eTrust® Access Control for Windows

管理者ガイド

r8 SP1

CA Product References

このマニュアルが参照している CA の製品は以下のとおりです。

- eTrust® Access Control（eTrust AC）
- eTrust® Single Sign-On（eTrust SSO）
- eTrust® Web Access Control（eTrust Web AC）
- eTrust® CA-Top Secret®
- eTrust® CA-ACF2®
- eTrust® Audit
- Unicenter® TNG
- Unicenter® Network and Systems Management（Unicenter NSM）
- Unicenter® Software Delivery

テクニカル サポートの連絡先

オンラインのテクニカル サポートと、サポート センターの所在地、営業時間、および電話番号については、 http://www.caj.co.jp/support/ をご覧ください。

目次

第 1 章: 概要

このセクションには、以下のトピックが含まれます。

本書の内容

本書では、eTrust AC for Windows に採用されているさまざまな概念について説明します。eTrust AC for Windows は、オープン システムに統合的なセキュリティ ソリューションを提供する製品です。 このガイドでは、eTrust AC について説明し、その中でも特に eTrust AC for Windows の管理に使用するユーザ インターフェースであるポリシー マネージャについて説明しています。

本書の対象読者

本書は、eTrust AC によって保護される環境の実装およびメンテナンスを担当するセキュリティ管理者およびシステム管理者を対象にしています。

コマンドの表記規則

eTrust AC のドキュメントでは、コマンド構文やユーザ入力の説明の際に、いくつかの特殊な表記法を使用しています。

[Format]	意味
等幅フォント	コードまたはプログラムの出力
斜体	情報を入力するためのプレースホルダ
太字	表示されているとおりに入力する必要のある要素
角かっこ([])で囲まれた文字列	省略可能な項目
中かっこ({})内でパイプ(|)で区切られた選択項目	選択すべき項目を 1 つだけ含む選択項目のセット
行の最後にあるスペースと円記号(¥)	次の行にコマンドが続く

注:

- 太字は、強調の意味にも使用されています。 次に例を示します。

  パスワードをモニタに表示したままに**しないでください**。

- 本書では、コマンドの記述が 1 行に収まらない場合があります。 このような場合、行末のスペースとそれに続く円記号（ ¥）は、そのコマンドが次の行に続いていることを示します。

  注: 円記号は実際のコマンド構文では不要なため、コピーしないでください。

- いずれか 1 つのみ選択する項目は、パイプ（|）で区切られています。 選択項目のセットは中かっこ（{}）で囲まれています。いずれかの項目を入力する場合、この中かっこは**入力しません**。 たとえば、以下の例では、ユーザ名**または**グループ名の**いずれか**を意味します。

  ```
  {username|groupname}
  ```

例： コマンドの表記規則

以下のコードは、本書でのコマンド表記規則の使用方法を示しています。

```
ruler className [props({all|{propertyName1[,propertyName2]})]
```

この例の内容:

- 太字で示されたコマンド名（ruler）は表示されているとおりに入力します。
- 斜体で示された className オプションは、クラス名（USER など）のプレースホルダです。
- 後半の角かっこの部分を指定しなくても、コマンドは実行できます。
- オプションのパラメータ（props）を使用する場合は、キーワード all を選択するか、またはカンマで区切られたプロパティ名を 1 つ以上指定します。

第 2 章: 基本概念

このセクションには、以下のトピックが含まれます。

eTrust AC

eTrust AC は、オープン システムのためのアクティブで包括的なセキュリティ ソフトウェア ソリューションであり、オペレーティング システムと動的に連携するソフトウェア製品です。 ファイルを開く、ユーザ ID を変更する、ネットワーク サービスを取得するなど、セキュリティ保護が必要な操作をユーザが要求するたびに、eTrust AC は各イベントをリアルタイムでインターセプトし、その妥当性を検証してから、オペレーティング システム(OS)標準機能に制御を渡します。

Access Control とは

eTrust AC は、ネイティブ プラットフォームのセキュリティ管理を行うための強力なツールを提供し、企業のセキュリティ要件に合わせて完全にカスタマイズできるセキュリティ ポリシーの実装を可能にします。 eTrust AC を使用すると、ユーザ、グループ、およびリソースに対して、ネイティブ オペレーティング システムでは実現できない強力なセキュリティを確保でき、組織全体のセキュリティを集中管理できます。また、異機種環境において Windows と UNIX のセキュリティ ポリシーを統合できます。

保護の対象

eTrust AC は、以下のエンティティを保護します。

- **ファイル**

  特定のファイルにアクセスする権限があるか?

  eTrust AC は、ファイルへのユーザのアクセスを制限します。 ユーザに対して、READ、WRITE、EXECUTE、DELETE、RENAME などのアクセス権限を 1 種類以上与えることができます。 アクセス権限は、個々のファイルに対して、または類似した名前を持つファイルの集合に対して指定できます。

- **端末**

  特定の端末を使用する権限があるか?

  このチェックは、ログイン プロセスで行われます。 個々の端末または端末グループを eTrust AC データベースに定義し、その端末または端末グループの使用を許可されているユーザまたはユーザ グループを明示したアクセス ルールを指定できます。 端末を保護することによって、強力な権限を持つユーザ アカウントへのログインに未許可の端末が使用されることを確実に防止します。

- **サインオン時間**

  ユーザには、特定の曜日の特定の時間にログインする権限があるか?

  通常、エンド ユーザは平日の勤務時間帯にのみ端末を使用します。そのため、休日のアクセス制限とともに平日の曜日と時間帯によるログイン制限を行うことによって、ハッカーやその他の権限のないアクセサから端末を保護できます。

- **TCP / IP**

  相手の端末には、ローカル コンピュータから TCP/IP サービスを受け取る権限があるか? 相手の端末には、ローカル コンピュータに TCP/IP サービスを供給する権限があるか? 相手の端末は、ローカル端末のすべてのユーザからサービスを受け取ることを許可されているか?

  オープン システムの長所は、コンピュータとネットワークの両方がオープンであるという点ですが、これは同時に短所でもあります。 いったんコンピュータが外部に接続されると、故意または過失により、外部ユーザがシステムに侵入したり、そのユーザが行った行為が損害をもたらしたりする危険が発生します。 eTrust AC には、「ファイアウォール」が用意されており、ローカルの端末やサーバが未承認の端末にサービスを提供することを防止します。

- **複数ログイン権限**

  ユーザは他の端末からログインできるか?

  同時ログインとは、ユーザが複数の端末からシステムにログオンできることを意味します。 eTrust AC では、1 人のユーザが複数の端末から同時にログインするのを防止できます。 この機能によって、すでにログインしているユーザのアカウントで外部からの侵入者がログインするのを防止します。

- **ユーザ定義エンティティ**

  標準エンティティ(TCP/IP サービスや端末など)および機能エンティティ(トランザクションの実行やデータベース内のレコードへのアクセスなどの抽象オブジェクト)の両方を定義して保護できます。 抽象オブジェクトの定義および保護に使用するAPI(Application Programmer's Interface)については、「SDK 開発者ガイド」を参照してください。

- **管理者権限**

  eTrust AC には、管理者権限をオペレータに委任する方法と、root 自体を制限する方法の両方が用意されています。

  eTrust AC には、管理者権限をオペレータに委任する方法、および管理者権限自体を制限する方法が用意されています。

- **レジストリ キー**

  ユーザには、特定のレジストリ キーにアクセスする権限があるか?

  eTrust AC は、レジストリ キーへのユーザのアクセスを制限します。 ユーザに対して、READ、WRITE、DELETE などのアクセス権限を 1 種類以上与えることができます。 アクセス権限は、個々のレジストリ キーに対して、または類似した名前を持つレジストリ キーの集合に対して指定できます。

- **プログラム**

  特定のプログラムを信頼できるか? ユーザには、このプログラムを起動する権限があるか? ユーザは、プログラムを使用して、特定のリソースにアクセスできるか?

  セキュリティ管理者は、プログラムをテストして、これらのプログラムに、アクセス権の不正取得に利用される可能性があるセキュリティ ホールがないことを確認できます。テストで安全とみなされたプログラムは、trusted プログラムとして定義されます。eTrust AC 自己防衛機能モジュール(eTrust AC の Watchdog 機能ともいう)は、ある特定の時点で制御の対象になっているプログラムを認識し、そのプログラムが、trusted と分類された後に変更または移動されたかどうかをチェックします。 trusted プログラムが変更または移動された場合、その時点で trusted とはみなされなくなり、eTrust AC はプログラムの実行を許可しません。

さらに、eTrust AC では、以下のような作為的または偶発的な脅威に対して防御を行います。

- **強制終了**

  eTrust AC を使用すると、重要なサーバやサービス、またはデーモンを強制終了から保護できます。

- **パスワード攻撃**

  eTrust AC は、さまざまな種類のパスワード攻撃からパスワードを保護します。サイトのパスワード定義ポリシーを適用し、パスワードの盗用による侵入を検知します。

- **不適切なパスワード**

  eTrust AC ポリシーでは、充分な品質のパスワードを作成して使用することをユーザに強制するルールが定義されます。 eTrust AC では、ユーザが基準に合ったパスワードを作成して使用することを確実にするために、最長および最短のパスワード有効期限の設定、特定の語句の使用制限、文字の繰り返しの禁止、およびその他の制限事項の適用を行うことができます。 パスワードを長期間継続して使用することは認められません。

- **アカウント管理**

  eTrust AC ポリシーによって、休止状態のアカウントの適切な処理が保証されます。

保護の方法

eTrust AC サービスは、オペレーティング システムの初期化が終了するとただちに開始できます。 eTrust AC によって、保護の必要なシステム サービスにフックが設定されます。 このようにして、サービスが実行される前に制御が eTrust AC に渡されます。 eTrust AC によって、サービスの使用をユーザに許可するかどうかが決定されます。

たとえば、eTrust AC によって保護されているリソースにユーザがアクセスしようとしたとします。 このアクセス要求によって、カーネルに対してリソースのオープンを指示するシステム コールが生成されます。 そのシステム コールは eTrust AC によってインターセプトされ、アクセスを許可するかどうかが決定されます。 アクセスが許可された場合は、eTrust AC によって通常のシステム サービスに制御が渡されます。アクセスが許可されない場合は、システム コールをアクティブにしたプログラムに、eTrust AC によってアクセス許可拒否の標準エラー コードが返され、システム コールの処理が終了します。

これは、データベースに定義されたアクセス ルールとポリシーに基づいて決定されます。データベースのレコードの大部分は、セキュリティ管理者が定義します。

データベースには、アクセサとリソースという 2 種類のオブジェクトが定義されています。アクセサとは、ユーザおよびグループのことです。 リソースとは、ファイルやサービスなど、保護対象のオブジェクトのことです。 データベース内の各レコードには、アクセサまたはリソースが定義されています。

各オブジェクトはクラスに属します。クラスは、同じタイプのオブジェクトの集合です。 たとえば、TERMINAL は、eTrust AC によって保護されている端末（ワークステーション）であるオブジェクトが含まれるクラスです。

クラスのアクティブ化

CLASS ステータス(そのクラスがアクティブか非アクティブかを示す)に関する情報は、データベースに格納されます。 リソースに対するアクセスの試みはすべて eTrust AC によってインターセプトされ、データベース内のステータスがチェックされます。 クラスがアクティブでない場合は、それ以上の権限チェックは行われずにアクセスが許可されます。

<eTrust AC は、Engine が起動するとき、およびユーザが CLASS のアクティビティ ステータスを変更するときに、アクティブ クラスのリストを発行します。 クラスがアクティブでない場合、リソースへのアクセスはインターセプトされないので、オーバーヘッドが軽減されます。

アクセサ エレメント

各ユーザは、アクセサ エレメント(ACEE)として表されます。ACEE は、データベースに格納されているユーザのレコードをメモリ内に展開したものです。 eTrust AC は、ログイン プロセス時にアクセサ エレメントを構築します。 アクセサ エレメントは、ユーザのプロセスと関連付けられます。 eTrust AC によって保護されているシステム サービスをプロセスが要求するたびに、またはリソースにアクセスするために暗黙的な要求を発行するたびに、eTrust AC はそのリソースのレコードにアクセスします。 次に、以前に作成されたアクセサ エレメントの情報(ユーザのセキュリティ レベル、モード、グループなど)から、ユーザがリソースへのアクセスを許可されているかどうかを判断します。

コンポーネント

eTrust AC には、データベース(seosdb)、2 つのドライバ(seosdrv および drveng)、多数のサービス(Watchdog、Agent、Engine(seosd)、Policy Model、タスク委任機能など)、およびグラフィカル ユーザ インターフェースが含まれます。

データベース

データベースには、以下の要素の定義が格納されます。

- 組織内のユーザおよびグループ
- 保護が必要なシステム リソース
- ユーザおよびグループによるシステム リソースへのアクセスを管理するルール

ドライバ

ドライバは、以下のタスクを実行することによって、eTrust AC のファイルとレジストリ キーをすべて保護します。

- ファイルを開く要求、レジストリ キーにアクセスする要求、プロセスを終了する要求、およびネットワーク アクティビティを実行する要求をインターセプトする
- これらの要求を eTrust AC Engine に渡し、Engine から要求の許可または拒否の決定を受け取る
- この決定をオペレーティング システムの元のシステム コールに転送する（オペレーティング システムは、ドライバから受け取った応答に基づいて処理を継続する）

サービス

Watchdog

Watchdog は、他の eTrust AC サービスが実行されていることを常時チェックします。Watchdog は、他のサービスが停止していることを検出すると（ただし、停止することはほとんどありません）、ただちにそのサービスを再開します。

Agent

Agent は以下のタスクを実行します。

- TCP/IP 上の専用アプリケーション プロトコルを介して eTrust AC クライアントと通信する
- eTrust AC ユーザのセキュリティを管理する

Engine

Engine は以下のタスクを実行します。

- すべてのデータベース更新の管理を含むデータベースの管理を行う
- ドライバおよび Agent から受け取ったアクセス要求を許可するかどうかを決定する
- Watchdog サービスが実行中かどうかをチェックし、実行停止を検出した場合は Watchdog サービスを再開する

Engine は、データベース アクセス要求を処理し、**かつ**アクセス許可の決定を行うことによって、効率的なサービスを作成します。

Policy Model

何百ものデータベースを個別に管理することは、現実的ではありません。 そのため eTrust AC には、1 台のコンピュータから多数のコンピュータを管理できるコンポーネントである Policy Model サービスが用意されています。 Policy Model サービスの使用は任意ですが、このサービスを使用すると、大規模なサイトでの管理を大幅に簡略化できます。

Policy Model データベース (PMDB) は、この Policy Model サービスと共に使用します。 PMDB には、他の eTrust AC データベースと同様に、ユーザ、グループ、保護されているリソース、およびリソースへのアクセスを管理するルールが保存されています。 さらに、PMDB にはサブスクライバ端末のリストが含まれています。 サブスクライバ端末は PMDB にリンクされた端末であるため、PMDB への変更はサブスクライバ データベースに自動的に送信されます。

ユーザは、組織に適用する基本的なセキュリティ ポリシーを作成し、必要なすべてのルールを単一のデータベース(Policy Model データベース)に実装できます。 サブスクライバには、Windows 端末と UNIX 端末の両方を含めることができるため、最小限の管理作業で一定のルールを保証できます。

PMDB は、システム管理者またはセキュリティ管理者が更新します。 PMDB によってすべての更新内容が PMDB からサブスクライバにバッチ モードで伝達されるため、管理者は他の作業を行うことができます。

PMDB のサブスクライバには、別の PMDB とローカル データベースの 2 種類があります。 また、この PMDB には、データベースの更新内容の伝達先となるサブスクライバの一覧が保存されています。 この機能によって、PMDB の階層を構築できます。 ローカル データベースは、端末に定義されているユーザ、グループ、およびリソースを保護するために使用できます。

グラフィカル ユーザ インターフェース

ポリシー マネージャ (P. 39) ポリシー マネージャは、eTrust AC のすべての機能を実行できるグラフィカル ユーザ インターフェース(GUI)です。

機能

eTrust AC によって、ネイティブ Windows を集中管理し、ネイティブ Windows のセキュリティを大幅に向上させることができます。 eTrust AC はそれ自体を保護することもできます。 以下のセクションでは、これらの各機能について説明します。

Windows の管理

ネットワークに分散している Windows 端末に eTrust AC をインストールすると、中央の 1 つの端末からそれらの端末(各端末が属しているドメインに関係なく)をすべて一元管理できます。 これを行うには、ポリシー マネージャ インターフェースを使用するか、selang というコマンド ライン言語を使用します。

自己防衛機能の提供

ハッカーまたはユーザが eTrust AC のサービスを故意または過失により停止させることは事実上不可能です。 eTrust AC の実行中に、権限のないユーザが eTrust AC のファイルおよびデータを変更または消去することも実質的に不可能です。

ネイティブ Windows セキュリティの管理

eTrust AC を使用すると、Windows セキュリティに関する以下の要素を管理できます。

レジストリ保護機能

Windows レジストリは、デバイス ドライバ、環境設定の詳細、ハードウェア、環境、およびセキュリティの設定を制御するパラメータをはじめ、大半のオペレーティング システム パラメータを集中管理するデータベースです。

eTrust AC では、権限のないユーザがシステム パラメータを変更しないように、レジストリが保護されます。 権限のあるユーザは、必要に応じてレジストリの設定を更新できます。

Active Directory

Active Directory は、Windows 2000 以降の Windows オペレーティング システムで使用されているディレクトリ サービスです。ユーザ、コンピュータ、サービスなど、ネットワーク内のオブジェクトに関する情報のリポジトリが階層構造で提供されます。

Active Directory が Windows オペレーティング システムにインストールされている場合は、eTrust AC を使用して、ユーザとグループ、および拡張ユーザとグループのプロパティを追加および変更できます。この処理は、ネイティブ Windows 環境でユーザやグループを管理する場合と同じように実行できます。

Active Directory が Windows サーバにインストールされている場合は、OU クラスを使用して、ユーザ、グループ、およびコンピュータを特定の組織単位内に作成できます。

ファイルの保護

Windows では、異なる複数の種類のファイル システムを使用できます。 最も一般的なファイル システムは、FAT および NTFS です。 NTFS ファイル システムを使用している場合は、各ファイルに対して ACL を作成および更新することにより、システム内のファイルが Windows によって保護されます。 eTrust AC では、ファイルの ACL がサポートされます。

パスワード保護機能

ネイティブ Windows セキュリティにより、さまざまな方法でパスワードを保護し、パスワードの品質を強化できます。 Windows では次の機能が提供されています。

- パスワードの最長有効期間の指定
- パスワードの最低文字数の指定
- ユーザのパスワード履歴を最大 24 件まで保存できます。
- ログインに繰り返し失敗した場合のアカウントのロックアウト
- パスワード変更前の Windows へのログオンの強制

<eTrust AC では、同じルールが独自のメカニズムによって適用されます。 さらに、eTrust AC では、メインフレーム コンピュータとの双方向のパスワード同期機能が実装されています。

ネイティブ Windows セキュリティの拡張

以下の eTrust AC の機能により、ネイティブ Windows セキュリティが拡張されます。

管理者アカウントの制限

通常、Windows を管理するユーザ(管理者)は、システム セットアップ時に自動的に作成される、事前定義されたグループのメンバです。 事前定義された各グループは、特定のシステム機能のセットを実行します。 グループのメンバであるユーザは、グループの機能をすべて実行できます。

Windows で最も強い権限を持つグループは、Administrators グループです Administrators グループのすべてのメンバは、ユーザの作成、削除、および変更から、サーバのロック、環境設定の変更、およびシャットダウンまでの広範なタスクを実行できます。

Windows におけるセキュリティ上の主なリスクの 1 つは、権限のないユーザが Administrators グループのユーザ アカウントに対する制御権を手に入れる可能性があることです。 Administrator アカウントへのアクセスを許可してしまうと、システムは重大な危険にさらされることになります。

eTrust AC を使用すると、Administrator アカウントに与える権限を制限し、Administrators グループに属するユーザの権限を制限できます。 これにより、Windows システムの脆弱性をカバーします。

一般ユーザへの管理者権限の付与

eTrust AC では、Administrators グループのメンバでなくても管理タスクを実行できるように、必要な権限を一般ユーザ(管理者以外)に与えることができます。 これを「タスクの委任」といいます。このようなきめ細かな方法でタスクを委任できる(つまり、管理権限を付与できる)機能は、eTrust AC の最も重要な機能の 1 つです。

- SUDO クラスのレコードには、コマンド スクリプトが格納されています。ユーザは、借用した権限でそのスクリプトを実行できます。
- data プロパティの値はコマンド スクリプトです。 この値は、省略可能なスクリプト パラメータ値を追加することによって変更できます。
- SUDO クラスの各レコードは、あるユーザが別のユーザの権限を借用できるようにするためのコマンドを識別します。
- SUDO クラス レコードのキーは、SUDO レコードの名前です。 この名前は、ユーザが SUDO レコードでコマンドを実行する際に、コマンド名の代わりに使用されます。

SUDO レコードの定義

SUDO クラスのレコードには、コマンド スクリプトが格納されています。ユーザは、借用した権限でそのスクリプトを実行できます。 権限を借用できるかどうかは、スクリプトを実行する sesudo コマンドと SUDO レコードの両方で厳密に制御されています。

注: ターミナル サービスがインストールされているコンピュータで、SYSTEM アカウント以外のユーザ アカウントが SeOS タスク委任機能サービスを実行しているときは、sesudo コマンドで対話処理を実行することができません。

SUDO レコードでは、comment プロパティを特別な目的に使用します。通常、このような comment プロパティは data プロパティといいます。

comment プロパティの値は、コマンド スクリプトです。禁止(prohibited)または許可(permitted)するスクリプト パラメータ値が必要に応じて追加される場合もあります。comment プロパティ値全体は一重引用符で囲む必要があります。トロイの木馬の侵入を防ぐために、実行可能ファイルは完全パス名で参照する必要があります。

comment プロパティの形式は、以下のとおりです。

```
comment('cmd[;[prohibited-values][;permitted-values]]')
```

prohibited 値および permitted 値のリストは省略できるため、comment プロパティの値は以下のように簡略化することもできます。

```
newres SUDO NET comment('net use')
```

このコマンドに指定されている簡略化された値は、sesudo NET コマンドで「net use」コマンドを実行することを表します。 特定のスクリプト パラメータ値が禁止されていないため、すべての値が許可されます。

ワイルドカードと強力な変数を使用すると、prohibited パラメータおよび permitted パラメータを柔軟に指定できるようになります。 使用できるワイルドカードは、Windows の標準的なワイルドカードです。 禁止するパラメータおよび許可するパラメータには、以下の変数を指定することもできます。

変数	説明
$A	英字
$G	既存の eTrust AC グループ名
$H	ユーザのホーム ディレクトリで始まるパラメータ
$N	数値

変数	説明
$O	sesudo を実行するユーザの eTrust AC での名前
$U	既存の eTrust AC ユーザ名
$e	空のエントリ。 ルールに対してパラメータが指定されていない SUDO コマンドを指定する場合に使用します。
$f	既存のファイル名
$g	既存の Windows グループ名
$h	既存のホスト名
$r	Windows 読み取りアクセス権がある既存のファイル
$u	既存の Windows ユーザ名
$w	Windows 書き込みアクセス権がある既存のファイル
$x	Windows 実行アクセス権がある既存のファイル

prohibited パラメータ値のリストをスクリプトに追加する場合は、以下の規則に従います。

- スクリプトと prohibited パラメータの値をセミコロンで区切り、全体を一重引用符で囲みます。 たとえば、ユーザによる-start の使用を禁止し、それ以外のすべてのパラメータの使用を許可する場合は、次のコマンドを入力します。

  ```
  newres SUDO scriptname comment('cmd;-start')
  ```

  cmd はユーザのスクリプトを表します。

  また、パラメータ値を許可せず、すべてのパラメータをデフォルトに設定する場合は、SUDO レコードを次のように定義します。

  ```
  newres SUDO scriptname comment('cmd;*')
  ```

- 1 つのスクリプト パラメータに対して複数の prohibited 値を指定する場合は、スペース文字を区切り記号として使用します。 たとえば、ユーザによる-start および -stop の使用を禁止し、それ以外のすべてのパラメータの使用を許可する場合は、次のコマンドを入力します。

  ```
  newres SUDO scriptname comment('cmd;-start -stop')
  ```

- 複数のスクリプト パラメータに対して prohibited 値を指定する場合は、パイプ（|）を区切り記号として使用して、それぞれの prohibited 値セットの間を区切ります。たとえば、スクリプトの最初のパラメータで -start および -stop を使用することを禁止し、2 番目のパラメータで既存の Windows ユーザ名（前出の変数の表を参照）を使用することを禁止する場合は、以下のコマンドを入力します。

  ```
  newres SUDO scriptname comment('cmd;-start -stop | $u')
  ```

  指定したパラメータよりスクリプトのパラメータが多い場合は、指定した最後の prohibited パラメータのセットが、残りすべてのパラメータに適用されます。

permitted パラメータ値のリストをスクリプトに追加する場合は、以下の規則に従います。

- sesudo ユーティリティがパラメータ値について以下の項目をチェックします。
  - 対応する prohibited 値のいずれにも一致しないこと。
  - 対応する少なくとも 1 つの permitted 値に一致すること。

  つまり、prohibited リストにあるパラメータ値は、permitted リストにも指定されていても、permitted にはなりません。

- permitted 値のリストと prohibited 値のリストをセミコロンで区切り、全体を一重引用符で囲みます。 prohibited 値のリストを指定しない場合でも、セミコロンは必要です。セミコロンがないと、permitted 値として指定した値が、prohibited 値として処理されます。 たとえば、スクリプトのパラメータ値として値 NAME のみを許可する場合は、以下のコマンドを入力します。

  ```
  newres SUDO scriptname comment('cmd;;NAME')
  ```

- 他のリストの指定も同様に行います。
  - 1 つのスクリプト パラメータに対して複数の permitted 値を指定する場合は、スペース文字を区切り記号として使用します。

- 複数のスクリプト パラメータに permitted 値を指定する場合は、パイプ (|) を区切り記号として使用して、それぞれの permitted の値セットの間を区切ります。

たとえば、2 つのパラメータがあるとします。最初のパラメータには Windows のユーザ名でない数字を指定し、2 番目のパラメータには Windows のグループ名でない英字を指定する必要がある場合は、次のコマンドを入力します。

```
newres SUDO scriptname comment('cmd;$u | $g ;$N | $A')
```

スクリプトのパラメータが指定したパラメータより多い場合は、指定した最後の permitted パラメータのセットが、残りすべてのパラメータに適用されます。

したがって、comment プロパティ全体の形式は、スクリプト、パラメータごとの prohibited 値、パラメータごとの permitted 値の順になります。

```
comment('cmd; ¥
param1_prohib1 param1_prohib2 ... param1_prohibN | ¥
param2_prohib1 param2_prohib2 ... param2_prohibN | ¥
 ...
paramN_prohib1 paramN_prohib2 ... paramN_prohibN ; ¥
param1_permit1 param1_permit2 ... param1_permitN | ¥
param2_permit1 param2_permit2 ... param2_permitN |
 ...
paramN_permit1 paramN_permit2 ... paramN_permitN')
```

sesudo ユーティリティでは、ユーザが入力した各パラメータを以下の方法でチェックします。

1. パラメータ N と許可されているパラメータ N が一致するかどうかを確認します (許可されているパラメータ N が存在しない場合、最後に許可されたパラメータが使用されます)。
2. パラメータ N と禁止されているパラメータ N が一致するかどうかを確認します (禁止されているパラメータ N が存在しない場合、最後に禁止されたパラメータが使用されます)。

すべてのパラメータが許可されているパラメータと一致し、禁止されているパラメータと一致するパラメータが存在しない場合、sesudo はコマンドを実行します。

タスクの委任の例

例 1

1. Access Control のプログラム バーの[リソース]アイコン  をクリックします。

   [リソース]ウィンドウが表示されます。

2. [タスクの委任リソース]ツリーを展開して、[タスク]を右クリックし、[新規作成]を選択します。

   [SUDO リソースの新規作成 - 全般]ダイアログ ボックスが表示されます。このダイアログ ボックスで新規ポリシーを作成します。

3. [名前]フィールドに、SUDO レコードの名前「NET」を入力します。

   [データ]フィールドに、以下のように入力します。

   ```
   net;start;send
   ```

   データ プロパティの形式は以下のとおりです。

   ```
   command; prohibited-values; permitted-values
   ```

   たとえば、ユーザ any_user に対して net send の実行を許可し、net start の実行を禁止するには、[データ]フィールドに以下のように入力し、この SUDO レコードに対して any_user を許可します。

   ```
   net;start;send computer_name message
   ```

4. [所有者]フィールドで[参照]をクリックし、[ユーザ]タブにある[nobody]を選択後、[OK]をクリックします。

5. [デフォルト アクセス権の設定]をクリックし、[None]を選択してから、[OK]ボタンをクリックします。

6. ダイアログ ボックスの左側のパネルで[権限の付与]アイコンをクリックします。

   ダイアログ ボックスの[権限の付与]ページが表示されます。

7. アクセサを追加するには、[挿入]  をクリックします。次に[名前]フィールドの横にある[参照]をクリックします。

8. 権限を委任するユーザまたはグループを選択し、[OK]ボタンをクリックします。

   [eTrust アクセサの追加と編集]ダイアログ ボックスが表示されます。

9. [OK]をクリックします。

10. [Permissions]の[Execute]チェック ボックスをオンにし、[OK]ボタンをクリックします。

    新しい SUDO リソースが作成されます。

11. SUDO レコード テストを実行します。

    a. SUDO レコードが適用されたユーザとしてログインします。

    b. コマンド プロンプトを開き、以下のコマンドを実行します。

    ```
    sesudo -do NET start
    ```

    以下のメッセージが表示されます。

    ```
    sesudo: 'start' をパラメータ番号 1 として使用することは許可されていません。
    ```

    注: net start は prohibited 値として定義されたので、実行されません。

    c. 以下の値を実行します。

    ```
    sesudo -do NET send
    ```

    このコマンドは実行されます。

例 2

以下の例で示すように、ユーザは任意のスナップイン MSC モジュールを使用して、高い権限を必要とする操作を実行できます。

1. [リソース]ウィンドウで、[タスクの委任リソース]ツリーを展開して、[タスク]を右クリックし、[新規作成]を選択します。

   [SUDO リソースの新規作成 - 全般]ダイアログ ボックスが表示されます。このダイアログ ボックスで新規ポリシーを作成します。

2. [名前]フィールドに「services」と入力します。

3. [データ]フィールドに「c:¥winnt¥system32¥mmc.exe」と入力します。

4. [所有者]フィールドで[nobody]を選択します。

5. [対話式]チェック ボックスをオンにします。

   この対話機能は、サービスが開始されている状態のときに、ログインしたすべてのユーザが使用できるデスクトップ ユーザ インタフェースを提供します。 このインタフェースは、サービスが LocalSystem アカウントとして実行されている場合にのみ使用可能です。

6. [デフォルト アクセス権の設定]をクリックし、[None]を選択してから、[OK]ボタンをクリックします。

7. ダイアログ ボックスの左側のパネルで[権限の付与]アイコンをクリックします。

   ダイアログ ボックスの[権限の付与]ページが表示されます。

8. アクセサを追加するには、[挿入]  をクリックします。次に[名前]フィールドの横にある[参照]をクリックします。

9. 権限を委任するユーザまたはグループを選択し、[OK]ボタンをクリックします。

10. [Permissions]の[Execute]チェック ボックスをオンにし、[OK]ボタンをクリックします。

新しい SUDO リソースが作成されます。

11. SUDO リソース テストを実行します。
    a. SUDO リソースが適用されたユーザとしてログインします。
    b. コマンド プロンプトを開き、以下のコマンドを実行します。

       ```
       sesudo -do services
       ```

    c. mmc.exe が起動します。
12. SUDO リソースの実行を拒否するには、SUDO 権限の付与のプロパティを編集し、[拒否]列のチェック ボックスをオンにします。
13. コマンド プロンプトを開き、以下のコマンドを入力します。

    ```
    sesudo -do services
    ```

    以下のメッセージが表示されます。

    ```
    sesudo: services コマンドを使用する権限がありません。
    ```

    注: この例では、作成される SUDO リソースの名前は services になります。SUDO リソースは services スクリプトの実行を許可しません。

ファイル保護の強化

eTrust AC では、論理ファイル名形式と絶対ファイル名形式の両方がサポートされます。たとえば、ファイル foo.txt が論理ドライブ D: の ¥tmp ディレクトリに格納されており、論理名「D:」が物理ディスク 1、パーティション 0 に割り当てられている場合は、以下のように、論理ファイル名か絶対ファイル名のいずれかを使用して、eTrust AC データベースに対してファイルを定義します。

```
nr file D:\tmp\foo.txt
```

または

```
nr file ¥Device¥HardDisk1¥Partition1¥tmp¥foo.txt
```

注: 2 番目の形式を使用する場合は、ディスクの論理名が変更されても、ファイルは保護されたままになります。 絶対ファイル名形式は、eTrust AC の汎用ファイル保護でもサポートされます。

eTrust AC では、現在 Windows で使用されているすべてのファイル システムが保護されます。 最も一般的に使用されるファイル システムは、Windows File System (NTFS)と File Allocation Table (FAT)の 2 種類です。 eTrust AC では、CDFS (CD-ROM ファイル システム)および HPFS (OS/2 のファイル システム)もサポートしています。

eTrust AC により、File Allocation Table (FAT)に対する総合的なセキュリティ ソリューション、および NTFS や CDFS などその他のファイル システムに対する特別なセキュリティ レイヤが提供されます。

汎用ファイル保護

eTrust AC では、論理ファイル名形式と絶対ファイル名の両方がサポートされます。 絶対ファイル名形式は、eTrust AC の汎用ファイル保護でもサポートされます。

汎用ファイル保護により、指定したワイルドカード パターン(正規表現)に適合するすべてのファイルを保護できます。 指定したワイルドカード パターンに一致する名前のリソースが、指定した包括的なアクセス ルールによって保護されます。 eTrust AC により、ファイルを包括的に保護できます。

リソースが複数の包括的なアクセス ルールに一致する場合は、eTrust AC によって、ファイルに対して最も厳密に一致するルールが選択されます。

汎用ファイル保護の機能を使用すると、ほんのわずかなセキュリティ ルールを定義するだけで、保護の必要な多数のファイルを保護できます。

パスワード保護の強化

Windows ネイティブ セキュリティによって、非常に多くのユーザ パスワードに関する保護 (P. 19)が提供されます。 eTrust AC では、パスワード保護が大幅に拡張されているため、ハッカーによるパスワード盗用の可能性はきわめて小さくなりました。

eTrust AC を使用すると、より安全で確実なパスワードをユーザが選択するように、ルールを追加できます。 たとえば、最低限必要な英字、数字、特殊文字、小文字、または大文字の数を選択するようにユーザに要求できます。 また、置き換えられる旧パスワードと、ユーザが選択した新しいパスワードで、前者の文字列が後者の文字列に含まれないようにすることもできます。

Program Pathing

Program Pathing は、特定のファイルが特定のプログラムを介してのみアクセスされるように要求する機能です。 Program Pathing により、機密ファイルのセキュリティを大幅に強化できます。 eTrust AC の Program Pathing を使用すると、システム内のファイルに対する保護を強化できます。

B1 セキュリティ レベル認証

eTrust AC には、セキュリティ レベル、セキュリティ カテゴリ、およびセキュリティ ラベルという「Orange Book」の B1 機能があります。

- データベースのアクセサとリソースには、セキュリティ レベルを割り当てることができます。 セキュリティ レベルは、1 から 255 までの範囲の整数です。リソースに割り当てられたセキュリティ レベルに等しいか、またはそれより大きいセキュリティ レベルを持つアクセサのみがリソースにアクセスできます。
- データベースのアクセサとリソースは、1 つ以上のセキュリティ カテゴリに属することができます。 アクセサがリソースに割り当てられているすべてのセキュリティ カテゴリに属している場合のみ、そのアクセサはリソースにアクセスできます。

- セキュリティ ラベルは、特定のセキュリティ レベルを 0 個以上のセキュリティ カテゴリの集合に関連付ける名前です。ユーザをセキュリティ ラベルに割り当てると、セキュリティ ラベルに関連付けられたセキュリティ レベルおよびセキュリティ カテゴリの両方がユーザに設定されます。

注: Orange Book の B1 機能の詳細については、「実装ガイド」を参照してください。

eTrust AC の実行

eTrust AC を管理するには、ポリシー マネージャ インターフェースを使用するか、selang というコマンド ライン言語を使用します。 これらのツールを使用して、ローカル ワークステーション、および eTrust AC がインストールされているその他すべての Windows ワークステーションを管理できます。

ポリシー マネージャ

ポリシー マネージャは、eTrust AC の管理ツールです。

selang

コマンド ライン言語 selang を使用すると、eTrust AC のすべての機能を実行できます。 selang のコマンドを使用するには、コマンド プロンプト ウィンドウを開き、selang を起動します。 selang はスクリプトでも使用できます。

selang とそのコマンドの詳細については、「リファレンス ガイド」の章「selang - eTrust AC のコマンド言語」を参照してください。

Windows および UNIX のセキュリティ管理

大規模な組織では、Windows および UNIX の 2 つのシステムが混在する場合がよくあります。 こうした状況では、完全なセキュリティを維持することは困難です。 最も望ましいのは、すべてのシステムに実装できる単一のセキュリティ ポリシーを作成することです。

eTrust AC を使用して、以下の作業をすべて行うことができます。

- UNIX および Windows に使用する、単一の共通セキュリティ ポリシーを作成する
- eTrust AC を使用して、作成したポリシーを実装します。
- 1 台の Windows ワークステーションを使用して、Windows および UNIX の環境のセキュリティを集中管理する

eTrust AC のセキュリティ ポリシーの変更とその変更を異なる環境の多数のワークステーションに伝達する機能により、管理オーバーヘッドが大幅に削減されます。

以下のセクションで、共通のセキュリティ ポリシーで特に重要ないくつかの要素について説明します。

ユーザの一括メンテナンス

サイトに eTrust AC をインストールすると、すべてのユーザを管理している 1 つの eTrust AC データベースに対してメンテナンスを行うことができます。 これは、ユーザ メンテナンスを一度だけ行う必要があることを意味します。 eTrust AC では、更新内容を受け取る必要があるすべてのワークステーション(UNIX および Windows の両方)に対して、追加、変更、および削除を伝達できます。

グループの一括メンテナンス

多くの場合、特定のプロジェクト、または組織内の特定の部門に所属するユーザをグループ化すると管理が容易になります。 Windows、UNIX、および eTrust AC では、ユーザのグループを定義できます。 ユーザに権限を割り当てるのと同様に、グループに対して権限を割り当てることができます。 グループを使用すると、同じ権限を個々のユーザに繰り返し割り当てるのではなく、グループに対して 1 度割り当てればよいため、作業の負荷を軽減できます。

eTrust AC を使用して管理を行うと、UNIX および Windows の両方の環境で使用できるグループを作成し、メンテナンスできます。

アクセス ルールの一括メンテナンス

Policy Model サービスにより、Windows および UNIX の両方に使用できる 1 組のアクセス ルールを作成し、メンテナンスできます。 PMDB により、セキュリティ データベースの内容およびセキュリティ データベースに対して行われた変更をすべてのサブスクライバに伝達できます。 Windows および UNIX ワークステーションを同一の PMDB にサブスクライブできます。

PMDB とサブスクライバ間の通信は、通常、PMDB のデータベースからサブスクライバに変更内容を送信する一方向通信です。 サブスクライバは、オンラインであることを PMDB に通知するときにのみ PMDB と通信し、サブスクライバの停止中に PMDB が送信したすべての変更内容の再送を要求します。 このように設計されているため、ネットワーク通信量を最小限に抑えながら、サブスクライバの整合性が保証されます。

管理者の設定

eTrust AC のインストール時には、1 人以上の eTrust AC 管理者の名前を設定する必要があります。 eTrust AC 管理者には、ルール データベースのすべてまたは一部を変更する権限があります。 すべての権限を持つ管理者を最低 1 人は設定する必要があります。 この管理者は、アクセス ルールを自由に変更または作成することができ、他のレベルの管理者を指定できます。

システムのユーザを定義した後、管理者以外のユーザに ADMIN 属性を割り当てることによって、管理者権限を割り当てることができます。

注: ADMIN 属性が割り当てられたユーザには、強力な権限が与えられます。 このため、ADMIN ユーザの数は厳しく制限する必要があります。 また、Windows Administrator の役割と ADMIN の役割を分離し、1 人以上の eTrust AC セキュリティ管理者の設定が終了した後に、Administrator から ADMIN 属性を削除する方法もお勧めします。

eTrust AC では、データベースを管理する権限を持つユーザは常に最低 1 人は必要となるため、ADMIN 属性を持つ最後のユーザを削除することはできません。 したがって、Administrator から ADMIN 属性を削除する場合は、まず ADMIN 属性を他のユーザに与える必要があります。

eTrust AC 管理者がこのワークステーションから他のホストを管理する可能性がある場合は、そのホスト上のデータベースに、このワークステーションからの READ アクセス権と WRITE アクセス権の両方を管理者に与えるルールが定義されていることを確認してください。

両方の環境での推奨事項: サブ管理者の作成

eTrust AC には、一般ユーザが特定のクラスを管理できるようにする特定の権限を管理者が与えることができるサブ管理機能があります。 このようなユーザをサブ管理者といいます。

たとえば、特定のユーザに対して、ユーザとグループを管理できる権限を与えることができます。

また、特定のクラスに対してだけではなく、そのクラスの特定のオブジェクトに対してアクセス権を許可することにより、より高いレベルのサブ管理を指定することもできます。

監査手順の設定

eTrust AC では、データベースに定義されている監査ルールに基づいて、アクセス拒否およびアクセス許可のイベントに関する監査レコードが保存されます。 特定のイベントをログに記録するかどうかの決定は、以下のルールに基づいて行われます。

- すべてのアクセサおよびリソースに AUDIT プロパティがあり、このプロパティを設定すると、アクセスの成功または失敗、あるいはその両方のイベントをログに記録するかどうかを指定できる。さらに、アクセサの AUDIT プロパティでは、ログインの成功または失敗、あるいはその両方のイベントをログに記録するかどうかを指定できる。
- リソースまたはアクセサに AUDIT (ALL) 属性が割り当てられている場合は、eTrust AC によって保護されているリソースに関係するすべてのイベントが、アクセスが失敗したか成功したかに関わりなく、ログに記録されます。
- eTrust AC によって保護されているリソースへのアクセスが成功し、ユーザまたはリソースに AUDIT (SUCCESS) が割り当てられている場合は、イベントがログに記録されます。
- eTrust AC によって保護されているリソースへのアクセスが失敗し、ユーザまたはリソースに AUDIT (FAIL) が設定されている場合は、イベントがログに記録されます。

システム監査担当者 (AUDITOR 属性が割り当てられているユーザ) のみが、ユーザおよびリソースに割り当てられた監査属性の変更などの監査タスクを実行できます。

特定のリソースに警告モードが設定してあり、そのリソースに対するアクセス ルール違反が発生した場合は、監査レコードが生成され、警告モードが有効であるために違反が許可されたことが記録されます。

監査レコードによって、監査ログ (seos.audit) というファイルが構成されます。 監査ログの場所は、エラー ログの場所と同様にレジストリで指定されます。

監査ログ (およびエラー ログ) は、以下のレジストリ キーで指定されます。

```
HKEY_LOCAL_MACHINE\Software\ComputerAssociates\eTrustAccessControl\logmgr
```

監査ログはバイナリ ファイルであるため、編集または変更することはできません。 ただし、ポリシー マネージャを使用すると、記録されたイベントを表示し、時間制限やイベント タイプなどでイベントをフィルタ処理することができます (また、seaudit ユーティリティを使用しても、同様のタスクを実行できます)。

後からイベントを調査できるように、古い監査ログおよびエラー ログをアーカイブ (バックアップ) することをお勧めします。

Unicenter TNG への監査イベントの送信

Unicenter TNG との統合は、インストール時に設定します。

監査データを Unicenter TNG に送信するか、または Unicenter TNG から eTrust AC を起動できるようにするか、あるいはその両方を行うかを選択できます。 この 2 つのオプションには関連性がありません。

最初のオプションを選択することにより、以下のサブキーにレジストリ値が設定されます。

```
HKEY_LOCAL_MACHINE¥Software¥ComputerAssociates¥eTrustAccessControl¥UCTNG
```

値 Integration を 1(yes)に設定すると、値 EvtManagerServer が Unicenter TNG ホストの名前を文字列で受け取ります。

Unicenter TNG に渡される監査イベントは、[Unicenter エンタープライズ管理]-[エンタープライズ マネージャ]-[Windows NT]-[イベント]ウィンドウのコンソール ログに表示されます。

監査イベント	表示色	重大度
成功	青	S
拒否	オレンジ	F
失敗	オレンジ	F
警告	青	W
eTrust AC の停止(監査終了)	青	I
eTrust AC の開始(監査開始)	青	I

2 番目のオプションを選択すると、Unicenter の WorldView メニューから eTrust AC を起動できます。eTrust AC を起動するには、[管理されるオブジェクト]ウィンドウで、TCP/IP ネットワークを表すアイコンをポイントして右クリックし、表示されたメニューから[eTrust AC]を選択します。

また、イベントに関する以下の情報も送信されます。

- 製品名（eTrust Access Control + バージョン番号）
- ユーザ名
- 端末名
- クラス名
- リソース名
- プロセス名
- イベントの時刻
- eTrust AC 監査の形式の完全な監査メッセージ

[ユーザ名]、[端末名]、[クラス名]、[リソース名]、および[プロセス名]の各フィールドは、イベント タイプによっては送信されないこともあります。

Policy Model データベースの使用法

多数のコンピュータやワークステーションが共存する大規模サイトで、何百もの eTrust AC データベースを個別に管理するのは現実的ではありません。 企業内の大部分のコンピュータに同じセキュリティ ルールを適用する場合、ルールを 1 回適用するだけで、そのルールが正しく伝達される手段が必要となります。 eTrust AC では、Policy Model データベース(PMDB)によってその機能が提供されます。

PMDB には、ローカル データベース(eTrust AC またはネイティブ オペレーティング システム)と同じ種類の情報、およびサブスクライブしているデータベース(ローカル データベースまたは他のホスト上にある他の PMDB)のリストが保存されています。 PMDB は、基本的にはマスタ ルール データベースまたはテンプレートです。 PMDB に加えられたすべての変更と PMDB に定義されているルールは、サブスクライブしているデータベースに適用されます。

PMDB 機能にはシンプルな階層モデルが用意されていて、アクセス ルールを複数のシステムに配布し、ホストのグループに対して同一のアクセス ルールを作成することができます。 PMDB に階層構造を設定すると、複数レベルのポリシーをサポートできます。すなわち、企業内のすべてのホストに適用するアクセス ルールを定義する最上位レベルのテンプレートから、ホストのサブグループに適用可能な特定のルールを定義する下位レベルのテンプレートまで対応できます。

PMDB には複数のサブスクライバを設定できますが、伝達されたルール変更の受信を目的とする場合、PMDB またはローカル データベースは、1 つの親 PMDB に対してのみサブスクライブすることができます。 また、パスワード変更の伝達を目的とする場合、各 PMDB には、同じ親 PMDB、または(パスワード PMDB という)異なる親 PMDB を設定できます。 パスワード変更は、ルール変更とは異なり両方向に伝達されます。つまり、変更が行われたホスト上のローカル データベースから、パスワード Policy Model の最上位まで、および階層内のすべての下位サブスクライバまで伝達されます。

暗号化の設定

ネットワーク上に eTrust AC を実行している複数のサーバがある場合、異なるサーバ間での eTrust AC サービスの通信は暗号化されます。 デフォルトでは、高速で効率的なスクランブル アルゴリズムを使用して eTrust AC によって暗号化が行われます。

eTrust AC には、インストール時に選択できる AES、DES、および 3DES の暗号化オプションも用意されています。

標準暗号化

eTrust AC の暗号化形式は、ダイナミック リンク ライブラリ(DLL)で実装されます。 この DLL によって、すべてのデータの暗号化と復号化が実装され、安全な方法でクライアント プログラム(selang.exe または SeAM.exe)と Agent サービスとの間でデータを転送できます。 デフォルトの暗号化鍵を変更するには、コマンド プロンプトから sechkey ユーティリティを使用するか、または Windows から ChEncKey.exe ユーティリティを使用します。

eTrust AC では、以下のディレクトリに、defenc.dll(デフォルト暗号化用)、aes128enc.dll(128 ビット AES 暗号化用)、aes192enc.dll(192 ビット AES 暗号化用)、aes256enc.dll(256 ビット AES 暗号化用)、desenc.dll(DES 暗号化用)、および tripledesenc.dll(3DES 暗号化用)というファイルがインストールされます。

```
eTrustACDir\bin
```

ここで、eTrustACDir は eTrust AC をインストールしたディレクトリです。

defenc.dll、aesenc.dll、desenc.dll、または tripledesenc.dll の完全パスは、以下のレジストリ サブキーのレジストリ値 Encryption Package として保存されます。

```
HKEY_LOCAL_MACHINE\Software\ComputerAssociates\eTrustAccessControl
```

カスタム暗号化

独自の暗号化方式を使用する場合は、新しい DLL を作成する必要があります。 新しい暗号化 DLL を以下のディレクトリに格納する必要はありませんが、Encryption Package のレジストリ値は、新しい DLL の完全パスで表したディレクトリ名で置き換える必要があります。

```
eTrustACDir¥bin
```

ここで、eTrustACDir は eTrust AC をインストールしたディレクトリです。

暗号化 DLL には、次の 3 つのエクスポート関数が含まれている必要があります。

1. DWORD Init(PCHAR pKey, DWORD dwLength)

   この関数は、暗号化鍵を初期化します。

2. DWORD Scramble(void *param, char *buffWithPlainText, int sizeofbuffWithPlainText, char *buffWithEncryptText, int *sizeofbuffWithEncryptText)

   この関数は、第 2 パラメータで受け取ったデータを暗号化し、暗号化されたバッファを第 4 パラメータに格納します。

3. DWORD Unscramble(void *param, char *buffWithEncryptText, int sizeofbuffWithEncryptText, char *buffWithPlainText, int *sizeofbuffWithPlainText)

   この関数は、第 2 パラメータで受け取ったデータを復号化し、そのバッファを第 4 パラメータに平文として格納します。

重要: 暗号化鍵または暗号化 DLL を変更する場合は、相互に通信するすべてのホストに同じ変更を行う必要があります。 この変更を行わないと、暗号が異なる状況が発生し、ホスト間の通信が正常に実行できなくなります。

第 3 章: 管理者インターフェースの使用法

このセクションには、以下のトピックが含まれます。

ポリシー マネージャ

ポリシー マネージャは、Windows および UNIX 環境で eTrust AC ローカル データベースと PMDB を管理および監査するためのインターフェースです。 この章のセクションでは、ポリシー マネージャについて説明します。さらに、eTrust AC を使用してセキュリティ ポリシーを実装およびメンテナンスする方法について説明します。 ポリシー マネージャでは、ネイティブ Windows 環境およびネイティブ UNIX 環境も管理できます。 ポリシー マネージャでは、Windows のユーザ マネージャまたは UNIX のラインコマンドを使用して実行できる操作はほとんどすべて実行できます。

ポリシー マネージャ インターフェース

すべてのデータ管理は、ポリシー マネージャのメイン ウィンドウから始めます。 正常にログオンできると、このウィンドウが表示されます。

注: ポリシー マネージャを実行するには、インストール時に、使用する端末を管理コンソールとして定義する必要があります。 詳細については、「実装ガイド」を参照してください。

メニュー バー

メニュー バーには、ポリシー マネージャで使用できるコマンドのプルダウン メニューが登録されています。 メニュー バーの構造は動的です。つまり、実行するアクションに応じて適切なコマンドが表示されます。 たとえば、[Tree]メニューは、アクティブ ウィンドウにツリー構造が表示されている場合にのみ表示されます。

ツール バー

ツール バーを使用すると、使用頻度の高いコマンドに簡単にアクセスできます。これらのコマンドのほとんどは、メニュー バーからもアクセスできます。 メニュー バーと同様、ツール バーも動的です。つまり、実行するアクションに応じて適切なコマンドが表示されます。 共通のツールについては、以下のセクションで説明します。 特定のウィンドウに固有のツールについては、そのウィンドウの機能に関するセクションで説明します。

プログラム バー

プログラム バーでは、保護する項目、保護される項目を具体的に選択することができます。 プログラム バーのパネルを表示するには、[Access Control]、[Windows]、[Tools]と各ボタンをクリックしていきます。

ワークスペース

ワークスペースには、[ファイル]メニューまたはプログラム バーから開いたウィンドウが表示されます。 これらのウィンドウは、アプリケーション ウィンドウと呼ばれます。

出力バー

出力バーにはコマンド ログが表示されます。コマンド ログは、eTrust AC で selang のコマンドが書き込まれるファイルです。 出力バーには、作成されたコマンド、作成元のホスト、作成環境、およびそのコマンドの実行日時が表示されます。

ポリシー マネージャの新しいセッションを開始するたびに、eTrust AC では新しいコマンド ログが作成されます。 したがって、セッションのコマンドを保存する場合は、このログを保存するか出力する必要があります。

注: [ポリシー マネージャ]ウィンドウの出力バーの各行には、コマンド ログにある selang の複数のコマンドが表示される場合があります。

注: ポリシー マネージャとその使用方法の詳細については、ポリシー マネージャのオンライン ヘルプを参照してください。

アクセサの管理

アクセサ（アカウントともいう）とは、システム リソースにアクセスできるエンティティのことです。 最も一般的なアクセサ タイプはユーザです。ユーザとは通常、ログオンし、アクセス権限の割り当てとチェックの対象となる個人です。 グループ、プログラム、および端末もアクセサです。

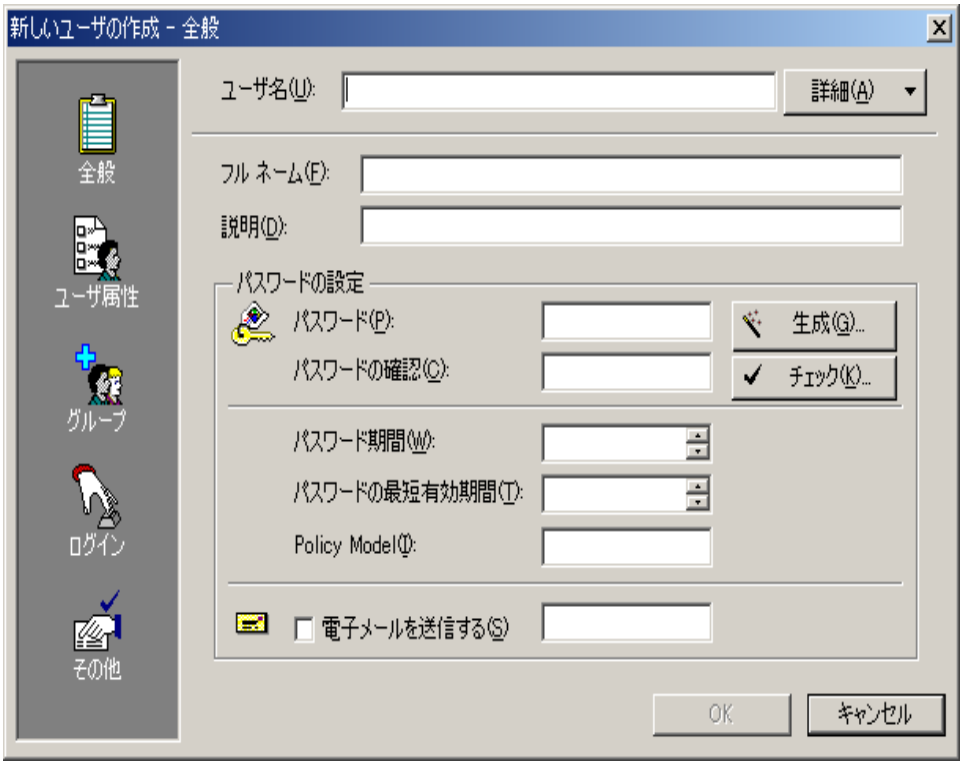
eTrust AC では、アカウント名のみ、あるいは Windows ドメイン名またはサーバ名（ユーザ アカウントが Windows ドメインの一部でない場合）が先頭に付加されたアカウント名でユーザを識別できます。識別方法は、eTrust AC データベースでユーザ レコードを作成したときに、いずれのアカウント名を使用したかによって決まります。

ネイティブ Windows オペレーティング システムおよび eTrust AC データベース（eTrust 環境）に定義されているすべてのユーザとグループを管理できます。 次の操作を実行できます。

- ユーザまたはグループを Windows または eTrust のいずれかの環境、あるいはその両方に追加する。
- ユーザまたはグループをいずれかの環境、あるいはその両方で更新する。
- ユーザまたはグループをいずれかの環境、あるいはその両方から削除する。
- ユーザの名前を変更する（Windows 環境のみ）。
- ユーザをグループに追加またはグループから削除する。
- グループをグループに追加またはグループから削除する。
- ユーザまたはグループの保護されているリソースを表示する。

これらの機能を実行するには、プログラム バーの Access Control のパネルで[ユーザ]または[グループ]をクリックし、次に、ツールバーの[新規作成]、[削除]、または[プロパティ]をクリックします。

次の図は、ユーザの追加に使用するダイアログ ボックスを示しています([Users]-[New]をクリック)。



左側にあるアイコンをクリックすると、別のパネルが表示されます。 たとえば、図に示されている[全般]パネルでは、ユーザの名前と説明を入力し、eTrust AC または Windows の環境を指定し([詳細]ボタン)、パスワード情報を設定します。

注: eTrust AC には、アクセサの管理に必要なタスクのいくつかを実行するためのウィザードが用意されています。 ウィザードを実行するには、[Access Control]パネルで[Users]または[Groups]をクリックし、[Tools]メニューから選択するか、[Wizards]ツールバー ボタンをクリックします。

重要: ユーザの定義に Windows NT のバックアップ ドメイン コントローラ(BDC)を使用しないことを強くお勧めします。 ネイティブ Windows でユーザー マネージャやドメイン ユーザー マネージャを使用して実行できる機能のほとんどは、プログラム バーの[Access Control]パネルと[Windows]パネルで実行できます。

インストールの際、または NT インポート ウィザードを使用してインストール後に、ユーザおよびグループを Windows システムから eTrust AC データベースにインポートすることができます。

注: 詳細な情報および手順については、ポリシー マネージャのオンライン ヘルプを参照してください。

アクセサへの Windows 権限の割り当て

Windows では、標準権限および拡張権限をユーザとグループに割り当てることができます。 拡張権限のほとんどは、Windows ワークステーションまたは Windows サーバが動作しているコンピュータ用のアプリケーションを作成するプログラマにのみ有用です。通常、グループまたはエンド ユーザには拡張権限は与えられません。

注: 拡張権限の詳細については、Windows サーバのプログラミング マニュアルを参照してください。

ユーザ ログオンの制限

ユーザのログイン権限は、以下の方法で制限できます。

- 有効期限を指定する。
- アカウントを一時停止し、eTrust AC データベースにアカウントを残したままユーザがログオンできないようにします。
- 猶予ログイン回数を指定する。
- ユーザがログオンできる端末台数の最大値を指定する。
- アカウントが無効になるまでの日数を指定する。
- ログオン権限を特定の日時に制限する。

デフォルトでは、アカウントは有効期限が指定されていないため無効になりません。また、アカウントの一時停止もなく、ユーザが同時にログオンできる端末台数にも制限はありません。

ログイン権限を制限するには、[ログイン]パネルを使用します。

監査対象のユーザ アクティビティの選択

eTrust AC データベースに定義されたユーザに対して、eTrust AC で監査対象とするユーザ アクティビティを指定できます。

注: 監査プロパティを指定できるのは、データベースで AUDITOR 属性が割り当てられているユーザのみです。 ユーザがネイティブ環境のみに定義されている場合、このオプションは淡色(グレー)表示されます。

以下に示す監査モードによって、eTrust AC の監査ログに記録するユーザ アクティビティが指定されます。 これらのオプションは、ユーザの作成および編集のためのダイアログ ボックスにある[各種設定]パネルから使用できます。

[Success]

eTrust AC に定義されたリソースへのアクセスに成功した場合、ログに記録されます。

[ログインの成功]

ログオンに成功した場合、ログに記録されます。

[ログインの失敗]

ログオンに失敗した場合、ログに記録されます。

[Failure]

データベースに定義されたリソースへのアクセスに失敗した場合、ログに記録されます。

[All]

成功したか失敗したかに関わらず、すべてのユーザ アクティビティがログに記録されます。

[None]

ユーザ アクティビティはいっさいログに記録されません。

個人情報の入力

ユーザの個人情報は、ユーザの作成および編集に使用するダイアログ ボックスにある[各種設定]パネルから入力できます。 以下のプロパティの入力は必須ではありません。

[場所]

ユーザの所在地(本社または東部事業所など)を指定する 128 文字以内の英数字からなる文字列。

[国]

ユーザの国名を示す 19 文字以内の英数字からなる文字列。

[組織名]

ユーザが所属する組織を示す 256 文字以内の英数字からなる文字列。

[組織単位]

ユーザが所属する組織単位を示す 256 文字以内の英数字からなる文字列。

［電話］

ユーザの電話番号を示す 19 文字以内の英数字からなる文字列。

［電子メール］

ユーザの電子メール アドレスを示す 256 文字以内の英数字からなる文字列。

アカウント情報の設定

ユーザのアカウント情報は、ユーザの作成および編集に使用するダイアログ ボックスにある［Miscellaneous］パネルから設定できます。 次のプロパティの入力は必須ではありません。

ホーム ディレクトリ

ユーザのホーム ディレクトリ。 ユーザは自動的に自分のホーム ディレクトリにログインするようになっています。

スクリプト

ユーザがログインしたときに自動的に実行されるファイル名です。このログイン スクリプトによって作業環境が設定されます。

プロファイル パス

ユーザのプロファイルが保存されているファイルの完全パス。 ユーザがワークステーションにログインすると、毎回同じ環境が画面に表示されます。

ユーザ権限の割り当て

ユーザ権限は、ユーザの作成および編集ダイアログ ボックスにある［Miscellaneous］パネルから割り当てることができます。 ユーザ権限には、次のようなものがあります。

- システム時間の変更
- デバイス ドライバのロードおよびアンロード
- ローカル ログイン
- ファイルおよびディレクトリの復元
- ファイルおよびディレクトリのバックアップ

B1 セキュリティ機能の使用

「Orange Book」の B1 セキュリティ機能を使用して、さらにセキュリティを強化することもできます。 ユーザの作成および編集に使用するダイアログ ボックスにある[その他]パネルで[B1 機能]ボタンをクリックし、セキュリティのラベル、カテゴリ、およびレベルを[B1 機能]ダイアログ ボックスから選択します。

- セキュリティ レベルは、1 から 255 までの間でユーザに割り当てることができます。ユーザのセキュリティ レベルが、リソースに割り当てられたセキュリティ レベル以上である場合にのみ、そのユーザはリソースにアクセスできます。
- ユーザは 1 つ以上のセキュリティ カテゴリに属することができます。 ユーザがリソースに割り当てられているすべてのセキュリティ カテゴリに属している場合のみ、そのユーザはリソースにアクセスできます。
- セキュリティ ラベルは、特定のセキュリティ レベルをセキュリティ カテゴリの集合に関連付けます。 ユーザをセキュリティ ラベルに割り当てると、セキュリティ ラベルに関連付けられたセキュリティ レベルおよびセキュリティ カテゴリの両方がユーザに設定されます。

セッション グループの割り当て

ユーザの作成および編集に使用する[その他]パネルの[セッション グループ]ボタンをクリックして、任意の eTrust SSO セッション グループをユーザに割り当てることができます。

グループへのユーザの割り当て

ユーザをグループに追加すると、ユーザの管理作業を容易にすることができます。ユーザの作成および編集のためのダイアログ ボックスにある[グループ]パネルを使用します。

ネストしたグループの追加

ネストされたグループは、グループの作成および編集に使用するダイアログ ボックスにある[各種設定]ペインから追加または変更することができます（プログラム バーの[グループ]をクリックし、ツールバーの[新規作成]または[プロパティ]をクリックします）。

[ネストされたグループ]ダイアログ ボックスでは、既存のグループに対して、スーパーグループ（親）およびメンバ グループ（子）の追加や削除を実行できます。 スーパーグループのプロパティは、そのメンバ グループに継承されます。

Active Directory のプロパティの設定

Active Directory を使用している Windows 2000 コンピュータに接続している場合は、[ユーザのプロパティ]または[グループのプロパティ]ダイアログ ボックスの[Active Directory]パネルを使用して、Active Directory のユーザまたはグループのプロパティを設定できます。 これらのプロパティは、Windows NT や、Active Directory がインストールされていない Windows 2000 コンピュータ、または eTrust AC ネイティブ環境のデータベースではサポートされていません。

Active Directory をサポートする Windows 2000 マシンに接続していない場合、このパネルを起動するアイコンは表示されません。

注: Active Directory を使用すると、ユーザを複数のフォルダに分けて管理できます。ポリシー マネージャでは、1 つの[ユーザ]パネルにすべての Active Directory ユーザが表示されます。

ネイティブ オペレーティング システムとのデータの同期

selang のコマンドを実行すると、ネイティブ オペレーティング システムのデータを変更することなく、データベース内のアクセサに関するデータを変更できます。 同様に、Windows のユーザ マネージャを使用すると、eTrust AC のデータを変更することなく、Windows でアクセサに関するデータを変更できます。 いずれの方法でデータを変更した場合も、アクセサの定義が各データベースで一致しなくなります。

eTrust AC により、eTrust AC およびネイティブ オペレーティング システムの定義が監視され、Windows と eTrust AC の定義が一致しない場合は[プロパティの同期]パネルが表示されます。 定義が一致している場合、[プロパティの同期]アイコンは表示されません。

eTrust AC リソースの管理

リソースとは、ユーザおよびグループがアクセスできるエンティティのことです。 最も一般的なタイプのリソースは、ファイルです。 ユーザは、ファイル情報を参照する場合またはファイルに情報を書き込む場合に、ファイルにアクセスします。

リソースは、クラス(リソース タイプの名前)別にグループ化されています。 たとえば、TERMINAL クラスには、端末として定義されたすべてのオブジェクト(tty1、tty2 など)が格納されています。SHARE クラスには、共有されるすべてのオブジェクトが格納されています。FILE クラスには、ファイルとディレクトリの定義が格納されています。

注: eTrust AC のクラスの詳細については、「リファレンス ガイド」を参照してください。

保護されているリソースのプロパティは、そのリソースのレコードに保存されています。レコードは、リソースの名前とプロパティで構成されたデータの集合です。 特定のクラスの各レコードには、同じプロパティ セット(そのクラスが表すオブジェクト タイプに対応する)の値が保存されています。

プロパティによって、リソースを定義したユーザ、定義した日などの情報がわかります。通常、リソース レコードに保存されている最も重要な情報は、そのリソースにアクセスする権限があるアクセサのリストです。 このリストは、「アクセス制御リスト(ACL)」といいます。 多くのリソースにはもう 1 つアクセサのリストがあり、このリストによりアクセスが拒否されます。 このリストは「Negative Access Control List(NACL)」といいます。

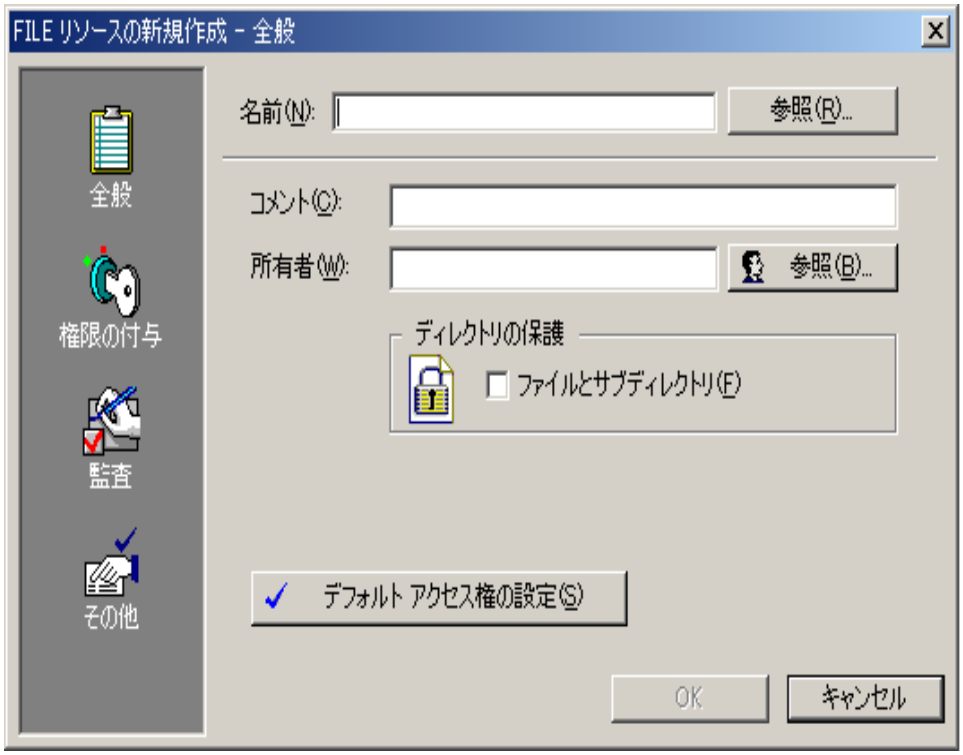
注: ユーザ名またはグループ名を右クリックしたときに表示されるメニューから[保護されたリソース]を選択すると、特定のユーザまたはグループの ACL または NACL を表示できます。

eTrust AC データベース内のすべてのリソースは、以下の操作を実行することで管理できます。

- eTrust AC データベースの任意のクラスにリソースを追加します
- eTrust AC データベースの任意のクラスのリソースを更新します
- eTrust AC データベースから任意のクラスのリソースを削除します
- ユーザがログオンできる端末および端末グループを定義する
- ログイン時に追加権限が必要な休日を定義する
- タスクの委任およびタスク グループを定義する

これらの機能を実行するには、プログラム バーの[Access Control]パネルで[リソース]をクリックし、ワークスペースでリソースを選択して、ツールバーの[新規作成]、[削除]、または[プロパティ]をクリックします。

次の図は、FILE クラスのリソースを作成するためのダイアログ ボックスを示しています([リソース]-[新規作成]をクリックします)。



左側にあるアイコンをクリックすると、別のパネルが表示されます。 たとえば、図に示されている[全般]パネルでは、リソースの名前と説明の入力や、所有者の指定などができます。

カレンダを使用した eTrust AC リソースの管理

eTrust AC では、Unicenter TNG カレンダに基づく、ユーザ、グループ、およびリソースへのアクセスをサポートします。 カレンダは、15 分間隔で ON または OFF に設定できます。 カレンダの時間間隔を OFF に設定すると、リソースへのアクセスが拒否されます。ON に設定すると、リソースへのアクセスが許可されます。 eTrust AC は、一定の時間間隔でアクティブな Unicenter TNG のカレンダを取得します。

カレンダのリソースは、[リソース]ビューを使用して追加、編集、または削除することができます。 リソース ツリーから[ログインの保護]を選択します。 [Calendar]ツリー エントリをクリックします。右クリックしてオプションを選択します。

Windows リソースの管理

ネイティブ Windows データベースのリソースは、リソースの作成および編集に使用するダイアログ ボックスによって管理できます。 次の操作を実行できます。

- Windows データベースの REGISTRY クラスおよび SHARE クラスにリソースを追加する。
- Active Directory データベースを含む、Windows データベースの任意のクラスのリソースを更新する。
- Windows データベースから任意のクラスのリソースを削除する。

注: Windows リソースの詳細については、「リファレンス ガイド」の章「Windows 環境のクラスとプロパティ」を参照してください。

Windows ドメインの管理

ポリシー マネージャの使用することで、次の操作を実行できます。

- Windows ドメインに関する情報を表示する。
- 新しいコンピュータを Windows ドメインに追加する。
- Windows ドメインからコンピュータを削除する。
- Windows ドメイン間の信頼関係を作成および削除する。

リソース ツリーから[NT 固有]を選択します。 [ドメイン]ツリー エントリをクリックし、右クリックしてオプションを選択します。

eTrust AC では、ポリシー マネージャや selang などの eTrust AC クライアントがこれらの操作を実行した場合、その操作の妥当性がチェックされます。 操作の妥当性のチェック時には、ドメイン コントローラ内の eTrust AC データベースにあるアクセス権限ルールが eTrust AC によって適用されます。

eTrust AC の DOMAIN クラスの各レコードによって、Windows のドメインを定義します。 DOMAIN クラスのレコードに対して、次の 3 タイプのアクセス方法を指定できます。

READ

ドメインのプロパティを表示できます

CHMOD(信頼関係の変更)

ドメイン間の信頼関係を作成または削除できます

EXEC(実行)

ドメインに対してメンバを追加または削除できます

プロセスの保護

PROCESS クラスのオブジェクトにより、eTrust AC による保護が必要なプロセス アプリケーションを定義します。

eTrust AC を使用してプロセスを保護するには、以下の手順に従います。

1. eTrust AC ポリシー マネージャを起動します。
2. プログラム バーから[Resources]を選択します。
3. eTrust AC リソース ツリーを展開して[System Resources]を表示し、[プロセス]を選択します。
4. 保護するプロセスを新しく追加するには、[Name]列を右クリックし、[New]を選択するか、キーボードの Insert キーを押します。
5. Windows のタスク マネージャ(taskmgr.exe)を起動します。 これ以降の手順を実行するには、タスク マネージャが実行されている必要があります。
6. [Name]フィールドの横にある[Browse]をクリックします。 保護するプロセスを選択します(taskmgr.exe)。 [OK]をクリックします。
7. [Owner]フィールドの横にある[Browse]をクリックします。
8. [nobody]を選択します。 [OK]をクリックします。
9. [Set Default Access]ボタンをクリックします。

   [Set Default Access]ダイアログが表示されます。
10. [None]が選択されていることを確認します。 [OK]をクリックします。
11. ルールをテストするには、タスク マネージャ(taskmgr.exe)を開き、[Processes]タブを選択します。[taskmgr.exe]を選択し、[End Process]をクリックします。
12. [Task Manager Warning]ダイアログ ボックスが表示されます。 [はい]をクリックします。

    ルールが正常に実行された場合は、[Unable to Terminate Process]というタイトルのダイアログ ボックスが表示されます。
13. 選択したプロセスを終了する権限があるユーザを追加するには、eTrust AC リソース ツリーを展開して[System Resources]を表示し、[プロセス]を選択します。
14. [Process]を右クリックして、[Properties]を選択します。
15. [Authorize]をクリックします。
16. [Insert]をクリックします。
17. [Name]フィールドの横にある[Browse]をクリックします。
18. [Users]または[Groups]タブを選択して、[OK]をクリックします。
19. [Read]権限の[Allow]チェック ボックスをオンにし、[OK]をクリックします。

20. 新しいルールをテストするには、プロセスを終了する権限があるユーザとしてログインします。
21. タスク マネージャ(taskmgr.exe)を開き、[Process]タブを選択します。
    [taskmgr.exe]を選択し、[End Process]をクリックします。

    プロセスが終了します。

注: Windows のサービスは、そのほとんどが GUI や対話形式のアプリケーションではなく、バックグラウンドで実行されるため、eTrust AC プロセスの保護対象として適しています。

SPECIALPGM によるリソースの保護

SPECIALPGM クラスのオブジェクトにより、eTrust AC による特別な権限保護が必要なアプリケーションを定義します。 このクラスは、システム サービスなど、通常はシステム アカウントで実行する必要のあるプログラムを保護する場合に特に有用です。 このようなプログラムを保護するには、SPECIALPGM クラスのレコードとして定義し、(eTrust AC データベースで USER レコードとして定義されている)論理ユーザ名をプログラムの実行に必要な Windows ユーザ名に関連付け、この論理ユーザのみにプログラムを実行する権限を付与します。

Windows では、SPECIALPGM 作成ウィザードを使用して、この保護を設定することができます。 このウィザードをウィンドウから実行するには、プログラム バーの[リソース]ボタンをクリックします。 次に、[ツール]メニューから[SPECIALPGM 作成ウィザード]を選択します。

Policy Model の管理

ポリシー マネージャを使用すると、いくつかの PMDB 機能を管理できます。 管理内容は、PMDB の指定、サブスクライバの管理、エラー ログの管理、Policy Model デーモンの起動と停止(UNIX の場合)、使用不可のサブスクライバの再アクティブ化、プロパティの表示などです。

PMDB の指定

eTrust AC では、単一ホスト上で複数の Policy Model をサポートします。 PMDB は、ポリシー マネージャまたは selang 使用して指定することができます。

[Policy Model]ウィンドウの表示

[Policy Model]ウィンドウは、プログラム バーの[ツール]パネルから開きます。このウィンドウには、接続先の端末上に定義された、適用可能なサブスクライバを含むすべての PMDB が一覧表示されます。

[Policy Model]ウィンドウには、次の列が表示されます。

[Name]

選択した PMDB のサブスクライバを一覧表示します。

[Type]

eTrust データベース、PMDB、MF(メインフレーム)などのサブスクライバのタイプを表示します。

[Status]

選択したサブスクライバに対して[AVAILABLE]または[UNAVAILABLE 使用不可]を表示します。 実行予定のコマンドがない場合は、[AVAILABLE 使用可能]と表示されます。 サブスクライバの親 PMDB によって、まだ実行されていない 1 つ以上のコマンドが送信されている場合は、[UNAVAILABLE 使用不可]と表示されます。 コマンドは、updates.dat ファイルに保存されます。デフォルトの格納場所は eTrustACDir¥data¥pmdb です(eTrustACDir は、eTrust AC をインストールしたディレクトリです)。

[次のコマンド]

実行予定のコマンドを表示します。

サブスクライバのステータスが Available の場合、この列は空になります。

[エラー]

選択したサブスクライバのエラー回数を表示します。 エラーとは、失敗したコマンド、つまりサブスクライバを更新しなかったコマンドのことです。 接続の失敗はこれに該当しません。

[実行されたコマンド]

実行されたコマンドの割合(%)を表示します。 サブスクライバのステータスが Available の場合、この列には 100% と表示されます。

Policy Model 階層の管理

PMDB のサブスクライバには、次のデータベースを指定できます。

- 同じホストまたはリモート ホスト上の別の PMDB
- 同じホストまたはリモート ホスト上の eTrust データベース
- メインフレームのデータベース

ポリシー マネージャの使用することで、次の操作を実行できます。

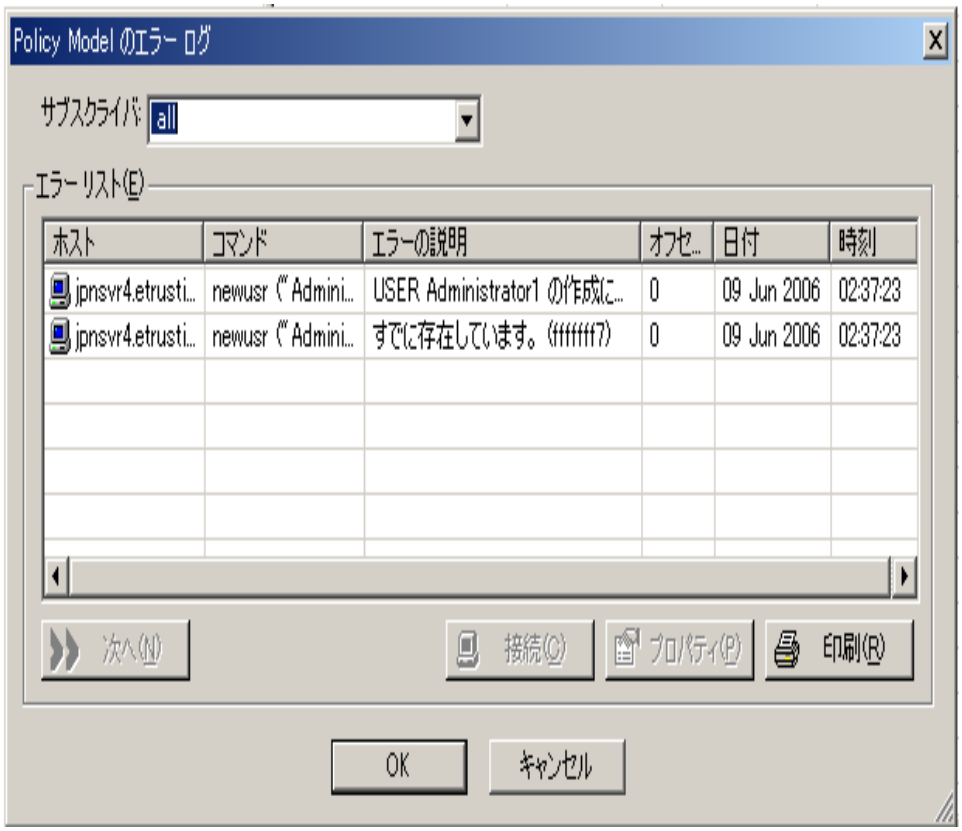
- サブスクライバを PMDB に追加する。
- PMDB からサブスクライバを削除する。
- サブスクライバに送信されたが、更新に失敗したコマンド（エラー ログに表示されるエラー）を表示します。
- エラー ログの内容を消去する。

サブスクライバを追加する際には、親 PMDB とこれにサブスクライブするすべての端末が同じネットワークの構成要素で、名前によって相互に通信できることを確認してください。この通信によって、eTrust AC はサブスクライバのレジストリにある parent_pmd キーを更新できます。

エラー ログの使用

この Policy Model のエラー ログには、サブスクライバ端末が適用を拒否したトランザクションのリストが保存されています。

ポリシー マネージャを使用して、PMDB とその全サブスクライバのエラーを表示できます。また、サブスクライバごとのエラー表示も可能です。 さらに、エラー ログの内容を消去することもできます。



[Policy Model Error Log]ダイアログ ボックスには、次の列が表示されます。

[ホスト]

コマンドが失敗した PMDB のフルネーム。

[コマンド]

失敗した eTrust AC のフル コマンド。

[エラーの説明]

コマンドが失敗した理由。

[オフセット]

updates.dat ファイル内のコマンドの位置。

[日付]

コマンドが失敗した日付。

[時刻]

コマンドが失敗した時刻。

注: [次へ]ボタンをクリックすると、次のレコード セットが eTrust AC によって表示されます。 query_size レジストリ キーによって、1 セット内のレコード数が定義されます (デフォルト値は 100 です)。次のレコード セットは表示画面に追加されます。 つまり、[Next]を 1 回クリックすると(キーの値が 100 のままである場合)、200 件のレコードが表示されます。

プロパティの表示

PMDB またはサブスクライバのプロパティを表示するには、[表示]メニューまたは右クリックして開くメニューから[プロパティ]を選択します。

次に、親 PMDB に対して表示されるプロパティについて説明します。

[Policy Model 名]

PMDB の名前。

[親 Policy Model]

PMDB が親であるかどうかを示します。

[パスワード ファイル]

ローカルで定義されたユーザの情報(ユーザのフルネーム、ID、ユーザが属するグループの ID、ユーザのホーム ディレクトリ、暗号化されたパスワードなど)が含まれているファイルの名前。UNIX の場合のみ該当します。

[グループ ファイル]

ローカルで定義されたグループの情報(グループ ID、グループに属するユーザのリストなど)が含まれているファイルの名前。UNIX の場合のみ該当します。

eTrust AC では、[Policy Model]ウィンドウ (P. 53)を使用してサブスクライバのプロパティを表示します。

eTrust AC for Windows による UNIX の管理

ポリシー マネージャを使用して、eTrust AC がインストールされている UNIX マシンを管理できます。 UNIX 環境および eTrust AC 環境で定義したユーザ、グループ、リソース、および PMDB 階層を管理できます。

管理者リソース

ADMIN クラス

ポリシー マネージャのクラスによるアクセス機能を使用して、ADMIN クラスにオブジェクトを追加できます。

- ADMIN クラスの各レコードには、ADMIN 以外のユーザに対して特定のクラスの管理を許可するための定義が含まれます。
- 1 つの ADMIN レコードが、委任されたユーザによって管理される eTrust AC の各クラスを表します。
- このクラスのレコードには、レコード単位のアクセス権限を持つアクセサのリストが保存されます。
- ADMIN クラスのレコードのキーは、保護対象クラスの名前です。

例

ADMIN クラスの機能を実行するように ADMIN 以外のユーザを定義する場合には、次の例を参考にしてください。

注: 次の例では、2 台のワークステーションを必要とします。 ここでは、ローカル端末を **computer1**、リモート端末を **computer2** とします。 これらのコンピュータ名は、対象の環境によって異なります。

1. ローカル端末(computer1)で、sub_admin という名前のユーザを作成します。
   a. eTrust AC を起動します。
   b. プログラム バーの[User]をクリックし、新しいユーザを作成します。
   c. [User Name]フィールドに「sub_admin」と入力します。
   d. [Full Name]フィールドに「sub_admin」と入力します。
   e. [Description]フィールドに「sub_admin」と入力します。
   f. [Password]フィールドに、今後使用するパスワードを入力します。
   g. [Advanced]ドロップダウン メニューをクリックします。 次の両方のチェックボックスがオンになっていることを確認します。
      - [Create in Native OS environment]
      - [eTrust AC 環境で作成]
   h. [OK]ボタンをクリックします。

2. リモート端末（computer2）で、sub_admin という名前のユーザを作成します。
   a. eTrust AC を起動します。
   b. プログラム バーの［User］をクリックし、新しいユーザを作成します。

      注：リモート アクセス ユーザ名を次の形式で入力します。

      ```
      computer_name\user_name
      ```

   c. ［User Name］フィールドに「computer1¥sub_admin」と入力します。
   d. ［Full Name］フィールドに「computer1¥sub_admin」と入力します。
   e. ［Description］フィールドに「computer1¥sub_admin」と入力します。
   f. ［Advanced］ドロップ ダウン メニューをクリックします。［eTrust AC 環境で作成］のみを選択します。［OK］をクリックします。
3. リモート端末（computer2）で、ローカル端末（computer1）からのアクセスを許可します。
   a. リモート端末（computer2）で、eTrust AC のプログラム メニューをクリックし、［リソース］をクリックします。
   b. ［Login Protection］ツリーを展開し、［Terminal］を選択して、［Name］列の空白のフィールドを右クリックし、［New］を選択します。
   c. ローカル端末（computer1）の情報を入力します。
   d. ［Name］フィールドにローカル端末（computer1）の名前を入力します。
   e. ［Comment］フィールドに「Terminal (Computer1)」と入力します。
   f. ［所有者］フィールドに「nobody」と入力します。
   g. ［デフォルト アクセス権の設定］をクリックし、None が選択されていることを確認します。
   h. ［OK］をクリックします。
   i. ［Authorize］をクリックします。
   j. ［Insert］をクリックします。
   k. ［名前］フィールドの横にある［参照］ボタンをクリックします。
   l. 前の手順で作成したユーザ computer1¥sub_admin を選択して［OK］をクリックし、［Add/Edit eTrust Accessor］ダイアログ ボックスに戻ります。
   m. ［Add/Edit eTrust Accessor］ダイアログ ボックスが表示されます。［OK］をクリックします。
   n. 今追加したユーザを選択し、［Read］権限の［Allow］チェック ボックスをオンにして、［OK］をクリックします。

4. リモート端末（computer2）で、[Administration]ツリーを展開し、[Access by Class]を選択して、[USER]クラスを選択します。

   a. [USER]クラスを右クリックし、[Properties]を選択して、[Authorize]をクリックします。

   b. [Insert]をクリックします。

   c. [Name]フィールドの横にある[Browse]をクリックします。

   d. 前の手順で作成したユーザ computer1¥sub_admin を選択して[OK]をクリックし、[Add/Edit eTrust Accessor]ダイアログ ボックスに戻ります。

   e. [Add/Edit eTrust Accessor]ダイアログ ボックスが表示されます。[OK]をクリックします。

   f. 今追加したユーザを選択し、[Read]権限の[Allow]チェック ボックスをオンにして、[OK]をクリックします。

5. 両方のコンピュータの eTrust AC が同じネットワーク上で実行されていることを確認します。

6. ローカル マシン（computer1）で、ユーザ sub_admin として Windows にログインします。

7. ローカル マシン（computer1）で eTrust AC を起動します。 左上部にある接続アイコンをクリックします。

8. リモート端末のホスト（computer2）の名前を入力し、[OK]をクリックします。

9. [Connection Information]ダイアログ ボックスが表示されます。[OK]をクリックします。

   [OK]をクリックすると、ポリシー マネージャが開きます。 リモート端末（computer2）に接続されます。

10. ポリシー マネージャで[ユーザ]をクリックします。

11. 次のエラー メッセージが表示されます。

12. ［Access by Class］は、サブ管理に使用することもできます。

    サブ管理者によって管理者特有のクラスおよびリソースを管理するには、リモート端末（computer2）のみで、以下の手順に従います。

    a. ［Tools］メニューの［Options］を選択し、［Startup］タブを選択します。

    b. ［ユーザとグループのサブ管理を有効にする］チェック ボックスをオンにし、［OK］をクリックします。

13. ［Users］を再度クリックします。

    ユーザのリストとプロパティが表示されます。 ユーザのプロパティを変更しようとすると、「Operation not allowed」というエラー メッセーが表示されます。 これは、ADMIN クラスまたは USER クラスには、ユーザは読み取り権限のみが設定されているためです。

    「Operation not allowed」というエラー メッセージは、［Groups］をクリックしても表示されます。これは、ユーザが ADMIN クラスまたは GROUP クラスに対して権限がないためです。

14. ［Group Properties］ダイアログ ボックスでグループおよびファイルを参照するには、［Set Admin Properties］ダイアログ ボックスで読み取り権限をオブジェクト グループに追加する必要があります。

    a. リモート端末で、［Administration Resource］ツリーを展開し、［Access by Class］を選択して、［Group］クラスを右クリックし、［Properties］を選択します。

    b. ［Authorize］をクリックし、［Insert］をクリックし、作成した端末を選択し、ユーザの権限を設定します。

    c. ローカル端末からリモート端末に接続し、［Group］をクリックします。

    d. ［Group］クラスが表示されます。

CONTAINER クラス

CONTAINER クラスの各レコードは、他のリソース クラスのオブジェクト、ファイル、およびユーザのグループ化を定義します。これにより、複数の異なるクラスのオブジェクトに 1 つのルールを適用する場合に、アクセス ルールを定義する作業が簡略化されます。CONTAINER クラスのレコードのメンバとしては、eTrust AC クラスのオブジェクト、ファイル、およびユーザを指定できます。

例

この例では、CONTAINER クラスに MyContainer という名前のオブジェクトを作成します。 このオブジェクトに、FILE クラス(c:¥winnt¥notepad.exe)および TERMINAL クラス(MyComp.ca.com)から 1 つずつ、2 つのメンバを登録します。 次に、John の読み取りまたは実行権限を MyContainer に付与または許可します。

1. [System Resources]ツリーを展開し、[File]を右クリックし、[New]をクリックして、次の情報を入力します。
   a. [Name]フィールドに、「c:¥winnt¥notepad.exe」と入力します。
   b. [Comment]フィールドに「Container Test Rule」と入力します。
   c. [Owner]フィールドに「nobody」を指定します。
   d. [Set Default Access]に[None]を設定します。

2. ［Administration Resources］ツリーを展開し、［Container］を右クリックし、［New］をクリックし、次の情報を入力します。

   a. ［Name］フィールドに「MyContainer」と入力します。

   b. ［Comment］フィールドに「MyContainer test」と入力します。

   c. ［Owner］フィールドに「nobody」を指定します。

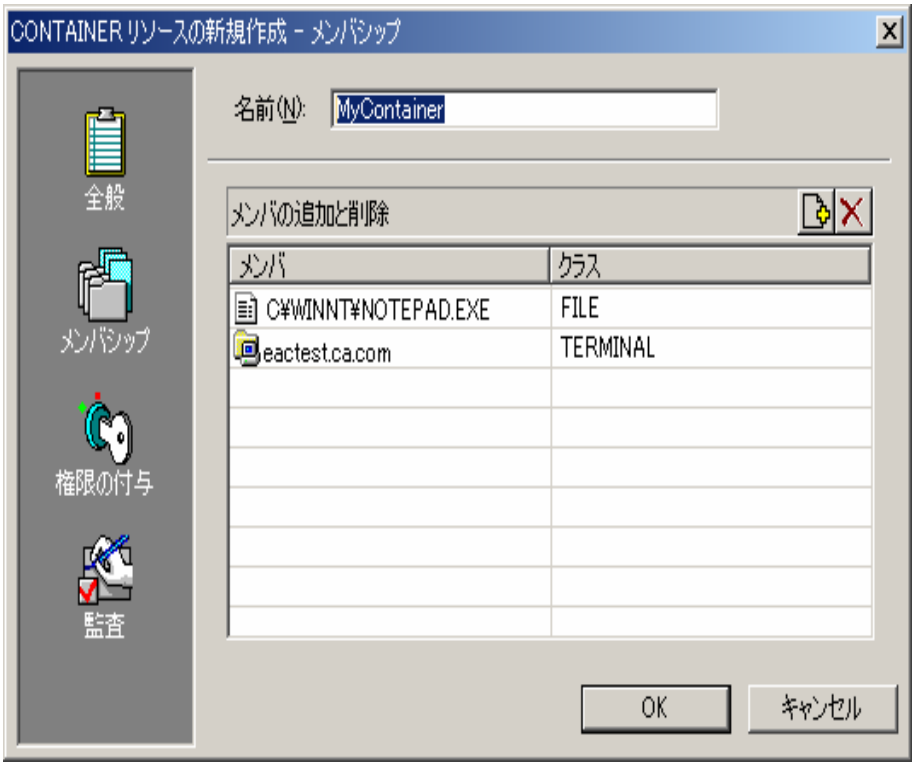
3. ［Membership］をクリックして、［Members］列の空白のフィールドを右クリックし、［Add］をクリックします。

4. ［FILE］クラスを選択し、作成したファイル ルール（c:¥winnt¥notepad.exe）を強調表示し、［OK］をクリックします。

5. ［Members］列の空白のフィールドを右クリックし、［Add］をクリックします。

6. ［TERMINAL］クラスを選択し、追加する端末の名前を選択し、［OK］をクリックします。

   コンテナ リソース ダイアログ ボックスが、次のように表示されます。



7. ［Authorize］をクリックし、［Insert］をクリックします。［Name］フィールドの横にある［Browse］をクリックし、ユーザを選択します。［OK］をクリックします。

   ［Add/Edit eTrust Accessor］ダイアログ ボックスが表示されます。

8. ［OK］をクリックします。

9. 選択したユーザに[Read]および[Execute]権限を付与します。

10. 前の手順で指定したユーザとしてログインします。 MyContainer でこのユーザに読み取りアクセスを許可しているので、ログインは成功します。
11. c:¥winnt¥notepad.exe を起動し、実行させます。
12. c:¥winnt¥notepad.exe の削除を試みます。 この操作は拒否されます。
13. eTrust AC クライアントである、ポリシー マネージャまたは selang のセッションを開くことを試みます。 MyContainer で John に書き込みアクセスが許可されていないため、この操作は拒否されます。
14. MyContainer で John のユーザ権限を全て[Deny]に変更します。

    John によるログインは拒否されるようになります。 また、c:¥winnt¥notepad.exe の実行も拒否されます。

サブ管理者の作成

サブ管理者を設定してポリシー マネージャからユーザとグループを管理するには、以下の手順に従います。

1. ポリシー マネージャを起動します。

   注: このマシンに eTrust AC サーバがインストールされている場合は、ポリシー マネージャにログインし、eTrust AC サービスを停止してください。

2. ポリシー マネージャのツールバーで、[ツール]メニューから[オプション]を選択します。

   [オプション]ダイアログ ボックスが表示されます。

3. [スタートアップ]タブを選択し、[ユーザとグループのサブ管理を有効にする]チェック ボックスをオンにします。

4. [OK]をクリックします。

サブ管理者が特定の端末からポリシー マネージャにアクセスできるようにするには、以下の手順に従います。

1. eTrust AC のプログラム バーの[リソース]アイコンをクリックし、[リソース]ウィンドウを表示します。

2. [ログインの保護]フォルダを展開します。

3. [端末]フォルダを選択し、使用可能な端末のリストを表示します。

4. 選択する端末をダブルクリックします。[端末プロパティの表示と設定 - 全般]ダイアログ ボックスが表示されます。

5. [権限の付与]アイコンをクリックし、[端末プロパティの表示と設定 - 権限の付与]ダイアログ ボックスを表示します。

6. 権限を与えるサブ管理者を選択し、[読み取り]および[書き込み]許可のチェック ボックスをオンにします。

7. [OK]をクリックします。

ユーザを管理する権限をサブ管理者に対して定義するには、以下の手順に従います。

1. eTrust AC のプログラム バーの[リソース]アイコンをクリックし、[リソース]ウィンドウを表示します。

2. [管理]フォルダを展開します。

3. [クラスによるアクセス]を選択し、使用可能なクラスのリストを表示します。

4. USER クラスをダブルクリックし、[プロパティ]を選択します。[ADMIN プロパティの表示と設定 - 全般]ダイアログ ボックスが表示されます。

   注: サブ管理者が他のクラスを管理できるようにするには、USER クラスを他のクラス(GROUP、USER_DIR など)に置き換えます。

5. ［権限の付与］アイコンをクリックし、［ADMIN プロパティの表示と設定 - 権限の付与］ダイアログ ボックスを表示します。
6. ［追加］をクリックし、［eTrust AC アクセサの追加］ダイアログ ボックスを表示します。
7. ［名前］フィールドにサブ管理者の名前を入力するか、［参照］をクリックして名前を参照します。
8. サブ管理者に与える権限を確認します。
9. ［OK］をクリックし、［ADMIN プロパティの表示と設定 - 権限の付与］ダイアログ ボックスに戻ります。
10. OK をクリックして終了します。

第 4 章: ユーザ パスワードの管理

このセクションには、以下のトピックが含まれます。

パスワード管理ユーティリティ

ユーザ パスワードの管理には、次のソフトウェアを使用できます。

Password Manager

ユーザ パスワードを設定または変更できます。

パスワード マネージャは、ポリシー マネージャが実行されていないマシンにインストールできるパスワード管理用の独立したユーティリティです。 パスワード マネージャをインストールすると、管理者以外のユーザにパスワード管理タスクを簡単に割り当てることができるようになります。

ポリシー マネージャ

Windows に定義されたユーザを作成または更新するたびに、ユーザ パスワードを設定または変更できます。

ポリシー マネージャ (P. 39)を使用して、パスワード ポリシーを設定することもできます。

selang のコマンド

selang のコマンド newusr、editusr、および chusr を使用して、ユーザのパスワードを設定できます。

注: これらのコマンドの詳細については、「リファレンス ガイド」を参照してください。

Windows ユーティリティ

Windows のユーザ マネージャを使用するか、またはコマンド プロンプト ウィンドウで Windows コマンドを使用して、ユーザ パスワードを管理できます。

注: 詳細については、関連する Windows のマニュアルを参照してください。

パスワード マネージャ、ポリシー マネージャ、または selang を使用してユーザのパスワードを設定または変更する場合、eTrust AC はデータベースにパスワードを追加する前にパスワード ルールをチェックしません。 eTrust AC では、これらの機能を使用して入力するパスワードはすべて許可されます。 このため、新しいパスワードが、eTrust AC のパスワード ルールに基づいた有効なパスワードではない可能性があります。
Windows のユーザ マネージャまたは eTrust AC 以外のソフトウェアを使用してパスワードを変更した場合、eTrust AC では、新しいパスワードの有効性がチェックされます。

パスワードおよびロックアウト ポリシーの管理

パスワードは最も一般的な認証手段ですが、パスワードの保護方法には、以下のようなよく知られた問題があります。 たとえば、簡単なパスワードは推測されやすい、同じパスワードを長期間使用したり、繰り返し使用すると解読されやすい、ネットワーク経由でパスワードを平文で送信すると盗まれる危険性があるなどです。

Windows には独自のパスワード ルールとポリシーがあり、それに準拠したパスワードをユーザが使用することで、このような問題のほとんどを回避できます。 eTrust AC に追加されたルールでは、より安全なパスワードをユーザが確実に選択することができます。

eTrust AC で指定できるルールは次のとおりです。

- 新しいパスワードは以前に使用したものと一致してはいけない。 eTrust AC に格納される使用済みのパスワードの数は、パスワード ポリシーで指定されます。
- 新しいパスワード中にユーザ名を使用することはできない。
- 新しいパスワードは変更前のパスワードを含むことはできない。
- 新しいパスワードと変更前のパスワードは違う必要があります。eTrust AC では、大文字と小文字は区別されません。
- 新しいパスワードには、パスワード ポリシーで指定されている、英数字、特殊文字、数字、小文字、および大文字を、それぞれ最低文字数以上使用しなければならない。
- 新しいパスワードで繰返し使用される文字の数が、パスワード ポリシーで指定されている数を超えてはいけない。
- eTrust AC の辞書で使用が禁止されている単語を、新しいパスワードに使用することはできません。辞書は以下のレジストリ サブキーの Dictionary 値で指定されています。

  `HKEY_LOCAL_MACHINE\Software\ComputerAssociates\eTrustAccessControl\passwd`

パスワードごとに、最長有効期限を指定する必要があります。つまり、有効期限を過ぎたパスワードは失効し、ユーザが新しいパスワードを選択する必要があります。

- パスワードごとに、最短有効期限を指定する必要があります。 最短有効期限を指定すると、ユーザが短期間に何度もパスワードを変更することを防止できます。 パスワード変更が頻繁に行われると、パスワード履歴スタックがオーバーフローし、使用済みパスワードが再使用される場合があります。

Password Manager の使用法

Password Manager がインストールされている端末では、Password Manager を使用して新しいパスワードを設定したり、既存のパスワードを変更することができます。

パスワードの設定または変更を行うと、次のことを実行できます。

- ユーザが次回ログオンするときにパスワードの変更を要求する。
- ロックされたユーザ アカウントをリセットして、ユーザのロックアウトを解除する。
- リモート ホスト上のユーザ、ローカル ホストまたはリモート ホストの PMDB に登録されているユーザのパスワードを設定できるように、ターゲット ホストを変更する。eTrust AC のデフォルトでは、ユーザはローカル ホスト上にあると想定されています。
- 組織のパスワード ポリシーに適合するユーザ パスワードを生成するように eTrust AC を設定できます。
- 管理者がパスワードを変更した場合は、新しいパスワードをユーザに電子メールで通知するように eTrust AC を設定できます。

パスワードの生成

Password Manager では、既存ユーザのパスワードを生成できます。 eTrust AC によって生成されたパスワードは、常に、サイト用に確立された基準に適合します。 パスワードを生成するには、システム オプションとして[Enable Password Generation]を選択する必要があります。 パスワードの生成オプションはデフォルトで選択されます。

ターゲット ホストの変更

ターゲット ホストを変更することにより、ローカル ホストやリモート ホストの PMDB に登録されているユーザまたはリモート ホスト上のユーザのパスワードを設定できます。

ユーザ パスワード変更の設定

ポリシー マネージャでは、ユーザにパスワードの変更を促す機能を設定できます。ユーザがポリシー マネージャにログインするときに、パスワードの変更を促すダイアログ ボックスが表示されます。［はい］をクリックすると、［パスワードの変更］ダイアログ ボックスが表示されます。

この機能を設定するには、以下の手順を実行してください。

1. Access Control のプログラム バーの［リソース］アイコンをクリックします。
2. ［ツール］メニューから［eTrust クラスの有効化］を選択します。［パスワード］および［パスワードの変更］チェック ボックスをオンにします。
3. ［OK］をクリックします。

エラー メッセージの解決

Windows システム上でユーザのパスワードを設定している場合、以下のメッセージが表示されることがあります。

パスワードが必要な長さよりも短い。

このエラーは、パスワードがポリシー要件を満たしていないことを意味します。 このエラーの原因は、以下のいずれかです。

- パスワードが必要な長さよりも短いか、または長い。
- パスワードが最近使用されており、Windows NT Change History フィールドに存在する。
- パスワードに充分に一意である文字が含まれていない。
- パスワードが他のパスワード ポリシー要件（eTrust AC パスワード ポリシーで設定された要件など）を満たしていない。

このエラーを回避するには、該当するすべての要件を満たすパスワードを設定するようにしてください。

第 5 章: アカウントの保護

このセクションには、以下のトピックが含まれます。

別のユーザとしての実行要求

アカウントからログインが行われた後は、システム リソースに対して許可されている機能のみが実行されるようにアカウントを監視する必要があります。 オペレーティング システムでは、アクセサのユーザ SID に基づく、ある程度のファイル保護機能が用意されています。 この保護機能を使用しないようにするには、ユーザはまず別のユーザ SID として実行する必要があります。 未許可の別のユーザとしての実行要求を防ぐために、オペレーティング システムでは、要求したユーザに対して、ターゲット ユーザのパスワードが要求されます。

この仕組みには、多くの欠陥があります。 別のユーザ SID として実行するユーザは、ターゲット ユーザのパスワードを記憶するか、書き留めるか、または簡単なパスワードを使用するようにターゲット ユーザに依頼する必要があります。 このような行為は、いくつかのパスワード ポリシーに反します。 さらに、アカウンタビリティが実質的に失われ、特定ユーザの ID を変更したユーザを識別できません。 さらに、スーパーユーザのパスワードがユーザに知られた場合、すべてのセキュリティは機能せず、そのユーザはシステムに無制限にアクセスできることになります。

eTrust AC では、より高度な方法で別のユーザとしての実行を保護します。ユーザは、SID 変更が特定のルールで許可されている場合のみ、自分の SID を別のユーザの SID に変更できます。

たとえば、ユーザ X がユーザ Y としてあるタスクを実行するプログラムを起動しているとします。ユーザ X がユーザ Y のパスワードを知っていても、ユーザ Y になることを許可されていない場合は、プログラムの要求は拒否されます。

このように、管理者ユーザのパスワードを知っているだけでは、管理者ユーザにはなれません。管理者ユーザになることを許可するルールもデータベース内に設定されている必要があります。

各ユーザ SID およびグループ SID のアクセス ルールはデータベースに定義できます。 eTrust AC では、この種の保護に SURROGATE クラスを割り当てています。 初期段階で、すべての別のユーザとしての実行要求にアクセスを許可する場合は、次のコマンドを入力します。

```
eTrust> editres SURROGATE _default defaccess(READ)
```

このコマンドは、別のユーザ ID としての実行要求があり、データベース内のレコードでユーザ ID 一時変更が明示的に保護されていない場合、アクセスを許可するように eTrust AC に指示します。

SID をスーパーユーザ SID に変更できないようにするには、以下のコマンドを入力します。

```
eTrust> newres SURROGATE USER.Administrator defaccess(NONE)
```

このコマンドは、管理者ユーザ名を保護するように、および管理者ユーザ名の使用を明示的に許可されていないユーザが別のユーザとしての実行で管理者ユーザになれないように、eTrust AC に指示します。 セキュリティ管理者に管理者ユーザの使用を許可するには、以下のコマンドを使用して明示的に許可を指定する必要があります。

```
eTrust> authorize SURROGATE USER. Administrator gid("Security Admins")
```

注:

- ユーザの SURROGATE レコードで、特定ユーザによる ID 変更が明示的に許可されていない場合、ユーザにはそのレコードのデフォルト アクセス権が与えられます。上記の例では、デフォルトは NONE です。これは、許可のないユーザは別のユーザとしての実行で管理者ユーザになれないことを意味します。
- USER._default というレコードは、独自のレコードを持たないすべてのユーザを表します。 同様に、GROUP._default というレコードは、独自のレコードを持たないすべてのグループを表します。 特定アクセサの SURROGATE レコードが存在しない場合、そのアクセサになる要求は、SURROGATE USER._default レコード、SURROGATE GROUP._default レコード、または _default レコード（ユーザおよびグループの両方）に指定されているデフォルト設定に従って処理されます。
- _default レコードのデフォルト値は READ です。未定義の SURROGATE レコードは、別のユーザとしての実行でユーザになれる許可を暗黙的に示します。 このデフォルト設定は、「eTrust AC で定義されていないものはすべて、eTrust AC では保護しない」という、実装時の一般的なルールに沿ったものです。このルールは、実装段階が終了した後に、「eTrust AC で許可されていないものはすべて、eTrust AC では自動的に禁止する」という逆のルールに変更できます。
- Windows の多くのユーティリティとサービス（例： 別のユーザとして実行）では、それらを実行した元のユーザではなく、ユーザ「NT AUTHORITY¥SYSTEM」として識別されます。これらのユーティリティとサービスを使用するユーザが別のユーザとして実行できるようにするには、eTrust AC データベースにこの SYSTEM ユーザを作成し、ターゲット ユーザとして実行する権限を与えます。 次に例を示します。

  ```
  auth SURROGATE USER.Administrator uid("NT AUTHORITY\SYSTEM") acc(R)
  ```

Surrogate DO 機能のセットアップ

多くの場合、オペレータ、プロダクション担当者、およびエンド ユーザは、スーパーユーザのみが実行できるタスクを実行する必要があります。

これまでの方法では、これらのタスクを実行する必要があるすべてのユーザに、スーパーユーザのパスワードを知らせていました。これはサイトのセキュリティを脅かすことにつながります。 このため、安全な代替策としてパスワードの公開を禁止すると、システム管理者はユーザからの正当な要求によってさまざまなルーチン タスクを実行しなければならず、システム管理者の負荷が大きくなります。

Surrogate DO（sesudo）ユーティリティは、このジレンマを解消します。 このユーティリティは、SUDO クラスに定義されているアクションの実行をユーザに許可します。SUDO クラスの各レコードにはスクリプトが保存されていて、スクリプトを実行できるユーザとグループが指定されています。それらのユーザやグループに、目的に応じて必要な許可が与えられます。

たとえば、ユーザがシステム ユーザであるかのように、「Print Spooler」サービスを起動する SUDO リソースを定義するには、以下のコマンドを入力します。

```
eTrust> newres SUDO StartSpooler data("net start spooler")
```

この newres コマンドによって、一部のユーザだけが実行のシステム権限を使用できる保護されたアクションとして、StartSpooler が定義されます。

重要: data プロパティには、完全な絶対パス名を使用してください。相対パス名を使用すると、保護されていないディレクトリに仕掛けられたトロイの木馬プログラムが、誤って実行される可能性があるからです。

さらに、authorize コマンドを使用して、StartSpooler アクションを実行する権限をユーザに与えることもできます。 たとえば、ユーザ operator1 に「Print Spooler」サービスの起動を許可するには、以下のコマンドを入力します。

```
eTrust> authorize SUDO StartSpooler uid(operator1)
```

また、authorize コマンドを使用して、保護されたアクションの実行をユーザに対して明示的に禁止することもできます。 たとえば、ユーザ operator2 に「Print Spooler」サービスの起動を許可しないようにするには、以下のコマンドを入力します。

```
eTrust> authorize SUDO StartSpooler uid(operator2) access(None)
```

sesudo ユーティリティを実行すると、保護されたアクションが実行されます。 たとえば、ユーザ operator1 が「Print Spooler」サービスを起動するには、以下のコマンドを入力します。

```
cmd> sesudo -do StartSpooler
```

この sesudo ユーティリティは、最初に SUDO アクションの実行権限がユーザにあるかどうかをチェックし、そのユーザにリソースの権限がある場合は、そのリソースに定義されているコマンド スクリプトを実行します。 この例に示した sesudo は、StartSpooler アクションの実行権限が operator1 にあるかどうかをチェックした後に、「net start spooler」コマンドをシステム権限で起動します。

注: sesudo ユーティリティの詳細については、「ユーティリティ ガイド」を参照してください。 SUDO レコードの data プロパティの書式設定の詳細については、「リファレンス ガイド」にある chres、editres、および newres のコマンドの説明を参照してください。

ユーザ非アクティブ状態のチェック

ユーザの非アクティブ状態をチェックする機能を使用して、不在または会社を退職したユーザのアカウントを使用した不正なアクセスからシステムを保護します。 非アクティブな日とは、ユーザがログインできない日のことです。ユーザ アカウントを一時停止してログインできなくするまでの非アクティブな日数を指定できます。一時停止したアカウントは、手動で再びアクティブにする必要があります。

注: 非アクティブ状態のチェックでは、パスワード変更はアクティビティとしてカウントされます。 ユーザのパスワードが変更された場合、非アクティブ状態を理由としてそのユーザのアカウントを一時停止することはできません。

非アクティブ日数は、USER クラスまたは GROUP クラスのレコードの inactive プロパティを使用して設定できます。 GROUP クラスのレコードでの設定は、そのグループがプロファイル グループであるユーザのみに影響します。 SEOS クラスの INACT プロパティを使用して、システム全体のすべてのユーザに非アクティブ状態を設定することもできます。

非アクティブ状態は、selang および Security Administrator の両方で設定できます。 selang では、次のコマンドを使用して、非アクティブ状態をグローバルに指定します。

```
setoptions inactive ( numdays)
```

非アクティブ日数をグループに設定するには、以下のコマンドを使用します(この設定は、そのグループに対するシステム全体の非アクティブ設定よりも優先されます)。

```
{chgrp | editgrp | newgrp} groupName inactive ( numdays)
```

非アクティブ日数をユーザに設定するには、以下のコマンドを使用します(この設定は、そのユーザに対するグループおよびシステム全体の設定よりも優先されます)。

```
{chusr | editusr | newusr} userName inactive ( numdays)
```

一時停止しているユーザ アカウントを再びアクティブにするには、以下のコマンドを使用します。

```
{chusr | editusr} userName resume
```

一時停止しているプロファイル グループを再びアクティブにするには、以下のコマンドを使用します。

```
{chgrp | editgrp} userName resume
```

システム全体レベルで非アクティブ ログイン チェックを無効にするには、以下のコマンドを使用します。

```
setoptions inactive-
```

グループに対する非アクティブ ログイン チェックを無効にするには、以下のコマンドを使用します。

```
{chgrp | editgrp} groupName inactive-
```

ユーザに対する非アクティブ ログイン チェックを無効にするには、以下のコマンドを使用します。

```
{chusr | editusr} userName inactive-
```

第 6 章: ポリシーの一元管理

このセクションには、以下のトピックが含まれます。

Policy Model データベース

何百ものデータベースを個別に管理することは、現実的ではありません。 eTrust AC には、1 台の中央データベースから多数のデータベースを管理できるコンポーネントである Policy Model サービスが用意されています。 Policy Model(PMD)サービスの使用は任意ですが、このサービスを使用すると、大規模なサイトでの管理を大幅に簡略化できます。

注: Windows のタスク マネージャでは、Policy Model サービスは sepmdd.exe と表示されます。

Policy Model サービスは、Policy Model データベース(PMDB)を使用します。 PMDB には、他の eTrust AC データベースと同様に、ユーザ、グループ、保護されているリソース、およびリソースへのアクセスを管理するルールが保存されています。 PMDB にはこの他に、サブスクライバ データベースのリストが含まれます。 各サブスクライバは、別々のコンピュータに存在する eTrust AC データベース、あるいは、同じコンピュータまたは別のコンピュータに存在する別の PMDB です。 サブスクライバを更新する PMDB は、サブスクライバの親です。

PMDB は、同様の許可制約およびアクセス ルールが適用される多数のデータベースを管理するための便利なツールです。

注: sepmd ユーティリティを使用した PMDB の管理および PMDB のリモート管理の方法については、「リファレンス ガイド」を参照してください。

ディスク上の PMDB の場所

1 台のコンピュータ上にある PMDB はすべて、共通ディレクトリに保存されます。 ディレクトリ名は、次の Windows レジストリ サブキーの_pmd_directory_値で指定します。

```
HKEY_LOCAL_MACHINE¥Software¥ComputerAssociates¥eTrustAccessControl¥Pmd
```

NTFS ルート ディレクトリの _pmd_directory_ のデフォルト値は、eTrustACDir¥data です。eTrustACDir は eTrust AC のインストール ディレクトリです（デフォルトでは C:¥Program Files¥CA¥eTrustAccessControl）。

各 PMDB は、共通ディレクトリ内のサブディレクトリに格納されます。 サブディレクトリ内のファイルには、Policy Model を定義するために必要なすべてのデータが含まれています。 Policy Model の環境設定は、eTrust AC のレジストリ設定の Pmd サブキーに格納されます。 Policy Model の名前がそのままサブキーの名前になります。

ローカル PMDB の管理

eTrust AC には、PMDB を管理するためのユーティリティが用意されています。

sepmd

以下を実行できる PMDB 管理ユーティリティ。

- サブスクライバの管理
- 更新ファイルの切り捨て
- 二重チェックの管理
- Policy Model のログ ファイルの管理
- その他の管理タスクの実行

注: sepmd の詳細については、「リファレンス ガイド」を参照してください。

リモート PMDB の管理

eTrust AC には、pmd 環境で使用できるさまざまな selang コマンドも用意されています。 これらのコマンドを使用して、PMDB をリモートで管理できます。

createpmd

PMDB を作成します。

deletepmd

PMDB を削除します。

findpmd

コンピュータ上のすべての PMDB の名前を表示します。

listpmd

PMDB に関する以下の情報を表示します。

- サブスクライバおよびそのステータス
- PMDB の説明およびそのステータス
- 更新ファイル内のコマンドおよび各コマンドのオフセット
- エラー ログの内容

pmd

以下を実行できる PMDB 管理コマンド。

- 使用不可のサブスクライバのリストからのサブスクライバの削除
- Policy Model のエラー ログの消去
- Policy Model サービスの開始と停止
- 更新ファイルの切り捨て
- レジストリ設定の再ロード

subs

以下を実行できる PMDB サブスクリプション コマンド。

- 親 PMDB へのサブスクライバの追加
- データベース(eTrust AC または別の PMDB)への親 PMDB の割り当て

subspmd

ローカル データベースに親 PMDB を割り当てます。

unsubs

PMDB からサブスクライバを削除します。

注: pmd 環境で使用できる selang コマンドの詳細については、「リファレンス ガイド」を参照してください。

アーキテクチャ依存関係

eTrust AC を展開するときは、環境の階層を考慮する必要があります。 多くのサイトで、ネットワークにはさまざまなアーキテクチャが採用されています。 Trusted プログラムのリストなど、一部のポリシー ルールはアーキテクチャに依存します。 一方、ほとんどのルールは、システムのアーキテクチャに関係なく適用されます。

階層を使用すると、両方の種類のルールを適用できます。 アーキテクチャに依存しないルールをグローバル データベースで定義し、そのグローバル データベースのサブスクライバ PMDB で、アーキテクチャに依存するルールを定義できます。

注: ルート PMBD とそのすべてのサブスクライバは、環境の物理的ニーズに応じて、同じコンピュータ上に存在することも、別々のコンピュータ上に存在することも可能です。

例： 2 層の展開階層

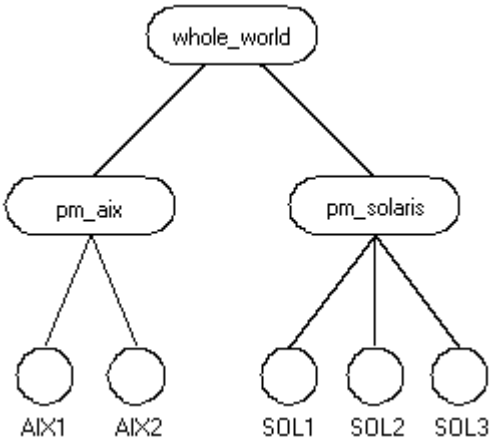
次の UNIX の例は、少し変更して Windows アーキテクチャにも適用できます。

この例では、サイトは IBM AIX システムと Sun Solaris システムで構成されています。 IBM AIX の trusted プログラムのリストは Sun Solaris でのリストとは異なるため、アーキテクチャの依存関係を考慮した PMDB が必要です。

複数アーキテクチャに対応した PMDB をセットアップするには、PMDB を以下のようにセットアップします。

1. whole_world という PMDB を定義し、ユーザ、グループ、およびアーキテクチャに依存しないその他のすべてのポリシーを格納します。
2. pm_aix という PMDB を定義し、IBM AIX 固有のすべてのルールを格納します。
3. pm_sol という PMDB を定義し、Sun Solaris 固有のすべてのルールを格納します。

   pm_aix および pm_solaris という PMDB は、whole_world という PMDB のサブスクライバです。 サイト内のすべての IBM AIX コンピュータは pm_aix のサブスクライバです。 サイト内のすべての Sun Solaris コンピュータは pm_sol のサブスクライバです。 この概念を次の図に示します。



4. ユーザの追加や SURROGATE ルールの設定など、プラットフォームに依存しないコマンドを whole_world に入力すると、サイト内のすべてのデータベースが自動的に更新されます。
5. Trusted プログラムを pm_aix に追加すると、IBM AIX コンピュータのみが更新されます。Sun Solaris システムには影響はありません。

ポリシーの一元管理の方法

eTrust AC を使用すると、以下の 2 通りの方法で 1 台のコンピュータから複数のデータベースを管理できます。

- 自動的なルール ベース ポリシー更新

  中央のデータベース(PMDB)で定義した通常のルールは、設定された階層内のデータベースに自動的に伝達されます。

  注: 二重チェックは、この方法でのみ使用できます。また、UNIX でのみ使用可能です。

- 拡張ポリシー管理およびレポート

  展開したポリシー(ルールの集まり)は、設定された階層内のすべてのデータベースに伝達されます。 ポリシーを展開解除(削除)したり、展開のステータス、展開の偏差、および展開の階層について報告することもできます。 この機能を使用するには、追加のコンポーネントをインストールおよび設定する必要があります。

注: 拡張ポリシー管理およびレポートには、自動的なルール ベース ポリシー更新では使用できない追加の機能が用意されています。 ただし、拡張ポリシー管理を使用するには、まず環境に自動的なルール ベース ポリシー更新を設定する必要があります。

自動的なルール ベース ポリシー更新

中央データベースで単一ルール ポリシー更新（標準の selang ルール）を行うと、サブスクライバ データベースに自動的に伝達されます。 複数のコンピュータを同じデータベースにサブスクライブし、データベースを別のデータベースにサブスクライブすることによって、階層を作成できます。 インストール後に、自動的なルール ベース ポリシー更新を環境に設定します。

注: このポリシー管理方法は、単一ルール ポリシー更新を階層全体に伝達することだけに制限されます。 その他の機能を使用するには、拡張ポリシー管理およびレポート (P. 95)を実装する必要があります。

自動的なルール ベース ポリシー更新のしくみ

環境に自動的なルール ベース ポリシー更新を設定すると、中央データベースで定義した各ルールは、以下の方法ですべてのサブスクライバに自動的に伝達されます。

1. 少なくとも 1 つのサブスクライバを持つ任意の PMDB にルールを定義します。
2. PMDB がすべてのサブスクライバ データベースにコマンドを送信します。
3. 伝達されたコマンドをサブスクライバ データベースが適用します。
   a. サブスクライバ データベースから応答がない場合、PMDB はサブスクライバ データベースが更新されるまで、定期的に（デフォルトでは 30 分間隔）コマンドを送信し続けます。
   b. サブスクライバ データベースから応答があっても、コマンドの適用が拒否された場合、PMDB はこのコマンドを Policy Model のエラー ログ (P. 92)に記録します。
4. サブスクライバ データベースが別のサブスクライバの親である場合は、サブスクライバ データベースはそのサブスクライバにコマンドを送信します。

例： 階層内のすべてのコンピュータからユーザを削除する

rmusr コマンドによってユーザが PMDB から削除されると、同じ rmusr コマンドがすべてのサブスクライバ データベースに送信されます。 このように、rmusr コマンドを 1 回実行すれば、さまざまな種類のコンピュータ上にある多数のデータベースからユーザを削除できます。

階層のセットアップ方法

eTrust AC は、Policy Model サービスを使用して、設定された階層全体にルール ベース ポリシー更新を伝達します。 複数の eTrust AC コンピュータを同じ PMDB にサブスクライブし、PMDB を別の PMDB にサブスクライブすることによって、階層を作成できます。

PMDB の階層構造は、eTrust AC のインストール時にセットアップするのが最も簡単です。したがって、インストール作業を始める前に、どのように階層を構成するか考えておくことをお勧めします。 親 PMDB とそのサブスクライバは互いに通信可能である必要があるため、PMDB 階層構造内のすべてのホストは同じネットワークに属している必要があります。 つまり、親 PMDB とそのサブスクライバは、名前を指定して互いに接続できることが必要です。

注: eTrust AC のインストールの詳細については、「実装ガイド」を参照してください。

インストール時に行った設定を変更するか、またはインストール時に PMDB 構造を作成しなかった場合は、いつでも PMDB の設定を変更または作成することができます。 これには、次のいずれかの方法で行います。

- ポリシー マネージャを使用する
- sepmd ユーティリティを使用する

インストール後に、PMDB 階層を作成し、自動的なルール ベース ポリシー更新を有効にするには、以下の手順に従います。

1. マスタ PMDB を作成し、設定します。
2. (オプション) サブスクライバ PMDB を作成し、設定します。
3. サブスクライバ コンピュータ (エンドポイント) に親 PMDB を定義します。

サブスクライバの更新

サブスクライバを更新すると、Policy Model は以下のアクションを実行します。

1. Policy Model からサブスクライバ名が追加または削除される場合、そのサブスクライバの名前を完全修飾しようとします。
2. PMDB サービスの sepmdd が、サブスクライバ データベースの更新を試みます。
3. 制限時間が経過した時点でサブスクライバを更新できなかった場合、サービスはそのサブスクライバの更新処理を省略して、サブスクライバ リストにある残りのサブスクライバの更新を試みます。
4. sepmdd は、サブスクライバ リストの 1 回目のスキャンが終了した後、2 回目のスキャンを実行します。2 回目のスキャンでは、1 回目のスキャンで更新できなかったサブスクライバの更新を試みます。

注: サブスクライバへの更新情報の伝達時に PMDB でエラーが発生すると、sepmdd サービスによって Policy Model のエラー ログ ファイル (P. 92)にエントリが作成されます。 このファイル(ERROR_LOG)は PMDB ディレクトリ (P. 80)に格納されます。

Policy Model データベースの更新

PMDB が格納されているコンピュータで操作を行っても、PMDB 自体は自動的に更新されません。 PMDB を更新するには、PMDB をターゲット データベースとして指定する必要があります。

PMDB は、selang またはポリシー マネージャを使用して指定することができます。 selang を使用してターゲット データベースを指定するには、selang のコマンド シェルで hosts コマンドを使用します。

```
eTrustAC> hosts <pmd_name>@<pmd_host>
```

これで、すべての selang コマンドで、指定した Policy Model データベースが更新されます。 次に、このコンピュータおよびすべてのサブスクライバ コンピュータ上のアクティブなデータベースにコマンドが自動的に伝達されます。

注: ポリシー マネージャの詳細については、ポリシー マネージャのオンライン ヘルプを参照してください。

例： ターゲット PMDB を指定する

ターゲット データベースを myPMD_host の policy1 に設定するには、以下のコマンドを使用します。

```
eTrustAC> hosts policy1@myPMD_host
```

ここで、newusr コマンドを入力すると、新規ユーザは policy1 データベースに追加される以外に、このコンピュータおよびすべてのサブスクライバ コンピュータ上のアクティブデータベースにも追加されます。

更新ファイルのクリーンアップ

sepmd ユーティリティは、受信した各更新情報を updates.dat ファイルに自動的に書き込みます。 このファイルのサイズが大きくなりすぎないように、処理済みの更新情報をファイルから定期的に削除することをお勧めします。

更新ファイルをクリーンアップするには、以下のコマンドを使用します。

```
<eTrustAC_InstallDir>/bin sepmd -t pmdbName auto
```

sepmd は、まだ伝達されていない最初の更新エントリのオフセットを計算して、その前にあるすべての更新エントリを削除します

パスワードの伝達と同期

PMDB の階層を設定すれば、Windows のユーザー マネージャまたは eTrust AC 以外のソフトウェアでユーザ パスワードが変更された場合にも、この階層を使用してシステム全体でユーザ パスワードの同期を維持できます。

注: eTrust AC では、メインフレームとのパスワード同期 (P. 153)もサポートされています。

パスワードを伝達および同期するには

1. PMDB 階層を作成します。
2. ユーザまたは管理者がパスワードを変更する可能性がある各端末で、レジストリの passwd_pmd エントリの値に適切な親 PMDB の名前を入力します。

   ```
   HKEY_LOCAL_MACHINE¥Software¥ComputerAssociates¥eTrustAccessControl¥ eTrustAccessControl¥passwd_pmd
   ```

   次に、PMDB からすべてのサブスクライバにパスワードの変更が伝達されます。passwd_pmd 値が空の場合には、eTrust AC は secondary_pmd 値をチェックし、secondary_pmd 値が空でない場合は、この値で指定された PMDB に新しいパスワードと更新されたパスワードを送信します。

注: ユーザが定義されていないサブスクライバに PMDB からユーザ パスワードが送信された場合、設定は変更されず、ユーザはそのサブスクライバに対して未定義のままになります。

サブスクライバの削除

更新情報が特定のサブスクライバに伝達されないようにする場合は、そのサブスクライバを削除する必要があります。

サブスクライバを削除するには

1. コンピュータをサブスクライバ リストから削除します。

   ```
   sepmd -u <PMDB_name> <computer_name>
   ```

   コンピュータが Policy Model のサブスクライバ リストから削除されます。

2. サブスクライバ リストから削除したコンピュータで seosd を停止します。

   ```
   secons -s
   ```

   seosd サービスが停止されます。

3. サブスクライバ リストから削除したコンピュータで以下のレジストリ キーの parent_pmd レジストリの値を削除します。

   ```
   HKEY_LOCAL_MACHINE¥Software¥ComputerAssociates¥eTrustAccessControl¥ eTrustAccessControl
   ```

   コンピュータは親 PMDB から更新情報を受け取らなくなります。

4. seosd を再起動します。

   サブスクライバ リストから削除したコンピュータ上のアクティブ データベースは、指定した PMDB のサブスクライバではなくなりました。

注: データベースが PMDB からサブスクライブ解除されると、PMDB はコマンドを送信しなくなります。

更新情報のフィルタ処理

1 つの PMDB を使用して、複数の異なるサブスクライバ データベースでデータのさまざまなサブセットを更新する場合は、サブスクライバ データベースにどのレコードを送信するかを定義する必要があります。

更新情報をフィルタ処理するには

1. サブスクライバのサブセットの親として PMDB を設定します。
2. 親 PMDB のレジストリ キーの Filter レジストリのエントリを変更し、同じコンピュータで設定するフィルタ ファイルを参照するようにします。

   このように指定すると、フィルタ条件に該当するレコードのみがサブスクライバ データベースに更新情報として送信されます。

Policy Model のフィルタ ファイル

フィルタ ファイルは、各行に 6 つのフィールドを持つ複数の行で構成されます。フィールドには次の情報が含まれます。

- 許可または拒否されるアクセスの種類。

  例：EDIT または MODIFY

- 影響を受ける環境

  eTrust、UNIX、またはネイティブ

- レコードのクラス。

  例：USER または TERMINAL

- ルールが適用される、クラスのオブジェクト。

  たとえば、User1、AuditGroup、または COM2 になります。

- レコードによって許可または取り消されるプロパティ。

  たとえば、フィルタ行の OWNER および FULL_NAME は、これらのプロパティを持つコマンドはすべてフィルタ処理されることを意味します。 各プロパティは、「リファレンス ガイド」に記載されているとおりに、正確に入力する必要があります。

- 該当するレコードをサブスクライバ データベースに転送するかどうか。

  PASS または NOPASS

フィルタ ファイルの各行に以下のルールが適用されます。

- どのフィールドでも、アスタリスク（*）を使用して可能なすべての値を指定することができます。
- 同じレコードが複数の行に該当する場合は、最初の該当する行が使用されます。
- フィールドをスペースで区切ります。
- フィールドに複数の値がある場合は、値をセミコロンで区切ります。
- # で始まる行はコメント行とみなされます。
- 空白行は使用できません。

例： フィルタ ファイル

次の例では、フィルタ ファイルの行について説明します。

CREATE	eTrust	USER	*	FULL_NAME;OBJ_TYPE	NOPASS
↑	↑	↑	↑	↑	↑
アクセスの種類	環境	クラス	レコード名 (* = すべての名前)	プロパティ	処理方法

この例では、この行を指定したファイルの名前が Printer1_Filter.flt で、PMDB PM-1 のレジストリを編集してフィルタを C:¥Program Files¥CA¥eTrustAccessControl¥Printer1_Filter.flt と指定した場合、PMDB PM-1 は、FULL_NAME と OBJ_TYPE プロパティを指定してユーザを新規作成するレコードをサブスクライバに伝達しません。

Policy Model のエラー ログ ファイル

Policy Model のエラー ログ(発生順に書き込まれる)の例を次に示します。

エラー テキスト	エラー カテゴリ
20 Nov 03 11:56:07 (pmdb1): fargo nu u5 0 Retry エラー: ログイン処理に失敗しました (10068) エラー: 親以外の PMDB からの更新は、受け取ることができません (pmdb1@name.company.com) (10104)	環境設定エラー
20 Nov 03 19:53:17 (pmdb1): fargo nu u5 0 Retry エラー: 接続に失敗しました (10071) ホストに到達不能です(12296)	接続エラー
20 Nov 03 11:57:06 (pmdb1): fargo nu u5 560 Cont エラー: USER u5 の作成に失敗しました (10028) すでに存在します (-9) 20 Nov 03 11:57:06 (pmdb1): fargo nu u5 1120 Cont エラー: USER u5 の作成に失敗しました (10028) すでに存在します (-9)	データベース更新エラー

Policy Model のエラー ログはバイナリ フォーマットであるため、以下のコマンドを入力することでのみ表示できます。

```
<eTrustAC_InstallDir>/bin sepmd -e pmdname
```

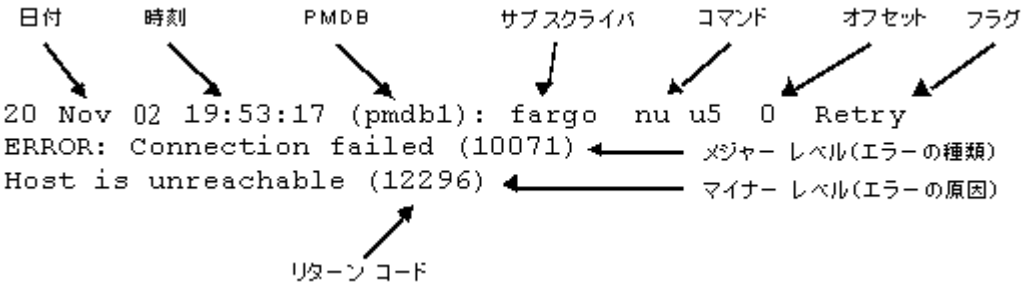
注: エラー ログは手動で(たとえば、UNIX の rm コマンドを使用して)削除しないでください。 ログを削除するには、次のコマンドのみを使用します。

```
<eTrustAC_InstallDir>/bin sepmd -c pmdname
```

重要: eTrust AC r5.1 以降のバージョンでのエラー ログのフォーマットには、旧バージョンのフォーマットとの互換性はありません。sepmd を使用して、旧バージョンのエラー ログを処理することはできません。 このバージョンのフォーマットにアップグレードする際に、旧エラー ログは ERROR_LOG.bak としてコピーされ、sepmdd を起動すると新しいログ ファイルが作成されます。

例： PMDB 更新のエラー メッセージ

次の例は、標準的なエラー メッセージを示しています。



- 先頭行には必ず、日付、時刻、およびサブスクライバが表示されます。 次に、エラーを発生させたコマンドが表示され、その後に、更新ファイル内の失敗した更新の位置を示すオフセット（10 進数）が続きます。 最後のフラグは、PMDB が更新を自動的に再試行するか、または再試行せずに継続するかを示します。
- 2 行目は、メジャー レベル メッセージ（発生したエラーの種類）とリターン コードの例を示します。
- 3 行目は、マイナー レベル メッセージ（エラーの発生理由）とリターン コードの例を示します。

例： エラー メッセージ

1 つのコマンドによって、複数のエラーが生成および表示される場合があります。 また、エラーは、メジャー レベル メッセージ、マイナー レベル メッセージ、またはその両方で構成される場合があります。

以下のエラーには、メッセージ レベルが 1 つしかありません。

```
Fri Dec 29 10:30:43 2003 CIMV_PROD:リリースに失敗しました。 リターン コード = 9241
```

このメッセージは、すでに使用可能なサブスクライバのリリースを sepmd pull が試みた場合に表示されます。

Policy Model のネイティブ リポジトリ

PMDB には、ネイティブ環境のすべての種類のユーザおよびグループ オブジェクトを保存できます。 このような情報を PMDB に保存すると、show コマンド(show user または show group)を使用して、オブジェクトに関する情報を取得できます。 返されるオブジェクトは、Windows サブスクライバまたは UNIX サブスクライバで定義されている実際のオブジェクトのイメージです。

Policy Model への接続後に、ユーザは次の環境のいずれかを選択できます。

- eTrust
- Native
- NT
- UNIX

注: Native を選択すると、Windows オペレーティング システムで作業している場合は Windows と同様に、UNIX オペレーティング システムで作業している場合は UNIX と同様に機能します。

Native 環境のリポジトリを使用するには、以下のコマンドを使用します。

- selang のプロンプトで次のコマンドを入力します。

  ```
  env NT; find
  ```

  このコマンドを実行すると、Native 環境のすべてのオブジェクトの種類が表示されます

  注: これらのオブジェクトの種類の詳細については、「リファレンス ガイド」の Windows 環境のクラスとプロパティに関する説明を参照してください。

- NT および Active Directory の USER プロパティの一覧を取得するには、次のコマンドを入力します。

  ```
  env NT; ruler user
  ```

- NT および Active Directory の GROUP プロパティの一覧を取得するには、次のコマンドを入力します。

  ```
  env NT; ruler group
  ```

Policy Model が別の(親)Policy Model のサブスクライバである場合、この Policy Model は伝達により親からのデータを受け取り、このデータを参照および変更できるように、すべてのユーザ プロパティとグループ プロパティをデータベースに保存します。

注: 詳細については、「リファレンス ガイド」の sepmd ユーティリティに関する説明を参照してください。

拡張ポリシー管理およびレポート

作成した複数ルールのポリシー（スクリプト ファイル）を格納し、設定された階層に展開できます。 このポリシー ベースの方法を使用して、ポリシーのバージョンを格納し、それらを展開および展開解除できます。 展開のステータス、展開の偏差、および展開の階層に関するレポートを作成することもできます。

注: 二重チェックは、この方法では使用できません。また、UNIX でのみ使用可能です。

環境アーキテクチャ

拡張ポリシー管理およびレポートを使用するには、以下の追加コンポーネントをインストールして設定する必要があります。

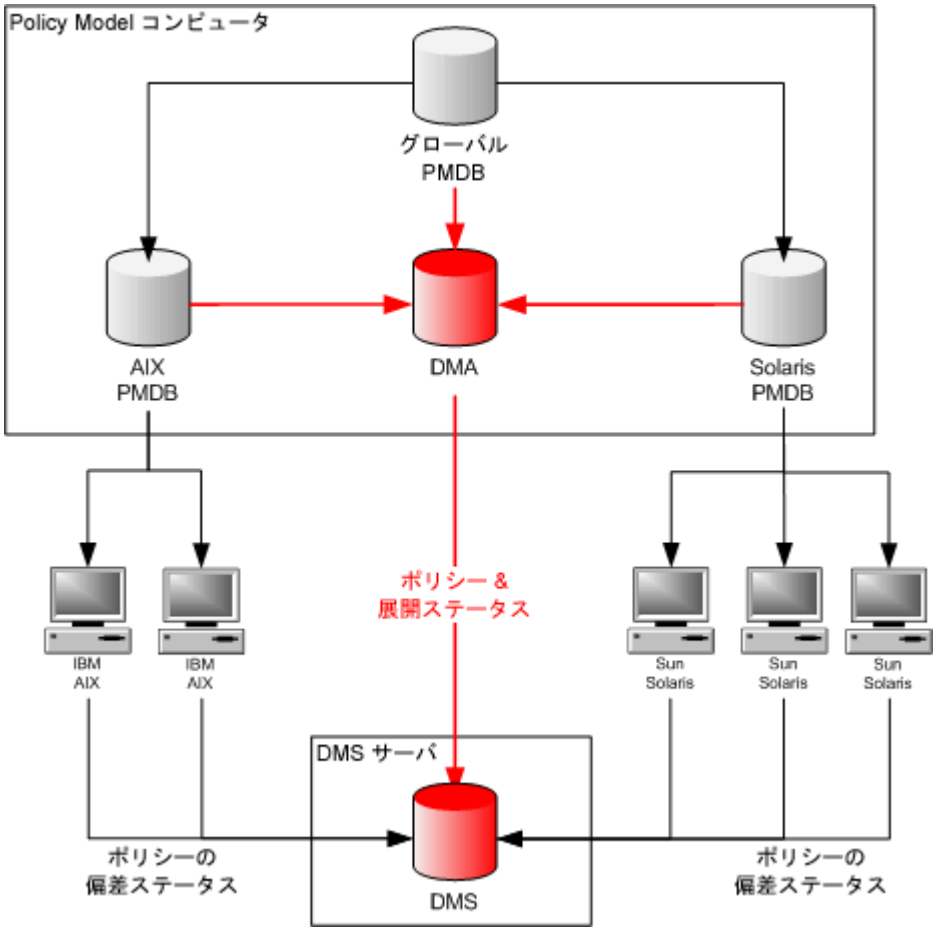
- この用途専用の中央コンピュータに、展開マップサーバ（DMS） (P. 97)。
- 少なくとも 1 つの PMDB を含む各コンピュータに、展開マップ エージェント（DMA） (P. 97)。

注: 拡張ポリシー管理およびレポートを使用するには、環境に自動的なルール ベース ポリシー更新を設定する必要があります。 DMS と DMA を適切なコンピュータにインストールした後、親データベースとサブスクライバ データベースを設定します (P. 85)。

例： 中央 DMS を持つ 2 層階層

注: 次の UNIX の例は、少し変更して Windows アーキテクチャにも適用できます。

この例では、サイトは IBM AIX システムと Sun Solaris システムで構成されています。IBM AIXのtrustedプログラムのリストはSun Solarisでのリストとは異なるため、アーキテクチャの依存関係を考慮したPMDBが必要です。 管理およびレポートを使用するために、DMS と DMA をセットアップして、複数 PMDB の環境 (P. 82)の設定時に作成した環境をサポートします。 DMS には、ポリシーの展開および偏差のすべての情報が格納され、eTrust AC でこの情報からレポートを作成できます。



展開マップ サーバ(DMS)

DMS は、拡張ポリシー管理およびレポートの中核となります。 DMS は、eTrust AC 展開階層と各コンピュータに展開されたポリシーのステータスの最新マップを保持することを目的としています。 中央の 1 箇所にこのデータを置くことによって、(階層内の各データベースに接続するよりも)レポートの生成に必要な時間が短縮されます。 また、DMS には、ポリシーのバージョンが格納され、後から必要に応じてこれらのバージョンを展開および展開解除できます。

DMS は PMD ノードであり、データ リポジトリとして PMDB を使用します。 設定された各 PMD ノードからの通知で受信したデータを収集します。

展開マップ エージェント (DMA)

DMA は、DMS との PMD 通信を担当するエージェントです。 各 PMD ノード(少なくとも 1 つの PMDB)に DMA をインストールする必要があります。 親 PMDB 上の DMA の役割は、ポリシー展開のステータスと階層の変更を DMS に通知することです。 エンドポイントはポリシー偏差ステータスだけを DMS に直接送信します。

注: エンドポイント コンピュータに DMA をインストールしないでください。

DMA は、DMS との通信に eTrust AC の標準の通信メカニズムと暗号化メカニズムを使用します。

注: DMA¥DMS 通信を有効にするために DMA を DMS の親として定義する必要はありません。

拡張ポリシー管理クラス

DMS は、eTrust AC 展開階層と各コンピュータに展開されたポリシーのステータスの最新マップを保持するために、特定の eTrust AC クラスを使用します。

HNODE

各 HNODE オブジェクトは階層内のノードを表します。 これらのオブジェクトは、それぞれが表す特定ノード、そのサブスクライバ、および親 PMDB に関する情報を保持します。 また、各 HNODE オブジェクトは、そのオブジェクトが表すノードに展開されている必要があるポリシーおよび各ポリシーのステータス(展開されている、展開されているがエラーがある、など)に関する情報を保持します。

HNODE オブジェクトの名前は、そのオブジェクトが表すノードのタイプによって異なります。

- エンドポイントの実際のホスト名。

  例: myhost.mydomain.com

- PMDB の Policy Model 名。

  例: mypmd@hostB.domain.com

POLICY

各 POLICY オブジェクトは、HNODE 階層の任意の部分に展開できるポリシーのバージョンを表します。 このオブジェクトには、関連付けられたポリシー スクリプトが格納されている場所(どの RULESET オブジェクトか)およびそれが展開される必要があるノードに関する情報が含まれます。

オブジェクトの名前は、ポリシーの名前にバージョン番号のサフィックスが付いたものです(policy_name#xx)。

RULESET

各 RULESET オブジェクトは、ポリシー バージョンに関連付けられた展開および展開解除(削除)スクリプトを保持します。

オブジェクトの名前は、対応する POLICY オブジェクト名に基づきます。

注: これらのクラスの詳細については、「リファレンス ガイド」を参照してください。

拡張ポリシー ベース管理およびレポートのための階層のセットアップ方法

eTrust AC は、DMS を使用して、eTrust AC 展開階層と各コンピュータに展開されたポリシーのステータスの最新マップを保持します。 階層内の各コンピュータに適切なコンポーネントをインストールおよび設定して、ポリシー ベース管理およびレポートを有効にします。

ポリシー ベース管理およびレポートを有効にするには、以下の手順に従います。

1. 中央のコンピュータに DMS をインストールします。

   DMS は、eTrust AC のインストール時、または dmsmgr ユーティリティでインストールできます。

2. 各 PMD ノードに DMA をインストールします。

   DMA は、eTrust AC のインストール時、または dmsmgr ユーティリティでインストールできます。

3. 各 eTrust AC コンピュータに拡張ポリシー管理およびレポート機能をインストールします。

   これで、偏差計算でポリシー偏差ステータスを DMS に送信するように設定されます。

4. 階層をセットアップします (P. 85)。

   階層をセットアップすると、階層内の各ノードを現す HNODE オブジェクトが DMS に追加されます。

   重要: コンピュータから eTrust AC をアンインストールするか、PMDB を削除しても、HNODE オブジェクトはそのまま残ります。 古いオブジェクト ノードを削除する (P. 101)必要があります。 階層からデータベースをサブスクライブ解除すると、HNODE オブジェクトはそのまま残りますが、親ノードへのリンクは削除されます。このオブジェクトを削除する必要はありませんが、以前にそのノードに展開されたポリシー オブジェクトへのリンクは維持されます。

注: DMS、DMA のインストール方法と、拡張ポリシー管理およびレポート機能の設定方法については、「実装ガイド」を参照してください。

DMS 通知

環境に拡張ポリシー管理およびレポートを設定すると、階層内のコンポーネントが以下の 3 種類のステータス変更を DMS に通知します。

- 階層の変更。

  サブスクライバ(PMD ノードまたはエンドポイント)が追加または削除されると、通知が送信されます。

- ポリシーの展開および展開解除。

  サブスクライバ(PMD ノードまたはエンドポイント)にポリシーが展開または展開解除されると、通知が送信されます。 次に、ポリシーの詳細と展開のステータスが、操作の結果(成功、失敗など)に応じて更新されます。

- 偏差ステータス。

  eTrust AC エンドポイントがポリシーの偏差を計算し、結果(偏差が検出されたかされなかったか)を送信すると、通知が送信されます。

注: 階層の変更とポリシーの展開および展開解除の通知は、PMD ノードの DMA によってのみ送信されます。 偏差ステータスの通知は、eTrust AC エンドポイントの偏差計算機能によってのみ送信されます。

階層の通知とポリシー ステータスの通知のしくみ

DMA は DMS に階層の変更とポリシー ステータスの通知を送信します。 DMA 通知は以下の方法で処理されます。

1. DMA が通知メッセージを更新ファイルに格納します。

   これらは、階層の変更とポリシーの展開および展開解除の通知です。

2. DMA が DMS にアクセスします。

   - DMS が使用可能でない場合、すべてのメッセージが正常に送信されるまで DMA は定期的に DMS との通信を試行します。
   - DMS が使用可能な場合、DMA は格納した通知を送信します。

   注: 各 DMA が階層を無視して DMS と直接通信し、依存性が低くなります。

3. DMS が各 DMA から受け取った情報を、後で使用するために格納します。

   レポートを作成するたびに、eTrust AC は DMS にある情報を取得します。

偏差通知のしくみ

偏差計算機能は、eTrust AC と共にインストールされ、偏差を計算するエンドポイント上でローカルに実行されます。 偏差計算機能は以下のアクションを実行し、これらの通知を DMS に送信します。

1. selang コマンド(start devcalc)をエンドポイントに送信すると、計算プロセスが開始されます。

   この操作は、スクリプトをカスタマイズし、実行をスケジュールして行うことをお勧めします。

2. 計算が完了すると、偏差計算機能はその結果をデータ ファイルに格納します。

   ファイルは <eTrustAC_Dir>¥data¥devcalc¥deviation.dat です。

   注: レポート ユーティリティを使用して偏差の詳細を取得できます(-dev オプションを使用)。 または、エンドポイントで get devcalc コマンドを使用してデータ ファイルの内容を取得できます。

3. 次に、偏差計算機能は偏差ステータス(偏差が検出されたかされなかったか)を DMS に送信します。

   ステータス通知では、偏差自体(データ ファイルの内容)は送信されません。

階層からの古いノードの削除

DMS は階層に関する情報を格納します。 コンピュータから eTrust AC をアンインストールしたときに階層からそのコンピュータを削除した場合でも、DMS はそのノードへの参照を保持します。 定期的な保守手順として、これらの古いノードから DMS を消去する必要があります。

階層から古いノードを削除するには、DMS コンピュータ上で dmsmgr ユーティリティを実行して定期的なクリーンアップを行います。

```
dmsmgr -dms -cleanup <number_of_days>
```

ここで、<number_of_days> は、eTrust AC ノードが使用可能でなくなってからの期間の最小日数です。

注: DMS コンピュータ上で以下の selang コマンドを発行して、特定のノードを手動で削除することもできます。

```
rr HNODE <HNODE_name>
```

拡張ポリシー ベース管理のしくみ

拡張ポリシー ベース管理では、ポリシー バージョンを格納、展開、および展開解除することができ、後から展開のステータス、展開の偏差、および展開の階層に関するレポートを作成することができます。 各ポリシーは、作成した 1 組の selang スクリプト ファイルです。 1 つ目のスクリプト ファイルは、「展開スクリプト」といい、ポリシーを作成する selang コマンドのセットが含まれます。 2 つ目のスクリプト ファイルは、「展開解除スクリプト」といい、エンドポイント データベースからポリシーを展開解除（削除）するために必要なコマンドが含まれます。

各ポリシーは、2 つの段階で、指定したターゲット データベースに適用されます。

1. DMS にポリシーの詳細を格納します。

   ポリシーの詳細には、展開および展開解除スクリプト、および自動的に作成されたポリシーのシグネチャ（同じポリシーのバリエーションを検出するために使用される）が含まれます。

   DMS にポリシーの詳細を格納できない場合、以下のことを確認してください。

   - DMS に対する TERMINAL 権限を持つコンピュータからポリシーを格納する。
   - DMS の POLICY および RULESET クラスに対するサブ管理権限を持っている。
   - 構文エラーを含む展開または展開解除スクリプトがない。

2. ユーティリティが、自動バージョン制御によりポリシーを格納します。

   ポリシーが DMS にすでに存在するかどうかに応じて、ユーティリティは以下のいずれかを実行します。

   - ポリシー名が DMS に存在しない場合、そのポリシーの最初のバージョンを作成する（policy_name#01）。
   - ポリシー名が DMS にすでに存在する場合、検出された最新のポリシー バージョンに 1 を加えた新しいポリシー バージョンを作成する。

3. 格納されたポリシー バージョンをターゲット データベースに展開します。

   格納されたポリシーをターゲット階層に展開できない場合、以下のことを確認してください。

   - ターゲットの Policy Model ルートに対する TERMINAL 権限を持つコンピュータから展開する。
   - DMS と、ポリシーを展開する階層内の各データベースの、POLICY、RULESET、および HNODE クラスに対するサブ管理権限を持っている。
   - ターゲットの Policy Model ルート コンピュータのサブ管理権限を持っている。

- 展開階層に含まれるホストにすでに展開されたポリシーと同じポリシーのバージョンを持っていない。

4. 各ルール(展開スクリプトで指定された selang コマンド)がターゲット データベースで実行されます。

   特定のデータベースにルールを展開できない場合、そのポリシーは、展開されているがエラーがあると見なされます(ステータス Failed)。

   これは、展開スクリプトに以下が含まれる場合にホストにポリシーを展開しようとしたときに発生します。

   - 存在しないオブジェクトへの参照。 次に例を示します。

     ```
     cr FILE /does_not_exist comment(123)
     ```

     したがって、ポリシー展開スクリプトは自己完結可能である必要があります。 つまり、展開スクリプトは、使用するすべてのリソースを作成するように構築する必要があります。

   - エラーが発生するコマンド。
   - 実行するためのサブ管理権限がないコマンド。

5. ポリシー ステータスが記録されます。

   ポリシー ステータスには、Deployed、Undeployed、Transferred、Failed(展開されたがエラーがある)、Queued、TransferFailed、SigFailed(シグネチャが失敗した)、UndeployFailed(展開解除されたがエラーがある)、Unknown があります。

   注: ポリシーが展開されたがエラーがある場合、ポリシーが展開されてエラーが発生したコンピュータのログ ファイルを参照する必要があります。

ポリシーの詳細が DMS に格納され、ターゲット データベースに展開されると、ターゲット データベースが PMDB の場合は、自動的なルール ベース ポリシー更新メカニズムによりポリシーが階層全体に伝達されます。

階層に新しいサブスクライバが追加されると、すべてのポリシーが階層全体に伝達され、階層にノードが追加されたことが DMS に通知されます。

管理要件

ポリシー展開ユーティリティ(policydeploy)は、eTrust AC がインストールされていれば、どのコンピュータからでも実行できます。 DMS にポリシーを格納する、または階層内のデータベースにポリシーを展開および展開解除するには、ユーザとその使用コンピュータに適切な権限が必要です。

DMS にポリシーを格納するには:

- policydeploy ユーティリティを実行する「コンピュータ」に DMS に対する端末権限（TERMINAL クラス）が必要です。
- 「ユーザ」に DMS の POLICY および RULESET クラスに対するサブ管理権限が必要です。

階層全体にポリシーを展開および展開解除するには:

- policydeploy ユーティリティを実行する「コンピュータ」にターゲットの Policy Model ルートであるコンピュータに対する端末権限（TERMINAL クラス）が必要です。
- 「ユーザ」に以下が必要です。
  - DMS の POLICY および RULESET クラスに対する読み取り権限、および HNODE クラスに対するサブ管理権限。
  - ポリシーを展開する階層内の各データベースの POLICY、HNODE、および RULESET クラスに対するサブ管理権限。
  - ポリシーを展開する階層内の各データベースの適切なサブ管理権限。

    これらは、これらの各コンピュータのポリシーを構成する selang コマンドを展開するために必要な権限です。

    たとえば、新しいファイル リソースを作成する場合、FILE クラスに対するサブ管理権限が必要になります。

    ```
    nr FILE /inetpub/* defaccess(none)
    ```

承認されたポリシー バージョンの展開方法

拡張ポリシー ベース管理を使用して、ポリシーのドラフト バージョンを格納し、それを確認して必要に応じて変更してから、承認バージョンを展開することができます。

承認されたポリシー バージョンを展開するには、以下の手順に従います。

1. DMS にポリシー バージョンを格納します。

   ポリシー バージョンを格納したら、ポリシーを確認および展開できます。

2. ポリシーを確認します (P. 108)。

   POLICY および RULESET オブジェクトに対する読み取り権限を持つすべてのユーザが、ポリシーとそれに関連付けられたルールを参照できます。

3. 必要に応じて、承認された変更を含む新しいバージョンのポリシーを格納できます。

   ポリシーを更新する必要がある場合は必ず、変更された必要なポリシー展開および展開解除ルールを含む新しいバージョンのポリシーを格納する必要があります。

4. 承認されたポリシー バージョンを階層に展開します (P. 109)。

ポリシー バージョンの格納

DMS にポリシーを格納するたびに、自動的にバージョン番号が付けられます。 ポリシーを最初に格納したときは、バージョン番号「01」が付けられます。 たとえば、ポリシー myPolicy を初めて格納すると、policydeploy ユーティリティによって myPolicy#01 という POLICY オブジェクトが作成されます。 DMS にすでに存在するポリシーを格納するたびに、格納されているポリシーの最新バージョンに 1 を加えて新しいポリシー バージョンが作成されます。 たとえば、myPolicy のバージョンを 28 回目に格納したときは、policydeploy ユーティリティによって myPolicy#28 という POLICY オブジェクトが作成されます。

これで、格納したポリシーを参照し、必要に応じて階層に展開できます。

ポリシー バージョンを格納するには

1. selang 展開コマンドを含む新しいスクリプト ファイルを作成します。

   これらは、階層内の各コンピュータに展開するポリシーを作成するために必要なコマンドです。

   重要: ポリシーの展開では、ユーザ パスワードを設定するコマンドはサポートされていません。 そのようなコマンドを展開スクリプト ファイルに含めないでください。 Windows（ネイティブ）selang コマンドはサポートされていますが、偏差レポートには示されません。

2. selang 展開解除コマンドを含む新しいスクリプト ファイルを作成します。

   これらは、階層内のコンピュータからポリシーを展開解除（削除）するために必要なコマンドです。

   注: これらのコマンドは、ポリシー展開解除の実行時に新しいポリシー展開解除スクリプトを指定しない限り、ターゲットの階層からポリシーを展開解除するときにデフォルトで使用されます。

3. -store オプションを指定して policydeploy ユーティリティを実行します。

   ```
   policydeploy -store name -ds file1 -uds file2 [-dms list]
   ```

   ここで、name は格納するポリシーの名前、file1 は展開スクリプト ファイルの完全パスと名前、file2 は展開解除スクリプト ファイルの完全パスと名前、list はオプションの DMS ノードのカンマ区切りリストです。

   policydeploy ユーティリティによって、DMS にポリシーの新しいバージョンを作成するかどうかを確認するメッセージが表示されます。

   注: ポリシー名（names）には #（シャープ）文字を使用できません。この文字は、ポリシーのバージョン番号を示すために予約されており、自動的に追加されます。

4. 「y」と入力してこのアクションを確定します。

   policydeploy ユーティリティによって、DMS にポリシーの新しいバージョンが作成されます。

例： IIS 5 保護ポリシーを格納する

次の例は、インターネット インフォメーション サービス(IIS)5 Web サーバを保護するためのポリシーの格納方法を示します。 今回初めて DMS にこのポリシーを格納するとします。

ポリシー IIS5 を、iis5@host.company.com がルート PMDB である Policy Model 階層に展開します。

1. 以下の IIS スクリプトを含む IIS5.selang というファイルを保存します。

   ```
   nu inet_pers owner(nobody)
   nr FILE c:\InetPub\wwwroot\* defaccess(none) owner(nobody)
   authorize FILE c:\InetPub\wwwroot\* uid(inet_pers) access(all)
   nr FILE c:\InetPub\wwwroot\scripts defaccess(none) owner(nobody)
   nr FILE *.asp defaccess(none) owner(nobody)
   authorize FILE *.asp uid(inet_pers) via(pgm(inetinfo.exe)) access(read, execute)
   ```

   これらは、IIS 5 保護ポリシーを展開するために必要なコマンドです。

2. 以下のスクリプトを含む IIS5_rm.selang というファイルを保存します。

   ```
   ru inet_pers
   rr FILE c:\InetPub\wwwroot\*
   rr FILE c:\InetPub\wwwroot\scripts
   rr FILE *.asp
   ```

   これらは、手順 1 で作成した IIS 5 保護ポリシーを展開解除するために必要なコマンドです。

3. policydeploy ユーティリティを実行します。

   ```
   policydeploy -store IIS5 -ds IIS5.selang -uds IIS5_rm.selang
   ```

   これで、DMS の IIS5.selang と IIS5_rm.selang に定義されたように、IIS5 ポリシーの最初のバージョン(IIS5#01)が格納されます。

ポリシーに関連付けられたルールの表示

DMS にポリシーを格納したら、POLICY および RULESET オブジェクトに対する読み取り権限を持つすべてのユーザが、ポリシーとそれに関連付けられたルールを参照できます。 格納されているポリシーの最新バージョンが不明な場合、まずそれを確認できます。

ポリシーに関連付けられたルールを表示するには

1. selang を使用して DMS に接続します。

   ```
   hosts dms_name@hostname
   ```

   これで、selang を使用して DMS のクエリを実行できます。

2. ポリシーの最新バージョンを確認する場合、以下の selang コマンドを発行してポリシーのすべてのバージョンを検出します。

   ```
   find POLICY policy_name#*
   ```

   selang ウィンドウに、policy_name ポリシーのすべてのバージョンが一覧表示されます。

3. 以下の selang コマンドを発行して、ポリシー展開および展開解除スクリプトを表示します。

   ```
   sr RULESET policy_name#xx
   ```

   ここで、xx はルールを表示するポリシーの番号です。

   selang ウィンドウに、policy_name ポリシーの xx バージョンに関連する展開および展開解除ルールを含む policy_name#xx RULESET オブジェクトが表示されます。

格納されたポリシー バージョンの展開

後からポリシーを展開解除したり、展開のステータス、展開の偏差、および展開の階層に関するレポートを作成したりできるように、複数ルールのポリシーの格納されているバージョンを階層に展開できます。

格納されたポリシー バージョンを展開するには

以下のいずれかの操作を行います。

- 格納されているポリシーの「最新」バージョンを展開する場合は、ポリシー名とターゲット階層を指定して policydeploy ユーティリティを実行します。

  ```
  policydeploy -deploy name -root db1[,db2] [-dms list]
  ```

  DMS に格納されている指定した名前のポリシーの最新バージョンが検出され、ターゲット データベースへの展開が試行されます。 次に、ポリシー コマンドが、サブスクライブしているデータベース(存在する場合)に伝達されます。

- 格納されているポリシーの「特定の」バージョンを展開する場合は、ポリシー名、バージョン、およびターゲット階層を指定して policydeploy ユーティリティを実行します。

  ```
  policydeploy -deploy name#xx -root db1[,db2] [-dms list]
  ```

  指定したバージョンのポリシーのターゲット データベースへの展開が試行されます。次に、ポリシー コマンドが、サブスクライブしているデータベース(存在する場合)に伝達されます。

注: policydeploy ユーティリティの詳細については、UNIX の「ユーティリティ ガイド」または Windows の「リファレンス ガイド」を参照してください。

重要: 展開階層内のいずれかのホストにすでに展開されているポリシーのバージョンは展開できません。

例： IIS 5 保護ポリシーを展開する

次の例は、インターネット インフォメーション サービス(IIS)5 Web サーバを保護するためのポリシーの展開方法を示します。 ポリシー IIS5 の 4 つ目のバージョンを確認し、iis5@host1.company.com がルート PMDB である Policy Model 階層に展開します。 ポリシー IIS5 は crDMS@cr_host.company.com DMS ノードに格納されています。

1. selang を使用して DMS に接続します。

   ```
   hosts crDMS@cr_host.company.com
   ```

   これで、selang を使用して DMS のクエリを実行できます。

2. ポリシーの最新バージョンが不明な場合、以下の selang コマンドを発行してポリシーのすべてのバージョンを検出します。

   ```
   find POLICY IIS5#*
   ```

selang ウィンドウに、IIS5 ポリシーのすべてのバージョンが一覧表示されます。

3. 以下の selang コマンドを発行して、ポリシー展開および展開解除スクリプトを表示します。

   ```
   sr RULESET IIS5#04
   ```

   selang ウィンドウに、IIS5 ポリシーの 4 つ目のバージョンに関連する展開および展開解除ルールを含む IIS5#04 RULESET オブジェクトが表示されます。

4. コマンド ライン ウィンドウで、policydeploy ユーティリティを実行します。

   ```
   policydeploy -deploy IIS5#04 -root iis5@host1.company.com
   ```

   これで、IIS5 ポリシーの 4 つ目のバージョンが PMD 階層の iis5@host.company.com の下に展開されます。

ポリシーの展開解除

コンピュータに展開された複数ルールのポリシーが不要になった場合、ターゲットの階層からそのポリシーを展開解除できます。 ポリシーを変更する(ポリシーの更新バージョンを作成する)場合も、ポリシーを展開解除する必要があります。

ポリシーを展開解除するには

1. (オプション)selang 展開解除コマンドを含む新しいスクリプト ファイルを作成します。

   これらは、階層内のコンピュータからポリシーを展開解除(削除)するために必要なコマンドです。

   新しい展開解除スクリプトを作成および指定しない場合、展開解除コマンドでは、このポリシーが展開されたときにポリシーに割り当てられたスクリプトが使用されます。

   重要: ポリシー展開解除スクリプトを指定しても、DMS には、ポリシーを展開解除するために使用した新しいスクリプトではなく、ポリシーを格納したときに指定されたルールが引き続き記録されます。

2. 以下のいずれかの操作を行います。

   - ポリシーの「最新」バージョンを展開解除する場合は、ポリシー名とターゲット階層を指定して policydeploy ユーティリティを実行します。

     ```
     policydeploy -undeploy name -root db1[,db2] [-dms list] [-uds file2]
     ```

     DMS に格納されている指定した名前のポリシーの最新バージョンが検出され、ターゲット データベースからの展開解除が試行されます。 次に、ポリシー展開解除コマンドが、サブスクライブしているデータベース(存在する場合)に伝達されます。

     重要: 階層内のいずれかのエンドポイントのポリシー バージョンに、DMS で検出された最新バージョン以外のバージョンが含まれる場合、それらの特定のバージョンをそれぞれ明示的に展開解除する必要があります。

   - ポリシーの「特定の」バージョンを展開解除する場合は、ポリシー名、バージョン、およびターゲット階層を指定して policydeploy ユーティリティを実行します。

     ```
     policydeploy -undeploy name#xx -root db1[,db2] [-dms list] [-uds file2]
     ```

     指定したバージョン(xx)のポリシーのターゲット データベースからの展開解除が試行されます。 次に、ポリシー展開解除コマンドが、サブスクライブしているデータベース(存在する場合)に伝達されます。

   注: policydeploy ユーティリティの詳細については、UNIX の「ユーティリティ ガイド」または Windows の「リファレンス ガイド」を参照してください。

注: ポリシーを展開解除すると、DMS はポリシーのステータスが Undeployed であると報告します。 POLICY および RULESET オブジェクトは、後から再展開またはクエリを実行できるように、ポリシー バージョンが展開されたすべてのホスト(DMS を含む)に残ります。

展開されたポリシーの変更

展開されたポリシーを変更するには、まず展開されたポリシー バージョンを展開解除し、変更した展開および展開解除スクリプトを含む新しいバージョンのポリシーを格納してから、ポリシーを新しいバージョンで再展開する必要があります。

展開されたポリシーを変更するには

1. 新しいポリシー バージョンを格納します。

   ポリシーの新しいバージョンが DMS に格納されます。

2. ポリシーを展開解除します (P. 111)。

   ポリシーがターゲット階層から展開解除されます。

3. ポリシーの新しいバージョンを展開します (P. 109)。

   変更されたポリシーがターゲット階層に再展開されます。

拡張ポリシー レポートのしくみ

拡張ポリシー レポートでは、設定された階層や、拡張ポリシー ベース管理の方法で作成されたポリシーの、展開のステータス、展開の偏差、および展開の階層に関するレポートを作成することができます。 レポート生成ユーティリティ(policyreport)は、DMS の内容に基づく特定の時点の(静的)HTML レポートを生成します。

policyreport ユーティリティは、以下のアクションを実行して階層レポートおよびポリシーレポートを作成します。

1. DMS に、要求された情報のクエリを実行します。

   取得される情報は、生成するレポートのタイプによって異なります。

2. 偏差計算が要求されている場合、エンドポイントにポリシー偏差結果のクエリを実行します。

   偏差ステータスは DMS にありますが、実際の偏差は各エンドポイントから取得する必要があります。

3. XML ドキュメントのセットを生成します。

   これは、XML レポートです。

4. XML レポートを HTML 形式に編成します。

   これで、レポートをブラウザで表示できるようになります。

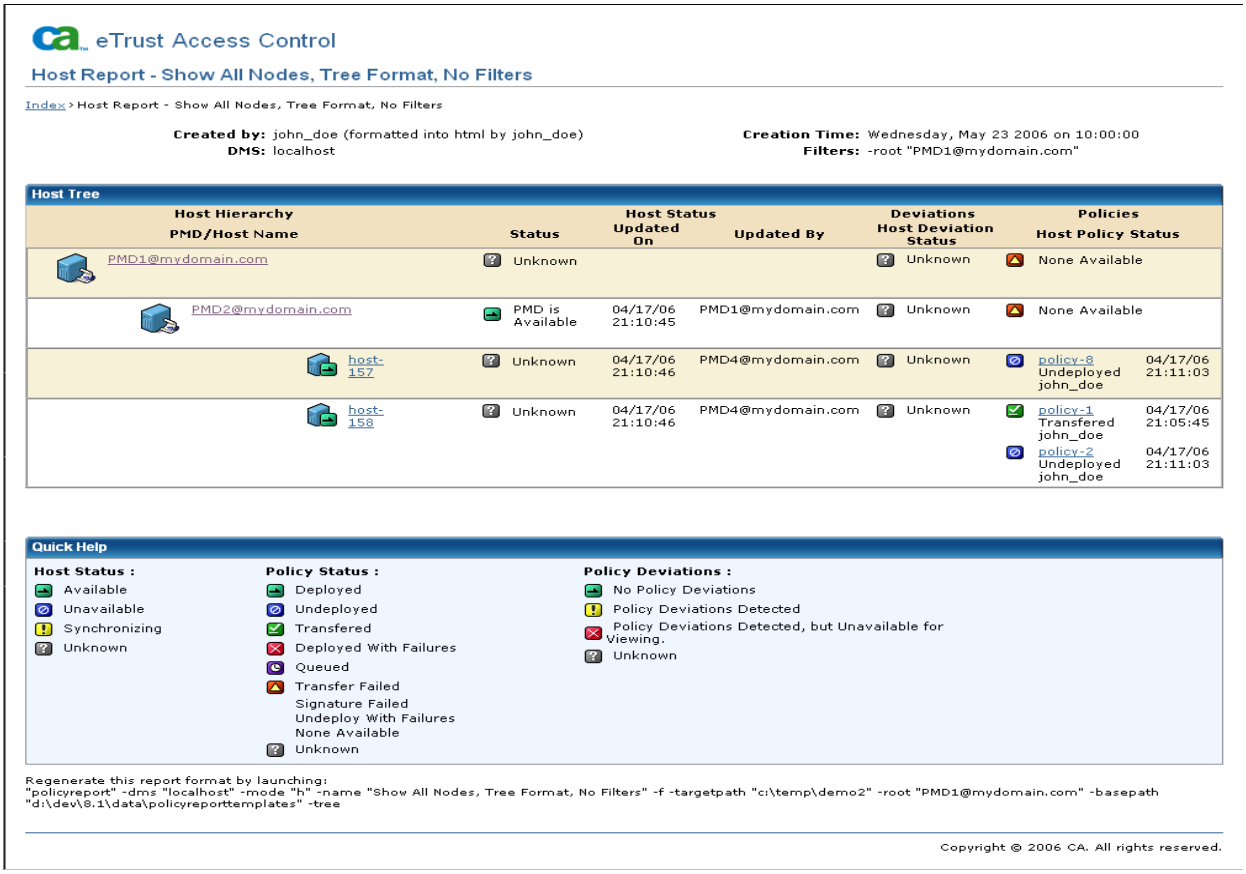
注: policyreport ユーティリティは、-targetpath オプションで指定したディレクトリの下の、-name オプションで指定したサブディレクトリに、レポートを格納します。

レポート タイプ

レポート生成ユーティリティでは、階層全体に展開されたポリシーを以下の 2 つのモードで表示できます。

- ホスト別

  ホスト レポートは、コンピュータ中心の情報を提供します。 このモードは、ホストごとの環境を確認する場合に使用します。 このモードで確認できるのは、各コンピュータがどのように設定されているか、階層内の各コンピュータのステータス、どのコンピュータにどのポリシーがあるか、そのステータス、各コンピュータに展開されている実際のルールと展開されるルールとの偏差です。



eTrust Access Control

Host Report - Show All Nodes, Tree Format, No Filters

Index > Host Report - Show All Nodes, Tree Format, No Filters

Created by: john_doe (formatted into html by john_doe)
DMS: localhost
Creation Time: Wednesday, May 23 2006 on 10:00:00
Filters: -root "PMD1@mydomain.com"

Host Tree

Host Hierarchy PMD/Host Name	Status	Host Status Updated On	Updated By	Deviations Host Deviation Status	Policies Host Policy Status
PMD1@mydomain.com	Unknown			Unknown	None Available
PMD2@mydomain.com	PMD is Available	04/17/06 21:10:45	PMD1@mydomain.com	Unknown	None Available
host-157	Unknown	04/17/06 21:10:46	PMD4@mydomain.com	Unknown	policy-8 Undeployed john_doe 04/17/06 21:11:03
host-158	Unknown	04/17/06 21:10:46	PMD4@mydomain.com	Unknown	policy-1 Transfered john_doe 04/17/06 21:05:45 policy-2 Undeployed john_doe 04/17/06 21:11:03

Quick Help

Host Status :
- Available
- Unavailable
- Synchronizing
- Unknown

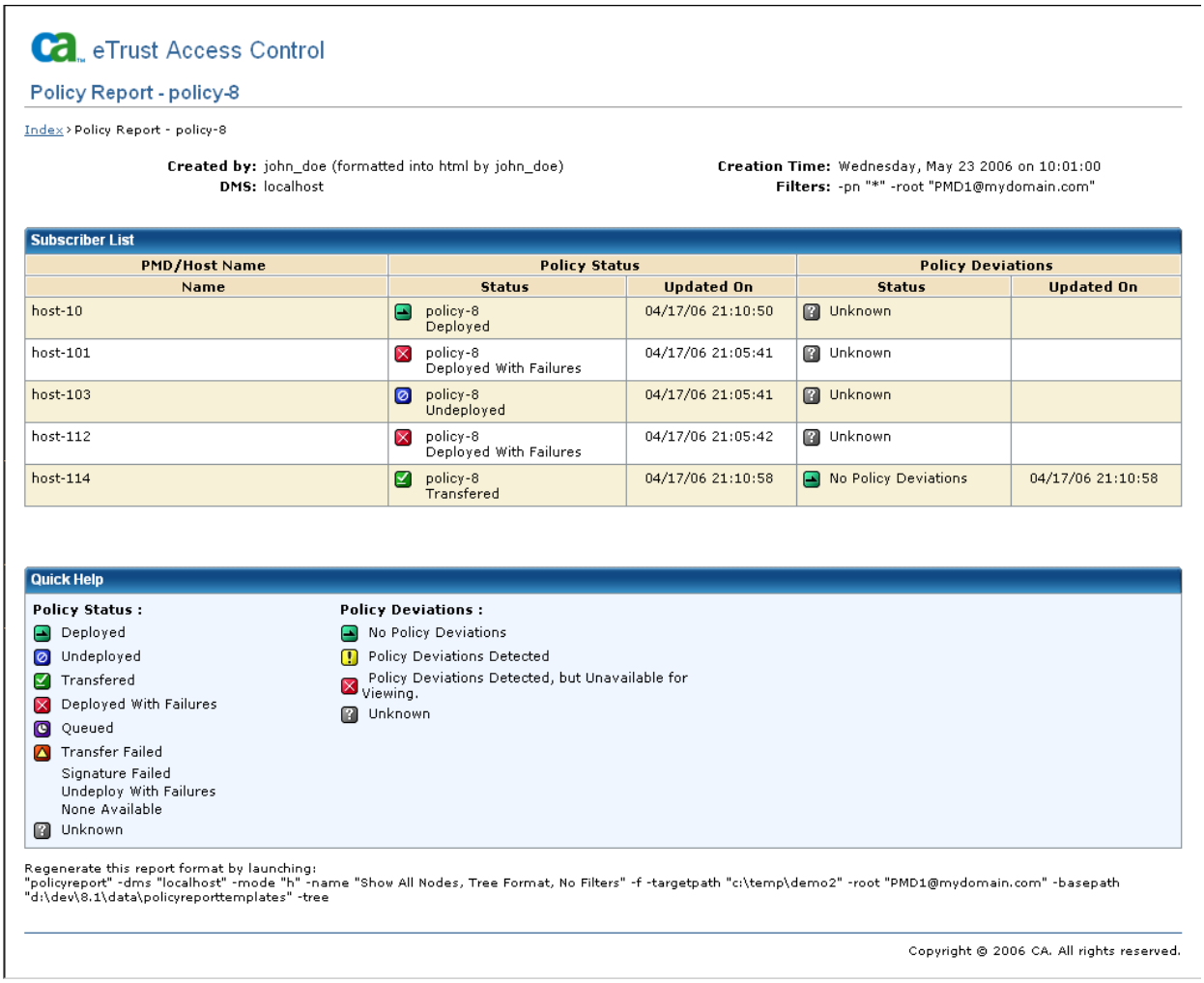
Policy Status :
- Deployed
- Undeployed
- Transfered
- Deployed With Failures
- Queued
- Transfer Failed
- Signature Failed
- Undeploy With Failures
- None Available
- Unknown

Policy Deviations :
- No Policy Deviations
- Policy Deviations Detected
- Policy Deviations Detected, but Unavailable for Viewing.
- Unknown

Regenerate this report format by launching:
"policyreport" -dms "localhost" -mode "h" -name "Show All Nodes, Tree Format, No Filters" -f -targetpath "c:\temp\demo2" -root "PMD1@mydomain.com" -basepath "d:\dev\8.1\data\policyreporttemplates" -tree

- ポリシー別

  ポリシー レポートは、ポリシー中心の情報を提供します。 このモードは、環境内の 1 つまたは複数のポリシーのステータスを確認する場合に使用します。



eTrust Access Control

Policy Report - policy-8

Index > Policy Report - policy-8

Created by: john_doe (formatted into html by john_doe)
DMS: localhost

Creation Time: Wednesday, May 23 2006 on 10:01:00
Filters: -pn "*" -root "PMD1@mydomain.com"

Subscriber List

PMD/Host Name	Policy Status		Policy Deviations	
Name	Status	Updated On	Status	Updated On
host-10	policy-8 Deployed	04/17/06 21:10:50	Unknown	
host-101	policy-8 Deployed With Failures	04/17/06 21:05:41	Unknown	
host-103	policy-8 Undeployed	04/17/06 21:05:41	Unknown	
host-112	policy-8 Deployed With Failures	04/17/06 21:05:42	Unknown	
host-114	policy-8 Transfered	04/17/06 21:10:58	No Policy Deviations	04/17/06 21:10:58

Quick Help

Policy Status :
- Deployed
- Undeployed
- Transfered
- Deployed With Failures
- Queued
- Transfer Failed
- Signature Failed
- Undeploy With Failures
- None Available
- Unknown

Policy Deviations :
- No Policy Deviations
- Policy Deviations Detected
- Policy Deviations Detected, but Unavailable for Viewing.
- Unknown

Regenerate this report format by launching:
"policyreport" -dms "localhost" -mode "h" -name "Show All Nodes, Tree Format, No Filters" -f -targetpath "c:\temp\demo2" -root "PMD1@mydomain.com" -basepath "d:\dev\8.1\data\policyreporttemplates" -tree

レポートのタイプ以外に、以下のように出力を指定できます。

- 階層の一部を選択してレポートを生成する。
- 1 つのコンピュータのレポートを生成する。
- ホストの名前、ステータス、またはステータス更新時刻、あるいはポリシーの名前またはステータスでフィルタ処理する（ワイルドカードがサポートされています）。
- 偏差計算結果を含める、または除外する。
- ツリーのような形式を選択する。
- レポート列を非表示にする。

ホスト レポートの作成

ホスト レポートで確認できるのは、各コンピュータが階層内でどのように設定されているか、階層内の各コンピュータのステータス、または、どのコンピュータにどのポリシーがあるか、およびそのステータスです。

ホスト レポートを作成するには、policyreport ユーティリティを h モードで実行します。

```
policyreport -name <name> -mode h -dms <dms_name> -root <pmd1>[,<pmd2>] -tree \
-targetpath <path>
```

注: オプションのフラグを追加して、レポートを調整することもできます。 policyreport ユーティリティの詳細については、「リファレンス ガイド」を参照してください。

例： ホスト名が指定のマスクに一致するコンピュータのレポートを作成する

次の例は、policyreport ユーティリティを使用して以下のタスクを実行する方法を示します。

- 以下のディレクトリにホスト レポートを生成します。

  C:¥eac_data¥reports¥production_March2006

- DMS から情報を取得します。

  mainhost.domain.com コンピュータの **mainDMS**。

- 階層内で以下の PMDB の下にあるコンピュータのみを含めます。

  rootPMD@root.domain.com

- ホスト名が以下の文字列で始まるコンピュータのみを含めます。

  prod

```
policyreport -name production_March2006 -mode h \
-dms mainDMS@mainhost.domain.com -root rootPMD@root.domain.com -hn prod* \
-targetpath C:\eac_data\reports
```

policyreport ユーティリティは、-targetpath オプションで指定したディレクトリ（C:¥eac_data¥reports）の下の、-name オプションで指定したサブディレクトリ（production_March2006）に、レポートを格納します。 出力ディレクトリにレポートがすでに存在する場合でも、レポートを作成する -f オプションを追加することで、後からこのレポートを更新できます。

注: **-tree** フラグも指定すると、レポートに階層がグラフィカル表示されます。 この場合、ホスト名が指定のマスクに一致しない親も含め、すべてのコンピュータの親がレポートに含まれます。

例： 特定の日付範囲内で最後にステータスが変更されたコンピュータのレポートを作成する

次の例は、policyreport ユーティリティを使用して以下のタスクを実行する方法を示します。

- 以下のディレクトリにホスト レポートを生成します。

  C:¥eac_data¥reports¥Feb06-Mar06

- DMS から情報を取得します。

  mainhost.domain.com コンピュータの **mainDMS**。

- 階層内で以下の PMDB の下にあるコンピュータのみを含めます。

  rootPMD@root.domain.com

- 2006 年 2 月にホスト ステータスが最後に更新されたコンピュータのみを含めます。

```
policyreport -name Feb06-Mar06 -mode h \
-dms mainDMS@mainhost.domain.com -root rootPMD@root.domain.com \
-sd 01-02-2006 -ed 28-02-2006 -targetpath C:\eac_data\reports
```

例： ポリシー偏差結果を再計算するレポートを作成し、現在の作業ディレクトリに格納する

次の例は、policyreport ユーティリティを使用して以下のタスクを実行する方法を示します。

- 以下のディレクトリにホスト レポートを生成します。

  <working_directory>**/hierarchy_20March06**

- DMS から情報を取得します。

  mainhost.domain.com コンピュータの **mainDMS**。

- 階層内で以下の PMDB の下にあるコンピュータのみを含めます。

  rootPMD@root.domain.com

- 偏差計算結果を含めます。
- インデントを使用して階層をグラフィカル表示します。

```
policyreport -name hierarchy_20March06 -mode h -dms mainDMS@mainhost.domain.com \
-root rootPMD@root.domain.com -targetpath -dev -tree
```

ポリシー レポートの作成

ポリシー レポートでは、どのポリシーがどのコンピュータに展開されているかを確認できます。

ポリシー レポートを作成するには、policyreport ユーティリティを p モードで実行します。

```
policyreport -name <name> -mode p -dms <dms_name> -root <pmd1>[,<pmd2>] \
-targetpath <path>
```

注: オプションのフラグを追加して、レポートを調整することもできます。 policyreport ユーティリティの詳細については、「リファレンス ガイド」を参照してください。

例： 指定したポリシーのすべてのバージョンのレポートを作成する

次の例は、policyreport ユーティリティを使用して以下のタスクを実行する方法を示します。

- 以下のディレクトリにポリシー レポートを生成します。

  C:¥eac_data¥reports¥iis5Policies_March2006

- 以下の DMS から情報を取得します。

  mainhost.domain.com コンピュータの **mainDMS**。

- 階層内で以下の PMDB の下にあるコンピュータ (サブスクライバ) のみを含めます。

  rootPMD@root.domain.com

- 以下のポリシーのバージョンのみを含めます。

  iis5

```
policyreport -name prodPolicies_March2006 -mode p \
-dms mainDMS@mainhost.domain.com -root rootPMD@root.domain.com -pn iis5#* \
-targetpath C:\eac_data\reports
```

policyreport ユーティリティは、-targetpath オプションで指定したディレクトリ (C:¥eac_data¥reports) の下の、-name オプションで指定したサブディレクトリ (iis5Policies_March2006) に、レポートを格納します。 出力ディレクトリにレポートがすでに存在する場合でも、レポートを作成する -f オプションを追加することで、後からこのレポートを更新できます。

例： 展開されたがエラーがあるポリシーのレポートを作成する

次の例は、policyreport ユーティリティを使用して以下のタスクを実行する方法を示します。

- 以下のディレクトリにポリシー レポートを生成します。

  C:¥eac_data¥reports¥policyErrors

- DMS から情報を取得します。

  mainhost.domain.com コンピュータの **mainDMS**。

- 階層内で以下の PMDB の下にあるコンピュータ（サブスクライバ）のみを含めます。

  rootPMD@root.domain.com

- 展開されたがエラーがある（Failed）ポリシーのみを含めます。

```
policyreport -name policiesErrors -mode p \
-dms mainDMS@mainhost.domain.com -root rootPMD@root.domain.com \
-pstat "Failed" -targetpath C:\eac_data\reports
```

ポリシーまたはホスト レポートの表示

レポートを生成したら、レポートが格納されているフォルダに移動し、レポートを開いてブラウザで表示する必要があります。

ポリシーまたはホスト レポートを表示するには

1. レポートが格納されているフォルダに移動します。

   これは、<target_path>/<report_name>/html です。

   ここで、<target_path> は、レポート生成時に -targetpath フラグで指定したディレクトリ、<report_name> は -name フラグで指定したレポート名です。

2. **index.html** ファイルを開き、ブラウザで表示します。

   レポートのメイン ページがブラウザに表示されます。

ポリシー偏差計算のしくみ

拡張ポリシー管理およびレポートでは、エンドポイントに展開されるポリシー ルールと、エンドポイントに適用されている実際のポリシー ルールの間の違いを確認できます。これにより、ポリシーの展開に関する問題に対処できます。

ポリシー偏差計算は、各エンドポイントで実行され、以下のアクションが行われます。

1. エンドポイントに展開されるルールのリストをローカル ホストから取得します。

   これらは、展開される各ポリシーに指定されたルールです。展開される各ポリシーの POLICY オブジェクトに関連付けられたローカルの RULESET オブジェクトに指定されています。

2. これらの各ルールがエンドポイントに適用されるかどうかをチェックします。

   重要: 偏差計算では、Windows (ネイティブ) ルールが適用されるかどうかはチェックされません。データベースからオブジェクト(ユーザまたはオブジェクト属性、ユーザまたはリソース権限、あるいは実際のユーザまたはリソース) を削除するルールも無視されます。 たとえば、偏差計算では、以下のルールが適用されるかどうかは確認できません。
   rr FILE C:¥tmp¥tmp.txt

3. (オプション) ローカルの HNODE オブジェクトに関連付けられたポリシーと、使用可能な最初の DMS にあるポリシーを比較します。

   通常、偏差計算機能はローカル ホスト上でのみ偏差をチェックします。 -strict オプションを指定すると、偏差計算機能はローカルのポリシーとリストの最初の DMS にあるポリシーも比較します。 比較される内容は以下のとおりです。

   a. ローカル ホストを表す HNODE オブジェクトに関連付けられたポリシーのリスト。

   b. HNODE オブジェクトに関連付けられた各 POLICY オブジェクトのポリシーのステータス。

   c. HNODE オブジェクトに関連付けられた各 POLICY オブジェクトのポリシーのシグネチャ。

4. 以下の 2 ファイルが出力されます。

   - <eTrustACDir>¥data¥devcalc¥deviation.log

     最後の偏差計算で収集されたログとエラー メッセージ。

   - <eTrustACDir>¥data¥devcalc¥deviation.dat

     ポリシーとその偏差のリスト。

   注: eTrust AC は監査イベントも送信します。これは、**seaudit -a** を使用して表示できます。 seaudit ユーティリティの詳細については、「リファレンス ガイド」を参照してください。

5. 検出された偏差を 1 つまたは複数の DMS に通知します。

通知先の DMS を手動で指定するか(-dms オプション)、DMS が指定されていない場合は、偏差計算機能はローカルの eTrust AC データベースに指定された DMS リストを使用します。

エンドポイントのポリシー偏差計算の設定

カスタム インストールを使用して eTrust AC をインストールまたはアップグレードする場合、インストール プロセスで拡張ポリシー管理オプションを選択すると、ポリシー偏差計算機能が設定されます。 インストール後に、偏差を計算して偏差ステータス通知を指定の DMS データベースに送信するようにデータベースを設定することもできます。

エンドポイントにポリシー偏差計算を設定するには

注: この操作は、この eTrust AC コンピュータをインストールまたはアップグレードするときに拡張ポリシー管理オプションを選択しなかった場合にのみ行う必要があります。 eTrust AC のインストールの詳細については、「実装ガイド」を参照してください。

1. selang コマンド ウィンドウを開きます。

   selang コマンド ウィンドウが開き、selang コマンドを入力できるようになります。

2. 次のコマンドを入力します。

   ```
   nu ("+devcalc") admin auditor
   ```

   これで、ADMIN 属性を持つ +devcalc という名前の新しいユーザが作成されます。 このユーザを使用して、eTrust AC で偏差計算を実行します。

3. 次のコマンドを入力します。

   ```
   nr SPECIALPGM ("<devcalc.exe_path>") seosuid("+devcalc") nativeuid("SYSTEM")
   ```

   ここで、<devcalc.exe_path> は eTrust AC インストール ディレクトリの bin ディレクトリにある devcalc.exe アプリケーションの完全パスです。

   これで、偏差計算機能用の新しい SPECIALPGM リソースが作成され、この SPECIALPGM を実行する権限を持つ論理ユーザとネイティブ Windows ユーザが指定されます。

4. 次のコマンドを入力します。

   ```
   so dms+(<DMS1>[,<DMS2>)
   ```

   ここで、<DMSx> は、偏差計算でポリシー偏差ステータス通知を送信する DMS の名前です。 各 DMS は DMS_name@hostname の形式で指定する必要があります。

例： 中央の DMS にポリシー偏差ステータスを送信するようにエンドポイントを設定する

次の例は、標準インストール（およびデフォルトのインストール ディレクトリ）を使用してインストールしたエンドポイントで以下のタスクを行うために実行する必要があるコマンドを示します。

- 偏差計算を実行できるようにエンドポイントを設定します。
- ポリシー偏差ステータスを以下の DMS に送信します。

  centralhost.com コンピュータ上の **prodDMS**。

```
nu ("+devcalc") admin auditor
nr SPECIALPGM ("C:¥Program Files¥eTrustAccessControl¥bin¥devcalc.exe") ¥
seosuid("+devcalc") nativeuid("SYSTEM")
so dms+(prodDMS@centralhost.com)
```

ポリシーの偏差ログおよびエラー ファイル

ポリシー偏差計算では、各偏差計算の実行時に新しいログが作成されます。 このログにはエラー メッセージも含まれ、<eTrustACDir>¥data¥devcalc¥deviation.log に格納されます。

このログは、レポートに示された（DMS から取得した）偏差が、最後に偏差計算が実行された時点から収集されていない場合に使用します。 このログで、偏差計算結果が DMS に送信されなかった理由を診断できます。

重要: 偏差ログに「エラー: DB ライブラリの初期化に失敗しました。データベースが開いています」というエラーが含まれる場合、データベースのインデックス ファイルを再作成する必要があります。 そのためには、selang を終了し、<eTrustACDir>¥data¥devcalc¥init_ac_db ディレクトリから以下のコマンドを実行して、偏差計算を再実行します (P. 123)。

```
selang -l -d
```

例： 偏差ログおよびエラー ファイル

偏差ログおよびエラー ファイルの例を以下に示します。

```
開始時刻: Mon Jan 23 13:04:48 2006
WARNING,¥"DMS ホスト名の取得に失敗しました。偏差はローカルに保存されます¥"
ポリシー 'iis8#02' の偏差が見つかりました
終了時刻: Mon Jan 23 13:05:04 2006
```

ポリシー偏差データ ファイル

ポリシー偏差計算では、ポリシーとその偏差のリストを含むデータ ファイルが作成されます。 このデータ ファイルは <eTrustACDir>¥data¥devcalc¥deviation.dat に格納されます。

注: データ ファイルに含まれるポリシーのリストは、偏差が計算されるポリシーに応じて異なります(デフォルトでは、すべてのポリシーと、エンドポイントのすべてのポリシー バージョン)。

重要: 偏差計算では、Windows(ネイティブ)ルールが適用されるかどうかはチェックされません。データベースからオブジェクト(ユーザまたはオブジェクト属性、ユーザまたはリソース権限、あるいは実際のユーザまたはリソース)を削除するルールも無視されます。 たとえば、偏差計算では、以下のルールが適用されるかどうかは確認できません。

```
rr FILE C:¥tmp¥tmp.txt
```

偏差ステータスは(偏差があってもなくても)DMS に送信されますが、実際の偏差はローカルに保存されます。 レポートの作成時に、実際の偏差結果をこのファイルから取得してレポートに追加できます。

ポリシー偏差データ ファイルに以下の行が表示されることがあります。

日付

偏差計算の日付。

形式: DATE, DDD MMM DD hh:mm:ss YYYY

Strict

偏差計算が -strict オプションを指定して実行されたことを示します。

形式: STRICT, DMS@hostname, policy_name#xx, {1|0}

ここで、{1|0} は、ローカルの HNODE オブジェクトに関連付けられたポリシーと、DMS@hostname(使用可能な最初の DMS)の HNODE オブジェクトに関連付けられたポリシーとの間に偏差が検出されたか(1)されなかったか(0)を意味します。

ポリシーの開始

このポリシーの偏差を定義するポリシー ブロックの開始です。

形式: POLICYSTART, policy_name#xx

違い

検出されたポリシーの偏差を示します。 偏差に対応するポリシーの名前は、この行の上の直近の**ポリシー行**にあります。

偏差には 4 つのタイプがあり、これらを以下の表に示します。

偏差のタイプ	形式
クラスが見つからない	DIFF, (<class_name>), (*), (*), (*)
オブジェクトが見つからない	DIFF, (<class_name>), (<object_name>), (*), (*)
プロパティが見つからない	DIFF, (<class_name>), (<object_name>), (<property_name>), (*)
プロパティ値の不一致	DIFF, (<class_name>), (<object_name>), (<property_name>), (<expected_value>)

ポリシーの終了

このポリシーの偏差を定義するポリシー ブロックの終了です。

形式: POLICYEND, policy_name#xx, {1|0}

ここで、{1|0} は、偏差が検出されたか(1)されなかったか(0)を意味します。

警告

警告を示します。

形式: WARNING, "warning_text"

例： 偏差データ ファイル

```
Date, Sun Mar 19 08:30:00 2006
WARNING, "DMS ホスト名の取得に失敗しました。偏差はローカルに保存されます"
POLICYSTART, iis8#02
DIFF, (USER), (am), (*), (*)
POLICYEND, iis8#02, 1
```

偏差計算の実行

DMS にポリシー偏差ステータスの最近の情報が含まれるように、偏差計算を定期的に実行する必要があります。 レポート要件を満たす間隔でポリシー偏差計算が行われるようにスケジュールすることをお勧めします。

エンドポイントで偏差計算を実行するには、selang ウィンドウで以下のコマンドを入力します。

```
start DEVCALC
```

例： 定期的な偏差計算をスケジュールする

次の例は、Solaris で以下のような偏差計算タスクを作成する方法を示します。

- 毎日午前 0 時に実行します。
- 偏差ステータスを DMS に送信します。

  mainhost.domain.com コンピュータの **mainDMS**。

そのためには、以下の手順に従います。

1. 以下の行を含むバッチ ファイルを作成します。

   ```
   selang -c "start DEVCALC params('-dms mainDMS@mainhost.domain.com')"
   ```

2. 以下のように選択して、スケジュールされたタスクを追加します。
   - 新しいバッチ ファイルを参照し、選択します。
   - このタスクを毎日実行します。
   - 開始時刻: 12:00 AM

PMDB と Unicenter の統合

PMDB を Unicenter TNG と統合すると、さまざまな Unicenter TNG コンポーネント(コマンド プロセッサ、Event Management、Workload Management など)により Unicenter TNG オブジェクトが操作される際のセキュリティ保護のルールを作成できます。

この統合は手動で実行する必要があります。

PMDB を Unicenter TNG と統合するには

1. PMDB を作成します。
2. 次のコマンドを使用して、Unicenter Security オプションを PMDB に移行します。

   ```
   MigOpts pmdb-name
   ```

   pmdb-name は PMDB の名前です。

   注: この手順は、Unicenter セキュリティを使用し、かつ eTrust AC のインストール中に[Unicenter Integration]の[Security Data Migration]を選択した場合にのみ実行する必要があります。 Unicenter Security を使用しなかった場合、セキュリティ オプションを設定していないので、PMDB に移行する必要のあるオプションはありません。
3. 以下のコマンドを使用して、ユーザ定義の Unicenter TNG のアセット タイプに関するクラスを作成します。

   ```
   defclass.bat. pmdb-name
   ```

   pmdb-name は PMDB の名前です。

   注: この手順は、Unicenter セキュリティを使用し、かつユーザ定義のアセット タイプを作成した場合にのみ実行する必要があります。 eTrust AC のインストール時に Unicenter Integration を選択した場合、Unicenter TNG のアセット タイプは新しい PMDB を作成するたびに自動的に定義されます。

第 7 章: トランザクション マネージャの使用法

このセクションには、以下のトピックが含まれます。

トランザクション マネージャ

トランザクション マネージャは、eTrust AC、UNIX、および Windows のセキュリティを管理するためのツールです。トランザクション マネージャは、ローカル ホストで実行される eTrust AC のトランザクションを自動的に複数のホストに送信します。このトランザクション モードは、Policy Model に代わる、または Policy Model を補助することを目的としたスピーディで効率的な方法です。このトランザクション モードは、すべてのサブスクライバのセキュリティ データベースに対して同じ変更内容の伝達を保証するものではありませんが、Policy Model より使いやすく、Policy Model 階層の一部として定義されていない複数のデータベースに対して変更を行う場合には特に有効です。

トランザクション マネージャのセットアップ

トランザクション マネージャを使用する前に、以下の点について準備しておくことが必要です。

1. ローカル ホスト以外に、アクセス先の各リモート ホストに対する ADMIN 権限もあることを確認します。

   アクセス先の各ホストに、管理元のコンピュータの TERMINAL レコードを作成します。

   これらの要件は、リモート ホストを管理する場合も同じです。

   トランザクション マネージャを有効にします。ポリシー マネージャから[ツール]-[オプション]を選択し、[トランザクション マネージャ]タブを開きます。[マルチホスト トランザクションを有効にする]チェック ボックスをオンにします。また、有効にするトランザクション マネージャのオプションを選択することもできます。

2. ターゲット ホスト ファイルを作成します。

マルチホスト トランザクションのオプション

トランザクション マネージャは、プログラム バーから開くことができるすべてのウィンドウ（[Security Policies]および[Audit]を除く）で使用できます。 デフォルトでは、[Users]、[Groups]、および[Resources]チェック ボックスがオンになっています。

トランザクション マネージャのオプションを使用して、トランザクション モードの動作をカスタマイズできます。 以下のオプションがあります。

[General]オプション

[Stop sending data at first error]

デフォルトでは、エラーが発生してもトランザクションの送信を続行します。これにより、可能な限り多くのトランザクションを伝達し、後でエラーを処理することができます。 ただし、多数のホストにトランザクションを送信していて、1 台のホストの障害によって他のホストにも障害が起きることが予想される場合などは、最初にエラーが発生した時点で伝達を停止した方が、エラー処理に要する時間は短くなります。

[終了時にトランザクション マネージャを閉じる]

このチェック ボックスをオンにすると、トランザクション マネージャは、キューにあるトランザクションの送信が完了しているかどうかにかかわらず、ポリシー マネージャと一緒に終了します。 このオプションのデフォルト設定では、ポリシー マネージャが終了してもトランザクション マネージャは終了しません。

注: トランザクション マネージャのメモリは揮発性であるため、トランザクション マネージャのトランザクション ログは、終了時に消去されます。

[Activate transaction mode upon startup]

ポリシー マネージャを起動すると、常に、トランザクション マネージャが起動されます。 ただし、デフォルトでは、ツール バーの[トランザクション モード]ボタンをクリックしないと、トランザクション モードにはなりません。 このチェック ボックスをオンにした場合は、ポリシー マネージャの起動時に自動的にトランザクション モードになります。

[Target host file]

このオプションを使用すると、ターゲット ホスト ファイルのディレクトリ パスを設定できます。 デフォルトのパスは、eTrustACDir¥data¥hosts.txt です（eTrustACDir は、eTrust AC をインストールしたディレクトリです）。

[Interval (in seconds) between each refresh...]

このオプションを使用して、[トランザクション マネージャのステータス]ウィンドウでトランザクション マネージャを監視する間隔を調節できます。 デフォルトは 10 秒ですが、トランザクションが短いと、進行状況が表示されない場合があります。

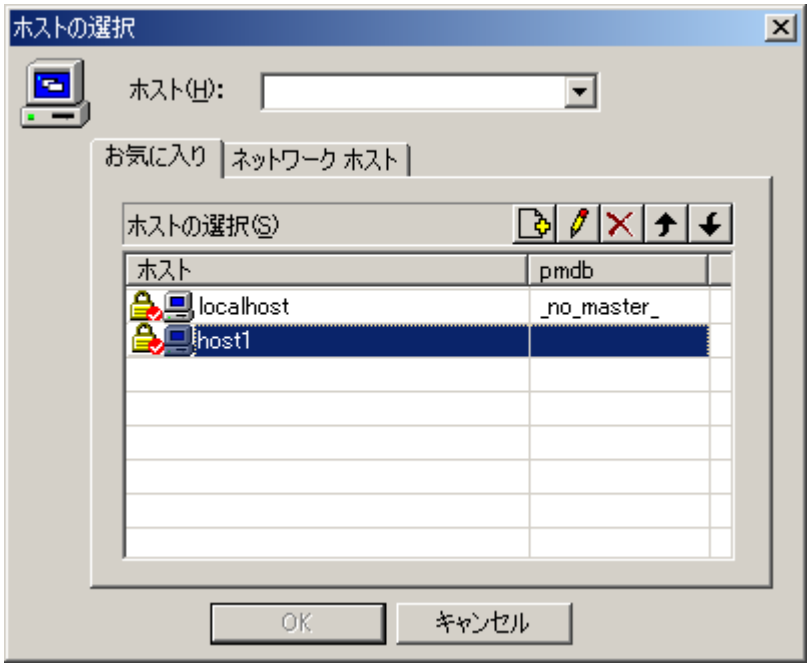
コマンドとスクリプト

通常、ポリシー マネージャのトランザクションでは、トランザクション モードを使用します。トランザクション モードを使用して selang のコマンドやスクリプトを複数のホストに送信することもできます。 これらのチェック ボックスをオンにすると、[Commands and Scripts]ダイアログ ボックス([Tools]-[Execute Command])で、トランザクション モードを使用して selang のコマンドを複数のホストに送信できます。

ターゲット ホスト ファイルのセットアップ

ターゲット ホスト ファイルは、トランザクション モードで作業しているときに、トランザクションを受信するホストまたはホストのグループを指定します。

[Host Selection]ウィンドウの［Favorites］リストまたは［Network Neighborhood］からホストを追加します。



ホストにはローカル データベースまたは PMDB を指定できます。 ホストのグループを作成して、選択を効率化することもできます。 グループ名をクリックすると、そのグループのすべてのメンバが選択されます。 また、ホストを個別に選択または選択解除することも可能です。［OK］をクリックすると、選択内容がすぐに有効になり、次に変更するまでその選択が有効になります。 別のホスト グループにトランザクションを送信する場合は、そのたびにターゲット ホスト ファイルを手動で再設定する必要があります。

注：［トランザクション モード］を有効にすると、［ホストの選択］の設定は、トランザクション マネージャの他に、ユーザ コピー ウィザードおよびグループ コピー ウィザードにも適用されます。

トランザクション モードでの実行

トランザクション マネージャを有効にし、ターゲット ホスト ファイルを作成した後、ツール バーの[トランザクション モード]アイコンをクリックします。 ローカル データベースで実行するすべてのトランザクションは、選択したホストにも伝達されます。 たとえば、[Users]ウィンドウでユーザの名前を選択し、ツール バーの[Delete]をクリックしてそのユーザを削除すると、(通常どおり)ローカル ホスト データベースですぐにトランザクションが実行され、ターゲット ホスト ファイルに指定されているホストに自動的に伝達されます。

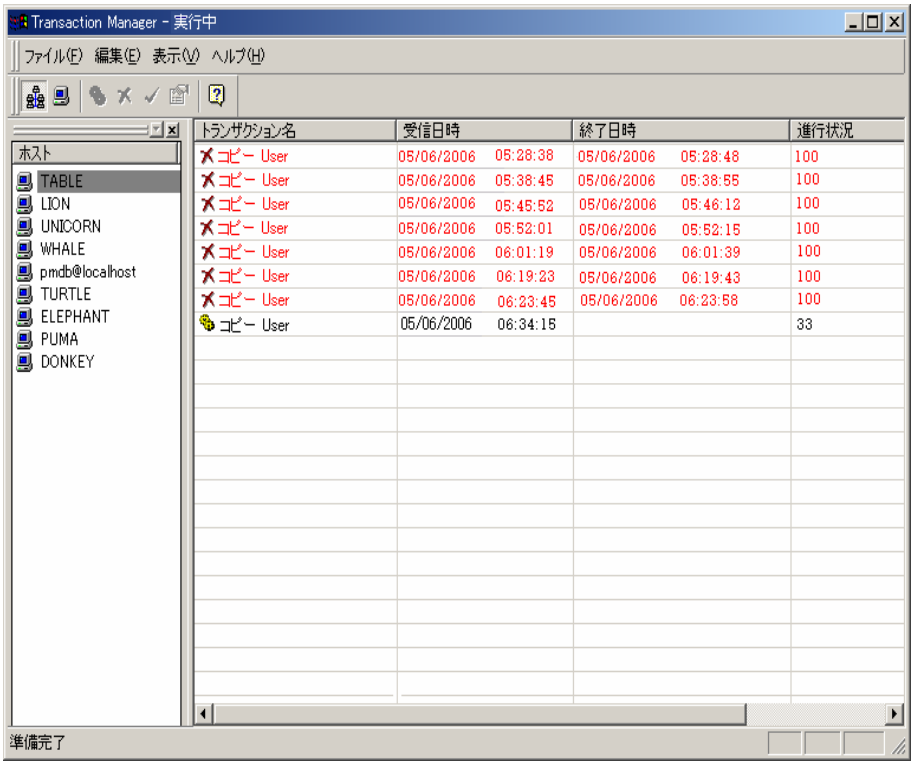
伝達の進行状況は、[Task Manager Status]ウィンドウで追跡できます。 実行時間の長いトランザクションを一時停止するには、タスクバー トレイの[トランザクション マネージャ]アイコンを右クリックし、[Suspend Transaction Manager]を選択します。 [Resume Transaction Manager]を選択すると、伝達が再開されます。 [トランザクション マネージャ]ウィンドウを閉じても、トランザクション マネージャ アプリケーションは終了しません(右上の[閉じる]ボタンをクリックした場合も同じです)。 トランザクション マネージャを終了するには、タスクバー トレイの[トランザクション マネージャ]アイコンを右クリックし、[Exit]を選択します。 トランザクション マネージャを終了すると、トランザクション マネージャのログからすべてのトランザクションが消去されることに注意してください。

[トランザクション マネージャ]ウィンドウ

Windows のタスクバーにある[トランザクション マネージャ]アイコンをダブルクリックすると、[トランザクション マネージャ]ウィンドウがアクティブになります。 このウィンドウには、複数のホストに伝達されたすべてのトランザクションのログが表示されます。 タスク マネージャのメモリは揮発性であるため、現在のセッションのトランザクションのみが表示されます。 表示方法は、[Host Status]と[Host Bar]の 2 種類から選択できます。 [View]メニューから選択するか、ツール バーのいずれか一方のアイコンをクリックして選択します。 これらのアイコンは両方とも同じ働きをします。いずれかのアイコンをクリックするたびに表示方法が切り替わります。 ツール バーにあるボタンを使用して、表示方法の選択、トランザクションの再実行、トランザクションの削除または削除取り消し、またはトランザクションのプロパティ表示を行うことができます。 キューにあるトランザクションが削除された場合、トランザクション マネージャは、削除されたトランザクションの処理を省略して、キュー内の次のトランザクションを処理します。 いつでも削除取り消しを実行してトランザクションの伝達を続行できます。 実行中のトランザクションを削除することはできません。

Host Status Bar 表示

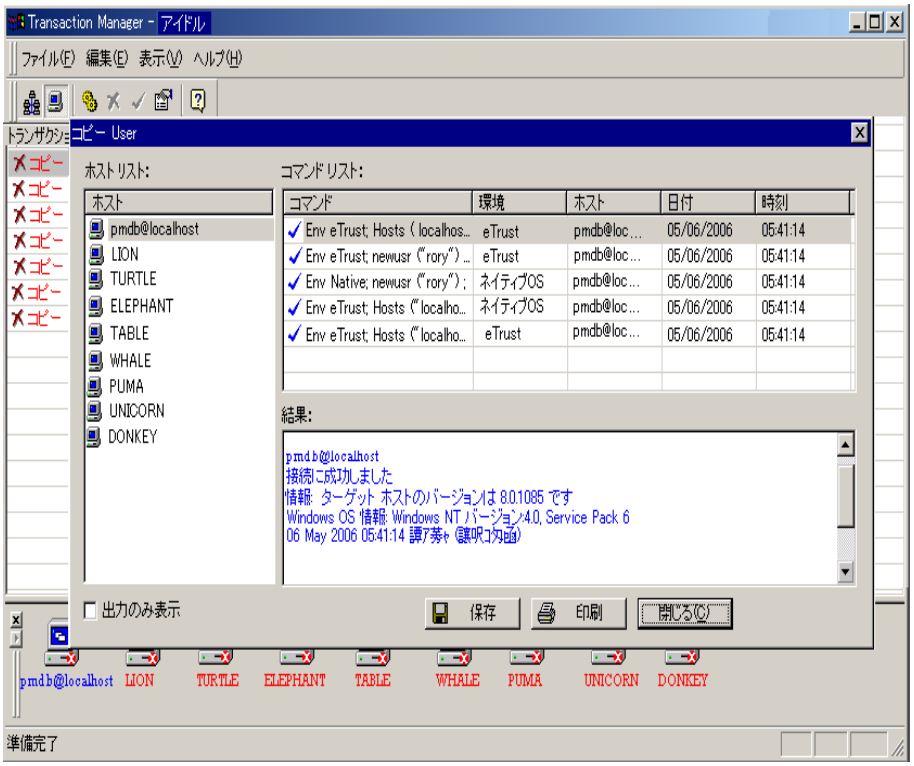
Host Status Bar 表示では、選択した各ホストのアイコンがウィンドウの左側のステータスバーに表示されます。 ホストのアイコンをクリックすると、そのホストに関係するトランザクションの一覧が表示されます。 トランザクションをダブルクリックすると、[更新]ダイアログ ボックスが表示されます。このダイアログ ボックスには、[コマンド リスト]テキスト ボックスと[結果]テキスト ボックスがあります。 [Command List]テキスト ボックスでコマンドを選択すると、そのコマンドの実行結果が[Result]テキスト ボックスに表示されます。結果は、印刷するか、ファイルに保存することができます。



Host Bar 表示

Host Bar 表示では、トランザクションの詳細を示す行がウィンドウの幅いっぱいに表示され、選択したホストのアイコンが画面の下部に表示されます。 トランザクションをダブルクリックすると、[更新]ダイアログ ボックスが表示されます。このダイアログ ボックスには、ホスト リスト バー、[コマンド リスト]テキスト ボックス、および[結果]テキスト ボックスがあります。 ホストのアイコンを選択すると、そのホストに関係するコマンドが[Command List]に表示されます。 コマンドを選択すると、そのコマンドの実行結果が[Result]に表示されます。

または、トランザクションを選択し、ウィンドウの下部にあるホストのアイコンをダブルクリックして、 [Update]ダイアログ ボックスを表示させることもできます。



第 8 章: 監視と監査

このセクションには、以下のトピックが含まれます。

セキュリティ監査者

セキュリティ監査者およびシステム管理者の最も重要な仕事の 1 つは、システムでのアクティビティを監査または監視して、疑わしいアクティビティや不正なアクティビティを検出することです。 セキュリティで保護された環境において、セキュリティ監査は重要な役割を果たします。eTrust AC には、以下のセキュリティ監査機能があります。

- システムにアクセスしたユーザ、アクセスされたリソース、およびその日時を特定する情報を提供する機能
- セキュリティ違反の試みがあったときは、その試みが失敗に終わった場合でも、適切なユーザに通知および警告する機能
- セキュリティ ルールに対して行われた変更の内容と、変更を行ったユーザを表示する機能
- アクセス ルールを適用する前に、ルールの影響をテストする機能

eTrust AC での監査は、実社会での監査をモデルにしています。 つまり、セキュリティ監査者は、システム管理者およびセキュリティ管理者とは独立して任務を実行します。ただし、運用する環境に最も適したモデルが他にある場合は、この実装を変更できます。

セキュリティ監査者は、AUDITOR 属性が割り当てられているユーザです。 セキュリティ監査者として定義されているユーザは、ユーザおよびリソースに割り当てられた監査ルールの変更などの監査タスクを実行できます。 また、このユーザには ADMIN 属性がなくても eTrust AC の監査ユーティリティを使用できる権限も与えられています。

Access Control のアクティビティの監視

eTrust AC トレースは、eTrust AC によって実行されるすべてのアクションを確認できるリアルタイム ログです。トレース レコードは、eTrustACDir¥log¥seosd.trace に蓄積されます（eTrustACDir は eTrust AC をインストールしたディレクトリです）。

または、次のレジストリ サブキーで trace_file 値として指定されたファイルに蓄積されます。

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\eTrustAccessControl\SeOSD\
```

トレース ファイルのレコードはフィルタ処理できますが、トレース機能は本来セキュリティ監査ではなくシステム監視を目的として設計されたメカニズムです。

デフォルトでは、eTrust AC の初期化時にのみトレース メッセージが生成されます。eTrust AC の初期化が終わると、トレース メカニズムは停止し、トレース メッセージは生成されません。

トレース レコードのフィルタ処理

トレース フィルタ ファイルを使用すると、特定の種類のアクティビティがトレース ファイルに書き込まれないように指定できます。トレース フィルタ ファイルは、trace_filter 値を使用して、次のレジストリ キーで指定します。

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\eTrustAccessControl\SeOSD
```

デフォルト値は、eTrustACDir¥log¥trcfilter.ini です（eTrustACDir は、eTrust AC をインストールしたディレクトリです）。

重要: eTrust AC のインストール時には、*seosd.trace* という 1 行が書き込まれたトレース フィルタ ファイルが作成されます。このレコードは、決して削除しないでください。

トレース フィルタ ファイル内の各行は、**トレース対象としない**アクセスまたはアクティビティを表します。たとえば、Microsoft Word へのユーザ アクセスをトレース対象外にするには、トレース フィルタ ファイルに以下の行を追加します。

```
*winword.exe*
```

監査レコードのフィルタ処理

audit.cfg ファイル(eTrustACDir¥data 内にある)を使用して、生成する必要がない監査レコードを定義し、ホスト上の監査レコードをフィルタ処理できます。 各行は、監査情報をフィルタ処理によって除外するためのルールを表します(つまり、各行の基準に一致した監査レコードは、監査ファイルに記録されません)。 このフィルタは、必要なレコードのみを保持して、seos.audit ファイルのサイズを制限する際に有用です。 クラス名、オブジェクト名、ユーザ名、プログラム名、アクセス権、および認証結果をフィルタ処理するルールを設定できます。 eTrust AC Engine(seosd)は、このファイルを起動時に読み込みます。

構文

```
<class>;ログイン情報;<user>;<program-path>;<access-mode>;<auth-result>
```

注: 各列のワイルドカード「*****」は、任意の値を表します。

メッセージが監査ファイルに送信されるとき、そのメッセージが audit.cfg ファイルにある以下のいずれかのエントリと一致するかどうかがチェックされます。

フィールド	ルール
クラス	クラス名は大文字で記述する必要があります。
オブジェクト	リソース名はパターン(*)を使用して記述できます。
ユーザ	ユーザ名はパターン(*)を使用して記述できます。
プログラム パス	使用するプログラムはパターン(*)を使用して記述できます。
アクセス モード	アクセス権は、ルールに従う必要があります。
認証結果	認証結果は P(許可)または D(拒否)になる必要があります。 **注**: 値を「P」に設定すると、警告モードで生成されたリソースについての監査レコードもフィルタ処理の対象になります。

TCP クラスの構文

```
<class>;ログイン情報;<host>;<program-path>;<access-mode>;<auth-result>
```

例: 監査アクセス フィルタ

次の例では、管理者ユーザがファイルの読み込みに成功した場合、メッセージは seosd によって監査ファイルに送信されません。 管理者がファイルを読み込めなかった場合は、seosd によって監査ファイルにメッセージが送信されます。

```
FILE;*;Administrator;*;R;P
```

監査ルールの設定

eTrust AC では、セキュリティ監査のために、データベースに定義されている監査ルールに基づいて、アクセス拒否およびアクセス許可のイベントに関する監査レコードが保存されます。

すべてのアクセサおよびリソースに AUDIT プロパティがあり、このプロパティでは以下の 1 つ以上の値を設定できます。

FAIL

アクセサによるリソースへの失敗したアクセスをログに記録します。

SUCCESS

アクセサによるリソースへの成功したアクセスをログに記録します。

LOGINFAIL

アクセサによる失敗したすべてのログオンをログに記録します（この値はリソースには適用されません）。

LOGINSUCCESS

アクセサによる成功したすべてのログオンをログに記録します（この値はリソースには適用されません）。

ALL

アクセサの FAIL、SUCCESS、LOGINFAIL、および LOGINSUCCESS、またはリソースの FAIL および SUCCESS と同じ情報をログに記録します。

NONE

アクセサまたはリソースに関して、ログに何も記録しません。

データベースにアクセサまたはリソース レコードを作成または更新する場合はいつでも、AUDIT プロパティを指定できます。 また、ログに記録されたイベントを電子メールで通知するかどうか、通知する場合は誰に通知するかを指定することもできます（データベースのレコードの作成および更新には、「管理者インターフェースの使用法」で説明しているポリシー マネージャを使用するか、または「リファレンス ガイド」の章「selang - eTrust AC のコマンド言語」で説明している selang のコマンドを使用できます）。

監査ログのレコードは、これらの監査ルールに従って蓄積されます。 イベントをログに記録するかどうかは、以下のルールに基づいて決定されます。

- リソースまたはアクセサに AUDIT（ALL）が割り当てられている場合は、そのアクセサのすべてのログイン イベント、および eTrust AC によって保護されているリソースに関するすべてのイベントが、アクセスが失敗したか成功したかにかかわらず、ログに記録されます。
- eTrust AC によって保護されているリソースへのアクセスが成功し、アクセサまたはリソースに AUDIT（SUCCESS）が割り当てられている場合は、イベントがログに記録されます。

- eTrust AC によって保護されているリソースへのアクセスが失敗し、アクセサまたはリソースに AUDIT（FAIL）が割り当てられている場合は、イベントがログに記録されます。

Windows での監査ポリシーの設定

アクセサおよびリソースに関するアクセス ルールを設定するだけでなく、監査ログに書き込む Windows イベントを指定できます。 このような監査ポリシーは、組織全体に対して、またはユーザ単位で指定できます。

監査ログ

監査ルールおよび監査ポリシーで定義したイベントまたはアクセスによって作成された監査レコードは、監査ログというファイルを構成します。 以下の Windows レジストリ サブキーの audit_log 値で、監査ログの場所を指定します。

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\eTrustAccessControl\logmgr
```

このキーのデフォルト値は、次のとおりです。

```
*\Program Files\CA\eTrustAccessControl\log\seos.audit
```

eTrust AC のデフォルトでは、監査ログのサイズが 1,024 KB になると、その監査ログファイルの名前が変更されて新しいファイルが作成され、自動的にバックアップが作成されます。 次のサブキーの audit_size 値を変更することにより、バックアップ処理の起動のトリガとなる監査ログのサイズを変更できます。

```
HKEY_LOCAL_MACHINE\Software\ComputerAssociates\eTrustAccessControl\logmgr
```

また、監査ログを定期的に（毎日、毎週、または毎月）バックアップするように設定することもできます。そのためには、以下の Windows レジストリ サブキーの BackUp_Date 値を変更します。

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\eTrustAccessControl\logmgr
```

注: これらのレジストリ サブキーの詳細については、「リファレンス ガイド」を参照してください。

後でイベントを調査できるように、古い監査ログをテープに保管することをお勧めします。

監査ログの使用法

eTrust AC には、監査ログの表示、フィルタ処理、および検索に使用する以下の 2 つの付属ツールがあります。

- ポリシー マネージャ
- seaudit ユーティリティ

監査ログのすべてのレコードを表示することも、フィルタを使用して監査ログから特定のレコードを選択することもできます。

次に、ポリシー マネージャで監査フィルタを使用して監査ログのレコードを表示する方法について説明します。

監査フィルタ

監査ログのレコードは、膨大な数になる場合があります。 eTrust AC で表示するレコード数を減らすには、フィルタを使用して、特定の種類のレコードのみを表示対象として選択します。 時間やイベント タイプなどのさまざまな基準に基づいて、イベントをフィルタ処理できます。

ポリシー マネージャでフィルタを作成するには、フィルタに名前を付け、少なくとも 1 つのスイッチを選択します。 スイッチを追加することもできます。また、オプションの指定は任意で、複数のオプション指定も可能です。 seaudit ユーティリティでレコードをフィルタ処理することもできます。

ポリシー マネージャには、複数の事前定義フィルタが用意されています。また、独自のフィルタを作成することもできます。

eTrust AC のデフォルト設定では、すべての監査フィルタが eTrustACDir¥data¥AuditFilters.flt に保存されます（ eTrustACDir は、eTrust AC をインストールしたディレクトリです）。

自分で作成したフィルタは、このファイルに保存することも、別のファイルに保存することもできます。

スイッチ

INet Host Service

指定されたサービスの指定されたホストから受け取った TCP 要求の INET 監査レコードを一覧表示します。 host および service は、検索対象のホストおよびサービスを特定するマスクです。

LOGON User Terminal

以下の情報を表示します。

- 指定した端末における指定したユーザの LOGIN レコード。 user と terminal はいずれもマスクです。
- 無効なパスワードが複数回入力されたときに、承認エンジンによって作成されるレコード。

Resource Class Resource User

リソース レコードを一覧表示します。 以下の項目を指定できます。

- Class - アクセスされたリソースが属しているクラスを特定するマスクです。
- Resource - アクセスされたリソースの名前を特定するマスクです。
- User - リソースにアクセスしたユーザの名前を特定するマスクです。

開始

eTrust AC サービスの起動メッセージおよび停止メッセージを一覧表示します。

Update（Command、Class、Object、User）

データベース更新の監査レコードを表示します。 以下を指定できます。

- Cmd - 検索対象の selang のコマンドのセットを特定するマスクです。
- Class - 検索対象のクラスを特定するマスクです。
- Object - 検索対象のレコードを特定するマスクです。
- User - コマンドを実行したユーザを特定するマスクです。

Watchdog

Watchdog の監査レコードを一覧表示します。

すべて

トレース機能によって監査ログに送信されたレコードを除く、すべてのレコードを一覧表示します。

オプション

Ending Date

終了日を指定します。 指定された日付より後にログに記録されたレコードは表示されません。

Ending Time

終了時刻を指定します。 指定した時刻より後にログに記録されたレコードは表示されません。

No Failure

失敗したレコードが表示されないように指定します。

No Granted

成功した(許可された)アクセスのレコードが表示されないように指定します。

No Logout

ログアウトのレコードが表示されないように指定します。

Internet Address

TCP/IP レコードのホスト名ではなく、インターネット アドレスが表示されるように指定します。

No Notify

NOTIFY 監査レコードが表示されないように指定します。

No Password

パスワード試行レコードが表示されないように指定します。

Origin host

指定したホストから送信されたレコードのみが表示されるように指定します。 このオプションは、UNIX ワークステーションに接続している場合にのみ適用可能です。

Starting Date

開始日を指定します。 指定された日付より前にログに記録されたレコードは表示されません。

Starting Time

開始時刻を指定します。 指定された時刻より前にログに記録されたレコードは表示されません。

Show Port Number

サービス名ではなくポート番号が表示されるように指定します。

No warning

警告レコードが表示されないように指定します。

事前定義フィルタ

eTrust AC には、以下の事前定義フィルタがあります。

ALL

監査ログのすべてのレコードを表示します。 フィルタ処理は行われません。

Today

今日作成されたレコードをすべて表示します。

Records from the last 2 days

昨日と今日に作成されたすべてのレコードを表示します。

Records from the last 7 days

過去 7 日間に作成されたすべてのレコードを表示します。

Connection to Access Control servicesz

ユーザがポリシー マネージャや selang などの eTrust AC サービスにいつ接続したかを示すレコードを表示します。

注: UNIX ワークステーションに接続している場合は、このフィルタの名前は Login Records となります。 このレコードは、ユーザ ログインを表します。

管理アクティビティ

eTrust AC またはオペレーティング システムのデータベースを更新するすべてのレコードを表示します。 データベースの更新には、すべての種類のレコードの追加、削除、および変更が含まれます。

ユーザ定義のフィルタ

フィルタは、必要な数だけ作成できます。 重要なフィールドを選択し、フィルタに名前を付けます。 eTrust AC では、フィルタは自動的に保存されるので、ポリシー マネージャを起動したときにいつでも繰り返し使用できます。

警告モード

セキュリティ ポリシーを導入する場合、特定のリソース アクセス制約を実際に適用せずに、その制約の動作をテストできると便利です。 これは以下の場合に特に役立ちます。

- 設定しようとしているルールが極端に厳密または寛容かどうかを判定し、その判定に従ってセキュリティ ポリシーを修正する場合
- システム アプリケーションの実行に不都合な影響を与える制約があると疑われる場合

eTrust AC では、制約を指定して、その制約を適用する代わりに警告メッセージを発行することができます。

警告モードの実装

警告モードを実装するには、以下の方法があります。

- テストするルールの影響を受けるすべてのリソース レコードに WARNING パラメータを設定します。
- ポリシー マネージャでリソースを作成または変更するときに、[監査]を選択し、[警告モード] チェック ボックスをオンにします。
- selang の newres、editres、または chres コマンドで WARNING パラメータを指定します

  注: 詳細については、「リファレンス ガイド」の chres コマンド、editres コマンド、および newres コマンドの説明を参照してください。

リソースに対する警告モードが有効で、要求された方法でアクセサにリソースへのアクセスが許可されていない場合、eTrust AC では、警告メッセージが発行され、アクセスがログ(警告モードが有効であったために、違反が認められたことを示します)に記録されて、リソースへのアクセスが許可されます。

注:

- 警告モードの場合、eTrust AC では、リソース グループに対する警告メッセージは作成されません。
- eTrust AC の実装中に警告モードを使用する場合は、監査ログを書き込むための充分なディスク容量があることと、監査ログのサイズ制限の設定が充分な大きさであることを確認してください。

第 9 章: Unicenter の移行と統合

このセクションには、以下のトピックが含まれます。

Unicenter Integration ツールのインストール

eTrust AC は、Unicenter のエンタープライズ マネジメント環境に完全に統合されます。以下のセクションでは、eTrust AC で統合がどのように処理されているかについて説明します。

重要: Unicenter TNG と eTrust AC を統合するには、Unicenter TNG を eTrust AC と同じマシンにインストールする必要があります。

注: Windows 環境における完全なインストール手順については、「実装ガイド」を参照してください。

Unicenter の統合機能

以下のセクションでは、eTrust AC を Unicenter TNG と統合する方法について説明します。

SSF/EMSec API サポート

Windows の EMSec API では、呼び出しが単一の DLL に渡されます。 Unicenter Integration に対する EMSec のサポートは、セキュリティ API の代わりとなる CAUSECR.DLL で構成されます。この DLL は、EMSec API への呼び出しを受け取り、同等の eTrust AC API にこれらの要求を再フォーマットして送ります。 eTrust AC API からのリターン コードはその API に対応する EMSec API のリターン コードに変換され、EMSec API の呼び出し側に制御が返されます。 このアプローチにより、EMSec API を現在使用している既存のアプリケーションの整合性が保護されます。

EMSec サポートは、Unicenter の統合のセットアップ手順が完了すると、有効になります。 Unicenter の統合のセットアップを実行すると、Unicenter のインストール パス（CAIGLBL0000 ディレクトリ）にある現在の CAUSECR.DLL が置き換えられます。 この時点で、EMSec API の要求を受信すると、代替 CAUSER.DLL によってインターセプトされます。また、要求された情報は、eTrust AC API の使用によってシームレスに取得されます。

Unicenter セキュリティ データの移行機能

以下のセクションでは、Unicenter セキュリティ データを eTrust AC に移行する方法について説明します。

Unicenter セキュリティオプションの移行

eTrust AC には、MigOpts.exe というプログラムが同梱されています。このプログラムを使用すると、選択した Unicenter セキュリティのオプションを抽出し、そのオプションに基づいて、対象の eTrust AC データベースをカスタマイズできます。 この機能を有効にするには、Unicenter セキュリティ データの移行のセットアップ手順を使用して、Unicenter の統合を実行する必要があります。 このセットアップ手順は、MigOpts.exe により自動的に実行されます。

注: 以下の Unicenter セキュリティ オプションは、eTrust AC 環境に**完全に**移行できます。

- AUDIT_LOGIN
- MODIFY_PWDNEVEREXP
- PWDQUEUESIZE
- SEC_AUDIT_DBUPDATE
- SEC_AUDIT_SEND
- SEC_PASSWORD_ALPHA
- SSF_MAXPWDVIO
- SSF_MINPWDLEN
- SSF_SECPWEXCL
- USER_DEFSESID
- USER_PWDCHANGE
- USER_PWDCHGMAXDAYS
- USER_PWDCHGMINDAYS
- USER_PWDMAINT

注: **USER_PWDMAINT** は、既存の Unicenter セキュリティ データ特有の eTrust AC 環境に移行されます。eTrust AC では、パスワード情報が保持されますが、このプロセスは自動化されていません。 既存の Unicenter TNG ユーザが Unicenter セキュリティから eTrust AC データベースへエクスポートされる際に、USER_PWDMAINT オプションの値が yes の場合、この手動プロセスは自動的に実行されます。 ただし、移行が完了した後、トレースを必要とするパスワード情報を持つ新規ユーザを管理者が追加する場合、その管理者は__workload__アプリケーション オブジェクトの存在を確認する必要があります。 次に例を示します。

```
eTrust> na __workload__;
```

次に、管理者は、「__workload__」アプリケーション オブジェクトを含めるために、ユーザのログイン情報を更新する必要があります。 次に例を示します。

```
eTrust> el (Username) appl('__workload__');
```

また、ExportTngDb.exe により、**SSF_AUTH** Unicenter セキュリティ オプションのメンバである Unicenter TNG ユーザが eTrust AC 環境に移行されます。その際に、ユーザを eTrust AC に追加する前に、そのユーザに admin 属性が設定されます。

Unicenter セキュリティ データベースの移行

eTrust AC の ExportTngDb.exe では、Unicenter セキュリティ データベースからデータを抽出し、そのデータを eTrust AC コマンドに変換して eTrust AC データベースを作成できます。 ExportTngDb.exe により、以下の項目を移行します。

- Unicenter セキュリティ ユーザ
- Unicenter セキュリティ ユーザ グループ
- Unicenter セキュリティ ルール

注:

- Unicenter Integration および Migration Installation プロセスの実行後に、Unicenter TNG のログイン インターセプトを実行しないことをお勧めします。Unicenter の統合および移行インストール プロセスを正常に実行した後、Unicenter TNG のログイン インターセプトが無効になっていることを確認する必要があります。
- Unicenter TNG のデータ スコーピング ルール（-DT サフィックスの付いた Unicenter TNG のアセット タイプを対象とするルール）は、eTrust AC の移行プロセスではサポートされていません。 移行プロセスでは、このタイプのルールは無視されます。
- Unicenter セキュリティは現在使用されていないため、以下の Unicenter セキュリティのアセット タイプ（CA-USER、CA-ACCESS、CA-USERGROUP、CA-ASSETGROUP、CA-ASSETTYPE、および CA-UPSNODE）に対して実装された Unicenter セキュリティ ルールはどれも使用されていません。 このようなアセット タイプまたはそれらから派生したタイプを対象とするルールは、移行プロセスではすべて無視されます。

ExportTngDb.exe を有効にするには、Unicenter セキュリティ データの移行のセットアップ手順を使用して、Unicenter の統合を実行する必要があります。 このセットアップ手順では、Unicenter セキュリティ データの移行プロセスが自動的に実行されます。

注: すべての Unicenter TNG オブジェクトの作成および変更に関する統計情報は、移行プロセスで失われます。

Unicenter TNG と eTrust AC の製品の相違により、Unicenter セキュリティ **ユーザ**の以下の属性は eTrust AC に移行されません。

統計

以下のユーザ統計情報は eTrust AC ではサポートされていません。

- 最終ログイン統計情報(日時、最後にログインが実行されたノード)
- パスワード変更統計情報(日時、ノード、最後にパスワードを変更したユーザ、およびパスワードの有効期限)
- パスワード違反統計情報(日時、最後に不正にログインされたノード、および最後にログインが成功した後に失敗したログインの数)
- アクセス違反統計情報(日時、最後にアクセス違反が発生したノード、およびアクセス違反の数)
- 停止統計情報(停止日時)

PWDCHANGE VALUE (RANDOM)

無作為なパスワードの生成

UPSSTATGROUP

UPS 端末のグループ

- 以下は、eTrust AC ではサポートされていません。

USERORIGIN

ユーザ元(NIS または Local)

VIOLMODE

違反モード(FAIL、MONITOR、WARN、QUIET)

- eTrust AC では、FAIL モードのみがサポートされています。

VIOLACTION

違反アクション(CANUSER、CANU&LOG、CANU&LOG&SUS)

- eTrust AC では、CANUSER アクションのみがサポートされています。

Unicenter TNG と eTrust AC の製品の違いにより、Unicenter セキュリティ **ルール**の以下の属性は eTrust AC に移行されません。

EXPIRES

ルールの有効期限は、eTrust AC ではサポートされていません。

Unicenter user exit のサポート

移行を支援するために、eTrust AC では、eTrust AC 環境でも変更することなく、既存の Unicenter セキュリティ user exit を実行できます。 移行の一環としてすべての user exit を記述し直す必要はありません。

Unicenter セキュリティおよび eTrust AC で既存の user exit インターフェースのみを使用すると、インストールした各コンポーネントが標準の eTrust AC user exit として登録されます。これにより、対応する Unicenter セキュリティ exit が起動します。

この機能を開始するには、Unicenter セキュリティ データの移行のセットアップ手順を使用して、Unicenter の統合を実行する必要があります。 セットアップ手順を完了すると、この機能はアクティブになります。

注: Unicenter TNG と eTrust AC のアーキテクチャの違いにより、Unicenter セキュリティと eTrust AC の間で比較可能な exit ポイントおよびデータ項目のみがサポートされています。 以下の Unicenter セキュリティ exit ポイントがサポートされています。

EmSec_CredExit()

Unicenter 資格情報認証 exit に対する入力 EmSec_CredExit() は、EMSECSIGNON によってマップされます。 eTrust AC では、この構造体内のユーザおよびノード メンバのデータのみが意味を持ちます。 ユーザ メンバは認証されたユーザ名に設定され、ノード メンバは現在のローカル ノード名に設定されます。EMSECSIGNON 構造の他のすべてのメンバは、バイナリ ゼロに設定されます。 他のパラメータ、詳細なリターン コード、Unicenter Resource Check exit から返されたメッセージは無視されます。

EmSec_PwExitNew()

Unicenter パスワード検証用 exit に対する入力 EmSec_PwExitNew は、ユーザ（パスワードの変更対象のユーザ）、パスワード（新しいパスワード）、およびノード名（eTrust AC でサポートされている、通常のローカル ノード名）で構成されています。失敗した場合には、古い exit、EmSec_PwExit が使用されます。 この exit にはユーザおよびパスワードのみが入力として含まれ、eTrust AC で完全にサポートされています。

EmSecSSFResCheck()

リソース チェック exit に対する入力 EmSecSSFResCheck() は、EMSECRESCHECK によってマップされます。 EMSECRESCHECK のユーザ メンバは、アクセス ユーザの値に設定されます。 EMSECRESCHECK のクラス メンバは、アクセスのリソース クラスの値に設定されます。 EMSECRESCHECK のエンティティ メンバは、オブジェクト名の値に設定されます。eTrust AC のアクセス情報は、Unicenter 形式のアクセス許可に変換され、EMSECRESCHECK の属性メンバに割り当てられます。 EMSECSIGNON の他のすべてのメンバは、バイナリゼロに設定されます。 他のパラメータ、詳細なリターン コード、Unicenter Resource Check exit から返されたメッセージは無視されます。 Unicenter の統合のセットアップが完了すると、この機能が有効になります。

Unicenter カレンダ

Unicenter TNG にはカレンダ機能が提供されており、ユーザ、グループ、リソースに対して時間制限を設定できます。 カレンダは、15 分間隔で ON または OFF に設定できます。 カレンダの間隔を OFF に設定すると、リソースへのアクセスが拒否されます。 ON に設定すると、リソースへのアクセスが許可されます。

Windows では、セキュリティの起動前のみ、管理者がカレンダの使用を設定できます。

注: Unicenter TNG はローカル マシンにインストールする必要があります。 eTrust AC では、ローカル Unicenter TNG サービスを使用して、カレンダの設定を取得します。

1. eTrust AC セキュリティを停止します。 次のように入力します。

   ```
   secons -s
   ```

2. Windows レジストリで、以下のサブキーに移動します。

   ```
   HKEY_LOCAL_MACHINE\Software\ComputerAssociates\eTrustAccessControl\UCTNG
   ```

   このサブキーで、TNG_calendars 値を「yes」に設定します。 TNG_refresh_interval 値を適切な時間（分単位）に設定します。

3. eTrust AC セキュリティを開始します。 次のように入力します。

   ```
   seosd -start
   ```

eTrust AC リソースをカレンダとリンクするには、コマンド プロンプトから以下のデータベース コマンドを発行する必要があります。

```
eTrust> nr CALENDAR calendar_name
eTrust> nr file C:\myfile.txt calendar (calendar_name) defaccess (a)
```

Unicenter TNG カレンダ アクセス制御リスト（ACL）は、高度なセキュリティ制限機能です。 Unicenter TNG カレンダの標準プロパティでは、適切な Unicenter TNG カレンダ ステータスに基づいて現在のリソースが制限されます。 Unicenter TNG カレンダ ACL のプロパティにより、Unicenter TNG カレンダ ステータスに基づいて、現在のリソースに対する特定のユーザまたはグループのアクセス権が制限されます（またはアクセス権が与えられます）。

ACL Unicenter TNG カレンダ プロパティには、標準プロパティと制限プロパティの 2 種類があります。

- カレンダ ACL の標準プロパティでは、ACL アクセスに基づいて、ユーザまたはグループにリソースへのアクセスが許可されます。
- カレンダ ACL の制限（拒否）プロパティでは、ACL アクセスに基づいて、ユーザまたはグループはリソースへのアクセスが拒否されます。

標準のカレンダ ACL（CALACL）にユーザまたはグループを追加するには、次の selang のコマンドを入力します。

```
eTrust> auth resource_class_name object_name uid_or_gid_name calendar(calendar name)
access(access_value)
```

次に例を示します。

```
eTrust> auth file file1 uid(george) calendar(basecalendar) access(r w)
```

拒否のカレンダ ACL にユーザまたはグループを追加するには、次の selang のコマンドを入力します。

```
eTrust> auth resource_class_name object_name uid_or_gid_name
calendar(TNG_calendar_name) deniedaccess(access_value)
```

次に例を示します。

```
eTrust> auth file file2 uid(george) calendar(holidays) access(r w)
```

標準プロパティおよび制限プロパティを同じ(カレンダおよび UID など)リソースに対して使用できます。 次のコマンドにより、読み取りアクセス権を持つ George という名前のユーザを file1 の拒否カレンダ ACL に追加します。

```
eTrust> auth file file1 uid(george) calendar(holidays) deniedaccess(r)
```

Unicenter TNG カレンダ ACL プロパティからユーザまたはグループを削除するには、auth- を使用します。

```
eTrust> auth- file file2 uid(george) calendar(holidays)
```

特定のリソースに割り当てられたすべての Unicenter TNG カレンダ ACL を参照するには、Show Resource(sr)コマンドを使用します。

```
eTrust> sr file file1
```

検証済み Unicenter 統合機能

- 以下の機能は、Unicenter TNG 2.2 SP1、Unicenter TNG 2.4、または Unicenter NSM 3.0 に準拠しています。
  - 「イベント」の送信
  - メインフレーム パスワードの同期
  - Unicenter TNG カレンダの使用

付録 A: メインフレームとのパスワード同期

このセクションには、以下のトピックが含まれます。

パスワードの同期のサポート

eTrust AC では、eTrust CA-Top Secret Security、eTrust CA-ACF2 Security、または RACF セキュリティ製品を実行しているメインフレームと、eTrust AC を実行している Windows または UNIX マシンとのパスワード同期をサポートしています。同期は、eTrust AC の標準のパスワード Policy Model メカニズムによって実現します。

パスワード Policy Model 方式

ネットワークに存在するメインフレームとのパスワード同期を実装するには、eTrust AC を実行している Windows マシンをメインフレームの親として選択し、Mainframe Password Synchronization オプションがインストールされていることを確認します。 次に、eTrust AC に対してメインフレームを定義し、親となる Windows マシンからパスワード Policy Model にメインフレームをサブスクライブします。 このような設定を行うと、メインフレームのユーザがパスワードを変更するたびに、パスワード Policy Model 階層内のすべてのマシンにその変更が伝達されます。

メインフレーム管理者にパスワードを変更するための eTrust AC の権限を与えた場合、管理者がユーザ パスワードの変更、ユーザ アクションの一時停止または再開をメインフレームで実行すると、その情報がメインフレームからパスワード Policy Model 階層全体に伝達されます。 同様に、管理者がパスワード Policy Model 階層で実行したパスワード変更、ユーザ アクションの一時停止または再開も、メインフレームに伝達されます。

パスワード同期のインストール要件

メインフレーム側

コンピュータに Unicenter TNG 2.2 SP1、Unicenter TNG 2.4、Unicenter NSM 3.0、または CA Common Services がインストールされている必要があります。 パスワード同期ユーティリティは、Unicenter TNG に組み込まれている CAICCI（Common Communication Interface）に依存しています。

メインフレームにパスワード同期の環境設定を行う方法については、以下を参照してください。

- eTrust CA-ACF2 Security については、「eTrust CA-ACF2 Security 管理者ガイド」
- eTrust CA-Top Secret Security については、「eTrust CA-Top Secret Security ユーザ ガイド」
- RACF については、CA Common Services の製品版 CD

Windows 側

パスワード同期用にメインフレームの親として使用する各 Windows マシンに、eTrust AC を Mainframe Password Synchronization オプションと共にインストールする必要があります。

注: eTrust AC をすでにインストールしている場合は、インストール プログラムを再実行し、Mainframe Password Synchronization オプションを選択します。 再インストールを実行しても、現在のデータベースや設定が変更されることはありません。

インストールを始める前に、このマシンから Policy Model にサブスクライブする各メインフレームのホスト名、SYSID、および管理者名を調べておきます。 インストールの時点でこの情報が不明な場合は、この部分を省略し、後でメインフレームをサブスクライブしてもかまいません。

eTrust AC インストール プログラムを起動し、[カスタム インストール]を選択して、[Mainframe Password Synchronization]オプションをオンにします。 インストール ウィザードでは Unicenter CAICCI パッケージが使用されますが、Unicenter TNG を再起動して、CAICCI の環境設定を更新する必要があります。

インストールでは、Policy Model にホストをサブスクライブできます。 メインフレームのホスト名と SYSID がわかっている場合は、ここでサブスクライブできます。 インストールの時点でこの情報が不明な場合は、この手順を省略して、後でホストをサブスクライブすることができます。

インストールの確認

eTrust AC のインストール終了後に、次の手順に従い、必要なサービスおよびプロセスが正常にインストールされていることを確認します。

1. Windows の[サービス](Windows NT の場合は、[スタート]-[設定]-[コントロールパネル]-[サービス]、Windows 2000 の場合は、[スタート]-[設定]-[コントロールパネル]-[管理ツール]-[サービス])でサービスの一覧を表示します。

   サービスの一覧に、次のサービスが表示されていることを確認します。

   - Unicenter (NR-Server)
   - Unicenter (Remote)
   - Unicenter (Transport)

   eTrust AC Main Frame Sync

2. Windows のタスク マネージャを開いて[プロセス]タブを選択します。

   一覧に次のプロセスが表示されていることを確認します。

   - mfscpfd.exe
   - mfsd.exe
   - eacmfs.exe

インストール時に Policy Model にメインフレーム ホストをサブスクライブした場合は、以下の手順に従って、これらのホストがサブスクライバのリストに表示されることを確認します。

1. [スタート]メニューから[プログラム]-[CA]-[eTrust Access Control]-[ポリシー マネージャ]を選択します。
2. 左パネルの最下部にある[Tools]ボタンをクリックします。
3. [Policy Model]アイコンをクリックします。
4. ツリー ビューで、インストール時にメインフレームをサブスクライブした Policy Model を選択します。
5. サブスクライブしたメインフレーム ホストが右側のリストに表示されることを確認してください。

Policy Model の環境設定

パスワード同期に必要な環境設定を完了するには、メインフレームおよび親である Windows システムに適切なソフトウェアをインストールした後、Windows システムで以下の手順を実行する必要があります。

- Policy Model にサブスクライブする各メインフレーム ホストの PMDB に MFTERMINAL レコードを作成します。

  ローカル データベースではなく、PMDB にレコードを作成すると、このレコードが Policy Model 階層内のすべてのホストに伝達されます。

- パスワード変更コマンドを発行する各メインフレーム管理者の USER レコードを PMDB およびネイティブ Windows 環境に作成します。これらのユーザがパスワードの変更権限を持っていることを eTrust AC が認識できるように、ユーザに管理者権限またはパスワード管理者権限を与えます。

- 同様に PMDB では、Windows マシンの TERMINAL レコードに対するフル アクセス権（読み取り/書き込み）、およびパスワード変更コマンドを発行する任意のメインフレームの MFTERMINAL レコードに対する読み取りアクセス権をメインフレーム管理者に与えます。

- これらのメインフレーム管理者に、ネイティブ Windows 環境でのローカル ログオン権限を与えます。 メインフレームから伝達されたパスワード変更は、ローカル マシン上の eTrust AC で、適切なメインフレーム管理者であるユーザの権限によって実行できる必要があります。

- メインフレームを Policy Model にサブスクライブします（インストール時にサブスクライブしなかった場合）。

- Windows レジストリの passwd_pmd キーをチェックして、ローカル マシンおよび、そのサブスクライバから受け取ったパスワード変更の同期処理を行うパスワード Policy Model が指定されていることを確認します。 必要に応じて、レジストリのエントリを更新します。

上記の手順は、特定の順序で実行する必要はありません（ただし、当然ながら、管理者のユーザ レコードおよびメインフレームの MFTERMINAL レコードの両方を作成してからでなければ、メインフレーム管理者に MFTERMINAL レコードに対するアクセス許可を与えることはできません）。

これらの手順を実行する方法の 1 つを次に示します。

1. [スタート]メニューから[プログラム]-[CA]-[eTrust Access Control]-[ポリシー マネージャ]を選択します。
2. インストール時に、メインフレームを Policy Model にサブスクライブしなかった場合は、次の手順を実行してください。 サブスクライブした場合は、手順 3 に進みます。

   a. 左側のプログラム バーにある[Tools]ボタンをクリックします。

   b. [Policy Model]アイコンをクリックします。

c. 適切な Policy Model を選択します。

d. [Edit]メニューから[Add Subscriber]を選択します。

e. [Subscriber Name]に、メインフレームの完全修飾ホスト名を入力します。

f. [Mainframe Subscriber]チェック ボックスをオンにします。

g. このメインフレームのホスト タイプ(ACF、ACF2、RACF、TNG、または TSS)を選択し、メインフレームの SYSID および管理者名を入力します。

h. [OK]をクリックします。

i. 追加するメインフレーム サブスクライバごとに、この手順を繰り返します。

3. 次の手順に従って、passwd_pmd キーを確認します。

a. 左側のプログラム バーにある[Windows NT]ボタンをクリックします。

b. [Registry Editor]アイコンをクリックします。

c. ツリー ビューから、次の場所に移動します。

```
HKEY_LOCAL_MACHINE\Software\ComputerAssociates\eTrustAccessControl\eTrustAccessControl
```

d. 右側のリストにある passwd_pmd キーをダブルクリックします。

e. [値]領域で、[文字列]ボタンをクリックします。パスワード Policy Model 階層にローカル マシンの親の完全修飾名がまだ正しく指定されていない場合は入力します。

f. [OK]をクリックします。

4. 以下の手順に従って、メインフレーム管理者の USER レコードを作成し、メインフレーム管理者に eTrust AC でパスワードを変更する権限を与え、さらに Windows でローカル ユーザとしてログオンする権限を与えます。

a. [File]-[Connect]を選択するか、ツール バーの[Connect]ボタンをクリックします。

b. [ホストの選択]ダイアログ ボックスで、[localhost pmdb]を選択するか、または[ホスト]テキスト ボックスに「pmdbName@localhost」と入力します。pmdbName には、該当する Policy Model を指定します。

c. [OK]をクリックします。

d. 左側のプログラム バーにある[Access Control]ボタンをクリックします。

e. [Users]アイコンをクリックします。

f. ツールバーの[New]ボタンをクリックします。

g. [新しいユーザの作成 - 全般]ダイアログ ボックスで、メインフレーム管理者の名前を入力します。

h. [User Attributes]アイコンをクリックします。

i. [User Attributes]ダイアログ ボックスでは、このユーザに与える権限に応じて、[Administrator]または[Password Manager]チェック ボックスをオンにします。

j. [Miscellaneous]アイコンをクリックします。

k. [Miscellaneous]ダイアログ ボックスの[User Privileges]ボタンをクリックします。

l. 指定可能なユーザの権限のリストから、[ローカルでのログオン]を選択して[>>]ボタンをクリックし、このユーザの権限を[付与された権限]リストに移動します。

m. [OK]をクリックして[User Privileges]ダイアログ ボックスを閉じます。

n. [OK]をクリックして、このユーザを追加します。

o. メインフレーム管理者ごとに、手順 e から n を繰り返します。

p. [Resources]アイコンをクリックします。

q. ツリー ビューから、[Logon Protection]-[Terminal]を選択します。

r. TERMINAL レコードのリストから、ローカル マシンのレコードを選択します。

s. [View or set TERMINAL properties]ダイアログ ボックスで、左側にある[Authorize]アイコンをクリックします。

t. [Add Accessors]の右にある[Insert]ボタンをクリックします。

u. アクセサ リストを参照して、リストからメインフレーム管理者を選択します。

v. [OK]をクリックします。

w. [Permissions]領域の[All]ボタンをクリックします。

x. メインフレーム管理者ごとに、手順 e から i を繰り返します。

y. [OK]をクリックします。

5. ポリシー マネージャを終了して、selang を起動します。

6. 次のように入力して、Policy Model に接続します。

```
eTrust> host pmd@localhost
```

7. 次のコマンドを使用して、メインフレームごとに MFTERMINAL レコードを作成します。

```
eTrust> newres MFTERMINAL mfSYSID defaccess (none) owner (userName)
```

mfSYSID にはメインフレームの SYSID を、userName にはこの MFTERMINAL レコードを所有するユーザの名前を指定します。

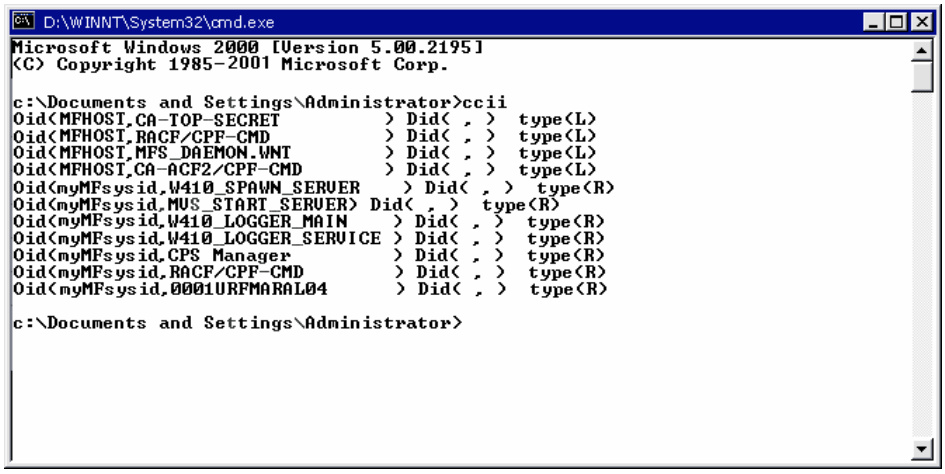
8. 次のコマンドを使用して、各メインフレーム管理者に適切な MFTERMINAL レコードへのアクセス権を与えます。

```
eTrust> authorize MFTERMINAL mfSYSID uid (mfAdmin) access (read)
```

mfSYSID にはメインフレームの SYSID を、mfAdmin にはメインフレーム管理者であるユーザを指定します。

メインフレーム同期の開始

接続が確立されたことを確認するには、コマンド プロンプトから ccii ユーティリティを実行します。



CAICCI 環境設定ファイル

eTrust AC では、インストール時、および Policy Model にメインフレームをサブスクライブするたびに、CAICCI 環境設定ファイルが自動的に更新されます。

何らかの理由で環境設定ファイルを手動で更新する必要がある場合は、以下の手順に従います。

1. メモ帳で CAICCI 環境設定ファイル（cciDirectory¥tng¥caiuser¥ccirmtd.rc）を開きます。
2. 次の行を追加します。

   ```
   REMOTE = mfName mfSYSID 1024 startup port 1721
   ```

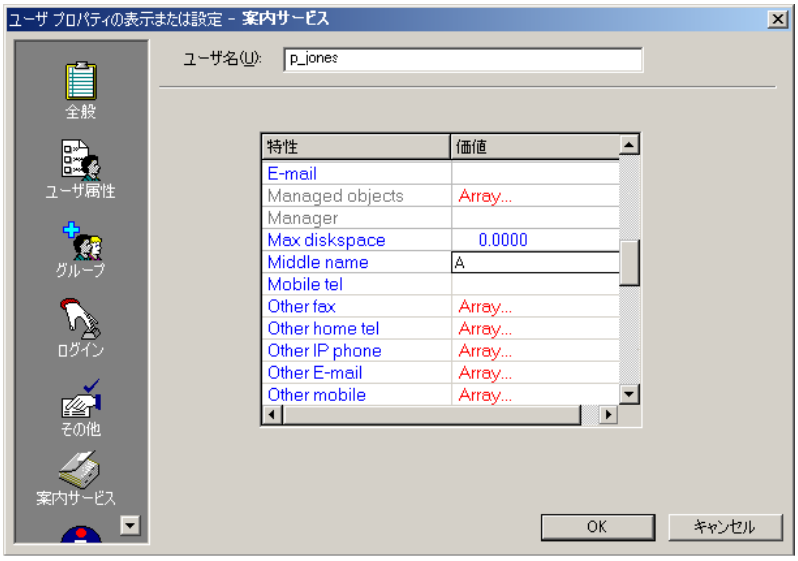
   mfName にはメインフレーム名、mfSYSID にはメインフレームの SYSID を指定します。
3. ファイルを保存します。
4. 次のコマンドを実行して、リモート CAICCI サービスを停止します。

   ```
   ccicntrl stop rmt
   ```
5. リモート CAICCI サービスを再起動します。

   ```
   ccicntrl start rmt
   ```

Active Directory のユーザまたはグループのプロパティの設定

Active Directory を使用している Windows 2000 マシンに接続している場合は、[ユーザのプロパティ]または[グループのプロパティ]ダイアログ ボックスの[Active Directory]パネルを使用して、Active Directory のユーザ プロパティまたはグループのプロパティを設定できます。 これらのプロパティは、ネイティブ Windows NT や Active Directory がインストールされていない Windows 2000 マシン、または eTrust AC データベース環境では、サポートされていません。



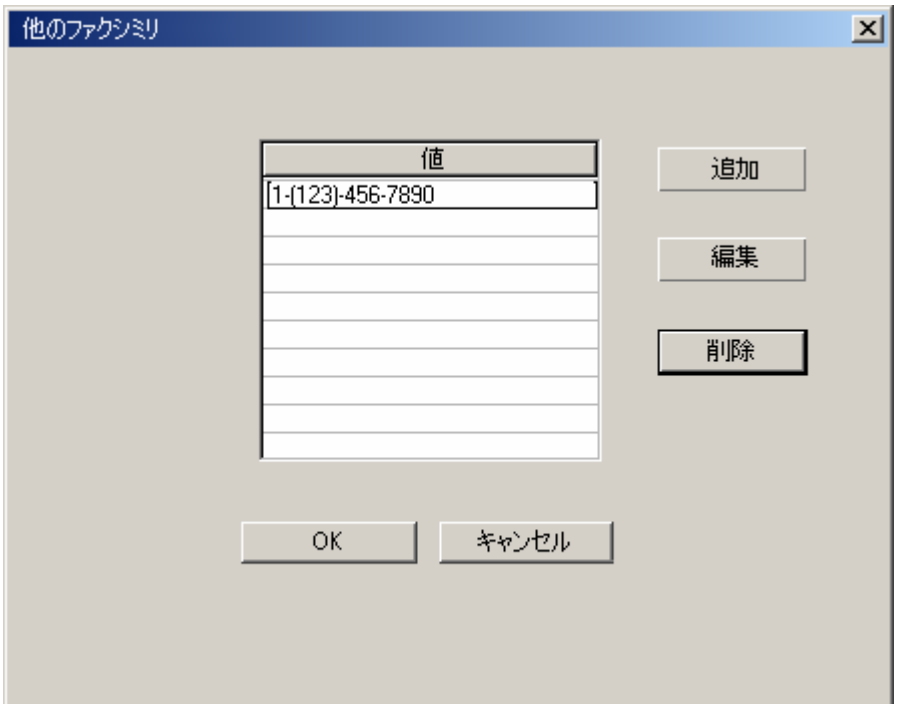
使用しているコンピュータに Active Directory がインストールされていない場合は、Active Directory がインストールされているコンピュータに接続します（そのコンピュータに eTrust AC をインストールしておく必要があります）。

1. 新しいユーザを作成し、[Active Directory]パネルを開きます。

   注: Active Directory をサポートする Windows 2000 マシンに接続していない場合、このパネルをアクティブにするアイコンは表示されません。

2. リストをスクロールします。 エントリを作成する場合は、テーブルの[Value]（右）の側でダブルクリックします。 セルの周囲に表示された枠の中にエントリを入力します。

   注: [配列]という文字をダブルクリックすると、入力するテーブルが表示されます。



索引

T

U

W

あ

か

さ

た

な

は

ま

や

ら