



Postgres Enterprise Manager

リリース 7.10

PEM Getting Started Guide

2019 年 10 月 20 日

目次

第 1 章	PEM Overview	2
1.1	Postgres Enterprise Manager を使用する理由	3
1.2	一般的なアーキテクチャ	5
1.3	サポートされているバージョンとプラットフォーム	7
1.4	PEM のインストール-概要	9
1.5	PEM Web インターフェイスの使用	10
1.5.1	PEM ツールバー	12
1.5.2	チャート、グラフ、テーブルの制御とカスタマイズ	22
1.5.3	オンラインヘルプとドキュメント	23
第 2 章	サーバーの登録	24
2.1	サーバーの手動登録	24
2.2	自動サーバー検出	35
2.3	pemworker ユーティリティを使用してサーバーを登録する	38
2.3.1	pemworker ユーティリティを使用してサーバーの登録を解除する	41
2.4	接続とバインドの検証	42
第 3 章	証明書の管理	43
3.1	SSL 証明書の置き換え	44
3.2	エージェント SSL 証明書の更新	48
第 4 章	PEM サーバーの管理	50
4.1	PEM サーバーとエージェントの起動と停止	50
4.2	監視対象サーバーのリモート起動と停止	52

4.3	Linux での PEM サーバーまたは PEM エージェントの制御	53
4.4	Windows での PEM サーバーまたは PEM エージェントの制御	54
4.5	HTTPD サーバーの制御	55
4.6	pg_hba.conf ファイルの変更	56
4.7	データベースとオブジェクトの作成と保守	58
4.8	PEM 認証の管理	60
4.9	プロキシサーバーを使用するための PEM の変更	62
4.10	PEM サーバー構成の編集	64
4.11	セキュリティの管理	65
4.11.1	ログインロール	65
4.11.2	グループの役割	67
4.11.3	PEM の事前定義されたロールを使用して PEM 機能へのアクセスを管理する	68
4.11.4	チームロールを使用する	71
4.11.5	オブジェクト権限	72
4.12	ジョブ通知の管理	73
4.12.1	ジョブレベルでのジョブ通知の構成	73
4.12.2	エージェントレベルでのジョブ通知の構成	74
4.12.3	サーバーレベルでのジョブ通知の構成	76
4.13	PEM スケジュール済みジョブの管理	77
第 5 章	PEM エージェントの管理	86
5.1	エージェント権限	86
5.2	エージェント構成	89
5.3	エージェントのプロパティ	95
第 6 章	結論	97
	索引	99

このドキュメントでは、Postgres Enterprise Manager™ (PEM) の概要を説明します。このガイドでは、ツールセットの基本を理解し、データベース管理活動を成功させるのに役立ちます。このガイドは、次のコアセクションとカテゴリに分かれています。

- **Postgres Enterprise Manager の概要**-このセクションでは、PEM 機能、コンポーネント、アーキテクチャ、およびサポートされているプラットフォームに関する情報を提供します。このセクションでは、PEM の Web インターフェイスの概要も説明します。Web インターフェイスは PEM サーバーとともにインストールされ、選択したブラウザから使用できます。
- **サーバーの登録**-このセクションでは、サーバーの登録に役立つさまざまなツールに関する情報を提供します。
- **証明書の管理**-このセクションでは、PEM の証明書の管理に関する情報を提供します。
- **PEM サーバーの管理**-このセクションでは、PEM サーバーとエージェントの再起動、PEM サーバーまたは PEM エージェントの制御、Linux および Windows での HTTPD サービスの制御、HTTPD サーバーの制御など、PEM サーバーに関連するタスクに関する情報を提供します。PEM 認証とセキュリティの管理、pg_hba.conf ファイルの変更、プロキシサーバーを使用するための PEM の変更など。
- **PEM エージェントの管理**-このセクションでは、PEM エージェントの構成と管理に関する情報を提供します。

このドキュメントでは、PostgreSQL または EDB Postgres Advanced Server データベースを意味する* Postgres *を使用しています。

PEM Overview

Postgres Enterprise Manager (PEM) は、PostgreSQL および EnterpriseDB Advanced Server データベースサーバーの管理、監視、およびチューニングにおいて、データベース管理者、システム設計者、およびパフォーマンスアナリストを支援するために設計されたエンタープライズ管理ツールです。 PEM は、一握りから数百台のサーバーまで、単一のコンソールからどこでも管理および監視できるように設計されており、データベースのあらゆる側面を完全かつリモートで制御できます。

1.1 Postgres Enterprise Manager を使用する理由

PEM には、他の PostgreSQL 管理ツールにはない多くの利点があります。

- **大規模デザインの管理。** PEM はエンタープライズデータベース管理用に設計されており、地理的な境界を越えて多数のサーバーの管理に取り組むために構築されています。グローバルダッシュボードにより、すべてのサーバーのアップ/ダウン/パフォーマンスステータスを一目で把握できます。
- **分散アーキテクチャ。** PEM は、オペレーティングシステムプラットフォームに関係なく、統計情報を収集し、マシン上でリモートで操作を実行する能力を最大化するように設計されています。
- **グラフィカル管理。** データベース管理のすべての側面は、グラフィカルインターフェイスを介して PEM クライアントで実行できます。サーバーの起動とシャットダウン、構成管理、ストレージとセキュリティの制御、オブジェクトの作成、パフォーマンス管理などを単一のコンソールから処理できます。
- **フル SQL IDE。** PEM には、アドホック SQL クエリ、ストアドプロシージャ/関数開発、およびグラフィカルデバッガを提供する堅牢な SQL 統合開発環境 (IDE) が含まれています。
- **エンタープライズパフォーマンスモニタリング。** PEM は、すべての管理対象データベースサーバーのエンタープライズクラスのパフォーマンス監視を提供します。軽量で効率的なエージェントは、各データベースサーバーの操作のすべての側面と各マシンの基盤となるオペレーティングシステムを監視し、インターフェイス内の簡単にナビゲートされたパフォーマンスページに詳細な統計を返します。
- **プロアクティブなアラート管理。** PEM には、24 時間監視される各主要メトリック（メモリ、ストレージなど）のパフォーマンスしきい値を作成する機能が付属しています。しきい値違反が発生すると、アラートが集中ダッシュボードに送信され、問題の性質と、その状況がサーバーの全体的なパフォーマンスを危険にさらさないようにするために必要なアクションを通知します。
- **簡素化された容量計画。** すべての主要なパフォーマンス関連の統計が自動的に収集され、PEM のリポジトリに指定された期間保持されます。Capacity Manager ユーティリティを使用すると、さまざまな統計を選択し、時間の経過に伴う傾向分析を実行して、ピーク負荷期間、ストレージ消費傾向などを把握できます。ツールの予測メカニズムにより、将来のリソース使用量を予測し、それに応じて計画/予算を立てることができます。
- ****監査マネージャー。** **監査マネージャーは、Advanced Server インスタンスの監査ログを構成します。データベースへの接続、データベースからの切断、データベースに対して実行される

SQL ステートメントなどのアクティビティをログに記録できます。その後、監査ログダッシュボードを使用して、ログをフィルタリングおよび表示できます。

- ****ログマネージャー。** ****ログマネージャーウィザード**は、サーバーログパラメーターを構成し、（オプションの）中央テーブルへのログ収集を行います。ウィザードを使用して、ログファイルのローテーション、ログの宛先、エラーメッセージの重大度などのロギング動作の設定を指定します。サーバーログダッシュボードを使用して、収集されたサーバーログエントリをフィルタリングおよび確認します。
- **SQL ワークロードプロファイリング。** PEM には、1 つ以上のサーバーに対して実行される SQL ステートメントをトレースできる SQL プロファイリングユーティリティが含まれています。SQL プロファイリングは、アドホックまたはスケジュールされた方法で実行できます。キャプチャーされた SQL ステートメントはフィルターに掛けられるため、実行が不十分な SQL ステートメントを簡単に識別して調整できます。SQL ステートメントを Advanced Server の Index Advisor に入力して、各ステートメントを分析し、パフォーマンスを向上させるために作成する必要がある新しいインデックスに関する推奨事項を作成することもできます。
- **エキスパートデータベース分析。** PEM には Postgres Expert ユーティリティが含まれています。Postgres Expert は、ベストプラクティスの実施目的で選択されたデータベースを分析します。一般的な構成、セキュリティのセットアップなどの領域が調べられます。推奨されるベストプラクティスからの逸脱は、特定の各問題の説明と、物事を正しくするために何をすべきかについての専門家のヘルプとともに報告されます。
- ****ストリーミングレプリケーションの構成と監視。** ****ストリーミングレプリケーションウィザード**は、Postgres ストリーミングレプリケーションシナリオに新しいサーバーを追加したり、既存のサーバーを構成してレプリケーションシナリオを作成したりするプロセスを簡素化します。レプリケーションシナリオを構成したら、ストリーミングレプリケーションダッシュボードでシナリオを監視するか、PEM クライアントのオプションを使用して、スタンバイノードをマスターノードに昇格できます。
- ****セキュアクライアント接続。** ****PEM** は、暗号化された SSH トンネルを介したセキュアクライアント接続をサポートします。フル機能の PEM クライアントには、安全な接続のための接続情報を提供できる SSH トンネル定義ダイアログが含まれています。
- **幅広いプラットフォームのサポート。** PEM は、ほとんどの主要な Linux および Windows プラットフォームをサポートしています。

1.2 一般的なアーキテクチャ

PEM は、3 つの主要なコンポーネントで構成されています。

PEM サーバー

PEM サーバーは、Postgres Enterprise Manager のコアで機能を提供します。サーバーは以下を担当します。

- 管理機能を実行します。
- エージェントから受け取った情報を処理します。
- リポジトリ内の情報を維持します。

PEM エージェント

PEM エージェントは、各管理対象マシンでタスクを実行し、データベースサーバーとオペレーティングシステムの統計を収集します。

PEM Web インターフェイス

PEM サーバーと一緒に配布される PEM Web インターフェイスにより、選択したブラウザでサーバーに接続し、Postgres サーバーを管理および監視できます。

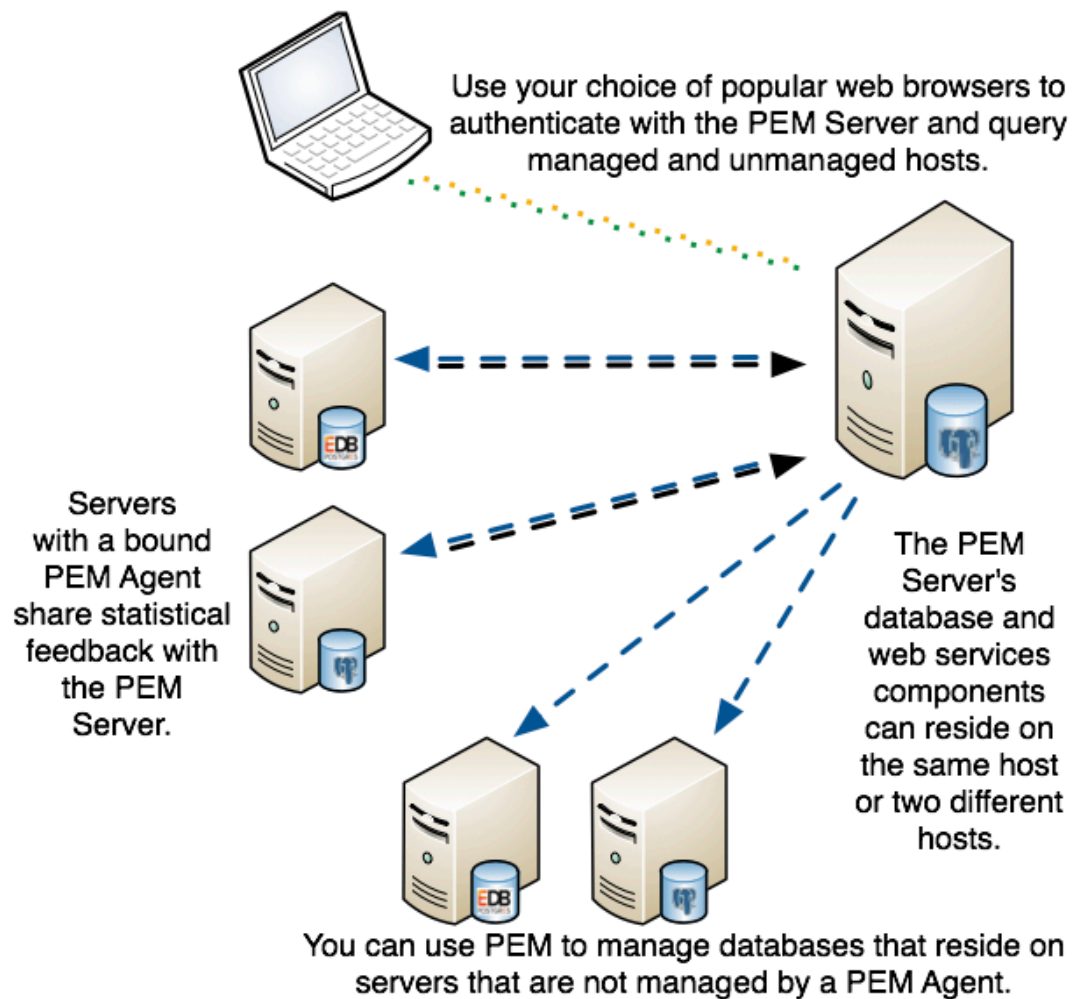


図 1 The Postgres Enterprise Manager general architecture

1.3 サポートされているバージョンとプラットフォーム

PEM サーバーは以下でサポートされています。

PEM サーバーとエージェント

64 ビット Windows:

- Windows Server 2012 R2
- Windows Server 2016
- Windows Server 2019

64 ビット Linux:

- SLES 12.x
- CentOS 6.x および 7.x
- OEL 6.x および 7.x
- Red Hat Enterprise Linux 6.x および 7.x
- Debian 9.x (ストレッチ)
- Ubuntu 18.04+ (バイオニック)

Where *x* is the latest minor release of the operating system.

注釈: オペレーティングシステムは、ベンダーからの最新の更新で継続的に最新の状態に保つことをお勧めします。そうしないと、互換性またはセキュリティの問題が発生する可能性があります。

PEM サーバーは、Advanced Server または PostgreSQL バックアップデータベース (バージョン 9.4 以降) を使用して、これらのプラットフォームでサポートされています。

PEM エージェントは、Advanced Server または PostgreSQL バージョン 9.4 以降がサポートされている Linux または Windows プラットフォームでサポートされています。

ブラウザ

PEM Web インターフェイスは、次のブラウザでサポートされています。

- Google Chrome バージョン 65 以降

- Mozilla Firefox バージョン 59 以降
- Microsoft Edge バージョン 42 以降
- Apple Safari バージョン 11.1 以降

PostgreSQL バージョンのサポート

PEM は以下を管理および監視できます。

- PostgreSQL バージョン 9.4 以降
- Advanced Server 9.4 以降

SQL プロファイラー

Postgres Enterprise Manager の SQL プロファイラは、EDB Postgres Advanced Server 9.4 以降、および PostgreSQL バージョン 9.4 以降の EnterpriseDB ディストリビューションでサポートされています。

1.4 PEM のインストール-概要

PEM のインストールに関する詳細な手順については、次の場所にある* PEM インストールガイド*を参照してください。

<https://www.enterprisedb.com/edb-docs/p/edb-postgres-enterprise-manager>

PEM のインストールプロセスに含まれる基本的な手順は次のとおりです。

1. PEM サーバーコンポーネントをインストールします。 PEM サーバーソフトウェアとバックエンドデータベース（名前は pem）は、サポートする httpd サーバーと同じホストに存在する場合と、別のホストに存在する場合があります。

PEM サーバーインストーラーは、PEM サーバーのホストに PEM エージェントと PEM クライアントをインストールします。

2. PEM で管理する追加の各物理マシンまたは仮想マシンを登録します。便宜上、PEM はリモート監視をサポートしています（エージェントは、監視するサーバーと同じホストに存在する必要はありません）。

リモートエージェントは、監視対象サーバーから利用可能なすべての情報を取得できないことに注意してください。各サーバーホストにエージェントをインストールすることもできます。

3. SQL プロファイラコンポーネントを、SQL キャプチャおよび分析を実行する各 Postgres インスタンスにインストールします。 SQL Profiler インストーラーは、Postgres のインストール場所の入力を求め、必要なソフトウェアをそのディレクトリに配置します。 SQL Profiler プラグインは Advanced Server インスタンスに既にインストールされており、プロファイリングを有効にするための設定のみが必要です。

1.5 PEM Web インターフェイスの使用

PEM Web インターフェイスは、PEM サーバーとともにインストールされます。サーバーのインストールが完了したら、次の場所へ移動して、選択したブラウザで PEM インターフェイスを開くことができます。

```
https://<ip_address_of_PEM_host>:8443/pem
```

ここで、`ip_address_of_PEM_host` は PEM サーバーのホストの IP アドレスを指定します。Postgres Enterprise Manager ログインウィンドウが開きます。

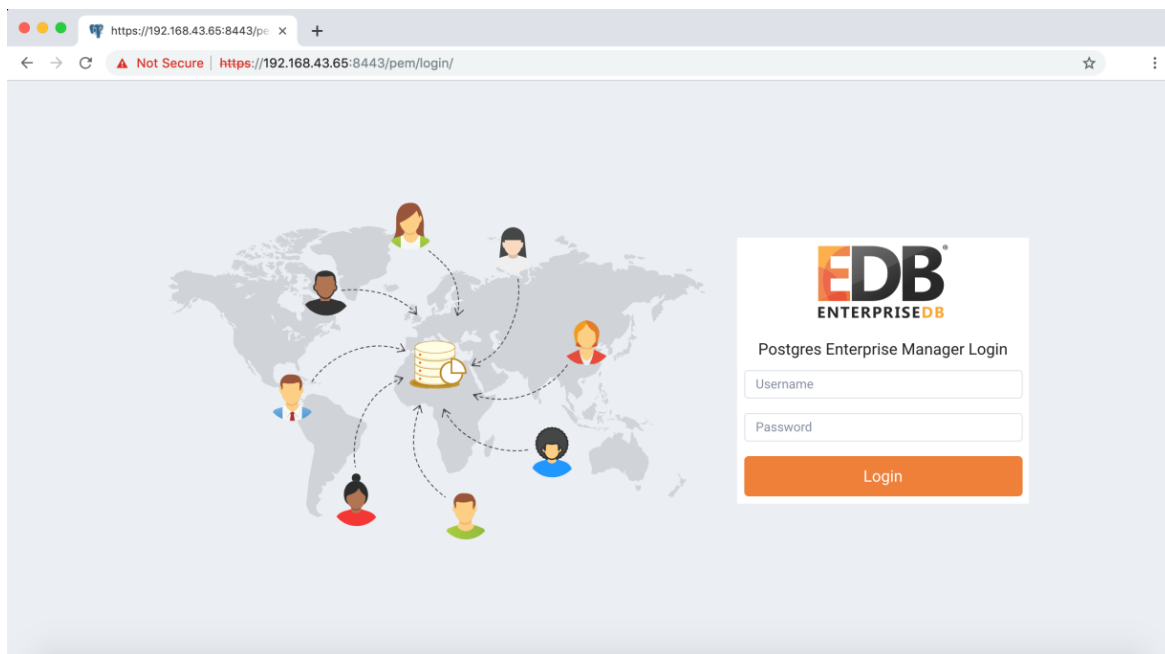


図 2 The PEM Login page

Postgres Enterprise Manager Login ウィンドウのフィールドを使用して、PEM サーバーで自分自身を認証します。

- Username フィールドに `pem` データベースユーザーの名前を入力します。最初に接続するユーザーの場合、これは PEM サーバーのインストール時に指定された名前になります。
- ユーザーに関連付けられたパスワードを Password フィールドに入力します。

資格情報を入力したら、[ログイン] をクリックして PEM に接続します。

PEM Web インターフェイスが開き、Global Overview ダッシュボードが表示されます。

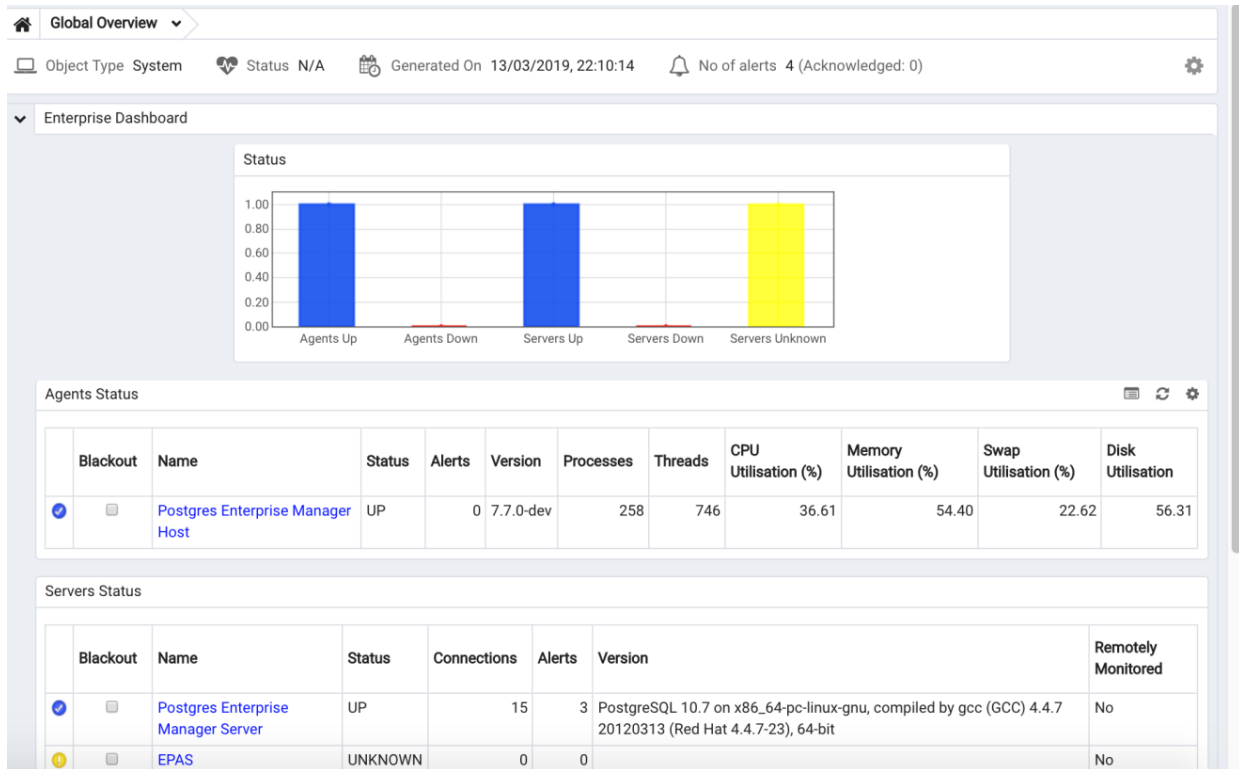


図 3 クライアントに表示されるグローバル概要ダッシュボード

Browser ペインには、各サーバーにあるデータベースオブジェクトに関する情報へのアクセスを提供するツリーコントロールが表示されます。ツリーコントロールが展開され、PEM サーバーによって監視されているサーバーとオブジェクトの階層ビューが表示されます。

PEM メニューバーを使用すると、データベースサーバーとそれらのサーバーに存在するオブジェクトを管理するために使用できるコマンドと機能にアクセスできます。オプションが無効になっている場合:

- 現在接続しているデータベースサーバーは、選択した機能をサポートしていない可能性があります。
- 選択されたメニューオプションは、現在のオブジェクトに対して（設計上）有効でない場合があります。
- サーバーへの接続に使用した役割には、選択したオブジェクトを変更するための十分な権限がない可能性があります。

1.5.1 PEM ツールバー

PEM Web インターフェイスの上部にあるコンテキスト依存メニューにより、環境をカスタマイズし、PEM のエンタープライズ管理機能へのアクセスを提供できます。

ファイルメニュー

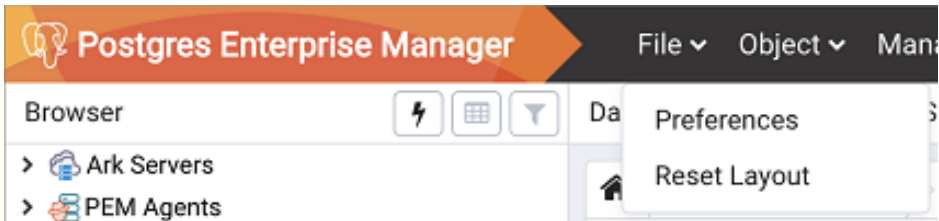


図 4 ファイルメニュー

File メニューを使用して、次のオプションにアクセスします:

メニューオプション	アクション
プリファレンス	クリックして [設定] ダイアログを開き、PEM クライアント設定をカスタマイズします。
レイアウトをリセット	ワークスペースを変更した場合、クリックしてデフォルトのレイアウトに戻します。

オブジェクトメニュー

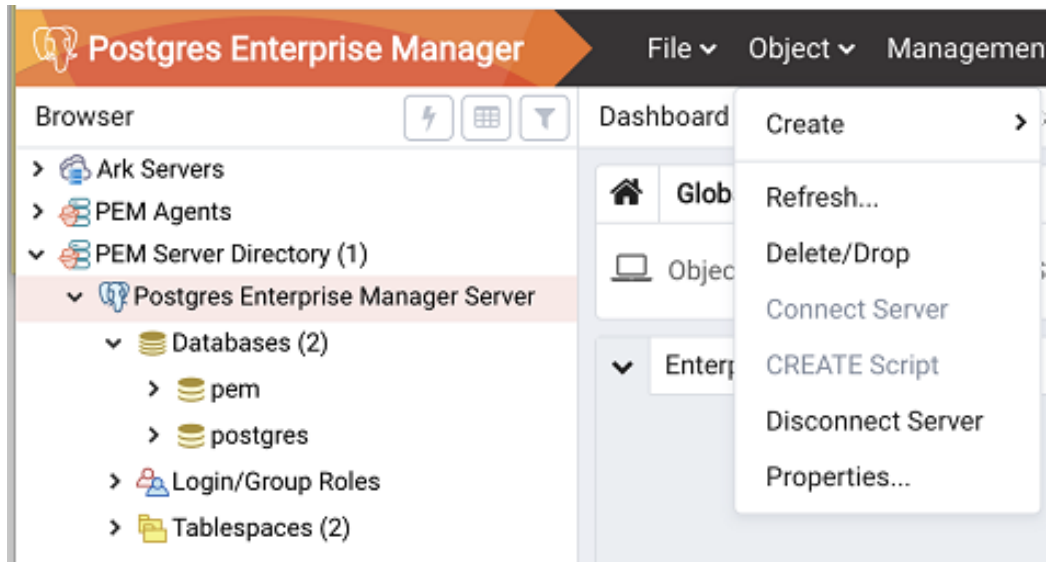


図 5 オブジェクトメニュー

Object メニューは状況依存です。Object メニューを使用して、次のオプションにアクセスします:

メニューオプション	アクション
サーバーを接続...	[サーバーに接続] ダイアログをクリックして開き、サーバーとの接続を確立します。
作成する	/ * Create / * をクリックして、コンテキスト依存の選択を提供するコンテキストメニューにアクセスします。
CREATE スクリプト	クリックしてクエリツールを開き、選択したスクリプトを編集または表示します。
削除/ドロップ	クリックすると、現在選択されているオブジェクトがサーバーから削除されます。
サーバーを切断...	クリックすると、現在選択されているオブジェクトが更新されます。
ドロップカスケード	クリックすると、現在選択されているオブジェクトとすべての依存オブジェクトがサーバーから削除されます。
隠す...	クリックすると、現在選択されているグループが非表示になります。非表示のグループを表示するには、環境設定の“非表示のグループを表示”オプションを有効にします。
プロパティ...	クリックして、現在選択されているオブジェクトのプロパティを確認または変更します。
更新...	クリックすると、現在選択されているオブジェクトが更新されます。
トリガー	現在選択されているテーブルの / * Disable / * または / * Enable / * トリガーをクリックします。
切り捨て	クリックすると、テーブルからすべての行が削除されます（切り捨て）。または、テーブルとその子テーブルからすべての行が削除されます（カスケードの切り捨て）。
データを見る	クリックして、データを表示するためのいくつかのオプションを提供するコンテキストメニューにアクセスします。

管理メニュー

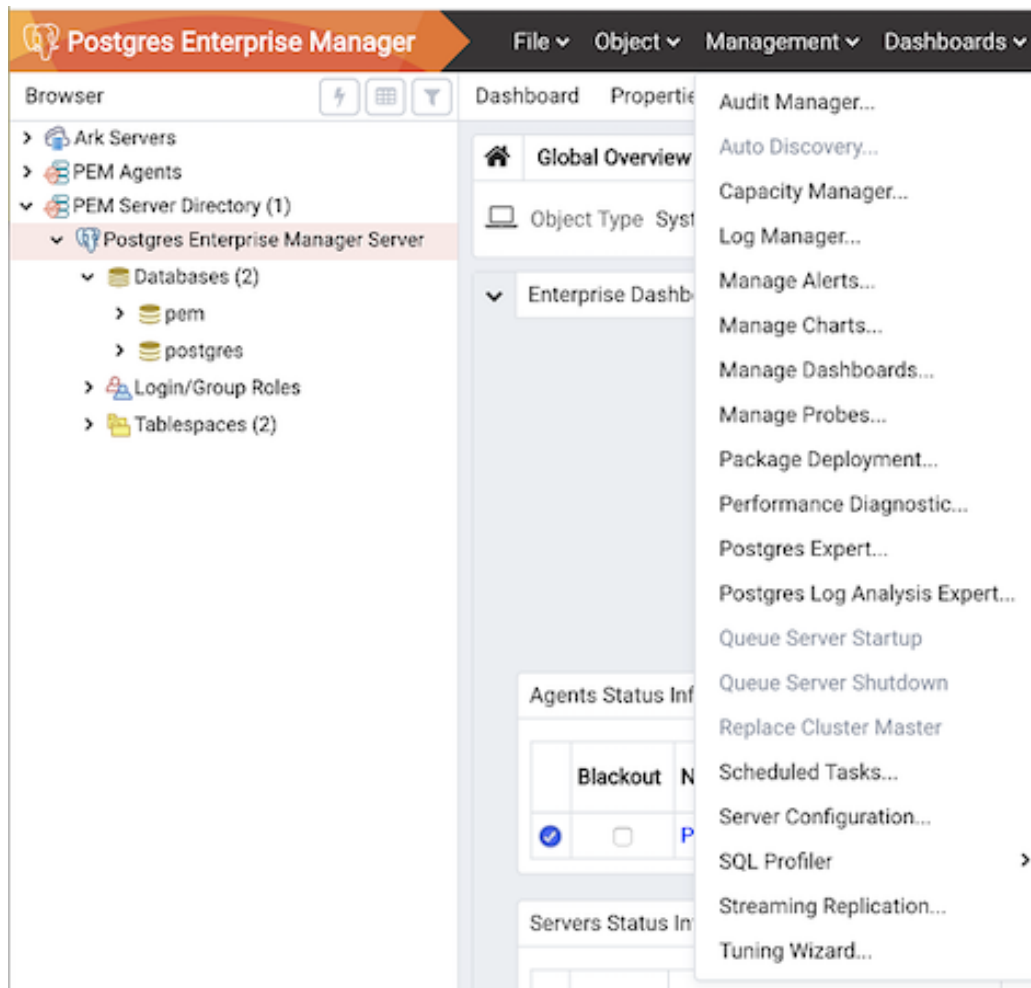


図 6 管理メニュー

Management メニューを使用して、次の PEM 機能にアクセスします。

メニューオプション	アクション
監査マネージャー...	クリックして監査マネージャーを開き、監視対象サーバーで監査を構成します。
自動検出...	クリックして [自動検出] ダイアログを開き、監視対象のデータベースサーバーを見つけてバインドするように PEM エージェントに指示します。
キャパシティマネージャ...	クリックして Capacity Manager ダイアログを開き、履歴またはプロジェクトの将来のリソース使用量を分析します。
ログマネージャー...	クリックして [ログマネージャー] ダイアログを開き、サーバーのログ収集を構成します。
アラートを管理...	クリックして [アラートの管理] タブにアクセスし、アラート動作を作成または変更します。
チャートを管理...	クリックして [チャートの管理] タブを開き、PEM チャートを作成または変更します。
ダッシュボードを管理...	クリックして、[ダッシュボードの管理] ダイアログを開き、VACUUM、ANALYZE、REINDEX、または CLUSTER を選択します。
プローブの管理...	クリックして VACUUM、ANALYZE、REINDEX、または CLUSTER の Manage Probes ダイアログを開きます。
パッケージの展開...	クリックしてパッケージ展開ウィザードを開き、パッケージをインストールまたは更新します。
Postgres エキスパート...	クリックして Postgres Expert ウィザードを開き、サーバーとデータベースの静的分析を実行します。
Postgres ログ分析エキスパート...	クリックして Postgres Log Analysis Expert ダイアログにアクセスし、使用傾向のログファイルの内容を分析します。
キューサーバーの起動...	クリックすると、現在選択されているサーバーを起動するように PEM サーバーに指示します。
キューサーバーのシャットダウン	クリックして、現在選択されているサーバーをシャットダウンするように PEM サーバーに指示します。
クラスターマスターの交換	クリックして、レプリケーションシナリオの現在のマスターノードを置き換えます。
スケジュールされたタスク	[スケジュールされたタスク] タブをクリックして開き、保留中または最近完了したタスクを確認します。
サーバー構成...	[サーバー構成] ダイアログをクリックして開き、PEM サーバーの構成設定を更新します。
SQL プロファイラ...	クリックして SQL プロファイラダイアログを開き、SQL クエリパフォーマンスに関する統計情報をキャプチャします。
ストリーミングレプリケーション...	[ストリーミングレプリケーション] ダイアログにアクセスして、ストリーミングレプリケーションシナリオを構成します。
チューニングウィザード...	クリックして [チューニングウィザード] ダイアログを開き、サーバーの一連のチューニング推奨事項を生成します。

ダッシュボードメニュー

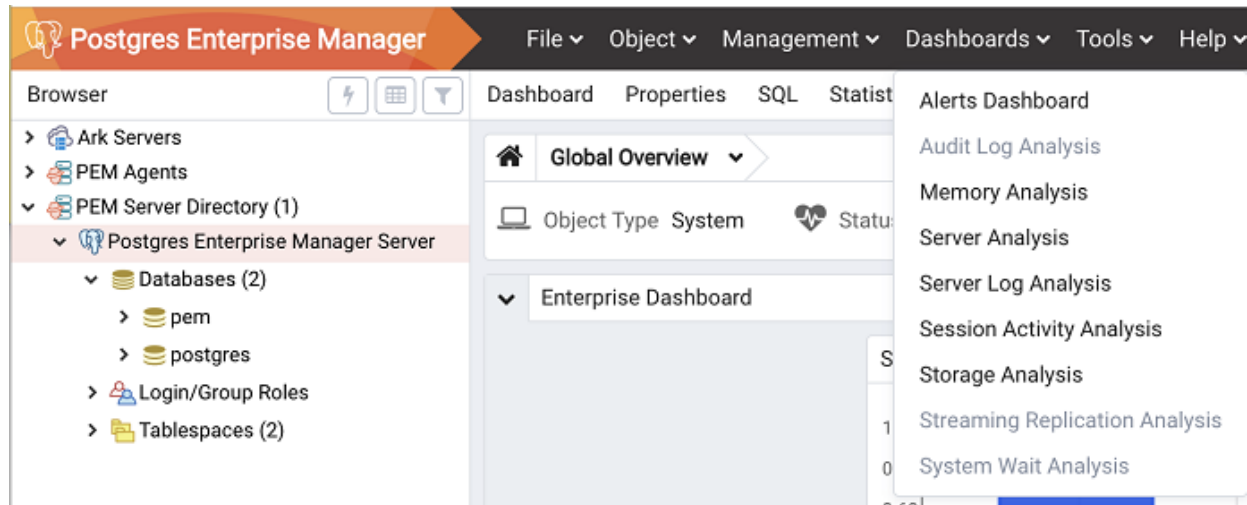


図 7 ダッシュボードメニュー

状況依存の Dashboards メニューを使用して、ダッシュボードにアクセスします。

オプション	アクション
アラートダッシュボード	クリックして、選択したノードのアラートダッシュボードを開きます。
監査ログ分析	クリックして、選択したノードの監査ログ分析ダッシュボードを開きます
データベース分析	クリックして、選択したノードのデータベース分析ダッシュボードを開きます。
I/O 分析	クリックして、選択したノードの I/O 分析ダッシュボードを開きます。
メモリ分析	クリックして、選択したノードのメモリ分析ダッシュボードを開きます
オブジェクトアクティビティ分析	クリックして、選択したノードのオブジェクトアクティビティ分析ダッシュボードを開きます。
オペレーティングシステム分析	クリックして、選択したノードのオペレーティングシステム分析ダッシュボードを開きます。
プローブログ分析	クリックして、選択したノードのプローブログ分析ダッシュボードを開きます。
サーバー分析	クリックして、選択したノードのサーバー分析ダッシュボードを開きます。
サーバーログ分析	クリックして、選択したノードのサーバーログ分析ダッシュボードを開きます。
セッション活動分析	クリックして、選択したノードのセッションアクティビティ分析ダッシュボードを開きます。
ストレージ分析	クリックして、選択したノードのストレージ分析ダッシュボードを開きます。
ストリーミングレプリケーション分析	クリックして、選択したノードのストリーミングレプリケーション分析ダッシュボードを開きます。
システム待機分析	クリックして、選択したノードのシステム待機分析ダッシュボードを開きます。

[ツール] メニュー

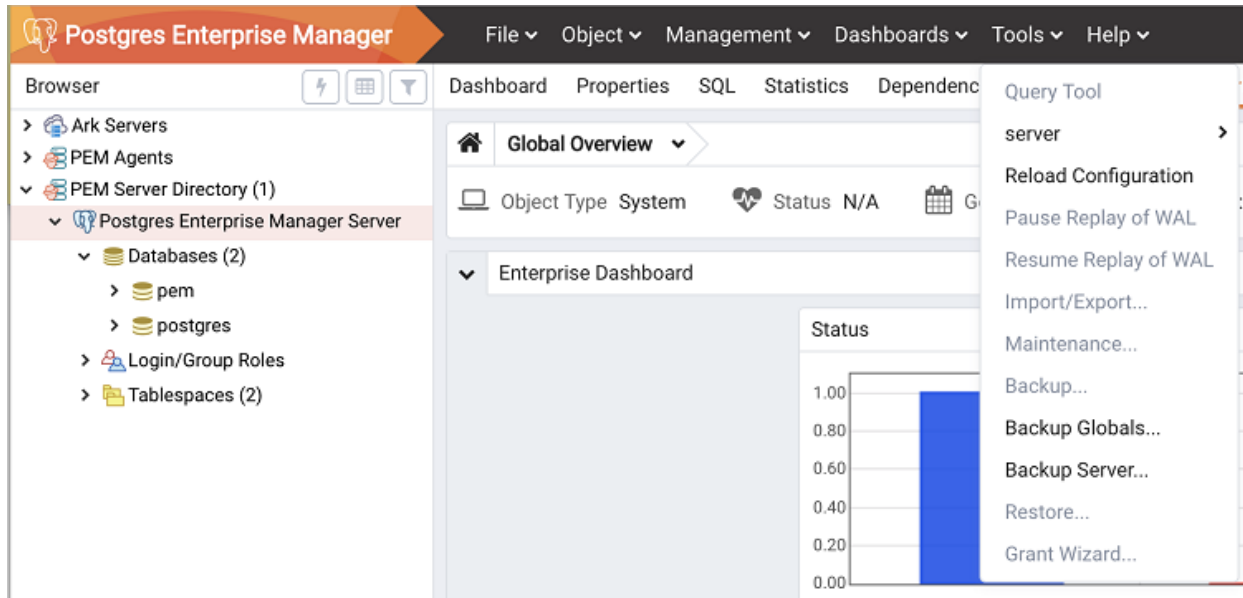


図 8 The Tools menu

Tools メニューのオプションを使用して、次の機能にアクセスします:

オプション	アクション
名前付き復元ポイントを追加	クリックしてダイアログを開き、現在のサーバー状態のポイントインタイムスナップショットを作成します。
バックアップ...	クリックして [バックアップ] ダイアログを開き、データベースオブジェクトをバックアップします。
バックアップグローバル...	クリックして [グローバルのバックアップ...] ダイアログを開き、クラスターオブジェクトをバックアップします。
バックアップサーバー...	[サーバーのバックアップ...] ダイアログをクリックして、サーバーをバックアップします。
グラントウィザード...	クリックして、 Grant Wizard ツールにアクセスします。
インポート/エクスポート...	クリックして [データのインポート/エクスポート...] ダイアログを開き、テーブルからデータをインポートまたはエクスポートします。
メンテナンス...	VACUUM、ANALYZE、REINDEX、または CLUSTER の [メンテナンス...] ダイアログをクリックして開きます。
WAL のリプレイを一時停止	クリックして、WAL ログの再生を一時停止します。
クエリツール	クリックすると、現在選択されているオブジェクトのクエリツールが開きます。
設定の再読み込み...	サーバーを再起動せずに構成ファイルを更新する場合にクリックします。
復元...	クリックして [復元] ダイアログにアクセスし、バックアップからデータベースファイルを復元します。
WAL のリプレイを再開する	クリックして、WAL ログの再生を再開します。

ヘルプメニュー

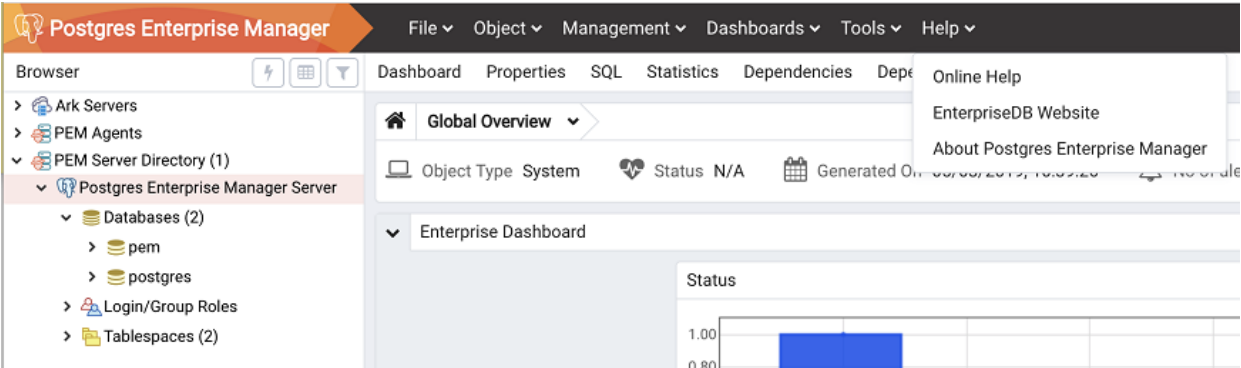


図 9 ヘルプメニュー

Help メニューのオプションを使用して、オンラインヘルプドキュメントにアクセスするか、PEM インストールに関する情報を確認します。

オプション	アクション
オンラインヘルプ	クリックして、Postres Enterprise Manager のドキュメントを開きます。
EnterpriseDB Web サイト	ブラウザウィンドウで EnterpriseDB Web サイトをクリックして開きます。
Postgres Enterprise Manager について	クリックして、Postgres Enterprise Manager のバージョニングおよびユーザー情報を見つけます。

1.5.2 チャート、グラフ、テーブルの制御とカスタマイズ

ダッシュボードの各グラフィックの右上隅にあるアイコンを使用して、現在のユーザーセッション用に PEM クライアントに表示されるチャート、グラフ、および表を制御およびカスタマイズします。

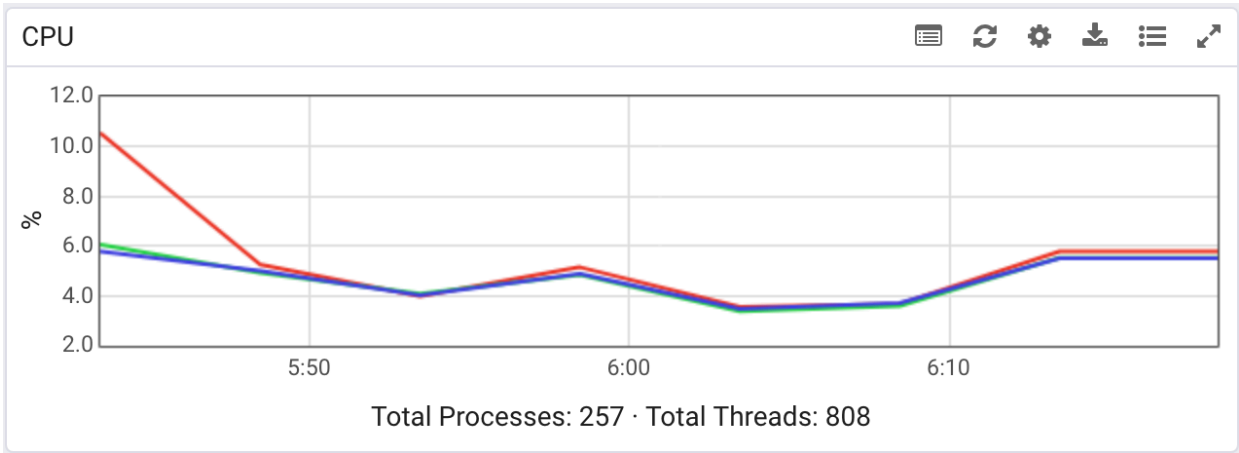


図 10 The PEM Client chart control icons

アイコンを選択して:

	チャート、グラフ、または表のコンテンツを更新します。
	チャートまたはグラフをフルスクリーンに展開します。
	現在のユーザーのチャート、グラフ、またはテーブル設定をカスタマイズします。
	チャートまたはグラフの画像をダウンロードする
	チャート、グラフ、またはテーブルに関する情報を表示します。

PEM ダッシュボードに表示されるグラフィックのカスタマイズの詳細については、PEM クライアントのオンラインヘルプを参照してください。

1.5.3 オンラインヘルプとドキュメント

PEM には、ツールの使用を支援する組み込みヘルプが含まれています。PEM のオンラインヘルプにアクセスするには、[ヘルプ] メニューの [ヘルプ] オプションから [オンラインヘルプ] を選択します。

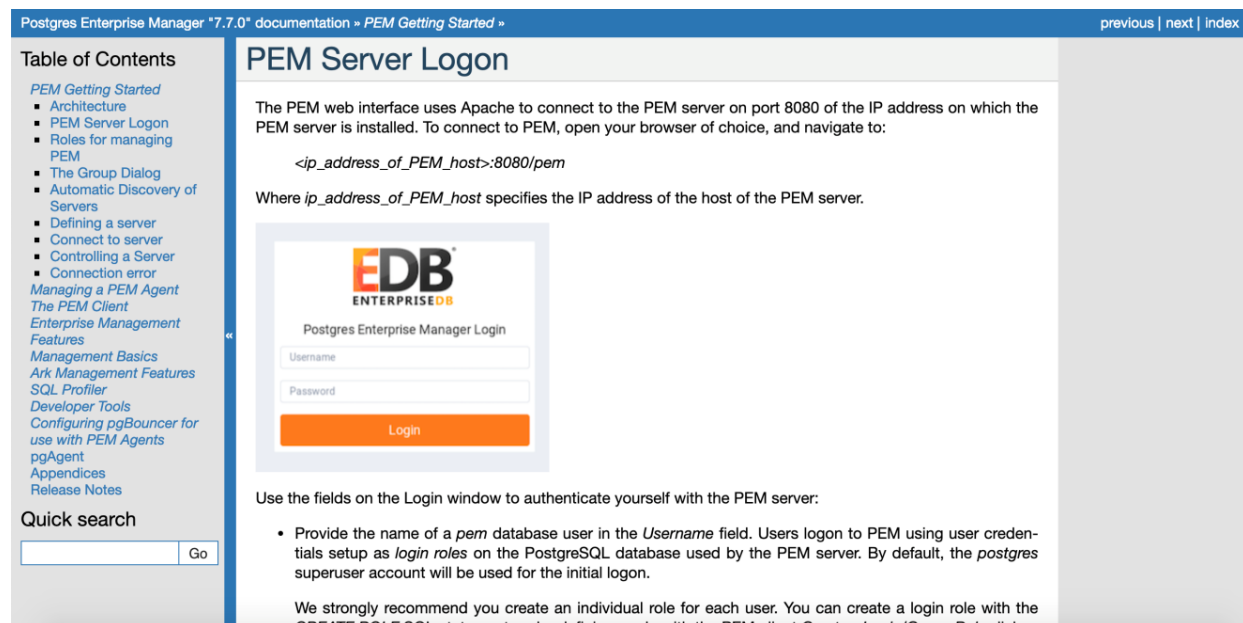


図 11 The PEM online help

Help メニューを使用すると、EnterpriseDB Web サイトにすばやくアクセスできます。

サーバーの登録

PEM でサーバーを管理または監視する前に、サーバーを PEM に登録し、エージェントをバインドする必要があります。サーバーはリモートエージェント（別のホストに存在するエージェント）にバインドできますが、エージェントが同じホストに存在しない場合、インスタンスに関するすべての統計情報にアクセスできません。

2.1 サーバーの手動登録

PEM を使用してサーバーを管理または監視するには、次のことを行う必要があります。

- Advanced Server または PostgreSQL サーバーを PEM サーバーに登録します。
- サーバーを PEM エージェントにバインドします。

Create - Server ダイアログを使用して、サーバーの登録情報を提供し、PEM エージェントをバインドし、サーバーを PEM クライアントツリーコントロールに表示できます。Create - Server ダイアログを開くには、Object メニュー（またはサーバーグループのコンテキストメニュー）の Create オプションをナビゲートし、Server... を選択します。

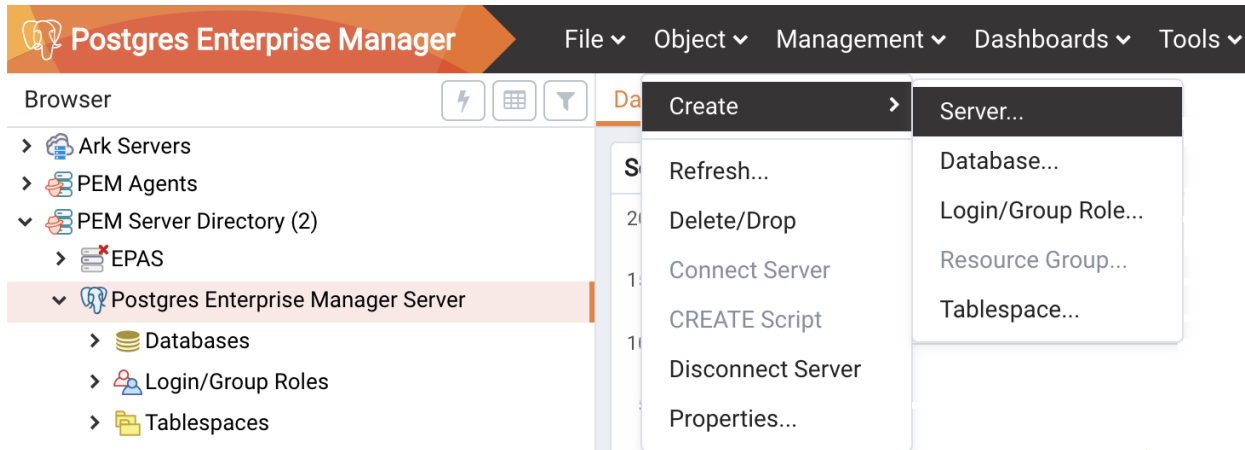


図1 “サーバーの作成”ダイアログへのアクセス

注釈: 登録する Postgres サーバーの `pg_hba.conf` ファイルが、接続を試みる前に PEM クライアントのホストからの接続を許可していることを確認する必要があります。

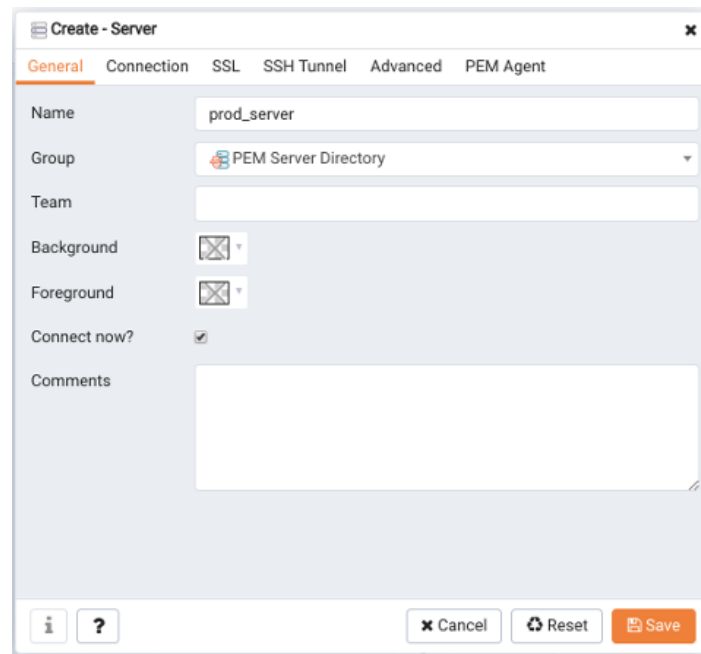


図2 The General tab of the Create – Server dialog

General タブのフィールドを使用して、サーバーの一般的なプロパティを説明します。

- Name フィールドを使用して、サーバーのわかりやすい名前を指定します。指定された名前は、

PEM Browser ツリーコントロールでサーバーを識別します。

- グループを使用して、サーバーとエージェントをツリーコントロールで整理できます。グループを使用すると、多数のサーバーをより簡単に管理できます。たとえば、実稼働グループ、テストグループ、または LAN 固有のグループが必要な場合があります。Group ドロップダウンリストボックスを使用して、新しいサーバーが表示されるサーバーグループを選択します。
- Team フィールドを使用して、Postgres ロール名を指定します。このロールのメンバーである、サーバーを最初に作成した、または PEM サーバーでスーパーユーザー特権を持つ PEM ユーザーのみが、PEM にログオンしたときにこのサーバーを表示します。このフィールドを空白のままにすると、すべての PEM ユーザーにサーバーが表示されます。
- Background カラーセレクターを使用して、サーバーに保存されているデータベースオブジェクトの背後にある PEM ツリーコントロールに表示される色を選択します。
- Foreground カラーセレクターを使用して、サーバーに保存されているオブジェクトの PEM ツリーコントロールのラベルのフォントの色を選択します。
- Connect now? の横にあるチェックボックスをオンにして、Connect now? ボタンをクリックしたときにサーバー接続を試行するように PEM に指示します。PEM クライアントが、後の接続試行まで指定された接続パラメーターを検証しないようにする場合は、Connect now? をオフのままにします。
- Comments フィールドにサーバーに関するメモを入力します。

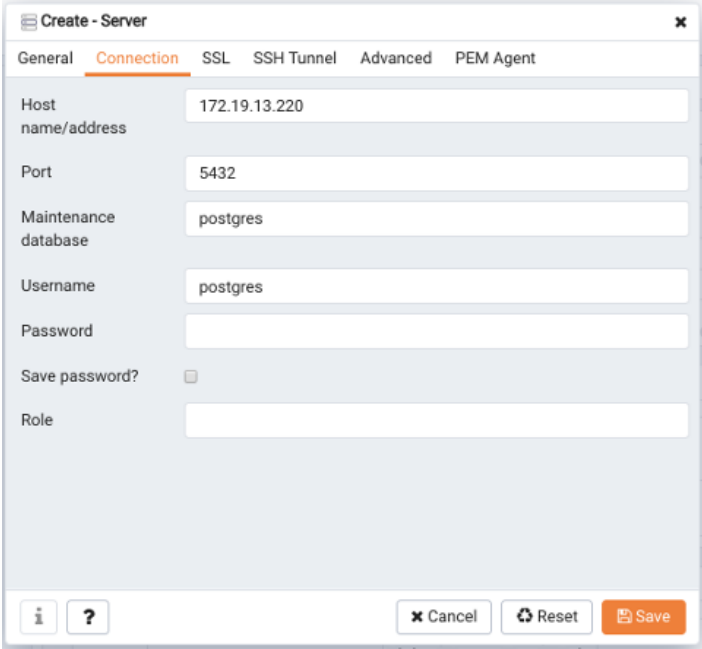


図 3 The Connection tab of the Create – Server dialog

Connection tab のフィールドを使用して、サーバーの接続の詳細を指定します。

- サーバーホストの IP アドレス、または Host name/address フィールドに完全修飾ドメイン名を指定します。Unix ベースのシステムでは、ローカルマシンでデフォルトの PostgreSQL Unix ドメインソケットを使用するためにアドレスフィールドを空白のままにするか、PostgreSQL ソケットを含む代替パスに設定できます。パスを入力する場合、パスは Host name/address で始まる必要があります。
- ホストのポート番号を Port フィールドに指定します。
- Maintenance database フィールドを使用して、PEM が接続する初期データベースの名前を指定します。これには、インストールされた pgAgent スキーマと adminpack オブジェクトが含まれます（両方ともオプション）。PostgreSQL 8.1 以降では、メンテナンス DB は通常 postgres と呼ばれます。以前のバージョンでは template1 がよく使用されますが、テンプレートデータベースが乱雑になるのを避けるために postgres データベースを作成することをお勧めします。
- サーバーで認証するときに使用される名前を Username フィールドに指定します。
- 指定したユーザーに関連付けられているパスワードを Password フィールドに入力します。
- Save password? の隣のボックスをチェックして、後で再利用できるように ~/.pgpass ファイル (Linux の場合) または %APPDATA%/postgresql/pgpass.conf (Windows の場

合) にパスワードを保存するよう PEM に指示します。詳細については、pgpass のドキュメントを参照してください。保存されたパスワードは、すべての libpq ベースのツールに使用されます。パスワードを削除するには、サーバーから切断し、サーバーの [プロパティ] ダイアログを開いて、選択を解除します。

- Role フィールドを使用して、サーバーへの接続後にクライアントが使用する必要がある特権が割り当てられているロールの名前を指定します。これにより、1 つの役割として接続し、接続が確立されたときに別の役割（このフィールドで指定した役割）の許可を引き受けることができます。接続する役割は、指定された役割のメンバーでなければなりません。

図 4 The SSL tab of the Create – Server dialog

SSL を構成するには、SSL タブのフィールドを使用します:

- SSL mode フィールドのドロップダウンリストボックスを使用して、サーバーが使用する SSL 接続のタイプを選択します。SSL 暗号化の使用の詳細については、次の PostgreSQL ドキュメントを参照してください。

<https://www.postgresql.org/docs/current/static/libpq-ssl.html>

プラットフォーム固有の File マネージャーダイアログを使用して、SSL 暗号化をサポートするファイルをサーバーにアップロードできます。ファイルマネージャーにアクセスするには、次の各フィールドの右側にあるアイコンをクリックします。

- Client certificate フィールドを使用して、クライアント SSL 証明書を含むファイル

を指定します。このファイルは、PEM がデスクトップモードでインストールされている場合はデフォルトの `~/.postgresql/postgresql.crt` ファイルを置き換え、PEM が Web モードでインストールされている場合は `<STORAGE_DIR>/<USERNAME>/.postgresql/postgresql.crt` を置き換えます。SSL 接続が確立されない場合、このパラメーターは無視されます。

- Client certificate キーフィールドを使用して、クライアント証明書に使用される秘密キーを含むファイルを指定します。このファイルは、PEM がデスクトップモードでインストールされている場合はデフォルトの `~/.postgresql/postgresql.key` を置き換え、PEM が Web モードでインストールされている場合は `<STORAGE_DIR>/<USERNAME>/.postgresql/postgresql.key` を置き換えます。SSL 接続が確立されない場合、このパラメーターは無視されます。
- Root certificate フィールドを使用して、SSL 認証局を含むファイルを指定します。このファイルはデフォルトの `~/.postgresql/root.crt` ファイルを置き換えます。SSL 接続が確立されない場合、このパラメーターは無視されます。
- Certificate revocation list フィールドを使用して、SSL 証明書失効リストを含むファイルを指定します。このリストは、`~/.postgresql/root.crl` にあるデフォルトのリストを置き換えます。SSL 接続が確立されない場合、このパラメーターは無視されます。
- SSL compression? を **True** に設定すると、SSL 接続を介して送信されるデータが圧縮されます。デフォルト値は **False** です（圧縮は無効です）。SSL 接続が確立されない場合、このパラメーターは無視されます。

警告: 証明書、秘密キー、および失効リストは、PEM サーバープロセスが実行されるユーザーアカウントが所有するサーバー上のユーザーごとのファイルストレージ領域に格納されます。これは、サーバーの管理者がこれらのファイルにアクセスできることを意味します。この機能の使用を選択する前に、適切な注意を払う必要があります。

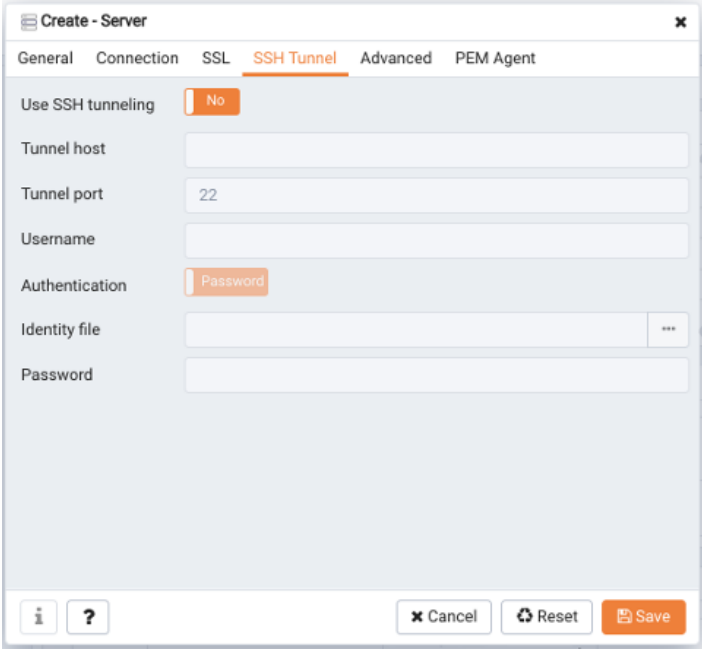


図 5 The SSH Tunnel tab of the Create – Server dialog

SSH トンネリングを設定するには、SSH Tunnel タブのフィールドを使用します。トンネルを使用して、データベースサーバーを（中間プロキシホストを介して）クライアントが直接接続できないネットワーク上にあるサーバーに接続できます。

- 指定したサーバーに接続するときに PEM が SSH トンネルを使用するように指定するには、Use SSH tunneling を Yes に設定します。
- Tunnel host フィールドに、SSH ホストの名前または IP アドレス（クライアント接続の転送に使用される）を指定します。
- Tunnel port フィールドに SSH クライアントのポート（クライアント接続が転送される）を指定します。
- Username フィールドに SSH ホストのログイン特権を持つユーザーの名前を指定します。
- Authentication フィールドで SSH ホストに接続するときに使用される認証のタイプを指定します。
- Password を選択して、PEM が SSH ホストへの認証にパスワードを使用することを指定します。これがデフォルトです。
- 接続時に PEM が秘密鍵ファイルを使用することを指定するには、Identity file を選択します。

- SSH ホストが認証用の秘密鍵ファイルを予期している場合、Identity file フィールドを使用して鍵ファイルの場所を指定します。
- SSH ホストがパスワードを要求している場合、Password フィールドを使用してパスワードを指定するか、ID ファイルを使用している場合はパスフレーズを指定します。

図 6 The Advanced tab of the Create – Server dialog

Advanced タブのフィールドを使用して、サーバーの管理に使用される詳細を指定します。

- Host Address1 フィールドにサーバーホストの IP アドレスを指定します。
- DB restriction フィールドを使用して、**‘pg_database に対して使用される SQL 制限を指定します’**<http://www.postgresql.org/docs/current/interactive/catalog-pg-database.html> ‘_テーブルは、ツリーコントロールに表示されるデータベースを制限します。たとえば、'live_db'、'test_db' と入力して、PEM ブラウザーに live_db および test_db データベースのみを表示するように指示します。pg_namespace に対する制限を入力することにより、データベースプロパティダイアログからデータベースに表示されるスキーマを制限することもできます。_。
- Password file フィールドを使用して、パスワードファイル（.pgpass）の場所を指定します。.pgpass ファイルにより、ユーザーは接続時にパスワードを入力せずにログインできます。詳細については、次の Postgres のドキュメントを参照してください。

<http://www.postgresql.org/docs/current/static/libpq-pgpass.html>

注釈: パスワードファイルの使用は、PEM が libpq v10.0 以降を使用してサーバーに接続している場合にのみサポートされます。

- データベースサービスプロセスを制御するパラメーターを指定するには、Service ID フィールドを使用します。Enterprise Manager ディレクトリに保存されているサーバーの場合、サービス ID を入力します。Windows マシンでは、これは Windows サービスの識別子です。Linux マシンでは、これは /etc/init.d でサーバーを起動するために使用される init スクリプトの名前です。たとえば、Advanced Server 10 サービスの名前は edb-as-10 です。ローカルサーバーの場合、設定はオペレーティングシステムに依存します。

- PEM クライアントが Windows マシンで実行されている場合、十分なアクセス権があれば、ポストマスターサービスを制御できます。サービスの名前を入力します。リモートサーバーの場合は、マシン名を先頭に追加する必要があります (例: PSE1/pgsql-8.0)。PEM は、ローカルマシンで実行されているサービスを自動的に検出します。

- PEM クライアントが Linux マシンで実行されている場合、十分なアクセス権があれば、ローカルマシンで実行されているプロセスを制御できます。pg_ctl プログラムにアクセスするためのフルパスと必要なオプションを提供します。サービス制御機能を実行すると、PEM はこれに status / start / stop キーワードを追加します。例:

```
sudo /usr/local/pgsql/bin/pg_ctl -D /data/pgsql
```

- サーバーが Failover Manager クラスターのメンバーである場合、PEM を使用してクラスターの状態を監視し、必要に応じてマスターノードを交換できます。PEM がフェールオーバーマネージャーを監視できるようにするには、EFM cluster name フィールドを使用してクラスター名を指定します。クラスター名は、Failover Manager クラスタープロパティファイルの名前のプレフィックスです。たとえば、クラスタープロパティファイルの名前が efm.properties の場合、クラスター名は efm です。
- PEM を使用して Failover Manager クラスターのステータスを監視している場合、EFM installation path フィールドを使用して Failover Manager バイナリファイルの場所を指定します。デフォルトでは、Failover Manager バイナリファイルは /usr/efm-2.x/bin にインストールされます。* x * は Failover Manager のバージョンを指定します。

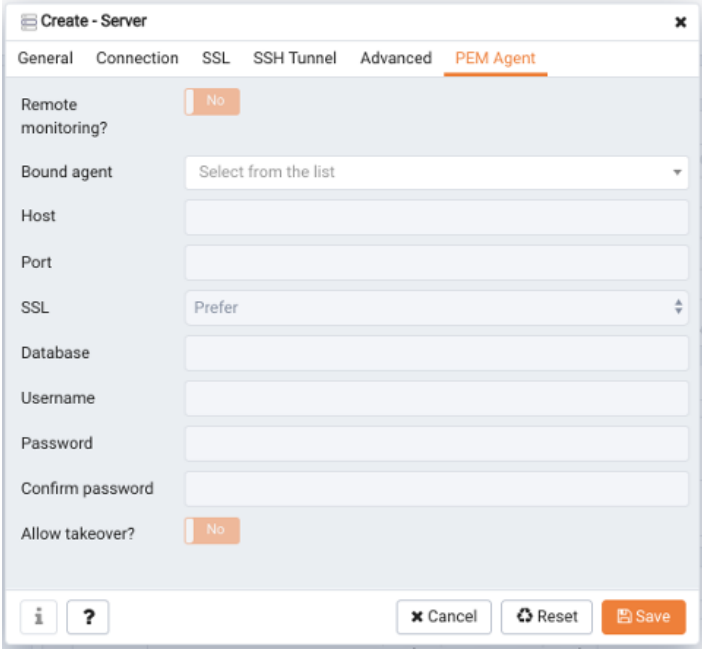


図 7 The PEM Agent tab of the Create – Server dialog

PEM Agent タブのフィールドを使用して、PEM エージェントの接続の詳細を指定します。

- Remote monitoring? スライダーを Yes に移動して、PEM エージェントが監視対象サーバーと同じホストにないことを示します。リモート監視が有効になっている場合、監視対象サーバーのエージェントレベルの統計はカスタムチャートとダッシュボードで利用できず、一部の PEM ユーティリティ（Audit Manager、Capacity Manager、Log Manager、Postgres Expert など）からリモートサーバーにアクセスできません（チューニングウィザード）。
- Bound agent ラベルの右側にあるドロップダウンリストボックスを使用して、Enterprise Manager エージェントを選択します。1 つのエージェントで複数の Postgres サーバーを監視できます。
- データベースサーバーに接続するときにエージェントが使用する IP アドレスまたはソケットパスを Host フィールドに入力します。デフォルトでは、エージェントは General タブに表示されるホストアドレスを使用します。Unix サーバーでは、ソケットパス、たとえば /tmp を指定できます。
- サーバーに接続するときにエージェントが使用する Port 番号を入力します。デフォルトでは、エージェントは Properties タブで定義されたポートを使用します。
- SSL 動作モードを指定するには、SSL フィールドのドロップダウンリストボックスを使用します。require、prefer、allow、disable、verify-ca または verify-full を指定します。SSL 暗号化の

使用の詳細については、次の PostgreSQL ドキュメントを参照してください。

<http://enterprisedb.com/docs/en/10/pg/libpq-ssl.html>

- Database フィールドを使用して、エージェントが最初に接続するデータベースの名前を指定します。
- User name フィールドで、サーバーに接続するときにエージェントが使用するロールの名前を指定します。指定されたロールがデータベースのスーパーユーザーでない場合、一部の機能が期待どおりに機能しないことに注意してください。指定されたロールがデータベーススーパーユーザーでない場合に機能しない機能のリストについては、[エージェント権限を参照してください](#)。

Postgres バージョン 10 以降を使用している場合、pg_monitor ロールを使用して、必要な特権を非スーパーユーザーに付与できます。pg_monitor ロールの詳細については、以下を参照してください。

<https://www.postgresql.org/docs/current/default-roles.html>

- サーバーに接続するときにエージェントが使用するパスワードを Password フィールドに指定し、Confirm password フィールドにもう一度入力して確認します。パスワードを指定しない場合、エージェントの認証を手動で構成する必要があります。たとえば、.pgpass ファイルを使用できます。
- Allow takeover? スライダーを Yes に設定して、サーバーが別のエージェントに引き継がれることを指定します。この機能により、エージェントは、たとえば、高可用性フェイルオーバープロセスの一環としてサーバーが別のホストに移動された場合に、データベースサーバーの監視を行うことができます。

サーバーのプロパティを表示するには、**PEM** クライアントツリーコントロールでサーバー名を右クリックし、コンテキストメニューから Properties… オプションを選択します。サーバーのプロパティを変更するには、Properties ダイアログを開く前にサーバーから切断します。

2.2 自動サーバー検出

監視したいサーバーが監視エージェントと同じホストにある場合、Auto Discovery ダイアログを使用して登録とバインドのプロセスを簡素化できます。

特定のエージェントの自動検出を有効にするには、Server Auto Discovery プローブを有効にする必要があります。Manage Probes タブにアクセスするには、PEM クライアントツリーコントロールで PEM エージェントの名前を強調表示し、Management メニューから Manage Probes... を選択します。Manage Probes タブが開いたら、Enabled? 列のスライダーコントロールが Yes に設定されていることを確認します。

Auto Discovery ダイアログを開くには、PEM クライアントツリーコントロールで PEM エージェントの名前を強調表示し、Management メニューから Auto Discovery... を選択します。

図 8 The PEM Auto Discovery dialog

Auto Discovery ダイアログが開くと、Discovered Database Servers ボックスに現在 PEM エージェントによって監視されていないサーバーのリストが表示されます。サーバー名の横にあるチェックボックスをオンにして、Server Connection Details ボックスにサーバーに関する情報を表示し、Agent Connection Details ボックスにエージェントの接続プロパティを表示します。

Check All ボタンを使用して、表示されているすべてのサーバーの横にあるボックスを選択するか、Uncheck All を使用してサーバー名の左側にあるすべてのボックスの選択を解除します。

Server Connection Details ボックスのフィールドは、PEM が監視するサーバーに関する情報を提供します:

- Name フィールドで監視対象サーバーの名前を受け入れるか変更します。指定した名前は、PEM クライアントのツリーコントロールに表示されます。
- Server group ドロップダウンリストボックスを使用して、サーバーが PEM クライアントツリーコントロールに表示されるサーバーグループを選択します。
- 監視対象サーバーの IP アドレスを指定するには、Host name/address フィールドを使用します。
- Port フィールドには、サーバーによって監視されているポートが表示されます。このフィールドは変更できません。
- Service ID フィールドにサービスの名前を入力します。一部の PEM 機能を有効にするには、サービス名を指定する必要があることに注意してください。
- デフォルトでは、Maintenance database フィールドは、選択したサーバーが Postgres メンテナンスデータベースを使用することを示します。インストールの Maintenance database フィールドの内容をカスタマイズします。

Agent Connection Details ボックスのフィールドは、サーバーへの接続時に PEM エージェントが使用するプロパティを指定します。

- Host フィールドには、PEM エージェントのバインドに使用される IP アドレスが表示されます。
- User name フィールドには、選択したサーバーに接続するときに PEM エージェントが使用する名前が表示されます。
- Password フィールドには、指定されたユーザー名に関連付けられたパスワードが表示されます。
- SSL mode フィールドのドロップダウンリストボックスを使用して、SSL 接続設定を指定します。

監視用にバインドしているサーバーの接続プロパティの指定が終了したら、OK ボタンをクリックしてサーバーを登録します。変更を保存せずに終了するには、Cancel をクリックします。

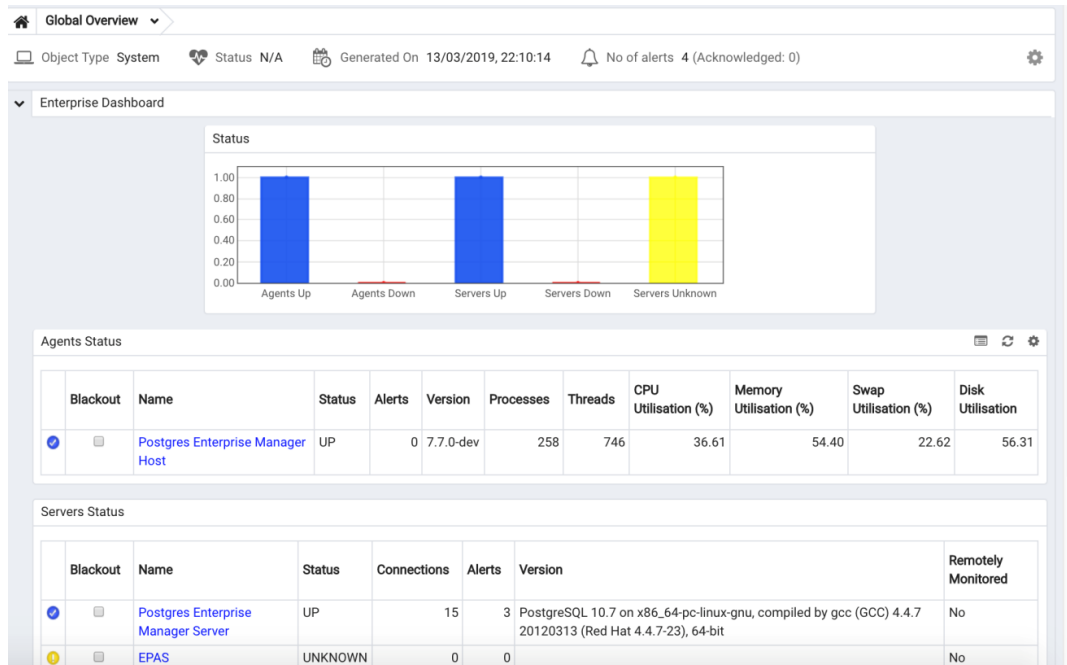


図 9 登録済みサーバー

OK ボタンをクリックすると、新しく登録されたサーバーが PEM ツリーコントロールに表示され、PEM サーバーによって監視されます。

2.3 pemworker ユーティリティを使用してサーバーを登録する

pemworker ユーティリティを使用して、PEM サーバーによる監視用にサーバーを登録したり、データベースサーバーの登録を解除したりできます。登録中、pemworker ユーティリティは、登録コマンドを呼び出したシステム上にあるエージェントに新しいサーバーをバインドします。サーバーを登録するには:

Linux ホストでは、次のコマンドを使用します。

```
pemworker --register-server
```

Windows ホストでは、次のコマンドを使用します。

```
pemworker.exe REGISTER-SERVICE
```

pemworker ユーティリティを呼び出すときに、コマンドラインオプションをコマンド文字列に追加します。各オプションの後には、対応する値が続く必要があります。

オプション	説明
--pem-user	PEM 管理ユーザーの名前を指定します。必須です。
--server-addr	サーバーホストの IP アドレス、または完全修飾ドメイン名を指定します。Unix ベースのシステムでは、ローカルマシンでデフォルトの PostgreSQL Unix ドメインソケットを使用するためにアドレスフィールドを空白のままにするか、PostgreSQL ソケットを含む代替パスに設定できます。パスを入力する場合、パスは/で始まる必要があります。必須です。
--server-port	ホストのポート番号を指定します。必須です。
--server-database	サーバーが接続するデータベースの名前を指定します。必須です。
--server-user	サーバーのモニター時にエージェントが使用するユーザーの名前を指定します。必須です。
--server-service-name	登録されるサーバーでの操作を制御するデータベースサービスの名前を指定します (STOP、START、RESTART など)。オプション。
--remote-monitoring	--remote-monitoring 句と値 false (デフォルト) を含めて、サーバーが PEM エージェントと同じマシンにインストールされていることを示します。リモート監視が有効 (true) の場合、監視対象サーバーのエージェントレベルの統計はカスタムチャートおよびダッシュボードで利用できず、一部の PEM ユーティリティ (監査マネージャー、キャパシティマネージャー、ログマネージャー、Postgres Expert and Tuning Wizard)。必須です。
--efm-cluster-name	サーバーを監視するフェールオーバーマネージャークラスターの名前を指定します (該当する場合)。オプション。
--efm-install-path	フェールオーバーマネージャーのインストールディレクトリへの完全なパスを指定します (該当する場合)。オプション。
--asb-host-name	エージェントが接続しているホストの名前を指定します。
--asb-host-port	データベースに接続するときにエージェントが使用するポート番号を指定します。
--asb-host-db	エージェントが接続するデータベースの名前を指定します。
--asb-host-user	データベースでの認証時にエージェントが提供するデータベースユーザー名を指定します。
--asb-ssl-mode	接続に使用される SSL 認証のタイプを指定します。サポートされる値には、prefer、require、disable、verify-CA、verify-full があります。
--group	サーバーが表示されるグループの名前を指定します。
--team	サーバーへのアクセスを許可されるグループロールの名前を指定します。
2.3. pemworker	監視対象サーバーを登録するユーザーの名前を指定します。

環境変数 `PEM_SERVER_PASSWORD` を設定して PEM サーバーのパスワードを提供し、`pemworker` が PEM 管理ユーザーとして接続できるようにします。

環境変数 `PEM_MONITORED_SERVER_PASSWORD` を設定して、`pemagent` によって登録および監視されているデータベースサーバーのパスワードを提供します。

パスワードを提供しないと、パスワード認証エラーが発生します。PEM サーバーは、サーバーが適切に登録されたことを確認します。

2.3.1 pemworker ユーティリティを使用してサーバーの登録を解除する

pemworker ユーティリティを使用して、データベースサーバーの登録を解除できます。サーバーの登録を解除するには、pemworker ユーティリティを呼び出します:

Linux ホストでは、次のコマンドを使用します。

```
pemworker --unregister-server
```

Windows ホストでは、次のコマンドを使用します。

```
pemworker.exe UNREGISTER-SERVICE
```

pemworker ユーティリティを呼び出すときに、コマンドラインオプションをコマンド文字列に追加します。各オプションの後には、対応する値が続く必要があります。

オプション	説明
--pem-user	PEM 管理ユーザーの名前を指定します。必須です。
--server-addr	サーバーホストの IP アドレス、または完全修飾ドメイン名を指定します。Unix ベースのシステムでは、ローカルマシンでデフォルトの PostgreSQL Unix ドメインソケットを使用するためにアドレスフィールドを空白のままにするか、PostgreSQL ソケットを含む代替パスに設定できます。パスを入力する場合、パスは/で始まる必要があります。必須です。
--server-port	ホストのポート番号を指定します。必須です。

環境変数 PEM_SERVER_PASSWORD を設定して PEM サーバーのパスワードを提供し、pemworker が PEM 管理ユーザーとして接続できるようにします。

パスワードを提供しないと、パスワード認証エラーが発生します。PEM サーバーは、サーバーが登録解除されたことを確認します。

2.4 接続とバインドの検証

登録されると、新しいサーバーは PEM の Browser ツリーコントロールに追加され、Global Overview に表示されます。

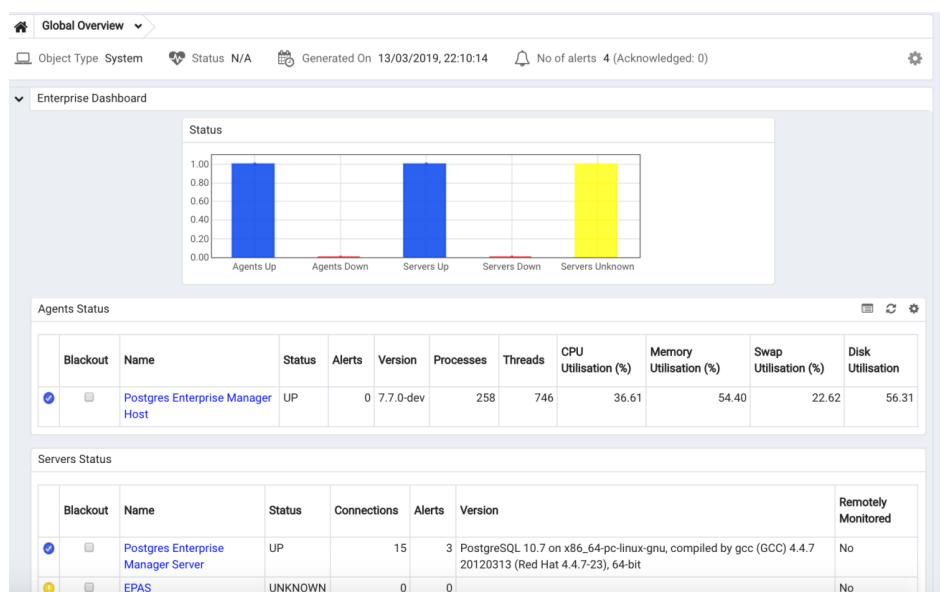


図 10 グローバル概要ダッシュボード

新しくバインドされたサーバーに最初に接続するとき、Global Overview ダッシュボードは、サーバーリストに Global Overview のステータスで新しいサーバーを表示する場合があります。サーバーを認識する前に、バインドされたエージェントは多数のプローブを実行してサーバーを検査する必要がありますが、ネットワークの可用性によっては完了するまでに数分かかる場合があります。

数分以内に、Global Overview ダッシュボードの棒グラフに、エージェントが正常に接続されたことが表示され、新しいサーバーが Postgres Server Status リストに追加されます。

5 分経過しても Global Overview ダッシュボードに新しいサーバーが表示されない場合は、監視エージェントのログファイルを確認してエラーを確認する必要があります。ツリーコントロールでエージェントの名前を右クリックし、コンテキストメニューの Dashboards サブメニューから Probe Log Analysis オプションを選択します。

第 3

証明書の管理

PEM サーバーバックアップデータベースのデータディレクトリに保存されているファイルには、PEM サーバーが安全な接続を利用するのに役立つ情報が含まれています。

- `ca_certificate.crt`
- `ca_key.key`
- `server.crt`
- `server.key`
- `root.crl`
- `root.crt`

PEM サーバーと共にインストールされる PEM エージェントは、`ca_certificate.crt` ファイルの有効期限を監視します。証明書の有効期限が切れそうになると、PEM は次のことを行います。

- 既存の証明書ファイルのバックアップを作成します。
- 新しい証明書ファイルを作成し、新しい CA 証明書ファイルを PEM サーバーの `root.crt` ファイルに追加します。
- アクティブなエージェントの証明書ファイルを更新するジョブを作成します。

- PEM サーバーを再起動します。

エージェントをアンインストールすると、そのエージェントに関連付けられた証明書が証明書失効リスト（`root.crl` ファイルに保持）に追加され、証明書を使用して PEM サーバーに接続できなくなります。

次のセクションには、証明書ファイルを手動で置き換えることに関する詳細情報が含まれています。

3.1 SSL 証明書の置き換え

次の手順では、既存の PEM インストールでの SSL 証明書の置換について詳しく説明します。サーバーを同時に新しいバージョンにアップグレードする場合は、SSL 証明書を置き換える前に、すべての PEM インストーラー（最初にサーバーインストーラー、次にエージェントインストーラー）を呼び出します。次に：

1. 最初にサーバーホストで実行中のすべての PEM エージェントを停止し、次に監視対象ノードで停止します。

Linux ホストで PEM エージェントを停止するには、ターミナルウィンドウを開き、スーパーユーザー権限を引き受けて、次のコマンドを入力します。

```
/etc/init.d/pemagent stop
```

Windows ホストでは、Services アプレットを使用して PEM エージェントを停止できます。PEM エージェントサービスの名前は **Postgres Enterprise Manager Agent** です。Services ダイアログでサービス名を強調表示し、**Stop the service** をクリックします。

2. 既存の SSL キーと証明書のバックアップを取ります。SSL キーと証明書は、PEM インストールの `data` ディレクトリに保存されます。たとえば、Linux システムのデフォルトの場所は次のとおりです。

```
/opt/PostgreSQL/10/data
```

次のファイルのコピーを作成し、各ファイルに拡張子を追加して、名前を一意にします。

- `ca_certificate.crt`
- `ca_key.key`
- `root.crt`
- `root.crl`

- server.key
- server.crt

たとえば、次のコマンド:

```
# cp ca_certificate.crt ca_certificate_old.crt
```

エントリに old という単語が追加された ca_certificate ファイルのバックアップを作成します。

3. openssl_rsa_generate_key() 関数を使用して ca_key.key ファイルを生成します:

```
/opt/PostgreSQL/10/bin/psql -U postgres -d pem --no-psqlrc -t  
-A -c /"SELECT public.openssl_rsa_generate_key(1024)/" > /opt/  
PostgreSQL/10/data/ca_key.key
```

ca_key.key ファイルを作成した後、ca_certificate.crt ファイルを生成し、ca_key.key ファイルの権限を変更するときに使用する変数 CA_KEY の内容を cat にします:

```
CA_KEY=$(cat /opt/PostgreSQL/10/data/ca_key.key)
```

```
chmod 600 /opt/PostgreSQL/10/data/ca_key.key
```

4. キーを使用して ca_certificate.crt ファイルを生成します。簡単にするために、SQL クエリを一意的な名前で一時ファイルに配置します。

```
echo /"SELECT openssl_csr_to_cert(openssl_rsa_key_to_csr('${CA_KEY}',  
'PEM','US','MA','Bedford','Postgres Enterprise Manager',  
'support@enterprisedb.com'), NULL, '/opt/PostgreSQL/10/data/  
ca_key.key')/" > /tmp/_random.$$
```

次に、変数を使用してクエリを実行し、コンテンツを ca_certificate.crt ファイルに配置します。

```
/opt/PostgreSQL/10/bin/psql -U postgres -d pem --no-psqlrc -t -A  
-f /tmp/_random.$$ > /opt/PostgreSQL/10/data/ca_certificate.crt
```

ca_certificate.crt ファイルの権限を変更し、SQL コマンドを含む一時ファイルを削除します。

```
chmod 600 /opt/PostgreSQL/10/data/ca_certificate.crt
```

```
rm -f /tmp/_random.$$
```

5. `ca_certificate.crt` ファイルを `root.crt` ファイルとして再利用します:

```
cp /opt/PostgreSQL/10/data/ca_certificate.crt /opt/PostgreSQL/10/data/root.crt
```

`root.crt` ファイルのパーミッションを変更します:

```
chmod 600 /opt/PostgreSQL/10/data/root.crt
```

6. `openssl_rsa_generate_crl()` 関数を使用して、証明書失効リスト (`root.crl`) を作成します。

```
/opt/PostgreSQL/10/bin/psql -U postgres -d pem --no-psqlrc -t -A  
-c /"SELECT openssl_rsa_generate_crl('/opt/PostgreSQL/9.5/data/  
ca_certificate.crt', '/opt/PostgreSQL/10/data/ca_key.key')/" >  
/opt/PostgreSQL/10/data/root.crl
```

`root.crl` ファイルのパーミッションを変更します:

```
chmod 600 /opt/PostgreSQL/10/data/root.crl
```

7. `openssl_rsa_generate_key()` 関数を使用して `server.key` ファイルを生成します:

```
/opt/PostgreSQL/10/bin/psql -U postgres -d pem --no-psqlrc -t  
-A -c /"SELECT public.openssl_rsa_generate_key(1024)/" >> /opt/  
PostgreSQL/10/data/server.key
```

`server.key` ファイルを作成した後、`server.crt` ファイルを生成し、`server.key` ファイルの権限を変更するときに使用する変数 `SSL_KEY` の内容を `cat` にします:

```
SSL_KEY=$(cat /opt/PostgreSQL/10/data/server.key)
```

```
chmod 600 /opt/PostgreSQL/10/data/server.key
```

8. `SSL_KEY` を使用してサーバー証明書を生成します。証明書を `server.crt` ファイルに保存します。簡単にするために、最初に `SQL` クエリを一意的な名前を持つ一時ファイルに配置します。

```
echo /"SELECT openssl_csr_to_crt(openssl_rsa_key_to_csr('${SSL_KEY}'),  
'PEM', 'US', 'MA', 'Bedford', 'Postgres Enterprise Manager',  
'support@enterprisedb.com'), '/opt/PostgreSQL/10/data/  
ca_certificate.crt', '/opt/PostgreSQL/10/data/ca_key.key')/" >  
/tmp/_random.$$
```

```
/opt/PostgreSQL/10/bin/psql -U postgres -d pem --no-psqlrc -t -A  
-f /tmp/_random.$$ >> /opt/PostgreSQL/10/data/server.crt
```

9. server.crt ファイルの特権を変更し、一時ファイルを削除します。

```
chmod 600 /opt/PostgreSQL/10/data/server.crt
```

```
rm -f /tmp/_random.$$
```

10. Postgres サーバーを再起動します。

```
/etc/init.d/postgresql-10 restart
```

3.2 エージェント SSL 証明書の更新

PEM サーバーと対話するエージェントごとに、以下を行う必要があります。

- rsa キーと証明書を生成します。
- キーと証明書をエージェントにコピーします。
- エージェントを再起動します。

各エージェントには、pem データベースの pem.agent テーブルに保存される一意の識別子があります。キーと証明書のファイルを、エージェントの識別子に対応するキーまたは証明書に置き換える必要があります。次のキーファイルペアを生成する前に、agent.key および agent.crt ファイル（ステップ 2 および 3 で生成されたファイルをそれぞれの PEM エージェントホストの所定の場所に移動する必要があることに注意してください。後続のコマンドは以前に生成されたファイルを上書きします。

PEM エージェントキーファイルペアを生成するには:

1. psql を使用して、エージェントの数とそれに対応する識別子を見つけます。

```
/opt/PostgreSQL/10/bin/psql -U postgres -d pem --no-psqlrc -t -A  
-c /"SELECT ID FROM pem.agent/"
```

- Linux では、/etc/postgres-reg.ini ファイルの PEMagent セクションで、エージェント識別子とキーと証明書の場所を見つけることもできます。

- Windows では、情報はレジストリに保存されます。

- 64 ビット Windows インストールでは、次を確認します。

```
HKEY_LOCAL_MACHINE/SOFTWARE/Wow6432Node/EnterpriseDB/PEM/  
agent
```

- 32 ビット Windows インストールでは、次を確認します。

```
HKEY_LOCAL_MACHINE/SOFTWARE/EnterpriseDB/PEM/agent
```

2. キーファイルを必要とするエージェントを特定した後、各エージェントに対して agent.key を生成します。キーを生成するには、次のコマンドを実行して、出力をファイルにキャプチャします。

```
/opt/PostgreSQL/10/bin/psql -U postgres -d pem --no-psqlrc -t -A  
-c /"SELECT openssl_rsa_generate_key(1024)/" > agent.key
```

agent.key ファイルの特権を変更します:

```
chmod 600 agent.key
```

3. 各エージェントの証明書を生成します。証明書を生成するには、次のコマンドを実行して、出力を証明書ファイルにキャプチャします。

```
/opt/PostgreSQL/10/bin/psql -U postgres -d pem --no-psqlrc -t  
-A -c /"SELECT openssl_csr_to_cert(openssl_rsa_key_to_csr('$(cat  
agent.key)', 'agent<$ID>', 'US', 'MA', 'Bedford', 'Postgres  
Enterprise Manager', 'support@enterprisedb.com'), '/opt/  
PostgreSQL/10/data/ca_certificate.crt', '/opt/PostgreSQL/10/data/  
ca_key.key')/" > agent.crt
```

ここで、*\$ID*はエージェントのエージェント番号です (psql コマンドラインから取得)。

4. agent.crt ファイルの特権を変更します:

```
chmod 600 agent.crt
```

5. PEM エージェントサービスを再起動する前に、各エージェントのキーと証明書ファイルを新しく生成されたファイルで置き換えます。

- Linux では、次のコマンドを使用してサービスを再起動します。

```
/etc/init.d/pemagent start
```

- Windows ホストでは、サービスアプレットを使用して PEM エージェントを起動できます。PEM エージェントサービスの名前は Postgres Enterprise Manager エージェントです。Postgres Enterprise Manager ダイアログでサービス名を強調表示し、Start the service をクリックします。

PEM サーバーの管理

以下のセクションでは、PEM サーバーとエージェントの再起動、PEM サーバーまたは PEM エージェントの制御、Linux および Windows での HTTPD サービスの制御、HTTPD サーバーの制御、PEM 認証とセキュリティの管理、変更など、PEM サーバーに関連するタスクに関する情報を提供しますプロキシサーバーなどを使用するように PEM を変更する `pg_hba.conf` ファイル

4.1 PEM サーバーとエージェントの起動と停止

PEM サーバーは、それが存在する Postgres サーバーインスタンスが起動、停止、または再起動したときに起動、停止、および再起動します。Postgres サーバーの制御に使用するのと同じコマンドを使用して PEM サーバーを制御します。Linux プラットフォームでは、サービススクリプトを停止および開始するコマンドは、プラットフォームおよび OS バージョンによって異なります。

PEM エージェントは、`pemagent` という名前のサービスによって制御されます。

Windows オペレーティングシステムには、サーバーの状態を表示し、ポイントアンドクリックのサーバー制御を提供するグラフィカルサービスコントローラーが含まれています。Windows の Control Panel から Services ユーティリティにアクセスできます。ユーティリティが開いたら、スクロールバーを使用してリストされたサービスをナビゲートし、サービス名を強調表示します。

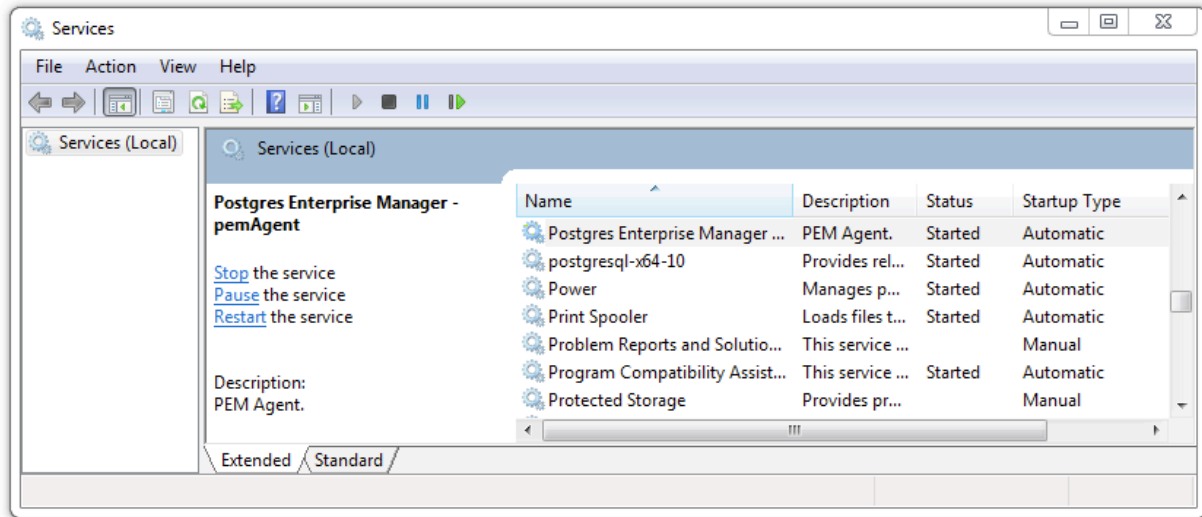


図 1 The PEM service in the Windows Services window

サービスの状態を制御するには、Stop、Pause、Start、またはRestart ボタンを使用します。

サービスを停止すると、Postgres サーバーに接続しているユーザー（またはクライアントアプリケーション）が突然切断されることに注意してください。サービスの制御の詳細については、次の EnterpriseDB Web サイトから入手できる* EDB Postgres Advanced Server インストールガイド*を参照してください。

<https://www.enterprisedb.com/resources/product-documentation>

4.2 監視対象サーバーのリモート起動と停止

PEM では、PEM クライアントを使用して管理対象サーバーインスタンスを起動およびシャットダウンできます。PEM がサービスを管理できるようにサーバーを構成するには、サーバー登録ダイアログを完了して、データベースサーバーを PEM エージェントに登録し、以下を実行します。

- Properties ダイアログで Store on PEM Server オプションを指定します。
- Advanced タブの Service ID フィールドにサービススクリプトの名前を指定します:
 - Advanced Server の場合、サービス名は edb-as-*x* または ppas-*x* です。
 - PostgreSQL の場合、サービス名は postgresql-*x* です。

Where *x* indicates the server version number.

サーバーに接続した後、ツリーコントロールでサーバー名を強調表示し、Management メニューから Queue Server Startup または Queue Server Shutdown を選択することにより、サーバーを起動または停止できます。

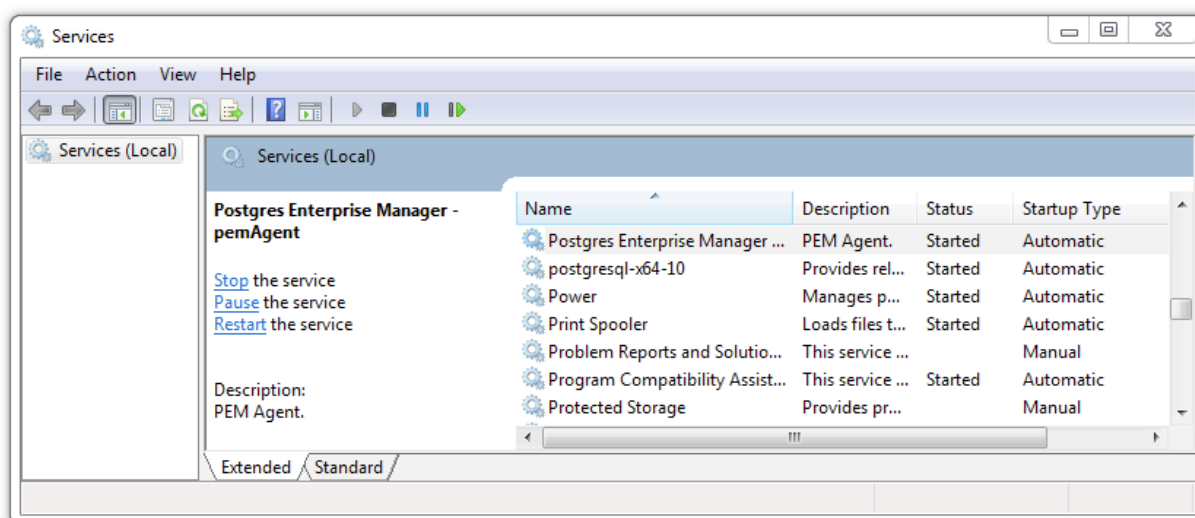


図 2 管理対象サーバーの管理メニュー

4.3 Linux での PEM サーバーまたは PEM エージェントの制御

Linux プラットフォームでは、以下を制御するサービススクリプトの名前:

- Advanced Server 上の PEM サーバーは `edb-as-<x>` または `ppas-<x>` です
- PostgreSQL 上の PEM サーバーは `postgresql-<x>` です
- PEM エージェントは `pemagent` です

Where *x* indicates the server version number.

サービススクリプトを使用して、サービスを制御できます。

- RHEL または CentOS バージョン 6.x でサービスを制御するには、コマンドラインを開き、スーパーユーザー権限を引き受けて、次のように入力します。

```
/etc/init.d/<service_name> <action>
```

- RHEL または CentOS バージョン 7.x でサービスを制御するには、コマンドラインを開き、スーパーユーザー権限を引き受けて、コマンドを発行します。

```
systemctl <service_name> <action>
```

どこで:

service_name is the name of the service.

action specifies the action taken by the service. Specify:

- `start` はサービスを開始します。
- `stop` はサービスを停止します。
- サービスを停止してから開始するには `restart`。
- `status` はサービスのステータスを確認します。

4.4 Windows での PEM サーバーまたは PEM エージェントの制御

Windows オペレーティングシステムには、サーバーの状態を表示し、ポイントアンドクリックのサーバー制御を提供するグラフィカルサービスコントローラーが含まれています。以下を制御するサービスの登録名:

- PostgreSQL 上の PEM サーバーホストは `postgresql-<x>` です
- Advanced Server 上の PEM サーバーホストは、`edb-as-<x>` または `ppas-<x>` です
- PEM エージェントは `Postgres Enterprise Manager - pemAgent` です

Where *x* indicates the server version number.

Windows の Control Panel をナビゲートして、Services ユーティリティを開きます。ユーティリティが開いたら、スクロールバーを使用してサービスのリストを参照します。

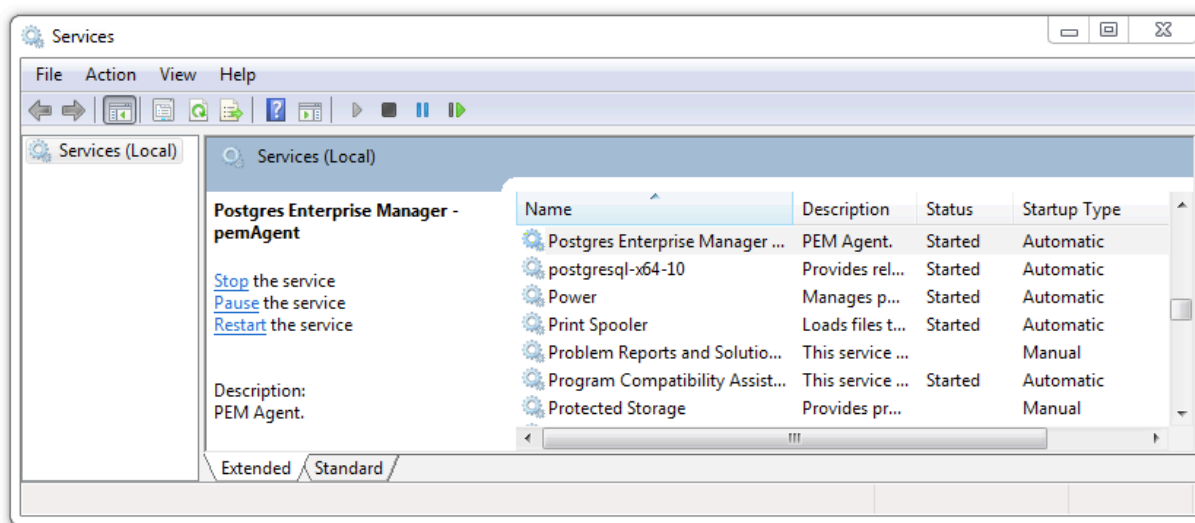


図 3 The Windows Services window

サービスを停止するには、`Stop the service` オプションを使用します。サーバーに接続しているユーザー（またはクライアントアプリケーション）は、サービスを停止すると突然切断されます。

`Pause the service` オプションを使用して、サービスの構成パラメーターを再ロードするように Postgres に指示します。`Pause the service` オプションは、多くの設定パラメーターのユーザーセッションを中断することなくパラメーターをリセットする効果的な方法です。

サービスを開始するには `Start the service` オプションを使用します。

4.5 HTTPD サーバーの制御

Linux では、コマンドラインを開き、次のコマンドを入力して、PEM-HTTPD サービスのステータスを確認できます。

```
ps -ef | grep httpd
```

Linux が次の例のような回答で応答した場合、httpd は実行されていません。

```
user 13321 13267 0 07:37 pts/1 00:00:00 grep httpd
```

CentOS または RHEL 6.x システムでサービスを開始するには、次のコマンドを使用します。

```
/etc/init.d/httpd start
```

CentOS または RHEL 7.x システムでサービスを開始するには、次のコマンドを使用します。

```
systemctl start httpd
```

Windows では、Services アプレットを使用して PEM HTTPD サービスのステータスを確認できます。サービスアプレットを開いた後、リストをスクロールして PEM HTTPD サービスを見つけます。

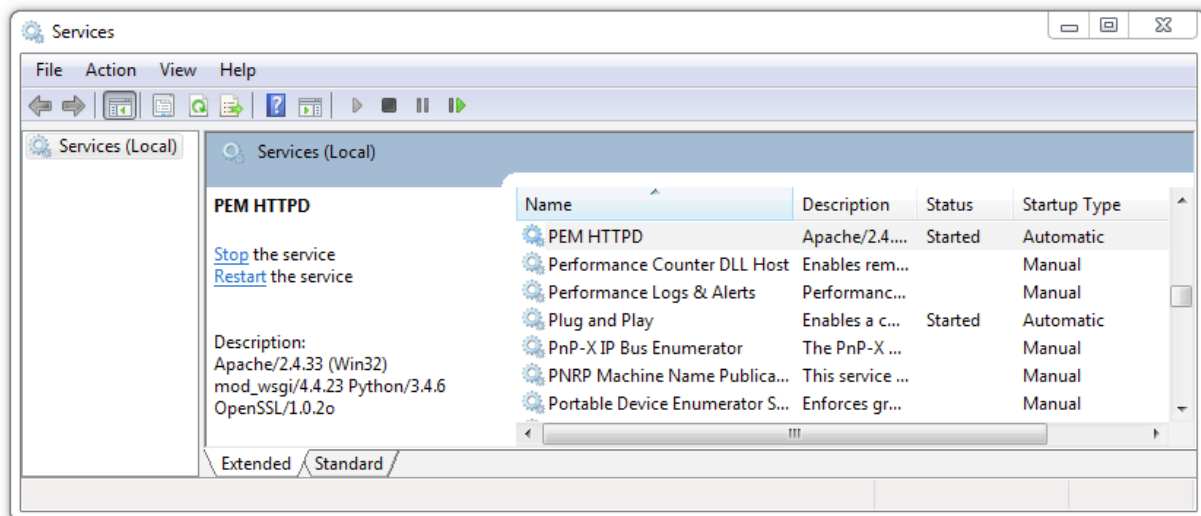


図 4 The PEM HTTPD Windows service

Status 列には、サーバーの現在の状態が表示されます。サービスが実行されていない場合は、Start リンクをクリックして PEM HTTPD を開始します。

4.6 pg_hba.conf ファイルの変更

pg_hba.conf ファイルのエントリは、ネットワークの認証と承認を制御します。PEM サーバーホスト上の pg_hba.conf ファイルは、PEM サーバーと PEM-HTTPD、PEM エージェント、および監視対象サーバー間の接続を許可する必要があります。

PEM サーバーのインストールプロセス中に、PEM によって監視されるホストの IP アドレスと接続情報の入力を求められます。この情報は、PEM バックエンドデータベースの pg_hba.conf ファイルの先頭に追加されます。

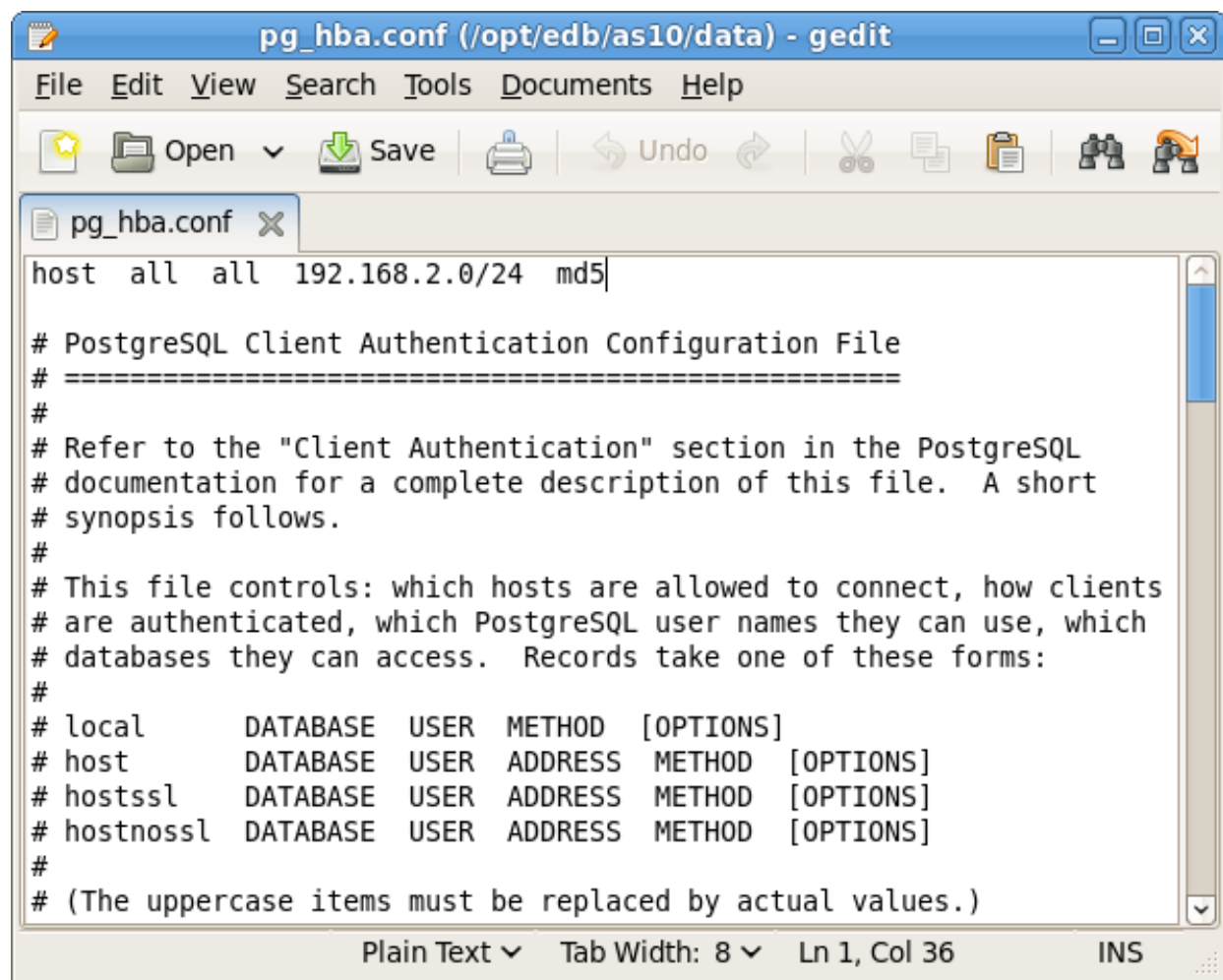


図 5 PEM entries in the pg_hba.conf file

PEM サーバーと他のコンポーネント間の接続を許可するには、pg_hba.conf ファイルを手動で変更する必要がある場合もあります。たとえば、PEM-HTTPD インストールが PEM サーバーと同じホスト上にない場合、PEM サーバーホスト上の pg_hba.conf ファイルを変更して、PEM-HTTPD がサー

バーに接続できるようにする必要があります。

デフォルトでは、`pg_hba.conf` ファイルは Postgres インストールの下のデータディレクトリにあります。たとえば、Advanced Server 10 ホストでは、`pg_hba.conf` のデフォルトの場所は次のとおりです。

```
/opt/edb/as10/data/pg_hba.conf
```

選択したエディターで `pg_hba.conf` ファイルを変更できます。ファイルを変更した後、サーバーを再起動して変更を有効にします。

次の例は、`postgres` という名前のユーザーから `pg_hba.conf` ファイルが存在するホストの `postgres` データベースへの `md5` パスワード認証接続を許可する `pg_hba.conf` エントリを示しています。接続は 192.168.10.102 の IP アドレスから来ています：

#	TYPE	DATABASE	USER	CIDR-ADDRESS	METHOD
#	IPv4	local	connections:		
	host	postgres	postgres	192.168.10.102/32	md5

ネットワークホストのアドレスまたはネットワークアドレス範囲を指定できます。たとえば、アドレスが 192.168.10.23、192.168.10.76、192.168.10.184 のサーバーからの接続を許可する場合は、**CIDR-ADDRESS** に 192.168.10.0/24 を入力して、すべてのサーバーからの接続を許可しますそのネットワーク内のホスト：

#	TYPE	DATABASE	USER	CIDR-ADDRESS	METHOD
#	IPv4	local	connections:		
	host	postgres	all	192.168.10.0/24	md5

`pg_hba.conf` ファイルエントリのフォーマットの詳細については、次の PostgreSQL コアドキュメントを参照してください。

<http://www.postgresql.org/docs/10/static/auth-pg-hba-conf.html>

PEM で Postgres サーバーに接続する前に、両方のサーバーの `pg_hba.conf` ファイルが接続を許可していることを確認する必要があります。

データベースサーバーへの接続時にこのエラーを受け取った場合、接続を許可するエントリを追加して `pg_hba.conf` ファイルを変更します。

4.7 データベースとオブジェクトの作成と保守

Postgres サーバーの各インスタンスは、1 つ以上のデータベースを管理します。各ユーザーは、データベースに含まれる情報にアクセスする前に、データベースに接続するための認証情報を提供する必要があります。PEM クライアントは、データベース、およびデータベースを構成するさまざまなオブジェクト（テーブル、インデックス、ストアドプロシージャなど）をすべて作成および管理できるダイアログを提供します。

PEM でのデータベースの作成は簡単です。管理対象サーバーの Databases ノードを右クリックし、Create メニューから Database… を選択するだけです。データベースを定義したら、新しいデータベース内にオブジェクトを作成できます。

たとえば、新しいテーブルを作成するには、Tables ノードを右クリックし、Create メニューから Table… を選択します。New Table ダイアログが開いたら、新しいテーブルの属性を指定します。

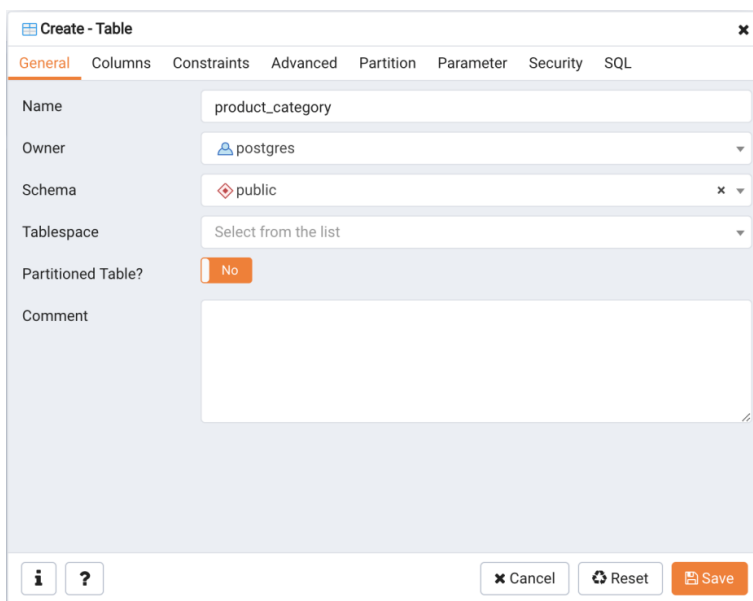


図 6 Use PEM's dialogs to create and manage database objects

PEM は、他のデータベースオブジェクトの作成と管理のための同様のダイアログを提供します。

- テーブル
- インデックス
- ストアドプロシージャ
- 機能

- 引き金
- 景色
- 制約など

各オブジェクトタイプはツリーコントロールに表示されます。オブジェクトタイプに対応するノードを右クリックして Create メニューにアクセスし、新しいオブジェクトを作成するか、名前付きノードのコンテキストメニューから Properties を選択して、強調表示されたオブジェクトの管理タスクを実行します。

4.8 PEM 認証の管理

Postgres は多くの認証方法をサポートしています:

- 安全なパスワード (md5)
- GSSAPI
- SSPI
- Kerberos
- アイデント
- LDAP
- RADIUS
- 証明書 (SSL)
- PAM

Postgres (および PEM) 認証は `pg_hba.conf` 構成ファイルによって制御されます。構成ファイル内のエントリは、特定のデータベースに接続できるユーザーと、そのユーザーが接続を許可される前に必要な認証の種類を指定します。

`postgres` という名前のユーザーがセキュアパスワード (md5) 認証接続を使用してローカルホスト (127.0.0.1/32) からすべてのデータベースに接続できるようにする `pg_hba.conf` ファイルの一般的なエントリは、次の形式を取ります。

```
host all postgres 127.0.0.1/32 md5
```

システムの構成によっては、PEM エージェントがサーバーへの接続に使用するユーザーアカウントのパスワードファイルを作成し、エージェントがサーバーの認証要求に適切に応答できるようにする必要があります。1safepwd のパスワードを持つ `postgres` という名前のユーザーのパスワードファイルのエントリは、次の形式になります。

```
localhost:5432:*:postgres:1safepwd
```

パスワードファイルは通常、Linux システムでは `~root/.pgpass`、Windows では `%APPDATA%/postgresql/pgpass.conf` という名前です。パスワードファイルの設定の詳細については、次の EnterpriseDB Web サイトにアクセスしてください。

<http://www.postgresql.org/docs/10/static/libpq-pgpass.html>

Postgres でサポートされている認証方法の詳細については、次の PostgreSQL コアドキュメントを参照してください。

<http://www.postgresql.org/docs/10/static/client-authentication.html>

4.9 プロキシサーバーを使用するための PEM の変更

ネットワーク構成が PEM と EnterpriseDB Web サイト間の直接通信を妨げる場合、次の場合に PEM が使用するプロキシサーバーを構成できます。

- インストールまたは更新に利用可能なパッケージに関する情報で package_catalog テーブルを更新する
- パッケージオプションの読み取り
- パッケージのダウンロード

ネットワークでプロキシサーバーを構成した後、PEM サーバー構成を変更し、プロキシの接続プロパティを指定し、プロキシサーバーを使用するように PEM に指示します。

Server Configuration			
Search by parameter name			
proxy_server	127.0.0.1		
proxy_server_authentication		False	t/f
proxy_server_enabled		False	t/f
proxy_server_password			
proxy_server_port	80		
proxy_server_username			
reminder_notification_interval	24		hours
server_log_retention_time	30		days
show_data_points_on_graph		False	t/f
show_data_tab_on_graph		False	t/f
smtp_authentication		False	t/f
smtp_enabled		True	t/f
smtp_encryption		False	t/f

? Cancel Reset Save

図 7 The PEM Server Configuration dialog

Server Configuration ダイアログにアクセスしてサーバー構成を変更するには、PEM Web インターフェースに接続し、Management メニューから Server Configuration... を選択します。

パラメーター値を変更するには、パラメーターを見つけて、“値”列でパラメーター値を変更します。次の PEM サーバー構成パラメーターを使用して、PEM がプロキシサーバーに接続できるようにする接続の詳細を指定します。

- proxy_server パラメーターを使用して、プロキシサーバーの IP アドレスを指定します。
- プロキシサーバーが接続時に PEM による認証を要求することを示すために、proxy_server_authentication パラメーターにもその値を指定します。認証が必要な

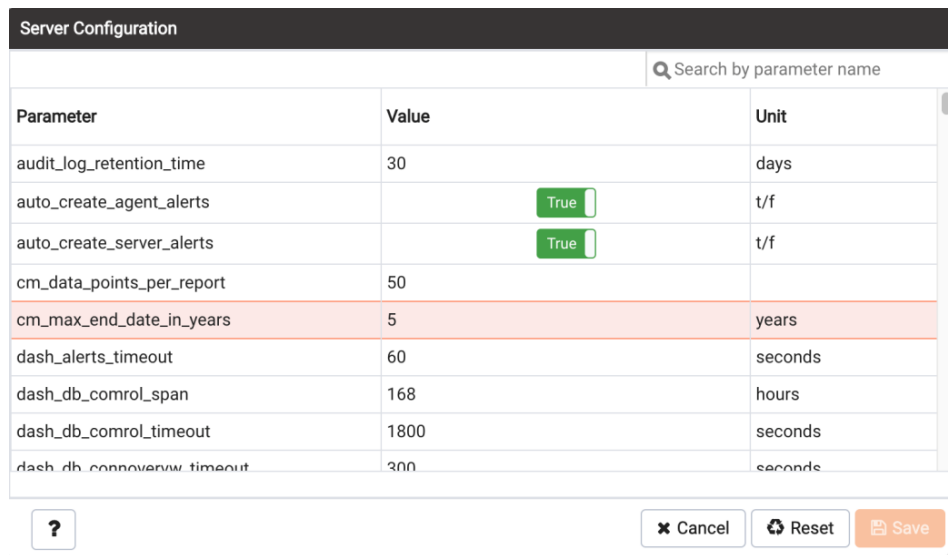
い場合は `f` を指定します。

- パッケージリストを取得するときに **PEM** がプロキシサーバーを使用する必要がある場合は、`proxy_server_enabled` パラメーターに `t` の値を指定します。プロキシサーバーが設定されていない場合は `f` を指定します。
- `proxy_server_username` で指定されたユーザーに関連付けられたパスワードを提供するには、`proxy_server_password` パラメーターを使用します。
- `proxy_server_port` パラメーターでプロキシサーバーのポート番号を指定します。
- `proxy_server_username` パラメーターで、プロキシサーバーで認証するときに使用するユーザー名を指定します。

プロキシサーバーの設定に必要なパラメーターの更新が完了したら、ダイアログを閉じる前に、ダイアログの右上隅にある **Save** アイコンをクリックします。

4.10 PEM サーバー構成の編集

PEM クライアントを使用して、PEM サーバーの構成パラメーターをグラフィカルに管理し、機能を有効にしたり、デフォルト設定を変更したりできます。Server Configuration ダイアログを開くには、Management メニューから Server Configuration… を選択します。



Parameter	Value	Unit
audit_log_retention_time	30	days
auto_create_agent_alerts	☒	t/f
auto_create_server_alerts	☒	t/f
cm_data_points_per_report	50	
cm_max_end_date_in_years	5	years
dash_alerts_timeout	60	seconds
dash_db_comrol_span	168	hours
dash_db_comrol_timeout	1800	seconds
dash_db_connoverw_timeout	300	seconds

? Cancel Reset Save

図 8 サーバー構成ダイアログ

パラメーター値を変更するには、パラメーター名の右側の Value フィールドに表示される内容を編集します。Save ボタンをクリックして変更を保存するか、Close ボタンをクリックして変更を適用せずにダイアログを終了します。Reset ボタンを使用して、パラメーターを元の値に戻します。

4.11 セキュリティの管理

PEM は、Postgres の役割とサーバーを管理するグラフィカルな方法を提供します。

4.11.1 ログインロール

PEM サーバーに接続する場合、PEM サーバーがデータを保存するデータベースへのアクセスを許可するロール資格情報を提供する必要があります。デフォルトでは、サーバーへの最初の接続には postgres スーパーユーザーアカウントが使用されますが、接続ユーザーごとに個別のロールを作成することを強くお勧めします（セキュリティと監査の両方の目的のため）。ロールを作成するには、PEM クエリツール、PEM Web インターフェイス Create - Login/Group Role ダイアログ、またはコマンドラインクライアント（psql など）を使用できます。

Create - Login/Group Role ダイアログを使用してロールを作成するには、PEM ツリーコントロールでロールが存在するサーバーのノードを展開し、コンテキストメニューにアクセスするために Login/Group Roles ノードを右クリックします。次に、Create メニューから Login/Group Role... を選択します。

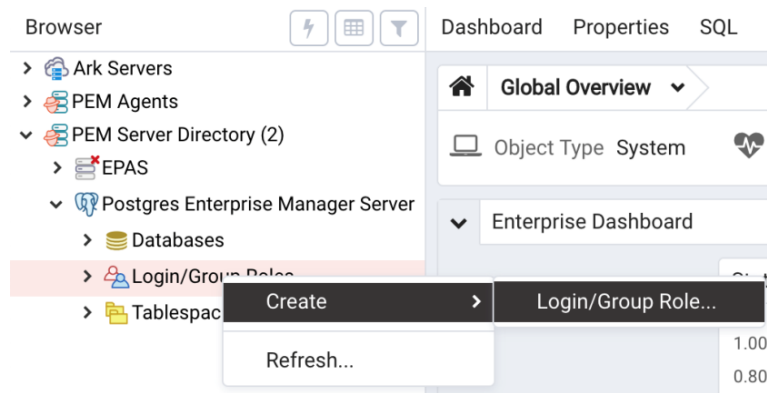
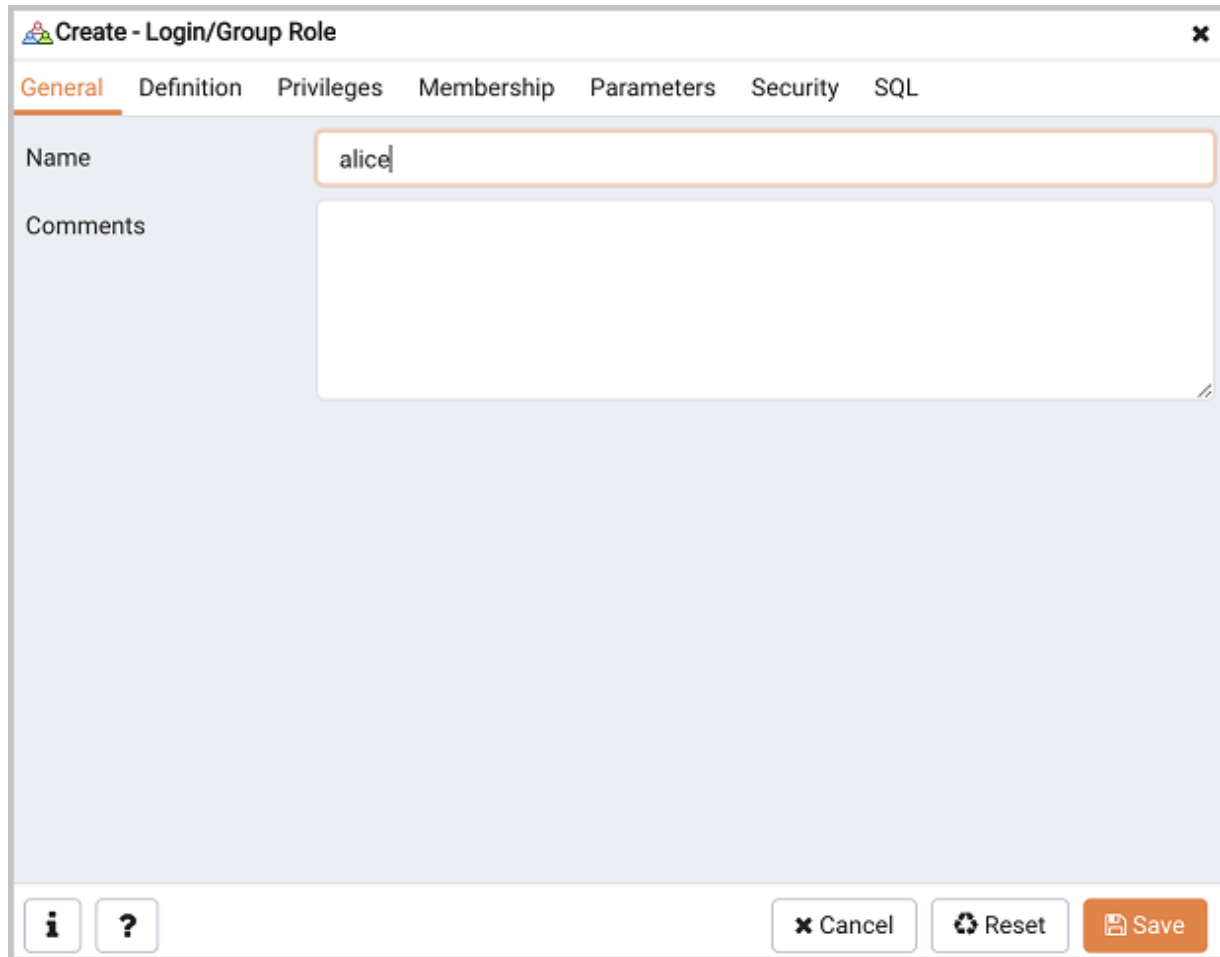


図9 ログインロールノードのコンテキストメニュー

Create - Login/Group ロールダイアログのタブのフィールドを使用して、ロールを定義します。ブラウザタブに PEM オンラインヘルプを表示するには、ダイアログの左下隅にあるヘルプ（？）ボタンをクリックします。

新しいロールの定義が終了したら、Save をクリックしてロールを作成します。



The screenshot shows a window titled "Create - Login/Group Role" with a close button (X) in the top right corner. Below the title bar is a tabbed interface with the following tabs: "General" (selected), "Definition", "Privileges", "Membership", "Parameters", "Security", and "SQL". The "General" tab contains two main input areas: a "Name" field with the text "alice" and a "Comments" text area. At the bottom of the window, there are three buttons: "Cancel" (with a close icon), "Reset" (with a circular arrow icon), and "Save" (with a floppy disk icon). On the far left of the bottom bar, there are two small icons: an information icon (i) and a help icon (?).

図 10 ログインロールダイアログ

既存のログインロールのプロパティを変更するには、ツリーコントロールでログインロールの名前を右クリックし、コンテキストメニューから `Properties` を選択します。ログインロールを削除するには、ロールの名前を右クリックして、コンテキストメニューから `Delete/Drop` を選択します。

ロールの作成と管理の詳細については、PostgreSQL オンラインドキュメントを参照してください。

<http://www.postgresql.org/docs/10/static/sql-createrole.html>

4.11.2 グループの役割

グループロールは、システム権限（データベースの作成など）およびオブジェクト権限（特定のテーブルへのデータの挿入など）を分配するために使用されるコンテナとして機能できます。グループロールの主な目的は、DBA にとってシステムとオブジェクトのパーミッションの一括管理をはるかに簡単にすることです。多くの異なるログインアカウントにわたって個別に特権を割り当てたり変更したりするのではなく、単一のロールの特権を割り当てたり変更したりして、そのロールを一度に多くのログインロールに付与できます。

グループロールを作成および管理するには、Group Roles ノード（PEM ツリーコントロールで登録されている各サーバーの名前の下にあります）を使用します。コンテキストメニューのオプションを使用すると、新しいロールを作成したり、既存のロールのプロパティを変更したりできるダイアログにアクセスできます。ロールの作成に関する詳細については、次を参照してください。

<http://www.postgresql.org/docs/10/static/sql-createrole.html>

4.11.3 PEM の事前定義されたロールを使用して PEM 機能へのアクセスを管理する

Login/Group Role ダイアログを使用して、制限された権限を持つロールが監査マネージャー、キャッシュティマネージャー、SQL プロファイラーなどの PEM 機能にアクセスできるようにすることができます。PEM の事前定義されたロールにより、PEM 機能にアクセスできます。これらのロールのメンバーシップが割り当てられたロールは、関連する機能にアクセスできます。

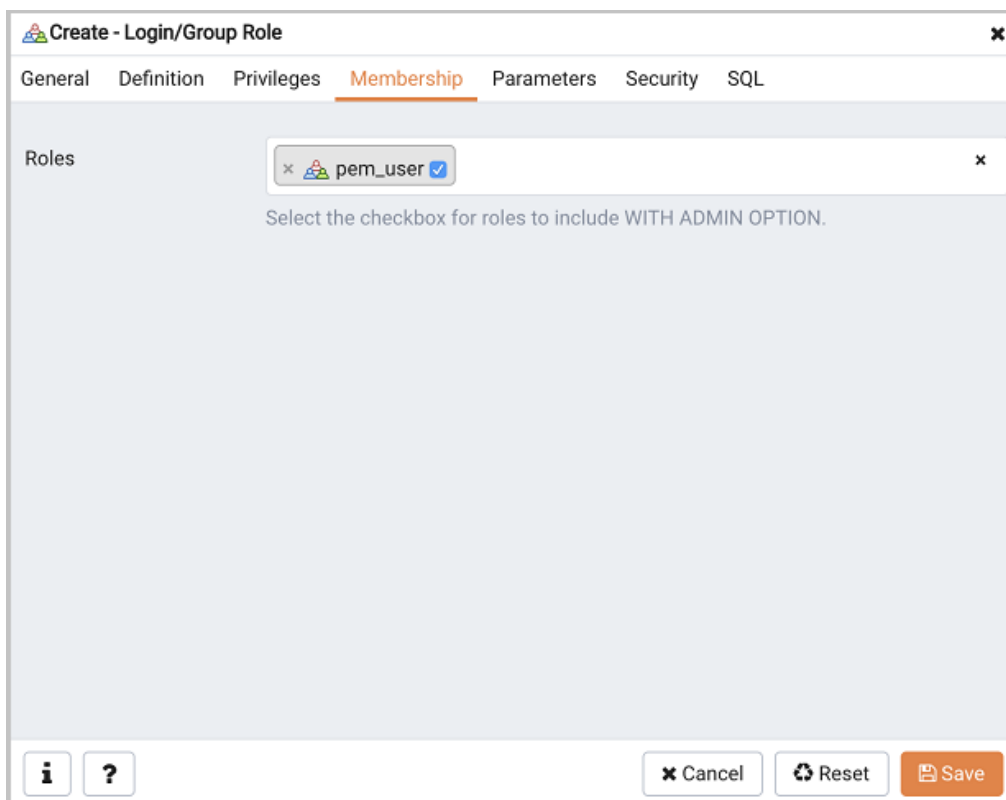


図 11 The Membership tab

ユーザーを定義するとき、Membership タブを使用して、新しいユーザーがメンバーになっているロールを指定します。新しいユーザーは、メンバーである各ロールに関連付けられた特権を共有します。ユーザーが PEM 拡張機能にアクセスするには、ロールは pem_user ロールと、機能へのアクセスを許可する事前定義されたロールのメンバーである必要があります。Roles フィールドを使用して、ドロップダウンリストから事前定義されたロール名を選択します。

SQL タブには、Save をクリックしたときにサーバーが実行する SQL コマンドが表示されます。

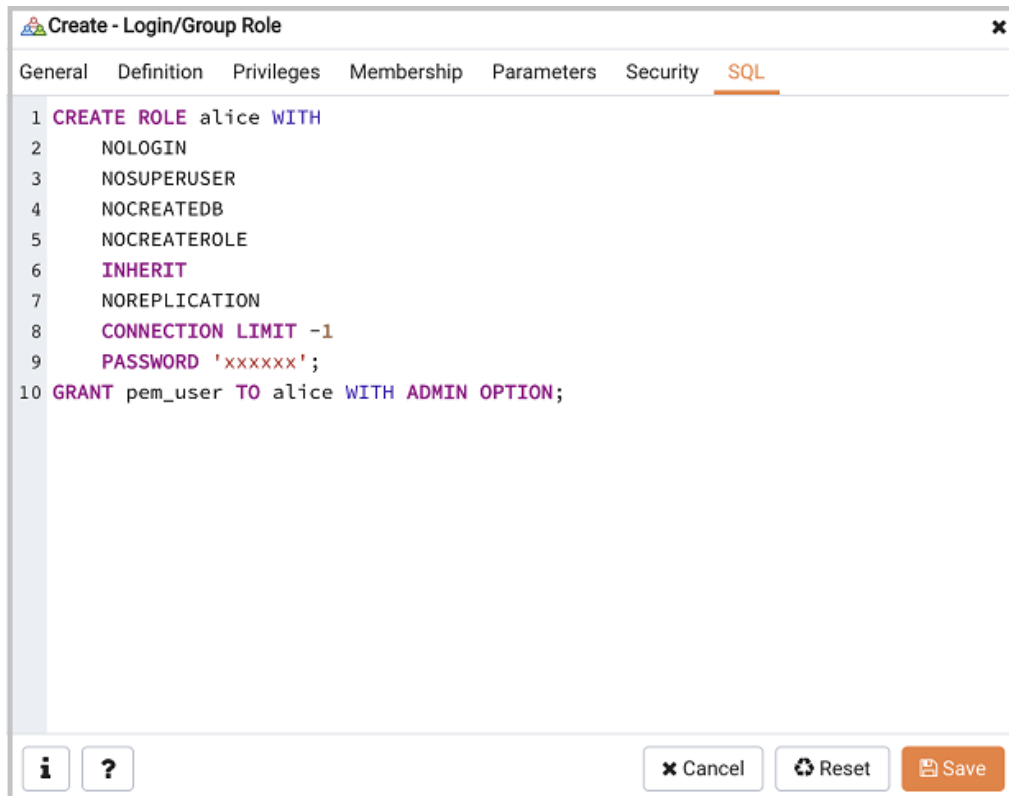


図 12 The SQL tab

上記の例では、Audit Manager にアクセスできる `acctg_clerk` という名前のログインロールを作成します。ロールは、いつでもサーバーに無制限に接続できます。

PEM の事前定義されたロールを使用して、以下の表にリストされている機能へのアクセスを許可できます。

価値	親の役割	説明
pem_super_admin		Postgres Enterprise Manager ですべてを管理/設定する役割。
pem_admin	pem_super_admin	すべての表示可能なエージェント/サーバー、および監視対象オブジェクトの管理/管理/構成の役割。
pem_config	pem_admin	Postgres Enterprise Manager の構成管理の役割。
pem_component	pem_admin	すべてのウィザード/ダイアログベースのコンポーネントを実行/実行する役割。
pem_rest_api	pem_admin	REST API にアクセスする役割。
pem_server_service_manager	pem_admin	監視対象データベースサーバーの再起動/再読み込みを許可するロール（サーバー ID が提供されている場合）。
pem_manage_schedule_task	pem_admin	スケジュールタスクを構成する役割。
pem_manage_alert	pem_admin	アラートとそのテンプレートを管理/構成するためのロール。
pem_config_alert	pem_config、pem_manage_alert	監視対象オブジェクトのアラートを設定するためのロール。
pem_manage_probe	pem_admin	カスタムプローブを作成、更新、削除し、カスタムプローブ設定を変更する役割。
pem_config_probe	pem_config、pem_manage_probe	表示されているすべての監視対象オブジェクトでのプローブ設定の役割（履歴保持、実行頻度、プローブの有効化/無効化）。
pem_database_server_registration	pem_admin	データベースサーバーを登録するロール。
pem_comp_postgres_expert	pem_component	Postgres Expert を実行する役割。
pem_comp_auto_discovery	pem_component	データベースサーバーダイアログの自動検出を実行するロール。
pem_comp_log_analysis_expert	pem_component	ログ分析エキスパートを実行する役割。
pem_comp_sqlprofiler	pem_component	SQL プロファイラーを実行するロール。
pem_manage_efm	pem_admin	フェールオーバーマネージャーの機能を管理する役割。
pem_comp_capacity_manager	pem_component	Capacity Manager を実行する役割。
pem_comp_log_manager	pem_component	Log Manager を実行する役割。

4.11.4 チームロールを使用する

PEM による監視のためにサーバーを登録するとき、サーバーに関連付けられる* **Team** *を指定できます。チームは、1 つまたは複数の監視対象サーバーへのアクセスを、限られた役割メンバーのグループに許可または制限するために使用できるグループ役割です。PEM クライアントは、指定されたチームを持つサーバーのみを以下のユーザーに表示します。

- チームの役割のメンバー
- サーバーを作成した役割
- PEM サーバーでのスーパーユーザー特権を持つロール。

チームの役割を作成するには、PEM ツリーコントロールで役割を配置するサーバーのノードを展開し、Login/Group Roles ノードを右クリックしてコンテキストメニューにアクセスします。次に、Create メニューから Login/Group Role… を選択します。Create - Login/Group Role ダイアログが開いたら、提供されたフィールドを使用してチームロールのプロパティを指定します。

4.11.5 オブジェクト権限

データベースオブジェクトにアクセス、実行、または作成する前に、ロールに十分な特権を付与する必要があります。PEM では、PEM クライアントのグラフィカルインターフェイスを使用して、グループロールまたはログインアカウントにオブジェクト権限を割り当て（GRANT）および削除（REVOKE）できます。

オブジェクト権限は、特定のオブジェクトごとにグラフィカルオブジェクトエディターを介して管理されます。たとえば、データベーステーブルにアクセスするための特権を割り当てるには、ツリーコントロールでテーブル名を右クリックし、コンテキストメニューから [プロパティ] オプションを選択します。[特権] タブに表示されるオプションを使用して、テーブルに特権を割り当てます。

PEM クライアントには、Grant Wizard（Tools メニューからアクセス）が含まれているため、一度に多くのオブジェクト権限を管理できます。

4.12 ジョブ通知の管理

システムで生成されたジョブ（スケジュールされたタスクの下にリストされている）またはカスタム定義のエージェントジョブの成功または失敗時に SMTP トラップを送信するように、PEM コンソールで設定を構成できます。カスタム定義のエージェントジョブの詳細については、“**PEM スケジュールジョブの作成**”を参照してください。これらの電子メール通知設定は、指定されたユーザーグループに電子メール通知を送信するために、次の3つのレベルで（優先順に）構成できます。

- 職務レベル
- エージェントレベル
- PEM サーバーレベル（デフォルトレベル）

4.12.1 ジョブレベルでのジョブ通知の構成

次のいずれかの方法で、カスタム定義のエージェントジョブに対してのみ、ジョブレベルで電子メール通知設定を構成できます。

- 新しいエージェントジョブの場合、ジョブ自体の作成中に、[エージェントジョブの作成] ウィザードの [通知] タブでメール通知設定を構成できます。
- 既存のカスタム定義ジョブの場合、ジョブのプロパティを編集して通知設定を構成できます。

Create - Agent Job

General Steps Schedules **Notifications** SQL

Send the notifications **ALWAYS**

Determines when to send a notification for the job:

ON FAILURE :
Send a notification on the failure/interruption of the job.

ALWAYS :
Send a notification on the completion of the job regardless of the result.

NEVER :
Do not send a notification for the job.

DEFAULT :
Use the agent/system level job notification configuration to determine whether, and when to send the notification.

Email group **<Default>**

Select the email-group to get the job/scheduled-task notification on completion.

Cancel Reset Save

図 13 ジョブ通知設定:ジョブレベル

[通知] タブのフィールドを使用して、ジョブレベルで電子メール通知設定を構成します。

- [通知の送信] フィールドを使用して、電子メール通知をいつ送信するかを指定します。
- [電子メールグループ] フィールドを使用して、電子メール通知を受信する電子メールグループを指定します。

4.12.2 エージェントレベルでのジョブ通知の構成

ツリービューでエージェントを選択し、右クリックして [プロパティ] を選択します。[プロパティ] ダイアログで、[ジョブ通知] タブを選択します。



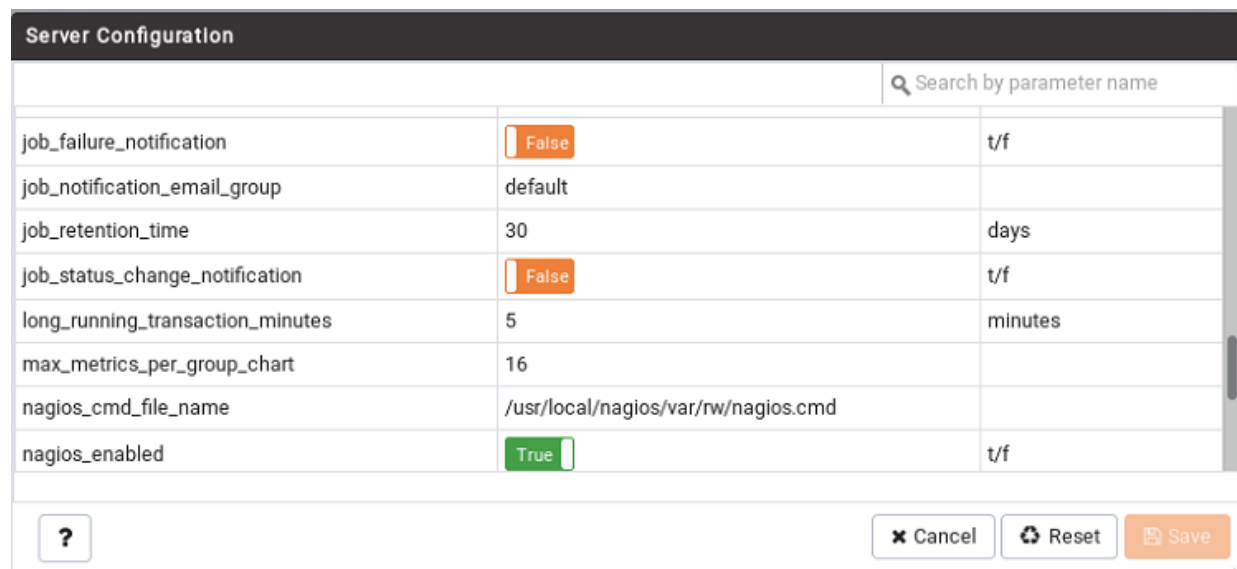
図 14 ジョブ通知設定:エージェントレベル

[ジョブ通知] タブのフィールドを使用して、エージェントレベルで電子メール通知設定を構成します。

- デフォルト設定の上書き？*スイッチを使用して、エージェントレベルのジョブ通知設定でデフォルトのジョブ通知設定を上書きするかどうかを指定します。このスイッチで [はい] を選択すると、このダイアログの残りの設定を使用して、ジョブ通知をいつ誰に送信するかを定義できます。このダイアログの残りの設定は、Override default configuration？*スイッチを有効にした場合にのみ機能することに注意してください。
- *ジョブ完了時に電子メール？*スイッチを使用して、ジョブ完了が成功したときにジョブ通知を送信するかどうかを指定します。
- *ジョブの失敗時に電子メール？*スイッチを使用して、ジョブの失敗時にジョブ通知を送信するかどうかを指定します。
- *メールグループ*フィールドを使用して、ジョブ通知の送信先のメールグループを指定します。

4.12.3 サーバーレベルでのジョブ通知の構成

*サーバー構成*ダイアログを使用して、PEM サーバーレベルでの電子メール通知構成に関する情報を提供できます。[サーバー構成] ダイアログを開くには、PEM クライアントの [管理] メニューから [サーバー構成...] を選択します。



The screenshot shows the 'Server Configuration' dialog box with a search bar and a table of parameters. The parameters are:

Parameter Name	Value	Unit/Type
job_failure_notification	False	t/f
job_notification_email_group	default	
job_retention_time	30	days
job_status_change_notification	False	t/f
long_running_transaction_minutes	5	minutes
max_metrics_per_group_chart	16	
nagios_cmd_file_name	/usr/local/nagios/var/rw/nagios.cmd	
nagios_enabled	True	t/f

At the bottom, there are buttons for '?', 'Cancel', 'Reset', and 'Save'.

図 15 ジョブ通知設定:サーバーレベル

4 つのサーバー構成パラメーターは、PEM サーバーレベルでのジョブ通知設定に関する情報を指定します。

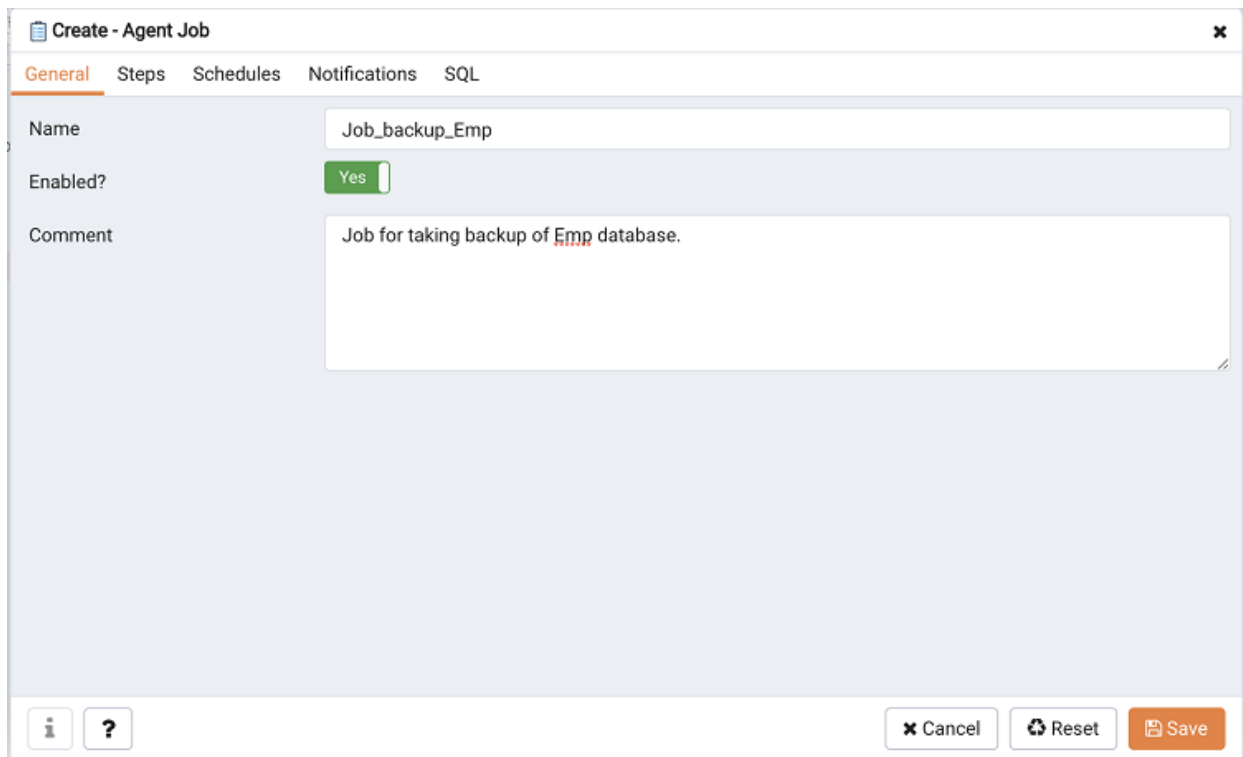
- Use the *job_failure_notification* switch to specify if you want to send email notification after each job failure.
- Use the *job_notification_email_group* parameter to specify the email group that should receive the email notification.
- Use the *job_retention_time* parameter to specify the number of days that non-recurring scheduled tasks should be retained in the system.
- Use the *job_status_change_notification* switch to specify if you want to send email notification after each job status change, irrespective of its status being a failure, success, or interrupted.

4.13 PEM スケジュール済みジョブの管理

PEM スケジュールジョブを作成して、指定された順序で一連のカスタム定義のステップを実行できます。これらの手順には、エージェントにバインドされているサーバーで実行できる SQL コードまたはバッチ/シェルスクリプトが含まれている場合があります。これらのジョブは、ビジネス要件に合わせてスケジュールできます。たとえば、特定のデータベースサーバーのバックアップを取得するジョブを作成し、毎月特定の日時に実行するようにスケジュールできます。

PEM スケジュールジョブを作成または管理するには、PEM ツリーコントロールを使用して、ジョブを作成する PEM エージェントを参照します。ツリーコントロールに [ジョブ] ノードが表示され、その下に現在定義されているジョブが表示されます。新しいジョブを追加するには、Jobs ノードを右クリックし、コンテキストメニューから Create Job... を選択します。

Create - Agent Job ダイアログが開いたら、Create - Agent Job ダイアログのタブを使用して、PEM スケジュールジョブを構成するステップとスケジュールを定義します。



The screenshot shows a dialog box titled "Create - Agent Job" with a close button (X) in the top right corner. Below the title bar are five tabs: "General" (selected), "Steps", "Schedules", "Notifications", and "SQL". The "General" tab contains three fields: "Name" with the value "Job_backup_Emp", "Enabled?" with a checked "Yes" button, and "Comment" with the text "Job for taking backup of Emp database." At the bottom of the dialog are three buttons: "Cancel", "Reset", and "Save".

図 16 PEM scheduled job dialog create schedule

General タブのフィールドを使用して、ジョブに関する一般情報を提供します。

- Name フィールドにジョブの名前を入力します。

- ジョブを有効にするには Enabled スイッチを Yes の位置に移動し、ジョブを無効にするには No の位置に移動します。
- ジョブに関するメモを保存するには、Comment フィールドを使用します。

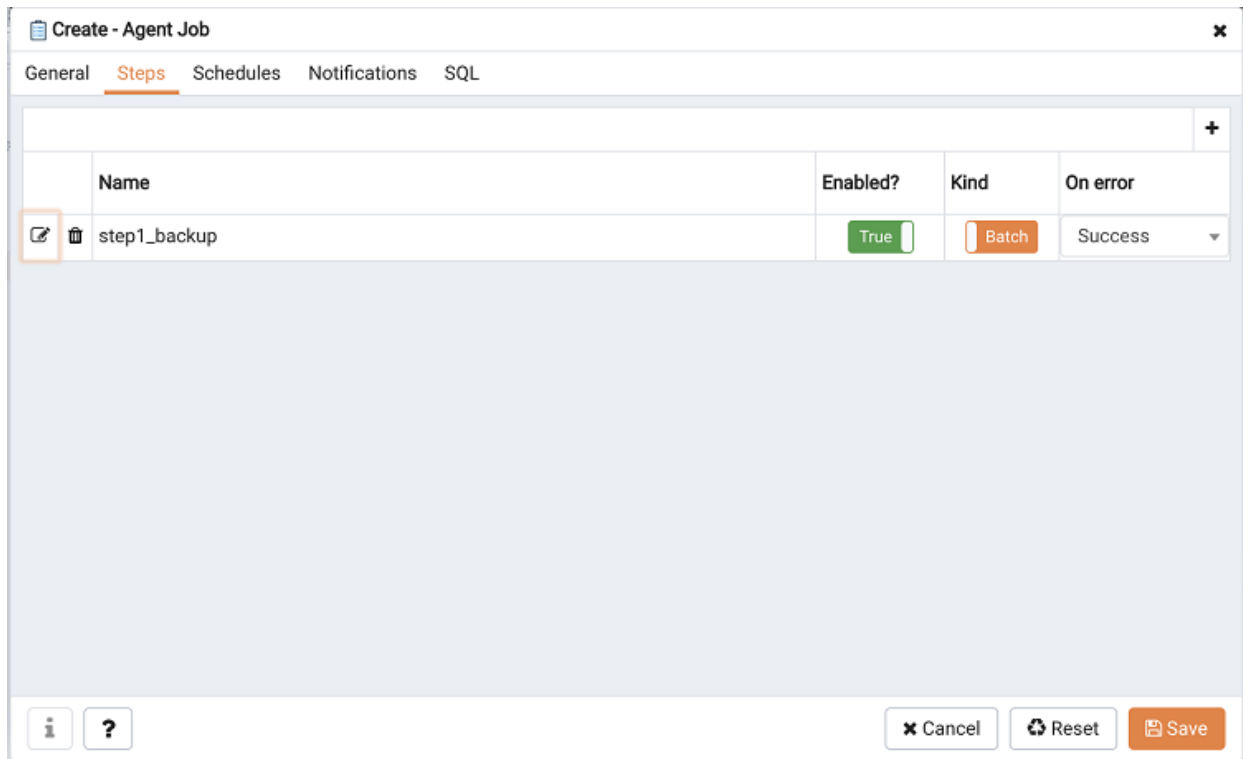


図 17 PEM scheduled job dialog add steps

ジョブが実行するステップを定義および管理するには、Steps タブを使用します。[追加] アイコン (+) をクリックして、新しいステップを追加します。次に、ヘッダーの左側にある作成アイコンをクリックして、ステップ定義ダイアログを開きます。

Name	Enabled?	Kind	On error
step1_backup	True	Batch	Success

General Code

Name

step1_backup

Enabled?

Yes

Kind

Batch

On error

Success

Server

Database

Comment

Cancel

Reset

Save

図 18 PEM scheduled job steps definition

ステップ定義ダイアログのフィールドを使用して、ステップを定義します。

- Name フィールドにステップの名前を入力します; 手順は名前の英数字順に実行されることに注意してください。
- Enabled スイッチを使用して、ジョブの実行時にステップを含める (True) か、ステップを無効にします (False)。
- Kind スイッチを使用して、ジョブステップが SQL コード (SQL) を呼び出すか、バッチスクリプト (Batch) を呼び出すかを示します。
 - SQL を選択した場合は、Code タブを使用してステップの SQL コードを提供します。
 - Batch を選択した場合、Code タブを使用して、ステップ中に実行されるバッチスクリプトを提供します。
- On error ドロップダウンを使用して、ステップの実行中にエラーが発生した場合の pgAgent の動作を指定します。以下から選択してください:
 - 失敗-このステップの処理中にエラーが発生した場合、ジョブを停止します。

- 成功-ステップを正常に完了したとしてマークし、続行します。
- 無視-エラーを無視して続行します。
- Kind スイッチの入力として **SQL** を選択した場合、以下の追加情報を提供します。
 - Server フィールドを使用して、**PEM** スケジュールジョブを作成しているエージェントにバインドされているサーバーを指定します。
 - Database フィールドを使用して、選択したサーバーに関連付けられているデータベースを指定します。
- ステップに関するコメントを提供するには、Comment フィールドを使用します。

The screenshot shows a dialog box titled "Job_backup_Emp" with a close button (X) in the top right corner. Below the title bar are four tabs: "General", "Steps" (which is selected and highlighted in orange), "Schedules", "Notifications", and "SQL".

Under the "Steps" tab, there is a table with the following columns: "Name", "Enabled?", "Kind", and "On error". There is a plus sign (+) button to the right of the table header. The table contains one row with the following data:

Name	Enabled?	Kind	On error
step1_backup	True	Batch	Success

Below the table, there are two sub-tabs: "General" and "Code" (which is selected and highlighted in orange). Under the "Code" sub-tab, there is a text area labeled "Script" containing the following code:

```
1 #!/bin/bash
2
3 # Find the current directory path
4 directory=`pwd`
5
6 echo "Directory is path is '$directory'"
7
```

At the bottom of the dialog box, there are three buttons: "Cancel", "Reset", and "Save".

図 19 PEM scheduled job steps definition code

- ステップ定義ダイアログの Code タブの状況依存フィールドを使用して、ステップ中に実行される SQL コードまたはバッチスクリプトを提供します。
 - ステップが SQL コードを呼び出す場合、SQL query フィールドに 1 つ以上の SQL ステートメントを指定します。
 - ステップがバッチスクリプトを呼び出す場合、Code フィールドにスクリプトを指定します。Windows サーバーで実行している場合は、標準のバッチファイル構文を使用する必要があります。

があります。Linux サーバーで実行する場合、最初の行に適切なインタープリターが指定されていれば（たとえば、`*#! /bin / sh *`）、任意のシェルスクリプトを使用できます。定義済みのインラインコードに加えて、ファイルシステム上のバッチスクリプト、シェルスクリプト、または SQL ファイルのパスを指定することもできます。

ステップに必要なすべての情報を提供した後、Save ボタンをクリックして、ステップ定義ダイアログを保存して閉じます。

追加アイコン (+) をクリックして各ステップを追加するか、Schedules タブを選択してジョブスケジュールを定義します。

[追加] アイコン (+) をクリックして、ジョブのスケジュールを追加します。次に、ヘッダーの左側にある作成アイコンをクリックして、スケジュール定義ダイアログを開きます。

The screenshot shows the 'Create - Agent Job' dialog box with the 'Schedules' tab selected. At the top, there are tabs for 'General', 'Steps', 'Schedules', 'Notifications', and 'SQL'. Below these is a table of schedules. The table has columns for 'Name', 'Enabled?', 'Start', and 'End'. There is one schedule listed: 'Backup_Emp_schedule1', which is enabled (indicated by a green 'True' button) and has a start time of '2019-07-01 11:54 +05:30' and an end time of '2019-07-02 11:54 +05:30'. Below the table, there is a detailed form for the selected schedule. This form has tabs for 'General', 'Repeat', and 'Exceptions'. The 'General' tab is active, showing fields for 'Name' (Backup_Emp_schedule1), 'Enabled?' (Yes), 'Start' (2019-07-01 11:54 +05:30), 'End' (2019-07-02 11:54 +05:30), and 'Comment'. At the bottom of the dialog, there are buttons for 'Cancel', 'Reset', and 'Save'.

Name	Enabled?	Start	End
Backup_Emp_schedule1	True	2019-07-01 11:54 +05:30	2019-07-02 11:54 +05:30

General Repeat Exceptions

Name: Backup_Emp_schedule1

Enabled?: Yes

Start: 2019-07-01 11:54 +05:30

End: 2019-07-02 11:54 +05:30

Comment:

Buttons: Cancel, Reset, Save

図 20 PEM scheduled job dialog add schedule tab

Schedules definition タブのフィールドを使用して、ジョブを実行する日時を指定します。

- Name フィールドにスケジュールの名前を入力します。
- Use the *Enabled* switch to indicate that pgAgent should use the schedule (Yes) or to disable the schedule (No).

- Start フィールドのカレンダーセレクトを使用して、スケジュールの開始日時を指定します。
- End フィールドのカレンダーセレクトを使用して、スケジュールの終了日時を指定します。
- Comment フィールドを使用して、スケジュールに関するコメントを入力します。

Repeat タブを選択して、スケジュールを実行する日を定義します。

図 21 PEM scheduled job dialog schedule repeat tab

Repeat タブのフィールドを使用して、cron スタイル形式でスケジュールに関する詳細を指定します。ジョブは、Repeat タブで選択された各日付または時間要素で実行されます。

フィールド内をクリックして、そのフィールドの有効な値のリストを開きます。特定の値をクリックして、その値をフィールドの選択値のリストに追加します。フィールドから値をクリアするには、フィールドの右側にある X をクリックします。

- Days ボックス内のフィールドを使用して、ジョブを実行する日を指定します。
 - Week Days フィールドを使用して、ジョブを実行する日を選択します。
 - Month Days フィールドを使用して、ジョブを実行する日数を選択します。日付に関係なく、ジョブを月の最後の日に実行する必要があることを示すには、Last Day を指定し

ます。

- Months フィールドを使用して、ジョブを実行する月を選択します。
- Times ボックス内のフィールドを使用して、ジョブを実行する時間を指定します。
 - Hours フィールドを使用して、ジョブを実行する時間を選択します。
 - Minutes フィールドを使用して、ジョブを実行する分を選択します。

Exceptions タブを選択して、スケジュールが not を実行する日を指定します。

Name	Enabled?	Start	End
Backup_Emp_schedule1	True	2019-07-01 11:54 +05:30	2019-07-02 11:54 +05:30

Date	Time
2019-08-01	00:00

図 22 PEM scheduled job dialog exceptions tab

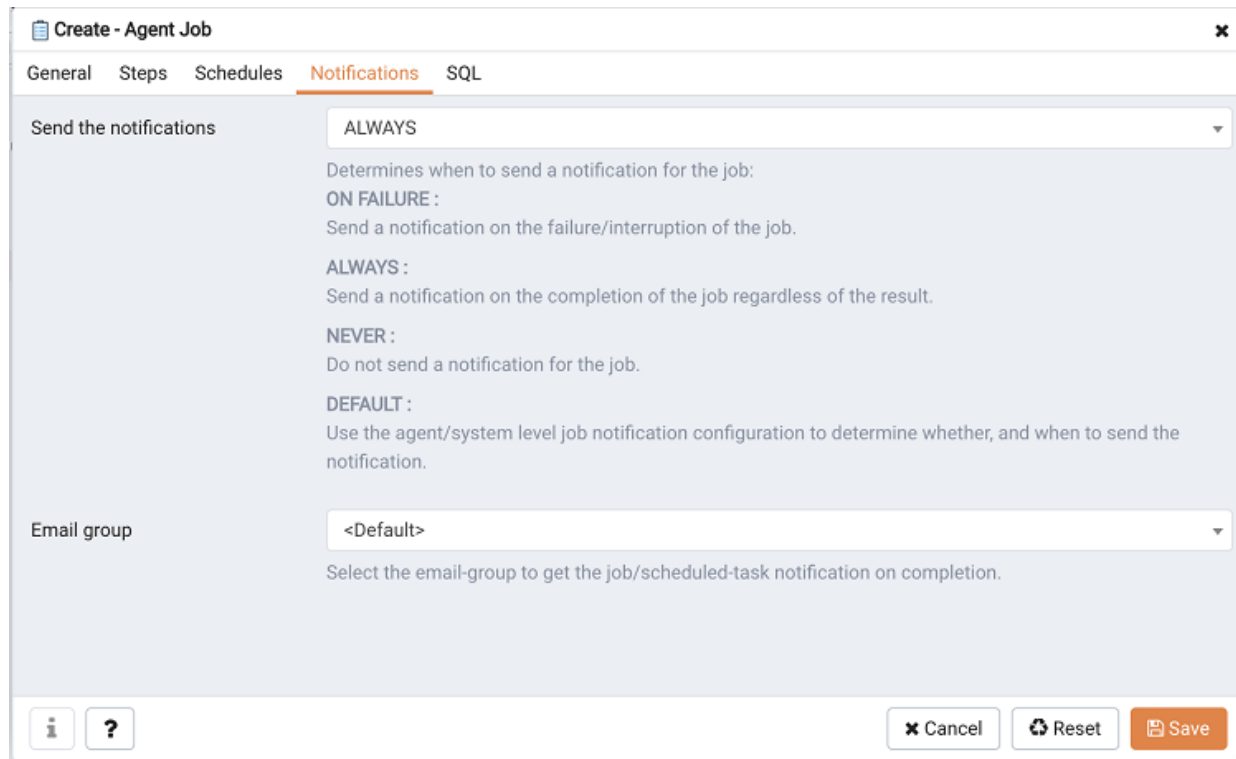
Exceptions タブのフィールドを使用して、ジョブを実行しない日を指定します。たとえば、国民の祝日にジョブを実行しないようにしたい場合があります。

[追加] アイコン (+) をクリックして、例外テーブルに行を追加し、次の操作を行います。

- Date 列内をクリックしてカレンダーセレクトを開き、ジョブが実行されない日付を選択します。Date 列に <Any> を指定して、選択した時刻のどの日にもジョブを実行しないことを示します。

- Time 列内をクリックして時間セレクターを開き、ジョブが実行されない時間を指定します。
Time 列に <Any> を指定して、選択した日のどの時点でもジョブを実行しないことを示します。

ジョブレベルで電子メール通知設定を構成するには、Notifications タブを選択します:



The screenshot shows the 'Create - Agent Job' dialog box with the 'Notifications' tab selected. The 'Send the notifications' dropdown menu is set to 'ALWAYS'. Below this, there is explanatory text: 'Determines when to send a notification for the job:'. Under 'ON FAILURE:', it says 'Send a notification on the failure/interruption of the job.' Under 'ALWAYS:', it says 'Send a notification on the completion of the job regardless of the result.' Under 'NEVER:', it says 'Do not send a notification for the job.' Under 'DEFAULT:', it says 'Use the agent/system level job notification configuration to determine whether, and when to send the notification.' The 'Email group' dropdown menu is set to '<Default>'. Below this, it says 'Select the email-group to get the job/scheduled-task notification on completion.' At the bottom of the dialog are buttons for 'Cancel', 'Reset', and 'Save'.

図 23 PEM scheduled job dialog notifications tab

Notifications タブのフィールドを使用して、ジョブの電子メール通知設定を構成します。

- Send the notifications フィールドを使用して、電子メール通知をいつ送信するかを指定します。
- Email group フィールドを使用して、電子メール通知を受信する電子メールグループを指定します。

スケジュールの定義が完了したら、SQL タブを使用して、ジョブを作成または変更するコードを確認できます。

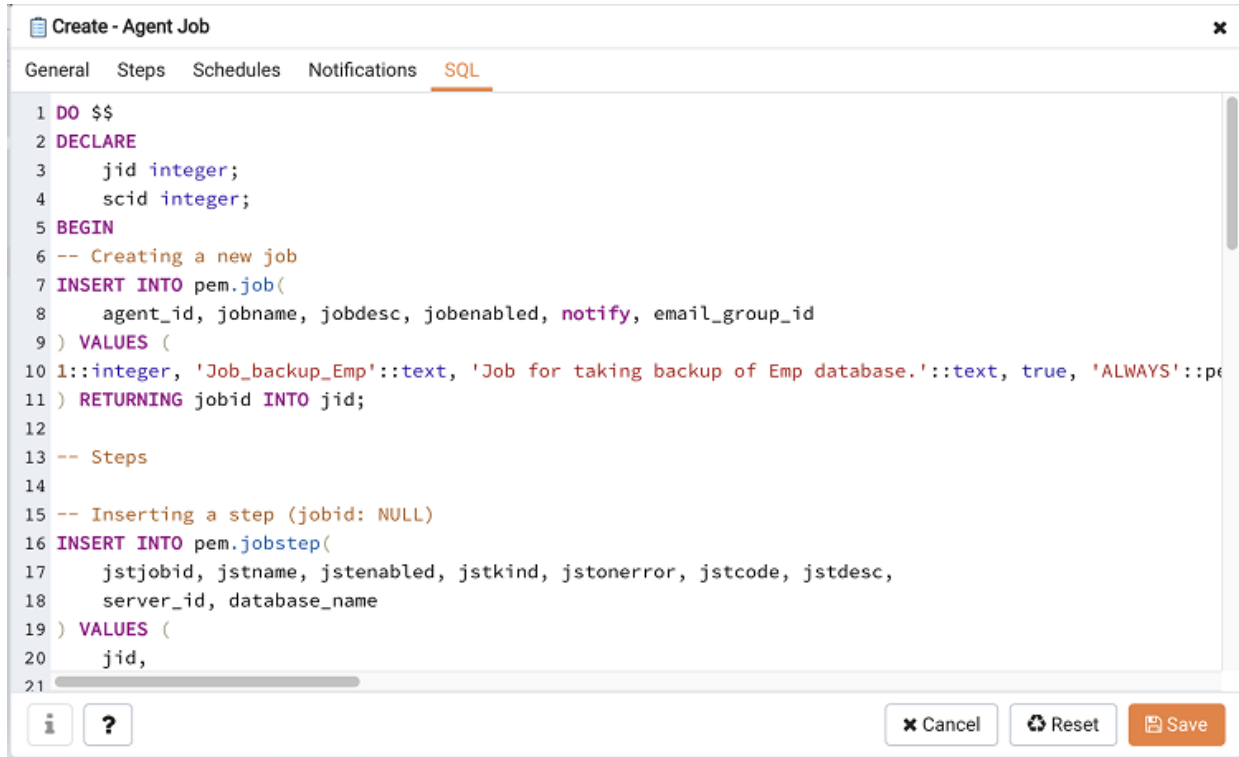


図 24 PEM scheduled job dialog SQL tab

Save ボタンをクリックしてジョブ定義を保存するか、Cancel をクリックして保存せずにジョブを終了します。Reset ボタンを使用して、ダイアログから未保存のエントリを削除します。

ジョブを保存した後、ジョブは、それが定義されたサーバーの PEM ツリーコントロールの Jobs ノードの下にリストされます。PEM コンソールの Properties タブには、選択したジョブの概要が表示され、Jobs タブにはジョブの各実行の詳細が表示されます。既存のジョブを変更するか、ジョブに関する詳細情報を確認するには、ジョブ名を右クリックして、コンテキストメニューから Properties を選択します。

PEM エージェントの管理

以下のセクションでは、PEM エージェントの動作と管理に関する情報を提供します。

5.1 エージェント権限

デフォルトでは、PEM エージェントはオペレーティングシステムホストの `root` 特権とデータベースサーバーのスーパーユーザー特権でインストールされます。これらの権限により、PEM エージェントは、システムの使用状況について監視対象ホストおよびデータベースサーバーで制限のないプローブを呼び出し、情報を取得して PEM サーバーに返すことができます。

PEM エージェントの権限が低下すると、PEM 機能が低下することに注意してください。完全な機能を実現するには、PEM エージェントを `root` として実行する必要があります。PEM エージェントがデータベースサーバーのサービスアカウントで実行されている場合、PEM プローブはレポートの生成に使用される統計情報に完全にアクセスできず、機能はそのアカウントの機能に制限されます。PEM エージェントが別のより低い特権のアカウントで実行されている場合、機能はさらに制限されます。

PEM エージェントのオペレーティングシステム権限を制限すると、PEM プローブの一部が情報を返さず、次の機能が影響を受ける可能性があります。

プローブまたはアクション	オペレーティングシステム	影響を受ける PEM 機能
データとログファイルの分析	Linux / Windows	Postgres Expert は完全な情報にアクセスできません。
セッション情報	Linux	プロセスごとの統計は不完全です。
PG HBA	Linux / Windows	Postgres Expert は完全な情報にアクセスできません。
サービス再起動機能	Linux / Windows	監査ログマネージャー、サーバーログマネージャー、ストリーミングレプリケーション、ログ分析エキスパート、および PEM は、要求された変更を適用できない場合があります。
パッケージ展開	Linux / Windows	PEM はダウンロードしたインストールモジュールを実行できません。
バッチタスク	窓	PEM は、Windows でスケジュールされたバッチジョブを実行できません。
サーバーからデータを収集する (ルートアクセスが必要)	Linux / Windows	スワップ使用率、CPU 使用率、IO 読み取り、IO 書き込みなどの列は、セッションアクティビティダッシュボードで 0 として表示されます。

注釈: 上記のリストは包括的なものではありませんが、制限される機能の種類の概要を提供する必要があります。

PEM エージェントのデータベース特権を制限すると、次の PEM 機能が影響を受ける可能性があります。

プローブ	オペレーティングシステム	影響を受ける PEM 機能
監査ログの収集	Linux / Windows	PEM は PEM データベースから空のデータを受け取ります。
サーバーログ収集	Linux / Windows	PEM はサーバーログ情報を収集できません。
データベース統計	Linux / Windows	データベース/サーバー分析ダッシュボードには、不完全な情報が含まれます。
セッション待機/システム待機	Linux / Windows	セッション/システム待機ダッシュボードには、不完全な情報が含まれます。
ロック情報	Linux / Windows	データベース/サーバー分析ダッシュボードには、不完全な情報が含まれます。
ストリーミングレプリケーション	Linux / Windows	ストリーミングレプリケーションダッシュボードには情報が表示されません。
Slony レプリケーション	Linux / Windows	データベース分析ダッシュボードの Slony 関連のグラフには情報が表示されません。
表領域サイズ	Linux / Windows	サーバー分析ダッシュボードには、完全な情報は表示されません。
xDB レプリケーション	Linux / Windows	PEM は xDB アラートとトラップを送信できません。

プローブが十分な権限を持たないオペレーティングシステムを照会している場合、プローブは permission denied エラーを返すことがあります。

プローブが不十分な特権でデータベースを照会している場合、プローブは permission denied エラーを返すか、返されたデータを空の値として PEM チャートまたはグラフに表示します。

プローブが失敗すると、プローブの名前、プローブが失敗した理由、問題の解決に役立つヒントを含むエントリがログファイルに書き込まれます。

サーバーで発生したプローブ関連のエラーを Probe Log Dashboard で表示したり、PEM ワーカーのログファイルでエラーメッセージを確認したりできます。Linux では、ログファイルのデフォルトの場所は次のとおりです。

```
/var/log/pem/worker.log
```

Windows では、ログ情報は Event Viewer で利用できます。

5.2 エージェント構成

多数のユーザー設定可能なパラメーターとレジストリエントリが PEM エージェントの動作を制御します。PEM エージェントのパラメーター設定を変更して、Streaming Replication ウィザードなどの一部の PEM 機能を有効にする必要がある場合があります。PEM エージェント構成ファイルの値を変更した後、変更を適用するには PEM エージェントを再起動する必要があります。

PEM_MAXCONN パラメーターを除き、PEM 機能を有効にするために変更が必要な場合を除き、EnterpriseDB サポートの専門家に相談することなく、以下にリストされている構成パラメーターまたはレジストリエントリを変更しないことを強くお勧めします。

Linux システムでは、PEM 設定オプションは /opt/edb/pem/agent/etc にある agent.cfg ファイルに保存されます。agent.cfg ファイルには次のエントリが含まれます:

パラメータ名	説明	デフォルト値
pem_host	PEM サーバーの IP アドレスまたはホスト名。	127.0.0.1。
pem_port	PEM サーバーと通信するためにエージェントが接続するデータベースサーバーポート。	ポート 5432。
pem_agent	PEM エージェントに割り当てられた一意の識別子。	最初のエージェントは “1”、2 番目のエージェントは “2” などです。
agent_ssl_key	PEM エージェントのキーファイルへの完全なパス。	/root/.pem/agent.key
agent_ssl_cert	PEM エージェントの証明書ファイルへの完全なパス。	/root/.pem/agent.crt
agent_flag_dir	HA サポートに使用されます。別のサーバーの監視を引き継ぐための要求をチェックするディレクトリパスを指定します。要求は、指定されたフラグディレクトリ内のファイルの形式で行われます。	デフォルトでは設定されていません。
log_level	ログレベルは、PEM ログファイルに書き込まれるイベントのタイプを指定します。	警告
log_location	PEM ワーカーログファイルの場所を指定します。	127.0.0.1。

次のページに続く

表 1 – 前のページからの続き

パラメータ名	説明	デフォルト値
agent_log_location	PEM エージェントのログファイルの場所を指定します。	/var/log/pem/agent.log
long_wait	最初の接続試行が失敗した場合に、PEM エージェントが PEM サーバーへの接続を試行する前に待機する最大時間（秒単位）。	30 秒
short_wait	PEM エージェントがキュー内の次のプローブ（実行待ち）をチェックする前に待機する最小時間（秒単位）。	10 秒
alert_threads	エージェントによって生成されるアラートスレッドの数。	PEM サーバーのホストに常駐するエージェントに対して 1 に設定します。他のすべてのエージェントの場合は 0。
enable_smtp	true に設定すると、SMTP 電子メール機能が有効になります。	PEM サーバーホストの場合は true。他のすべての場合は false。
enable_snmp	true に設定すると、SNMP トラップ機能が有効になります。	PEM サーバーホストの場合は true。他のすべての場合は false。
enable_nagios	true に設定すると、Nagios アラートが有効になります。	PEM サーバーホストの場合は true。他のすべての場合は false。
connect_timeout	エージェントが接続を待機する最大時間（秒単位）（10 進整数文字列）。	デフォルトでは設定されていません。0 に設定すると、エージェントは無期限に待機する必要があります。
allow_server_restart	TRUE に設定すると、エージェントは、監視するデータベースサーバーを再起動できます。このパラメーターの値に応じて、一部の PEM 機能を有効/無効にすることができます。	本当だ

次のページに続く

表 1 – 前のページからの続き

パラメータ名	説明	デフォルト値
allow_package_management	TRUE に設定すると、更新モニターおよびパッケージ管理機能が有効になります。	偽
max_connections	接続スロットルで使用するプローブ接続の最大数。	0（無制限の数）
connection_lifetime	ConnectionLifetime（または connection_lifetime）を使用して、開いているがアイドル状態の接続が保持される最小秒数を指定します。MaxConnections で指定された値に達し、待機中の要求を満たすために（別のデータベースへの）新しい接続が必要な場合、このパラメーターは無視されます。	デフォルトでは、0 に設定されます（エージェントの処理ループの後に接続がアイドル状態になると、接続はドロップされます）。
allow_batch_probes	TRUE に設定すると、ユーザーはカスタムプローブ機能を使用してバッチプローブを作成できます。	偽
heartbeat_connection	TRUE に設定すると、ハートビートの送信に専用の接続が使用されます。	偽
allow_streaming_replication	TRUE に設定すると、ユーザーはストリーミングレプリケーションを設定およびセットアップできます。	偽
batch_script_dir	スクリプトファイル（警告用）が保存されるパスを指定します。	/tmp
connection_custom_setup	監視対象サーバーとの新しい接続が確立されたときに呼び出される SQL コードを提供するために使用します。	デフォルトでは設定されていません。
ca_file	CA 証明書が存在するパスを指定します。	/opt/PEM/agent/share/certs/ca-bundle.crt。

64 ビット Windows システムでは、PEM レジストリエントリは次の場所にあります。

HKEY_LOCAL_MACHINE/Software/Wow6432Node/EnterpriseDB/PEM/agent

レジストリには次のエントリが含まれます。

パラメータ名	説明	デフォルト値
--------	----	--------

次のページに続く

表 2 – 前のページからの続き

PEM_HOST	PEM サーバーの IP アドレスまたはホスト名。	127.0.0.1。
PEM_PORT	PEM サーバーと通信するためにエージェントが接続するデータベースサーバーポート。	ポート 5432。
AgentID	PEM エージェントに割り当てられた一意の識別子。	最初のエージェントは “1”、2 番目のエージェントは “2” などです。
AgentKeyPath	PEM エージェントのキーファイルへの完全なパス。	% APPDATA %/ Roaming / pem / agent.key。
AgentCrtPath	PEM エージェントの証明書ファイルへの完全なパス。	% APPDATA %/ Roaming / pem / agent.crt
AgentFlagDir	HA サポートに使用されます。別のサーバーの監視を引き継ぐための要求をチェックするディレクトリパスを指定します。要求は、指定されたフラグディレクトリ内のファイルの形式で行われます。	デフォルトでは設定されていません。
LogLevel	ログレベルは、PEM ログファイルに書き込まれるイベントのタイプを指定します。	警告
LongWait	最初の接続試行が失敗した場合に、PEM エージェントが PEM サーバーへの接続を試行する前に待機する最大時間（秒単位）。	30 秒
shortWait	PEM エージェントがキュー内の次のプローブ（実行待ち）をチェックする前に待機する最小時間（秒単位）。	10 秒
AlertThreads	エージェントによって生成されるアラートスレッドの数。	PEM サーバーのホストに常駐するエージェントに対して 1 に設定します。他のすべてのエージェントの場合は 0。

次のページに続く

表 2 – 前のページからの続き

EnableSMTP	true に設定すると、SMTP 電子メール機能が有効になります。	PEM サーバーホストの場合は true。他のすべての場合は false。
EnableSNMP	true に設定すると、SNMP トラップ機能が有効になります。	PEM サーバーホストの場合は true。他のすべての場合は false。
ConnectTimeout	エージェントが接続を待機する最大時間（秒単位）（10 進整数文字列）。	デフォルトでは設定されていません。0 に設定すると、エージェントは無期限に待機します。
AllowServerRestart	TRUE に設定すると、エージェントは、監視するデータベースサーバーを再起動できます。このパラメーターの値に応じて、一部の PEM 機能を有効/無効にすることができます。	本当
AllowPackageManagement	TRUE に設定すると、更新モニターおよびパッケージ管理機能が有効になります。	偽
MaxConnections	接続スロットルで使用されるプローブ接続の最大数。	0（無制限の数）
ConnectionLifetime	ConnectionLifetime（または connection_lifetime）を使用して、開いているがアイドル状態の接続が保持される最小秒数を指定します。MaxConnections で指定された値に達し、待機中の要求を満たすために（別のデータベースへの）新しい接続が必要な場合、このパラメーターは無視されます。	デフォルトでは、0 に設定されます（エージェントの処理ループの後に接続がアイドル状態になると、接続はドロップされます）。
AllowBatchProbes	TRUE に設定すると、ユーザーはカスタムプローブ機能を使用してバッチプローブを作成できます。	偽
HeartbeatConnection	TRUE に設定すると、ハートビートの送信に専用の接続が使用されます。	偽
AllowStreamingReplication	TRUE に設定すると、ユーザーはストリーミングレプリケーションを設定およびセットアップできます。	偽

次のページに続く

表 2 – 前のページからの続き

BatchScriptDir	スクリプトファイル（警告用）が保存されるパスを指定します。	/tmp
ConnectionCustomSetup	監視対象サーバーとの新しい接続が確立されたときに呼び出される SQL コードを提供するために使用します。	デフォルトでは設定されていません。
ca_file	CA 証明書が存在するパスを指定します。	/opt/PEM/agent/share/certs/ca-bundle.crt。

5.3 エージェントのプロパティ

PEM Agent Properties ダイアログは、ダイアログが開かれた PEM エージェントに関する情報を提供します。ダイアログを開くには、PEM クライアントツリーコントロールでエージェント名を右クリックし、コンテキストメニューから Properties を選択します。

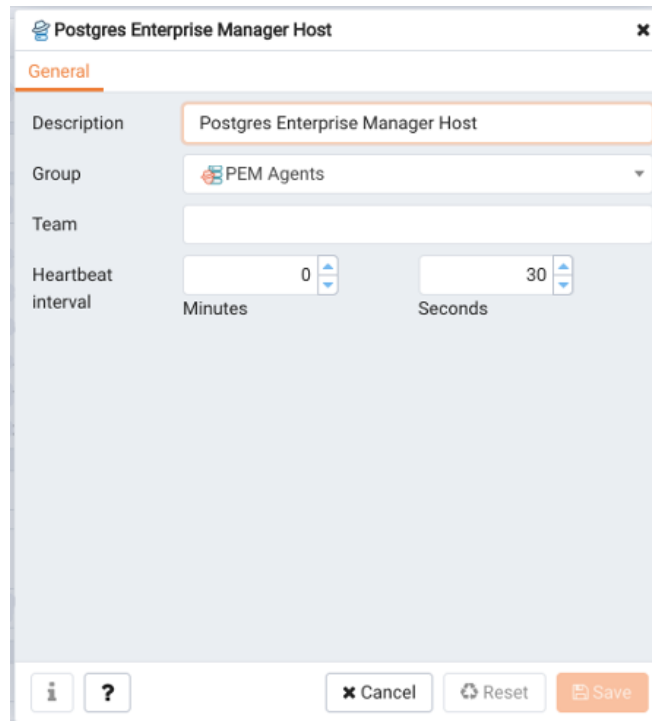


図 1 The PEM Agent Properties dialog

PEM エージェントプロパティダイアログのフィールドを使用して、PEM エージェントに関する情報を確認または変更します。

- Description フィールドには、PEM エージェントの変更可能な説明が表示されます。この説明は、PEM クライアントのツリーコントロールに表示されます。
- グループを使用して、PEM クライアントツリーコントロールでサーバーとエージェントを整理できます。Group ドロップダウンリストボックスを使用して、エージェントが表示されるグループを選択します。
- Team フィールドを使用して、エージェントによって監視されているサーバーにアクセスできるグループロールの名前を指定します。このエージェントによって監視されるサーバーは、接続されたチームメンバーの PEM クライアントツリーコントロールに表示されます。これは便利な機能であることに注意してください。[チーム] フィールドは真の分離を提供しないため、セキュ

リティの目的で使用しないでください。

- Heartbeat interval フィールドは、PEM エージェントから PEM サーバーへのレポート間の経過時間を表示します。間隔を変更するには、Minutes または Seconds フィールドの横にあるセクターを使用します。

第 6

結論

Postgres Enterprise Manager 入門ガイド

Copyright©2013-2019 EnterpriseDB Corporation。無断複写・転載を禁じます。

EnterpriseDB®Corporation 34 Crosby Drive, Suite 201, Bedford, MA 01730, USA

T +1 781 357 3390 F +1 978 467 1307 E info@enterprisedb.com www.enterprisedb.com

- EDB は、Postgres Enterprise Manager が存在する場合、ログオン UI の設計、コーディングのベストプラクティスの確立、レビュー、入力検証を検証します。EDB は追加の入力コンポーネントに対して同じアプローチに従いますが、製品の性質上、限定された検証が可能な信頼できるユーザーによって入力および送信されるフリーフォーム SQL、WMI またはその他の文字列を受け入れる必要がある場合があります。このような場合、ユーザーが誤った入力やその他の危険な入力を防ぐことはできません。
- EDB は、将来、認証された信頼できるユーザーからのフリーフォーム SQL、WMI またはその他の潜在的に危険な入力を受け入れる製品に機能を追加する権利を留保しますが、そのような機能はすべて、可能な限り最小のリスクを提供するように設計およびテストされます可能であれば、スーパーユーザーまたは同等の特権が必要です。
- EDB は、すべての潜在的な脅威を予測できることを保証しません。したがって、すべての潜在的な脆弱性が対処または検討されたことをプロセスが完全に保証することはできません。

索引

Agent Configuration, 89
Agent Privileges, 86
Agent Properties, 95
agent ssl certificate, 48

control server remotely, 51
Control service on Linux, 52
Control service on Windows, 53
Creating PEM scheduled jobs, 77

Group Roles, 66

HTTPD server, 54

Login Roles, 65

PEM
 architecture, 5
 installlation, 9
 overview, 1
PEM Overview, 2
PEM Server Configuration, 63
PEM Web インターフェイスの使用, 10
PEM エージェントの管理, 86
PEM サーバーの管理, 50
PEM のインストール-概要, 9
pg_hba.conf file, 55
Postgres Enterprise Manager を使用する理由, 3

replacing, 44
replacing ssl certificates, 44
restart PEM agent, 50
restart PEM service, 50

Security, 64
Sending email notifications for a job, 73
ssl certificates, 44
starting service, 50

update agent ssl certificate, 48

サーバーの登録, 24

サポートされているバージョンとプラットフォーム, 7

一般的なアーキテクチャ, 5

結論, 97

証明書の管理, 43