号が下位 CA 証明書のシリアル番号より大きくなるように、下位 CA 証明書の現行のシリアル番号が更新されます。また、以前の CA が発行した管理者証明書はパスワード・ストアから削除されます。

この時点で、ルート・ユーザーとして、次の手順を実行する必要があります。

1. 既存の CA SSL は以前の CA によって署名されているため、新しい CA SSL Wallet を生成します。次のコマンドを使用します。

```
ocactl generatewallet -type CASSL
```

生成された CA SSL Wallet は、新しい下位 CA 証明書によって署名されます。

2. 次のコマンドを実行して、この Wallet を SSO 形式に変換します。

```
ocactl convertwallet -format SSO
```

3. コマンドライン・ツール opmn を使用して、HTTP Server を起動します。
4. 同じコマンドライン・ツールを使用して、OC4J を起動します。
5. Oracle Application Server Certificate Authority を起動します。その後の証明書要求はすべて、新しい下位 CA 証明書を使用して署名されます。

**関連項目：**『Oracle Application Server 10*g* セキュリティ・ガイド』。特に、付録「Oracle Wallet Manager での PKI 資格証明の管理」。

# 別の CA の下位 CA にするための OCA インスタンスの構成

大規模な組織で複数の部門が各地に分散している場合、ルート CA から下位 CA Wallet を取得して、その下位 CA を、別にインストールしてある OCA にインストールすることをお薦めします。ルート CA Wallet を持つ親組織は、下位の組織または部門のそれぞれに、下位 CA Wallet を発行できます。このような下位 CA は、それぞれの場所で認証局（CA）として機能し、その組織に固有の証明書を管理します。下位 CA が別の下位 CA Wallet を発行しないようにするには、その下位 CA の Wallet がルート CA によって発行されていないときにパス長を設定します。

次の手順を実行すると、Oracle Application Server Certificate Authority から下位 CA Wallet を生成および使用できます。

1. 新しい Wallet を作成し、Oracle Wallet Manager（OWM）を使用して PKCS #10 証明書要求を生成します。要求をコピーして OCA に送信します。

   **関連項目：** 『Oracle Application Server 10*g* セキュリティ・ガイド』。特に、付録「Oracle Wallet Manager での PKI 資格証明の管理」。

2. 第 7 章で説明したユーザー・インタフェースのサーバー / 下位 CA 登録フォームを使用して、OWM で生成した PKCS #10 要求に貼り付け、証明書の使用方法に「CA 署名」を選択します。

3. 第 3 章で説明した管理インタフェースの OracleAS Certificate Authority 管理フォームを使用して、下位 CA 証明書を発行します。パス長（下位 CA が所有できるレベルの数）を指定します。

4. 証明書が承認されたら、サーバー / 下位 CA 登録フォームに戻り、「CA 証明書のダウンロード」をクリックすると、CA 証明書が表示されます。上位 CA が存在する場合は、その内容も記述されます。

   **関連項目：** 第 7 章「Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース」の「「サーバー / 下位 CA 証明書」タブ」の項

5. 「拡張」をクリックして、BASE64 エンコード済の証明書を表示します。

6. CA の BASE64 証明書を画面からコピーして、信頼できる証明書として Oracle Wallet Manager にインポートします。この CA が CA 階層中で下位 CA の場合は、階層内のすべての CA を OWM にインポートする必要があります。「信頼できる証明書のインポート」オプションを使用して、1 つずつ Oracle Wallet Manager にコピーします。

この時点で、次の手順を実行し、証明書の詳細を OWM にコピーし、その Wallet を保存する必要があります。

1. サーバー / 下位 CA 登録フォームで、下位 CA のシリアル番号または一般名を使用して、目的の証明書を検索します。
   a. シリアル番号を使用する場合は、その左側にあるラジオ・ボタンをクリックして選択し、右側のハイパーテキスト・リンクをクリックして証明書を表示します。
   b. 一般名を使用する場合は、それを入力して「実行」をクリックし、表示されたリストから目的の証明書を選択します。
2. 「詳細表示」をクリックして、下位 CA 証明書を BASE64 形式で表示します。
3. BASE64 形式の下位 CA 証明書をコピーして、ユーザー証明書として Oracle Wallet Manager にインポートします。
4. Oracle Wallet Manager を使用して、下位 CA Wallet を保存します。Wallet は ewallet.p12 として格納されます。

## 下位 CA 用の CA SSL Wallet および CA SMIME Wallet の生成

第 6 章の「CA SSL Wallet および CA SMIME Wallet の再生成」の項に示すとおり、CA SSL Wallet はインストール時に生成されます。CA SSL Wallet を使用すると、Oracle Application Server Certificate Authority は、HTTPS モードでリスニングできるようになります。また必要に応じて、セキュアな通信を再確立するために CA SSL Wallet を再生成できます。こうした再生成が必要になる状況には、Wallet が危殆化または破壊された場合、CA Wallet が再生成された場合、新しい下位 CA 証明書がインポートされた場合などがあります。

下位 CA SSL Wallet は、OCA が実行されていない場合に次のコマンドで生成することもできます。

```
ocactl generatewallet -type CASSL
```

この Wallet は下位 CA によって署名され、Wallet の生成中に指定したパスワードで暗号化されて、ディレクトリ $ORACLE_HOME/oca/wallet/ssl に格納されます。

下位 CA をインストールすると、SSL 証明書を発行した以前の CA は存在しなくなります。OCA に接続中のクライアントは、現行の CA 証明書を信頼します。以前の CA が発行した CA SSL は信頼されないため、下位 CA をインポートした後、または CA SSL Wallet が破壊または危殆化された後に、CA SSL 証明書を再生成する必要があります。

同様に、下位 CA をインポートした後、以前の CA が発行した CA SMIME Wallet は無効になります。「SMIME 電子メールの送信」が、OCA 管理ページの「構成管理」の「通知」ページで有効な場合は、CA SMIME Wallet を生成して、アラートおよび通知に署名する必要があります。次のコマンドを使用して、CA SMIME Wallet を生成します。

```
ocactl generatewallet -type CASMIME
```

CA SSL Wallet および CA SMIME Wallet を生成した後、次の手順を実行します。

1. OC4J および HTTP Server を起動します。
2. Oracle Application Server Certificate Authority を起動します。

Oracle Application Server Certificate Authority は、署名証明書の要求に下位 CA 証明書を使用するようになります。

# C

# トラブルシューティングの既知のヒント

この付録では、Oracle Application Server Certificate Authority のインストールまたは管理で発生する可能性のある問題について説明します。

この付録の次の各項では、今回のリリースで前述の問題をどのように解決するかを説明しています。

1. 基礎的な問題および警告

a. 問題 : 証明書要求で鍵のペアが生成されない（Windows）。

b. 問題 : 通常のユーザーでログインした後、管理者でログインできない。

c. 問題 : パスワードの変更には、OCA のコマンドライン・ツール ocactl を使用する必要がある。

2. ブラウザの問題

1. a. 問題 : CA SSL サーバーの CN がマシン名と一致しない場合に、ブラウザが警告を表示する。
2. b. 問題 : 最初（最も右側）の CN 構成要素しか使用されない。
3. c. Netscape の場合
   - i. 問題 : 複数の証明書が使用可能であるにもかかわらず、ポップアップ・ウィンドウに証明書が 1 つしか表示されない。
   - ii. 問題 : CA 証明書が信頼できるかどうかの質問が引き続き表示される。
   - iii. 問題 : 証明書の有効期限が切れたという警告が表示される。
   - iv. 問題 : 下位 CA と CA SSL 両方のクライアント証明書が表示される。
4. d. Internet Explorer（IE）の場合
   - i. 問題 : 「ページを表示できません」というメッセージが表示される。
   - ii. 問題 : ブラウザに CRL をインポートできない。

- iii. 問題 : セキュアな情報とセキュアでない情報の両方がページに含まれているというメッセージが表示される。
- iv. 問題 : オンライン・ヘルプを開くと、セキュリティ・アラートが生成される。

3. ネットワークの問題

a. 問題 : SSO ユーザー名 / パスワードを使用して OCA にログインすると、エラー・メッセージが表示される。

b. 問題 : ネットワーク・エラーのメッセージが表示される。

c. 問題 : OCA が動作しなくなる。あるいはネットワークまたはサーバーのメッセージが表示される。

4. 証明書の問題

a. 問題 : ユーザー証明書をインポートしても CA 証明書がインポートされない（Netscape）。

b. 問題 : 「認証管理」タブへのアクセスまたは使用ができない。

c. 問題 : 管理者が別のマシンから作業する必要がある。

5. シングル・サインオン（SSO）の問題

a. 問題 : SSO の証明書に表示される名前が「USER」になる。

b. 問題 : 鍵の生成中に VB スクリプトのエラー・メッセージが表示される。

c. 問題 : 「ページを表示できません」というメッセージが表示される（Internet Explorer）。

6. 検索の問題

a. 問題 : 検索画面で [Enter] を押すと内部エラーが発生する。

7. バックアップの保護の問題

a. 問題 : OCA の内部リポジトリのリカバリ可能性を保証する。

# 1. 基礎的な問題および警告

OCA の使用を進める前にあらかじめ対処しておく必要がある特定の問題があります。このような問題は「前提条件」と呼ばれます。

## a. 問題 : 証明書要求で鍵のペアが生成されない（Windows）。

Windows クライアント・マシンでは、この操作には、Service Pack 5 以上が必要です。

対策

- Microsoft 社の Web サイトにアクセスして、構成に必要なアップグレードをダウンロードします。

OCA に関連するパスワードを変更するには、`ocactl` を常に使用します。その他のツールは使用しないでください。

## b. 問題 : 通常のユーザーでログインした後、管理者でログインできない。

最初に SSL を介して通常のユーザーで OCA にログインし、次に Certificate Management に移動しようとすると、JAZN エラーが発生します。これは、Web 管理者として登録されている場合でも、その権限でログインしなければ Web 管理者として認識されないためです。OCA と管理者以外のユーザーの間に確立された SSL セッションはアクティブなままで、SSL セッションは変更されません。

対策

Web 管理者でログインするには、次の手順を実行します。

1. Web 管理者として登録します。
2. ブラウザを終了します。
3. 認証用の Web 管理者証明書を選択して、Web 管理者としてログインします。

## c. 問題 : パスワードの変更には、OCA のコマンドライン・ツール ocactl を使用する必要がある。

CA SSL Wallet、OCA の内部リポジトリまたは OCA 管理者のパスワードを変更することが望ましい場合があります。`ocactl` 以外のツールを使用して、これらのパスワードのいずれかを変更すると、OCA が停止します。

# 2. ブラウザの問題

特定の種類またはバージョンのブラウザを使用する場合しか発生しない問題もあります。この項では、ブラウザに関連する既知の問題について説明します。

## a. 問題 : CA SSL サーバーの CN がマシン名と一致しない場合に、ブラウザが警告を表示する。

マシン名は、広範囲で使用されるため変更が煩雑になります。したがって、CA SSL サーバーの CN は、マシン名と同一にする必要があります。この場合、新しい証明書が必要です。

## b. 問題 : 最初（最も右側）の CN 構成要素しか使用されない。

DN に複数の CN 構成要素があるとき、その DN の証明書の名前には、（右から）最初の CN 構成要素しか使用されません。この証明書は、Microsoft Internet Explorer でも Netscape（4.7x および 7.x）でも、SSL 相互認証のポップアップ・ウィンドウに「ユーザーの証明書」として一覧表示されます。

## c. Netscape の場合

次の問題は Netscape クライアントにのみ影響します。

### i. 問題 : 複数の証明書が使用可能であるにもかかわらず、ポップアップ・ウィンドウに証明書が 1 つしか表示されない。

Netscape 4.79 では、このポップアップ・ウィンドウに表示されるのは最新の証明書だけです。

対策

- 目的の証明書がリストの最後に表示されるように、証明書の順序を変更します。

### ii. 問題 : CA 証明書が信頼できるかどうかの質問が引き続き表示される。

Netscape 4.7x バージョンでは、CA 証明書が自動的に信頼されず、CA 証明書が信頼できることをユーザーが明示的に示す必要があります。それまでは、信頼できる証明書として扱われません。

対策

- 詳細は、第 7 章「Oracle Application Server Certificate Authority のエンド・ユーザー・インタフェース」の「ブラウザへの新規発行の証明書のインポート」の項を参照してください。

### iii. 問題 : 証明書の有効期限が切れたという警告が表示される。

クライアントのタイムゾーンがサーバーのタイムゾーンよりも遅れている場合は、証明書の有効期限が切れたという警告が、一定期間表示される場合があります。これは、ユーザーのタイムゾーンでは、CA SSL 証明書がまだ有効ではないためです。

対策

- この問題は、タイムゾーンの差に応じて、比較的短時間で自然に解決されます。

### iv. 問題 : 下位 CA と CA SSL 両方のクライアント証明書が表示される。

この問題は、Netscape 7.x ブラウザで生じます。ユーザーが、CA からとその CA の下位 CA からの 2 つの SSL クライアント証明書を持つ場合は、下位 CA へのクライアント認証時に、両方の証明書が一覧表示されます。その SSL サイトで使用されている CA に適した証明書を選択してください。

## d. Internet Explorer（IE）の場合

次の問題は Internet Explorer クライアントにのみ影響します。

### i. 問題 :「ページを表示できません」というメッセージが表示される。

これらの断続的なエラーは、SSL モードで対話中に発生する場合があります。たとえば、ユーザー名とパスワードを入力して SSO にログインした後、SSL を選択して認証を変更したときに発生します。このエラーは、IE の既知の不具合です。

対策

- ページの再ロードを試行します。解決されない場合は、ブラウザの現行セッションを終了し、Oracle Application Server Certificate Authority に再度アクセスして再試行します。

### ii. 問題 : ブラウザに CRL をインポートできない。

IE で「ブラウザに CRL をインポート」ボタンを使用すると、CRL は表示されますが、ブラウザには実際にインポートされません。

対策

- IE のメニューから、「ツール」→「インターネット オプション」→「コンテンツ」→「証明書」→「インポート」コマンドを選択します。

### iii. 問題 : セキュアな情報とセキュアでない情報の両方がページに含まれているというメッセージが表示される。

「ユーザー・ページ」→「手動認証」→「CA 証明書のダウンロード」→「拡張」で「ヘルプ」をクリックすると、新しいウィンドウが開きます。このウィンドウに、セキュアな情報とセキュアでない情報の両方がページに含まれているというエラー・メッセージが表示される場合があります。これは、セキュリティが侵害されているわけではありません。

### iv. 問題 : オンライン・ヘルプを開くと、セキュリティ・アラートが生成される。

OCA の使用中にオンライン・ヘルプを開くと、セキュリティ・アラートが表示されます。https URL を使用しているときに第 2 の https URL をコールすると、アラートが生成されることが報告されています。

対策
「ツール」→「インターネット オプション」→「セキュリティ」→「レベルのカスタマイズ」でセキュリティのオプションを変更すると、この動作を回避できます。「設定」で「混在したコンテンツを表示する」を探し、その下の「有効にする」オプションを選択します。

## 3. ネットワークの問題

次のメッセージまたは問題は、特にネットワークに関連しています。

### a. 問題 : SSO ユーザー名 / パスワードを使用して OCA にログインすると、エラー・メッセージが表示される。

次のメッセージが表示されます。

```
Forbidden
You don't have permission to access /oca/sso/ssoInitServlet on this
server
```

このメッセージは、IP アドレスの確認時に、ブラウザと SSO Server の間で、複数の IP アドレスを持つプロキシ・サーバーが使用されている場合に表示されます。

対策

- イントラネットを介してアクセスしている場合は、ブラウザのドキュメントに従って、プロキシを使用しないようにブラウザを構成する必要があります。
- イントラネットを介してアクセスしていない場合、または前述のとおりに変更しても問題が解決しない場合は、サーバー側で次の変更を行う必要があります。SSO 構成ファイルのディレクティブ OssoIpCheck の値を、「off」に設定します。この作業を行うには、次のファイルに移動します。

  ```
  $ORACLE_HOME/Apache/Apache/conf/mod_osso.conf
  ```

  OssoIpCheck を含む行を編集して「OssoIpCheck off」にします。
- 構成ファイルを変更した後、次の停止コマンドおよび起動コマンドを実行して、Oracle HTTP Server を再起動する必要があります。

  ```
  $ORACLE_HOME/opmn/bin/opmnctl stopproc type=ohs
  $ORACLE_HOME/opmn/bin/opmnctl startproc type=ohs
  ```

## b. 問題 : ネットワーク・エラーのメッセージが表示される。

このメッセージは、Oracle Application Server Certificate Authority がある期間アクティブでなかった後に操作しようとしたため、ブラウザの認証が再度必要になるときに表示される場合があります。

対策

- 「認証管理」タブに移動して、メッセージが表示されたら「Web 管理者証明書」を選択し、OCA に対する認証を再度行う必要があります。

## c. 問題 : OCA が動作しなくなる。あるいはネットワークまたはサーバーのメッセージが表示される。

このような問題は、OCA がリポジトリや（証明書の公開に使用する）Oracle Internet Directory への接続に使用する接続文字列が、構成変更によって変更された場合に発生することがあります。変更の例としては、ポートや RAC ノードの変更があります。「接続を確立できません」または「内部サーバー・エラー」というメッセージが表示される場合があります。

対策

- 次の ocactl command コマンドを発行して、新しい接続文字列を OCA に再取得させる必要があります。
- ocactl updateconnection

  このコマンドが完了すると、$ORACLE_HOME/oca/conf/oca.conf にある構成ファイルが更新されています。
- このコマンドの使用後は、次のコマンドを発行して、OCA を再起動する必要があります。

  $ORACLE_HOME/oca/bin/ocactl stop

  $ORACLE_HOME/oca/bin/ocactl start

# 4. 証明書の問題

次の問題は、主に証明書または証明書管理に関連しています。

## a. 問題 : ユーザー証明書をインポートしても CA 証明書がインポートされない（Netscape）。

ユーザー証明書のインポートを試行しても、実際にはインポートされません。

対策

- すべての CA 証明書または下位 CA 証明書には、サブジェクト DN に O（組織）という構成要素が含まれている必要があります。CA または下位 CA の DN に必須の構成要素は、C、O および CN です。
- Oracle Application Server Certificate Authority のインストール時またはルート CA の再生成時には、ユーザーは、少なくとも国、組織および一般名（C、O、CN）を含む DN を入力する必要があります。
- 下位 CA のインストール時には、CA 署名証明書のサブジェクト DN に、O（組織）RDN が含まれていることを確認します。

## b. 問題 : 「認証管理」タブへのアクセスまたは使用ができない。

「認証管理」の機能にアクセス、またはその機能を使用しようとしたら失敗します。

対策

- 「認証管理」にアクセスするには、有効な Web 管理者証明書がブラウザにインポートされている必要があります。そのため、「認証管理」をクリックする前に、それらの証明書を申請して受信しておく必要があります。証明書を申請するには、「管理設定」タブで「Web 管理者登録」というラベルの付いたボタンをクリックします。

## c. 問題 : 管理者が別のマシンから作業する必要がある。

Oracle Application Server Certificate Authority 管理者は、複数のマシンのうち、任意のマシンから証明書の管理タスクを行う場合があります。ただし、Web 管理者証明書は、OCA Web 管理者としての認証を最初に行ったマシンのブラウザに存在します。

対策

- あるマシンから別のマシンに切り替えても証明書の管理タスクが行えるようにするには、以前のブラウザから証明書をエクスポートして、新しいブラウザにインポートする必要があります。手順は次のとおりです。
- Netscape で証明書をエクスポートするには、「セキュリティ」→「証明書」→「本人」→「Web 管理者証明書の選択」→「エクスポート」を選択します。
- Netscape で証明書をインポートするには、「セキュリティ」→「証明書」→「本人」→「証明書をインポート」を選択します。

- Internet Explorer で証明書をエクスポートするには、「インターネット オプション」→「コンテンツ」→「証明書」→「個人」→「Web 管理者証明書を選択」→「エクスポート」を選択します。
- Internet Explorer で証明書をインポートするには、「インターネット オプション」→「コンテンツ」→「証明書」→「個人」→「インポート」を選択します。

# 5. シングル・サインオン（SSO）の問題

一部の問題は、主にシングル・サインオン機能に関連しています。

## a. 問題 : SSO の証明書に表示される名前が「USER」になる。

これらの証明書には、一般名または DN は表示されません。証明書は、証明書のシリアル番号によってのみ区別されます。

対策

- 「表示」をクリックして証明書のシリアル番号を確認し、目的のシリアル番号で特定される証明書を選択します。

## b. 問題 : 鍵の生成中に VB スクリプトのエラー・メッセージが表示される。

SSO では、ポップアップ・ウィンドウの「送信」をクリックして、証明書を要求します。待機するように求めるメッセージや進行状況が表示されないため、ユーザーが「送信」を再度クリックすることがあります。この場合にこのエラーが発生します。

対策

- 「送信」を 1 回だけクリックして、証明書が戻されるまで待機します。

## c. 問題 : 「ページを表示できません」というメッセージが表示される（Internet Explorer）。

ユーザー名とパスワードを入力して SSO にログインした後、SSL を選択して認証を変更した場合、IE の既知の不具合のため、「ページを表示できません」というエラーが表示されます。

対策

- ページの再ロードを試行します。解決されない場合は、ブラウザの現行セッションを終了し、Oracle Application Server Certificate Authority に再度アクセスして再試行します。

### d. 問題 : Internet Explorer で SSO ログイン・ページに進むと、セキュリティ警告ダイアログが表示される。

対策

- この警告は、https から http に切り替えたことが原因で発生します。特に処理は必要ありません。

## 6. 検索の問題

次の問題は検索の不具合にのみ影響します。

### a. 問題 : 検索画面で [Enter] を押すと内部エラーが発生する。

対策

- このエラーは、Oracle の既知の不具合 #2224035（Marlin）です。検索を開始するには、[Enter] ではなく「実行」ボタンを使用します。

## 7. バックアップの保護の問題

次の問題は、障害が発生した後に実行できるリカバリに関連しています。

### a. 問題 : OCA の内部リポジトリのリカバリ可能性を保証する。

エラーおよび予測できない事象によって、OCA 操作を継続できなくなる可能性があります。

対策

- OCA リポジトリのバックアップを定期的に取ります。Oracle Application Server の export などのコマンドライン・ツールを使用すると、ファイルに OCA リポジトリを保存できます。その後、import ツールを使用して同一のデータベースにリストアします。

# 8. 一般的な問題

次に、一般的な性質の様々な問題を示します。

## a. 問題 : ページのロードに時間がかかりすぎる、またはハングアップする。

Oracle Application Server Certificate Authority が長時間稼動していた後などに、こうした処理の遅延が生じることがあります。

対策

- OCA の OC4J インスタンスを再起動すると、操作が再び高速になります。

## b. 問題 : 新しい Web 管理者を登録すると JAZN エラーが生じる。

Web 管理者証明書を失効させた後、OCA を起動して新しい Web 管理者を登録する前に、OHS および OCA の OC4J を再起動する必要があります。

対策

- OHS および OCA の OC4J を起動してから OCA を起動し、新しい Web 管理者を登録します。

## c. 問題 : Outlook Express に SMIME 署名証明書が表示されない。

一部の Windows 環境では、Outlook Express で SMIME 署名証明書を選択しても、証明書は表示されません。これは、Microsoft Outlook がインストールされているためです。

対策

- Outlook Express ではなく、Microsoft Outlook を使用する必要があります。

## d. 問題 : CA SSL サーバーの CN について、警告が表示される。

CA SSL サーバーの CN がマシン名と一致しない場合に、この警告が表示されます。

対策

- CN とマシン名を同じにする必要があります。

# D

# 拡張領域

Oracle Application Server Certificate Authority は、X.509 v3 および IETF の PKIX の規格に準拠しています。また、次の拡張領域もサポートしています。

1. OCA の CA 証明書には、次の拡張領域が含まれます。
   a. basicConstraints の拡張領域 : 重要
      * CA フラグは TRUE に設定されています。
      * ルート（自己署名）証明書の PathLength は、3 にハードコードされています。
      * 下位 CA の PathLength は 0 ～ 2 です。これは、発行元（上位 CA）の証明書の PathLength によって決まります。
   b. keyUsage の拡張領域 : 重要

      次のビットが ON に設定されています。

      ```
      Digital Signature
      Key Cert Sign
      CRL Sign
      Non-Repudiation
      ```

2. OCA のエンド・エンティティの SSL/ 暗号化証明書には、次の拡張領域が含まれます。
   a. keyUsage の拡張領域 : 重要ではない

      次のビットが ON に設定されています。

      ```
      Digital Signature
      Key Encipherment
      Key Agreement
      Non-Repudiation
      ```

3. コード署名証明書には、次の拡張領域が含まれます。

   a. keyUsage の拡張領域 : 重要ではない

      次のビットが ON に設定されています。

      ```
      + Digital Signature
      ```

4. SMIME 署名証明書には、次の拡張領域が含まれます。

   a. keyUsage の拡張領域 : 重要ではない

      次のビットが ON に設定されています。

      ```
      Digital Signature
      Data Encipherment
      Non-Repudiation
      ```

# E

# SSO での SSL および PKI の有効化

この付録では、OracleAS 10*g*（9.0.4）において、SSO での SSL と PKI の有効化に必要なすべての手順について説明します。コンテキストの追加説明の詳細な説明は、次のマニュアルに収録されています。

- 『Oracle Application Server Single Sign-On 管理者ガイド』
- 『Oracle HTTP Server 管理者ガイド』
- 『Oracle Advanced Security 管理者ガイド』

デフォルトでは、SSO 認証はユーザー名とパスワードに基づきます。各ユーザーを本人の証明書に基づいて認証するよう、SSO を構成することができます。構成手順については、SSO および OHS のドキュメントですでに説明していますが、それらの説明は様々な場所に分散しています。ユーザーの便宜を考慮して、それらの手順をこの付録にまとめます。

この機能を構成するためには、SSO Server に対する SSL の有効化、証明書を使用するための SSO の構成、および SSL を有効化した SSO Server への OCA の登録という、3 つの手順を個別に実行する必要があります。

注意 : このドキュメントは、UNIX と Windows の両方のプラットフォームに適用されますが、Windows の場合はパスのセパレータとして、スラッシュ（/）ではなく円記号（¥）を使用します。

SSO での SSL および PKI の有効化という目的を達成するには、次の 3 つの手順を実行する必要があります。

- SSO での SSL の有効化
- SSO での PKI の有効化
- SSL を有効化した SSO への、OCA の仮想ホストの再登録

# SSO での SSL の有効化

この項で使用する ORACLE_HOME は、SSO Server がインストールされている場所です。

1. $ORACLE_HOME/opmn/conf/opmn.xml ファイルを編集します。

   「id="HTTP」を検索し、その 4 行下にある次の行を変更します。

   ```
   <data id=”start-mode value=”ssl-disabled”>

   to read instead as follows:
   <data id="start-mode value="ssl-enabled”>
   ```

2. 新しい xml ファイルを使用して opmn を再起動します。

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

3. $ORACLE_HOME/Apache/Apache/conf/ssl.conf ファイルを編集します。

   </VirtualHost> の前の行で、次の記述を追加します。

   ```
   RewriteEngine on
   RewriteOptions inherit
   ```

4. 次のように、SSL セッションのキャッシュを無効にして、SSO からのログアウト時にハンドシェイクを実行します。

   ssl.conf.sec の SSLSessionCache ディレクティブと SSLSessionCacheTimeout ディレクティブをコメントアウトします。

   ```
   # SSLSessionCache
   # SSLSessionCacheTimeout 15
   ```

   その後、次の行を追加します。

   ```
   SSLSessionCache none
   ```

5. Wallet を更新します。OCA が同じマシンにインストールされている場合は、SSO Server 用に OCA の SSL Wallet を使用できます。

   そうでない場合は、Oracle Wallet Manager を使用して SSO Server 用の Wallet を生成する必要があります。『Oracle Advanced Security 管理者ガイド』で該当する説明を参照してください。

   通常、OCA によって生成された既存の SSL Wallet は、/app/oracle/oca/wallet/ssl にあります。このファイル（ssl.conf）内の SSLWallet ディレクティブを探して、コメントアウトします。

   ```
   # SSLWallet file:/app/oracle/product/sec_
   inf/Apache/Apache/conf/ssl.wlt/default
   ```

その後、次のような新しいディレクティブを挿入します。

```
SSLWallet file:/app/oracle/oca/wallet/ssl
```

6. 次の行をコメントアウトすることによって、クライアント認証を設定します。

```
# SSLVerifyClient require
```

その後、次のような新しい行を挿入します。

```
SSLVerifyClient optional
```

7. $ORACLE_HOME/sso/conf/sso_apache.conf ファイルを編集するために、ファイルの最後に次の行を追加します。

```
<IfDefine SSL>
<location "/sso/auth">
SSLRequireSSL
</location>
<location "/sso/ChangePwdServlet">
SSLRequireSSL
</location>
</IfDefine>

<IfModule mod_ossl.c>
<Oc4jExtractSSL on
<Location /sso>
SSLOptions +ExportCertData +StdEnvVars
</Location>
</IfModule>
```

8. SSL ポートを使用するよう、SSO Server を再構成します。コマンドの形式は次のとおりです。

```
$ORACLE_HOME/sso/bin/ssocfg.sh https  hostname  ohs_ssl_port
```

したがって、ホスト名が sso.us.oracle.com で、ohs_ssl_port が 4443 の場合、コマンドは次の行になります。

```
$ORACLE_HOME/sso/bin/ssocfg.sh https  sso.us.oracle.com 4443
```

9. SSO がインストールされている Oracle ホームで、次のコマンドを実行して sso に mod_osso を再登録します。

```
$ORACLE_HOME/jdk/bin/java  -jar $ORACLE_HOME/sso/lib/ossoreg.jar
-oracle_home_path $ORACLE_HOME   -site_name sso   -config_mod_osso TRUE
-mod_osso_url  https://hostname.domain.com:ohs_ssl_port
–update_mode CREATE   -u root
```

10. 次のコマンドを実行して、SSO 用の OHS を再起動します。

```
$ORACLE_HOME/opmn/bin/opmnctl restartproc type=ohs
```

# SSO での PKI の有効化

この項で使用する ORACLE_HOME は、SSO Server がインストールされている場所です。

次の手順に従って、SSO で PKI を有効化します。

1. 次に示すように、$ORACLE_HOME/j2ee/OC4J_SECURITY/application-deployments/sso/web にある orion-web.xml ファイルにタグを追加して、証明書用の Single Sign-On システムを構成します。

   </orion-web-app> の前に、次のタグを配置します。

   ```
   <jazn-web-app runas-mode="true" />
   ```

   次の orion-web.xml ファイルの例では、タグが正しく配置されています。

   **<jazn-web-app runas-mode="true" />**

   </orion-web-app>

2. $ORACLE_HOME/sso/conf/policy.properties を編集するために、次のように、デフォルトの認証レベルを高に設定し、対応する正しいプラグインを設定します。

   ```
   DefaultAuthLevel = MediumHighSecurity

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

3. 次の形式の行を使用して、プロビジョニングにユーザー名とパスワードを使用するよう OCA を構成します。

   ```
   MediumSecurity_AuthPlugin = oracle.security.sso.server.auth.SSOServerAuth
   Oca_hostname¥:port = MediumSecurity
   ```

たとえば、OCA のホスト名が oca.us.oracle.com で、OCA のポート番号が 4400 の場合、前述の行は次のようになります。

```
oca.us.oracle.com¥:4400=MediumSecurity
```

4. これらのオプションをすべて設定すると、どのパートナ・アプリケーションにログインするユーザーも、証明書が必要になります。ただし、OCA では証明書を取得できるので不要です。

   次のコマンドを使用して、SSO Server を再起動します。

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

# SSL を有効化した SSO への、OCA の仮想ホストの再登録

この項で使用する ORACLE_HOME は、OCA がインストールされている場所です。

管理者は、SSO Server で SSL を使用可能にするたびに、SSL が使用可能になった SSO Server に OCA の仮想ホストを再登録する必要があります。SSO を使用するすべてのアプリケーションも同様です。再登録には、シングル・サインオン登録ツールの ossoreg.jar を使用します。ここでは、OCA でのこのツールの使用方法について説明します。Single Sign-On 対応アプリケーションでの一般的な使用方法は、『Oracle Application Server Single Sign-On 管理者ガイド』を参照してください。

1. 次のコマンドを実行して、OCA 用の mod_osso を再登録します。

```
$ORACLE_HOME/jdk/bin/java -jar $ORACLE_HOME/sso/lib/ossoreg.jar
-oracle_home_path $ORACLE_HOME -site_name oca -config_mod_osso TRUE
-mod_osso_url https://hostname.domain.com:oca_ssl_port  -u root
-virtualhost
-config_file $ORACLE_HOME/Apache/Apache/conf/osso/oca/osso.conf
```

   SSO Server のホストとなっているマシンでこのツールを実行すると、SSO Server の SSL 設定を反映して、OCA の mod_osso レコードが osso.conf ファイルに生成されます。

2. 次のコマンドを実行して、OCA 用の OHS を再起動します。

```
$ORACLE_HOME/opmn/bin/opmnctl restartproc type=ohs
```

## OCA の再登録の例

OCA ホスト名が myoca.mysite.com で、OCA サーバーの認証ポート番号が 4400 だとします。この場合、再登録を完了するには、次の手順を実行します。

1. 次の 2 つのコマンドを使用して、（手順 2 の）実際のコマンドで使用する変数を指定します。

```
setenv ORACLE_HOME /sso_server/oracle_home
setenv LD_LIBRARY_PATH $ORACLE_HOME/lib
```

2. 上で設定した変数を使用すると、実際のコマンドは次のようになります（ただしこれは 1 行です）。

```
$ORACLE_HOME/jdk/bin/java -jar $ORACLE_HOME/sso/lib/ossoreg.jar
-oracle_home_path $ORACLE_HOME -site_name "my_oca_site_name"
-config_mod_osso TRUE  -mod_osso_url https://myoca.mysite.com:4400
-u root -config_file $ORACLE_HOME/Apache/Apache/conf/osso/oca/osso.conf
-virtualhost
```

# F

# 用語集

表 F-1 に、用語とその定義を示します。また、関連する項目も示します。表内の関連項目はリンクになっていますが、索引からアクセスするものもあります。

**表 F-1 OracleAS Certificate Authority で使用される用語の定義**

| 用語 | 意味 | 関連項目 |
|---|---|---|
| 認証（Authentication） | 送信、メッセージまたは発信者の有効性を確立するためのセキュリティ対策。認証は、個々のエンティティの、特定の情報を受信する権限を検証する手段となる。 | |
| 証明書（Certificate） | ユーザーの識別情報とユーザーの公開鍵を、信頼できる証書として結び付けるデジタル表現。証明書は、証明書を発行した認証局、ユーザーの名前、処理方法、証明書のユーザーとなる装置、ユーザーの公開鍵を特定するもので、認証局によりデジタル署名される。 | 認証局（Certificate Authority）, コード署名証明書（Code Signing Certificate） |
| 認証局（Certificate Authority） | 認証局（CA）は、1 人または複数のユーザーから信頼される機関で、X.509 公開鍵証明書および証明書失効リストを発行および管理する。 | 証明書（Certificate）, 証明書失効リスト（Certificate Revocation List） |
| 証明書失効リスト（Certificate Revocation List） | 略称は CRL。認証局が公開する失効済証明書のリスト。証明書は、様々な理由により、満了日前に失効されることもある。たとえば、秘密鍵が危殆化した場合に証明書を失効させることがある。CRL は自動生成が可能で、その間隔が指定されている場合は、指定されている期間が経過するたびに、Oracle 認証局が CRL を自動生成する。 | 証明書（Certificate）, 認証局（Certificate Authority） |
| クライアントの Secure Socket Layer 証明書（Client Secure Socket Layer Certificate） | Secure Socket Layer によって、サーバーに対してクライアントを認証する（クライアント認証）際に使用する証明書。 | 証明書（Certificate） |

**表 F-1 OracleAS Certificate Authority で使用される用語の定義（続き）**

| 用語 | 意味 | 関連項目 |
|---|---|---|
| コード署名証明書（Code Signing Certificate） | Java コードや JavaScript など、署名ファイルの署名者の認証に使用する証明書。 | 証明書（Certificate） |
| デジタル署名（Digital Signature） | 記述された署名の電子版で、送信前のメッセージから生成され、メッセージが発信者によって署名されていることを受信者に証明する目的で使用できる。<br>デジタル署名システムでは、次のような 3 ステップの処理が必要になる。<br>1. ハッシュ・アルゴリズムにより、データをメッセージ・ダイジェストに圧縮する（大量のデータの暗号化には、公開鍵暗号を使用しない）。<br>2. 発信者の秘密鍵を使用して、メッセージ・ダイジェストを暗号化する。<br>3. 受信者は、受信したメッセージからメッセージ・ダイジェストを再作成する。次に、公開鍵を使用してデジタル署名を復号化し、結果を比較する。<br>デジタル署名は、公開鍵暗号を応用したものである。 | 鍵のペア（Key Pair），公開鍵暗号（Public Key Encryption） |
| ディレクトリ（Directory） | 自分自身や他のユーザーの公開鍵証明書の取得元となるリポジトリをユーザーに提供し、また、証明書が失効していないことを確認できる場所にもなる。 | |
| 識別名（Distinguished Name） | それぞれの証明書を他のすべての証明書と区別する一意の識別可能な特性を、証明書の所有者に付与する目的で使用する。<br>例 :<br>cn=Sara Will, ou=Sales, o=Acme Corporation, c=AU<br>cn は一般名、ou は組織単位、o は組織、c は国をそれぞれ表す。<br>注意 : DN におけるドメイン・コンポーネント・エントリは、組織または国のエントリに追加（または置換）して使用できます。たとえば、dc=be（ベルギー）、dc=us（米国）、dc=oracle または dc=com などを使用できる。<br>DN の DC コンポーネントおよび EMAIL コンポーネントでは、印刷可能な（ASCII）文字のみを使用する必要があります。マルチバイト・キャラクタ・セットを使用するロケールでも、識別名の DC コンポーネントおよび EMAIL コンポーネントには ASCII 文字を使用する必要がある。<br>CA 証明書および CA SSL 証明書の DN には、マルチバイト・キャラクタを使用できません。CA の DN にマルチバイト・キャラクタが使用されていると、インストールに失敗する（不具合 : 2991110）。 | ドメイン・コンポーネント属性（Domain Component Attribute） |

**表 F-1 OracleAS Certificate Authority で使用される用語の定義（続き）**

| 用語 | 意味 | 関連項目 |
|---|---|---|
| ドメイン・コンポーネント属性（Domain Component Attribute） | ドメイン名から DN を構築するときに使用できる。たとえば、Acme, Inc. という名前の組織がドメイン名として acme.com を登録している場合、このネーミング・プランに従ってディレクトリを配置する手順は次のようになる。最初に、ドメイン名から次の DN を構築する。<br>dc=acme, dc=com<br>次に、この DN を、ディレクトリ情報のサブツリーのルートとして使用する。<br>DN 本体の役割は、組織情報を表すディレクトリ編成オブジェクトを識別することである。そのため、DN の下位要素が、組織に関連するディレクトリ・オブジェクトになる。ドメイン・コンポーネント属性を使用して、組織単位や所在地など、組織の下位部門の名前を指定できる。<br>たとえば、Acme では、corporate.acme.com および richmond.acme.com というドメイン名を使用して、次の名前を構築することもできる。<br>dc=corporate, dc=acme, dc=com<br>dc=richmond, dc=acme, dc=com<br>この下に、それぞれのディレクトリ・オブジェクトを配置する。こうした組織の下位部門には、従来の X.509 ネーミング属性を使用して、次のような RDN を割り当てることもできる。<br>ou=corporate, dc=acme, dc=com<br>l=richmond, dc=acme, dc=com | 識別名（Distinguished Name） |
| 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通信の暗号化、あるいは同じ目的のセッション鍵の構築または交換に使用する公開鍵を含む証明書。 | 証明書（Certificate）, 公開鍵（Public Key） |
| 鍵のペア（Key Pair） | 数学的に関連する 2 つの鍵。1 つはメッセージの暗号化に使用でき、そのメッセージはもう 1 つの鍵を使用しないと復号化できない。 | 秘密鍵（Private Key）, 公開鍵（Public Key） |
| ポリシーの優先順位（Policy Precedence） | ポリシーは、ポリシーのメイン・ページに表示される順序で、受信リクエストに適用する。Oracle 認証局のポリシー・プロセッサ・モジュールがポリシーを解析するときは、ポリシー・リストの最上部に表示されているポリシーが、リクエストに対して最初に適用される。ポリシー・リストの最下部に表示されているポリシーが最後に適用され、他のポリシーよりも優先される。注意 : 受信リクエストに適用されるのは有効なポリシーのみ。 | |

**表 F-1 OracleAS Certificate Authority で使用される用語の定義（続き）**

| 用語 | 意味 | 関連項目 |
|---|---|---|
| 条件（Predicates） | ポリシーに適用できる論理式で、受信リクエストまたは失効に対するポリシーの適用方法を制限するために使用できる。たとえば、次の条件式では、ou=sales,o=acme,c=us を含む DN を持つクライアントからのリクエストまたは失効に対して、異なる処理を行うように指定できる。<br>Type=="client" AND DN=="ou=sales,o=acme,c=us"<br>条件および条件式の構文の詳細は、第 5 章「Oracle Application Server Certificate Authority でのポリシー管理」を参照。 | |
| 秘密鍵（Private Key） | デジタル署名の作成にペアで使用する署名鍵、または、機密情報の復号化にペアで使用する暗号化鍵。どちらの場合でも、秘密鍵は秘密にしておく必要がある。 | 鍵のペア（Key Pair）, 公開鍵（Public Key） |
| 公開鍵（Public Key） | デジタル署名の検証にペアで使用する署名鍵、または、機密情報の暗号化にペアで使用する暗号化鍵。どちらの場合でも、この鍵は誰でも利用できる。 | 鍵のペア（Key Pair）, 秘密鍵（Private Key） |
| 公開鍵暗号（Public Key Encryption） | 公開鍵暗号では、鍵のペアとして一般的に知られている、対応関係にある 2 つの鍵を使用する。1 つの鍵は秘密にされ（秘密鍵）、もう 1 つは公開される（公開鍵）。秘密鍵は所有者のみが知っている必要があり、公開鍵は誰でも入手して使用できる。秘密鍵は関係者の一方のみが知っていればよいため、関係者間で送信する必要はない。そのため、秘密鍵は傍受される危険性がない。第三者が公開鍵を知っていることで、データ通信のセキュリティが危殆化されることはない。<br>秘密鍵の所有者は、秘密鍵を使用して一意なメッセージ・ダイジェストを暗号化することで、ドキュメントにデジタル署名できる。ドキュメントのソースは、公開鍵を使用してデジタル署名を復号化し、それをメッセージ・ダイジェストと比較することで検証できる。 | 鍵のペア（Key Pair）, 秘密鍵（Private Key）, 公開鍵（Public Key） |
| 公開鍵インフラストラクチャ（Public Key Infrastructure） | ハードウェア、ソフトウェア、ポリシーおよびユーザーで構成されるシステムで、取扱いに注意を要する通信やトランザクションの保護にとって重要な、情報のセキュリティを保証する。 | |
| ルート CA（Root CA） | 階層的な公開鍵インフラストラクチャで、セキュリティ・ドメインに対して、最も信頼されるデータとして機能する公開鍵を持つ認証局。 | 認証局（Certificate Authority）, 公開鍵（Public Key） |

**表 F-1 OracleAS Certificate Authority で使用される用語の定義（続き）**

| 用語 | 意味 | 関連項目 |
|---|---|---|
| S/MIME | Secure Multipart Internet Mail Extensions の略で、インターネットの MIME メッセージにデジタル署名と暗号化を追加するプロトコル。 | |
| 下位 CA（Subordinate CA） | 階層的な公開鍵インフラストラクチャにおいて、証明書署名鍵が別の CA に承認され、アクティビティもその CA に制約される CA。 | 認証局（Certificate Authority） |
| X.509 | International Telecommunications Union Telecommunication Standardization Section（国際電気通信連合電気通信標準化部門）の推奨規格で、ディレクトリ形式の一元的な制御パラダイムによる認証サービスを規定するフレームワークを定義する。デジタル証明書および証明書失効リストのフォーマットを定義する規格としては、普及度が最も高い。 | |

# 索引

## 数字

1 行追加， 5-28
2 進数
　鍵， 1-2

## A

ADMIN， A-5
admin.log， 6-14
admin.trc， 6-13, 6-14
AFFILIATION_CHANGE（失効コード）， 3-7
allowExpiredCerts， 5-11
allowRenewal， 5-12
AND， 5-22
Apache， 6-21
　Oracle HTTP Server， 6-3
API， 5-23, 5-30
　プラグイン， 5-3

## B

BASE64 証明書， B-5
BasicConstraintsExtension， B-3

## C

CA， 1-3, A-5, A-10
　下位， 1-3
　階層， B-3
　証明書タイプ， 5-23
　署名， 1-3
　新規
　　新しい署名パスワード， B-4
　ルート， 1-3
　レベル， 1-3
CA SMIME Wallet， 6-2
　アラートと通知の署名， 6-3
　生成， B-6
CA SSL， A-12
CA SSL Wallet， 6-2
　再生成， 6-3
　生成， B-6
CA Wallet
　再生成， 6-2
CA Wallet の再生成， 6-2
CA_COMPROMISE（失効コード）， 3-7
ca_sign
　条件の使用方法のタイプ， 5-23
CASMIME， A-5, A-10
CASSL， A-5, A-10
CA 鍵
　危殆化， 6-2, 6-7
CA 危殆化（失効理由）， 3-13
CA 証明書
　新規， 6-2, A-11
CA 証明書のインポート， 6-6
CA 証明書のダウンロード， 7-13, B-5
CA 署名， 7-12
CA 署名証明書， 6-2
　無効， 6-2, A-11
CA の階層， B-3, B-5
　設定， B-1
CERTIFICATE_HOLD（失効コード）， 3-7
CESSATION_OF_OPERATION（失効コード）， 3-7
changeschema コマンド， A-3
changesecurity， 6-16, A-3
changesecurity コマンド， 6-15, A-3
clear， A-3
CN

DN, 5-24
code_sign
条件の使用方法のタイプ, 5-23
Collaboration Suite, 2-6
convertwallet, 6-6, A-3, A-7
CPS（認証局運用規程）, 4-14
CRL, 2-7, 2-8, 3-11, 3-18, 6-7, 7-3
インポート, 3-18
確認, 3-18
更新, 3-18
コピー, 3-18
サーバーで使用するパス, 3-18
使用方法, 3-18
スケジュールされた生成, 4-6
生成, 3-18
ダウンロード, 3-18
ファイル・システムへのダウンロード, 7-3
複数, 3-18
ブラウザへのインポート, 7-3
用途, 3-15
CRL_SIGN, B-3
CRL から削除（失効理由）, 3-13
CRL のアラート, 4-6
CRL の更新, 2-8, 3-18
CRL のコピー, 3-18
CRL の生成, 2-8, 3-18
CRL のダウンロード, 2-8, 3-18
CRL の次の更新までの日数, 3-18
CRL の有効期間, 3-18
次の更新までの日数, 3-18
cwallet.sso, 6-4, 6-6, 6-21, A-7

## D

DB, A-5, A-10, A-11
dc（ドメイン・コンポーネント）, F-3
Delegated Administration Services, 2-3, 2-6
DIGITAL_SIGNATURE, B-3
DN, 1-4, 2-11, 3-3, 3-4, 3-5, 3-15, 3-16, 3-17, 3-24, 5-10, 5-12, 5-16, 5-22, 5-24, 5-25, 5-31, 5-35, 6-9, 6-12, 6-22, 7-10, C-4, C-8, C-9, F-2, F-3, F-4
dc, F-3
RFC1779 に準拠, 5-24
下位要素による組織の下位部門の作成, F-3
拡張, 3-16, 3-17
完全, 5-24
最小の構成要素, 5-24
最大の構成要素, 5-24
識別名, 3-16
手動の登録情報用のデフォルトの構成, 4-11
照合, 5-24
照合のルール, 5-24
相対, 3-17
ディレクトリ情報のサブツリーのルート, F-3
ディレクトリ編成オブジェクトの識別, F-3
ドメイン・コンポーネント, F-3
部分, 5-24
無効, 5-24
有効, 5-24
ルート, 5-24
ルートに対する連続文字列, 3-15
連続および完全, 5-12
条件, 5-24
DN の最小構成要素, 5-24
DN の最大構成要素, 5-24
DN を使用した検索, 3-16

## E

E-Business Suite, 2-6
ewallet.p12, 6-3, 6-4, 6-6, 6-21, A-7, B-4, B-6

## F

format, A-7

## G

Gemplus, 3-6, 7-5, 7-6
generatewallet, A-3, A-4, A-11, A-12

## H

help, A-3, A-4
HTTP Server, 3-2, 6-18, A-7, B-7
SSL モード, 6-3
HTTP Server（Apache）, 6-21
httpd.conf, 6-21
HTTPS, 2-9, 2-10, 2-12, 6-3, B-6

## I

ias.properties ファイル, 6-16
Identity Management, 1-5, 2-3, 2-5, 2-6

Identity Management インフラストラクチャ， 1-7
「ID/ シリアル番号」， 3-14
IETF， 1-4, 2-7
importwallet， A-3, A-4
IM サービス
　OCA の変更， 6-15, 6-16
Internet Explorer， 2-7, 2-9, 3-6, 7-1, 7-5, 7-6, 7-13, 7-14, 7-17, 7-19
Internet Explorer での証明書発行元への信頼， 7-7

## J

J2EE， 2-5
JAAS， 2-5
jar， 5-15, 5-20, 5-31
Javadoc， 5-30
Java クラス， 5-2, 5-30, 5-31

## K

KEY_CERT_SIGN， B-3
KEY_COMPROMISE（失効コード）， 3-7
KeyUsageExtensions， B-3

## L

LDAP， 1-9, 2-7, A-5
linksso， 3-21, A-3, A-4
LOG， A-5

## M

Microsoft
　Base Cryptographic， 3-6, 7-6
　Enhanced Cryptographic， 3-6, 7-6
　Gemplus， 3-6
mod_osso， E-5
　SSO， 2-9
Mozilla， 7-19

## N

National Language Support（NLS）， 2-8, 6-9
Netscape， 2-9, 3-6, 7-1, 7-5, 7-6, 7-14, 7-16, 7-17, 7-18, 7-19
Netscape Communicator， 2-7
Netscape での証明書発行元への信頼， 7-7
NLS， 2-8, 6-9
NON_REPUDIATION， B-3

## O

OC4J， 2-12, 3-2, 6-18, A-5, A-8, A-15, A-17, B-3, B-4, B-7
　起動および停止， 3-21, 5-32, 6-8, A-8, A-15, B-3
　停止および起動， A-15, B-3
OCA， 1-7, A-5
　リポジトリ， 2-9
oca_cps.html， 4-14
oca/bin， A-2
oca.conf， 6-17, 6-21, A-6, A-17
ocactl， 2-8, 3-2, 3-7, 3-13, 6-2, 6-4, 6-8, 6-18, A-1 ～ A-17, C-7
　SSO との OCA リンクの構成， 3-21
　一般的な形式， A-2
　管理者パスワードの要求， 6-5
　操作およびパラメータ， A-3
oca.trc， 6-13, 6-14
OCA 接続情報
　格納場所および表示場所， 6-17
OCA と SSO のリンク， 3-19
OCA の IM サービスの変更， 6-15, 6-16
OCA リポジトリ， 6-2, A-11
ocm_apache.conf， 6-21
ocmpassword.p12， 6-21
OFF， A-5
OHS， 2-12, 3-2, A-8
　起動および停止， 5-32, A-8, A-15, B-3
　停止および起動， A-15, B-3
OID， 1-9, 2-12, 3-2, 6-17
　SSO の使用， 3-22
ON， A-5
OPMN， 6-3
Oracle Application Server Certificate Authority， 2-6
　必要なコンポーネント， 2-12
Oracle Application Server Certificate Authority インフラストラクチャの再関連付け， 6-15
Oracle Application Server Certificate Authority が信頼されるブラウザの構成， 7-7
Oracle Application Server Certificate Authority の構成操作， 6-5
Oracle Certificate Authority
　OCA， 1-7
Oracle Collaboration Suite， 2-6

Oracle HTTP Server, 6-6
Apache, 6-3
SSL の妥当性の確認, 3-18
Oracle Identity Management, 1-1, 1-5
Oracle Internet Directory, 1-7, 1-9, 2-3, 2-5, 2-10, 3-2, 6-17
SSO の使用, 3-22
Oracle Label Security, 2-5
Oracle Wallet, 1-5
Oracle Wallet Manager, 1-8, B-1, B-5
Oracle Wallet Manager（OWM）, B-5
ORACLE_HOME, 4-14, 5-20, 6-3, 6-6, 6-13, 6-14, 6-21, B-6
OracleAS PKI の概要, 1-7
OracleAS PKI のコンポーネント, 1-7
OracleAS Single Sign-On 認証, 2-10
Oracle ホーム, 2-13
OR 論理式, 5-23
osso.conf, E-5
osso.conf ファイル, 6-21, E-5, E-6
OWM, 1-8, 6-6, B-1, B-5

## P

.p12 ファイル, 7-18
PathLength, D-1
PKCS #10, 7-12, B-5
PKCS #10 証明書要求, 1-8, 2-7, B-1
PKCS #12, 1-8, 6-3, 6-6, 7-18, A-7
PKCS #7, B-2
PKCS 規格, 2-7
PKI, 1-1, 7-12
SSL が必要, 3-19
SSO および OCA での使用, 3-25
SSO での SSL の有効化, E-1
以前のコストおよび問題, 1-7
概要, 1-7
コンテナ, 1-5
コンポーネント, 1-8
資格証明, 1-5
証明書, 1-3
説明, 1-2
操作, 1-5
データの転送および格納の保護, 1-2
利点, 1-6, 1-7
PKIX, 2-7
PKI の利点, 1-6
PKI ベースのシングル・サインオン, 1-9
predicate
RenewalRequestConstraint, 5-12
RSAKeyConstraints, 5-5
鍵のサイズ, 5-5
有効期間, 5-7
Provisioning Integration, 2-6

## R

RA, 1-3, 1-6, 1-7
OCA 内, 1-6
RDN, 3-17, 5-24, F-3
RDN の子, 5-24
最小, 5-24, 5-25
複数の使用, 5-24
REMOVE_FROM_CRL（失効コード）, 3-7
renewalNotAfter, 5-12, 5-16
renewalNotBefore, 5-12
RenewalRequestConstraint, 5-4, 5-16
predicate, 5-12
renewcert, A-3, A-4
RevocationConstraintRule, 5-16
RevocationConstraints, 5-4, 5-10
revokecert, 6-7, A-3, A-4
RFC1779
DN の使用, 5-24
RSA, 2-7, 3-18
RSAKeyConstraints, 5-4
デフォルトの鍵の最小サイズ, 5-5
デフォルトの鍵の最大サイズ, 5-5
RSA 付き MD5, 3-18
RSA 付き SHA1, 3-18

## S

Secure Sockets Layer, 1-9
SSL, 1-9
Secure Sockets Layer（SSL ベース）認証, 2-10
set, A-3, A-5
setpasswd, A-3, A-5, A-9, A-11
Single Sign-On, 2-6
Single Sign-On（SSO）認証, 7-5
Single Sign-On（「SSO」を参照）, 3-19
SMIME, 3-18
SMIME Wallet, 6-2, 6-4
smime_enc

条件の使用方法のタイプ， 5-23
smime_sign
条件の使用方法のタイプ， 5-23
SMIME 電子メールの送信， 6-3
SSL， 1-4, 1-5, 1-9, 2-10, 7-3, 7-4, 7-9, A-9
OCA の使用， 6-3, B-6
PKI に必要， 3-19
SSO での PKI の有効化， E-1
SSO のデフォルトではない， 3-19
公開， 4-10
証明書， 2-11
条件の使用方法のタイプ， 5-23
妥当性の確認， 3-18
認証， 7-3
ポート， 3-8, 3-20
ユーザー
有効期間， 5-7
ユーザーによる更新， 7-11
ユーザーによる失効， 7-11
SSL Wallet， 6-2
SSL サーバー
Wallet のパスワード， 6-6
SSL サーバー Wallet， A-7
SSL 認証
サーバー， 6-3
SSL モード
自動的に構成， 6-6
SSL ユーザーおよび SSO ユーザーの自動的な証明書処理， 7-3
SSO， 1-9, 2-3, 2-8, 2-9, 2-10, 2-12, 3-19, 7-4, 7-5, A-7
mod_osso， 2-9
OCA での PKI の有効化， 3-25
OCA 証明書の使用， 3-21
OCA 証明書の直接取得， 3-19
OCA とのリンク， 3-21
OCA の構成オプション， 3-19
OCA 要求ページのブロードキャスト， 3-19, 3-20
SSL および PKI の有効化， E-1
ssl および pki の有効化， 3-25
Wallet， 6-6
アプリケーションの使用， 3-22
サーバーの再起動， 3-21
証明書の使用， 3-22
デフォルトの配置， 3-19
登録ツール， E-5
ブラウザへの証明書のインポート， 3-22
ユーザー
鍵のサイズの選択， 3-22
有効期間， 5-7
ユーザーによる更新， 7-11
ユーザーによる失効， 7-11
「ようこそ」ページ， 3-21
ログイン・ページ， 7-5
SSO Wallet
暗号化， 6-6
ファイル権限による保護， 6-6
SSO および OCA での PKI 認証の有効化， 3-25
SSO 証明書要求， 3-20
SSO での SSL および PKI の有効化， E-1
SSO とのリンクの削除， 3-21
SSO ユーザーへの OCA 要求ページのブロードキャスト， 3-19, 3-20
SSO 用の ssl および pki の有効化， 3-25
start， A-2, A-3, A-5, A-8
status， A-5
RenewalRequestConstraint， 5-11
RevocationConstraints， 5-11
RSAKeyConstraints， 5-4
UniqueCertificateConstraint， 5-9
有効期間のルール， 5-7
stop， A-3, A-6, A-8
SUBCA， A-4
SUPERSEDED（失効コード）， 3-7


## T


Thawte 社， 1-3
TRACE， A-5
type， A-3, A-10


## U


UniqueCertificateConstraint， 5-4, 5-8
使用方法および DN の確認， 5-8
パラメータ， 5-10
UNIX， 3-8
unlinksso， 3-21, A-3, A-6
UNSPECIFIED（失効コード）， 3-7
updateconnection， 4-12, A-3, A-6, A-17
URL
SSO ユーザー用証明書要求， 3-20
URLC トークン， 3-22

## V

validityPeriod
更新のデフォルト， 5-12
ValidityRule， 5-4, 5-6
VeriSign 社， 1-3

## W

Wallet， 1-8, 6-2, 6-4, A-3, A-12
CA SMIME， 6-3
再生成， 6-2, 6-3, A-11
CA SSL
再生成， 6-2, A-11
Oracle， 1-5
SMIME， 6-4
SSO 形式， 6-6
危殆化または破壊， 6-3, B-6
コンテナ， 1-5
再生成， 6-2, 6-3, B-6
内容， 1-5
バックアップ， 6-6
パスワード， 6-3, 7-18
変更， 6-5
優先されるパスワード， 6-6
wallet-location， A-7
walletwrl， A-7
Wallet 操作， 6-2
Web 管理者の証明書， 3-3, 3-7
失効， 6-8
Web ベースのインタフェース
エンド・ユーザー， 2-8
管理， 2-8
Web ベースの管理インタフェース， 3-1, 3-8
アクセス， 3-3
Windows， 3-8

## X

X.509， 1-4, 1-5, 1-8, 2-1, 2-3, 2-7, 2-9, 2-10, A-15, B-3, D-1, F-1, F-3, F-5

## あ

アイコン
ロック， 7-8, 7-12, 7-18
アスタリスク
条件式， 5-23
属性の一致， 5-23
文字列の不一致， 5-23
値， 5-2
条件， 5-23
パラメータ， 5-17
新しい CA のインストール
手順， 6-7
新しいポリシー・プラグインを作成する手順， 5-31
アプリケーション
SSO の使用， 3-22
アラート， 4-6
CA SMIME Wallet， 6-3
CRL の生成に失敗， 4-6
構成， 4-5, 6-3, 6-4
暗号化， 1-2, 1-4, 1-6, 1-9, 2-9, 7-3
アルゴリズム， 1-2
異なるユーザーに対して一意， 1-2
対称型， 1-2
非対称型， 1-2
方式， 1-2
メッセージ， 1-3
一貫性のない状態
CA の失効後， 6-7
一致
最初、最適ではない， 5-25
条件， 5-22
「一般」サブタブ， 4-7, 4-10
DN のデフォルト， 4-7, 4-10
SSL および SSO， 4-7, 4-10
公開， 4-7, 4-10
設定， 6-17, A-6
データベースおよびディレクトリの設定， 4-7, 4-10
パラメータ， 4-7, 4-10
「一般」サブタブのタスクおよび説明， 4-4
一般名， 3-4, 3-7, 3-14
下位 CA， B-6
委任管理サービス， 1-1
イベント
通知， 4-5
インストール， 2-12
下位 CA Wallet， B-3
インフラストラクチャ， 1-1, 1-5, 2-1, 2-5
再関連付け， 6-15
インポート， 1-8, 3-3, 3-12, 3-15, 7-3, 7-4, 7-7, 7-13, 7-15
CA 証明書， 6-6
下位 CA Wallet， B-3

管理者の証明書， 2-8, 3-3
証明書， 3-21
信頼されるアクティビティ， 7-7
ブラウザ
証明書または CRL， 7-3
埋込み HTML リンク
SSO ユーザー用， 3-20
運用規程， 4-14
要素， 4-14
運用停止（失効理由）， 3-13
影響が大きい操作， 3-13, 6-7
エクスポート， 1-8, 7-16
ブラウザからの証明書， 7-16
エラー， 7-5
演算子
論理， 5-22
エンタープライズ・ユーザー， 2-5
エンティティ， 1-3
関係の保証， 1-2
信頼できる， 1-2
エンド・エンティティ， 3-16, 3-18, 7-1
エンド・ユーザー， 3-16, 7-1
インタフェース， 7-1
エンド・ユーザーによる対話
2 つのタイプ， 7-3
エンド・ユーザー用のタブおよび処理， 7-3
オープン規格， 2-7
オープン規格に対するサポート， 2-7
大文字と小文字を区別しない
条件の文字列， 5-23
オペレーティング・システムのファイル権限
SSO Wallet の保護， 6-3
オペレーティング・システム・ファイル
削除， 6-14

## か

カード・リーダー， 7-5
下位 CA， 1-3, 2-11, 7-12
一般名， B-6
証明書， 7-12
シリアル番号， B-6
新規
以前の SMIME 証明書の無効化， B-6
以前の SSL 証明書の無効化， B-6
シリアル番号， B-4
地理的メリット， 2-11
下位 CA Wallet， B-5
インストール / インポート， B-3
生成， B-5
ディレクトリ， B-4
下位 CA 証明書， 3-12
取得およびインポート， B-1
下位 CA 証明書のインポート， B-1
下位 CA 証明書の取得， B-1
下位 CA による要求
手動， 2-11
階層的な認証局のサポート， 2-11
下位組織
下位 CA Wallet， B-5
解読， 7-3
下位認証局
取得およびインポート， B-1
下位部門
ドメイン・コンポーネント， F-3
鍵， 1-2
2 進数， 1-2
PKI 内， 1-2
検証， 1-3
公開， 1-2, 1-3
個別， 1-2
所有者， 1-3
対称型， 1-2
配布方法， 1-2
非対称型， 1-2
秘密， 1-2
ペア， 1-2, F-3
鍵サイズ， 3-6, 7-5
鍵のサイズ， 3-3, 3-6
predicate， 5-5
RSAKeyConstraints， 5-4, 5-5
最小および最大， 5-4
デフォルト， 5-16
デフォルトの最小， 5-5
デフォルトの最大， 5-5
範囲の調整， 5-16
鍵の長さ， 2-7
鍵のペア， 1-6, 3-6, 7-5, F-3
拡張 DN， 3-17
拡張 DN を使用した検索， 3-17
拡張検索の使用方法， 3-15
拡張領域， 1-4, D-1
カスタマイズ
ポリシー， 2-7

カスタム・ポリシー， 5-30
追加， 5-32
名前、説明およびクラス， 5-32
プラグイン， 5-1, 5-16
カスタム・ポリシー・プラグインの開発， 5-30
カット・アンド・ペースト， 1-7, 1-9, 3-3
完全
DN， 5-24
管理
構成， 3-1
証明書， 3-1, 3-11
ポリシー， 5-1, 5-3, 5-15
概要， 5-2
管理インタフェース， 3-8, 4-2
管理インタフェースの構成， 4-2
管理者
証明書， 2-8, 3-13
新規， 3-7, 6-8
タイプ， A-10
パスワード， 2-8, 3-3, 3-5
フォーム， 2-8
管理者およびエンド・ユーザーにとっての使いやすさ
， 2-8
管理者の証明書， 3-8
インポート， 2-8
管理者パスワード， B-4
ocactl で必要， 6-5
管理タスクの概要， E-1
管理パスワード， 3-7
概念
ポリシー， 5-2
概要
Web ベースの管理インタフェース， 3-8
キー危殆化（失効理由）， 3-13
キーストア， 3-6, 7-6
規格， D-1
期限切れ， 2-6
期限切れの証明書， 3-14, 5-4, 5-11
基準
条件の順序， 5-25
既存の証明書
使用， 4-10
危殆化
CA 鍵， 6-2, 6-7
危殆化された証明書， 3-11, 3-13
起動， 2-8, 3-1, 3-2, 3-8, A-6, C-7
OC4J， 3-21, 5-32, 6-8, A-8, A-15, B-3
OHS， 5-32, A-8, A-15, B-3
局
認証， 1-3
拒否， 2-8, 3-11, 3-12, 3-15
拒否済， 2-8, 3-11, 3-16, 3-17
組込みプラグイン・ポリシー・モジュール， 2-7
クライアント
証明書タイプ， 5-23
クラス， 5-15, 5-20
登録， 5-30
権限， 1-9
権限付きパスワードの変更， A-9
検索， 3-14, 7-4, 7-11
DN の使用， 3-16
拡張， 3-15, 3-16
条件， 3-15
拡張 DN の使用， 3-17
証明書のステータスの使用， 3-17
シリアル番号の範囲の使用， 3-17
条件
DN または DN 構成要素， 3-15
シリアル番号， 3-15
電子メール， 3-15
すべての保留要求， 3-15
単一の証明書または要求， 3-14
単一の発行済証明書， 3-15
単一の要求， 3-15
要求ステータスの使用， 3-16
検索（「リスト」および「検索」を参照）， 3-14
検証
鍵， 1-3
原因
失効， 6-8
公開， 2-6
SSO 証明書， 3-22
SSO ユーザー用の OCA URL， 3-20
証明書， 4-10, 6-17
公開鍵， 1-2, 7-3, 7-12
CA 署名の検証可能， 1-3
暗号化用， 1-2, F-4
所有者， 1-3
公開鍵インフラストラクチャ， 1-1
公開鍵証明書， 1-6
高可用性， 1-1
高可用性機能， 6-1, 6-17
更新， 1-6, 3-11, 3-14, 3-15, 5-4, 5-11, 5-12, 5-16, 6-4, 7-3,
7-11

可否またはタイミング， 5-16
期限切れの証明書， 5-4, 6-4
重要な Wallet， 6-4
すべてのポリシー・ルール， 5-14
デフォルトの有効期間， 5-11, 5-12, 5-16
ポリシー， 5-16
更新の時間枠， 3-11, 3-14, 5-11, 5-12, 5-16
構成
ocactl の使用， 6-5
Oracle HTTP Server， 6-6
Web， 6-5
下位 CA， B-5
現場， 6-5
コールド・フェイルオーバー， 6-18
署名されたアラートと通知の送信， 4-5, 6-3, 6-4
自動的に SSL， 6-6
ログおよびトレース， 4-11
構成オプション， 3-19
構成管理， 3-1
アラート， 4-6
サブタブ， 4-3
タブ， 4-3
構成タスク， 4-4
構成ファイル， 6-21, A-6, A-8
構成変更， C-7
高度なトピック， 6-1
構文， A-3, A-7
コード
失効， 3-7
コード署名， 7-3
証明書， F-2
コールド・フェイルオーバー
構成， 6-18
配置， 6-18
コピー
BASE64 証明書， B-5
CRL， 3-18
トラスト・ポイント， B-5
コマンド， A-3
changeschema， A-3
changesecurity， A-3
clear， A-3
generatewallet， A-3
help， A-3
importwallet， A-3
linksso， A-3
renewcert， A-3
revokecert， A-3
set， A-3
setpassword， A-3
start， A-3
stop， A-3
unlinksso， A-3
有効， 6-5
コマンドライン・インタフェース， 3-1
コマンド例， A-7
コンテナ， 1-8
PKI， 1-5
Wallet， 1-5
証明書用， 1-5
データベース、キャッシュまたは Wallet， 1-5
内容， 1-5
コンポーネント
OCA に必要， 2-12
OracleAS PKI， 1-8


## さ


サーバー， 3-16
SSL 認証， 6-3
証明書， 5-6, 7-3, 7-12
タイプ， 7-12
証明書タイプ， 5-23
複数， 3-18
サーバー・エンティティ， 7-1
検証， 3-18
サーバー / 下位 CA
証明書要求， 7-12, 7-13, B-2, B-5, B-6
登録フォーム， 7-12, 7-13, B-2, B-5, B-6
「サーバー / 下位 CA 証明書」タブ， 2-8, 7-4, 7-12
サーバー証明書の取得， 7-12
サーバーによる要求
手動認証， 2-11
サーバーの証明書
取得， 7-12
再関連付け
インフラストラクチャ， 6-15
リポジトリ， 6-15
再起動， 3-2, 3-7, A-6, C-7
SSO Server， 3-21
最小の RDN， 5-25
再生成
CA SMIME Wallet， 6-2, 6-3, A-11
CA SSL Wallet， 6-2, 6-3, A-11

CA SSL 証明書
状況， B-6
CA Wallet， 6-2
CA 署名証明書， 6-2
Wallet， 6-2, 6-3, B-6
再登録
SSO で OCA， E-5
削除， 5-16
オペレーティング・システム・ファイル， 6-14
条件， 5-17
ポリシー， 5-15
サブスクライバ名， 3-22
サブタブ， 3-9, 5-14
「一般」， 4-7, 4-10
資格証明
PKI， 1-5
式
演算子， 5-22
条件， 5-2, 5-22
完全， 5-12
連続， 5-12
論理， 5-22
識別情報， 1-3, 1-7
識別情報管理， 2-1
ソリューション， 2-2
識別名， 3-16, 5-24
DN， 1-4
識別名（DN）， 1-4, F-2
施行
ポリシー， 5-3
失効， 1-6, 2-6, 2-8, 2-10, 3-7, 3-11, 3-13, 3-15, 7-3, 7-5, 7-11
Web 管理者の証明書， 6-8
新しい CA のインストール前に実行， 6-7
期限切れの証明書， 5-10, 5-16
原因， 6-8
すべてのポリシー・ルール， 5-14
認証局の証明書， 6-7
理由， 3-7
ルート認証局の証明書， 6-7
失効された CA
管理者がアクセスできない， 6-7
失効済， 3-15
失効済証明書
リスト， 3-15
失効理由， 3-13
指紋
証明書， 1-4
集中管理， 1-1
手動， 7-4
認証， 7-10
手動による承認， 2-10, 7-3
サーバーおよび下位 CA， 2-11
追加オプション， 2-11
必要な情報， 2-11
手動認証， 7-10
手動認証ユーザーの証明書， 5-6
消去
ログ・データまたはトレース・データ， 6-14
ログまたはトレース
内容の消去， 6-14
照合
DN， 5-24
DN のルール， 5-24
一致しない場合の結果， 5-25
ポリシーの評価， 5-24
詳細表示， 3-12, 3-15
承認， 2-8, 3-11, 3-12, 3-15
手動， 7-3
使用方法
CA 署名， B-5
条件， 5-23
証明書， 3-13, F-1
BASE64， B-5
PKCS #10 要求， 2-7
PKI， 1-3
SMIME の無効化， B-6
SSL， 1-4
SSL の無効化， B-6
SSL ユーザーおよび SSO ユーザーの自動的な処理， 7-3
SSO/SSL 認証ユーザーの要求による発行， 4-10
SSO の公開， 3-22
SSO の使用， 3-21, 3-22
SSO 用の要求 URL， 3-20
X.509， 1-4, 1-5, 1-8, 2-1, 2-3, 2-7, 2-9, 2-10, A-15, B-3, D-1, F-1, F-3, F-5
新しい CA， 6-2, A-11
新しく必要， 6-7
一貫性のない状態， 6-7
インポート， 3-7, 3-21, 7-3
下位 CA， 3-12
拡張領域， 1-4
管理， 3-1, 3-11

管理者， 3-8, 3-13
管理者の置換， 3-7
管理者の要求， 3-3
期限切れ， 3-14, 5-4, 5-11, 6-4
期限切れの失効， 5-10
既存の使用， 4-10
危殆化， 3-11, 3-13
拒否， 3-12
検索， 3-14
公開， 4-10, 6-17
更新， 3-14, 6-4, 7-3, 7-11
更新の時間枠， 3-11, 3-14, 5-11, 5-12, 5-16
サーバー， 5-6, 7-3, 7-12
サーバー / 下位 CA， 7-12
サーバー，取得， 7-12
失効， 3-13, 7-3, 7-11
指紋， 1-4
取得， 2-10, 7-11
手動認証， 5-6
詳細の表示， 3-12
所有者， 3-16
署名， 1-4, 7-3
署名者， 7-6, 7-9
シリアル番号， 1-4
新規要求， 7-3
信頼できる， B-5
　使用方法の編集， 7-6, 7-7
条件のタイプ， 5-17, 5-23
ステータス， 3-16, 3-17
すべて無効， 6-2, A-11
タイプ， 7-3
ダウンロード， 7-3
デジタル， 1-3
内容， 1-4
内容および使用方法， 1-4
パラメータ値
　制限， 5-2
必要な管理者情報， 3-6
表示， 7-3
ファイル・システムへのインポート， 7-18
ファイル・システムへのダウンロード， 7-3
複数， 5-4
複数の制約， 5-8
ブラウザへのインポート， 3-3, 7-3
プロパティ， 2-7
別々， 1-4
保留要求のアラート， 4-6
ポリシー， 5-2
無効化， 6-7
ユーザー， 7-4
要求， 1-8, 2-7
　SSO， 3-20
　ステータス， 2-8
　保留， 3-10
用途， 2-10
ライフ・サイクル， 1-9
ルート CA， 3-13
証明書失効リスト， 3-18, 6-7, F-1
証明書失効リスト（CRL）， 2-8
証明書失効リスト（CRL）のダウンロード， 7-13, 7-15
証明書のインポート， 3-7
証明書の管理， 3-1, 3-11
証明書の検索，更新および失効， 7-11
証明書の更新， 3-14, 7-11
証明書の失効， 3-13, 7-11
証明書の取得， 2-10, 7-11
証明書の詳細の表示， 3-12
証明書の使用方法
　条件， 5-23
証明書の署名， 2-11
証明書のステータスを使用した検索， 3-17
証明書の保留要求， 3-10
証明書要求の拒否， 3-12
「証明書要求の詳細」画面， 3-12
証明書要求の承認， 3-12
証明書要求の承認または拒否， 3-12
「証明書要求」フォーム， 7-5
証明書要求または発行済証明書の表示， 3-14
所有者， 3-16
所在地
　ドメイン・コンポーネント， F-3
所属変更（失効理由）， 3-13
署名， 1-3, 2-9, 7-3, 7-6, 7-12, A-2, A-11
　コード， 7-3
　証明書， 7-3
　ソフトウェア， 7-3
　デジタル， 1-1, 1-4
　認証局， 1-3
　メッセージ・ダイジェスト， 7-3
署名アルゴリズム， 3-18
署名者， 7-6, 7-9
シリアル番号
　新しい下位 CA， B-4
　下位 CA， B-6

証明書， 1-4
範囲， 3-16
範囲の検索， 3-17
シリアル番号検索， 3-15
シリアル番号の範囲を使用した検索， 3-17
シングル・サインオン， 1-1, 1-7, 1-9, 2-3
信頼
パス， 2-11
レベル， 1-3
信頼できるエンティティ， 1-2, 1-3, 3-12
信頼できる環境， 3-18
信頼できる証明書， B-5
使用方法の編集， 7-6, 7-7
信頼できる証明書の DN
要求の許可または禁止， 5-16
信頼の階層， 1-3, 2-11
地理的に分散， 2-11
時間枠
更新， 3-11, 3-14, 5-11, 5-12, 5-16
実行（[Enter] ではない）， 3-15
自動認証のクライアント・ユーザー， 5-7
柔軟なポリシー， 2-7
「述語式」テキスト・ボックス， 5-17
上位， B-5
条件， 5-2, 5-17
値， 5-23
アスタリスク， 5-23
一致しない場合， 5-25
演算子， 5-22
カスタム・ポリシー以外， 5-22
削除， 5-17
式， 5-2
使用される対応値， 5-22
証明書タイプ， 5-23
順序， 5-25
属性， 5-23
追加， 5-28
特定， 5-22
並び替え， 5-26
任意， 5-22
複雑， 5-5
複数， 5-23
評価の例， 5-25, 5-26
複数のセット， 5-5
ポリシー， 5-15
文字列
大文字と小文字を区別しない， 5-23
要求の要素と一致， 5-22
例， 5-5
条件式
完全， 5-12
評価， 5-22
不一致， 5-22
連続， 5-12
論理， 5-22
条件の順序， 5-25
基準， 5-25
条件の属性， 5-23
条件の追加， 5-28
条件の並び替え， 5-26
条件は上から順に評価， 5-26
状態， A-9
情報メッセージ， 5-20
ジョブ
スケジュールされた， 4-6
推奨の配置， 2-13
インストール手順， 2-13
メリット， 2-13
スケーラビリティ， 1-1
スケーラビリティ、パフォーマンスおよび高可用性
， 2-9
スケジュールされたジョブ， 4-6
ステータス， 3-2
証明書
有効，失効済，満了， 3-16, 3-17
認可済，拒否済または保留， 3-15
すべての保留要求， 3-14
スマートカード， 2-7, 2-9, 3-6, 7-5
制限
証明書のパラメータ値， 5-2
整合性， 1-6
生成
下位 CA Wallet， B-5, B-6
製品に付属の証明書更新ポリシー， 5-16
製品に付属の証明書失効ポリシー， 5-16
製品に付属の証明書要求ポリシー， 5-16
制約固有のデフォルトのポリシー・ルール， 5-4
セキュリティ・アイコン， 7-16
セキュリティ・ポリシー， 2-10
セッション鍵管理， 1-9
接続
OCA リポジトリおよびディレクトリ， 6-17
ノードまたはポートの変更， A-6
接続情報の表示， 6-17

接続情報
格納場所および表示場所， 6-17
文字列の変更， A-6
接続情報の格納， 6-17
接続文字列， C-7
設定
「一般」サブタブ， 6-17, A-6
使用するディレクトリのホスト / エージェント / ポート， 4-12
データベース， 4-12
セルフ・サービス， 2-6
相互運用性， 1-8
操作， A-3
PKI， 1-5
送信
署名されたアラートと通知， 4-5, 6-3, 6-4
相対 DN， 3-17
相対識別名， 5-24
ソフトウェア
署名， 7-3
属性， 1-9
アスタリスクの一致， 5-23
条件， 5-23

## た

対称型， 1-2
対象名， 3-4
タイプ
証明書， 7-3
条件， 5-23
タスク
「一般」サブタブ， 4-4
構成， 4-4
「通知」サブタブ， 4-4
「ポリシー」サブタブ， 4-4
タブ， 2-8
管理設定， 2-8
認証管理， 2-8, 3-10
単一の証明書または要求
検索， 3-14
第三者， 7-12
SSL Wallet， 6-6
信頼できる， 1-3
第三者機関の Wallet， A-7
ダウンロード， 7-13
CA 証明書， 7-3
CRL， 7-3
ファイル・システム
証明書または CRL， 7-3
置換
管理者の証明書， 3-7
追加， 5-16, 5-17
カスタム・ポリシー， 5-32
ポリシー， 5-15, 5-30
通信の保護， 1-1
通知
CA SMIME Wallet， 6-3
イベント， 4-5
構成， 4-5, 6-3, 6-4
「通知」サブタブ， 4-5
「通知」サブタブのタスクおよび説明， 4-4
停止， 2-8, 3-1, 3-2, 3-7, A-6, C-7
OC4J， 3-21, 5-32, 6-8, A-8, A-15, B-3
OHS， 5-32, A-8, A-15, B-3
適用
ポリシー， 5-3
ポリシーのデフォルトの値， 5-25
ディレクトリ
下位 CA Wallet， B-4
接続， 6-17
ディレクトリ・サービス， 1-1
ディレクトリ情報のサブツリーのルート
DN， F-3
ディレクトリ統合サービス， 1-1
ディレクトリの設定， 4-12
ディレクトリの同期
スケジュールされた， 4-6
ディレクトリ編成オブジェクト， F-3
DN， F-3
データ整合性， 1-1
データベース
使用する接続文字列， 4-12
データベース接続プール， A-6, A-8
データベースの設定， 4-12
デジタル証明書， 1-3, 1-6
SSL， 2-9
暗号化， 2-9
管理， 3-11
拒否， 3-12
更新， 3-14
失効， 3-13
署名， 2-9
署名 /SSL， 2-11

内容および使用方法， 1-4
バイナリ・ファイル， A-11
表示， 3-12
保留， 2-9
要求， 2-7, 2-8, 2-9, 2-10, 2-11
要求の承認， 3-12
デジタル署名， 1-1, 1-4, 1-6, 1-7, 2-7
デジタル・トランザクション
署名， 1-6
デジタル・トランザクションへの署名， 1-6
デフォルト， 5-2, 5-17
鍵のサイズ， 5-16
更新有効期間， 5-12
ポリシー， 5-4
ポリシー内
使用時， 5-22
有効期間， 5-16
デフォルトの配置， 2-12
インストール手順， 2-12
メリット， 2-12
デフォルトのベース DN コンポーネント， 4-11
デフォルトのポリシー・ルール， 2-7
デフォルトの有効期間
更新， 5-11, 5-12, 5-16
電子メール， 3-12, 4-5
OCA URL の SSO ユーザー， 3-20
サーバー，送信者，テンプレート， 4-5
電子メール・アドレス検索， 3-15
電子メールのクライアント
CRL の使用， 3-18
受信した SMIME メッセージの検証， 3-18
電子メールの保護， 2-6
伝播， 2-6
透過的， 2-6
登録
クラス， 5-30
ポリシー・プラグイン， 5-3
登録局， 1-6, 1-7
RA， 1-3
登録ツール
SSO， E-5
登録フォーム
サーバー / 下位 CA， 7-12, 7-13, B-2, B-5, B-6
トラスト・ポイント， 6-6, B-1
コピー， B-5
トレース， 4-11, 6-13
oca.trc， 6-14
消去， 6-14
トレース出力， A-6, A-8
トレース・ファイル， 4-11
同期
ディレクトリ， 4-6
ドメイン・コンポーネント， 2-11
組織の下位部門や所在地， F-3
属性， F-3
ドメイン・コンポーネント、例， F-3

## な

内容
コンテナ， 1-5
証明書， 1-4
名前
証明書署名者， 7-6, 7-9
名前付け
ポリシー・プラグイン， 5-3
並び替え， 5-16, 5-17
ポリシー， 5-15
ニックネーム， 3-22
認可済， 2-8
認証， 1-2, 1-6, 1-7, 1-9, 2-6, 2-9, 3-23, 7-1, F-1
CRL の確認， 3-18
mod_osso， 2-9
SSL， 7-3, 7-9
SSL および SSO の構成， 4-10
SSL サーバー， 6-3
SSL ベース， 2-10
SSO， 3-21
クライアントの証明書， 3-7
手動， 7-10
証明書ベース， 2-10
パスワード・ベース， 2-10
フォーム， 3-3
方式の変更， 2-8, 7-3
ユーザー， 3-12
「認証管理」タブ， 2-8, 3-10
認証局， 1-3, 1-7, F-1
CA， 1-3
署名， 1-3
認証局運用規程， 4-14
認証局の SSL 証明書および Wallet の再生成， A-12
認証済， 3-11, 3-16, 3-17
ノード
変更， A-6

## は


配置， 2-12
　コールド・フェイルオーバーの使用， 6-18
　推奨， 2-13
　　インストール手順， 2-13
　　メリット， 2-13
　デフォルト， 2-12
　　インストール手順， 2-12
　　メリット， 2-12
　方法， 2-12
破棄（失効理由）， 3-13
範囲， 5-2
バックアップ
　Wallet， 6-6
バックアップおよびリカバリ
　考慮事項， 6-19
バックアップとリカバリの手順， 6-1
パス
　CRL， 3-18
パス長， 3-12
　下位 CA のレベルの数， B-5
パスワード， 3-7, 7-16, 7-18, A-2, A-8, A-10, A-12
　CA， 6-5
　CASMIME， 6-5
　CA SSL Wallet， 6-5
　DB， A-11
　SSL サーバー Wallet， 6-6
　Wallet， 6-3, 7-18
　　変更， 6-5
　管理者， 2-8, 3-2, 3-3, 3-5, 3-7, B-4
　　ocactl で必要， 6-5
　新規， A-10
　ストア， B-4
　生成中に要求， 6-2, A-11
　データベース， 6-5
　秘密鍵の暗号化， 6-2, A-11
　紛失， 6-8
　ブラウザのセキュリティ， 3-6, 3-7
　変更， A-10
パスワード・ストア， A-11
パスワードの変更， 6-5
パラメータ， 5-2, 5-17, A-3
　allowExpiredCerts， 5-11
　値， 5-17
　デフォルト，範囲および値， 5-2
　ポリシー， 5-15
　有効期間の制約， 5-6, 5-7
非対称型， 1-2
必要なフィールド， 2-10
等しい， 5-22
等しくない， 5-22
否認防止， 1-1, 1-6
　署名されたメッセージ， 1-2
秘密鍵， 1-2, 1-6, 3-13, 7-3, 7-11, 7-17, 7-18, F-4
　新しい CA， 6-2, A-11
　暗号化， 6-2, A-11
　危殆化， 3-7, 6-8
　公開鍵を使用した検証， 1-3
　証明書への署名， 1-3
　盗難， 3-7, 6-8
　復号化用， 1-2
　紛失， 3-7
　紛失したパスワード， 6-8
評価
　複数の条件， 5-25
評価の例
　複数の条件， 5-25, 5-26
表示， 3-12, 7-3
　ログまたはトレース， 4-11
ビッグ・エンディアンの順序， 5-24
ビット
　拡張領域の設定， B-3
ピアの識別情報， 1-5
ファイル
　admin.log， 6-14
　admin.trc， 6-13, 6-14
　cwallet.sso， 6-21
　ewallet.p12， 6-21
　httpd.conf， 6-21
　ias.properties， 6-16
　oca_cps.html， 4-14
　oca.conf， 6-17, 6-21
　oca.trc， 6-13, 6-14
　ocm_apache.conf， 6-21
　ocmpassword.p12， 6-21
　osso.conf， 6-21, E-5, E-6
　.p12， 7-18
　オペレーティング・システム， 6-14
　トレース， 4-11
　ログ， 4-11
ファイル権限
　SSO Wallet の保護， 6-6
ファイル・システムからの証明書のインポート， 7-18

フィールド名
フォーム， 3-4
フォーム
管理者， 2-8
認証， 3-3
フィールド名， 3-4
復号化， 1-2, 7-3
時間と手間， 1-6, 1-9
実行不可能， 1-9
適切な受信者のみ， 1-2
メッセージ， 1-3
複数
CRL， 3-18
条件， 5-5
複数のサーバー， 3-18
複数の証明書， 5-4
同じ使用方法， 5-16
許可または禁止， 5-16
制約， 5-8
複数の条件， 5-23
評価の例， 5-25, 5-26
複数の条件による評価， 5-25
複数の条件による評価の例， 5-25
不正アクセス， 1-6
防止， 1-2
部門
下位 CA Wallet， B-5
ブラウザ， 1-8, 2-7
CRL の使用， 3-18
SSO に対する証明書の表示， 3-22
SSO 証明書のインポート， 3-22
構成， 7-7
証明書のインポート， 3-21
パスワード， 3-6, 3-7
ブラウザへのインポート， 7-6
SSO， 3-22
ブラウザへのインポート（CRL）， 3-18
ブラウザへの証明書のインポート， 7-15
プライベート・メッセージ， 1-2
プラグイン， 5-1, 5-2, 5-3, 5-23, 5-30, 5-31
jar， 5-15
カスタム
例， 5-30
カスタム・ポリシー， 5-16
クラス， 5-15
デフォルト， 5-30
プラグイン・ポリシー・モジュール， 2-7
プロトコル
PKCS #10， 2-7
Signed Public Key and Challenge， 2-7
プロパティ
証明書， 2-7
プロパティ・ファイル， 6-16
プロビジョニング， 2-10
自動， 2-9
従来型， 2-9
変更
Wallet のパスワード， 6-5
認証方式， 7-3
ポートまたはノード， A-6
ポリシー， 5-15
要求， 5-4
編集， 5-16
信頼できる使用， 7-6, 7-7
「ポリシー」サブタブ， 5-2
ホームページ， 3-8, 7-2
ホストのポート番号， 3-20
保留， 2-8, 3-11, 3-16, 3-17
保留（失効理由）， 3-13
防止
署名されたメッセージの否認， 1-2
不正アクセス， 1-2
傍受者， 1-2
ポート， 3-3, 3-8, 7-2
SSL， 3-20
情報， 3-8
変更， A-6
ホスト， 3-20
リスト， 3-8
ポリシー， 2-1, 2-7, 2-10, 3-3
jar， 5-15
Java クラス， 5-2
RenewalRequestConstraint， 5-4, 5-11
RevocationConstraints， 5-4, 5-10
RSAKeyConstraints， 5-4
UniqueCertificateConstraint， 5-4, 5-8
ValidityRule， 5-4
上書き
証明書の発行時， 5-15
オブジェクト・クラス， 5-20
カスタム， 5-30
条件なし， 5-22
カスタムの例， 5-16
カスタム・プラグイン， 5-1

管理， 5-1, 5-2, 5-3, 5-15
概念、用語および定義， 5-2
クラス， 5-15
更新， 5-16
削除， 5-15, 5-17
削除（カスタムのみ）， 5-17
作成
手順， 5-31
様々なユーザー用， 5-22
施行， 5-3
指定対象， 5-15
処理， 5-3
順次， 5-3
柔軟， 2-7
順序， 5-3, 5-15
条件， 5-2, 5-15
セキュリティ， 2-7, 2-10
説明， 5-20
追加， 5-15
追加（カスタムのみ）， 5-20
提供， 5-4
提供されるルール， 5-4
定式化および適用， 5-2
適用， 5-3
デフォルト
使用時， 5-22
デフォルトのルール， 5-4
名前， 5-20
並び替え， 5-15, 5-18
認証局の運用， 4-14
パラメータ， 5-15
パラメータ値の制限， 5-2
プロセッサ・モジュール， 5-3
変更には再起動が必要， 5-15
編集， 5-17
無効化， 5-15
有効化， 5-17
要求の変更， 5-4
要求の評価， 5-2
ルール， 5-2
ポリシーの上書き
証明書の発行時， 5-15
「ポリシー」サブタブ， 5-2, 5-14
「ポリシー」サブタブのタスクおよび説明， 4-4
ポリシー操作
編集，有効化，無効化，削除，並び替え，追加
，5-16
ポリシーの削除， 5-17
ポリシーの順序， 5-3
ポリシーの追加（カスタムのみ）， 5-20
ポリシーのデフォルトの値
適用， 5-25
ポリシーの並び替え， 5-18
ポリシーの評価
DN の照合， 5-24
ポリシーの表示， 5-14
ポリシーの編集， 5-17
ポリシーの有効化， 5-17
ポリシー・プラグインの作成， 5-3
ポリシー・モジュール， 2-7
カスタマイズ， 2-7
ポリシー・ルール
作成， 5-3
すべての更新， 5-14
すべての失効， 5-14
すべての要求， 5-14
複数の条件， 5-25
プラグイン， 5-3
有効化、無効化または変更， 5-2
ポリシー・ルールの条件， 5-21
ポリシー・ルールの変更， 5-2
ポリシー・ルールの無効化， 5-2
ポリシー・ルールの有効化， 5-2
「ポリシーを適用」チェック・ボックス， 5-15

## ま

未指定（失効理由）， 3-13
無効化， 5-16
RenewalRequestConstraint， 5-11
RevocationConstraints， 5-11
RSAKeyConstraints， 5-4
UniqueCertificateConstraint， 5-9
証明書， 6-7
ポリシー， 5-3, 5-15
有効期間のルール， 5-7
メッセージ
秘密， 1-2
変更の表示， 5-20
メッセージ・ダイジェスト
署名， 7-3
文字列の値， 5-23
問題， 1-1

## や


有効化， 5-16
  RenewalRequestConstraint， 5-11
  RevocationConstraints， 5-11
  RSAKeyConstraints， 5-4
  UniqueCertificateConstraint， 5-9
  ポリシー・プラグイン， 5-3
  有効期間のルール， 5-7
有効期間， 3-3, 3-6, 3-12, 3-15, 5-4, 7-6, 7-12
  CA 用， 5-8
    デフォルト， 5-8
  predicate， 5-7
  renewcert， 6-4
  SSL または SSO の認証済ユーザーの場合， 3-14
  拒否， 5-6
  最小および最大， 5-6
  デフォルト， 5-16
  デフォルトの最小， 5-7
  デフォルトの最大， 5-7
  デフォルトの有効期間， 5-7
  範囲の調整， 5-16
ユーザー・インタフェース
  CA 証明書のダウンロード， 7-13
  CRL のダウンロード， 7-13, 7-15
  OCA が信頼されるブラウザの構成， 7-7
  SSL， 7-9
  SSO， 7-5
  アクセス， 7-2
  エンド・ユーザー用のタブおよび処理， 7-3
  下位 CA 証明書， 7-12
  「サーバー / 下位 CA 証明書」タブ， 7-12
  手動認証， 7-10
  証明書の更新， 7-11
  証明書の失効， 7-11
  証明書の取得， 7-11
  証明書の操作， 7-11
  ファイル・システムからの証明書のインポート
    ， 7-18
  ブラウザからの Wallet のエクスポート， 7-16
  ブラウザへの証明書のインポート， 7-15
  「ユーザー証明書」タブ， 7-4
ユーザー・インタフェースへのアクセス， 7-2
「ユーザー証明書」タブ， 2-8
「ユーザー証明書」ページ， 2-8
要求， 1-8, 2-7, 2-8, 2-9, 2-10, 2-11, 3-3, 3-11, 3-12, 3-16,
  7-4
  CA 署名， 7-12
  SSL/ 暗号化， 7-12
  コード署名， 7-12
  署名， 7-12
  新規， 7-3
  妥当性， 5-2
  保留， 3-10
  ポリシーによる変更， 5-4
  ポリシーの拒否， 5-3
  ポリシーの対象， 5-3
要求ステータスを使用した証明書要求の検索， 3-16
要求の評価
  ポリシー， 5-2
「ようこそ」ページ， 3-3
  SSO ユーザー用， 3-21
用語集， F-1
要素
  運用規程， 4-14
  ログに存在， 4-13


## ら


リスト， 3-14
  失効済証明書， 3-15
  ポート， 3-8
利点
  OracleAS PKI， 1-7
リトル・エンディアンの順序， 5-24
リポジトリ， 2-9, 2-10, 2-12, 3-2
  OCA， 6-2, A-11
  再関連付け， 6-15
  接続， 6-17
  別々， 6-15
  ログを含む， 6-14
理由コード
  失効， 3-7
ルート， 2-11, 7-12, A-11
  CA， 1-3
ルート CA
  証明書， 3-13
ルート CA Wallet， B-5
ルート・ストア， 7-7
ルート認証局（CA）， 6-2
ルート認証局の証明書の再生成， A-11
例
  条件での DN の照合， 5-24
レベル

CA, 1-3
信頼, 1-3
連続
DN, 5-24
連続した DN, 5-12
連続文字列, 3-15
ローカル・エントリ名, 5-24
ローカル ディスクにダウンロード（CRL）, 3-18
ロール, A-5, A-10
ロギング, 4-11
ログ, 6-13
OCA の使用中のエラー・メッセージ, 4-13
消去, 6-14
表示, 3-1, 4-13
要素, 4-13
リポジトリに格納, 6-14
ログ出力, A-6, A-8
ログの表示, 3-1
「ログの表示」タブ, 4-13
ログ・ファイル, 4-11
ロック・アイコン, 7-8, 7-12, 7-18
論理
演算子, 5-22
論理式
条件で使用, 5-22

## わ

ワンタイム・セッション・パスワード, 1-9