IBM® WebSphere® Commerce

セキュリティー・ガイド

バージョン 5.4

IBM® WebSphere® Commerce

セキュリティー・ガイド

バージョン 5.4

ご注意!

本書および本書で紹介する製品をご使用になる前に、 125 ページの『特記事項』に記載されている情報をお読みください。

本書の内容は、新版で特に指定のない限り、IBM® WebSphere Commerce バージョン 5.4 以降のすべてのリリースおよびモディフィケーションに適用されます。 製品のレベルにあった版を使用していることをご確認ください。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

http://www.ibm.com/jp/manuals/main/mail.html

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは
http://www.ibm.com/jp/manuals/ の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

原 典 :	IBM WebSphere Commerce Security Guide Version 5.4
発 行 :	日本アイ・ビー・エム株式会社
担 当 :	ナショナル・ランゲージ・サポート

第1刷 2002.6

この文書では、平成明朝体™W3、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財) 日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W9、平成角ゴシック体™W3、
平成角ゴシック体™W5、平成角ゴシック体™W7

目次

まえがき

本書は、WebSphere Commerce 5.4 のセキュリティー・フィーチャーについて、およびそのフィーチャーの構成方法について説明します。

本書は、認証、許可、およびアクセス・コントロール・ポリシーなどの、WebSphere Commerce のセキュリティー上の懸案事項を詳述しています。本書の目的は、それぞれのサイトのセキュリティー担当者 (システム管理者や WebSphere Commerce サイト管理者も含まれると想定されます) によって、 WebSphere Commerce の実動サイトを確実に安全化するのに役に立つ包括的な資料として用いられることにあります。

本書の対象読者は、WebSphere Commerce サイトのセキュリティー担当責任者またはセキュリティー管理者です。

本書の記載内容の多くは、WebSphere Commerce 5.4 のオンライン・ヘルプ、*WebSphere Commerce インストール・ガイド、バージョン 5.4*、*WebSphere Commerce プログラマーズ・ガイド、バージョン 5.4* などの、 WebSphere Commerce 5.4 情報ライブラリー内のその他の資料から転載されていることに注意してください。具体的には次のとおりです。

- 15 ページの『第 3 章 許可 (アクセス・コントロール)』の内容は、*WebSphere Commerce アクセス・コントロール・ガイド、バージョン 5.4* にも記載されています。
- 43 ページの『第 4 章 サイト・セキュリティーの機能強化』と 71 ページの『第 6 章 セッション管理』の内容は、 WebSphere Commerce 5.4 オンライン・ヘルプにも記載されています。 61 ページの『第 5 章 WebSphere Application Server のセキュリティーの使用可能化』の内容は、*WebSphere Commerce インストール・ガイド、バージョン 5.4* にも記載されています。
- 77 ページの『第 3 部 システム管理者のセキュリティー・タスク』の内容は、*WebSphere Commerce インストール・ガイド、バージョン 5.4* にも記載されています。
- 97 ページの『第 4 部 WebSphere Commerce 開発者のセキュリティー・タスク』の内容は、*WebSphere Commerce プログラマーズ・ガイド、バージョン 5.4* にも記載されています。

重要

本書では、e-commerce サイトの配置に関連した WebSphere Commerce のセキュリティー上の案件のみを取り上げています。オペレーティング・システムの弱点に関する内容は述べていません。オペレーティング・システムを安全化するのに講じる必要のある対策を確かめるには、オペレーティング・システムのベンダーに問い合わせてください。

本書の編成

本書は次のような内容に分かれています。

- 1 ページの『第 1 部 WebSphere Commerce セキュリティー・モデル』は、WebSphere Commerce セキュリティー・モデルを取り上げ、 WebSphere Commerce のセキュリティーの概念について概略しています。第 1 部は、WebSphere Commerce セキュリティーの一般概要を知りたい人や、 WebSphere Commerce サイトのセキュリティーを計画する人すべてにとって必読の項です。
- 41 ページの『第 2 部 WebSphere Commerce サイト管理者のセキュリティー・タスク』は、サイト・セキュリティーにまつわる種々の WebSphere Commerce サイト管理タスクを解説しています。第 2 部は、サイト・セキュリティー関連のサイト管理タスクを担うすべての人を対象に説明しています。
- 77 ページの『第 3 部 システム管理者のセキュリティー・タスク』は、サイト・セキュリティーに付随した種々の WebSphere Commerce システム管理タスクを解説しています。第 3 部は、システム管理タスクを担当している人と、システム・セキュリティーに取り組んでいる人すべてに有用な解説です。
- 97 ページの『第 4 部 WebSphere Commerce 開発者のセキュリティー・タスク』は、開発者の観点にたって WebSphere Commerce アクセス・コントロールを説明しています。第 4 部は、アクセス・コントロールの概念を理解したいすべての人と、コード内にアクセス・コントロール・ポリシーを実装するすべての人の役に立ちます。

継続的セキュリティー評価

WebSphere Commerce 製品ラインに関しては、 IBM セキュリティーの専門家から成る独立グループが実施するセキュリティー分析が絶えず行われています。そのような専門家は、ブラウザーを使って WebSphere Commerce にアクセスするだけのユーザーから、 WebSphere Commerce サーバーが稼働するのと同じシステム上にアカウントを有するもっと高い特権のユーザーにいたるまでの観点でセキュリティー分析を行っています。このセキュリティー専門家の分析によるフィードバックが、WebSphere Commerce のセキュリティーを高めるために継続的に使用されています。

WebSphere Commerce 5.4 でのセキュリティーの向上

以下の項では、WebSphere Commerce Suite 5.1 から見て WebSphere Commerce 5.4 において強化されたセキュリティーの内容を一覧で示しています。この強化内容の大半は、WebSphere Commerce Business Edition 5.1 リリースで行われたものです。この機能強化は概して以下の担当者を対象とします。

- WebSphere Commerce サイト管理者
- システム管理者
- WebSphere Commerce 開発者

場合によっては上記の担当は入れ替わる可能性があることに注意してください。

サイト管理者を対象とした機能強化

概してシステム管理者を対象とする WebSphere Commerce 5.4 のセキュリティーの機能強化は次のとおりです。

アクセス・コントロール

- **アクセス・コントロール・フレームワーク** — 主要な機能強化は、新規のアクセス・コントロール・フレームワークが WebSphere Commerce 5.4 でインプリメントされた点にあります。この新規のフレームワークは、アクセス・コントロール・ポリシーを使用して、特定のユーザーが特定のリソースで特定のアクションを実行することを許可されているかどうかを判別します。この新規のアクセス・コントロールのフレームワークは、きめ細かいアクセス・コントロールの手段になります。これは、WebSphere Application Server に備わったアクセス・コントロールと共同で稼働しますが、それに代わるものではありません。この新規のアクセス・コントロール・フレームワークについては、 99 ページの『第 11 章 アクセス・コントロール』に詳しく説明されています。

  この新規のアクセス・コントロール・フレームワークは、これまでのアクセス・コントロールを次のように強化しています。

  多様性の実現...

  多種多様なアクセス・ポリシーの目標が取り込まれています。このフレームワークは汎用であるため、広範囲にわたるユーザー・グループ、リソース・グループ、アクション・グループ、および関係グループを扱うことができます。

  階層化...

  ある組織が所有するアクセス・コントロール・ポリシーを、その下位組織にも適用することができます。

  カスタマイズ可能...

  アクセス・コントロール・ポリシーは、アプリケーション・コードの外部に置くことができるので、ポリシーに変更を加えてもコードを再コンパイルしなくて済みます。

  コンパクト...

  新規のフレームワークは簡単に縮尺できます。アクセス・コントロール・ポリシーの数は、オブジェクトの数の増加によってではなく、ビジネス・プロセス数の増加によって増加します。グループ設定用のフレームワークの多くは暗黙条件をベースにするので、条件が満足されている限り同じポリシーが適用されるからです。

- **サイト間スクリプト記述** — WebSphere Commerce 構成マネージャーの「サイト間スクリプト保護」ノードを使って、不許可と指定された属性や文字を使用しているユーザー要求を拒否します。これについては、 43 ページの『第 4 章 サイト・セキュリティーの機能強化』に詳述されています。

認証

- **「Password storage (パスワード・ストレージ)」** — WebSphere Commerce 5.4 は、パスワードそのものを保管するのではなく、WebSphere Commerce データベース内の SHA-1 ハッシュ体系を使ってパ

スワードの一方向ハッシュを暗号化して保管します。それによってユーザー・パスワードは、サイト管理者やシステム管理者も含め誰にも解読できないようになります。

- **「パスワード無効化」** — ユーザーが初めてシステムにログインしたときに、 WebSphere Commerce 構成マネージャーの「パスワード無効化」ノードを使って各自のパスワードを変更することを義務付けます。これについては、 43 ページの『第 4 章 サイト・セキュリティーの機能強化』に詳述されています。
- **「アカウント・ポリシー」** — WebSphere Commerce 管理コンソールのアカウント・ポリシー・ページを使って、使用中のアカウント関連のポリシーを定義するためのサイト用のアカウント・ポリシーをセットアップします。これについては、 43 ページの『第 4 章 サイト・セキュリティーの機能強化』に詳述されています。
- **「パスワード・ポリシー」** — WebSphere Commerce 管理コンソールのパスワード・ポリシー・ページを使って、ユーザーのパスワード選択特性を制御するためのサイト用のパスワード・ポリシーをセットアップします。これについては、 43 ページの『第 4 章 サイト・セキュリティーの機能強化』に詳述されています。
- **「アカウント・ロックアウト・ポリシー」** — WebSphere Commerce 管理コンソールの「アカウント・ロックアウト・ポリシー」ページを使って、ユーザー・アカウントに不祥事が起きる可能性を減少するためにサイト用のアカウント・ロックアウト・ポリシーをセットアップします。これについては、 43 ページの『第 4 章 サイト・セキュリティーの機能強化』に詳述されています。

許可

「Password protected commands (パスワード保護されたコマンド)」 — WebSphere Commerce 構成マネージャーの「Password protected commands (パスワード保護されたコマンド)」ノードを使って、指定コマンドを実行する要求を実行する場合はパスワードを入力することをユーザーに義務付けます。これについては、 43 ページの『第 4 章 サイト・セキュリティーの機能強化』に詳述されています。

データの暗号化

「データベース更新ツール」 — WebSphere Commerce 構成マネージャーのデータベース更新ツール・ノードを使って、パスワードやクレジット・カードの情報などの暗号化データならびに WebSphere® Commerce データベース内のマーチャント・キーを更新します。これについては、 43 ページの『第 4 章 サイト・セキュリティーの機能強化』に詳述されています。

セッション管理

「ログイン・タイムアウト」 — 「ログイン・タイムアウト」ノードを使って、一定期間を超えて非アクティブになっているユーザーをログオフさせ、元のシステムにログオンするよう要求します。この強化機能は、WebSphere Commerce 構成マネージャーを使って起動しますが、それについては、 43 ページの『第 4 章 サイト・セキュリティーの機能強化』に詳述されています。

ロギング

「Access logging (アクセス・ロギング)」 — アクセス・ロギングの使用可能化によって、 WebSphere Commerce に対するセキュリティー上の脅威

をすべて速やかに特定します。この強化機能は、WebSphere Commerce 構成マネージャーを使って起動しますが、それについては、 43 ページの『第 4 章 サイト・セキュリティーの機能強化』に詳述されています。

システム管理者を対象とした機能強化

概してシステム管理者を対象とする WebSphere Commerce 5.4 のセキュリティーの機能強化は次のとおりです。

- セキュリティーの重要な機能強化の 1 つとして、非標準のポート番号 (たとえば、ポート 443 に対してポート 8000) を実行するように WebSphere Commerce 管理ツールを構成することができます。このポートへのアクセスを制限すれば、ローカル・ネットワークまたはイントラネットだけが管理ツールにアクセスできるよう制限を設けることができます。
- WebSphere Commerce 管理コンソールから「Launch security check (セキュリティー検査の立ち上げ)」ページを使って、機密漏れの可能性があると思われる一時 WebSphere Commerce ファイルの検査と削除を行うためのセキュリティー・プログラムを立ち上げます。

WebSphere Commerce プログラマーを対象とした機能強化

主要な機能強化は、WebSphere Commerce 5.4 において新規のアクセス・コントロール・フレームワークがインプリメントされた点にあります。この新規のフレームワークは、アクセス・コントロール・ポリシーを使用して、特定のユーザーが特定のリソースで特定のアクションを実行することを許可されているかどうかを判別します。この新規のアクセス・コントロールのフレームワークは、きめ細かいアクセス・コントロールの手段になります。これは、WebSphere Application Server に備わったアクセス・コントロールと共同で稼働しますが、それに代わるものではありません。この新規のアクセス・コントロール・フレームワークについては、 99 ページの『第 11 章 アクセス・コントロール』に詳しく説明されています。

この新規のアクセス・コントロール・フレームワークは、これまでのアクセス・コントロールを次のように強化しています。

多様性の実現...
多種多様なアクセス・ポリシーの目標が取り込まれています。このフレームワークは汎用であるため、広範囲にわたるユーザー・グループ、リソース・グループ、アクション・グループ、および関係グループを扱うことができます。

階層化...
ある組織が所有するアクセス・コントロール・ポリシーを、その下位組織にも適用することができます。

カスタマイズ可能...
アクセス・コントロール・ポリシーは、アプリケーション・コードの外部に置くことができるので、ポリシーに変更を加えてもコードを再コンパイルしなくて済みます。

コンパクト...
新規のフレームワークは簡単に縮尺できます。アクセス・コントロール・ポリシーの数は、オブジェクトの数の増加によってではなく、ビジネス・プロ

セス数の増加によって増加します。グループ設定用のフレームワークの多くは暗黙条件をベースにするので、条件が満足されている限り同じポリシーが適用されるからです。

WebSphere Commerce Suite 5.1 Pro Edition におけるセキュリティー上の向上

Commerce Suite 5.1 は新規の e-commerce アーキテクチャーを具現化したものであり、 C++ ベースの Commerce Suite 4.1 の全面的書き直しであった一方で、それ以前のバージョンの WebSphere Commerce Suite のすべてのセキュリティー・フィーチャーに加え、セキュリティー上の新規の改善点も盛り込まれていました。そのような改善点は、WebSphere Commerce 5.4 でも引き継がれています。

Commerce Suite 5.1 では引き続き以下のようにして、旧リリースで備えられた WebSphere Commerce Suite 管理者およびショッパー・リソースへの無許可アクセスに対する保護が行われていました。

- 認証を受けた WebSphere Commerce Suite ユーザーであるか、または SSL モードになっていることを確認してから機密情報へのアクセスやその送信を行えるようにするためのアクセス・コントロール・フィーチャーのサポートの継続。
- Commerce Suite 4.1 と同じモデルに準じて、サイト管理者またはストア・レベルの管理者のみが特定のコマンドを実行できるようにするための、グループに対する WebSphere Commerce Suite コマンドの割り当て。

セキュリティーの一般的な機能強化

Commerce Suite 5.1 を Java™ で書き直したことによって、 C++ で書かれたソフトウェアでは免れえないセキュリティー上の問題が取り除かれました。 Java はポインターを使用しないので、 C++ ベースのほとんどのソフトウェアのセキュリティー上の短所であるバッファーのオーバーフローがなくなりました。業界標準の J2EE 仕様に準拠することで WebSphere Commerce Suite は、厳格な検査を行って、不正行為によって指定された妨害ステートメントをサーバーが実行することのないようにしました。

業界標準の Triple-DES (データ暗号化規格) アルゴリズムを使って WebSphere Commerce Suite システムの機密情報が保護されました。 Triple-DES アルゴリズムを収めたパッケージは、改ざんされた場合は WebSphere Commerce Suite サーバーが始動しないようにディジタル署名されています。

セッション管理

cookie を盗まれないようにするための独自の技法を使って WebSphere Commerce Suite セッション管理は全面的に書き換えられ、最大限のセキュリティーが実現されました。このように書き換えられたセッション管理は、SSL (secure sockets layer) のみを経由し、しかも暗号化タイム・スタンプで構成された認証 cookie を使用することによって、セッションのハイジャック対策を講じています。

認証

実行時に WebSphere Commerce Suite サーバーで必要なシステムおよびアプリケーションのパスワードは、マーチャント指定の 12 ビット鍵を使って確実に暗号化さ

れ、WebSphere Commerce Suite 構成ファイルに保管されます。ユーザーの URL エントリー・ボックスに表示される機密情報は、無許可の開示からショッパーを保護するために暗号化されます。

ロギング

WebSphere Commerce Suite ログ・システムは、セキュリティーを最重要課題として設計されているので、ショッパーのパスワードやクレジット・カード情報などの機密情報は、デフォルトでは WebSphere Commerce Suite ログ・ファイルに記録されませんでした。

本書の表記規則

本書では、以下のような強調表示の規則を使用しています。

- **太文字**は、コマンドまたは、フィールド名、アイコン、メニュー選択などのグラフィカル・ユーザー・インターフェース (GUI) コントロールを示します。
- モノスペース (Monospace) は、示されているとおりに入力するテキスト例、ファイル名、ディレクトリー・パスおよび名前を示します。
- *イタリック* は、語を強調するために使用します。イタリックは、ご使用のシステムの該当する値に置換しなければならない名前も示します。以下の名前が出現したら、説明に従ってご使用のシステムの値に置き換えてください。

  host_name
  WebSphere Commerce Studio マシンの完全修飾ホスト名 (たとえば、ibm.com という完全修飾名)。

  Windows

  drive この製品またはコンポーネントがインストールされているドライブを表す文字。 (たとえば、C:)

このアイコンは、ヒント (作業を完了するために役立つ追加情報) を表すマークです。

Windows は、WebSphere Commerce for Windows NT® および Windows® 2000 に固有の情報を示します。

AIX は、WebSphere Commerce for AIX® に固有の情報を示します。

Solaris は、WebSphere Commerce for Solaris™ オペレーティング環境ソフトウェアに固有の情報を示します。

400 は、WebSphere Commerce for the IBM @server iSeries™ 400® (以前の AS/400®) に固有の情報を示します。

Linux は、WebSphere Commerce for Linux に固有の情報を示します。

Professional WebSphere Commerce Professional Edition に固有の情報を示します。

Business WebSphere Commerce Business Edition に固有の情報を示します。

詳細情報の参照先

WebSphere Commerce 5.4 製品の詳細は、以下の Web サイトを参照してください。

- Business http://ibm.com/software/webservers/commerce/wc_be/lit-tech-general.html
- Professional http://www.ibm.com/software/webservers/commerce/wcs_pro/lit-tech-general.html

Commerce Studio, Professional Developer Edition 5.1 または旧リリースの WebSphere Commerce Studio の関連情報は、以下の Web サイトを参照してください。

http://www.ibm.com/software/webservers/commerce/commercestudio/lit-tech-general.html

第 1 部 WebSphere Commerce セキュリティー・モデル

第 1 部では、WebSphere Commerce のセキュリティーの概念について概略しています。

第 1 章 WebSphere Commerce セキュリティー・モデルの概要

この章は、WebSphere Commerce セキュリティー・モデルならびに WebSphere Commerce のさまざまなセキュリティー概念を説明しています。

概要

本書では、次のような認証、許可、ポリシー、および機密性の概念が説明されています。

認証とは

認証とは、ユーザーまたはアプリケーションが自称どおりのものかどうかを確認するためのプロセスのことです。 WebSphere Commerce システムでは、ゲスト・ユーザーを除き、システムにアクセスするすべてのユーザーとアプリケーションに認証が必要です。ユーザー認証プロセスは常に SSL のもとで実行されます。そのため、第三者はネットワークの不正使用プログラムを使っても、ユーザーからのパスワードの送信時にネットワークでスヌープ できなくなります。通常のセキュリティー措置の場合と同様、認証プロセス中にパスワードが暗号化解除されることはありません。すべてのユーザー・パスワードは、マーチャント・キー と呼ばれる 128 ビット鍵を使ってハッシュされます。マーチャント・キーは、WebSphere Commerce システムのインストールおよび構成時に指定します。

WebSphere Commerce システムには管理用の独自のパスワードがあります。そのパスワードは、WebSphere Commerce サイト全体のセキュリティー・ポリシーの一環として定期的に変更する必要があります。 WebSphere Commerce 5.4 システムのパスワードの変更方法の詳細は、 79 ページの『第 7 章 パスワードの設定と変更』を参照してください。

許可とは

許可とは、ユーザーがリソースに対して特定の操作を実行できるかどうかを決定するプロセスのことです。許可は、WebSphere Commerce リソースに対するアクセス・コントロール・ポリシーから決定されます。 WebSphere Commerce システムでは以下の 2 つの領域でアクセス・コントロールが必要です。

- 無許可アクセスが起きないように WebSphere Commerce Enterprise JavaBeans™ (EJB beans) を保護するため。このプロセスについては、 61 ページの『第 5 章 WebSphere Application Server のセキュリティーの使用可能化』に説明されています。
- 許可を受けた関係者のみが、さまざまな WebSphere Commerce コマンド・グループを実行できるようにするため。このプロセスについては、 99 ページの『第 11 章 アクセス・コントロール』に説明されています。

アクセス・コントロール・ポリシーとは

e-commerce サイトに参加する組織とユーザーの定義が完了したと仮定すると、その後一連のポリシーを通してそれらの人々のアクティビティーを管理することができます。このプロセスをアクセス・コントロール と呼びます。

アクセス・コントロール・ポリシーとは、サイトにおいてどのユーザーまたはユーザー・グループがどのアクティビティーを実行する許可を受けるかを定めた規則のことです。そのようなアクティビティーの範囲は、登録から始まって、オークションの管理、商品カタログの更新、オーダーの承認権の認可、さらには e-commerce サイトの運用と維持に必要なその他の数々のアクティビティーにまでいたります。

ポリシーとは、ユーザーがサイトにアクセスすることを認可する手段です。担当作業を実行する許可を 1 つ以上のアクセス・コントロール・ポリシーを通して受けない限り、ユーザーはサイトのどの機能にもアクセスすることはできません。

WebSphere Commerce 5.4 のアクセス・コントロール・モデルは、アクセス・コントロール・ポリシーの規定内容に基づきます。アクセス・コントロール・ポリシーは、アクセス・コントロール・ポリシー・マネージャーによって規定されます。一般的に、保護される可能性のあるリソースにユーザーがアクセスしようとすると、アクセス・コントロール・ポリシー・マネージャーはまず、そのユーザーに対してどのアクセス・コントロール・ポリシーを適用できるかを判別してから、適用可能なそのアクセス・コントロール・ポリシーに基づいて、そのユーザーが要求した操作を特定のリソースで実行してもよいかどうかを決定します。

監査記録とは

コンピューターでは監査記録 とは、コンピューターのアクティビティーを追跡記録するのに使われる電子または書面ログを言う語です。たとえば、企業の社員は売掛管理などの一部の社内ネットワークにアクセスできても、給与計算などの他のシステム部分へのアクセスは許可されないことがあります。その社員がパスワードを入力して無許可セクションへのアクセスを試みた場合、その不適切なアクティビティーは監査記録に記録されます。

e-commerce システムでは監査記録は、顧客アクティビティーを記録するのに使われます。システムに対する顧客の最初のコンタクトや、商品またはサービスの決済や納品などのその後のアクションが監査記録に記録されます。企業はこの監査記録を使って、すべての照会または苦情に対処することができます。また、監査記録を使って、アカウントの調整、今後の計画と予算設定に関する分析と履歴情報の提供、および税務監査の場合の販売記録の提供を行うことも可能です。

さらに監査記録を使えば、サイバースペースやインターネットを介したコンピューター犯罪を調査することもできます。システムに対して不純な意図をもって不正行為を働いた人物が残した監査記録をたどって調査すれば、犯人を特定することができます。コンピューター犯罪の実行者は、インターネット・サービス・プロバイダーでのアクティビティーのログや、チャット・ルームのログを介して、うかつにも監査記録を残していることがあるからです。

機密性とは

機密性とは、指定外の宛先が機密情報を暗号解読しないように保護するためのプロセスのことです。 WebSphere Commerce システムで機密性が必要になるのは、機密情報がユーザーのブラウザーから WebSphere Commerce サーバーに送られるときと、 WebSphere Commerce サーバーから元のユーザーのブラウザーへ返送されるときです。 87 ページの『第 8 章 IBM HTTP Server での実動のための SSL の使用可能化』に説明されているとおり、 SSL (Secure Sockets Layer) が使われて、このシナリオの機密性が実現されます。

機密性は、セッション管理の分野においても重大な要件です。 HTTP (Hypertext Transfer Protocol) はステートレスであるため、 WebSphere Commerce サーバーに対して継続的にユーザーを識別するために通常は *cookie* が使用されます。この cookie が傍受されると、ユーザー・アカウントに不祥事が起きる可能性があります。通常はこれをセッション・ハイジャック と呼んでいます。 WebSphere Commerce では、 71 ページの『第 6 章 セッション管理』に説明されているとおり、 cookie を指定するための独自のフィーチャーを介してセッション・ハイジャックが防止されています。

第 2 章 認証

WebSphere Commerce では認証は、ユーザーまたはアプリケーションが当人であるかどうかを検査するプロセスと見なされます。この項では、WebSphere Commerce の認証のいくつかの側面を詳しく説明します。

WebSphere Commerce 認証モデル

WebSphere Commerce の認証モデルは、次のような概念に基づいています。

- チャレンジ機構
- 認証機構
- ユーザー・レジストリー

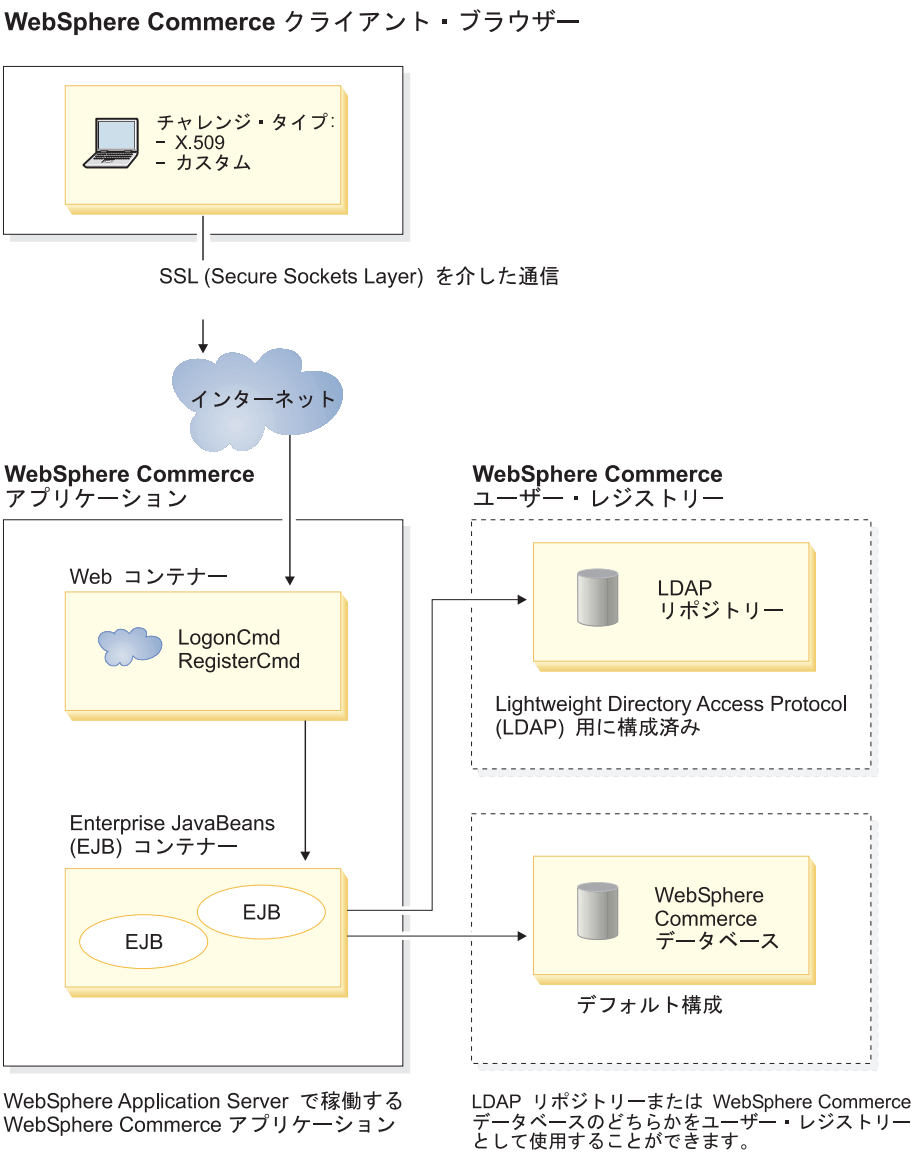


図 1. WebSphere Commerce 5.4 セキュリティー・モデル

チャレンジ機構

チャレンジ機構は、サーバーがユーザーの認証データにどのように取り組んで取り出すかを指定します。 WebSphere Commerce 5.4 は、以下の認証方式またはチャレンジ機構をサポートしています。

フォーム・ベース認証またはカスタム認証

この認証機構では、HTML ページまたは JSP フォームを介したサイトまたはストア独自のログインが許可されます。

証明書ベースの認証 (X.509 証明書)

証明書チャレンジ機構は、 SSL を通して相互認証を実行するよう Web サーバーが構成されているという意味を含みます。接続を確立しようとするクライアントは、証明書の提示を要求されます。この証明書は次に、ユーザー・レジストリーにマップされる証明書になります。

認証機構

認証機構 は、ユーザーに関連付けられたユーザー・レジストリーに照らして認証データを検証してユーザーを認証します。認証プロセスが完了した後は WebSphere Commerce 5.4 は、要求があるたびにユーザーに結び付く証明書トークンを発行します。そのトークンは、ユーザーがブラウザーをログオフまたはクローズすると終了します。

証明書の妥当性検査

これは、X.509 クライアント証明書が Web サーバーによって信頼されていて、しかもその Web サーバーの証明書ポリシーに準拠していることを検証するプロセスです。また WebSphere Commerce は、WebSphere Commerce データベースにも照らし合わせて X.509 証明書を検証します。 Web サーバーは証明書を概括的にアクセス・コントロールするのに対して、 WebSphere Commerce は証明書をち密にアクセス・コントロールします。

LDAP バインド

これは、LDAP バインド操作の実行によってユーザーを認証することで、入力されたチャレンジ情報が正しいことを検証するプロセスです。

データベースのバインド

これは、認証プロセス中に入力されたユーザー ID とパスワードが、 WebSphere Commerce データベースに保管されている認証情報と比較して正しいことを検証するプロセスです。

ユーザー・レジストリー

ユーザー・レジストリーとは、ユーザー情報と、ユーザーの認証情報 (パスワードなど) の入ったリポジトリーのことです。プリンシパル (つまり、実際のユーザーまたはユーザー・レジストリー内のシステム・エンティティーを表します) によって入力された認証情報は、このユーザー・レジストリーに突き合わせて検証または検査することができます。

WebSphere Commerce 5.4 は、 LDAP ユーザー・レジストリーと WebSphere Commerce データベースの 2 つのユーザー・ドメインを基盤としてユーザー・レジストリーをサポートします。

WebSphere Commerce 5.4 は以下の LDAP プロバイダーをサポートします。

- IBM SecureWay® Directory Windows AIX Solaris Linux 400
- Netscape® Directory Server Windows AIX Solaris
- Windows 2000 Active Directory 2000

認証情報

WebSphere Commerce 5.4 サーバーは、証明書、トークン、またはユーザー ID とパスワードなどの認証情報の検証に基づいた認証機構をサポートします。認証情報は、そのような体系をサポートするユーザー・レジストリーに照らし合わせて検証されます。

WebSphere Commerce トークン

WebSphere Commerce は、セキュア認証 cookie を使って認証データを管理します。認証 cookie は SSL を通してのみやりとりされ、しかもセキュリティーの最大化のためにタイム・スタンプが押されます。たとえばユーザーのクレジット・カード番号をたずねる `DoPaymentCmd` といった機密性の高いコマンドが実行されるたびに、ユーザーを認証するのにこの cookie が使用されます。この cookie が盗まれて無許可のユーザーによって使用される危険性は最小化されています。

SSL または非 SSL のどちらの接続の場合にもブラウザーとサーバーとがやりとりするまた別の cookie がありますが、これは、非 SSL 接続を介するユーザーを検証するのに使われます。

WebSphere Application Server LTPA トークン

LTPA トークンとは、ユーザーから要求されたリソースに対するアクセス許可を確認するのに必要なユーザー情報の入ったデータのことです。これには、認証データならびに WebSphere Application Server LTPA サーバーのディジタル・シグニチャーが入っています。

WebSphere Application Server の LTPA (Lightweight Third Party Authentication) の場合、ユーザーに関する情報の入った LDAP が、認証の実行対象のユーザー・レジストリーになります。リソース・サーバーは、WebSphere Application Server セキュリティー・サーバーに連絡をとって、認証機構として LTPA を指定します。また、その要求に関連した認証データも提供します。次に WebSphere Application Server セキュリティー・サーバーは LTPA サーバーに対して認証データを検証し、 LTPA トークンを戻します。

単一サインオン

複数の HTTP 要求で一貫してユーザー認証を保持するというのが、HTTP 単一サインオンの背後にある考え方です。その目標は、以下を含め、特定の信頼されたドメイン内でセキュリティー認証情報を何回もユーザーにたずねなくて済むようにすることにあります。

- 共同で稼働する異種の WebSphere Application Server Web サーバー同士
- LDAP サーバー (IBM SecureWay Directory Server など) のような、共同作業を担うアプリケーション

単一サインオン (SSO) のシナリオでは、種々の Web サーバーにユーザーの認証情報を伝搬して、クライアント・サーバー・セッションが新しくなっても、そのつどユーザーが認証情報を入力しなくて済む (基本認証を前提として) ようにするために HTTP cookie が使われます。

WebSphere Commerce での単一サインオンのインプリメントの詳細は、 95 ページの『第 10 章 単一サインオン』を参照してください。

認証ポリシー

認証ポリシーとは一連の規則のことですが、それらの規則は、認証プロセスに対してと、WebSphere Commerce での認証データの検証に対して適用されます。この後の項に説明されているとおり、WebSphere Commerce 5.4 は、アカウント・ポリシー、他の認証関連のポリシー、およびセッション・ポリシーをサポートします。

アカウント・ポリシー

以下の項では、WebSphere Commerce で利用できるアカウント・ポリシーについて説明します。

アカウント・ポリシー

WebSphere Commerce 管理コンソールの「アカウント・ポリシー」ページで、アカウント・ポリシーをセットアップすることができます。アカウント・ポリシーは、パスワード・ポリシーやアカウント・ロックアウト・ポリシーなどのアカウントに関連するポリシーを定義します。

アカウント・ポリシーの作成が完了したら、ユーザーに割り当てることができます。アカウント・ポリシーが使用中の (つまり、ユーザーがアカウント・ポリシーを割り当てられている) 場合は、そのポリシーを削除することはできません。

アカウント・ポリシーの詳細は、 54 ページの『アカウント・ポリシーのセットアップ』を参照してください。

WebSphere Commerce オンライン・ヘルプの「Default Authentication Policies」も参照してください。

アカウント・ロックアウト・ポリシー

WebSphere Commerce 管理コンソールの「アカウント・ロックアウト・ポリシー」ページで、WebSphere Commerce 内のさまざまなユーザー役割用のアカウント・ロックアウト・ポリシーをセットアップすることができます。アカウント・ロックアウト・ポリシーは、ユーザー・アカウントに対して不正アクションがとられた場合にそのアカウントを使用禁止にすることで、そのようなアクションによってアカウントが被害を受ける機会を減らします。

アカウント・ロックアウト・ポリシーは次のようなアイテムを統制します。

- アカウント・ロックアウトのしきい値。無効なログオンの試行回数がこの値に達すると、アカウントが使用不可になります。
- ログインの連続失敗による遅延。これは、ユーザーがログインに 2 回失敗した場合にその後ログインできなくなる期間を指します。ログインの失敗が続くと、この遅延はそのつど構成済みの時間遅延値 (たとえば 10 秒) ずつ増加されます。

アカウント・ロックアウト・ポリシーの作成の詳細は、 56 ページの『アカウント・ロックアウト・ポリシーのセットアップ』を参照してください。

パスワード・ポリシー

WebSphere Commerce 管理コンソールの「パスワード・ポリシー」ページでは、ユーザーのパスワード選択を制御して、サイトのセキュリティー・ポリシーが順守されるようにユーザーのパスワードの特性を定義することができます。

このフィーチャーは、パスワードが守らなければならない属性を定義します。パスワード・ポリシーで、以下の条件を決定します。

- ユーザー ID とパスワードが同じでよいか
- 連続する最大文字数
- 文字の最大インスタンス
- パスワードの最長存続時間
- 英字の最小文字数
- 数字の最小文字数
- パスワードの最低限の長さ
- ユーザーの以前のパスワードを再利用できるか

パスワード・ポリシーの詳細は、 55 ページの『パスワード・ポリシーのセットアップ』を参照してください。

WebSphere Commerce オンライン・ヘルプの「Default Authentication Policies」も参照してください。

その他のポリシー関連のポリシー

以下の項では、WebSphere Commerce で利用できるその他の認証関連のポリシーについて説明します。

パスワード無効化

パスワード無効化フィーチャーを使用可能または使用不可にするには、構成マネージャーの「パスワード無効化」ノードを使用します。このフィーチャーを使用可能にした場合に WebSphere Commerce ユーザーのパスワードの有効期限が切れると、そのユーザーはパスワードの変更を要求されます。その場合、ユーザーは、パスワードの変更を要求されるページにリダイレクトされます。ユーザーは、パスワードの変更を完了するまで、そのサイトのどのセキュア・ページにもアクセスすることができません。

「パスワード無効化」ノードの使用の詳細は、 48 ページの『パスワード無効化の活動化』を参照してください。

パスワード保護されたコマンド

「Password protected commands (パスワード保護されたコマンド)」フィーチャーを使用可能または使用不可にするには、「構成マネージャー」の「Password protected commands (パスワード保護されたコマンド)」ノードを使用します。このフィーチャーを使用可能にすると、WebSphere Commerce は、 WebSphere Commerce にログオンした登録済みユーザーに、まずパスワードを入力してから、指定した WebSphere Commerce コマンドの実行要求を続行するよう求めます。

注意: パスワード保護コマンドを構成する場合、コマンド選択リストに示されているコマンドの一部は、一般ユーザーまたはゲスト・ユーザーが実行できるコマンドであることに注意してください。そのようなコマンドを、保護されたパスワードとして構成すると、一般ユーザーおよびゲスト・ユーザーはそのコマンドを実行できなくなります。したがって、コマンドを構成して保護されたパスワードにする場合は注意を払う必要があります。

注: WebSphere Commerce では、使用可能コマンドのリストの URLREG テーブルで **認証済み** と指定されているコマンドか、または https フラグを使って設定されたコマンドのみが表示されます。

「Password protected commands (パスワード保護されたコマンド)」ノードの使用の詳細は、48 ページの『パスワード保護コマンドの使用可能化』を参照してください。

セッション・ポリシー

WebSphere Commerce 5.4 ではセッション・ポリシーは、ログイン・タイムアウト・ポリシーとして具体化されています。

ログイン・タイムアウト・ポリシーの使用時には WebSphere Commerce は、期間を超えて非アクティブになっているユーザーをログオフさせ、「ログイン・タイムアウト」ノードを使って元のシステムにログオンするよう要求します。この強化機能は、WebSphere Commerce 構成マネージャーを使って起動しますが、それについては、47 ページの『ログイン・タイムアウトの使用可能化』に説明されています。

第 3 章 許可 (アクセス・コントロール)

WebSphere Commerce では許可は、ユーザーまたはアプリケーションがリソースにアクセスするのに十分な権限をもっているかどうかを検査するプロセスと見なされます。この項では、WebSphere Commerce のアクセス・コントロールのいくつかの側面を詳しく説明します。

許可つまりアクセス・コントロールは、 WebSphere Commerce ではアクセス・コントロール・ポリシーを使って実行されます。アクセス・コントロール・ポリシーとは、どのユーザーまたはユーザー・グループが、一連のリソースで一連のアクティビティーを実行できるかを定めた規則のことです。 WebSphere Commerce には、一連のデフォルトのアクセス・コントロール・ポリシーが用意されています。このデフォルト・アクセス・コントロール・ポリシーは、XML 形式で指定されており、e-commerce サイトが必要とする典型的なアクセス・コントロール要件の大半に対処できるように設計されています。 WebSphere Commerce のアクセス・コントロール・コンポーネントを理解するには、 e-commerce サイトの通常の組織階層をまず理解する必要があります。

組織階層

WebSphere Commerce メンバー・サブシステム内のユーザーと組織エンティティーは、階層に編成されます。この階層は、典型的な組織階層をエミュレートしたものですが、そこでは組織と組織単位用のエントリーと、ユーザー用のエントリーがリーフ・ノードに置かれます。この階層では、最上部に*ルート組織* と呼ばれる人工的なエンティティーが置かれます。他の組織エンティティーとユーザーはすべて、そのルート組織の下位に配置されます。ルート組織の下には、1 つのセラー組織と複数のバイヤー組織を置くことができ、それらの組織にはいずれも、下位の副組織を設けることができます。バイヤーまたはセラーの管理者は、それらの組織の最上位に位置し、組織を保守する責任を負います。セラー組織サイドでは、各副組織内に 1 つ以上のストアを設けることができます。そのストアの管理者は、ストアを保守する責任を負います。以下の図は、企業間取り引きの e-commerce サイトの組織階層を示しています。

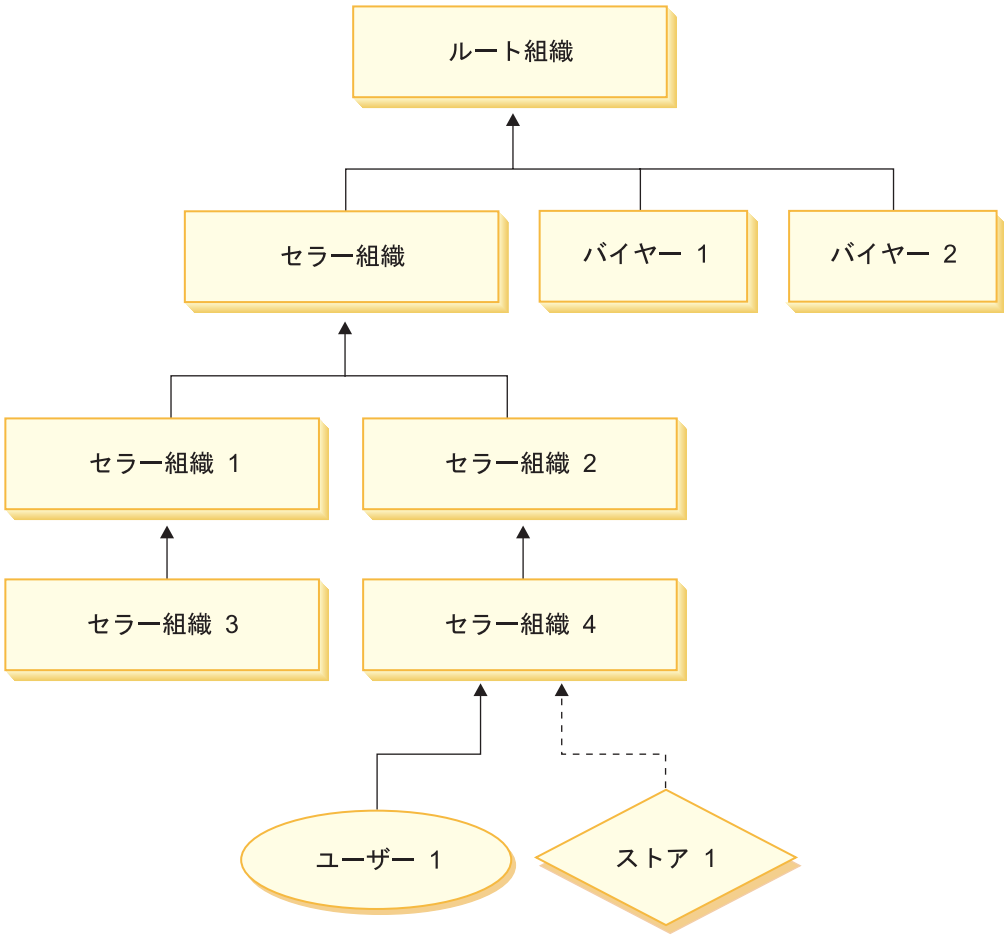


図 2. 企業間取り引きサイトの組織階層

ルート組織

ルート組織は、組織階層の最上位に位置します。サイト管理者は、WebSphere Commerce 内のすべての操作実行に対するスーパーユーザー・アクセス権を持ちます。サイト管理者は、WebSphere Commerce とそれに関連したソフトウェアおよびハードウェアのインストール、構成、および保守を行います。通常この役割は、アクセスおよび認証の制御 (メンバーを作成して適切な役割に割り当てる) と、 Web サイトの管理を行います。サイト管理者は、ユーザーに対する役割の割り当てと、ユーザーがその役割を果たす場所である組織の指定を行うことができます。許可を受けた関係者しか機密情報にアクセスできないようにするために、サイト管理者は、各管理者にパスワードを割り当てなければなりません。それによって、カタログの更新や見積要求 (RFQ) の承認などの重要な責務を制御することができます。

注: ユーザーが、親組織以外の組織において役割を果たす可能性もあります。

WebSphere Commerce サイトには、1 つのセラー組織しかありません。企業間取り引きサイトには、1 つ以上のバイヤー組織があります。したがってサイト管理者は、ストアを所有するセラー組織のアクセス・コントロール・ポリシーと、そのストアの購入客である各組織のアクセス・コントロール・ポリシーを両方とも定義す

ることができます。業者対消費者サイトには、バイヤー組織はありません。業者対消費者の顧客は、デフォルト組織のメンバーとしてモデル化されています。

組織 (セラー)

企業間取り引きサイトと業者対消費者サイトのどちらでも、サイト管理者は最上位のセラーを 1 つ作成します。そのセラー組織の下に、他の副組織または組織単位を作成することができます。そのような販売サイドの組織エンティティーは、1 つ以上のストアを所有することができます。その後サイト管理者は、セラー組織用の特別なアクセス・コントロール・ポリシーをすべて定義してから、そのセラー組織を管理するセラー管理者を割り当てます。セラー管理者はユーザーを登録してから、組織に対して定められているアクセス・コントロール・ポリシーに沿って、組織の業務上の要件に合うようにそれぞれに異なる役割を割り当てます。

セラー管理者の責務は次のように要約されます。

- ストアを所有できる副組織を作成します。 (オプション) 組織の中のどの処理に承認が必要であるかを定義します。このステップは、企業間取り引きのサイトにのみ必要です。
- 副組織に役割を割り当てます。
- ユーザーを作成します。
- ユーザーに役割を割り当てます。

組織 (バイヤー)

企業間取り引きサイトでは、取引上の要件に応じてサイト管理者は 1 つ以上のバイヤー組織を作成します。その後サイト管理者は、バイヤー組織用の特別なアクセス・コントロール・ポリシーをすべて定義してから、そのバイヤー組織を管理するバイヤー管理者を割り当てます。バイヤー管理者はユーザーを登録してから、組織に対して定められているアクセス・コントロール・ポリシーに沿って、組織の業務上の要件に合うようにそれぞれに異なる役割を割り当てます。

バイヤー管理者の責務は次のように要約されます。

- バイヤー組織内の副組織を作成および管理します。 (オプション) 組織の中のどの処理に承認が必要であるかを定義します。このステップは、企業間取り引きのサイトにのみ必要です。
- 副組織に役割を割り当てます。
- ユーザーを作成します。
- ユーザーに役割を割り当てます。

注: サイト管理者は、バイヤー組織のアクセス・コントロール・ポリシーを必要に応じて変更および管理することができます。サイト管理者が担当する作業の詳細は、18 ページの『サイト管理者』を参照してください。

役割

上記のとおり、WebSphere Commerce には一連のデフォルト役割が用意されています。サイト管理者は、それぞれの組織に個々の役割を割り当ててから、それらの役割にユーザーを割り当てる必要があります。組織は、親組織に割り当てられている役割のみを取り入れることができます。同様に、ユーザーもその親組織に割り当てられている役割のみを取り入れることができます。

WebSphere Commerce 内のすべての役割は、組織全体を有効範囲とします。たとえば、ユーザーが組織 X のプロダクト・マネージャーの役割を務めるとすると、このユーザーの親組織も、自身にプロダクト・マネージャーの役割が割り当てられている必要があります。またそのユーザーが、組織 X とその副組織のコンテキスト内でのみ商品管理業務を実行できるようにアクセス・コントロール・ポリシーをセットアップすることができます。

注: ユーザーおよび組織への役割の割り当ては、MBRROLE テーブルで行います。

WebSphere Commerce に用意されているデフォルト役割は、次のようなカテゴリーに分けることができます。

- サイト運用
- サイトおよびコンテンツの開発
- マーケティング管理
- 商品管理
- 販売管理
- ロジスティクスおよびオペレーション管理
- 組織管理

サイトの運用

以下の技術オペレーション役割が WebSphere Commerce でサポートされています。

- サイト管理者
- ストア管理者

サイト管理者

サイト管理者は、WebSphere Commerce とそれに関連したソフトウェアおよびハードウェアのインストール、構成、および保守を行います。管理者は、システムの警告、アラート、エラーに対して応答し、システムの問題を診断して解決します。この役割は、通常はアクセスおよび認証の制御 (メンバーを作成して適切な役割に割り当てる)、Web サイトの管理、パフォーマンスのモニター、およびロード・バランシング・タスクの管理を行います。サイト管理者には、さまざまな開発段階 (テスト、ステージング、実動など) のさまざまなサーバー構成を設定して保守する責任もあります。またこの役割は、重要なシステムのバックアップ処理や、パフォーマンス上の問題の解決も行います。

ストア管理者

ストア管理者は、ストア資産を管理し、税、配送およびストア情報の変更を更新して公開します。またストア管理者は、組織のアクセス・コントロール・ポリシーも管理することができます。ストア管理者 (通常はストア開発チームのリーダー) は、

ストア開発チームではストア・アーカイブを公開する権限のある唯一の役割です (サイト管理者も、ストア・アーカイブを公開できます)。通常、ストア管理者は、Web を十分に理解しており、ストアのビジネス手順に関する深い知識を持っています。

サイトおよびコンテンツの開発

WebSphere Commerce は、ストア開発者サイトとコンテンツ開発役割をサポートします。

ストア開発者

ストア開発者は、Java Server Pages ファイルと必要なすべてのカスタマイズ・コードを作成します。また、WebSphere Commerce に組み込まれている標準機能のすべてを修正することができます。ストア・アーカイブの作成が完了した後、ストア開発者はそれを手動で変更したり、またはストア・プロファイル・ノートブック、税ノートブック、および配送ノートブックを使って変更する権限を有します。ただし、ストア・アーカイブを WebSphere Commerce Server に対して発行する権限はありません。

ロジスティクスとオペレーション

WebSphere Commerce は、以下のロジスティクスとオペレーションをサポートしています。

- ロジスティクス・マネージャー
- オペレーション・マネージャー
- 受取人
- 返品担当者
- 梱包担当者

ロジスティクス・マネージャー

Business 配送管理者とも呼ばれるロジスティクス・マネージャーは、大量の貨物輸送や、運送会社から物流拠点または個々の顧客までの配送を管理および折衝します。この役割は、企業が最も望ましい配送者を最も望ましい料金で使って、企業ストラテジーを満足できるようにする責任を負います。配送は顧客サービスにおける重要な側面であるので、オンライン・ビジネスを成功させるための中心的要因となることがあります。

オペレーション・マネージャー

B2C この役割は、オーダーが正しく実行され、支払額が受け取られ、そしてオーダー内容が配送されるようにすることで、オーダー処理を管理します。オペレーション・マネージャーは、顧客オーダーの検索、詳細情報の表示、オーダー情報の管理、および返品の作成と編集を実行できます。

梱包担当者

梱包担当者は、実行センターから商品を受け取って、その商品を梱包して顧客に配送します。また梱包担当者は、オーダーの実行中に商品の配送を確認するのに使用されるピッキング・チケットとパッキング・スリップも管理します。

受取人

受取人は、実行センターでの在庫商品の受け取り、オーダー済み商品の予測在庫レコードおよび随時受け取りの追跡、顧客から返品された返品商品の受け取りを行います。

返品担当者

返品担当者は、返品商品の処置を管理します。

- 返品リスト
- 返品商品のリスト
- 返品商品の処置

商品の管理

以下の商品管理役割が WebSphere Commerce でサポートされています。

- バイヤー (セラー・サイド)
- カテゴリー・マネージャー
- プロダクト・マネージャーまたは取引管理マネージャー

バイヤー (セラー側)

バイヤーは、販売用の商品を仕入れます。バイヤーは、納品や支払いのオプションに関して有利な条件で思惑通りの商品を入手するための、取引先またはサプライヤーとの業務関係と折衝を担当します。バイヤーは、価格を設定することができます。仕入れる数量の決定と、在庫の適宜補充のために、バイヤーが在庫を管理します。

カテゴリー・マネージャー

カテゴリー・マネージャーは、カテゴリーを作成、変更、および削除することによってカテゴリー階層を管理します。カテゴリー階層は、ストアが提供する商品やサービスを編成します。カテゴリー・マネージャーは、商品、予測在庫レコード、取引先情報、および返品理由も管理します。

プロダクト・マネージャー / 取引管理マネージャー

Business 取引管理マネージャー、または B2C プロダクト・マネージャーは、顧客の購入をトレースし、割引案を提示し、オンライン・ストアにおける商品の最良の表示方法、価格設定方法、および販売方法を決定します。

- カテゴリー・マネージャーのすべてのタスクの実行
- マーケティング・マネージャーのすべてのタスクの実行

セールスの管理

以下の企業間関係管理役割が WebSphere Commerce でサポートされています。

- セールス・マネージャー
- アカウント担当者
- 顧客サービス・スーパーバイザー
- 顧客サービス担当者

セールス・マネージャー

セールス・マネージャーは、集客と顧客管理、販売予測の実現、顧客ビジネスを増大するための手掛かりの配備、契約の管理、価格設定条件の設定、在庫予測の確定のためのプロダクト・マネージャーとの協力、セールのためのマーケティング・マネージャーとの協力を行います。

アカウント担当者

アカウント担当者は、個々のアカウントを扱う業務を担って、業務関係の確立と顧客サービスに関する懸案事項の管理を行います。この担当者は、契約価格の変更、契約やプロファイルの折衝、およびアカウント・カテゴリー別の利益率の分析を行う許可を受けることができます。

顧客サービス・スーパーバイザー

この役割は、すべての顧客サービス・タスクにアクセスできます。顧客サービス・スーパーバイザーは、顧客照会 (顧客の登録、オーダー、返品、およびオークションなど) を管理し、システムが拒否した返品レコードの承認や、支払例外 (クレジット・カードの許可不能など) 関連での顧客への連絡といった、顧客サービス担当者からアクセスできないアクションを実行する権限を持ちます。

顧客サービス担当者

顧客がセルフサービス・フィーチャーを利用できるように巧みに設計されたオンライン・ストアの場合でも、ある種の顧客または場合によっては、Web に通じた顧客であっても直接連絡をとる必要が生じる場合があります。オンライン・ビジネスではたいてい、顧客が直接サービスを受けられるように E メール、ファクシミリ、または連絡電話番号が提示されます。顧客サービスの担当者が、顧客からのすべての問い合わせを処理する責任を負います。

マーケティングの管理

WebSphere Commerce は、マーケティング・マネージャーのマーケティング管理役割をサポートします。

マーケティング・マネージャー

マーケティング・マネージャーは、マーケット戦略およびブランド・メッセージを顧客に伝達します。この役割は、顧客の振る舞いをモニターしたり、分析したり、また把握したりします。さらに、マーケティング・マネージャーは目標とする販売のための顧客プロファイルを作成または変更します。また、キャンペーンおよび販売促進の作成と管理を行います。キャンペーン・イベントの計画を取り扱えるのは、マーチャント、マーケティング・マネージャー、および取引管理マネージャーで構成されるチームです。

組織の管理

WebSphere Commerce は、組織管理役割をサポートしています。

- セラー管理者
- バイヤー管理者
- バイヤー承認者

セラー管理者

セラー管理者は、販売組織に関する情報を管理します。セラー管理者は、該当するビジネス役割の割り当てを含め、販売組織内の副組織と、販売組織内の各種ユーザーを作成および管理します。

バイヤー管理者

バイヤー管理者は、購買組織に関する情報を管理します。この管理者は、購買組織内の副組織を作成および管理し、ユーザーをバイヤーと承認することを含め、さまざまなユーザーを管理します。バイヤー承認者や追加のバイヤー組織管理者などのその他の購買サイドの役割も、作成および管理することができます。

バイヤー承認者

バイヤー承認者とは、セラーからの購買のためのオーダーの送信の前に、バイヤーから出されたそのオーダーを承認する購買組織内の担当者のことです。

アクセス・コントロール・ポリシー

アクセス・コントロール・ポリシーは、 WebSphere Commerce 内の一連のリソースでの一連のアクティビティーの実行をユーザー・グループに許可します。 1 つ以上のアクセス・コントロール・ポリシーを通して許可を受けない限り、ユーザーはシステムのどの機能にもアクセスすることはできません。アクセス・コントロール・ポリシーを理解するには、ユーザー、アクション、リソース、および関係という 4 つの概念を理解する必要があります。ユーザーとは、システムの利用者のことです。リソースとは、保護の必要のあるシステム内のオブジェクトのことです。アクションとは、ユーザーがリソースで実行できるアクティビティーのことです。関係とは、ユーザーとリソースの間に存在するオプションの条件のことです。

アクセス・コントロール・ポリシーのエレメント

アクセス・コントロール・ポリシーは、4 つのエレメントで構成されています。

アクセス・グループ
制御ポリシーを適用されるユーザーのグループ。

アクション・グループ
リソースに対してユーザーによって実行されるアクションのグループ。

リソース・グループ
制御ポリシーが制御するリソース。リソース・グループには、**契約**や**オーダー**などのビジネス・オブジェクトや、ユーザーが実行できるオークション関連のコマンドなどの一連の関連コマンドを組み込むことができます。

関係 (オプション)
各リソース・クラスには、関係のセットを関連付けることができます。どのリソースも、各関係を成立させるための一連のユーザーを備えることができます。たとえば、オーダーの作成者のみがその変更を行えるとポリシーで指定することができます。その場合の関係は**作成者**になり、それはユーザーとオーダー・リソースの間の関係になります。

アクセス・コントロール・ポリシーの概念

アクセス・コントロール・ポリシーは、サイトへのユーザーのアクセスを認可します。担当作業を実行する許可を 1 つ以上のアクセス・コントロール・ポリシーを通して受けない限り、ユーザーはサイトのどの機能にもアクセスすることはできません。

アクセス・コントロール・ポリシーはそれぞれ以下の形式をとります。

```
AccessControlPolicy [AccessGroup,ActionGroup,ResourceGroup,Relationship]
```

アクセス・コントロール・ポリシー内のエレメントでは、特定のユーザー・グループに属しているユーザーは、リソースとの特定の関係を保つ限り、指定されたリソース・グループに属しているリソースで、指定されたアクション・グループのアクションを実行できることが指定されます。関係は、必要時のみ指定する必要があります。たとえば、[AllUsers, UpdateDoc, doc, creator] は、文書の作成者であれば、すべてのユーザーが文書を更新できることを指定します。

以下の項では、アクセス・コントロールに関連した概念に関する解説と用語について述べています。

メンバー・グループ

WebSphere Commerce ではメンバー・サブシステムは、さまざまな業務上の根拠に基づいて分類されたユーザー・グループであるメンバー・グループを作成することができます。たとえば、アクセス・コントロールや承認を目的としたり、割引および価格の計算や商品の表示などのマーケティングを目的とするなど、さまざまな目的でグループ分けを使用することができます。アクセス・グループ (-2) のタイプのメンバー・グループはアクセス・コントロールを目的とするのに対して、ユーザー・グループ (-1) のタイプのメンバー・グループは一般使用を目的とします。メンバー・グループは、MBRGRPUSG テーブルにおいてメンバー・グループ・タイプに関連付けられます。

アクセス・グループ: アクセス・グループ (-2) のタイプのメンバー・グループは、アクセス・コントロールの目的によってユーザーを分類したグループです。アクセス・グループは、アクセス・コントロール・ポリシーのエレメントの 1 つであり、特にアクセス・コントロールを目的として定義されたユーザーのグループと定義されます。アクセス・グループへのメンバーシップの基準では、通常は役割、ユーザーが所属する組織、またはユーザーの登録状況がベースになります。たとえば、**バイヤー管理者**というアクセス・グループを、バイヤー管理者の役割を果たすユーザーが属するグループとすることができます。

WebSphere Commerce には、いくつかのデフォルト役割が組み込まれていますが、それぞれの役割に対応して、その役割を暗黙で参照するデフォルト・アクセス・グループがあります。サイト内でユーザーが実行する活動のタイプに基づいて、そのユーザーをアクセス・グループに追加するときの属性として役割を使用することができます。たとえば、デフォルトではセラー管理者という役割と、それに対応する**セラー管理者**というメンバー・グループがあります。サイト管理者は WebSphere Commerce 管理コンソールを使って、サイトのアクセス・グループを作成、保守、および削除します。バイヤー管理者またはセラー管理者は、WebSphere Commerce の組織管理コンソールを使って、ユーザーへの役割の割り当てやアクセス・グルー

プへのユーザーの明示的割り当てを行います。アクセス・グループは、明示的または暗黙的のどちらか一方または両方にすることができます。

暗黙アクセス・グループ: 暗黙アクセス・グループは、一連の基準によって定義されます。その基準を満たす全員がグループのメンバーになります。通常、基準は、ユーザーの役割、親組織、または登録状況をベースとします。メンバー・グループのメンバーシップを定義する暗黙条件は、 MBRGRP テーブルの CONDITIONS 列に置かれます。ユーザーの属性を指定する暗黙アクセス・グループを使用すれば、互いに似通ったユーザーに対してアクセスを簡単に許可することができ、個々のユーザーを明示的に割り当てたり割り当て解除したりする手間が省けます。またその場合、ユーザーの属性が変更されても、グループのメンバーを更新しなくて済みます。アクセス・グループの単純な基準では、どの組織でユーザーが役割を果たすかに関係なく、特定の役割に割り当てられている全員がグループに編入されます。もっと複雑な基準では、特定の組織で一連の役割のうちの 1 つを果たすユーザーだけがそのアクセス・グループに属することが指定されます。

明示的アクセス・グループ: メンバー・グループに対するユーザーの明示的な追加または除去が可能です。そのような明示的な指定はどちらも、MBRGRPMBR テーブルを使って行うことができます。明示的アクセス・グループには、共通属性を共有するユーザーかどうかに関係なく、明示的に割り当てられたユーザーが属します。また、暗黙で定義されたグループへの編入の条件を満たしているけれども、さしあたって入れたくない個人を除外することもできます。

ユーザー・グループ: ユーザー・グループ (-1) のタイプのメンバー・グループは、マーチャントによる定義どおりの、共通の買い物候補を共有するユーザーの集まりです。ユーザー・グループは、常連客や上得意向けに大型ストアが開設するクラブに似通っています。ユーザー・グループの一員になれば、顧客は商品の購入時に割引やその他の利点を享受することができます。たとえば、年配の顧客は旅行案内書やカバンを繰り返し購入することが市場調査で明らかになった場合、そのような顧客を Seniors' Travel Club (シルバー・トラベル・クラブ) というメンバー・グループに割り当てることができます。同様に、商売上の常連顧客を優遇するためのユーザー・グループを作成することもできます。

アクション

一般的に、アクションとは、リソースに対して実行される操作のことです。コントローラー・コマンド用の役割をベースとするポリシーでは、アクションは Execute であり、リソースは、実行されるコマンドです。ビュー用の役割をベースとするポリシーでは、アクションはそのビュー名であり、リソースは com.ibm.commerce.commands.ViewCommand です。リソース・レベルのアクセス・コントロールの場合、通常は WebSphere Commerce コマンドにマップされるアクション、およびリソースは通常は、保護された EJB (Enterprise Java Bean) のリモート・インターフェースです。たとえば、コントローラー・コマンド com.ibm.commerce.order.commands.OrderCancelCmd は、com.ibm.commerce.order.objects.Order リソースを操作します。最後に、Display アクションは、データ Bean リソースをアクティブにするのに使用されます。

WebSphere Commerce 管理コンソールをサイト管理者が使って、既存のアクションをアクション・グループに関連付けることができますが、これは、新しいアクショ

ンの作成には使えません。新しいアクションを作成するには、XML ファイルに定義してから、データベースにロードします。アクションは、ACACTION テーブルに保管されます。

アクション・グループ

アクション・グループは、関連しあったアクションのグループです。アクション・グループの例には AccountManage グループがありますが、これには以下のコマンドが組み込まれています。

- com.ibm.commerce.account.commands.AccountDeleteCmd
- com.ibm.commerce.account.commands.AccountSaveCmd

サイト管理者のみが、アクション・グループを表示、作成、更新、および削除できます。それは、WebSphere Commerce 管理コンソールと XML を使って行います。アクション・グループは、ACACTGRP テーブルに保管されます。アクションは、ACACTACTGP テーブル内のアクション・グループに関連付けられます。

リソース・カテゴリー

リソース・カテゴリーとは、アクセス・コントロールによる保護を必要とするリソースのクラスを指します。リソースでは、保護可能なインターフェース情報をインプリメントする必要があります。リソース・カテゴリーには、オーダー、RFQ、およびオークションなどの Java クラスがあります。リソースは、そのようなクラスのインスタンスです。たとえば、オークション管理者 A によって作成された Auction1 は 1 つのリソースになり、オークション管理者 B によって作成された Auction2 はまた別のリソースになります。この 2 つのリソースは、オークションというリソース・カテゴリーに属します。

注: 保護可能インターフェースの詳細は、 *IBM WebSphere Commerce プログラマーズ・ガイド* を参照してください。

リソース・カテゴリーは、ACRESCGRY テーブル内に定義します。これは、リソースと呼ばれることもあります。サイト管理者は、WebSphere Commerce 管理コンソールを使って、既存のリソース・カテゴリーをリソース・グループに関連付けることができます。新規のリソース・カテゴリーは、XML を使って作成することができます。

リソース

リソースとは、保護の必要のあるシステム内のすべてのオブジェクトのことです。たとえば、RFQ、オークション、ユーザー、およびオーダーが、WebSphere Commerce 内で保護の必要のあるリソースの一例です。どのリソースにも所有者がいます。どのアクセス・コントロール・ポリシーをそのリソースに適用するかを決めるのに、リソースの所有権が使われます。アクセス・コントロール・ポリシーにも所有者がいますが、それは組織エンティティーです。ポリシーが適用されるのは、そのポリシーを所有する同一の組織エンティティーによって所有されるリソースに対してだけです。祖先の組織エンティティーによって所有されるポリシーもまた、そのリソースに適用されます。

コントローラー・コマンド・リソース: コントローラー・コマンド用の役割ベースのアクセス・コントロールの場合、 Execute アクションはコントローラー・コマンド・リソースに対して実行されるようにポリシーが構造化されます。そのようなポ

リシーでは、コントローラー・コマンドを実行できるのは、指定された役割を担ったユーザーに限定されることになっています。そのようなポリシーのアクセス・グループは通常、たとえばプロダクト・マネージャーという 1 つの役割をもつ人 (プロダクト・マネージャーの役割をもつ人) になります。次に、そのリソース・グループは、プロダクト・マネージャーが実行できる一連のコントローラー・コマンドになります。

役割ベースのアクセス・コントロールをコントローラー・コマンドで実行するときは、そのコマンドの所有者を判別する必要があります。それには、このコマンドで `getOwner()` メソッド (インプリメントされている場合) を呼び出します。通常はこのメソッドはインプリメントされていないので、WebSphere Commerce ランタイムは以下のいずれかを行ってこのメソッドを評価します。

- 現在コマンド・コンテキスト内にあるストアを所有する組織を使用します。
- コマンド・コンテキストにストアがない場合は、所有者としてルート組織を使用します。

データ Bean リソース: すべてのデータ Bean に保護が必要なわけではありません。既存の WebSphere Commerce アプリケーションでは、保護を必要とするデータ Bean に必要なアクセス・コントロールはすでにインプリメントされています。どれを保護すべきかの懸案事項は、新規にデータ Bean を作成するときに発生します。どのリソースを保護するかは、アプリケーションによって異なります。データ Bean の入った JSP (Java サーバー・ページ) に対応するビュー上の役割ベースのアクセス・コントロールによって十分に保護されていない情報を表示する予定の場合は、データ Bean を保護する必要があります (直接または間接的に)。

データ Bean が保護を必要としていて、しかも独自に存在する可能性がある場合は、直接保護される必要があります。データ Bean が存在するかどうかが、別のデータ Bean の存在によって決まる場合は、他のデータ Bean にゆだねて保護を行わなければなりません。直接保護されるデータ Bean の例としては、`Order` データ Bean があります。 `Order` データ Bean がないと存在できないために間接的に保護されるデータ Bean の例としては、 `OrderItem` データ Bean があります。データ Bean リソースを保護する方法の詳細は、*WebSphere Commerce プログラマーズ・ガイド、バージョン 5.4* を参照してください。

データ・リソース: データ・リソースとは、 オークション、オーダー、RFQ、およびユーザーなどの、操作可能なビジネス・オブジェクトを指します。これは、通常はエンタープライズ Bean レベルで保護されますが、保護可能インターフェースをインプリメントしさえすれば、どのクラスでも保護することができます。データ・リソースは、リソース・レベルのアクセス・コントロール検査を使って保護します。そのための最も一般的な方法では、コントローラーまたはタスク・コマンドの `getResources()` メソッド内でデータ・リソースを戻します。詳細は、*WebSphere Commerce プログラマーズ・ガイド、バージョン 5.4*を参照してください。

リソース・グループ

リソース・グループは、一連の関連しあったリソースを識別します。リソース・グループには、契約などのビジネス・オブジェクトや、それに関連した一連のコマンドを組み込むことができます。アクセス・コントロールではリソース・グループは、アクセス・コントロール・ポリシーによってアクセスを許可されるリソースを指定します。

リソース・グループは、ACRESGRP テーブル内に定義します。サイト管理者は、WebSphere Commerce 管理コンソールを使用するか、または XML を使用して、リソース・グループを管理し、リソースをリソース・グループに関連付けすることができます。

暗黙的なリソース・グループ: 暗黙的なリソース・グループは、特定の一連の属性に一致するリソースを定義します。そのような属性のうちの 1 つは、Java クラス名でなければなりません。その他の属性には、状況、ストア ID、価格、などがあります。たとえば、保留状況 (ORDERS.STATUS=P) になっているすべてのオーダーを収容する暗黙リソース・グループを作成することができます。通常、暗黙リソース・グループが使用されるのは、リソースが Java クラス名の範囲外の共通属性を共用するときに、リソース・レベルのポリシー内で使用されるそれらのリソースをグループ分けする場合です。

暗黙的なリソース・グループは、ACRESGRP テーブルの CONDITIONS 列を使って定義します。単純な暗黙のリソース・グループは、WebSphere Commerce 管理コンソールを使って作成することができます。もっと複雑なグループの場合は、XML を使用することができます。

明示的なリソース・グループ: 明示的なリソース・グループは、 1 つ以上のリソース・カテゴリーをリソース・グループに関連付けて指定します。その関連付けは、ACRESGPRES テーブル内で行います。 Java クラス名をリストすることでリソース・カテゴリーを明示的にグループに追加すると、必ずしも共通属性を共有するとは限らない個々のリソースを個別にグループ分けすることができます。

関係

どのリソースに対しても、他とのある種の関係付けと、その関係を実現するための一連のメンバーとの関連付けが行われます。たとえばどのリソースにも、*所有者* の関係付けが行われ、その関係はリソースの所有者によって実現されます。その他の関係には、マニュアルの受取人やオーダーの作成者などがあります。特定のリソース・インスタンスに対して誰が特定のアクションを実行できるかを決めるのに、そのようなリソース関係が重要になります。たとえば、文書の作成者がその文書を削除できなくても、監査担当者はできる場合があります。同様に、校閲者は文書を読んで承認できるだけで、転送したり他の操作を実行したりできない場合があります。

関係は ACRELATION テーブルに保管されますが、必要があれば ACPOLICY テーブルの ACRELATION_ID 列を使ってアクセス・コントロール・ポリシーに指定します。ユーザーとリソースの関係を確立する必要のあるポリシーを評価すると、その評価のためにリソースで fulfills(Long Member, String relationship) メソッドが呼び出されます。そのような関係を関係グループと比較した場合に、その関係を単純関係と呼ぶことがあります。

関係グループ: アクセス・コントロール・ポリシーでは、アクセス対象のリソースとの特定の関係をユーザーが確立しなければならないことを指定できますが、ユーザーが関係グループ内で指定されている条件を実現しなければならないことを指定してもかまいません。大半の場合、1 つの関係で十分です。ただし、もっと複雑な関係が必要な場合、代わりに関係グループを使用することができます。関係グループを使用すれば、複数の関係やチェーンになった関係を指定できます。どちらの場合も、チェーン関係を使って指定します。関係チェーンは、単純関係 (ユーザーと

リソースの直接関係) を表すことのできる構成ですが、これを使って、ユーザーとリソースの間の一連の関係を表すこともできます。たとえば、リソースとの関係をもつ組織 (所有者組織以外の) 内でユーザーが役割を持っている必要があることを表すには、関係グループを使用しなければなりません。この例では、ユーザーと組織の間には役割関係があり、組織とリソースの間にも関係があります。

関係と関係グループの比較: たいていの場合、関係を使用すれば、アプリケーションのアクセス・コントロール要件を満たすことになるはずです。大半の関係はユーザーとリソースの直接関係だからです。たとえば、ユーザーがリソースの作成者でなければならないことがポリシーで規定されることがあります。しかし、複数の関係を指定する必要がある場合は、関係グループを使用しなければなりません。たとえば、ユーザーがリソースの作成者または実行依頼者でなければならないことをポリシーで規定する場合があります。

関係グループはまた、ユーザーとリソースの間のチェーン関係を表すのにも必要です。チェーン関係では、たとえばオーダーで指定された購買組織にユーザーが所属するといった、ユーザーとリソースの間に直接関係はありません。そのような場合、ユーザーは組織との子関係をもち、組織はオーダーとの購買関係をもつことになります。

関係チェーン: すべての関係グループは、`andListCondition` または `orListCondition` エレメント別にグループに分けられた 1 つ以上の RELATIONSHIP_CHAIN のオープン条件で構成されます。関係チェーンとは、一つながりになった 1 つ以上の関係のことです。関係チェーンの長さは、チェーンを構成する関係の数によって判別します。これは、関係チェーンの XML 表記にある `<parameter name= "X" value="Y"/>` エントリーの数を調べれば判別することができます。以下に、1 の長さの関係チェーンの例を示しています。

```
<openCondition name="RELATIONSHIP_CHAIN">
<parameter name="RELATIONSHIP"
value="aValue"/>
</openCondition>
```

1 の長さの関係チェーンの場合、 `<parameter name="Relationship" value="something">` エレメントは、ユーザーとリソースの間の直接関係を指定します。 value 属性は、ユーザーとリソースの関係を表すストリングです。またこれは、保護可能リソース上の `fulfills()` メソッドの関係パラメーターに対応している必要もあります。

関係チェーンの長さが 2 の場合、それは一つながりになった 2 つの関係になります。最初の `<parameter name= "X" value="Y"/>` エレメントは、ユーザーと組織エンティティーを結び付けます。最後の `<parmeter name= "X" value="Y"/>` エレメントは、その組織エンティティーとリソースを結び付けます。以下に、2 の長さの関係チェーンの例を示しています。

```
<openCondition name=RELATIONSHIP_CHAIN">
<parameter name="aValue1" value="aValue2"/>
<parameter name="RELATIONSHIP" value="aValue3"/>
</openCondition>
```

aValue1 に指定できる値には、HIERARCHY や ROLE などがあります。 HIERARCHY は、メンバーシップ階層においてユーザーと組織エンティティーの間に階層関係があることを指定します。 ROLE は、組織エンティティー内でユーザーが役割を務めることを指定します。

aValue1 の値が HIERARCHY の場合、指定できる値には child などがありますが、これは、メンバー階層内でユーザーが直接の子として属する組織エンティティーを戻します。 aValue1 の値が ROLE の場合に指定できる値には、 ROLE テーブルの NAME 列の任意の有効エントリーがありますが、これは、現在のユーザーがこの役割を果たす場所であるすべての組織エンティティーを戻します。

aValue3 エントリーは、最初のパラメーターとリソースの評価から取り出された 1 つ以上の組織エンティティー同士の関係を表すストリングです。この値は、保護可能リソース上の fulfills() メソッドの関係パラメーターに対応します。パラメーター aValue1 の評価から複数の組織エンティティーが戻された場合に、その組織エンティティーのうちの少なくとも 1 つが、パラメーター aValue2 で指定された関係を満たしていれば、 RELATIONSHIP_CHAIN のこの部分は満足されたということです。

注: 単一のパラメーター・エレメントとの単一の関係チェーンで構成される関係グループは、機能的には単純関係と同等です。そのような場合、ポリシー内では関係グループではなく関係を使用するほうが簡単です。

リソースおよびポリシーの所有権

すべてのポリシーは、組織エンティティーによって所有されます。すべてのアクセス・コントロール・リソースもまた、通常は組織エンティティーである所有者をもっています。たとえば、オーダーの発行先のストアを所有する組織によってオーダーが所有されます。ユーザーも自身のリソースを所有することができます。たとえば、登録済みユーザーは、独自のユーザー登録情報を所有します。リソースおよびアクセス・コントロール・ポリシーの所有権は、特定のリソースに対してどのポリシーを適用するかを決定するときに重要になります。どのリソースの場合も、その所有者である組織エンティティーと、その所有者の祖先組織エンティティーに属するポリシーが適用されます。

アクセス・コントロール・ポリシーのタイプ

アクセス・コントロール・ポリシーは、以下の 2 つのタイプに分かれます。

- 標準ポリシー
- テンプレート・ポリシー

標準ポリシー

標準ポリシーは、固定所有者を持ちます。たとえば、標準ポリシーがセラー組織によって所有されている場合、セラー組織によって所有されているリソースと、その子孫組織エンティティー (ある場合) にだけそのポリシーは適用されます。ルート組織は WebSphere Commerce 内の他のすべての組織の祖先組織であるため、ルート組織 (メンバー ID = -2001) によって所有されると定義されたすべてのポリシーが、そのサイトのすべてのリソースに対して適用されます。というわけで、ルート組織によって所有される標準ポリシーを、サイト・レベル・ポリシーと呼ぶこともあります。

ルート組織によって所有されていない標準ポリシーは、組織レベル・ポリシーと呼ばれますが、それはサイト全体には適用されないからです。これは、そのポリシーの所有者によってか、またはその子孫組織エンティティーのいずれかによって所有されるリソースにしか適用されません。ストア管理者は、自身の組織エンティティーとその子孫組織エンティティーのポリシーを管理することができます。サイト管理者は、すべてのポリシーを変更することができます。

テンプレート・ポリシー

テンプレート・ポリシーは、動的所有者を持ちます。テンプレート・ポリシーは、リソースと祖先組織エンティティーを所有する組織エンティティーに対して動的に適用されます。たとえば、ルート組織の下に 10 個の組織がある場合に、そのいずれもが、ストア管理者が変更できるのは、そのストア管理者が役割を果たすために所属している組織が所有しているリソースだけにしたいと考えているとします。これをセットアップするには、次の 2 通りの方法があります。

1. アクセス対象のリソースに応じて、 10 個の組織すべてに動的に適用されるテンプレート・ポリシーを 1 つ用意します。このテンプレート・ポリシー内のアクセス・グループに対する基準もまた動的です。たとえば、組織 3 によって所有されるリソースにユーザーからアクセスしようとした場合、テンプレート・ポリシーの所有者は動的に組織 3 に変更され、そのアクセス・グループもまた、組織 3 に対して動的に自身の有効範囲を合わせます。つまり、ユーザーは、組織 3 のストア管理者の役割を果たす必要があるということです。
2. それぞれが 10 個の組織のうちの 1 つを所有する 10 個のポリシーを用意します。組織 1 のアクセス・グループでは、ユーザーは組織 1 のストア管理者の役割を果たす必要があることが指定されます。組織 2 のアクセス・グループでは、ユーザーは組織 2 のストア管理者の役割を果たす必要があることが指定されます。

最初のソリューションの長所は、ポリシーの物理コピーは 1 つしかない点にあります。ただし、論理コピーは 10 個あります。テンプレート・ポリシーは、サイト管理者が管理することができます。

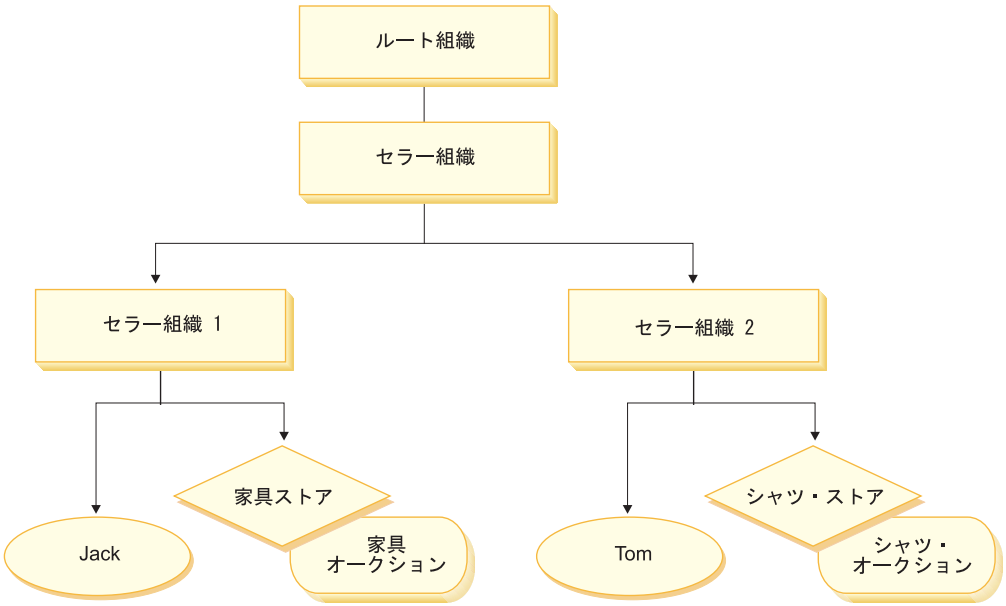
テンプレート・ポリシーのオーバーライド: テンプレート・ポリシーの別のフィーチャーとして、指定した組織エンティティーのテンプレート・ポリシーをオーバーライドすることができます。上記の例に戻って、WebSphere Commerce サイトに 11 番目の組織エンティティーが追加された場合に、その最新の組織エンティティーが上記のテンプレート・ポリシーを自身に適用したくないときは、そのように指定する方法があります。 ACORGPOL テーブルにエントリーを追加して、テンプレート・ポリシーの ID と、11 番目の組織の組織エンティティー ID を指定する必要があります。これは、特定の組織に関連してストア管理者がテンプレート・ポリシーを削除または更新するときに、WebSphere Commerce 管理コンソールを使って行うこともできます。

ルート組織の子孫組織用のテンプレート・ポリシーをオーバーライドしても、そのテンプレート・ポリシーは、ルート組織レベルではこれまでどおりに適用されます。テンプレート・ポリシーが、より厳密なポリシーを使って子孫組織レベルでオーバーライドされた場合、そのテンプレート・ポリシーをルート組織レベルでもオーバーライドする必要があります。次のような SQL を実行するのが、ルート組織のテンプレート・ポリシーをオーバーライドする唯一の方法です。

```
insert into ACORGPOL (acpolicy_id, member_id) values ( (select acpolicy_id from
ACPOLICY where policyname = 'policyToOverride'), -2001)
```

アクセス・コントロールのレベル

WebSphere Commerce でのアクセス・コントロールのレベルは、コマンド・レベル (役割ベースともいう) とリソース・レベル (インスタンス・ベースともいう) の 2 つに大別されます。



コマンド・レベルまたは役割ベースのアクセス・コントロール

コマンド・レベルまたは役割ベースのアクセス・コントロールは、概括的なアクセス・コントロールです。これは「誰が何を行えるか」を決定します。役割ベースのアクセス・コントロールでは、特定の役割をもつすべてのユーザーが、特定のコマンドを実行できるように指定することができます。セラーがセラー・コマンドを実行できるというアクセス・コントロール・ポリシーについて考察してみます。なおこのポリシーでは、セラー・コマンドの 1 つを ModifyAuction コマンドであるとします。上図では、Jack と Tom はどちらもセラーなので、両者ともオークションを作成することができます。

コントローラー・コマンドおよびビューでは、役割ベースのアクセス・コントロールを使用します。このタイプのアクセス・コントロールでは、コマンドの実行対象のデータ・リソースに対する配慮はなされていません。単に、ユーザーが特定のコントローラー・コマンドまたはビューを実行できるかどうかを判別するだけです。

このレベルのアクセス・コントロールは必須なので、ランタイムによって規定されています。すべてのコントローラー・コマンドは、コマンド・レベルのアクセス・コントロールによって保護されなければなりません。さらに、直接呼び出せるビュー、または別のコマンドからリダイレクトに起動できる (ビューへの転送によって起動される場合とは対照的に) ビューはすべて、コマンド・レベルのアクセス・コントロールによって保護されなければなりません。

コントローラー・コマンドの場合のコマンド・レベルのアクセス・コントロール:
コントローラー・コマンドを実行する場合は常に、ユーザーがコマンド・リソースに対して Execute アクションを実行することを認可するアクセス・コントロール・ポリシーがなければなりません。リソースはコントローラー・コマンドのインターフェース名です。アクセス・グループは通常、1 つの役割を当てはめられています。たとえば、アカウント担当者の役割をもつユーザーが、AccountRepresentativesCmdResourceGroup リソース・グループで任意のコマンドを実行できるように指定することができます。

ビューの場合のコマンド・レベルのアクセス・コントロール: ビューを URL から直接呼び出したり、またはコマンドからのリダイレクトの結果として呼び出すには、そのビューがアクセス・コントロール・ポリシーをもっていなければなりません。そのようなポリシーには、 ACACTION テーブル内でアクションとして viewname が指定されていなければなりません。次に、ACACTACTGP テーブルを使って、そのアクションをアクション・グループに関連付ける必要があります。次に、ACPOLICY テーブルで、該当するコマンド・レベル・ポリシーにおいてそのアクション・グループを参照する必要があります。

インスタンス・レベルまたはリソース・レベルのアクセス・コントロール

インスタンス・レベルまたはリソース・レベルのアクセス・コントロール・ポリシーは、 誰がどのリソースでどのコマンドを実行できるかを決定するち密なアクセス・コントロールを行う手段になります。上記の役割ベースのアクセス・コントロール・ポリシーの例ではセラーがオークションを変更できましたが、この例をリソース・レベルのアクセス・コントロールに沿って微調整すれば、セラーが役割を果たす対象の組織によって所有されているオークションを変更することができます。31 ページでは Jack は組織 1 でセラーの役割をもち、 Tom はセラー組織 2 でセラーの役割をもっており、Jack は家具ストアで家具オークションを作成し、 Tom は、シャツ・ストアでシャツ・オークションを作成します。 Jack は家具オークションを変更できますが、シャツ・オークションを変更することはできません。 Tom はシャツ・オークションを変更できますが、家具オークションを変更することはできません。

要するに、まずシステムはコマンド・レベルのアクセス・チェックを行うということです。ユーザーが特定のコマンドを実行できる場合は、その後続のリソース・レベルのアクセス・コントロール・ポリシーが適用され、該当するリソースにユーザーがアクセスできるかどうかが判別されます。

リソース・レベルのアクセス・コントロールは、コマンドとデータ Bean に対して適用されます。

コマンドでのリソース・レベルのアクセス・コントロール: コマンド・レベルのアクセス・コントロールの検査が完了し、アクセスが認可されると、次に以下の 2 つのケースのいずれかの場合にはリソース・レベルの検査が行われます。

- コマンドが getResources() をインプリメントしている — このメソッドは、現在のアクションに照らし合わせてチェックする必要のあるリソース・インスタンスを指定します。ただしコマンドは、ここではアクションになっています。WebSphere Commerce ランタイムでは、 getResources() で指定されたすべての

リソースに現行ユーザーからアクセスできることになります。デフォルトでは、getResources() はヌルを戻しますが、これは、リソース・レベルのチェックを行わないということです。

- コマンドが checkIsAllowed(オブジェクト・リソース，ストリング・アクション) を呼び出す — ランタイムによって getResources() が呼び出されたときに、どのリソースをチェックする必要があるかがコマンド作成者には分からない場合、コマンドは必要に応じてこの checkIsAllowed() メソッドを呼び出して、現在のアクションとリソースのペアは許可を受けているかどうかを確かめることができます。そのアクションは通常は、現在のコマンドのインターフェース名です。このメソッドの呼び出し時にアクセスが拒否されると、 ECApplicationException (ECMessage._ERR_USER_AUTHORITY, ..) という例外がスローされます。

データ Bean でのリソース・レベルのアクセス・コントロール: 上記のとおり、通常は役割をベースとするコマンド・レベルのポリシーによってビューは保護されます。たとえば、コマンド・レベル・ポリシーで、セラー管理者はある特定のビューにアクセスできると指定することができます。多くの場合、JSP 上のデータ Bean がすべて、セラー管理者の役割を果たすユーザーの組織に関連付けられていることをさらに確認する必要があります。そのためには、保護 (直接または間接的に) を必要とするすべてのデータ Bean で Delegator インターフェースをインプリメントします。そのようなデータ Bean が 1 次 (独立した) データ Bean を代行すると、後者の Bean は次に保護可能インターフェースをインプリメントします。 1 次データ Bean は自身を代行することになるので、両方のインターフェースがインプリメントされることになります。これで、データ Bean 管理者の activate() メソッドを使ってデータ Bean が呼び出されるたびに、現在のユーザーが 1 次データ Bean リソースで表示アクションを実行する権限を認可するポリシーがあるかどうかが WebSphere Commerce ランタイムによって確認されます。

アクセス・コントロールでの無許可アクションの防止

この項では、許可を受けたアクションだけをユーザーが実行できるようにするために、ポリシー・ベースのアクセス・コントロールがどのように作動するかについて説明します。

ユーザー始動のアクションの実行前の許可チェック

ポリシー・マネージャー は、指定されたリソースで指定されたアクションを現行ユーザーが実行することを許可されているかどうかを判別するアクセス・コントロール・コンポーネントです。アクセス・コントロール・ポリシーは XML 形式で指定されます。インスタンスの作成時に、該当するデータベース・テーブルにデフォルト・ポリシーがロードされます。 WebSphere Commerce アプリケーション・サーバーの始動時にアクセス・コントロール情報はキャッシュに入れられるので、ポリシー・マネージャーは必要があればユーザーが許可を受けているかどうかを機敏にチェックすることができます。 WebSphere Commerce 管理コンソールを使用したり、または XML ポリシー・データをロードしたりすることで、データベース内のアクセス・コントロール情報を変更した場合、アクセス・コントロールのキャッシュを更新する必要があります。更新するには、WebSphere Commerce 管理コンソール内のアクセス・コントロール・レジストリーを更新します。 WebSphere Commerce を再始動した場合も、キャッシュが更新されることになります。

アクセス・コントロールで保護されたアクションをユーザーが実行しようとすると、そのユーザーが許可を受けていることを確認するためのアクセス・コントロール・チェックが行われます。ポリシー・マネージャーは、リソースを所有する組織に適用されるすべてのアクセス・コントロール・ポリシーを探します。次に、それらのポリシーを調べて、ターゲット・リソースでそのアクションを実行する許可をユーザーが受けているかどうかを評価します。そのようなポリシーが少なくとも 1 つあれば、ポリシー・マネージャーはアクセスを認可し、なければアクセスを拒否します。

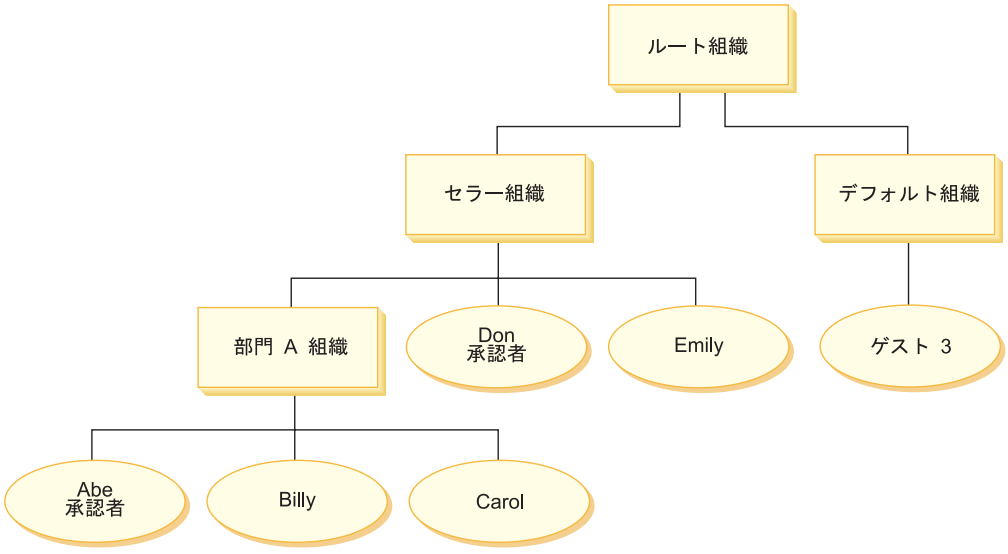
アクセス・コントロールの使用

デフォルトのアクセス・コントロール・ポリシーのカスタマイズや、シナリオのカスタマイズ、および XML ファイルを使ったアクセス・コントロール・ポリシーのカスタマイズなどのタスクの詳細は、 『WebSphere Commerce アクセス・コントロール・ガイド』を参照してください。

アクセス・コントロール・ポリシーの評価

この項は、アクセス・コントロール・ポリシーを評価する際の参考として活用することができます。この項では、シナリオを提示してから、標準およびテンプレートのアクセス・コントロール・ポリシーの評価方法の例を順を追って説明します。どの項も、関連しあったポリシーについてと、各ポリシーを使ったシナリオについての説明で始まっています。標準ポリシーとテンプレート・ポリシーの詳細は、 29 ページの『アクセス・コントロール・ポリシーのタイプ』を参照してください。

以下の図は、シナリオを示しています。



組織階層

この図では、サイト内に次のような 4 つの組織があります。

- ルート組織
- セラー組織
- デフォルト組織

- 部門 A 組織

図で明らかなように、ルート組織は、セラー組織とデフォルト組織の親です。また、セラー組織は、部門 A 組織の親です。

ユーザー

この図では、Don と Emily はセラー組織に登録されています。 Abe、Billy、および Carol は、部門 A 組織に登録されています。ゲスト 3 は未登録ですが、アクセス・コントロールの目的に準じてデフォルト組織に暗黙で所属しています。

役割

Don は、セラー組織において承認者の役割を担っています。 Abe は、部門 A 組織において承認者の役割を担っています。

アクセス・グループ

このシナリオでは、次のようなアクセス・グループが使われています。

- 登録済みユーザー: このグループには、登録されているすべてのユーザーが暗黙で所属します。
- セラーの承認者: このグループには、セラー組織で承認者の役割を担うすべてのユーザーが暗黙で所属します。
- 部門 A の承認者: このグループには、部門 A 組織で承認者の役割を担うすべてのユーザーが暗黙で所属します。

文書

文書オブジェクトは、保護対象のリソースです。文書の所有者は、その作成場所である組織であると定義されています。

文書の更新でのアクセス・コントロールの要件

以下に、文書を更新する際のアクセス・コントロール要件を示します。

1. 登録済みユーザーは、文書の作成者であればその文書を更新することができます。
2. 部門 A の承認者は、部門 A によって所有される文書を更新することはできますが、セラーによって所有される文書を更新することはできません。セラー組織の承認者は、組織 A とセラー組織によって所有される文書を両方更新することができます。

標準ポリシーの評価

この項では、標準ポリシーについてと、それを評価するシナリオについて順に説明しています。

文書の更新に関連したアクセス・コントロール・ポリシー

以下に、文書の更新に関連したポリシーの形式とアクセス・コントロール・ポリシーを示してあります。

ポリシーの形式: `[Access Group, Action Group, Resource Group, Relationship]`

ポリシー 1:

```
[Registered Users, Execute Command Action Group,  Update Document
Resource Group, - ]
```

これは、ルート組織によって所有される役割ベースの標準ポリシーです。このポリシーでは、登録ユーザーは Update Document コマンドを実行することができます。

ポリシー 2:

```
[Registered Users, Update Document Action Group, document, creator ]
```

これは、ルート組織によって所有されるリソース・レベルの標準ポリシーです。このポリシーでは登録ユーザーは、文書の作成者であればその文書を更新することができます。

ポリシー 3:

```
[Approvers for Seller, Update Document Action Group, document, - ]
```

これは、セラー組織によって所有されるリソース・レベルの標準ポリシーです。このポリシーでは、セラーの承認者は、セラーによって所有される文書を更新することができます。

ポリシー 4:

```
[Approvers for Division A, Update Document Action Group, document, - ]
```

これは、部門 A 組織によって所有されるリソース・レベルの標準ポリシーです。このポリシーでは、部門 A の承認者は、部門 A によって所有される文書を更新することができます。

シナリオ

シナリオ 1 : Billy は、自身の文書を更新しようとしています。: 以下は、このシナリオの場合のアクセス・コントロール評価です。

コマンド・レベル検査:

1. ストア ID が指定されていないので、このコマンドの所有者はルート組織に設定されます。したがって、ユーザーがコマンド・レベル・アクセスできるかどうかを評価するのに、ルート組織によって所有されているポリシーだけが使用されます。ちなみに、ポリシー 1 および 2 がルート組織によって所有されています。
2. ポリシー 1 がアクセスを認可します。Billy は登録ユーザー・アクセス・グループのメンバーであって、 Update Document コマンド・リソースに対して Execute アクションを実行するからです。

リソース・レベル検査:

1. Update Document コマンドは、文書リソースを保護するよう指定します。 Billy の文書は、部門 A によって所有されます。したがって、部門 A とその祖先組織が所有するポリシー、つまりポリシー 1、2、3、および 4 だけが適用されます。
2. ポリシー 2 がアクセスを認可します。Billy は登録ユーザー・アクセス・グループのメンバーであって、文書リソースに対して Update Document コマンド・アクションを実行し、しかもその文書に対して作成者の関係を確立しているからです。

Billy は、コマンド・レベルとリソース・レベルの両方のアクセス・コントロール検査に通ったので、自身の文書を更新してもかまいません。

シナリオ 2: Don は、Carol の文書を更新しようとしています。: 以下は、このシナリオの場合のアクセス・コントロール評価です。

コマンド・レベル検査:

1. ストア ID が指定されていないので、このコマンドの所有者はルート組織に設定されます。したがって、ユーザーがコマンド・レベル・アクセスできるかどうかを評価するのに、ルート組織によって所有されているポリシーだけが使用されます。ちなみに、ポリシー 1 および 2 がルート組織によって所有されています。
2. ポリシー 1 がアクセスを認可します。Don は登録ユーザー・アクセス・グループのメンバーであって、 `Update Document` コマンド・リソースに対して `Execute` アクションを実行しているからです。

リソース・レベル検査:

1. `Update Document` コマンドは、文書リソースを保護するよう指定します。 Carol の文書は、部門 A によって所有されます。したがって、部門 A とその祖先組織が所有するポリシー、つまりポリシー 1、2、3、および 4 だけが適用されます。
2. ポリシー 4 がアクセスを認可します。Don は登録ユーザー・アクセス・グループのメンバーであって、文書リソースに対して `Update Document` コマンド・アクションを実行しているからです。

Don は、コマンド・レベルとリソース・レベルの両方のアクセス・コントロール検査に通ったので、 Carol の文書を更新してもかまいません。

シナリオ 3: Abe は、Emily の文書を更新しようとしています。: 以下は、このシナリオの場合のアクセス・コントロール評価です。

コマンド・レベル検査:

1. ストア ID が指定されていないので、このコマンドの所有者はルート組織に設定されます。したがって、ユーザーがコマンド・レベル・アクセスできるかどうかを評価するのに、ルート組織によって所有されているポリシーだけが使用されます。ちなみに、ポリシー 1 および 2 がルート組織によって所有されています。
2. ポリシー 1 がアクセスを認可します。Abe は登録ユーザー・アクセス・グループのメンバーであって、 `Update Document` コマンド・リソースに対して `Execute` アクションを実行するからです。

リソース・レベル検査:

1. `Update Document` コマンドは、文書リソースを保護するよう指定します。 Emily の文書は、セラー組織によって所有されます。したがって、セラー組織とその祖先組織が所有するポリシー、つまりポリシー 1、2、および 3 だけが適用されます。
2. Abe はセラー・アクセス・グループの承認者のメンバーでは**ない**ので、ポリシー 3 はアクセスを認可**しません**。

Abe は、コマンド・レベル検査には通ったけれども、リソース・レベルのアクセス・コントロール検査には通らなかったので、 Emily の文書を更新することはできません。

シナリオ 4 : ゲスト 3 は、自身の文書を更新しようとしています。: 以下は、このシナリオの場合のアクセス・コントロール評価です。

コマンド・レベル検査:

1. ストア ID が指定されていないので、このコマンドの所有者はルート組織に設定されます。したがって、ユーザーがコマンド・レベル・アクセスできるかどうかを評価するのに、ルート組織によって所有されているポリシーだけが使用されます。ちなみに、ポリシー 1 および 2 がルート組織によって所有されています。
2. ゲスト 3 は**登録ユーザー・アクセス・グループのメンバーでは<u>ない</u>**ので、ポリシー 1 はアクセスを認可**<u>しません</u>**。

リソース・レベル検査:

1. コマンド・レベル検査に通らなかったため、リソース・レベル検査までは行われません。

ゲスト 3 はコマンド・レベル検査に通らなかったので、自身の文書を更新できません。

テンプレート・ポリシーの評価

以下の例は、上記のシナリオをベースにしています。

文書の更新に関連したアクセス・コントロール・ポリシー

テンプレート・ポリシーの評価の際は、標準ポリシーの評価に使ったアクセス・コントロール・ポリシー 1 および 2 がやはり適用されますが、今回は標準ポリシー 3 と 4 はテンプレート・ポリシー 5 に置き換えられています。ポリシー 1 と 2 に関する詳細は、 35 ページの『標準ポリシーの評価』を参照してください。

ポリシー 5:

```
[Approvers for Organization, Update Document Action Group, document, - ]
```

これは、リソース・レベルのテンプレート・ポリシーです。文書を所有する組織の承認者は、文書を更新することができます。

また、このテンプレート・ポリシーで使用される新規のパラメーター化アクセス・グループも必要になります。このシナリオには、次のようなアクセス・グループが追加されています。

- 組織の承認者: このグループには、? 組織で承認者の役割を担うすべてのユーザーが暗黙で所属します。 (実行時にテンプレート・ポリシーが適用されると、? パラメーターはポリシー所有者に動的に変更されます。)

シナリオ

以下のシナリオでは、ポリシー 1、2、および 5 のみが使用されています。

シナリオ 1: Don は、Carol の文書を更新しようとしています。: 以下は、このシナリオの場合のアクセス・コントロール評価です。

コマンド・レベル検査:

1. ストア ID が指定されていないので、このコマンドの所有者はルート組織に設定されます。したがって、ユーザーがコマンド・レベル・アクセスできるかどうかを評価するのに、ルート組織によって所有されているポリシーだけが使用されます。ちなみに、ポリシー 1 および 2 がルート組織によって所有されています。ポリシー評価中にテンプレート・ポリシーは、所有権を、リソースを所有する組織に動的に変更し、次にその組織の祖先に変更するので、ポリシー 5 も適用されます。
2. ポリシー 1 がアクセスを認可します。Don は登録ユーザー・アクセス・グループのメンバーであって、 `Update Document` コマンド・リソースに対して `Execute` アクションを実行しているからです。

リソース・レベル検査:

1. `Update Document` コマンドは、文書リソースを保護するよう指定します。 Carol の文書は、部門 A によって所有されます。したがって、部門 A とその祖先組織が所有するポリシー、つまりポリシー 1 と 2 が適用されます。ポリシー評価中にテンプレート・ポリシーは、所有権を、リソースを所有する組織に動的に変更し、次にその組織の祖先に変更するので、ポリシー 5 も適用されます。
2. テンプレート・ポリシー 5 は、まずリソースを所有する組織である部門 A に対して適用されます。この時点で、ポリシー 5 は次のように基本的にはポリシー 5a に似た行動をとります。

   ```
   [Approvers for Division A, Update Document Action Group, document, - ] standard
   resource-level policy owned by Division A.
   ```

3. Don は部門 A アクセス・グループの承認者のメンバーでは**ない**ので、ポリシー 5a はアクセスを認可**しません**。
4. テンプレート・ポリシー 5 は次に、部門 A の親組織であるセラー組織に対して適用されます。この時点で、ポリシー 5 は次のように基本的にはポリシー 5b に似た行動をとります。

   ```
   [Approvers for Seller, Update Document Action Group, document, - ] standard
   resource-level policy owned by Seller
   ```

5. ポリシー 5b はアクセスを認可します。Don はセラー・アクセス・グループの承認者のメンバーであって、文書リソースに対して `Update Document` コマンド・アクションを実行しているからです。

Don は、コマンド・レベルとリソース・レベルの両方のアクセス・コントロール検査に通ったので、 Carol の文書を更新してもかまいません。

シナリオ 2: Abe は、Emily の文書を更新しようとしています。: 以下は、このシナリオの場合のアクセス・コントロール評価です。

コマンド・レベル検査:

1. ストア ID が指定されていないので、このコマンドの所有者はルート組織に設定されます。したがって、ユーザーがコマンド・レベル・アクセスできるかどうかを評価するのに、ルート組織によって所有されているポリシーだけが使用されます。ちなみに、ポリシー 1 および 2 がルート組織によって所有されています。ポリシー評価中にテンプレート・ポリシーは、所有権を、リソースを所有する組織に動的に変更し、次にその組織の祖先に変更するので、ポリシー 5 も適用されます。

2. ポリシー 1 がアクセスを認可します。Abe は登録ユーザー・アクセス・グループのメンバーであって、 Update Document コマンド・リソースに対して Execute アクションを実行するからです。

リソース・レベル検査:

1. Update Document コマンドは、文書リソースを保護するよう指定します。 Emily の文書は、セラー組織によって所有されます。したがって、セラーとその祖先組織が所有するポリシー、つまりポリシー 1 と 2 が適用されます。ポリシー評価中にテンプレート・ポリシーは、所有権を、リソースを所有する組織に動的に変更し、次にその組織の祖先に変更するので、ポリシー 5 も適用されます。
2. テンプレート・ポリシー 5 は、まずリソースを所有する組織であるセラー組織に対して適用されます。この時点で、ポリシー 5 は次のように基本的にはポリシー 5a に似た行動をとります。

   ```
   [Approvers for Seller, Update Document Action Group, document, - ] standard
   resource-level policy owned by Seller
   ```

3. Abe はセラー・アクセス・グループの承認者のメンバーでは**ない**ので、ポリシー 5a はアクセスを認可**しません**。
4. テンプレート・ポリシー 5 は次に、セラー組織の親組織であるルート組織に対して適用されます。この時点で、ポリシー 5 は次のように基本的にはポリシー 5b に似た行動をとります。

   ```
   [Approvers for Root, Update Document Action Group, document, - ] standard
    resource-level policy owned by Root
   ```

5. Abe はルート・アクセス・グループの承認者のメンバーでは**ない**ので、ポリシー 5b はアクセスを認可**しません**。
6. ルート組織は親組織をもたないので、これでテンプレート・ポリシー 5 の評価は完了です。

Abe は、コマンド・レベル検査には通ったけれども、リソース・レベルのアクセス・コントロール検査には通らなかったので、 Emily の文書を更新することはできません。

第 2 部 WebSphere Commerce サイト管理者のセキュリティー・タスク

第 2 部では、WebSphere Commerce サイト管理者が通常実行できるセキュリティー・タスクについて説明します。

第 4 章 サイト・セキュリティーの機能強化

以下の WebSphere Commerce 構成マネージャーのフィーチャーのいずれかを使用可能にして、WebSphere Commerce サイトのセキュリティーを強化することができます。

- 「ログイン・タイムアウト」ノードを使って、一定期間を超えて非アクティブになっているユーザーをログオフさせ、元のシステムにログオンするよう要求します。詳細は、 47 ページの『ログイン・タイムアウトの使用可能化』を参照してください。
- ユーザーが初めてシステムにログインしたときに、「パスワード無効化」ノードを使って各自のパスワードを変更することを義務付けます。詳細は、 48 ページの『パスワード無効化の活動化』を参照してください。
- 「パスワード保護されたコマンド」ノードを使って、指定コマンドを実行する要求を実行する場合はパスワードを入力することをユーザーに義務付けます。詳細は、 48 ページの『パスワード保護コマンドの使用可能化』を参照してください。
- 構成マネージャーの「データベース更新ツール」ノードを使って、パスワードやクレジット・カードの情報などの暗号化データならびに WebSphere Commerce データベース内のマーチャント・キーを更新します。詳細は、 49 ページの『暗号化データの更新』を参照してください。
- 「サイト間スクリプト保護」ノードを使って、不許可と指定された属性や文字を使用しているユーザー要求を拒否します。詳細は、 50 ページの『サイト間スクリプト保護の使用可能化』を参照してください。
- アクセス・ロギングの使用可能化によって、WebSphere Commerce に対するセキュリティー上の脅威をすべて早急に特定します。詳細は、 53 ページの『アクセス・ロギングの使用可能化』を参照してください。

それ以外に、WebSphere Commerce の管理コンソールの「セキュリティー」ドロップダウンから、次のようなフィーチャーを使用可能にすることができます。

- 「アカウント・ポリシー」ページを使って、使用中のアカウント関連のポリシーを定義するためのサイト用のアカウント・ポリシーをセットアップします。詳細は、 54 ページの『アカウント・ポリシーのセットアップ』を参照してください。
- 「パスワード・ポリシー」ページを使って、ユーザーのパスワード選択特性を制御するためのサイト用のパスワード・ポリシーをセットアップします (ユーザーが WebSphere Commerce データベースに対する認証を受けている場合のみ)。詳細は、 55 ページの『パスワード・ポリシーのセットアップ』を参照してください。
- 「アカウント・ロックアウト・ポリシー」ページを使って、ユーザー・アカウントに不祥事が起きる可能性を減少するためにサイト用のアカウント・ロックアウト・ポリシーをセットアップします (ユーザーが WebSphere Commerce データベースに対する認証を受けている場合のみ)。詳細は、 56 ページの『アカウント・ロックアウト・ポリシーのセットアップ』を参照してください。

- 「セキュリティー検査の立ち上げ」ページを使って、機密漏れの可能性があると思われる一時 WebSphere Commerce ファイルの検査と削除を行うためのセキュリティー・プログラムを立ち上げます。詳細は、57 ページの『セキュリティー検査の立ち上げ』を参照してください。

関連概念の詳細は、以下の WebSphere Commerce オンライン・ヘルプの中のトピックを参照してください。

- 構成マネージャー
- WebSphere Commerce 構成ファイル
- 管理コンソール
- セキュリティー

関連タスクの詳細は、以下の WebSphere Commerce オンライン・ヘルプの中のトピックを参照してください。

- 構成マネージャーの立ち上げ
- 管理コンソールのオープン

セキュリティー用のビュー

WebSphere Commerce のいずれかのセキュリティー・フィーチャーを使用したい場合に、そのフィーチャーを使用可能にするには、ストア関連のビューを事前に定義する必要があります。この後の項では、次のような機能用にビューを定義する方法を説明しています。

- ログイン・タイムアウト (『ログイン・タイムアウト』を参照)
- パスワード無効化 (45 ページの『パスワードの無効化』を参照)
- パスワード保護コマンド (46 ページの『パスワード保護コマンド』を参照)
- サイト間スクリプト保護 (46 ページの『サイト間スクリプト保護』を参照)

ビューの作成とストア・フロントの開発に関する一般情報は、*ストア開発者ガイド* を参照してください。

ログイン・タイムアウト

ログイン・タイムアウトのセキュリティー・フィーチャーを使用するには、ストア用の LoginTimeoutErrorView と ReLogonFormView ビューを定義する必要があります。

LoginTimeoutErrorView

ログイン・タイムアウト情報が誤っていると、WebSphere Commerce はこのビューをユーザーにリダイレクトします。ビューがリダイレクトされた場合、その原因は誰かが cookie を悪用したためと考えられます。

表 1. LoginTimeoutErrorView の属性

ECConstants.EC_LOGIN_TIMEOUT_ERROR_MSGCODE	**1** 有効期限は誤った値に設定されています。 **2** ログオン時刻は誤った値に設定されています。 **3** 有効期限またはログオン時刻は誤った値に設定されています。

ReLogonFormView

このビューは、セッションの期限が切れた後でユーザーに対して表示されます。このビューでは、ユーザーのログオン ID とパスワードを入力するためのフォームが表示されなければなりません。送信ボタンを使うと、ログオン・コマンドが起動されます。また、たいていはストア・フロント・ページなどの別のページにユーザーをリダイレクトするための「取り消し」ボタンもなければなりません。

ReLogonFormView には属性はありません。

表 2. ReLogonFormView フォームの属性

ECUserConstants.EC_UREG_LOGONID	ユーザーのログオン ID。
ECUserConstants.EC_UREG_LOGONPASSWORD	ユーザーのログオン・パスワード。
ECUserConstants.EC_RELOGIN_URL	入力した認証情報が無効の場合に表示される URL。たいていの場合、それはこのビューの名前です。
ECConstants.EC_STORE_ID	ストア ID。
ECConstants.EC_URL	入力した認証情報が別のユーザーのものである場合に表示される URL。たいていの場合それはストアのホーム・ページであるか、またはログオン・ページで使用されているのと同じ URL です。

パスワードの無効化

パスワード無効化のセキュリティー・フィーチャーを使用するには、ストア用の ChangePassword ビューを定義する必要があります。

ChangePassword

このビューが表示されるのは、ユーザーのパスワードの期限が切れた場合です。このビューでは、現在 (期限の切れた) パスワードと新規パスワードを入力するためのフォームが表示されなければなりません。「送信」ボタンを使うと、ResetPassword コマンドが起動されます。また、たいていはストア・フロント・ページなどの別のページにユーザーをリダイレクトするための「取り消し」ボタンもなければなりません。

表 3. ChangePassword の属性

ECConstants.EC_PASSWORD_EXPIRED_FLAG	**1** ユーザーのパスワードの有効期限は切れています。この属性が必要なのは、同じパスワード変更のフィーチャーとして使用するビューとこのビューを区別するためです。パスワード変更のビューは、ユーザーによって起動されますが、このビューに割り当てられる JSP はどちらの場合も同じでなければなりません。 JSP はこの属性を見つけ出して、何を表示すればよいかを決める必要があります。 **ヌル** 属性は URL 上にありません。これは、パスワード変更時の通常の動作です。
ECUserConstants.EC_UREG_LOGONID	現行ユーザーのログオン ID。
ECConstants.EC_LOGIN_RETURN_URL	パスワード変更が正常に完了した後でブラウザーがリダイレクトされる先の URL。この URL は、ECConstants.EC_URL という名前でアクション・コマンドに渡されます。

表 4. ChangePassword フォームの属性

ECUserConstants.EC_UREG_LOGONID	ユーザーのログオン ID。現在のログオン ID は、ビューに渡されました。
ECUserConstants.EC_UREG_LOGONPASSWORDOLD	旧パスワード。
ECUserConstants.EC_UREG_LOGONPASSWORD	新規パスワード。

表 4. ChangePassword フォームの属性 (続き)

ECUserConstants.EC_UREG_LOGONPASSWORDVERIFY	新規パスワードの検査。
ECConstants.EC_URL	パスワード変更の正常完了後にユーザーがリダイレクトされる先の URL。値は、ビューに渡されました。
ECUserConstants.EC_RELOGIN_URL	パスワード変更が失敗した場合にブラウザーがリダイレクトされる先の URL。

パスワード保護コマンド

パスワード保護コマンドのセキュリティー・フィーチャーを使用するには、ストア用の PasswordReEnterErrorView および PasswordReEnterFormView ビューを定義する必要があります。

PasswordReEnterErrorView

このビューは、以下のシナリオで使用されています。

- ユーザーは、正しいパスワードを入力しなかったので、ログオフします。
- 認証が失敗しました。

どちらの場合も、ユーザーは現行ページ上のリンクを使って別のページに進むための手段を講じる必要があります。

表 5. PasswordReEnterErrorView の属性

ECConstants.EC_PASSWORD_REREQUEST_MSGCODE	**0** ユーザーを認証しようとしたときに問題が起きました。 **ヌル** 属性は URL 上にありません。ユーザーは、パスワードを入力しなかったのでログオフします。

PasswordReEnterFormView

このビューが表示されるのは、ユーザーがパスワード保護コマンドを実行しようとした場合です。パスワードを入力するためのフォームをユーザーに提示しなければなりません。パスワード用の入力フィールドが 2 つなければなりません。

表 6. PasswordReEnterFormView の属性

ECConstants.EC_PASSWORD_REREQUEST_URL	URL は、このフォームの「送信」ボタンを使って実行されます。
ECConstants.EC_PASSWORD_REREQUEST_MSGCODE	ユーザーに対して示されるメッセージを指定するメッセージ・コードは次のとおりです。 **1** 入力したパスワードは一致しません。 **2** パスワードが入力されていません。 **3** 誤ったパスワードを入力しました。

アクション: 以下の名前のパラメーターとして URL が渡されます。

表 7. PasswordReEnterFormView フォームの属性

ECConstants.EC_PASSWORD_REREQUEST_PASSWORD1	最初のパスワード。
ECConstants.EC_PASSWORD_REREQUEST_PASSWORD2	2 番目のパスワード。

サイト間スクリプト保護

サイト間スクリプトのセキュリティー・フィーチャーを使用するには、ストア用の ProhibitedAttrsErrorView、ProhibitedCharacterErrorView、および ProhibCharEncodingErrorView ビューを定義する必要があります。

ProhibitedAttrsErrorView
このビューがユーザーに対して示されるのは、禁止属性が要求内で使われているためにその要求を処理できないときです。

ProhibitedCharacterErrorView
このビューがユーザーに対して示されるのは、禁止文字が要求内で使われているためにその要求を処理できないときです。

ProhibCharEncodingErrorView
これは、上記の ProhibitedCharacterErrorView と同じです。

ログイン・タイムアウトの使用可能化

注: ログイン・タイムアウトのセキュリティー・フィーチャーをストアで使用するには、 44 ページの『ログイン・タイムアウト』に説明されているとおりに、ストア用の LoginTimeoutErrorView と ReLogonFormView ビューを定義する必要があります。

ログイン・タイムアウト・フィーチャーを使用可能または使用不可にするには、構成マネージャーの「ログイン・タイムアウト」ノードを使用します。このフィーチャーを使用可能にすると、一定期間を超えて非アクティブになっている WebSphere Commerce ユーザーは、システムからログオフされ、ログオンし直すように要求されます。その後ユーザーが正常にログオンすると、 WebSphere Commerce は、そのユーザーが出していた元の要求を実行します。ユーザーのログオンが失敗した場合は、元の要求は廃棄され、そのユーザーはシステムからログオフされたままになります。

WebSphere Commerce ツール (管理コンソール、WebSphere Commerce Accelerator、ストア・サービスなど) の場合、ログイン・タイムアウト・フィーチャーではユーザー向けの再ログイン・ページは表示されないことに注意してください。そのような場合、ブラウザー・ウィンドウはクローズされ、ツールに再ログオンするかどうかはユーザーにゆだねられます。よって、ツールの場合は、ユーザーが送信していた元の要求は処理されません。

この機能を使用可能にするには、以下のようにします。

1. 構成マネージャーを立ち上げて、次のようにして該当するインスタンスの「ログイン・タイムアウト」ノードまでたどっていきます。**「WebSphere Commerce」** > *host_name* > **「Instance List (インスタンス・リスト)」** > *instance_name* > **「インスタンス・プロパティー」** > **「ログイン・タイムアウト」** となります。
2. ログイン・タイムアウト・フィーチャーをアクティブにするには、**「使用可能」** チェック・ボックスをクリックします。
3. 「値」フィールドに、ログイン・タイムアウト値を秒単位で入力します。
4. 変更内容を構成マネージャーに適用するには、**「適用」** をクリックします。
5. インスタンスの構成が正常に更新されると、更新が正常に行われたことを示すメッセージが表示されます。
6. WebSphere Application Server 管理コンソールから WebSphere Commerce Server インスタンスを停止してから再始動します。

ログイン・タイムアウトの値は *instance*.xml ファイルにミリ秒数で保管されるのに対して、構成マネージャーには秒数で入ることに注意してください。

パスワード無効化の活動化

注: パスワード無効化のセキュリティー・フィーチャーを使用するには、 45 ページの『パスワードの無効化』に説明されているとおりに、ストア用の ChangePassword ビューを定義する必要があります。

パスワード無効化フィーチャーを使用可能または使用不可にするには、構成マネージャーの「パスワード無効化」ノードを使用します。パスワード無効化を使用可能にした場合に WebSphere Commerce ユーザーのパスワードの有効期限が切れると、そのユーザーはパスワードの変更を要求されます。その場合、ユーザーは、パスワードの変更が必要となるページにリダイレクトされます。ユーザーは、パスワードの変更を完了するまで、そのサイトのどのセキュア・ページにもアクセスすることができません。この機能を使用可能にするには、以下のようにします。

1. 構成マネージャーを立ち上げて、次のようにして該当するインスタンスの「パスワード無効化」ノードまでたどっていきます。**「WebSphere Commerce」>** *host_name* **>「Instance List (インスタンス・リスト)」>** *instance_name* **>「インスタンス・プロパティー」>「パスワード無効化」**となります。
2. パスワード無効化機能をアクティブにするには、**「使用可能」**チェック・ボックスをクリックします。
3. 変更内容を構成マネージャーに適用するには、**「適用」**をクリックします。
4. インスタンスの構成が正常に更新されると、更新が正常に行われたことを示すメッセージが表示されます。
5. WebSphere Application Server 管理コンソールから WebSphere Commerce Server インスタンスを停止してから再始動します。

パスワード保護コマンドの使用可能化

注: パスワード保護コマンドのセキュリティー・フィーチャーを使用するには、 46 ページの『パスワード保護コマンド』に説明されているとおりに、ストア用の PasswordReEnterErrorView および PasswordReEnterFormView ビューを定義する必要があります。

「パスワード保護されたコマンド」フィーチャーを使用可能または使用不可にするには、「構成マネージャー」の「パスワード保護されたコマンド」ノードを使用します。このフィーチャーを使用可能にすると、WebSphere Commerce は、WebSphere Commerce にログオンした登録済みユーザーに、まずパスワードを入力してから、指定した WebSphere Commerce コマンドの実行要求を続行するよう求めます。

注意: パスワード保護コマンドを構成する場合、コマンド選択リストに示されているコマンドの一部は、一般ユーザーまたはゲスト・ユーザーが実行できるコマンドであることに注意してください。そのようなコマンドを、保護されたパスワードとして構成すると、一般ユーザーおよびゲスト・ユーザーはそのコマンドを実行できなくなります。したがって、コマンドを構成して保護されたパスワードにする場合は注意を払う必要があります。

この機能を使用可能にするには、以下のようにします。

1. 構成マネージャーを立ち上げて、次のようにして該当するインスタンスの「パスワード保護されたコマンド」ノードまでたどっていきます。**「WebSphere Commerce」>** *host_name* **>「Instance List (インスタンス・リスト)」>** *instance_name* **>「インスタンス・プロパティー」>「パスワード保護されたコマンド」**となります。
2. 「一般」タブで、以下のようにします。
   a. 「パスワード保護されたコマンド」フィーチャーをアクティブにするには、**「使用可能」**をクリックします。
   b. 「再試行」フィールドに再試行の回数を入力します。 (デフォルトの再試行回数は 3 です。)
3. 「拡張」タブで、以下のようにします。
   a. 保護したい WebSphere Commerce コマンドを「Password Protected Command List (パスワード保護されたコマンドのリスト)」ウィンドウのリストから選択して、**「追加」**をクリックします。選択したコマンドが「Current Password Protected List (現行のパスワード保護されたコマンドのリスト)」ウィンドウにリストされます。
   b. いずれかの WebSphere Commerce コマンドのパスワード保護を使用不可にしたい場合は、「Current Password Protected Command list (現行のパスワード保護されたコマンドのリスト)」ウィンドウにあるコマンドを選択して、**「除去」**をクリックします。
4. 変更内容を構成マネージャーに適用するには、**「適用」**をクリックします。
5. インスタンスの構成が正常に更新されると、更新が正常に行われたことを示すメッセージが表示されます。
6. WebSphere Application Server 管理コンソールから WebSphere Commerce Server インスタンスを停止してから再始動します。

注: WebSphere Commerce では、使用可能コマンドのリストの `URLREG` テーブルで **認証済み** と指定されているコマンドか、または `https` フラグを使って設定されたコマンドのみが表示されます。

暗号化データの更新

構成マネージャーのデータベース・ノードからデータベース更新ツールを使って、すべての暗号化データ (パスワードやクレジット・カード番号などの) や、WebSphere Commerce データベース内にある特定のインスタンスのマーチャント・キーを更新します。このツールを使用するには、次のようにします。

1. 構成マネージャーを立ち上げて、次のようにして個々のデータベース・エントリーまでたどっていきます。**「WebSphere Commerce」>** *host_name* **>「Instance List (インスタンス・リスト)」 >** *instance_name* **>「インスタンス・プロパティー」>「データベース」>** *database_name* を選びます。
2. *database_name* を右マウス・ボタンでクリックし、**「データベース更新ツールの実行」**を選択します。
   - **「このインスタンスの全データベースの更新」**を選択し、すべてのデータベース用に選択したインスタンスの暗号化データをマイグレーションします。

400 iSeries は単一データベース構成をサポートするので、このオプションは iSeries には該当しません。

- **「選択したデータベースの更新」**を選択してから、ドロップダウン・リストでデータベースを選択して個々のデータベースごとに暗号化データをマイグレーションします (デフォルト)。

3. 実行したいアクションを「アクション・アイテム」ボックスで選択し、次のような必要な情報を「パラメーター」フィールドに入力します。

アクション	パラメーター	必要なアクション
マーチャント・キーの変更	古いマーチャント・キー	現在の WebSphere Commerce インスタンスの作成時に使った既存のマーチャント・キーを入力します。
	新しいマーチャント・キー	新しいマーチャント・キーを入力します。これは、構成マネージャーが現在暗号化されているデータを再暗号化するための 16 桁の 16 進数です。マーチャント・キーには 1 つ以上の英数字 (a ～ f) と 1 つ以上の数字 (0 ～ 9) がなければなりません。英数字は小文字で入力しなければならず、 1 行に 5 回以上同じ文字を入力することはできません。

4. **「OK」**をクリックして、選択した WebSphere Commerce データベースだけ、またはすべての WebSphere Commerce データベース用にデータベース更新ツールを実行します。
5. インスタンスの構成が正常に更新されると、更新が正常に行われたことを示すメッセージが表示されます。
6. WebSphere Application Server 管理コンソールから WebSphere Commerce Server インスタンスを停止してから再始動します。

サイト間スクリプト保護の使用可能化

注: サイト間スクリプトのセキュリティー・フィーチャーをストアで使用するには、 46 ページの『サイト間スクリプト保護』に説明されているとおりに、ストア用に `ProhibitedAttrsErrorView`、 `ProhibitedCharacterErrorView`、および `ProhibCharEncodingErrorView` ビューを定義する必要があります。

インスタンスのサイト間スクリプト保護機能を使用可能または使用不可にするには、「構成マネージャー」の「サイト間スクリプト保護」ノードを使用します。サイト間スクリプト保護を使用可能にすると、許可不能と指定されている属性またはストリングを使っているユーザー要求はすべて拒否されます。構成マネージャーのこのノードで、許可しない属性とストリングを指定することができます。また、サイト間スクリプト保護でコマンドを除外することもできます。それには、該当するコマンドに指定される属性の値の中で禁止ストリングを使用できるようにします。サイト間スクリプト保護は、デフォルトでは使用不可になっています。

警告: サイト間スクリプト保護フィーチャーは、構成に基づいてコマンドの実行を制限するという点で制約的なフィーチャーです。このフィーチャーは、どの属性またはストリングが禁止と定義されているかをチェックしないので、その構成時に

は、禁止属性がコマンドで使われる属性でないことを確認してください。また、禁止ストリングが、いつもコマンドに渡される値でないことも確認してください。このフィーチャーを構成するときは、特別な注意が必要です。

この機能を使用可能にするには、以下のようにします。

1. 構成マネージャーを立ち上げて、次のようにして該当するインスタンスの「サイト間スクリプト保護」ノードまでたどっていきます。**「WebSphere Commerce」** > *host_name* > **「Instance List (インスタンス・リスト)」** > *instance_name* > **「インスタンス・プロパティー」** > **「サイト間スクリプト保護」** を選びます。
2. サイト間スクリプト保護フィーチャーをアクティブにするには、次のように「一般」タブを使用します。
   a. **「使用可能」** をクリックします。
   b. WebSphere Commerce コマンドでの使用を許可しない属性を追加するには、「禁止属性」テーブルをマウスの右ボタンでクリックして、**「行の追加」** を選択します。使用を禁止したい属性を入力します。 1 行に 1 つの属性しか指定できません。
   c. 「禁止属性」テーブルから属性を除去するには、テーブル内でその属性が示されている行を強調表示してから、マウスの右ボタンでクリックして **「行の削除」** を選択します。
   d. WebSphere Commerce コマンドでの使用を許可しないストリングを追加するには、「禁止文字」テーブルをマウスの右ボタンでクリックしてから、**「行の追加」** を選択します。使用を禁止したいストリングを追加します。 1 行に 1 つのストリングしか指定できません。
   e. 「禁止属性」テーブルから文字を除去するには、テーブル内でその文字が示されている行を強調表示してから、マウスの右ボタンでクリックして **「行の削除」** を選択します。

   注: 以下のストリングは、デフォルトで「禁止文字」フィールドに指定されています。これらのストリングは、次のように、サイト・スクリプト記述に対して危害を加えようとする攻撃でのタグのスクリプト記述として最も多く使われます。
   - `<SCRIPT`
   - `<SCRIPT`
   - `<%` および `<%`
3. また、「拡張」タブを使って、サイト間スクリプト保護でコマンドを除外することもできます。それには、次のようにして、該当するコマンドに指定される属性の値の中で禁止ストリングを使用できるようにします。
   a. 「コマンド・リスト」ボックスからコマンドを選択します。
   b. 属性をコンマで区切ったリストを入力します。それらについては、「例外属性のリスト」ウィンドウで禁止文字が許可されます。**「追加」** をクリックします。
   c. コマンドをその属性とともに除去するには、「例外コマンドのリスト」ウィンドウからそのコマンドを選択して、**「除去」** をクリックします。

   属性を選択して **「除去」** をクリックしても、コマンドから特定の属性を除去することができます。

4. 変更内容を構成マネージャーに適用するには、**「適用」**をクリックします。
5. インスタンスの構成が正常に更新されると、更新が正常に行われたことを示すメッセージが表示されます。
6. WebSphere Application Server 管理コンソールから WebSphere Commerce Server インスタンスを停止してから再始動します。

注:

1. サイト間スクリプト保護からコマンドを除外すると、指定した属性の値は、HTML エンコード方式の記号でエンコードされます。たとえば、`cmd1?user=<Thomas>` というコマンドは、`ascmd1?user=&#60;Thomas&#62;` とエンコードされます。
2. 「禁止文字」フィールドにストリングを指定する場合、以下の点に注意してください。
   - 文字をある特定の並びにすると、 URL エンコード標準に準じてストリングが 1 つの文字に変換される可能性があります。たとえば、ストリング `<%bb` はストリング `<X` に変換されます。つまり *X* は、 16 進数 'bb' (10 進数の 187) という 16 進数で表される単一文字です。この場合、ストリング `<%bb` は、URL で渡されると、サイト間スクリプト保護ではキャッチされません。
   - 特定の並びになった文字の場合に、 URL エンコード標準に準じていないと、ストリングの変換が失敗する可能性があります。たとえば、ストリング `<%gg` は変換が失敗する原因になるかもしれません。16 進の 'gg' は、有効な 16 進値表現ではないからです。この場合、ストリング `<%gg` は例外の原因になるので、サイト間スクリプト保護が使用可能になっていてもいなくても、このストリングを使用する URL 要求に対しては応答が行われません。

例: 以下の例について考えてみます。

- 禁止ストリング: `<SCRIPT`, `<%`

  禁止属性: `mycomment`, `description`

コマンド	状況
cmd1?description=Available...	拒否
cmd2?userid=Thomas...	受諾
cmd3?mycomment=<SCRIPT>...	拒否
cmd4?password=<%...%>...	拒否

- `cmd1` コマンドの属性 `text` 内で禁止ストリング (`<SCRIPT`, `<%`) の使用を許可し、たとえば属性 `txt` などの他の属性の場合は許可しないようにするには、 `cmd1` を除外し、規定の属性として `text` を指定します。

コマンド	状況
cmd1?text=<SCRIPT>...	受諾
cmd1?text=<%...%>...	受諾
cmd1?txt=<SCRIPT>...	拒否
cmd1?txt=<%..%>...	拒否

アクセス・ロギングの使用可能化

アクセス・ロギング・フィーチャーを使用可能にすると、 WebSphere Commerce サーバーへの着信要求がすべて記録されるか、あるいはアクセス違反を起こした要求だけが記録されます。アクセス違反の例としては、認証失敗、コマンド実行に不十分な権限、あるいはサイトのパスワード規則に違反するパスワードのリセットがあります。アクセス・ロギングを使用可能にすると、WebSphere Commerce 管理者は WebSphere Commerce システムに対するセキュリティー上の脅威を速やかに確認できます。

認証障害や許可障害のイベントが発生すると、次のような情報がアクセス・ログ・ファイル・データベース ACCLOGMAIN および ACCLOGSUB にログ記録されます。

- クライアントのホスト名
- コマンドを実行するスレッドの ID
- クライアントのユーザー ID
- イベントが発生した時刻
- 実行されたコマンド
- コマンドの実行対象であったストア
- オペレーションが実行されたリソース
- アクセス・コントロール・チェックの結果

アクセス・ロギングを使用可能にするには、次のようにします。

1. 構成マネージャーを立ち上げます。
2. **「ホスト名」**>**「インスタンス」** > **「Instance_List」**を選択してから、**「コンポーネント」**フォルダーをオープンします。
3. **AccessLoggingEventListener** を選択します。
4. 「一般」パネルで**「コンポーネント使用可能」**チェック・ボックスをアクティブにします。
5. 「拡張」パネルを選択し、**「開始」**を使用可能にします。
6. **「適用」**をクリックします。
7. 構成マネージャーを終了します。
8. WebSphere Application Server を再始動します。

ログ・ファイルのサイズを変更したり、すべての要求をログ記録するかどうかを指定したりするには、次のようにして、 WebSphere Commerce インスタンス・サブディレクトリーに置かれている WebSphere Commerce インスタンスの *instance*.xml ファイルを手動で編集する必要があります。

1. インスタンスの *instance*.xml ファイルをエディターでオープンします。
2. <LogSystem>/<activitylog> ノードに置かれている次のようなノードを見つけ出します。

   ```
   <accessLogging cacheSize="aa" logAllRequests="bbbbb" />
   ```

   ここで、

- *aa* は、データベースへのエントリーの書き込みの前にメモリーにログ記録されるエントリーの最大数を指定する整数値です。一般的に、この数値が大きいほど、アクセス・ロギングに関するパフォーマンスは向上します。デフォルト値は 32 です。
- *bbbbb* は `true` または `false` です。 `true` の値は、すべての着信要求をログ記録することを意味します。 `false` の値は、アクセス違反のみをログ記録することを意味します。過多または不要なログ記録を行わないようにするには、`false` をお勧めします。 `true` を使用するのは、サイトにおいて認証上の問題やセキュリティー違反の恐れがある場合のみです。デフォルト値は `false` です。

3. 更新が完了したら、WebSphere Commerce インスタンスの *instance*.xml ファイルを保管します。
4. WebSphere Application Server を再始動します。

以下に示す例ではアクセス・ロギングを使って、データベース・テーブルへのエントリーのログ記録の前に、 3 つのエントリーをメモリー内に保存します。さらに、すべての着信要求を WebSphere Commerce サーバーにログ記録します。

```
<accessLogging cacheSize="3" logAllRequests="true" />
```

アカウント・ポリシーのセットアップ

WebSphere Commerce 管理コンソールの「アカウント・ポリシー」ページで、アカウント・ポリシーをセットアップすることができます。このページには、既存のアカウント・ポリシーがすべてリストされます。その中には、デフォルトで WebSphere Commerce に付属している、事前定義されたあらゆるアカウント・ポリシーも含まれます。アカウント・ポリシーは、パスワード・ポリシーやアカウント・ロックアウト・ポリシーなどのアカウントに関連するポリシーを定義します。このページでは以下を行うことができます。

- **「新規」**をクリックすれば、新しいアカウント・ポリシーを作成することができます。
- リストで既存のアカウント・ポリシーを選択してから**「変更」**をクリックすれば、そのポリシーを変更することができます。
- リストで既存のアカウント・ポリシーを選択してから**「削除」**をクリックすれば、そのポリシーを削除することができます。

新規アカウント・ポリシーを作成するには、次のようにします。

1. 管理コンソールをオープンします。
2. 管理コンソールの「セキュリティー」ドロップダウン・メニューで**「アカウント・ポリシー」**をクリックします。
3. 「アカウント・ポリシー」ページで**「新規」**をクリックして、新しいアカウント・ポリシーを作成します。
4. アカウント・ポリシーの名前を「名前」フィールドに入力します (たとえば、`my_account_policy` など)。
5. 「パスワード・ポリシー」メニューで、事前に存在するパスワード・ポリシーを選択します。

6. 「アカウント・ロックアウト・ポリシー」メニューで、事前に存在するアカウント・ロックアウト・ポリシーを選択します。
7. **「OK」**をクリックします。

アカウント・ポリシーを作成したら、ユーザーにそのポリシーを割り当てることができます。アカウント・ポリシーが使用中の (つまり、ユーザーがアカウント・ポリシーを割り当てられている) 場合は、そのポリシーを削除することはできません。

WebSphere Commerce オンライン・ヘルプの「Default Authentication Policies」も参照してください。

パスワード・ポリシーのセットアップ

WebSphere Commerce 管理コンソールの「パスワード・ポリシー」ページでは、ユーザーのパスワード選択を制御して、サイトのセキュリティー・ポリシーが順守されるようにユーザーのパスワードの特性を定義することができます。このページには、既存のパスワード・ポリシーがすべてリストされます。その中には、デフォルトで WebSphere Commerce に付属している、事前定義されたあらゆるパスワード・ポリシーも含まれます。

パスワード・ポリシーは、パスワードが守らなければならない属性を定義します。パスワード・ポリシーで、以下の条件を決定します。

- ユーザー ID とパスワードが同じでよいか
- 連続する最大文字数
- 文字の最大インスタンス
- パスワードの最長存続時間
- 英字の最小文字数
- 数字の最小文字数
- パスワードの最低限の長さ
- ユーザーの以前のパスワードを再利用できるか
- **「新規」**をクリックすれば、新しいパスワード・ポリシーを作成することができます。
- リストで既存のパスワード・ポリシーを選択してから**「変更」**をクリックすれば、そのポリシーを変更することができます。
- リストで既存のパスワード・ポリシーを選択してから**「削除」**をクリックすれば、そのポリシーを削除することができます。

新規のパスワード・ポリシーを作成するには、次のようにします。

1. 管理コンソールをオープンします。
2. 管理コンソールの「セキュリティー」ドロップダウン・メニューで**「パスワード・ポリシー」**をクリックします。
3. 「パスワード・ポリシー」ページで**「新規」**をクリックして、新しいアカウント・ポリシーを作成します。
4. パスワード・ポリシーの名前を「名前」フィールドに入力します (たとえば、`my_password_policy` など)。

5. 必要があれば以下を更新し、ショッパー用のデフォルト値を任意の値に変更します。
   - **Can the userID and password match? (ユーザー ID とパスワードは一致していてもかまいませんか？)** ユーザー ID とパスワードが同じでもよいかどうかを定義します。リストで「はい」または「いいえ」を選択します。
   - **最大連続文字タイプ**。パスワード内での連続文字の最大出現回数を定義します。連続文字の最小値は 2 回です。たとえば 2 の値の場合、ユーザーは aaabc のような値を入力することはできません。
   - **文字の最大インスタンス**。パスワード内に同一文字が出現してもかまわない最大回数を定義します。最小値は 1 つの文字インスタンスです。たとえば 2 の値の場合、ユーザーは abcaabc のような値を入力することはできません。
   - **パスワードの最大存続期間**。パスワードが存続できる最大期間を日数で定義します。最小値は 1 日です。その期間を過ぎると、ユーザーはパスワードを変更するようプロンプトで指示されます。
   - **英字の最低文字数**。パスワード内で使用しなければならない英字の最小数を定義します。最小値は 0 個の英字です。
   - **数字の最低文字数**。パスワード内で使用しなければならない数字の最小数を定義します。最小値は 0 個の数字です。
   - **パスワードの最小長**。パスワードの最短長を文字数で定義します。最小値は 1 文字です。
   - **Can the password be reused? (パスワードは再使用できますか ?)** ユーザーの旧パスワードを再利用できるかどうかを定義します。リストで「はい」または「いいえ」を選択します。
6. **「OK」**をクリックします。

注:

1. パスワード・ポリシーが使用中の (つまり、ユーザーがそのパスワード・ポリシーを割り当てられている) 場合は、そのポリシーを削除することはできません。
2. パスワード・ポリシーが有効化されるのは、ユーザーが WebSphere Commerce データベースに対して認証されている場合のみです。

WebSphere Commerce オンライン・ヘルプの「Default Authentication Policies」も参照してください。

アカウント・ロックアウト・ポリシーのセットアップ

WebSphere Commerce 管理コンソールの「アカウント・ロックアウト・ポリシー」ページで、WebSphere Commerce 内のさまざまなユーザー役割用のアカウント・ロックアウト・ポリシーをセットアップすることができます。このページには、既存のアカウント・ロックアウト・ポリシーがすべてリストされます。その中には、デフォルトで WebSphere Commerce に付属している、事前定義されたあらゆるアカウント・ポリシーも含まれます。アカウント・ロックアウト・ポリシーは、ユーザー・アカウントに対して不正アクションがとられた場合にそのアカウントを使用禁止にすることで、そのようなアクションによってアカウントが被害を受ける機会を減らします。

アカウント・ロックアウト・ポリシーは次のようなアイテムを統制します。

- アカウント・ロックアウトのしきい値。無効なログオンの試行回数がこの値に達すると、アカウントが使用不可になります。
- ログインが続けて失敗した場合の遅延。これは、ユーザーがログインに 2 回失敗した場合にその後ログインできなくなる期間を指します。ログインの失敗が続くと、この遅延はそのつど構成済みの時間遅延値 (たとえば 10 秒) ずつ増加されます。

アカウント・ロックアウト・ポリシーを設定するには、次のようにします。

1. 管理コンソールをオープンします。
2. 管理コンソールの「セキュリティー」ドロップダウン・メニューで**「アカウント・ロックアウト・ポリシー」**をクリックします。
3. 既存のすべてのアカウント・ロックアウト・ポリシーが「アカウント・ロックアウト・ポリシー」ページに示されます。 このページでは以下を行うことができます。
   - **「新規」**をクリックすれば、新しいポリシーを作成することができます。
   - リストでポリシーを選択してから**「変更」**をクリックすれば、既存のポリシーを変更することができます。
   - リストでポリシーを選択してから**「削除」**をクリックすれば、既存のポリシーを削除することができます。

新規のアカウント・ロックアウト・ポリシーの場合、「アカウント・ロックアウト・ポリシー」ページで次のようにします。

1. アカウント・ロックアウト・ポリシーの名前を「名前」フィールドに入力します (たとえば、`my_policy` など)。
2. 「アカウント・ロックアウトしきい値」フィールドにそのしきい値を入力します。たとえば 6 (6 回の試行) と入力します。
3. ログインが続けて失敗した場合の遅延を秒数で「Wait time (待ち時間)」フィールドに入力します。たとえば 10 (10 秒の場合) と入力します。
4. **「OK」**をクリックします。

注:

1. アカウント・ロック・ポリシーが使用中の (つまり、ユーザーがそのアカウント・ロック・ポリシーを割り当てられている) 場合は、そのポリシーを削除することはできません。
2. アカウント・ロック・ポリシーが実効化されるのは、ユーザーが WebSphere Commerce データベースに対して認証されている場合のみです。

セキュリティー検査の立ち上げ

400 このフィーチャーは、 WebSphere Commerce for iSeries では使用できません。

「セキュリティー検査の立ち上げ」ページを使って、機密漏れの可能性があると思われる一時 WebSphere Commerce ファイルの検査と削除を行うためのセキュリティー・プログラムを手動で立ち上げることができます。通常、セキュリティー検査プログラムは定期的なジョブとして実行され、デフォルトでは月に一度実行するように設定されています。

セキュリティー検査プログラムを起動するには、次のようにします。

1. 管理コンソールをオープンします。
2. 管理コンソールの「セキュリティー」ドロップダウン・メニューで**「セキュリティー・チェッカー」**をクリックします。
3. 「セキュリティー検査の立ち上げ」ページで、**「立ち上げ」**をクリックします。

プログラムによってとられたすべてのアクションを含め、セキュリティー検査の結果が「セキュリティー検査」ウィンドウと、次のようなログ・サブディレクトリー内の `sec_check.log` ファイルに書き込まれます。

NT *drive*:¥WebSphere¥Commerce¥instances¥*instance_name*¥log

2000 *drive*:¥Program Files¥WebSphere¥Commerce¥instances¥*instance_name*¥log

AIX /usr/lpp/Commerce/instances/*instance_name*/log

Solaris /opt/WebSphere/Commerce/instances/*instance_name*/log

Linux /opt/WebSphere/Commerce/instances/*instance_name*/log

Windows Windows 以外のプラットフォームでは、無許可のユーザーが機密ファイルにアクセスできないようにするため、ファイル許可は WebSphere Commerce で自動的に設定されます。 Windows プラットフォームでは、次のようにして、許可を手動で設定する必要があります。この手順を行えば、機密ファイルに対して管理者グループのみが**読み取り/書き込み/実行**の権限をもつようにすることができます。

1. Windows のエクスプローラで *drive*:¥WebSphere フォルダーを右マウス・ボタンでクリックします。
2. **「プロパティ」**をクリックし、次に**「セキュリティ」**をクリックします。デフォルトでは、「Everyone」グループが、このフォルダーに対する**すべての**許可を受けています。
3. **「追加」**をクリックします。
4. ウィンドウが表示されます (「Select users, computers... (ユーザー、コンピューターの選択)」)。このウィンドウで、**「Administrators」**グループを選択します。

   注: この場合、これは若干不明確になるかもしれません。`Administrator` をユーザーと見なすことができますが、 Administrator ユーザーではなく、Administrator グループを追加する必要があるからです。

   「追加」をクリックしてから、**「OK」**をクリックします。
5. 「セキュリティ」タブに Administrators グループが追加されました。「Everyone」を除去する必要があります。**「Everyone (全員)」**を選択してから、「継承可能なアクセス許可を...」と書かれているボックスのチェックを取り除きます。
6. 表示される「セキュリティ」ウィンドウの**「削除」**をクリックします。

構成マネージャーの PDI 暗号化フィールド

WebSphere Commerce インスタンスを構成するときは、「PDI Encrypt (PDI 暗号化)」チェック・ボックスを選択するようお勧めします。「PDI Encrypt (PDI 暗号化)」フィールドを使用可能にすると、 ORDPAYINFO と ORDPAYMTHD のテーブルに指定された情報が暗号化されるはずです。このチェック・ボックスを選択すると、支払情報が WebSphere Commerce データベースに暗号化された形式で保管されます。

第 5 章 WebSphere Application Server のセキュリティーの使用可能化

この章では、WebSphere Application Server のセキュリティーを使用可能にする方法について説明します。 WebSphere Application Server のセキュリティーを使用可能にすると、部外者からのリモート呼び出しによってすべての Enterprise JavaBean コンポーネントが開示されたりしないようにすることができます。

注: Windows AIX Solaris Linux WebSphere Application Server セキュリティーを使用可能にする場合には、ご使用のマシンで以下の要件を満たすよう強くお勧めします。

- 1 GB 以上のマシン・メモリー
- WebSphere Commerce アプリケーション用に、384 MB 以上のヒープ・サイズ

はじめに

セキュリティーを使用可能にする前に、セキュリティーを使用可能にする WebSphere Application Server がユーザー ID の妥当性を検査する方法を知る必要があります。 WebSphere Application Server は LDAP またはオペレーティング・システムのユーザー・レジストリーを WebSphere Application Server ユーザー・レジストリーとして使用できます。

Windows AIX Solaris Linux WebSphere Application Server のセキュリティーの実行に必要な最新の eFix の詳細は、以下の WebSphere Commerce の Web サイトに用意されている最新の WebSphere Commerce 5.4 README 資料を参照してください。

Business

`http://www.ibm.com/software/webservers/commerce/wc_be/lit-tech-general.html`

Professional

`http://www.ibm.com/software/webservers/commerce/wc_pe/lit-tech-general.html`

LDAP ユーザー・レジストリーを使用するセキュリティーの使用可能化

Windows LDAP を WebSphere Application Server ユーザー・レジストリーとして使用しているときに WebSphere Application Server セキュリティーを使用可能にするには、管理権限をもったユーザーとしてシステムにログインし、次のようなステップを行います。

400 LDAP を WebSphere Application Server ユーザー・レジストリーとして使用しているときに WebSphere Application Server セキュリティーを使用可能にするには、システムにログインして次のようなステップを行います。

AIX Solaris Linux LDAP を WebSphere Application Server ユーザー・レジストリーとして使用しているときに WebSphere Application Server セキュリティーを使用可能にするには、 `wasuser` としてシステムにログインし、次のようなステップを行います。

1. WebSphere Application Server 管理サーバーを開始して、WebSphere Application Server 管理コンソールをオープンします。
2. コンソールで、以下のようにグローバル・セキュリティー設定値を変更します。
   a. 「コンソール」メニューから、**「Security Center (セキュリティー・センター)」**を選択します。
   b. 「一般」タブで、**「Enable Security (セキュリティーを使用可能にする)」**を選択します。
   c. **「Authentication (認証)」**タブで、「Lightweight Third Party Authentication (LTPA)」を選択します。 LTPA 設定を入力し、この機能を使用しない場合は**「Enable Single Sign On (単一サインオンを使用可能にする)」**チェック・ボックスのチェックを外します。使用しているディレクトリー・サーバーのタイプに応じて、以下のように**「LDAP Settings (LDAP 設定)」**タブに値を入力します。

Windows AIX Solaris Linux 400

表 8. SecureWay ユーザー

フィールド名	定義	サンプル値	備考
セキュリティー・サーバー ID	ユーザー ID	*user_ID*	• これは LDAP 管理者にすることはできません。 • cn=xxx として指定されているユーザーは使用しないでください。 • このユーザーのオブジェクト・クラスが、「LDAP Advanced Properties (LDAP 拡張プロパティー)」ウィンドウの「User Filter (ユーザー・フィルター)」フィールドに指定されたオブジェクト・クラスと互換性があることを確認します。
セキュリティー・サーバー・パスワード	ユーザー・パスワード	*password*	
ディレクトリー・タイプ	LDAP サーバーのタイプ	SecureWay	
ホスト	LDAP サーバーのホスト名	*hostname.domain.com*	

表 8. *SecureWay* ユーザー *(続き)*

フィールド名	定義	サンプル値	備考
ポート	LDAP サーバーが使用しているポート		このフィールドは不要です。
基本識別名	検索に使用される識別名	o=ibm,c=us	
バインド識別名	検索時にディレクトリーにバインドするための識別名		このフィールドは不要です。
バインド・パスワード	バインド識別名のパスワード		このフィールドは不要です。

Windows

表 9. *Netscape* ユーザー

フィールド名	定義	サンプル値	備考
セキュリティー・サーバー ID	ユーザー ID	*user_ID*	• これは LDAP 管理者にすることはできません。 • cn=xxx として指定されているユーザーは使用しないでください。 • このユーザーのオブジェクト・クラスが、「LDAP Advanced Properties (LDAP 拡張プロパティー)」ウィンドウの「User Filter (ユーザー・フィルター)」フィールドに指定されたオブジェクト・クラスと互換性があることを確認します。
セキュリティー・サーバー・パスワード	ユーザー・パスワード	*password*	
ディレクトリー・タイプ	LDAP サーバーのタイプ	Netscape	
ホスト	LDAP サーバーのホスト名	*hostname.domain.com*	
ポート	LDAP サーバーが使用しているポート		このフィールドは不要です。
基本識別名	検索に使用される識別名	o=ibm	

表 9. *Netscape* ユーザー *(続き)*

フィールド名	定義	サンプル値	備考
バインド識別名	検索時にディレクトリーにバインドするための識別名		このフィールドは不要です。
バインド・パスワード	バインド識別名のパスワード		このフィールドは不要です。

Windows

表 10. *Domino*™ ユーザー

フィールド名	定義	サンプル値	備考
セキュリティー・サーバー ID	ショート・ネーム / ユーザー ID	*user_ID*	このユーザーのオブジェクト・クラスが、「LDAP Advanced Properties (LDAP 拡張プロパティー)」ウィンドウの「User Filter (ユーザー・フィルター)」フィールドに指定されたオブジェクト・クラスと互換性があることを確認します。
セキュリティー・サーバー・パスワード	ユーザー・パスワード	*password*	
ディレクトリー・タイプ	LDAP サーバーのタイプ	Domino 5.0	
ホスト	LDAP サーバーのホスト名	*hostname.domain.com*	
ポート	LDAP サーバーが使用しているポート		このフィールドは不要です。
基本識別名	検索に使用される識別名		このフィールドは不要です。
バインド識別名	検索時にディレクトリーにバインドするための識別名		このフィールドは不要です。
バインド・パスワード	バインド識別名のパスワード		このフィールドは不要です。

Windows

表 11. アクティブ・ディレクトリー・ユーザー

フィールド名	定義	サンプル値	備考
セキュリティー・サーバー ID	sAMAccountName	*user_ID*	• 任意の通常ユーザーのユーザー・ログオン名。 • cn=xxx として指定されているユーザーは使用しないでください。 • このユーザーのオブジェクト・クラスが、「LDAP Advanced Properties (LDAP 拡張プロパティー)」ウィンドウの「User Filter (ユーザー・フィルター)」フィールドに指定されたオブジェクト・クラスと互換性があることを確認します。
セキュリティー・サーバー・パスワード	ユーザー・パスワード	*password*	
ディレクトリー・タイプ	LDAP サーバーのタイプ	アクティブ・ディレクトリー	
ホスト	LDAP サーバーのホスト名	*hostname.domain.com*	
ポート	LDAP サーバーが使用しているポート		このフィールドは不要です。
基本識別名	検索に使用される識別名	CN=users, DC=domain1, DC=domain2, DC=com	
バインド識別名	検索時にディレクトリーにバインドするための識別名	CN=*user_ID*, CN=users, DC=domain1, DC=domain2, DC=com	*user_ID* 値は表示名です。これはユーザー・ログオン名と同じでなくてもかまいません。
バインド・パスワード	バインド識別名のパスワード	*bind_password*	これはセキュリティー・サーバー・パスワードと同じでなければなりません。

d. Windows 400 WebSphere Application Server 管理サーバーを再始動してから、WebSphere Application Server 管理コンソールを再オープンします。

e. 「**Role Mapping (役割マッピング)**」タブで、 WCS appserver を選択し、「**Edit Mappings... (マッピングの編集...)**」ボタンをクリックします。

1) 「WCSSecurity Role (WCSSecurity 役割)」を選択し、「**Select... (選択...)**」ボタンをクリックします。

2) 「Select users/groups (ユーザー / グループの選択)」チェック・ボックスをチェックし、ステップ 2c (62 ページ) で入力したユーザー ID を追加します。

f. 「**終了**」をクリックします。

3. 管理コンソールをクローズして、WebSphere Application Server 管理サーバーを停止してから、再始動します。この後は、WebSphere Application Server 管理コンソールをオープンするとき、セキュリティー・サーバー ID とパスワードの入力を求めるプロンプトが出されます。
4. WebSphere Commerce 構成マネージャーをオープンして、「**Instances (インスタンス)**」>「*instance_name*」>「**インスタンス・プロパティー**」>「**セキュリティー**」を選択し、「**使用可能**」チェック・ボックスをクリックします。ステップ 2c (62 ページ) で入力したユーザー名とパスワードを入力するよう促されます。「**適用**」をクリックして、構成マネージャーを終了します。
5. WebSphere Application Server 管理サーバーを停止してから、再始動します。

オペレーティング・システム・ユーザー・レジストリーを使用したセキュリティーの使用可能化

Windows 400 オペレーティング・システムのユーザー妥当性検査を WebSphere Application Server ユーザー・レジストリーとして使用しているときに WebSphere Application Server セキュリティーを使用可能にするには、管理権限をもったユーザーとしてシステムにログインし、次のようなステップを行います。

AIX Solaris Linux オペレーティング・システムをユーザー・レジストリーとして使用するには、 WebSphere Application Server をルートとして実行している必要があります。 WebSphere Application Server をルートとして実行し、次のようなステップを行います。

1. AIX Solaris Linux ルートとしてログインします。
2. AIX Solaris Linux ルートでログインしてから、 WebSphere Application Server を始動して WebSphere Application Server 管理コンソールを立ち上げます。

```
export DISPLAY=fully_qualififed_host_name:0.0
cd WAS_HOME/bin
./startupServer.sh &
./adminclient.sh remote_WAS_host_name port
```

fully_qualififed_host_name は、 WebSphere Application Server 管理コンソールにアクセスするのに使用するコンピューターの名前、*remote_WAS_host_name* は WebSphere Application Server の完全修飾ホスト名、 port は、WebSphere Application Server へのアクセス時に経由するポート (デフォルトのポートは 2222) です。

3. WebSphere Application Server 管理コンソールで、以下のようにグローバル・セキュリティー設定値を変更します。
   a. 「コンソール」メニューから、**「Security Center (セキュリティー・センター)」**を選択します。
   b. 「一般」タブで、**「Enable Security (セキュリティーを使用可能にする)」**チェック・ボックスを選択します。
4. **「Authentication (認証)」**タブを選択し、**「Local Operating System (ローカル・オペレーティング・システム)」** ラジオ・ボタンを選択します。
5. **「Security Server ID (セキュリティー・サーバー ID)」**フィールドにセキュリティー・サーバー ID を入力します。以下のようにユーザー名を入力します。

フィールド名	サンプル値	備考
ユーザー ID	*user_ID*	Windows ログインしたオペレーティング・システム管理権限のあるユーザー ID。マシンがドメインに属する場合、完全修飾ユーザー ID を使用します (たとえば、DomainXYZ¥user_id)。このアカウントがドメイン・サーバーに属していて、管理者のグループのメンバーであることを確認してください。 AIX Solaris Linux ルートまたはルート権限をもつユーザー ID。 400 iSeries でのユーザー ID は、 *SECOFR 権限をもっていなければなりません。
セキュリティー・サーバー・パスワード	*password*	これはログインの際に使用した、オペレーティング・システム管理権限のあるユーザーのパスワードです。

6. Windows 400 WebSphere Application Server 管理サーバーを再始動してから、WebSphere Application Server 管理コンソールを再オープンします。
7. **「Role Mapping (役割マッピング)」**タブで、 WC エンタープライズ・アプリケーションを選択し、**「Edit Mappings... (マッピングの編集...)」**ボタンをクリックします。
   a. 「WCSecurityRole」を選択し、**「Select... (選択...)」**ボタンをクリックします。
   b. 「Select users/groups (ユーザー / グループの選択)」チェック・ボックスを選択し、ステップ 5 で使用したユーザー ID を「検索」フィールドに入力して、**「検索」**をクリックします。「Available Users/Groups (使用可能なユーザ

ー / グループ)」リストからそのユーザーを選択し、**「追加」**をクリックして「Selected Users/Groups (選択したユーザー / グループ)」リストに追加します。次に、各パネルで**「OK」**をクリックし、セキュリティー・センターを終了します。

8. WebSphere Commerce 構成マネージャーをオープンし、**「Instances List (インスタンス・リスト)」→** *「instance_name」* **→「インスタンス・プロパティー」→「セキュリティー」**を選択し、**「Enable Security (セキュリティーを使用可能にする)」**チェック・ボックスを選択します。認証モードの**「オペレーティング・システム・ユーザー・レジストリー」**を選択し、ステップ 5 (67 ページ) で入力したユーザー名とパスワードを入力します。**「適用」**をクリックして、構成マネージャーを終了します。
9. WebSphere Application Server 管理サーバーを停止してから、再始動します。この後は、WebSphere Application Server 管理コンソールをオープンするとき、セキュリティー・サーバー ID とパスワードの入力を求めるプロンプトが出されます。

WebSphere Commerce EJB セキュリティーの使用禁止

WebSphere Commerce Business Edition を使用して、 EJB セキュリティーを使用不可にすることができます。 WebSphere Commerce EJB セキュリティーを使用不可にするには、以下のようにします。

1. WebSphere Application Server 管理コンソールを始動します。
2. **「コンソール」→「Security Center... (セキュリティー・センター...)」**をクリックし、**「一般」**タブの**「セキュリティー使用可能」**チェック・ボックスを選択解除します。
3. WebSphere Commerce 構成マネージャーをオープンして、**「Instances (インスタンス)」→** *「instance_name」* **→「インスタンス・プロパティー」→「セキュリティー」**を選択し、**「Enable Security (セキュリティーを使用可能にする)」**チェック・ボックスをクリアします。
4. WebSphere Application Server 管理コンソールを終了します。
5. WebSphere Application Server 管理サーバーを停止してから、再始動します。

WebSphere Commerce セキュリティー・デプロイメント・オプション

WebSphere Commerce は、さまざまなセキュリティー・デプロイメント構成をサポートしています。以下の表には、使用できるセキュリティー・デプロイメント・オプションが示されています。

表 12. 単一マシンのセキュリティーのシナリオ

WebSphere Application Server セキュリティーが使用可能。	• オペレーティング・システムを WebSphere Application Server レジストリーとして使用する。 • データベースを WebSphere Commerce レジストリーとして使用する。
	• LDAP を WebSphere Application Server レジストリーとして使用する。 • LDAP を WebSphere Commerce レジストリーとして使用する。
	• LDAP を WebSphere Application Server レジストリーとして使用する。
WebSphere Application Server セキュリティーが使用不可、および WebSphere Commerce サイトがファイアウォールに守られている。	• WebSphere Application Server レジストリーは不要。 • データベースを WebSphere Commerce レジストリーとして使用する。
	• WebSphere Application Server レジストリーは不要。 • LDAP を WebSphere Commerce レジストリーとして使用する。

表 13. 複数マシンのセキュリティーのシナリオ

WebSphere Application Server セキュリティーが使用可能。 LDAP が常にデプロイされている。	• LDAP を WebSphere Application Server レジストリーとして使用する。 • LDAP を WebSphere Commerce レジストリーとして使用する。
	• LDAP を WebSphere Application Server レジストリーとして使用する。 • データベースを WebSphere Commerce レジストリーとして使用する。 • LDAP をセットアップし、 LDAP レジストリー中に 1 つの管理エントリーを組み込む必要がある。

表 13. 複数マシンのセキュリティーのシナリオ (続き)

WebSphere Application Server セキュリティーが使用不可、および WebSphere Commerce サイトがファイアウォールに守られている。	• データベースを WebSphere Commerce レジストリーとして使用する。 • WebSphere Application Server レジストリーは不要。 • 単一サインオンはサポートされない。
	• LDAP を WebSphere Application Server レジストリーとして使用する。 • WebSphere Application Server レジストリーは不要。

注: ファイアウォールの内部で WebSphere Commerce サイトを操作する場合は、WebSphere Application Server セキュリティーを使用不可にすることができます。WebSphere Application Server セキュリティーを使用不可にするのは、ファイアウォールの内部で有害なアプリケーションが稼働していないことが確認されている場合に限る必要があります。

第 6 章 セッション管理

Web ブラウザーと e-commerce サイトは、HTTP を使って通信します。 HTTP はステートレス・プロトコル (つまり、どのコマンドも、前に発行されたコマンドを関知せずに独立して実行されます) であるため、ブラウザー・サイドとサーバー・サイドどうしのセッションを管理する手段が必要です。

WebSphere Commerce は、cookie ベースおよび URL 再書き込みの 2 つのタイプのセッション管理をサポートします。管理者は、cookie ベースのセッション管理だけのサポート、または cookie ベースと URL 再書き込みの両方のセッション管理のサポートを選択することができます。 WebSphere Commerce が cookie ベースのみをサポートする場合、ショッパーのブラウザーは cookie の受け入れ可能になっていなければなりません。 cookie ベースと URL 再書き込みの両方を選択すると、WebSphere Commerce はまず cookie を使ってセッション管理を試みます。ショッパーのブラウザーが cookie を受け入れないように設定されている場合に、 URL 再書き込みが使われます。

cookie ベースのセッション管理

cookie ベースのセッション管理を使用すると、ユーザー情報の入ったメッセージ (cookie) が Web サーバーからブラウザーに送られます。この cookie は、ユーザーが特定のページにアクセスしようとしたときにサーバーに返送されます。サーバーは、cookie の返送によってユーザーを識別することができ、セッション・データベースからそのユーザーのセッションを取り出します。それによって、ユーザーのセッションが保守されます。 cookie ベースのセッションは、ユーザーがブラウザーをログオフまたはクローズすると終了します。 cookie ベースのセッション管理は安全であり、しかもパフォーマンス上の利点があります。 cookie ベースのセッション管理が安全なのは、 SSL を通してのみやりとりされる識別タグを使用するからです。cookie ベースのセッション管理は、パフォーマンスの点から見るとかなり有利です。 WebSphere Commerce のキャッシング機構は cookie ベースのセッションだけをサポートし、 URL 再書き込みをサポートしないからです。 cookie ベースのセッション管理は、ショッパー・セッションの場合にお勧めします。

URL の再書き込みを使用していない場合に、ユーザーがブラウザー上で cookie を使用可能にしているかどうかを確認したければ、構成マネージャーの「セッション管理」ページの**「cookie 受け入れテスト」**にチェックしてください。これで、ショッパーのブラウザーが cookie をサポートしていない場合や、cookie がオフになっている場合に、 WebSphere Commerce サイトをブラウズするには cookie をサポートしているブラウザーが必要であることがショッパーに知らされます。

セキュリティー上の理由から、cookie ベースのセッション管理では次の 2 種類の cookie が使われます。

- 非セキュア・セッション cookie

  セッション・データを管理するのに使われます。セッション ID、ネゴシエーションされた言語、現在のストア、およびショッパーの希望通貨 (cookie の構成時の

もの) が入っています。この cookie は、SSL または非 SSL のどちらの接続でもブラウザーとサーバーとでやり取りすることができます。次の 2 タイプの非セキュア・セッション cookie があります。

– WebSphere Application Server セッションの cookie は、サーブレット HTTP セッション標準をベースとします。 WebSphere Application Server の cookie は、メモリーにか、またはマルチノード・デプロイメントのデータベースに密着しています。詳細は、`http://www.ibm.com/software/webservers/appserv/infocenter.html` に掲載されている『WebSphere Application Server InfoCenter』の中の『session management』を参照してください。

– WebSphere Commerce セッションの cookie は WebSphere Commerce から見て内部的であり、データベースには密着していません。

どのタイプの cookie を使用するかを選択するには、構成マネージャーの「Session Management (セッション管理)」ページの **「Cookie Session Manager (cookie セッション・マネージャー)」**パラメーターで `WCS` または `WAS` を選択します。

- セキュア認証 cookie

  認証データの管理に使用されます。認証 cookie は SSL を通してやりとりされ、セキュリティーの最大化のためにタイム・スタンプを押されます。これは、たとえばユーザーのクレジット・カード番号をたずねる `DoPaymentCmd` といった機密性の高いコマンドが実行されるたびに、ユーザーを認証するのに使用される cookie です。この cookie が盗まれて無許可のユーザーによって使用される危険性は最小化されています。 cookie ベースのセッション管理を使用する場合は、常に認証コード cookie が WebSphere Commerce によって生成されます。

セキュア・ページを閲覧するには、セッション cookie と認証コード cookie の両方が必要です。

以下の場合に cookie エラーが起きると、`CookieErrorView` が呼び出されます。

- ユーザーが同じログオン ID を使って別のロケーションからログインした場合。
- cookie が壊れたかまたは改ざんされた (またはこの両方) 場合。
- cookie の受け入れが「真」に設定されているのに、ユーザーのブラウザーが cookie をサポートしていない場合。

セッション管理での cookie の使用

WebSphere Commerce で cookie を使用するには、次のようにします。

1. 構成マネージャーをオープンします。
2. **「インスタンス」**を選択してから、**「セッション管理」**フォルダーをオープンします。
3. 該当するセッション値を選択します。
   - cookie 受け入れテスト

     顧客のブラウザーが、 cookie のみをサポートしているサイトの cookie を受け入れるかどうか調べるには、このチェック・ボックスを選択します。
   - cookie セッション・マネージャー

WebSphere Commerce または WebSphere Application Server のどちらで cookie を管理したいかを選択します。デフォルトは WebSphere Commerce です。

– WebSphere Application Server セッションの cookie は、サーブレット HTTP セッション標準をベースとします。 WebSphere Application Server の cookie は、メモリーにか、またはマルチノード・デプロイメントのデータベースに密着しています。詳細は、`http://www.ibm.com/software/webservers/appserv/infocenter.html` に掲載されている『WebSphere Application Server InfoCenter』の中の『session management』を参照してください。

– WebSphere Commerce セッションの cookie は WebSphere Commerce から見て内部的であり、データベースには密着していません。

4. **「拡張」**タブをクリックします。該当するセッション値を選択します。
   - Cookie パス

     通常、このフィールドを変更してはなりません。 cookie のパスを指定します。これは cookie の送信先の URL のサブセットです。
   - Cookie 有効期限

     このフィールドを変更してはなりません。デフォルトでは、ブラウザーがクローズされたときに cookie の有効期限が切れます。
   - Cookie ドメイン

     通常、このフィールドを変更してはなりません。ドメインの制限パターンを指定します。 cookie を受け取るサーバーを、ドメインで指定します。デフォルトでは、cookie はその発信元の WebSphere Commerce Server にだけ返送されます。またデフォルトでは cookie は、保管先のホストにのみ返送されます。ドメイン・ネーム・パターンを指定すると、それがオーバーライドされます。そのパターンは、ピリオドで始まっていて、少なくとも 2 つのピリオドが使われていなければなりません。パターンは、最初のピリオドの後は 1 つのエントリーにしか一致しません。たとえば、".ibm.com" は有効であり、`a.ibm.com` と `b.ibm.com` に一致しますが、 `www.a.ibm.com` には一致しません。ドメイン・パターンの詳細は、Netscape の cookie の仕様と RFC 2109 を参照してください。
5. **「適用」**をクリックします。
6. 構成マネージャーをクローズします。
7. WebSphere Application Server 管理コンソールからインスタンスを停止してから再始動します。

URL 再書き込み

URL 再書き込みを使った場合、ブラウザーに戻ってくるか、またはリダイレクトされたすべてのリンクには、セッション ID が付けられます。ユーザーがそのようなリンクをクリックすると、書き換えられたフォームの URL が、クライアント要求の一部としてサーバーに送信されます。サーブレット・エンジンは、URL 内のセッション ID を認識し、そのユーザーの正しいオブジェクトを取得するために保管します。 URL の再書き込みを使用するには、リンクに HTML ファイル (`.html` または `.htm` の拡張子の付いたファイル) は使用できません。 URL の再書き込みを使用するには、JSP ファイルを表示用に使用する必要があります。 URL の再書き込みを使用するセッションは、ショッパーがログオフすると期限が切れます。

注: WebSphere Commerce のキャッシングと URL の再書き込みの操作は両立しません。 URL 再書き込みをオンにする場合、WebSphere Commerce キャッシング・コンポーネントを使用不可にする必要があります。

URL 再書き込みセッション管理の使用

セッションを管理する方法を指定するには、次のようにします。

1. 構成マネージャーをオープンします。
2. **「インスタンス」**を選択してから、**「セッション管理」**フォルダーをオープンします。
3. 該当するセッション値を選択します。

   URL 再書き込み使用可能セッション管理に URL 再書き込みを使用する場合は、このチェック・ボックスを選択します。

   cookie セッション・マネージャー。 WebSphere Application Server を選択します。
4. **「適用」**をクリックします。
5. 構成マネージャーをクローズします。
6. WebSphere Application Server 管理コンソールからインスタンスを停止してから再始動します。

URL 再書き込み用の JSP テンプレートの作成

セッション状態の保守に URL 再書き込みを使用したい場合、 Web アプリケーションのパーツへのリンクをプレーン・テキストの HTML ファイルに組み込まないでください。この制約事項が必要なのは、プレーン・テキストの HTML ファイル内で URL エンコードを使用できないからです。 URL 再書き込みを使って状態を保守するには、セッション中のユーザー要求では、 Java インタープリターが理解できるコードを使用する必要があります。そのようなプレーン・テキストの HTML ファイルや、ユーザーがセッション中にアクセスする可能性のあるサイト部分が Web アプリケーションの中に入っている場合、それを JSP ファイルに変換してください。これによってアプリケーションの作成が影響を受けることになります。 cookie を使ってセッションを保守するのと違って、URL 再書き込みでセッションを保守するには、アプリケーション内の各 JSP テンプレートの中で `<A>` タグの各 `HREF` 属性ごとに URL エンコードを使用する必要があるからです。アプリケーション内の 1 つ以上の JSP テンプレートが `encodeURL(String url)` を呼び出さなかったり、 `RedirectURL(String url)` メソッドをエンコードしたりすると、セッションは消失します。

リンクの作成

URL 再書き込みでは、ブラウザーに戻ってくるか、またはリダイレクトされるすべてのリンクには、セッション ID が付けられている必要があります。たとえば、Web ページ内に次のようなリンクがあるとします。

```
<a href="store/catalog">
```

上記は次のように書きます。

```
<a href="store/catalog;$jsessionid$DA32242SSGE2">
```

ユーザーがこのリンクをクリックすると、書き換えられたフォームの URL が、クライアント要求の一部としてサーバーに送信されます。サーブレット・エンジンは、;$jsessionid$DA32242SSGE2 をセッション ID として認識し、このユーザーの正しい HttpSession オブジェクトを取得するために保管します。

以下の例は、 JSP ファイル内に Java コードを組み入れる方法を示しています。

```
<%
 response.encodeURL ("/store/catalog");
%>
```

ブラウザーに送り返される URL を再書き込みするには、 JSP テンプレート内で encodeURL() メソッドを呼び出してから、その URL を出力ストリームに送信します。たとえば、次のような、URL 再書き込みを使用しない JSP テンプレートがあるとします。

```
out.println("<a href=¥"/store/catalog¥">catalog</a>")"
```

上記を次のように書き換えます。

```
out.println("<a href=¥"");
out.println(response.encodeURL ("/store/catalog"));
out.println("¥">catalog</a>");
```

リダイレクトしようとする URL を再書き込みするには、 encodeRedirctURL() メソッドを呼び出します。たとえば、次のような JSP テンプレートがあるとします。

```
response.sendRedirect (response.encodeRedirectURL ("http://myhost/store/catalog"));
```

encodeURL() と encodeRedirectURL() メソッドは、 HttpServletResponse オブジェクトの一部を成します。どちらの場合も、URL のエンコードの前に URL が再書き込みされるよう構成されているかどうかが、それらの呼び出しによって検査されて確かめられます。そのように構成されていないと、元の URL が返送されます。

フォームの作成: 送信用のフォームを作成するには、フォーム・テンプレートの ACTION タグ上の response.encodeURL("Logon"); を呼び出します。以下は、その例です。

```
String strLoginPost = response.encodeURL("Logon");
<FORM NAME="Logon" METHOD="post" ACTION= <%= strLoginPost %> >
...
</FORM>
```

先頭ページの作成: 通常はホーム・ページである導入ページでは、フレームを使用することはできません。ストア内でフレームを使用したい場合、そのストアへのリンクをもつ非フレーム・ページをストアの導入ページとして機能させることができます。ただし、ストアがフレームを使用する場合に、顧客が先に導入ページを経由しないで、フレームを備えたページにアクセスしようとすると、そのセッションは消失することがあります。また顧客が、**「戻る」**ボタン (フレームにのみ装備) を使って導入ページに戻って、導入ページを最新表示にしようとした場合も、セッションが消失する可能性があります。導入ページを最新表示にすると、新規のセッション ID が発行されるからです。この種のセッションの消失を防止するには、**「戻る」**ボタンの代わりに、導入ページに戻るためのリンクが必要です。

第 3 部 システム管理者のセキュリティー・タスク

第 3 部では、必ずしも WebSphere Commerce サイト管理者であるとは限らないシステム管理者が、サイトで通常実行できるセキュリティー・タスクについて説明します。

第 7 章 パスワードの設定と変更

WebSphere Commerce のコンポーネントの大半は、オペレーティング・システムによって検証されるユーザー ID とパスワードを利用します。そのようなパスワードの変更の詳細は、オペレーティング・システムの資料を参照してください。この章では、オペレーティング・システムを介してユーザー ID とパスワードを検証しない WebSphere Commerce コンポーネント用のパスワードの設定と変更の方法について述べます。

ユーザー ID、パスワード、および Web アドレスの早見表

WebSphere Commerce 環境での管理には、さまざまなユーザー ID が必要です。それらのユーザー ID と、それに必要な権限のリストを、次の表に示します。各 WebSphere Commerce ユーザー ID ごとにデフォルトのパスワードが示されています。

Windows ユーザー ID

Windows ユーザー ID は管理者権限をもっている必要があります。 DB2® を使用する場合、ユーザー ID とパスワードに関して次の規則に従う必要があります。

- 長さが 8 文字を超えてはなりません。
- 使用できる文字は A〜Z、a〜z、0〜9、@、#、$、および _ だけです。
- 下線 (_) で始めることはできません。
- USERS、ADMINS、GUESTS、PUBLIC、LOCAL は、大文字小文字の別に関係なく、ユーザー ID として使用できません。
- IBM、SQL、SYS は、大文字小文字の別に関係なく、ユーザー ID の先頭の 3 文字として使用できません。
- Windows サービス名と同じユーザー ID は使用できません。
- ユーザー ID はローカル・マシン上で定義されていなければならず、ローカル管理者のグループに属していなければなりません。
- ユーザー ID には、拡張ユーザー権限として *Act as part of the operating system* が付与されていなければなりません。

Act as part of the operating system 拡張ユーザー権限を持っていなくてもインストールは実行できますが、 DB2 セットアップ・プログラムは、管理サーバーに指定したアカウントの妥当性検査を行うことができません。DB2 のインストールに使用するユーザー・アカウントはすべて、この拡張ユーザー権限を持つことをお勧めします。

重要

使用している Windows のユーザー ID に管理者権限がない場合、ユーザー ID の長さが 8 文字を超える場合、またはローカル・マシン上で定義されていない場合には、その問題についての通知が出され、インストールを続行することはできません。

DB2 を使用している場合は、後でこのユーザー ID を DB2 データベース・ユーザー名 (データベース・ユーザーのログオン ID) として使用します。

上述の基準を満たすユーザー ID を作成する必要がある場合、 Windows オンライン・ヘルプで Windows ユーザー ID の作成に関する情報を見つけることができます。

iSeries ユーザー・プロファイル 400

WebSphere Commerce をインストールして構成するときは、以下の 2 つの iSeries ユーザー・プロファイルを頻繁に使用および参照します。

- WebSphere Commerce をインストールしたり構成マネージャーにアクセスしたりするのに作成して使用するユーザー・プロファイル。 WebSphere Commerce をインストールして構成するには、USRCLS(*SECOFR) の iSeries ユーザー・プロファイルを使用するか、または QSECOFR ユーザー・プロファイルを使用しなければなりません。ユーザー・プロファイルを作成する必要がある場合、iSeries 用の *WebSphere Commerce インストール・ガイド、バージョン 5.4* を参照してください。
- WebSphere Commerce インスタンスの作成時に構成マネージャーによって作成されるユーザー・プロファイル。このユーザー・プロファイルは、「インスタンス・ユーザー・プロファイル」とも呼ばれます。USRCLS(*USER) のユーザー・プロファイルは、WebSphere Commerce インスタンスを作成するごとに構成マネージャーによって作成されます。ユーザー・プロファイルを作成する必要がある場合、iSeries 用の *WebSphere Commerce インストール・ガイド、バージョン 5.4* を参照してください。

構成マネージャーのユーザー ID

構成マネージャー・ツールのグラフィカル・インターフェースを使用すれば、 WebSphere Commerce の構成方法を変更できます。構成マネージャーのデフォルト・ユーザー ID およびパスワードは、 `webadmin` および `webibm` です。

Windows AIX Solaris Linux 構成マネージャーには、 WebSphere Commerce マシンからか、または WebSphere Commerce と同じネットワーク上の任意のマシンからアクセスできます。

400 iSeries の場合、構成マネージャーには、 iSeries サーバーと同じネットワーク上にある任意の Windows マシンからアクセスすることができます。

IBM HTTP Server のユーザー ID Windows AIX Solaris Linux

IBM HTTP Server を使用している場合、Web ブラウザーをオープンして、以下の Web アドレスを入力すれば Web サーバーのホーム・ページにアクセスできます。

```
http://host_name
```

Web サーバーをカスタマイズした場合、ホスト名の後に Web サーバーのフロントページの名前を入力する必要があります。

WebSphere Commerce Instance Administrator

インスタンス管理者のユーザー ID とパスワードは、以下の WebSphere Commerce ツールに適用されます。

- WebSphere Commerce Accelerator. Windows オペレーティング・システムが実行されているリモート・マシンから WebSphere Commerce Accelerator にアクセスするには、 Internet Explorer Web ブラウザーをオープンしてから、以下の Web アドレスを入力します。

  ```
  https://host_name:8000/accelerator
  ```

- WebSphere Commerce 管理コンソール. Windows オペレーティング・システムが実行されているリモート・マシンから WebSphere Commerce 管理コンソールにアクセスするには、 Internet Explorer Web ブラウザーをオープンしてから、以下の Web アドレスを入力します。

  ```
  https://host_name:8000/adminconsole
  ```

- ストア・サービス。ストア・サービスのページには、 Web ブラウザーをオープンし、以下の Web アドレスを入力することによってアクセスできます。

  ```
  https://host_name:8000/storeservices
  ```

Instance Administrator のデフォルト・ユーザー ID は wcsadmin、デフォルト・パスワードは wcsadmin です。

注: wcsadmin ユーザー ID は、決して削除しないようにしてください。また、それには常に Instance Administrator の権限が付与されていなければなりません。

WebSphere Commerce では、ユーザー ID とパスワードに関して次の規則に従う必要があります。

- パスワードの長さは、少なくとも 8 文字なければなりません。
- パスワード中では、少なくとも 1 字の数字を使用しなければなりません。
- パスワード中では、1 つの文字が 4 回を超えて出現してはなりません。
- パスワード中では、同じ文字を 3 回を超えて繰り返し使用してはなりません。

Payment Manager 管理者

Payment Manager をインストールすると、 WebSphere Commerce 管理者 ID wcsadmin に Payment Manager 管理者役割が自動的に割り当てられます。Payment Manager の Realm Class をまだ WCSRealm に切り替えていない場合、 *WebSphere Commerce インストール・ガイド、バージョン 5.4* の指示に従って切り替えてください。

Payment Manager 管理者の役割を使えば、ユーザー ID で Payment Manager を制御および管理することができます。

注: > 400

- ログオン・ユーザー ID `wcsadmin` は削除したり名前を変更したりしないでください。また、`wcsadmin` に事前に割り当てられている Payment Manager の役割は変更しないようにしてください。もし変更すると、Payment Manager の整合性に関連した WebSphere Commerce の機能が動作しなくなります。
- WebSphere Commerce の管理者に Payment Manager の役割を割り当てた場合、後でその管理者のログオン・ユーザー ID を削除したり名前を変更したりするときには、ユーザー ID を削除または名前変更する前に、まずその管理者に割り当てた Payment Manager の役割を削除してください。

重要

Payment Manager は、他の 2 つの管理 ID に Payment Manager 管理者役割を事前に割り当てています。

- `ncadmin`
- `admin`

あるユーザーが誤ってその Payment Manager 管理者役割を取得することがないようにするには、以下のようにします。

1. WebSphere Commerce 管理コンソールを使用して、WebSphere Commerce で上記の管理 ID を作成します。
2. Payment Manager のユーザー・インターフェースで、**「ユーザー」**を選択します。
3. 2 つの管理者 ID から Payment Manager 管理者の役割を削除します。

また、Payment Manager インスタンスの開始、停止、または削除に必要な Payment Manager インスタンス・パスワードにも注意が必要です。また、Payment Manager インスタンスにカセットを追加する必要もあります。 Payment Manager インスタンスを WebSphere Commerce 構成マネージャーで作成する場合、 Payment Manager インスタンスのパスワードは、インスタンス・ユーザー・プロファイル・パスワードとも呼ばれる WebSphere Commerce インスタンスのログオン・パスワードと同じになります。 Payment Manager インスタンスを、**CRTPYMMGR** コマンドを使って iSeries セッションから作成するか、または iSeries タスク・ページから作成する場合、パスワードを入力するようプロンプトで指示されます。

構成マネージャー・パスワードの変更

構成マネージャー・パスワードを変更するには、構成マネージャーを立ち上げてから、ユーザー ID とパスワードを入力するウィンドウで**「変更」**をクリックします。

Windows AIX Solaris Linux あるいは、構成マネージャーのユーザー ID とパスワードを変更するために、 WebSphere Commerce の下の `bin` サブディレクトリーに切り替えてから、コマンド・ウィンドウに以下のコマンドを入力します。

```
config_env
java com.ibm.commerce.config.server.PasswordChecker -action [action type]
   -pwfile [password file] -userid [user ID]
   -password [userid password] [-newpassword [new userid password]]
```

ここで、action type (アクション・タイプ) は、`Add`、`Check`、`Delete`、または `Modify` です。各パラメーターについて以下に説明します。

pwfile
パスワードが保管されるファイルのパス。デフォルト・パスは、WebSphere Commerce インストール・パスの下の `bin` サブディレクトリーです。このパラメーターは常に必須です。

userid
追加、検査、削除、または変更するユーザー ID を入力します。このパラメーターは常に必須です。

password
作成、検査、削除、または変更するパスワードを入力します。このパラメーターは、`userid` パラメーターと組み合わせて使用する必要があります。このパラメーターは常に必須です。

newpassword
特定のユーザー ID のパスワードを変更するには、このパラメーターを使用します。このパラメーターは、`userid` および `password` パラメーターと組み合わせて使用する必要があります。このパラメーターが必要なのは、アクション・タイプ `Modify` を指定する場合です。

IBM HTTP Server 管理者パスワードの設定

Windows AIX Solaris Linux IBM HTTP Server 管理者のパスワードを設定するには、次のようにします。

1. マシン上の IBM HTTP Server インストール・ディレクトリーに切り替えます。
2. 以下のコマンドを入力します。

   Windows `htpasswd -b conf¥admin.passwd *user password*`

   AIX Solaris Linux `htpasswd -b conf/admin.passwd *user password*`

   ここで、*user* および *password* は、IBM HTTP Server への管理権限を付与するユーザー ID およびパスワードです。

これで、IBM HTTP Server 管理パスワードを正しく設定できました。

SSL 鍵ファイル・パスワードの変更

Windows AIX Solaris Linux IBM HTTP Server を使用している場合、SSL 鍵ファイル・パスワードを変更するには、以下のステップに従います。

1. Windows **「スタート」メニュー → 「プログラム」 → 「IBM HTTP Server」 → 「Key Management Utility (鍵管理ユーティリティー)」**をクリックします。
2. **「Key Database File (鍵データベース・ファイル)」**メニューから、**「オープン」**を選択します。
3. IBM HTTP Server インストール・パスの下の `ssl` サブディレクトリーに切り替えます。鍵ファイル (ファイル拡張子 `.kdb`) は、このフォルダーに入っていなければなりません。入っていない場合は、87 ページの『第 8 章 IBM HTTP Server での実動のための SSL の使用可能化』で示されている指示に従って新しい鍵ファイルを作成します。
4. **「Key Database File (鍵データベース・ファイル)」**メニューから、**「パスワード変更」**を選択します。「パスワード変更」ウィンドウが表示されます。
5. 新しいパスワードを入力し、**「Stash the password to a file (パスワードをファイルに隠す)」**を使用可能にします。
6. **「OK」**をクリックします。パスワードが変更されます。

これで、SSL 鍵ファイルの管理パスワードを正しく変更できました。

WebSphere Commerce 暗号化パスワードの生成

Windows AIX Solaris Linux WebSphere Commerce を使用して、暗号化されたパスワードを生成できます。暗号化パスワードを生成するには、以下のようにします。

1. WebSphere Commerce インストール・ディレクトリーの下の `bin` サブディレクトリーに進みます。
2. コマンド行から以下のスクリプトを実行します。

   Windows `wcs_password.bat password SALT merchant_key`

   AIX Solaris Linux `./wcs_password.sh password SALT merchant_key`

   詳細は次のとおりです。

   - *password* はプレーン・テキストのパスワードです。
   - *SALT* は、パスワードの生成で使われるランダム・ストリングです。これは、パスワードを更新している特定ユーザーの `USERREG` データベース・テーブルの `SALT` 列にあります。
   - *merchant_key* はインスタンスの作成中に入力したマーチャント・キー

400 iSeries の場合にショッパー用の暗号化パスワードを変更するには、`CHGWCSPWD` コマンドを使用します。このコマンドの実行に関する詳細は、F1 オンライン・ヘルプを参照してください。

Payment Manager 暗号化パスワードの生成

WebSphere Commerce を使用して、Payment Manager の暗号化パスワードを生成できます。暗号化パスワードを生成するには、以下のようにします。

1. WebSphere Commerce インストール・ディレクトリーの下の `bin` サブディレクトリーに進みます。
2. コマンド行から以下のスクリプトを実行します。

   Windows `wcs_pmpassword.bat password SALT`

   AIX Solaris Linux `./wcs_pmpassword.sh password SALT`

   ここで、

   - *password* はプレーン・テキストのパスワードです。
   - *SALT* は、パスワードの生成で使われるランダム・ストリングです。これは、パスワードを更新している特定ユーザーの `USERREG` データベース・テーブルの `SALT` 列にあります。

400 iSeries の場合、Payment Manager 用の暗号化パスワードを生成するには、 `CRTWCSPMPW` コマンドを使用します。このコマンドの実行に関する詳細は、F1 オンライン・ヘルプを参照してください。

第 8 章 IBM HTTP Server での実動のための SSL の使用可能化

400 この項は、 iSeries プラットフォームには当てはまりません。 iSeries に関する詳細は、 91 ページの『IBM HTTP サーバーでの SSL の使用可能化 (iSeries)』を参照してください。

IBM HTTP Server で WebSphere Commerce インスタンスを作成し終えると、SSL (Secure Sockets Layer) を使ってテストすることができます。サイトをショッパーに対してオープンする前に、この章の以下のステップを実行して、SSL を実動用に使用可能にしなければなりません。

セキュリティーについて

IBM HTTP Server は暗号化テクノロジーを使用して、商取引のための機密保護機能のある環境を提供します。暗号化とは、インターネット上の情報トランザクションをスクランブルし、受信側がスクランブル解除するまで判読不能にすることです。送信側は算法パターンつまり鍵を使用してトランザクションをスクランブル (暗号化) し、受信側は復号鍵を使用します。これらの鍵は、Secure Sockets Layer (SSL) プロトコルで使用されます。

Web サーバーは認証プロセスを使用して、ビジネス上の取引をしている個人の識別を検証します (つまり、当人が呼称されるとおりの人物であることを確認します)。これには、認証局 (CA) と呼ばれる信頼のおける第三者機関によって署名された証明書を取得することが含まれます。 IBM HTTP Server ユーザーの場合、CA は Equifax® や VeriSign® Inc. などです。他の CA も同様に使用可能です。

実動鍵ファイルを作成するには、以下のステップを完了します。

1. 実動用のセキュリティー鍵ファイルを作成します。
2. 認証局からセキュアな証明書を要求します。
3. 実動鍵ファイルを現行鍵ファイルとして設定します。
4. 証明書を受け取り、実動鍵ファイルをテストします。

これらのステップについて、以下に詳細に説明します。

注:

1. 認証局が署名した実動鍵ファイルをすでに使用している場合、これらのステップを省略することもできます。この章を読んで決定してください。
2. これらのステップを実行する際に、ブラウザーにセキュリティー・メッセージが表示されることがあります。それぞれのメッセージに示された情報を注意深く確認して、続行する方法を判別してください。

実動用のセキュリティー鍵ファイルの作成

実動用のセキュリティー鍵ファイルを作成するには、 Web サーバー・マシンで以下のようにします。

1. IBM HTTP Server を停止します。
2. マシン上の IBM HTTP Server インストール・サブディレクトリーの下の conf サブディレクトリーにディレクトリーを変更します。
3. httpd.conf のバックアップ・コピーを作成します。
4. httpd.conf をテキスト・エディターでオープンします。
5. ポート 443 について、以下の行がコメント化されていないことを確認します。
   - Windows

     ```
     #LoadModule ibm_ssl_module modules/IBMModuleSSL128.dll
     #Listen 443#Listen 443#<VirtualHost host.some_domain.com:443>
     #SSLEnable
     #</VirtualHost>
     #SSLDisableKeyfile "drive:/WebSphere/HTTPServer/ssl/keyfile.kdb"
     #SSLV2Timeout 100
     #</VirtualHost>
     #SSLDisable
     ```

   - AIX Solaris Linux

     ```
     #LoadModule ibm_ssl_module modules/IBMModuleSSL128.dll
     #AddModule mod_ibm_ssl.c
     #Listen 443#<VirtualHost host.some_domain.com:443>
     #SSLEnable
     #</VirtualHost>
     #SSLDisableKeyfile "keyfile"
     #SSLV2Timeout 100
     #</VirtualHost>
     #SSLDisable
     ```

     ただし *keyfile* は、以下のいずれかです。

     AIX `/usr/HTTPServer/ssl/keyfile.kdb`

     Solaris `/opt/IBMHTTPD/ssl/keyfile.kdb`

     Linux `/opt/IBMHTTPServer/ssl/keyfile.kdb`
6. ポート 8000 について、以下の行がコメント化されていないことを確認します。
   a. `#Listen 8000`
   b. `#<VirtualHost host.some_domain.com:8000>` さらに、この行内の完全修飾ホスト名を置き換える必要もあります。
   c. `#SSLEnable`
   d. `#</VirtualHost>`

   注: WebSphere Commerce ツール用に構成したポート (デフォルトではポート 8000) への外部アクセスをファイアウォール・ソフトウェアで阻止することをお勧めします。その方法に関する詳細は、サイトでご使用のファイアウォール・ソフトウェアの資料を調べてください。
7. 変更を保管します。

8. `httpd.conf` ファイルに構文エラーが入らないようにするには、マシン上の IBM HTTP Server インストール・ディレクトリーの下の `bin` サブディレクトリーに移動し、次のコマンドを実行します。

   AIX Solaris Linux `./apachectl configtest`

   Windows `apachectl configtest`

9. IBM HTTP Server を開始します。

認証局に対するセキュアな証明書の要求

前のステップで作成したセキュリティー鍵を妥当性検査するには、 Equifax や VeriSign などの認証局 (CA) が発行した証明書が必要です。証明書には、サーバーの公開鍵、サーバーの証明書に関連した識別名、および証明書のシリアル番号と有効期限が含まれています。

他の CA を使用する場合、実行する手順については、直接その CA に問い合わせてください。

Equifax ユーザー

Equifax からセキュア・サーバー証明書を要求するには、以下の Web アドレスを参照して、示される指示に従ってください。

`http://www.equifax.com`

Equifax からの証明書は E メールで 2 ～ 4 日以内に送られてきます。

VeriSign ユーザー

VeriSign からセキュア・サーバー証明書を要求するには、以下の URL を参照して、示される指示に従ってください。

`http://www.verisign.com`

AIX IBM HTTP Server 用の手順を使用していても、 **Internet Connection Secure Server** (ICSS) のリンクをたどります。示される指示に従ってください。実動鍵ファイルをまだ作成していないならば、証明書ファイルを受け取ったときに、前の項で説明した方法によってそれを作成してください。

Solaris IBM HTTP Server 用の手順を使用していても、 **Internet Connection Secure Server** (ICSS) のリンクをたどります。後続のページには、その手順は OS/2® および AIX プラットフォームに適用されることが示されています。これらの指示は Solaris ソフトウェアにも適用されます。

示される指示に従ってください。要求を送信すると、証明書は 3 ～ 5 日以内に送られてきます。実動鍵ファイルをまだ作成していないならば、証明書ファイルを受け取ったときに、前の項で説明した方法によってそれを作成してください。

実動鍵ファイルの受け取りと現行鍵ファイルとしての設定

CA からの証明書が到着した後、Web サーバーが実動鍵ファイルを使用するように設定する必要があります。以下のステップを完了します。

1. 認証局から受け取った *certificatename*.kdb、 *certificatename*.rdb、および *certificatename*.sth ファイルを、マシン上の IBM HTTP Server インストール・パスの下の ssl サブディレクトリーにコピーします。 *certificatename* は認証要求と共に指定した証明書名です。
2. 鍵管理ユーティリティーをオープンします。
3. *certificatename*.kdb ファイルをオープンして、プロンプトが出たらパスワードを入力します。
4. **「Personal Certificates (個人用証明書)」**を選択して、**「受け取り」**をクリックします。
5. **「参照」**をクリックします。
6. 認証局から受け取ったファイルを格納しているフォルダーを選択します。 *certificatename*.txt ファイルを選択して、**「OK」**をクリックします。
7. これで**「Personal Certificates (個人用証明書)」**リスト・ボックスには、VeriSign *certificatename* 証明書または Equifax *certificatename* 証明書がリストされます。
8. 鍵管理ユーティリティーを終了します。
9. マシン上の IBM HTTP Server インストール・パスの下の conf サブディレクトリーにディレクトリーを変更します。
10. httpd.conf のバックアップ・コピーを作成します。
11. httpd.conf をテキスト・エディターでオープンします。
12. ステップ 5 (88 ページ) でリストされた行がコメント化されていないことを確認します。
13. Keyfile "*keyfile path name*" ディレクティブを検索して、上記のステップで作成されたファイルを指し示すようにパス名を変更します。
14. IBM HTTP Server を停止してから、再始動します。

実動鍵ファイルのテスト

実動鍵をテストするには、以下のようにします。

1. ブラウザーを使用して以下の URL を表示します。

   https://*host_name*

   注:

   a. Web サーバーをカスタマイズしている場合、ホスト名の後に Web サーバーのフロントページの名前を入力しなければならないことがあります。

   b. http ではなく https と入力します。

   鍵が正しく定義されていれば、新規の証明書に関するいくつかのメッセージが表示されます。

2. **「New Site Certificate (新規のサイト証明書)」**パネルで、この証明書を受け入れたい場合、**「Accept this certificate forever (until it expires) (この証明書を永続的に (有効期限が切れるまで) 受け入れる)」** ラジオ・ボタンを選択します。
3. Web ブラウザーから、キャッシングおよびプロキシー (または Socks) サーバーの設定値を初期値に戻します。

これで、サーバー上で SSL が使用可能になりました。

Payment Manager の場合の SSL に関する考慮事項

デフォルトでは、WebSphere Commerce と Payment Manager の間の通信は SSL を経由します。これに対して、たとえば次のようにして Payment Manager ユーザー・インターフェースを立ち上げたとします。

http://*host_name*/webapp/Paymentmanager/

この場合、非 SSL 通信を使って Payment Manager を呼び出すことになります。必ず SSL を通して通信するためには、

https://*host_name*/webapp/Paymentmanager/

を使用するか、または、以下のディレクトリー内の indexSSL.html ファイルの名前を index.html に変更します。

- Windows

  WAS_HOME¥installedApps¥IBM_PaymentManager.ear¥PaymentManager.war

- AIX Solaris Linux

  WAS_HOME/installedApps/IBM_PaymentManager.ear/PaymentManager.war

このようにすれば、**http://***host_name*/webapp/Paymentmanager/ ディレクトリーを引き続き使用することができ、名前を変更した index.html は https (SSL) にリダイレクトされます。

IBM HTTP サーバーでの SSL の使用可能化 (iSeries)

400 この項では、 iSeries プラットフォームに関連した説明を述べます。

SSL は、セキュリティー・プロトコルです。 SSL を使うと、クライアントとサーバーがやりとりするデータを専用データのままに保つことができます。それによって、クライアントはサーバーの ID を認証し、サーバーはクライアントの ID を認証することができます。

ディジタル証明書とは、インターネット上のセキュア・トランザクションに関与するサーバーとクライアントを認証するための電子文書のことです。ディジタル証明書の発行者を認証局 (CA) と呼びます。 iSeries システムは、イントラネット環境においてサーバーおよびクライアントの証明書を発行する CA の役割を果たすことで、 iSeries CA または VeriSign® のようなインターネット CA から発行されるサーバー証明書を持った認証済みサーバーとして稼働することができます。 IBM HTTP Server for iSeries が Web サーバーの場合、 SSL 対応のクライアントの認証用のクライアント証明書を要求するようこのサーバーを構成することもできます。

IBM HTTP Server for iSeries 上で SSL を使用可能にする方法の詳細は、以下の Web アドレスを参照してください。

www.ibm.com/software/webservers/commerce/servers/lit-tech-os400.html

中でも特に、「**Hints and Tips (ヒント)**」の項に目を通してください。

Payment Manager での SSL の使用

WebSphere Commerce インスタンスを作成した後でシステム証明書ストアを作成する場合、 Payment Manager と WebSphere Commerce インスタンスの両方に対して、そのシステム証明書ストアへのアクセスを認可する必要があります。たとえば、以下のコマンドは、 V5R1 システムで必要なアクセス権を Payment Manager インスタンスに認可します。

```
CHGAUT OBJ('/QIBM/UserData/ICSS/Cert/Server') USER(QPYMSVR) DTAAUT(*RX)
CHGAUT OBJ('/QIBM/UserData/ICSS/Cert/Server/DEFAULT.KDB') USER(QPYMSVR) DTAAUT(*R)
```

また、以下のコマンドは、 V5R1 システムで必要なアクセス権を WebSphere Commerce インスタンスに認可します。

```
CHGAUT OBJ('/QIBM/UserData/ICSS/Cert/Server') USER(QEJBSVR) DTAAUT(*RX)
CHGAUT OBJ('/QIBM/UserData/ICSS/Cert/Server/DEFAULT.KDB') USER(QEJBSVR) DTAAUT(*R)
```

リモート Payment Manager インスタンスを使用することにした場合、ディジタル証明書を発行するリモート認証局を信頼するように、 WebSphere Commerce インスタンスと Payment Manager インスタンスの両方を構成する必要があります。この 2 つのリモート・アプリケーションの間に信頼関係を確立したい場合は、以下の高レベルの手順を参照してください。

1. WebSphere Commerce マシンで、ディジタル証明書マネージャーを使ってサーバーの認証局をエクスポートします。
2. 証明書ファイルを Payment Manager マシンに転送します。
3. Payment Manager マシンで、ディジタル証明書マネージャーを使って WebSphere Commerce サーバーの認証局をインポートします。
4. インポートした WebSphere Commerce サーバーの認証局を信頼するように Payment Manager アプリケーション・サーバーを構成します。
5. Payment Manager マシンで、ディジタル証明書マネージャーを使ってサーバーの認証局をエクスポートします。
6. 証明書ファイルを WebSphere Commerce マシンに転送します。
7. WebSphere Commerce マシンで、ディジタル証明書マネージャーを使って Payment Manager サーバーの認証局をインポートします。
8. インポートした Payment Manager サーバーの認証局を信頼するように WebSphere Commerce アプリケーション・サーバーを構成します。

詳細は、以下の Web アドレスの**「Hints and Tips (ヒント)」**の項を参照してください。

```
www.ibm.com/software/webservers/commerce/servers/lit-tech-os400.html
```

第 9 章 IBM SecureWay Directory Server (LDAP) での SSL の使用可能化

以下に、IBM SecureWay Directory Server と WebSphere Commerce 用の SSL セキュリティーを構成するステップを示します。

SecureWay のセットアップ

IBM SecureWay Directory Server をセットアップするには、以下のようにします。

1. SecureWay Directory Server 製品のインストール指示に従って IBM SecureWay Directory Server をインストールします。必ず GSKit コンポーネントをインストールしてください。
2. インストールが完了したら、IBM Key Manager (Windows では *drive*:¥Program Files¥IBM¥GSK4¥bin¥gsk4ikm.exe) を呼び出します。
3. 新規の CMS 鍵データベース・ファイルを作成します。**「Stash the password to file (ファイルへのパスワードの stash)」**が選択されていることを確かめます (たとえば ldap_key.kdb)。
4. 自己署名証明書を作成します。
5. Base64 エンコード ASCII データのデータ・タイプで証明書を抽出します。
6. 新規の SSLight 鍵データベース・クラス (たとえば keyring.class) を作成します。
7. **「Singer Certificates (Singer 証明書)」**セクションで、ステップ 5 で作成した証明書ファイルを追加します。
8. ブラウザーをオープンしてアドレス http://hostname/ldap を表示します。
9. **「セキュリティー」→「SSL」→「設定」**をクリックし、次のような変更を加えます。
   - SSL 状況: SSL をオンまたは SSL のみ
   - 認証方法: サーバー認証
   - セキュア・ポート: 636
   - 鍵データベース・パスおよびファイル名:

     AIX Solaris Linux /Keys/ldap_key.kdb

     Windows *drive*:¥Keys¥ldap_key.kdb
   - 鍵ラベル: *your_label* (証明書のラベル)
10. **「更新」**をクリックし、SecureWay を再始動します。

WebSphere Commerce

SecureWay を扱えるように WebSphere Commerce をセットアップするには、次のように *instance*.xml ファイルを変更する必要があります。

```
java.naming.security.ssl.keyring = keyring
'keyring' is the name of the SSLight key database class (keyring.class)
This class file should put in the class path in WAS.

java.naming.security.ssl.authentication = ibm
'ibm' is the password specified when create the SSLight key database class.

java.naming.security.protocol = ssl
LdapPort = 636
```

```
<MemberSubSystem name="Member SubSystem"
                     ProfileDataStorage="LDAP"
                     AuthenticationMode="LDAP">
  <Directory LdapAdminDN="cn=root"
                 LdapAuthenticationMode="SIMPLE"
                 LdapTimeOut="0"
                 LdapVersion="3"
                 EntryFileName="WC_Install_Dir/xml/ldap/ldapentry.xml"
                 LdapPort="636"
                 SingleSignOn="0"
                 LdapAdminPW="EaDPFd9VAf0="
                 LdapHost="yazhuang.torolab.ibm.com"
                 MigrateUsersFromWCSdb="ON"
                 JNDIEnvPropName1="java.naming.security.ssl.keyring"
                 JNDIEnvPropValue1="keyring"
                 JNDIEnvPropName2="java.naming.security.ssl.authentication"
                 JNDIEnvPropValue2="ibm"
                 JNDIEnvPropName3="java.naming.security.protocol"
                 JNDIEnvPropValue3="ssl"
                 display="false"
                 LdapType="SECUREWAY" />
</Membersubsystem>
```

WebSphere Commerce を再始動します。

第 10 章 単一サインオン

この章では、WebSphere Commerce 用に単一サインオンをセットアップする方法の概要を述べます。

前提条件

単一サインオンを使用可能にするには、以下の要件を満たす必要があります。

- すでに LDAP サーバーがインストールおよび構成済みになっていなければなりません。 LDAP サーバーの構成方法の詳細は、 *IBM WebSphere Commerce 追加ソフトウェア・ガイド、バージョン 5.4* を参照してください。
- WebSphere Commerce がインストールされていて、LDAP を使用するように構成済みでなければなりません。
- WebSphere Application Server セキュリティーが使用可能になっている必要があります。 WebSphere Application Server セキュリティーを使用可能にする方法の詳細は、61 ページの『第 5 章 WebSphere Application Server のセキュリティーの使用可能化』を参照してください。

単一サインオンの使用可能化

制限事項

単一サインオンを WebSphere Commerce で使用する場合、次のようないくつかの主な制限事項があります。そのような制限事項を以下に示します。

- LTPA cookie は、さまざまな Web サーバー・ポート間でやりとりされる可能性があります。
- `ldapentry.xml` ファイルを変更し、オブジェクト・クラス `ePerson` を追加する必要があるかもしれません。それは、`ldapocs` エレメントの属性として追加します。
- `instance.xml` を変更し、 LDAP コンポーネント内でユーザー用にマイグレーションを必ずオンにする必要があります。
- 単一サインオン構成に属する各マシンは、それぞれのシステム・クロックを同期させる必要があります。
- 単一サインオンがサポートされるのは、 WebSphere Application Server LTPA (Lightweight Third Party Authentication) トークンの読み取りと発行を行えるアプリケーションどうしの場合のみです。

単一サインオンを使用可能にするには、以下を行う必要があります。

1. WebSphere Application Server 内で単一サインオンを使用可能にします。詳細は、以下のアドレスに掲載されている WebSphere Application Server InfoCenter の中の『single sign-on (単一サインオン)』を参照してください。

   ```
   http://www.ibm.com/software/webservers/appserv/doc/v40/ae
   /infocenter/index.html
   ```

「Single Sign-On: WebSphere Application Server (単一サインオン: WebSphere Application Server)」を選択してから、以下の項の説明どおりにします。

- **WebSphere Application Server 用の SSO の構成。**
  - **WebSphere Application Server のセキュリティー設定の変更。**

    注: LDAP フィールドへの入力方法について詳述したステップは、省いても問題はありません。
  - **ファイルへの LTPA 鍵のエクスポート。**

2. WebSphere Commerce マシンで WebSphere Commerce 構成マネージャーを開始します。
3. **「メンバー・サブシステム」**ノードを構成するには、次のようにします。
   a. **「WebSphere Commerce」→** *host_name* **→「インスタンス・リスト」→** *instance_name* **→「インスタンス・プロパティー」→「メンバー・サブシステム」**の順に拡張表示します。
   b. **「認証モード」**ドロップダウン・メニューで **LDAP** を選択します。
   c. **「Single Sign-On (単一サインオン)」**チェック・ボックスを使用可能にします。
   d. **「ホスト」**フィールドに LDAP サーバーの完全修飾ホスト名を入力します。
   e. 管理者の識別名を**「管理者識別名」**フィールドに入力します。これは、LDAP サーバーで使用したものと同じ名前でなければなりません。
   f. **「管理者のパスワード」**フィールドに管理者のパスワードを入力します。これは、LDAP サーバーで使用したものと同じパスワードでなければなりません。**「確認パスワード」**フィールドのパスワードを確認します。
   g. 残りのフィールドをすべて完了します。
   h. **「適用」**をクリックしてから、**「OK」**をクリックします。
4. WebSphere Application Server を再始動します。

第 4 部 WebSphere Commerce 開発者のセキュリティー・タスク

第 4 部では、WebSphere Commerce のプログラミングに関連したセキュリティー・タスクについて説明します。このタスクは通常は WebSphere Commerce プログラマーによって実行されます。

第 11 章 アクセス・コントロール

アクセス・コントロールの理解

WebSphere Commerce アプリケーションのアクセス・コントロール・モデルには 3 つの主な概念、すなわちユーザー、アクション、およびリソースがあります。ユーザーは、システムを使用する人間です。リソースは、アプリケーション内で、またはアプリケーションによって保守されるエンティティーです。たとえばリソースには、商品、文書、オーダーなどがあります。人間を表すユーザー・プロファイルもリソースです。アクションとは、ユーザーがリソースで実行できるアクティビティーのことです。アクセス・コントロールとは、特定のユーザーが特定のリソースで特定のアクションを実行できるかどうかを決定する、 e-commerce アプリケーションのコンポーネントです。

WebSphere Commerce アプリケーションでは、主要な 2 つのレベルのアクセス・コントロールがあります。アクセス・コントロールの第 1 レベルは WebSphere Application Server によって実行されます。ここでは、WebSphere Commerce が WebSphere Application Server を使用してエンタープライズ Bean およびサーブレットを保護します。アクセス・コントロールの 2 次レベルは、 WebSphere Commerce のきめ細かいアクセス・コントロール・システムです。

WebSphere Commerce アクセス・コントロール・フレームワークでは、アクセス・コントロール・ポリシーを使用して、特定のユーザーが特定のリソースで特定のアクションの実行を許可されているかどうかを判別します。このアクセス・コントロールのフレームワークは、きめ細かいアクセス・コントロールの手段になります。これは、WebSphere Application Server に備わったアクセス・コントロールと共同で稼働しますが、これに代わるものではありません。

WebSphere Application Server でのリソース保護の概要

以下の WebSphere Commerce リソースは、WebSphere Application Server によるアクセス・コントロールの下で保護されます。

- エンティティー Bean
  これらの bean は、e-commerce アプリケーション内のオブジェクトをモデル化します。これらは、リモート・クライアントからアクセスできる分散オブジェクトです。
- JSP テンプレート
  WebSphere Commerce は、表示ページに JSP テンプレートを使用します。各 JSP テンプレートには、データを から検索する 1 つまたは複数のデータ Bean を含めることができます。クライアントは、URL 要求を構成することによって JSP ページを要求することができます。
- コントローラーおよびビュー・コマンド
  クライアントは、URL 要求を構成することによってコントローラーおよびビュー・コマンドを要求することができます。加えて、 VIEWREG テーブルで登録されている JSP ファイル名またはビュー名を使用することにより、 1 つの表示ページに他の表示ページへのリンクを含めることができます。

通常、WebSphere Commerce Server は、以下の Web パスを使用するように構成されています。

- `/webapp/wcs/stores/servlet/*`
  これは、要求サーブレットの要求に使用します。
- `/webapp/wcs/stores/*.jsp`
  これは、JSP サーブレットの要求に使用します。

以下の図は、上記の Web パス構成の場合に要求が WebSphere Commerce リソースにアクセスする際に潜在的にたどる可能性のある経路を示しています。

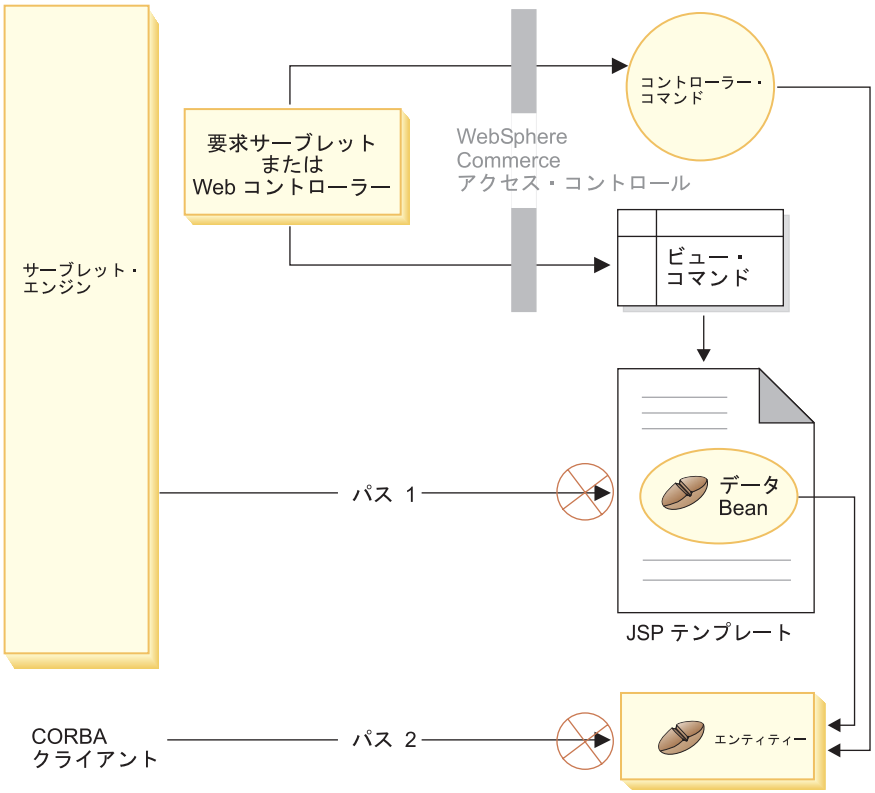


図 3.

正当な要求はすべて、要求サーブレットに送られる必要があります。次いで、要求サーブレットは、これを Web コントローラーに送ります。Web コントローラーは、コントローラー・コマンドおよびビューのためのアクセス・コントロールをインプリメントしています。しかしながら、上記の Web パスでは、悪質なユーザーが JSP テンプレート (パス 1) や (パス 2) に直接アクセスすることができます。これらの悪質なアタックが成功しないようにするため、実行時に拒否することが必要です。

JSP テンプレートと への直接アクセスは、以下のいずれかの方法によって防止することができます。

WebSphere Application Server セキュリティー

WebSphere Application Server はセキュリティー機能を備えています。この方法を使用すれば、すべてのエンタープライズ Bean メソッドと JSP テンプレートは、 System Identity だけが呼び出すように構成されます。これら

の WebSphere Commerce リソースにアクセスするには、 URL 要求を、Web コントローラーに渡す前に、 System Identity を現行スレッドに設定する要求サーブレットに経路指定する必要があります。次いで、Web コントローラーは、要求を対応するコントローラー・コマンドまたはビューに渡す前に、呼び出し元が必要な許可を持っているかどうかを確かめます。 JSP テンプレートやエンティティー Bean に直接 (つまり、 Web コントローラーを使用せずに) アクセスしようとする試みはすべて、 WebSphere Application Server セキュリティー・コンポーネントによって拒否されます。

WebSphere Commerce リソースを保護するための WebSphere Application Server 構成については、 *WebSphere Commerce インストール・ガイド* を参照してください。 WebSphere Application Server 内のセキュリティーについては、 WebSphere Application Server 資料のシステム管理のトピックを参照してください。

カスタマイズされたエンタープライズ Bean でメソッドの WebSphere Application Server セキュリティーを構成する方法の詳細は、 *WebSphere Commerce プログラマーズ・ガイド、バージョン 5.4* の中の『新しいエンタープライズ Bean のエンタープライズ・アプリケーションへのアセンブル』および『変更されたエンタープライズ Bean のエンタープライズ・アプリケーションへのアセンブル』の項を参照してください。

ファイアウォール保護

WebSphere Commerce Server がファイアウォールの背後で稼働していると、インターネット・クライアントはエンティティー Bean に直接アクセスできません。この方法を使用する場合、 JSP テンプレートの保護は、ページに組み込まれたデータ Bean によって提供されます。データ Bean は、データ Bean マネージャーによって活動化されます。データ Bean マネージャーは、 JSP テンプレートがビュー・コマンドによって転送されたかどうかを調べます。これがビュー・コマンドによって転送されなかった場合は、例外が戻され、JSP テンプレートの要求は拒否されます。

WebSphere Commerce アクセス・コントロール・ポリシーの概要

WebSphere Commerce のアクセス・コントロール・モデルは、アクセス・コントロール・ポリシーの実効化に基づきます。アクセス・コントロール・ポリシーでは、アクセス・コントロール規則をビジネス・ロジック・コードから外部化することができるため、アクセス・コントロール・ステートメントをコードにハードコーディングする必要がなくなります。たとえば、以下のようなコードを組み込む必要はありません。

```
if (user.isAdministrator())
   then {}
```

アクセス・コントロール・ポリシーは、アクセス・コントロール・ポリシー・マネージャーによって実行されます。一般に、保護されたリソースにユーザーがアクセスしようとすると、アクセス・コントロール・ポリシー・マネージャーは最初に、その保護されたリソースに適用できるアクセス・コントロール・ポリシーを決定し、それからその適用できるアクセス・コントロール・ポリシーに基づいて、要求されたリソースへのユーザーのアクセスを許可するかどうかを決定します。

アクセス・コントロール・ポリシーは、 ACPOLICY テーブルに保管される 4 タプルのポリシーです。アクセス・コントロール・ポリシーはそれぞれ以下の形式をとります。

```
AccessControlPolicy [UserGroup, ActionGroup, ResourceGroup, Relationship]
```

4 タプルのアクセス・コントロール・ポリシー内のエレメントでは、特定のユーザー・グループに属するユーザーは、該当するリソースに関して関係または関係グループに指定されている条件を満たす限り、指定されたリソース・グループに属しているリソースで、指定されたアクション・グループのアクションを実行できると指定されます。たとえば、[AllUsers, UpdateDoc, doc, creator] は、文書の作成者であれば、すべてのユーザーが文書を更新できることを指定します。

ユーザー・グループは、MBRGRP データベース・テーブル内で定義されている特定のタイプのメンバー・グループです。ユーザー・グループは、メンバー・グループ・タイプ -2 に関連付けられる必要があります。 -2 という値はアクセス・グループを表しますが、これは MBRGRPTYPE テーブルに定義されます。ユーザー・グループとメンバー・グループ・タイプのアソシエーションは、 MBRGRPUSG テーブルに保管されます。

特定のユーザー・グループへのユーザーのメンバーシップは、明示的または暗黙的のどちらで記述してもかまいません。明示的に指定されるのは、ユーザーが特定のメンバー・グループに属していると MBRGRPMBR テーブルに記述される場合です。 暗黙的に指定されるのは、 MBRGRPCOND テーブルに記述された条件 (たとえば、プロダクト・マネージャーの役割を果たすすべてのユーザー) を、ユーザーが満たす場合です。複合条件 (たとえば、プロダクト・マネージャーの役割を果たし、最低 6 か月間はその役割にあったすべてのユーザー) または明示的な除外も使用できます。

ユーザーをユーザー・グループに組み込むための条件のほとんどは、特定の役割を果たすユーザーに基づきます。たとえば、プロダクト・マネージャーの役割を果たすすべてのユーザーにカタログ管理操作を実行する許可を与えるアクセス・コントロール・ポリシーがあるとします。この場合、MBRROLE テーブルでプロダクト・マネージャーの役割を割り当てられたユーザーはすべて、暗黙的にユーザー・グループに組み込まれます。

メンバー・グループのサブシステムについての詳細情報については、 WebSphere Commerce のオンライン・ヘルプを参照してください。

ActionGroup エレメントは ACACTGRP テーブルからとられます。アクション・グループは、明示的に指定されたアクションのグループを参照します。アクションのリストは ACACTION テーブルに保管され、各アクションとアクション・グループ (単数または複数) の関係は ACACTACTGP テーブルに保管されます。アクション・グループの例としては、 ″OrderWriteCommands″ アクション・グループがあります。このアクション・グループには、オーダーの更新に使用される以下のアクションが組み込まれています。

- com.ibm.commerce.order.commands.OrderDeleteCmd
- com.ibm.commerce.order.commands.OrderCancelCmd
- com.ibm.commerce.order.commands.OrderProfileUpateCmd
- com.ibm.commerce.order.commands.OrderUnlockCmd

- com.ibm.commerce.order.commands.OrderScheduleCmd
- com.ibm.commerce.order.commands.ScheduledOrderCancelCmd
- com.ibm.commerce.order.commands.ScheduledOrderProcessCmd
- com.ibm.commerce.order.commands.OrderItemAddCmd
- com.ibm.commerce.order.commands.OrderItemDeleteCmd
- com.ibm.commerce.order.commands.OrderItemUpdateCmd
- com.ibm.commerce.order.commands.PayResetPMCmd

リソース・グループは、特定のタイプのリソースをグループ化するメカニズムです。リソース・グループ内のリソースのメンバーシップは、次の 2 つの方法のいずれかで指定できます。

- ACRESGRP テーブルの条件列を使用する
- ACRESGPRES テーブルを使用する

ほとんどの場合、リソースをリソース・グループに関連付けるには、ACRESGPRES テーブルを使用すれば十分です。この方法を使用すると、リソースは Java クラス名を使用して ACRESCGRY テーブルで定義されます。次に、これらのリソースは、ACRESGPRES アソシエーション・テーブルを使用して、適切なリソース・グループ (ACRESGRP テーブル) に関連付けられます。 Java クラス名だけではリソース・グループのメンバーを定義するのに十分でない場合 (たとえば、リソースの属性に基づいてこのクラスのオブジェクトをさらに限定する必要がある場合)、 ACRESGRP テーブルの条件列を使用してリソース・グループをすべて定義することができます。属性に基づいてこのようなリソースのグループ化を実行するには、リソースの Groupable インターフェースをインプリメントしていなければならないことに注意してください。

以下の図は、リソースのグループ化を指定する例を示しています。この例で、リソース・グループ 10023 には、 ACRESGPRES テーブル内でそれと関連付けられるすべてのリソースが組み込まれています。リソース・グループ 10070 は、ACRESGRP テーブルの条件フィールド列を使用して定義されています。このリソース・グループには Order リモート・インターフェースのインスタンスが組み込まれますが、それらのインスタンスも status = "Z" (共有要求リストを指定する) の状態になっています。

注: ACRESGRP テーブルの条件列に関する XML 情報の詳細については、*WebSphere アクセス・コントロール・ガイド* で説明されています。

ACRESGRP

AcResGrp_Id	GrpName	条件
10023	AccountRepresentatives CmdResourceGroup	ヌル
10070	SharedRequisitionList ResourceGroup	<profile> <andListCondition> <simpleCondition> <variable name="Status"/> <operator name="="/> <value data="Z"/> </simpleCondition> <simpleCondition> <variable name="classname"/> <operator name="="/> <value data="com.ibm.commerce.order.objects.Order"/> </simpleCondition> </andListCondition> </profile>

ACRESGRPES

AcResGrp_Id	AcResCgry_Id
10023	10246
10023	10247
10023	10248
10023	10249
10023	10250

ACRESCGRY

AcResCgry_Id	ResClassname
10246	com.ibm.commerce.contract.commands.ContractCreateCmd
10247	com.ibm.commerce.contract.commands.ContractCreateCmd
10248	com.ibm.commerce.contract.commands.ContractCreateCmd
10249	com.ibm.commerce.contract.commands.ContractCreateCmd
10250	com.ibm.commerce.contract.commands.ContractCreateCmd

図 4.

ACACTGRP、ACRESGRP、および ACRELGRP テーブルの MEMBER_ID 列の値は、 -2001 (ルート組織) でなければなりません。

アクセス・コントロール・ポリシーには、任意で 4 番目のエレメントとして Relationship または RelationshipGroup エレメントを含めることもできます。

アクセス・コントロール・ポリシーが Relationship エレメントを使用する場合、それは ACRELATION テーブルからとられます。一方、RelationshipGroup エレメントが含まれる場合、それは ACRELGRP テーブルからとられます。どちらも含める必要はありませんが、一方を含める場合には、他方を含めることはできないことに注意してください。 ACRELGRP からとられる RelationshipGroup 仕様は、ACRELATION テーブルからとられる Relationship より優先されます。

ACRELATION テーブルは、ユーザーとリソースの間に存在する関係のタイプを指定します。関係のタイプの例として、作成者、送信者、および所有者があります。

relationship エレメントの使用例には、このエレメントを使用して、オーダーの作成者が常にオーダーを更新できるようにしておくことなどがあります。

ACRELGRP テーブルは、特定のリソースと関連付けることができる関係グループのタイプを指定します。関係グループは、1 つ以上の関係チェーンをグループ化したものです。関係チェーンとは、1 つ以上の関係の系列です。関係グループの例として、ユーザーがリソースの作成者でなければならないこと、さらにリソースで参照される購買組織エンティティーに属していなければならないことを指定することがあります。

関係グループ (または関係) の指定は、アクセス・コントロール・ポリシーの任意の部分です。これは、独自のコマンドを作成しており、それらのコマンドが特定の役割に限定されていない場合に、共通して使用されます。そのような場合、ユーザーとリソースの関係を規定することができます。通常、コマンドを特定の役割用に限定するには、 Relationship エレメントを使用するのではなく、アクセス・コントロール・ポリシーの UserGroup エレメントを使用します。

アクセス・コントロール・ポリシーに関連するもう 1 つの重要な概念に、アクセス・コントロール・ポリシー*所有者* の概念があります。アクセス・コントロール・ポリシー所有者は、アクセス・コントロール・ポリシーを所有する、組織のエンティティーです。アクセス・コントロール・ポリシーは、アクセス・コントロール・ポリシー所有者が所有するリソースにしか適用できないため、アクセス・コントロール・ポリシー所有者を認識することが重要です。

問題となっているリソースごとに、アクセス・コントロール・ポリシー・マネージャーが、メンバー階層内の所有組織エンティティーまたはその上位の組織エンティティーによって所有されるアクセス・コントロール・ポリシーを適用します。これは、許可を与えるポリシーが見つかるか、またはすべてのポリシーが検査されてどれからも許可が得られないことが分かるまで続けられます。

メンバー階層を示す以下の図について考えてみます。

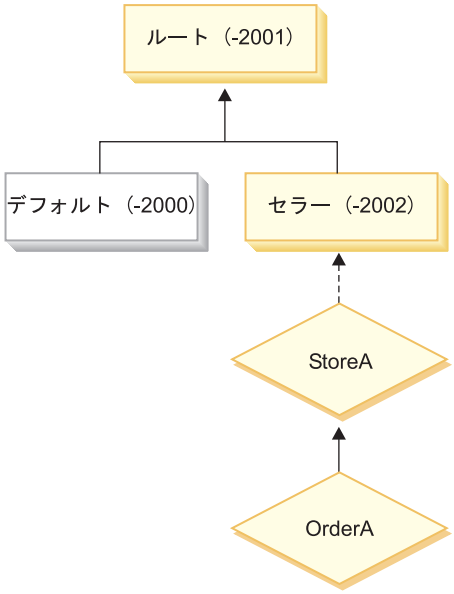


図 5.

リソース “OrderA” について、セラー組織またはルート組織が所有するすべてのアクセス・コントロール・ポリシーを適用できます。アクセス・コントロール・ポリシー・マネージャーは、これらの組織のどちらかによって所有された、(アクセス・コントロール・ポリシー内の 4 つのエレメントに基づいた) ユーザー許可を認可するポリシーを 1 つ検出した時点で、アクセス・コントロール・ポリシーの検索を即時に停止します。しかし、これらの組織によって所有されたすべてのアクセス・コントロール・ポリシーの中で、保護リソースでアクションを実行するためのユーザー許可を与えるものを検出できない場合は、アクセスは拒否されます。

関係グループ

関係グループを使用して、複数の関係を指定できます。関係は、ユーザーとリソースの間に直接指定することもできますし、ユーザーをリソースに間接的に関連付ける関係のチェーンにすることもできます。

注: 関係グループに関連したこの後の項では、 WebSphere Commerce Professional Edition で使用可能な組織は RootOrganization、DefaultOrganization、および SellerOrganization だけであることを認識することが重要です。他の組織を参照している例は、WebSphere Commerce Business Edition にだけ当てはまります。

関係と関係グループの比較: アクセス・コントロール・ポリシーでは、アクセスしているリソースについてユーザーが特定の関係を実現しなければならないことを指定できます。または、ユーザーが関係グループ内で指定されている条件を実現しなければならないことを指定できます。

たいていの場合、関係を指定すれば、アプリケーションのアクセス・コントロール要件を満たすことになるはずです。しかし、ユーザーとリソースが直接結び付いていない関係を指定することを規定したポリシーの場合に、実際にはそのユーザーとリソースの間に一連の関係があれば、リソース・グループを使用する必要があります。

たとえば、ユーザーと購買組織の間のアソシエーションを指定しなければならない場合に、その関係においてユーザーがその組織の特定の役割を果たしていること、またはユーザーが購買組織のメンバーであることが必要であれば、関係グループおよび関係のチェーンを使用しなければなりません。

該当するユーザーとリソースを直接結び付けるアソシエーションを規定すればよいだけの場合には、単純な関係を使用することができます。たとえば、ユーザーがリソースの作成者でなければならないことを規定する必要がある場合などがそれにあたります。

複数の単純な関係を結合する場合、たとえばユーザーが作成者または 送信者でなければならない場合には、これが関係のチェーンになり、関係グループを作成する必要があります。こうした単純な関係の結合は、WebSphere Commerce Professional Edition または WebSphere Commerce Business Edition を使用する際に生じることがあります。

関係グループに関する一般情報: 関係チェーンとは、1 つ以上の関係の系列です。関係チェーンの長さは、そこに含まれる関係の数によって決定されます。これを決定するには、関係チェーンの XML 表記の `<parameter name="aName" value="aValue" />` エントリーの数を調べます。

最後の `<parameter name="Relationship" value="aValue" />` エレメントだけがリソースの fulfills() メソッドによって処理されなければなりません。残りはアクセス・コントロール・ポリシー・マネージャーによって内部的に処理されます。

関係チェーンの長さが 2 の場合、最初の `<parameter name="aName" value="aValue" />` エレメントはユーザーと組織エンティティーの間にあります。最後の `<parameter name="aName" value="aValue" />` エレメントは組織エンティティーとリソースの間にあります。

関係グループを定義する必要がある場合、 XML ファイル内で関係グループ情報を定義することによってそれを行わなければなりません。`defaultAccessControlPolicies.xml` ファイルを変更するか、または独自の XML ファイルを作成することができます。このような XML ベースの情報を作成することについての詳細は、 *WebSphere Commerce アクセス・コントロール・ガイド* を参照してください。

以下のセクションでは、様々なタイプの関係グループの例を示します。

単一の関係チェーンで構成される関係グループ: Business アクセス・コントロール・ポリシーの一部として、ユーザーが組織エンティティー (リソースの BuyingOrganizationalEntity) に属していることを義務付ける必要がある場合があります。ここでは、長さが 2 である 1 つの関係チェーンで構成される関係グループを作成する必要があります。関係チェーンの長さが ″2″ と言えるのは、それが 2 つの別個の関係で成り立っているからです。最初の関係はユーザーとその親組織エンティティーの間にあります。その関係ではユーザーは「子」になります。 2 番目の関係の場合、アクセス・コントロール・ポリシー・マネージャーは、親組織エンティティーとリソースとの間の BuyingOrganizationalEntity 関係が成立しているかどうか調べます。言い換えれば、それがリソースの購買組織エンティティーである場合は、″true″ を戻します。

以下の XML 断片は defaultAccessControlPolicies.xml ファイルからとられており、このタイプの関係グループを定義する方法を示しています。

```
<RelationGroup Name="MemberOf->BuyerOrganizationalEntity"
      OwnerID="RootOrganization">
   <RelationCondition><![CDATA[
      <profile>
         <openCondition name="RELATIONSHIP_CHAIN">
            <parameter name="HIERARCHY" value="child"/>
            <parameter name="RELATIONSHIP" value="BuyingOrganizationalEntity"/>
         </openCondition>
      </profile>
   ]]></RelationCondition>
</RelationGroup>
```

Business 別の例として、ユーザーがリソースの購買組織エンティティーである組織エンティティーのアカウント担当者の役割を持たなければならないことを義務付けます。ここでも、長さが 2 である 1 つの関係チェーンで構成される関係グループを使用します。チェーンの最初の部分ではユーザーがアカウント担当者の役割を持っているすべての組織エンティティーを見付けます。このような組織エンティティーのセットの場合、アクセス・コントロール・ポリシー・マネージャーは、それらの組織エンティティーの少なくとも 1 つとリソースとの間で BuyingOrganizationalEntity 関係が成立しているかどうか調べます。言い換えれば、それがリソースの購買組織エンティティーである場合は、true を戻します。

以下の XML 断片は defaultAccessControlPolicies.xml ファイルからとられており、このタイプの関係グループを定義する方法を示しています。

```
<RelationGroup Name="AccountRep->BuyerOrganizationalEntity"
      OwnerID="RootOrganization">
   <RelationCondition><![CDATA[
      <profile>
         <openCondition name="RELATIONSHIP_CHAIN">
            <parameter name="ROLE" value="Account Representative"/>
            <parameter name="RELATIONSHIP" value="BuyingOrganizationalEntity"/>
         </openCondition>
      </profile>
   ]]></RelationCondition>
</RelationGroup>
```

複数の関係チェーンで構成される関係グループ: 関係グループに複数の関係チェーンが含まれるように、関係グループを構成することができます。これを行う際に、ユーザーがすべての関係チェーンを満たしていなければならないかどうか (つまり、それが *AND* シナリオであるか)、またはユーザーが少なくとも 1 つの関係チェーンを満たしていればよいか (つまり、それが *OR* シナリオであるか) を指定する必要があります。

Business このタイプの関係を示すために、以下の XML 断片が使用されます。ここでは、ユーザーがリソースの作成者でなければならないこと、さらにリソースで指定された BuyingOrganizationalEntity に属していなければならないことを強制します。最初のチェーンではユーザーがリソースの作成者でなければならず、そのチェーンの長さは 1 です。 2 番目のチェーンではユーザーがリソースで指定された BuyingOrganizationalEntity に属していなければならず、そのチェーンの長さは 2 です。

```
<RelationGroup Name="Creator_And_MemberOf->BuyerOrganizationalEntity"
      OwnerID="RootOrganization">
 <RelationCondition><![CDATA[
  <profile>
   <andListCondition>
    <openCondition name="RELATIONSHIP_CHAIN">
     <parameter name="RELATIONSHIP" value="creator" />
    </openCondition>
    <openCondition name="RELATIONSHIP_CHAIN">
     <parameter name="HIERARCHY" value="child"/>
     <parameter name="RELATIONSHIP" value="BuyingOrganizationalEntity"/>
    </openCondition>
   </andListCondition>
  </profile>
 ]]></RelationCondition>
</RelationGroup>
```

AND シナリオを使用するのではなく、ユーザーが 2 つの関係チェーンのどちらかを満たすように求める場合には、 <andListCondition> タグを <orListCondition> タグに変更します。

Professional Business WebSphere Commerce Professional Edition (WebSphere Commerce Business Edition と同様) で使用できる関係グループを示すために、ユーザーがリソースの作成者か送信者のどちらかでなければならないことを規定するのに使われる関係グループについて考えます。これは以下の XML 断片に示されています。

```
<RelationGroup Name="Creator_Or_Submitter"
     OwnerID="RootOrganization">
   <RelationCondition><![CDATA [
   <profile>
      <orListCondition>
         <openCondition name="RELATIONSHIP_CHAIN">
            <parameter name="RELATIONSHIP"value="creator"/>
         </openCondition>
         <openCondition name="RELATIONSHIP_CHAIN">
            <parameter name="RELATIONSHIP"value="submitter"/>
         </openCondition>
      </orListCondition>
   </profile>
   ]]></RelationCondition>
</RelationGroup>
```

アクセス・コントロールのタイプ

アクセス・コントロールのタイプには 2 つあります。コマンド・レベルのアクセス・コントロールとリソース・レベルのアクセス・コントロールで、両方ともポリシーに基づいています。

コマンド・レベル (「役割ベース」としても知られている) のアクセス・コントロールは、幅広いタイプのポリシーを使用します。特定の役割をもつすべてのユーザーが、あるタイプのコマンドを実行できるように指定できます。たとえば、アカウント担当者の役割をもつユーザーが、 AccountRepresentativesCmdResourceGroup リソース・グループで任意のコマンドを実行できるように指定できます。あるいは以下の図で示すように、別のポリシーの例では、すべてのストア管理者が、StoreAdminCmdResourceGrp によって指定される任意のリソースの ExecuteCommandAction グループで、指定された任意のアクションを実行できるように指定できます。

注: MBRGRPCOND テーブルの条件列に関する XML 情報は、管理コンソールを使用してアクセス・グループをセットアップする際に生成されます。管理コンソールを使用してアクセス・グループをセットアップすることについての詳細は、 WebSphere Commerce のオンライン・ヘルプを参照してください。

ACPOLICY

PolicyName	Member_Id	MbrGrp_Id	AcActGrp_id	AcResGrp_Id	AcRelGrp_Id
StoreAdministrators ExecuteStoreAdmin CmdResourceGroup	-2001	-8	10052	10018	ヌル

MBRGRP

MbrGrp_Id	MbrGrpName
-8	StoreAdministrators

MBRGRPCOND

MbrGrp_Id	条件
-8	<profile> <simpleCondition> <variable name="role"/> <operator name="="/> <value data="Store Administrator"/> </simpleCondition> </profile>

ACACTGRP

AcActGrp_Id	GroupName
10052	ExecuteCommandActionGroup

ACRESGRP

AcResGrp_Id	GrpName
10018	StoreAdminCmdResourceGroup

図 6.

コマンド・レベルのアクセス・コントロール・ポリシーは、コントローラー・コマンドのアクション・グループとして常に `ExecuteCommandActionGroup` を持ちます。ビューについては、リソース・グループは常に `ViewCommandResourceGroup` です。

すべてのコントローラー・コマンドは、コマンド・レベルのアクセス・コントロールによって保護されなければなりません。さらに、直接呼び出せるビュー、または別のコマンドからリダイレクトに起動できる (ビューへの転送によって起動される場合とは対照的に) ビューはすべて、コマンド・レベルのアクセス・コントロールによって保護されなければなりません。

コマンド・レベルのアクセス・コントロールは、コマンドが影響を及ぼすリソースのことを考えません。単に、ユーザーが特定のコマンドを実行できるかどうかを判別するだけです。ユーザーが特定のコマンドを実行できる場合は、ユーザーが問題

のリソースにアクセスできるかどうかを判別するために、後続のリソース・レベルのアクセス・コントロール・ポリシーが適用されます。

ストア管理者が管理用タスクの実行を試みる際のことを考えてみてください。アクセス・コントロール検査の第 1 レベルでは、このユーザーが特定のストア管理コマンドの実行を許可されているかどうかを決定します。ユーザーがこれについて実際に (ストア管理者は storeAdminCmds グループでのコマンドの実行を許可されるので) 許可されていると判別されたら、リソース・レベルのアクセス・コントロール・ポリシーが呼び出されます。このポリシーには、自身がストア管理者とされている組織が所有するストアについてのみ、ストア管理者に実行が許可されていることが記述してあるかもしれません。

要約すると、コマンド・レベルのアクセス・コントロールでは、「リソース」がコマンドそのものであり、「アクション」は単にコマンドを実行する (言い換えれば、コマンド・オブジェクトをインスタンス化する) だけです。アクセス・コントロール検査では、ユーザーにコマンドの実行が許可されているかどうかを決定します。これに対し、リソース・レベルのアクセス・コントロールでは、「リソース」はコマンドまたは bean がアクセスする保護可能な任意のリソースであり、「アクション」はコマンドそのものです。

アクセス・コントロールの相互作用

このセクションでは、WebSphere Commerce アクセス・コントロール・ポリシーのフレームワークでアクセス・コントロールがどのように動作するかを説明する、相互作用の図を示します。

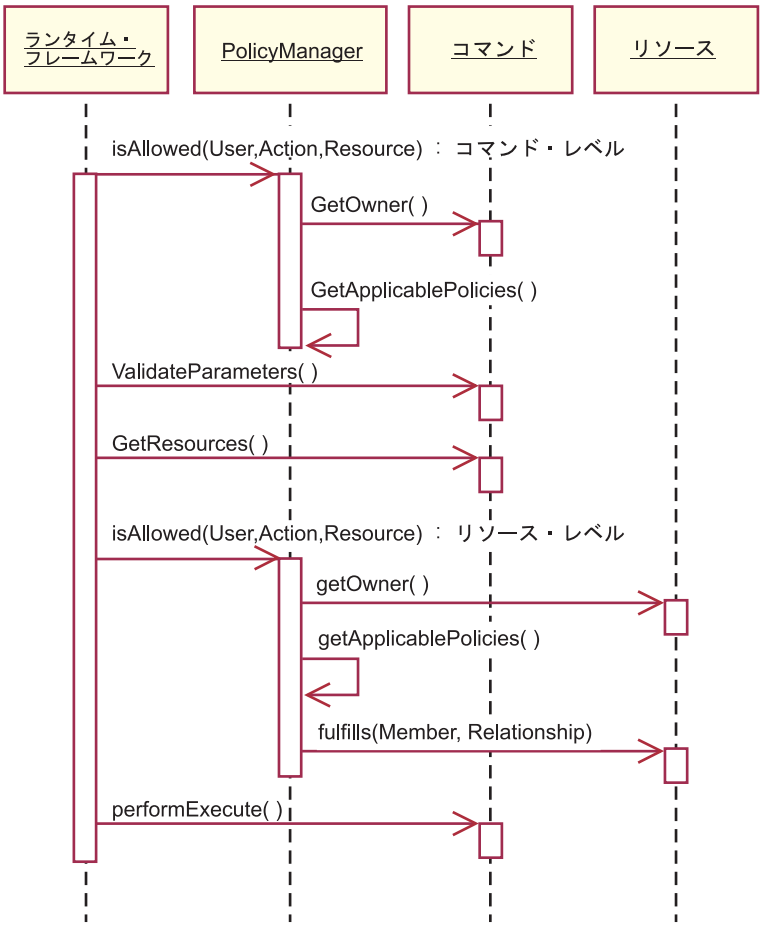


図 7.

上記の図は、アクセス・コントロール・ポリシー・マネージャー によって実行されるアクションを示しています。アクセス・コントロール・ポリシー・マネージャーは、現行ユーザーが指定されたリソースで指定されたアクションを実行することを許可されているかどうかを判別する、アクセス・コントロール・コンポーネントです。ポリシー・マネージャーは、リソースの所有者および上位の組織が所有するポリシーを検索して、これを決定します。少なくとも 1 つのポリシーがアクセスを認可していれば、許可が与えられます。

以下に、上記の対話の図のアクションを説明します。図の上から下という順序で並べてあります。

1. `isAllowed()`
   ユーザーが、コントローラー・コマンドまたはビューのどちらかにコマンド・レベル・アクセスできるかどうかをランタイム・コンポーネントが判断します。
2. `getOwner()`
   アクセス・コントロール・ポリシー・マネージャーが、コマンド・レベル・リソースの所有者を決定します。デフォルトのインプリメンテーションでは、コマンド・コンテキスト内にあるストア (`storeId`) の所有者のメンバー ID (`memberId`) が戻されます。コマンド・コンテキストにストア ID がない場合は、ルート組織 (`-2001`) が戻されます。

3. getApplicablePolicies()
   アクセス・コントロール・ポリシー・マネージャーが、指定されたユーザー、アクション、およびリソースに基づいて、適当なポリシーを検索して処理します。
4. validateParameters()
   初期のパラメーター検査および解決です。
5. getResources()
   リソースとアクションの組みのベクトルであるアクセス・ベクトルを戻します。

   何も戻されない場合は、リソース・レベルのアクセス・コントロール検査は行われていません。保護する必要のあるリソースがある場合は、 (リソースとアクションの組みからなる) アクセス・ベクトルが戻される必要があります。

   各リソース が保護可能なオブジェクト (com.ibm.commerce.security.Protectable インターフェースをインプリメントするオブジェクト) のインスタンスです。多くの場合、このリソースはアクセス Bean です。

   アクセス Bean は com.ibm.commerce.security.Protectable インターフェースをインプリメントしないことがありますが、 116 ページの 『エンタープライズ Bean でのアクセス・コントロールのインプリメント』の情報に従って対応するエンタープライズ Bean が保護される限り、アクセス・コントロール検査は発生します。

   アクション は、リソースで実行される操作を表すストリングです。ほとんどの場合、アクションはコマンドのインターフェース名です。
6. isAllowed()
   ユーザーが、 getResources() で指定されたすべてのリソース / アクションのペアにリソース・レベル・アクセスできるかどうかをランタイム・コンポーネントが判断します。
7. getOwner()
   リソースが、その所有者の memberId を戻します。これでどのポリシーが適用されるかが決定されます。リソース所有者と上位の組織によって所有されたポリシーだけが適用されます。
8. getApplicablePolicies()
   アクセス・コントロール・ポリシー・マネージャーが、適用できるポリシーを検索して、それを適用します。リソースとアクションの組みについて、ユーザーのリソースへのアクセス権を認可するポリシーが少なくとも 1 つ検出されればアクセスは認可されますが、検出されない場合はアクセスは拒否されます。
9. fulfills()
   適用可能なポリシーにおいて関係グループが指定されている場合、リソースに対して検査が行われて、そのリソースに関連した指定どおりの関係をメンバーが満たすかどうかが確かめられます。
10. performExecute()
    コマンドのビジネス・ロジックです。

保護可能なインターフェース

WebSphere Commerce アクセス・コントロール・ポリシーによってリソースを保護させるための重要な要素は、リソースが com.ibm.commerce.security.Protectable インターフェースをインプリメントしなければならないことです。このインターフェースはエンタープライズ Bean およびデータ Bean で最もよく使用されますが、保護を必要とするこれらの bean でのみこのインターフェースをインプリメントしなければなりません。

Protectable インターフェースでは、リソースは次の 2 つの鍵メソッドを提供しなければなりません。それは、getOwner() と fulfills(Long member, String relationship) です。

アクセス・コントロール・ポリシーは、組織または組織エンティティーによって所有されます。 getOwner メソッドは、保護可能なリソースの所有者の memberId を戻します。アクセス・コントロール・ポリシー・マネージャーは、リソースの所有者を判別してから、メンバー階層内の所有者の祖先それぞれの memberId も入手します。それから、オリジナルの getOwner 要求で戻された所有者に属する全アクセス・コントロール・ポリシーが、その所有者の任意の祖先に属する全アクセス・コントロール・ポリシーと同様に適用されます。

指定された所有者に適用されるアクセス・コントロール・ポリシーに加え、所有者のメンバーシップ階層内でより高位の祖先に適用されるアクセス・コントロール・ポリシーが適用されます。

指定されたメンバーがリソースの点で必要な関係を満たす場合は、 fulfills メソッドは true しか戻しません。一般にメンバーは単一のユーザーですが、組織であってもかまいません。アクセス・コントロール・ポリシーで関係グループを使用する場合、メンバーは組織になります。

グループ化可能なインターフェース

アクセス・コントロール・ポリシーのアプリケーションは、リソースのグループに固有のものです。リソースのグループ化は、クラス名、オーダーの状態または storeId 値などの属性に基づいて行われます。

アクセス・コントロール・ポリシーを適用する目的で、クラス名以外の属性によってリソースをグループ化する場合、 com.ibm.commerce.grouping.Groupable インターフェースをインプリメントしなければなりません。

以下のコードの断片は Groupable インターフェースを表しています。

```
Groupable interface {
Object getGroupingAttributeValue (String attributeName, GroupContext context)
}
```

たとえば、保留状態 (status = P (保留)) になっているオーダーにしか適用されないポリシーをインプリメントするには、 Order エンティティーのリモート・インターフェースが Groupable インターフェースをサポートし、 attributeName の値が "status" に設定されます。

Groupable インターフェースを使用することはめったにありません。

アクセス・コントロールについての情報の入手先

WebSphere Commerce アクセス・コントロール・モデルについての詳細は、*WebSphere Commerce アクセス・コントロール・ガイド* を参照してください。この資料では、アクセス・コントロールの概要について詳細に説明し、管理コンソールを、ポリシー、アクション・グループ、およびリソース・グループの作成または変更に使用する方法について説明します。

アクセス・コントロールのインプリメント

ここでは、カスタマイズ・コードのアクセス・コントロールをインプリメントする方法について説明します。

保護可能なリソースの識別

一般に、保護が必要なリソースはエンタープライズ Bean およびデータ Bean です。しかし、エンタープライズ Bean とデータ Bean すべてを保護するべきではありません。既存の WebSphere Commerce アプリケーションでは、保護が必要なリソースはすでに保護可能なインターフェースをインプリメントしています。何を保護すべきかという問題は、新規にエンタープライズ Bean およびデータ Bean を作成するときに発生します。どのリソースを保護するかは、アプリケーションによって異なります。

コマンドが `getResources` メソッドでエンタープライズ Bean を戻す場合は、エンタープライズ Bean を保護しなければなりません。これは、アクセス・コントロール・ポリシー・マネージャーがエンタープライズ Bean で getOwner メソッドを呼び出すためです。対応するリソース・レベルのアクセス・コントロール・ポリシーで関係が指定されている場合は、`fulfills` メソッドも呼び出されます。

ユーザー自身のエンタープライズ Bean およびデータ Bean すべてについて、保護可能なインターフェースをインプリメントしようとする場合 (したがって、リソースを保護下に置きたい場合) は、アプリケーションに多数のポリシーが必要です。ポリシーの数が増えると、パフォーマンスが悪くなる場合があり、ポリシー管理が難しくなります。

1 次リソースと従属リソースには理論上の違いがあります。 *1 次リソース* は自分だけで存在できます。*従属リソース* は、1 次リソースの存在に関係するときにのみ存在します。たとえば、WebSphere Commerce アプリケーション・コードでは、Order エンティティー Bean は保護可能なリソースですが、 OrderItem エンティティー Bean は違います。この理由は、OrderItem の存在が Order に依存していることにあります。 Order は 1 次リソースで、OrderItem は従属リソースということです。もし、ユーザーが Order にアクセスできるなら、 Order 内のアイテム (OrderItem) にもアクセスできることになります。

同様に、User エンティティー Bean は保護可能なリソースですが、Address エンティティー Bean は違います。この場合は、アドレスの存在がユーザーに依存しているため、ユーザーにアクセスできるものはすべてアドレスへのアクセスもできることになります。

1 次リソースは保護しなければなりませんが、従属リソースには保護が不要である場合がよくあります。ユーザーが 1 次リソースへのアクセスを許可される場合、デフォルトでユーザーがその従属リソースへのアクセスも許可されていると理解できます。

エンタープライズ Bean でのアクセス・コントロールのインプリメント

アクセス・コントロール・ポリシーによる保護が必要なエンタープライズ Bean を新規作成する場合は、以下を行ってください。

1. 新規のエンタープライズ Bean を作成し、それが `com.ibm.commerce.base.objects.ECEntityBean` から拡張していることを確認します。
2. その bean のリモート・インターフェースが `com.ibm.commerce.security.Protectable` インターフェースを拡張することを確認します。
3. bean が相互作用するリソースが、リソースの Java クラス名以外の属性によってグループ化される場合は、 bean のリモート・インターフェースも `com.ibm.commerce.grouping.Groupable` インターフェースを拡張しなければなりません。
4. エンタープライズ Bean クラスには、以下のメソッドについてのデフォルトのインプリメンテーションが含まれます。
   - `getOwner`
   - `fulfills`
   - `getGroupingAttributeValue`

   必要に応じてメソッドをオーバーライドします。 `getOwner` メソッドは必ずオーバーライドしてください。
   これらのメソッドのデフォルトのインプリメンテーションを以下のコードの断片に示します。

```
*****************************************************************************
public Long getOwner() throws Exception, java.rmi.RemoteException
{
   return null;
}
*****************************************************************************
*****************************************************************************
public boolean fulfills(Long member, String relationship)
  throws Exception, java.rmi.RemoteException
{
      return false;
}
*****************************************************************************
*****************************************************************************
public Object getGroupingAttributeValue(String attributeName,
   GroupingContext context) throws Exception, java.rmi.RemoteException
{
   return null;
}
*****************************************************************************
```

   以下に、OrderBean bean に基づくこれらのメソッドのサンプル・インプリメンテーションを示します。

```
*****************************************************************************
public Long getOwner() throws Exception, java.rmi.RemoteException
{
   com.ibm.commerce.common.objects.StoreEntityAccessBean storeEntAB = new
   com.ibm.commerce.common.objects.StoreEntityAccessBean();
   storeEntAB.setInitKey_storeEntityId(getStoreEntityId().toString());
   return storeEntAB.getMemberIdInEJBType();
}
*****************************************************************************
*****************************************************************************
public boolean fulfills(Long member, String relationship)
  throws Exception, java.rmi.RemoteException
   {
      if (relationship.equalsIgnoreCase("creator"))
      {
         return member.equals(getMemberId());
      }
      else if (relationship.equalsIgnoreCase (
         com.ibm.commerce.base.helpers.EJBConstants.
         SAME_ORGANIZATIONAL_ENTITY_AS_CREATOR_RELATION)) {
            com.ibm.commerce.user.objects.UserAccessBean creator = new
               com.ibm.commerce.user.objects.UserAccessBean();
            creator.setInitKey_MemberId(getMemberId().toString());
            com.ibm.commerce.user.objects.UserAccessBean ab = new
               com.ibm.commerce.user.objects.UserAccessBean();
            ab.setInitKey_MemberId(member.toString());
            if (ab.getParentMemberId().equals(creator.getParentMemberId()))
               return true;
          }
      return false;
}
*****************************************************************************
*****************************************************************************
public Object getGroupingAttributeValue(String attributeName,
   GroupingContext context) throws Exception
   {
      if (attributeName.equalsIgnoreCase("Status"))
         return getStatus();
   return null;
   }
*****************************************************************************
```

5. エンタープライズ Bean のアクセス Bean および生成コードを作成 (または再作成) します。

データ Bean でのアクセス・コントロールのインプリメント

データ Bean は、アクセス・コントロール・ポリシーによって直接または間接的に保護できます。データ Bean が直接保護される場合は、その特定のデータ Bean に適用されるアクセス・コントロール・ポリシーが存在します。データ Bean が間接的に保護される場合は、アクセス・コントロール・ポリシーが存在する別のデータ Bean に保護を代行させています。

アクセス・コントロール・ポリシーに直接保護される新規のデータ Bean を作成する場合は、データ Bean について以下のことを行わなければなりません。

1. `com.ibm.commerce.security.Protectable` インターフェースをインプリメントします。これにより、bean は `getOwner()` および `fulfills(Long member, String relationship)` メソッドをインプリメントする必要があります。これらは、bean のリモート・インターフェースでインプリメントしてください。

   データ Bean が Protectable インターフェースをインプリメントする際は、データ Bean マネージャーが `isAllowed` メソッドを呼び出して、現在のアクセス・

コントロール・ポリシーに従って、ユーザーに適切なアクセス・コントロール権限があるかどうかを決定します。 isAllowed メソッドは以下のコードの断片によって記述されます。

```
IsAllowed(Context, "Display", protectable_databean);
```

2. bean が相互作用するリソースが、リソースの Java クラス名以外の属性によってグループ化される場合は、bean が com.ibm.commerce.grouping.Groupable インターフェースをインプリメントしなければなりません。
3. com.ibm.commerce.security.Delegator インターフェースをインプリメントします。このインターフェースは以下のコードの断片によって記述されます。

```
Interface Delegator {
Protectable getDelegate();
}
```

注: 直接保護されるためには、 getDelegate メソッドがデータ Bean そのものを戻さなければなりません (つまり、データ Bean がアクセス・コントロールの目的で自分に代行させます)。

直接保護されるべきデータ Bean と間接的に保護されるべきデータ Bean との違いは、 1 次リソースと従属リソースとの違いに似ています。データ Bean オブジェクトが独自に存在できる場合は、直接保護される必要があります。データ Bean が存在するかどうかが、別のデータ Bean の存在によって決まる場合は、他のデータ Bean にゆだねて保護を行わなければなりません。

直接保護されるデータ Bean の例としては、Order データ Bean があります。間接的に保護されるデータ Bean の例としては、OrderItem データ Bean があります。

アクセス・コントロール・ポリシーに間接的に保護される新規のデータ Bean を作成する場合は、データ Bean について以下のことを行わなければなりません。

1. com.ibm.commerce.security.Delegator インターフェースをインプリメントします。このインターフェースは以下のコードの断片によって記述されます。

```
Interface Delegator {
Protectable getDelegate();
}
```

注: getDelegate データ Bean には、 Protectable インターフェースをインプリメントしなければなりません。

データ Bean が Delegator インターフェースをインプリメントしない場合は、アクセス・コントロール・ポリシーの保護なしで取り込まれます。

コントローラー・コマンドでのアクセス・コントロールのインプリメント

新規のコントローラー・コマンドを作成すると、新規コマンドのインプリメンテーション・クラスは com.ibm.commerce.commands.ControllerCommandImpl クラスを拡張し、そのインターフェースは com.ibm.commerce.command.ControllerCommand インターフェースを拡張するはずです。

コントローラー・コマンドのコマンド・レベル・ポリシーの場合、そのコマンドのインターフェース名をリソースとして指定します。リソースが保護されるようにす

るには、保護可能なインターフェースをインプリメントする必要があります。WebSphere Commerce プログラミング・モデルによれば、そのインプリメントの実現のためには、コマンドのインターフェースを com.ibm.commerce.command.ControllerCommand インターフェースから拡張し、コマンドのインプリメンテーションを com.ibm.commerce.commands.ControllerCommandImpl から拡張します。ControllerCommand インターフェースが com.ibm.commerce.command.AccCommand インターフェースに拡張された後、後者のインターフェースが Protectable を拡張します。コマンド・レベル・レベルのアクセス・コントロールによる保護を受けるためには、 AccCommand が、コマンドがインプリメントする必要のある最低限のインターフェースです。

コマンドが保護されるべきリソースにアクセスする場合は、 AccessVector タイプの専用インスタンス変数を作成して、リソースを保持します。それから、このメソッドのデフォルトのインプリメンテーションがヌル値を戻してから getResources メソッドをオーバーライドしてください。リソース検査は起こりません。

新規の getResources メソッドでは、コマンドが動作できるリソースの配列またはリソースとアクションの組みの配列を戻してください。アクションが明示的に指定されない場合、アクションのデフォルトは実行されるコマンドのインターフェース名になります。

さらに、メソッドがリソースをインスタンス化しなければならないのかどうか、またはリソースへの参照を持つ既存のインスタンス変数を使用できるかどうかを、メソッドが決定することをお勧めします。リソース・オブジェクトがすでに存在するかどうかを検査すると、システム・パフォーマンスが向上する可能性があります。必要に応じて、新規のコントローラー・コマンドの performExecute メソッドで、同じ getResources メソッドを使用できます。

以下は、getResources メソッドの例です。

```
private AccessVector resources = null;

public AccessVector getResources() throws ECException {

   if (resources == null) {
      OrderAccessBean orderAB = new OrderAccessBean();
      orderAB.setInitKey_orderId(getOrderId().toString());
      resouces = new AccessVector(orderAB);
      }
   return resources;
}
```

例として OrderItemUpdate コマンドについて考えてみます。このコマンドの getResources メソッドは、Order および User という保護可能なオブジェクトを戻します。アクションは指定されないので、デフォルトは OrderItemUpdate コマンドのインターフェースになります。

getResources メソッドによって複数のリソースが戻されることがあります。このような場合にアクションが実行されるためには、指定されたすべてのリソースについてユーザーにアクセスを許可するポリシーが検出されなければなりません。ユーザーが 3 つのリソースのうち 2 つについてアクセスを持っていてもアクションは進みません (3 つのうち 3 つが必要です)。

コントローラー・コマンドでさらにパラメーター検査またはパラメーターの解決を行う必要がある場合は、 validateParameters() メソッドを使用できます。これはオプションです。

追加のリソース・レベル検査

コントローラー・コマンドの getResources メソッドが呼び出されるときに、保護が必要なすべてのリソースを常に判別することができるとは限りません。

必要であれば、タスク・コマンドでも getResources メソッドをインプリメントして、コマンドを実行できるリソースのリストを戻すことができます。

リソース・レベル検査を呼び出すには、 checkIsAllowed(Object resource, String action) メソッドを使用して、アクセス・コントロール・ポリシー・マネージャーを直接呼び出す方法もあります。このメソッドは、com.ibm.commerce.command.AbstractECTargetableCommand クラスから拡張されたすべてのクラスで使用することができます。たとえば、以下のクラスは AbstractECTargetableCommand クラスから拡張しています。

- com.ibm.commerce.command.ControllerCommandImpl
- com.ibm.commerce.command.DataBeanCommandImpl

checkIsAllowed メソッドも、 com.ibm.commerce.command.AbstractECCommand クラスを拡張するクラスで使用することができます。たとえば、以下のクラスは AbstractECCommand クラスから拡張しています。

- com.ibm.commerce.command.TaskCommandImpl

以下は、checkIsAllowed メソッドのシグニチャーを示しています。

```
void checkIsAllowed(Object resource, String action)
   throws ECException
```

このメソッドは、指定のリソースに対して指定のアクションを実行する許可が現在のユーザーに与えられていない場合に ECApplicationException をスローします。アクセスが認可された場合は、このメソッドは単に戻るだけです。

「作成」コマンドのアクセス・コントロール

getResources メソッドはコマンド内で performExecute メソッドの前に呼び出されるので、まだ作成されていないリソースについては異なる方法のアクセス・コントロールが必要です。たとえば、WidgetAddCmd がある場合、getResources メソッドはこれから作成されようとしているリソースを戻すことはできません。この場合、getResources メソッドはリソースの作成者を戻すはずです。たとえば、コマンドはコマンド・ファクトリーによって作成され、オーダーはストア内で作成され、ユーザーは組織内で作成されます。

コマンド・レベルのアクセス・コントロールのデフォルトのインプリメンテーション

コマンド・レベルのアクセス・コントロールでは、storeId が指定されていれば、getOwner() メソッドのデフォルトのインプリメンテーションがストア所有者の memberId を戻します。 storeId が指定されていない場合は、ルート組織の memberId が戻されます (memberId = -2001)。

getResources() メソッドのデフォルトのインプリメンテーションは null を戻します。

validateParameters() のデフォルトのインプリメンテーションは何も行いません。

ビューでのアクセス・コントロール・ポリシーのインプリメント

ビューについてのリソース・レベルのアクセス・コントロールは、データ Bean マネージャーによって実行されます。データ Bean マネージャーは以下の場合に呼び出されます。

1. JSP テンプレートに <useBean> タグが組み込まれていて、データ Bean が属性リストにない場合。
2. JSP テンプレートに以下の activate メソッドが組み込まれている場合。

```
DataBeanManager.activate(xyzDatabean, request);
```

注: (直接または間接的に) 保護されるデータ Bean は、 Delegator インターフェースをインプリメントしなければなりません。直接保護されるデータ Bean は自分に代行させるため、 Protectable インターフェースのインプリメントする必要があります。間接的に保護されるデータ Bean は、Protectable インプリメントをインターフェースしたデータ Bean に代行させる必要があります。

これは推奨されませんが、以下の場合にアクセス・コントロール検査のバイパスが発生します。

1. JSP テンプレートがデータ Bean を使用せずに、アクセス Bean を直接呼び出す場合。
2. JSP テンプレートがデータ Bean の populate() メソッドを直接呼び出す場合。

コントローラー・コマンドの結果が (ForwardViewCommand を使用して) ビューに転送される場合、コマンド・レベルのアクセス・コントロールはビューでは実行されません。さらに、コントローラー・コマンドが、(ビューで使用される) 取り込まれたデータ Bean を応答プロパティーの属性リストに置いてからビューに転送する場合、 JSP テンプレートは、データ Bean マネージャーを介さずにデータにアクセスできます。これには、 <useBean> タグが JSP テンプレートで使用されている必要があります。これによって、ユーザーがコントローラー・コマンドを介してすでにアクセスを認可されているリソース (データ Bean) について、重複するリソース・レベルのアクセス・コントロール検査をすべてバイパスできるので、 JSP テンプレートをより効率的にすることができます。

第 5 部 付録

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものであり、米国以外の国においては本書で述べる製品、サービス、またはプログラムを提供しない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品、プログラムまたはサービスの操作性の評価および検証は、お客様の責任で行っていただきます。

本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。 IBM 製品、プログラムまたはサービスに代えて、 IBM の知的所有権を侵害することのない機能的に同等のプログラムまたは製品を使用することができます。ただし、IBM によって明示的に指定されたものを除き、他社の製品と組み合わせた場合の動作の評価と検証はお客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む。) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権の許諾については、下記の宛先に書面にてご照会ください。

〒106-0032 東京都港区六本木 3-2-31
IBM World Trade Asia Corporation
Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。

IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態で提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。 国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

本書は定期的に見直され、必要な変更 (たとえば、技術的に不適切な表現や誤植など) は、本書の次版に組み込まれます。 IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム（本プログラムを含む）との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Canada Ltd.
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
Canada

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、 IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性がありますが、その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確証できません。 IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があり、単に目標を示しているものです。

本書はプランニング目的としてのみ記述されています。記述内容は製品が使用可能になる前に変更になる場合があります。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

この製品で使用されているクレジット・カードのイメージ、商標、商号は、そのクレジット・カードを利用して支払うことを、それら商標等の所有者によって許可された人のみが、使用することができます。

商標

以下は、IBM Corporation の商標です。

400	AIX	AS/400
DB2	IBM	iSeries
OS/2	SecureWay	WebSphere

Domino は、Lotus Development Corporation の商標です。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。

UNIX は、The Open Group がライセンスしている米国およびその他の国における登録商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

他の会社名、製品名およびサービス名などはそれぞれ各社の商標または登録商標です。

Printed in Japan

日本アイ・ビー・エム株式会社
〒106-8711　東京都港区六本木3-2-12