



Postgres Enterprise Manager

リリース 7.14

PEM Administrators Guide

2020 年 05 月 22 日

目次

第 1 章	PEMOverview	2
1.1	一般的なアーキテクチャ	5
1.2	PEM のインストール-概要	7
1.3	PEMWeb インターフェイスの使用	8
1.3.1	PEM メニューバー	10
1.3.2	チャート、グラフ、表の制御とカスタマイズ	21
1.3.3	オンラインヘルプとドキュメント	22
第 2 章	サーバーの登録	23
2.1	サーバーの手動登録	23
2.2	自動サーバー検出	38
2.3	pemworker ユーティリティを使用してサーバーを登録する	41
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	44
2.4	接続とバインドの検証	45
第 3 章	証明書の管理	47
3.1	SSL 証明書の置き換え	48
3.2	エージェント SSL 証明書の更新	52
第 4 章	設定の管理	54
第 5 章	PEM サーバーの管理	55
5.1	PEM サーバーとエージェントの起動と停止	55
5.2	監視対象サーバーのリモート起動と停止	57

5.3	Linux での PEM サーバーまたは PEM エージェントの制御	58
5.4	Windows での PEM サーバーまたは PEM エージェントの制御	59
5.5	HTTPD サーバーの制御	60
5.6	pg_hba.conf ファイルの変更	61
5.7	データベースとオブジェクトの作成と保守	63
5.8	PEM 認証の管理	65
5.9	PEM サーバー構成の編集	67
5.10	セキュリティの管理	68
5.10.1	ログインロール	68
5.10.2	グループの役割	70
5.10.3	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	71
5.10.4	チームロールを使用する	74
5.10.5	オブジェクト権限	75
5.11	ジョブ通知の管理	76
5.11.1	ジョブレベルでのジョブ通知の構成	76
5.11.2	エージェントレベルでのジョブ通知の構成	78
5.11.3	サーバーレベルでのジョブ通知の構成	79
5.12	PEM スケジュール済みジョブの管理	80
第 6 章	PEM エージェントの管理	92
6.1	エージェント権限	92
6.2	エージェント構成	95
6.3	エージェントのプロパティ	101
第 7 章	結論	103
索引		105

このドキュメントでは、PostgresEnterpriseManager™ (PEM) の概要を説明します。PostgresEnterpriseManager (PEM) は、PostgreSQL および EnterpriseDBAdvancedServer 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。

PEM でサポートされているプラットフォームとバージョンについては、次の EnterpriseDBWeb サイトにアクセスしてください。

<https://www.enterprisedb.com/services-support/edb-supported-products-and-platforms#pem>

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

PEMOverview

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では、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。
- **完全な SQLIDE。** 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礎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。

- **予防的なアラート管理。** 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。しきい値違反があると、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伝えるアラートが集中ダッシュボードに送信されます。
- **簡素化された容量計画。** 主要なパフォーマンス関連の統計はすべて自動的に収集され、指定された期間 PEM のリポジトリに保持されます。CapacityManager 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、AdvancedServer 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。その後、キャプチャされた SQL ステートメントをフィルター処理できるため、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 AdvancedServer の IndexAdvisor 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 PostgresExpert ユーティリティが含まれています。PostgresExpert 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監視。** **ストリーミングレプリケーションダッシュボー

ドを監視するか、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させることができます。

- ****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。

1.1 一般的なアーキテクチャ

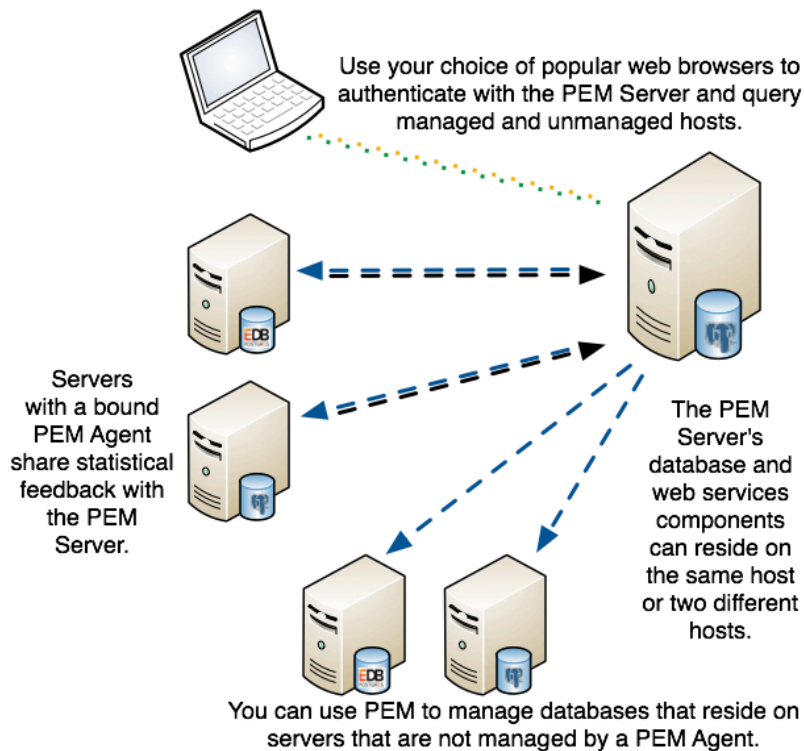


図 1 *PostgresEnterpriseManager* の一般的なアーキテクチャ

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、PostgresEnterpriseManager 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。

PEMWeb インターフェイス

PEM サーバーと一緒に配布される PEMWeb インターフェースにより、選択したブラウザー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。

1.2 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 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も、別のホストに存在する場合もあります。

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 で管理する追加の物理マシンまたは仮想マシンをそれぞれ登録します。便宜上、PEM はリモート監視をサポートします（監視するサーバーと同じホストにエージェントを配置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。SQLProfiler 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。SQLProfiler プラグインは AdvancedServer 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。

1.3 PEMWeb インターフェイスの使用

PEMWeb 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。PostgresEnterpriseManager ログインウィンドウが開きます。



図 2 PEM ログインページ

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。

PEMWeb 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。

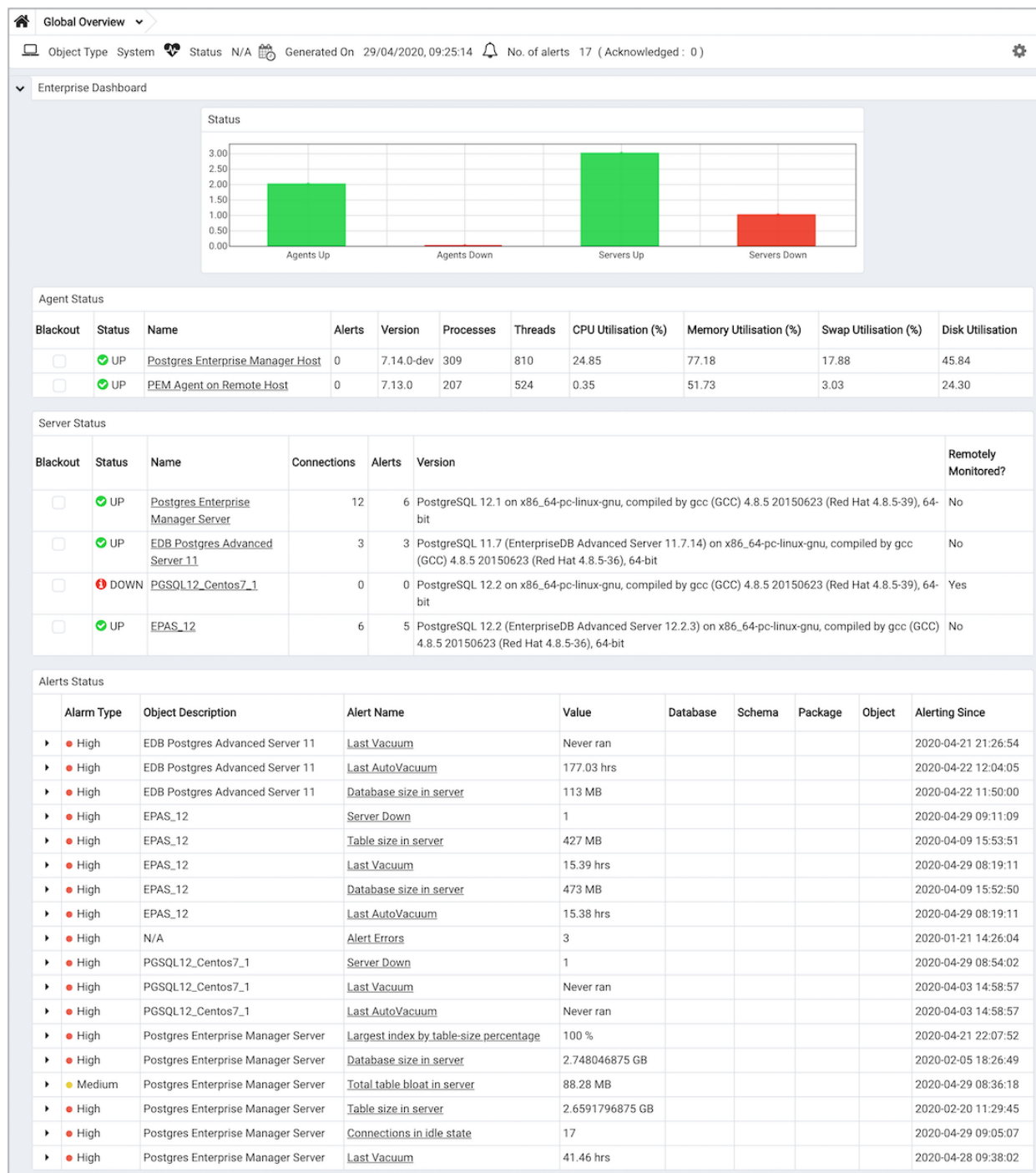


図 3 PEMGlobalOverview ダッシュボード

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。

1.3.1 PEM メニューバー

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有効でない場合があります（設計上）。
- 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。

PEMWeb 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

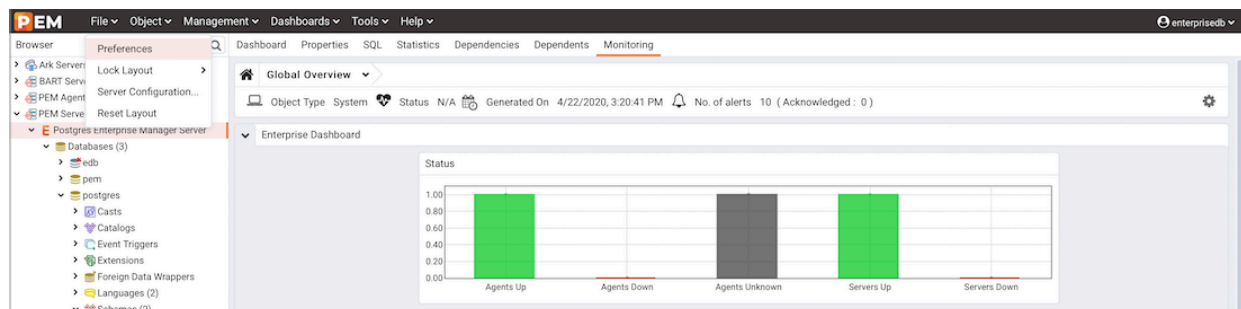


図4 ファイルメニュー

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。
レイアウトをロック	クリックしてサブメニューを開き、UI レイアウトをロックするレベルを選択します。
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。
レイアウトをリセット	ワークスペースパネルが誤ってまたは意図的にポップアウトされた場合、[レイアウトのリセット] を使用してデフォルトにリセットできます。

オブジェクトメニュー

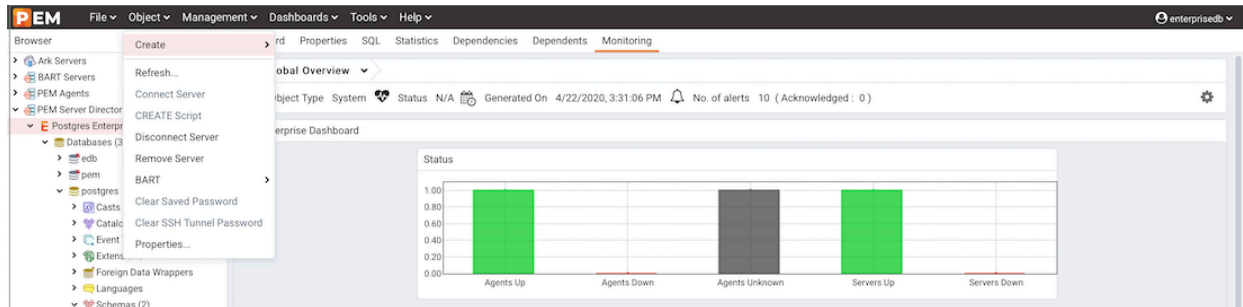


図 5 オブジェクトメニュー

The `Object` menu is context-sensitive. Use the `Object` menu to access the following options:

メニューオプション	アクション
作成する	[作成] 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。
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。
削除/ドロップ	クリックして、現在選択されているオブジェクトをサーバーから削除します。
接続サーバー	[サーバーに接続] ダイアログをクリックして、サーバーとの接続を確立します。
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。
サーバーの切断	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。
BART	クリックすると、BART 設定の削除、BART バックアップの取得、または BART 設定の再検証のため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。
保存したパスワードをクリア	データベースサーバーのパスワードを保存した場合は、クリックして保存したパスワードをクリアします。パスワードが保存された後にのみ有効になります。
SSH トンネルパスワードをクリア	ssh トンネルパスワードを保存した場合は、クリックして保存したパスワードをクリアします。パスワードが保存された後にのみ有効になります。
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
トリガー	現在選択されているテーブルの*無効*または*有効*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	クリックすると、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できます。

管理メニュー

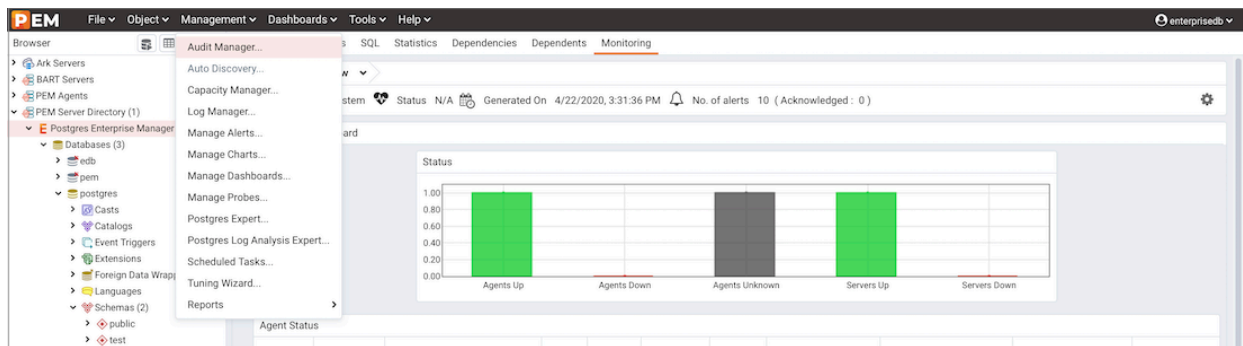


図 6 管理メニュー

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 CapacityManager 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、プローブの管理ダイアログを開き、VACUUM、ANALYZE、REINDEX、または CLUSTER を実行します。
Postgres エキスパート...	クリックして PostgresExpert 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 PostgresLogAnalysisExpert 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。
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。
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。
報告書	クリックしてレポートダイアログを開き、サーバーのシステム構成レポートとコア使用率レポートを生成します。

ダッシュボードメニュー

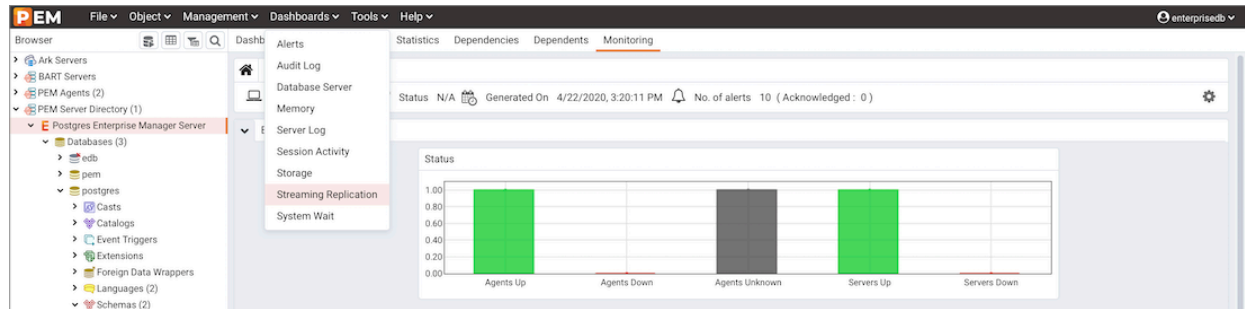


図7 ダッシュボードメニュー

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	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	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
データベースサーバー	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。
記憶	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	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	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	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。
サーバーログ	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	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。
保管	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	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	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

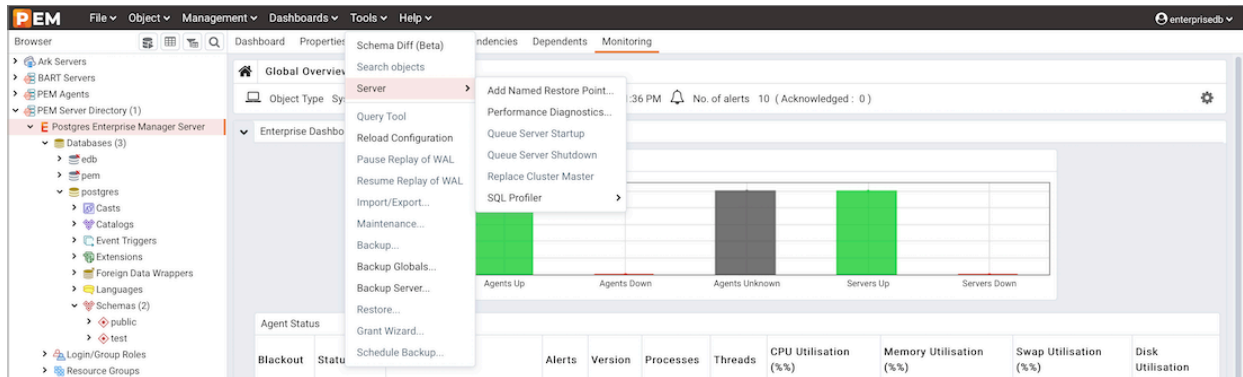


図 8 [ツール] メニュー

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
スキーマ差分	[スキーマの差分] ダイアログをクリックして開き、2つのデータベーススキーマ間でスキーマオブジェクトを比較します。
オブジェクトを検索する	クリックして [オブジェクトの検索] ダイアログを開き、データベース内のデータベースオブジェクトを検索します。
サーバー	クリックして、名前付き復元ポイントの追加、パフォーマンス診断、キューサーバーの起動、キューサーバーのシャットダウン、クラスターマスターの交換、SQL プロファイラーなどのさまざまなサーバー関連ツールにアクセスします。
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。
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。
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。
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...	クリックして、[メンテナンス...] ダイアログを開き、VACUUM、ANALYZE、REINDEX、または CLUSTER を選択します。
バックアップ...	データベースオブジェクトをバックアップするには、[バックアップ...] ダイアログをクリックして開きます。
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。
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。
グラントウィザード...	クリックして、GrantWizard ツールにアクセスします。
バックアップのスケジュール	クリックして、BART バックアップの [バックアップのスケジュール] ダイアログにアクセスします。

ヘルプメニュー

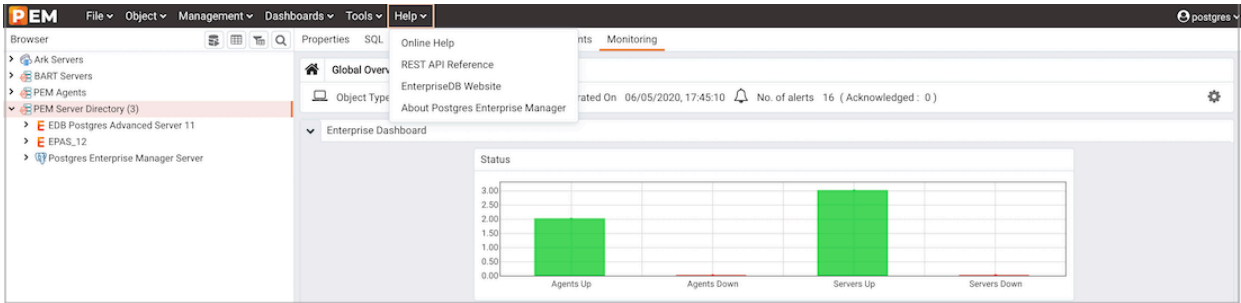


図9 ヘルプメニュー

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 PostgresEnterpriseManager のドキュメントを開きます。
RESTAPI リファレンス	クリックして、RESTAPI リファレンスを開きます。
EnterpriseDBWeb サイト	ブラウザウィンドウで EnterpriseDBWeb サイトをクリックして開きます。
PostgresEnterpriseManager について	クリックして、PostgresEnterpriseManager のバージョン情報とユーザー情報を見つけます。

1.3.2 チャート、グラフ、表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の 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。

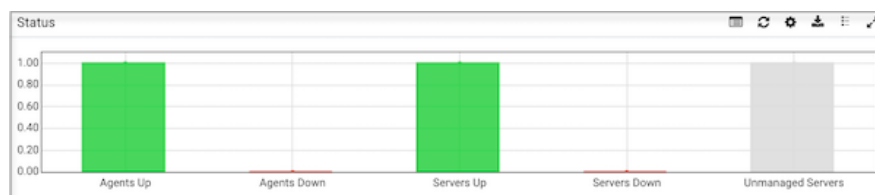


図 10 PEM クライアントチャートコントロールアイコン

アイコンを選択して:

	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。
	チャート、グラフ、または表のコンテンツを更新します。
	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します。
	チャート、グラフ、またはテーブルで使用されている凡例を表示します。
	チャートまたはグラフをフルスクリーンに展開します。

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。

1.3.3 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。

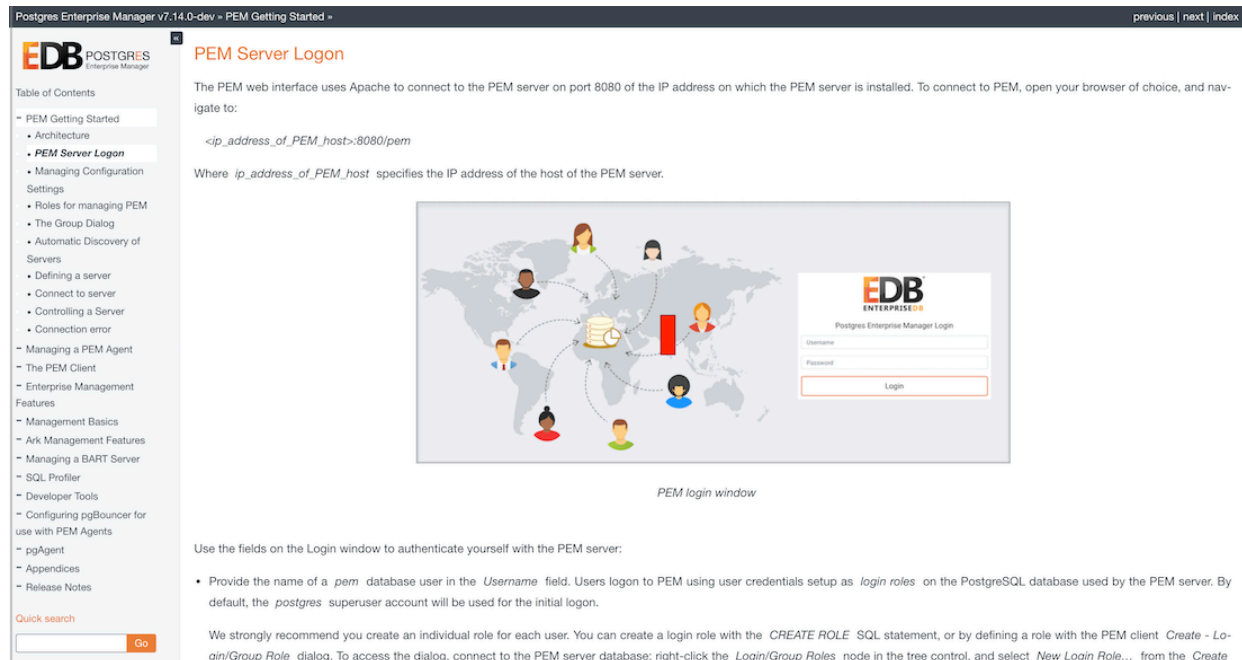


図 11 PEM オンラインヘルプ

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

第 2

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。

- AdvancedServer 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。

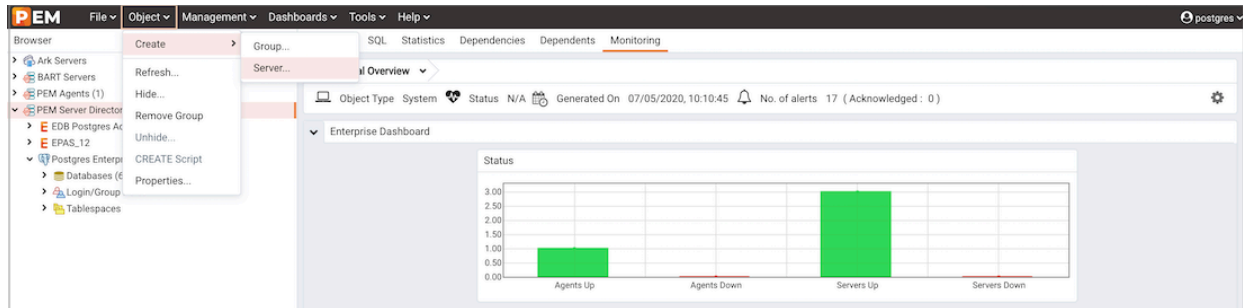


図1 「サーバーの作成」ダイアログへのアクセス

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。

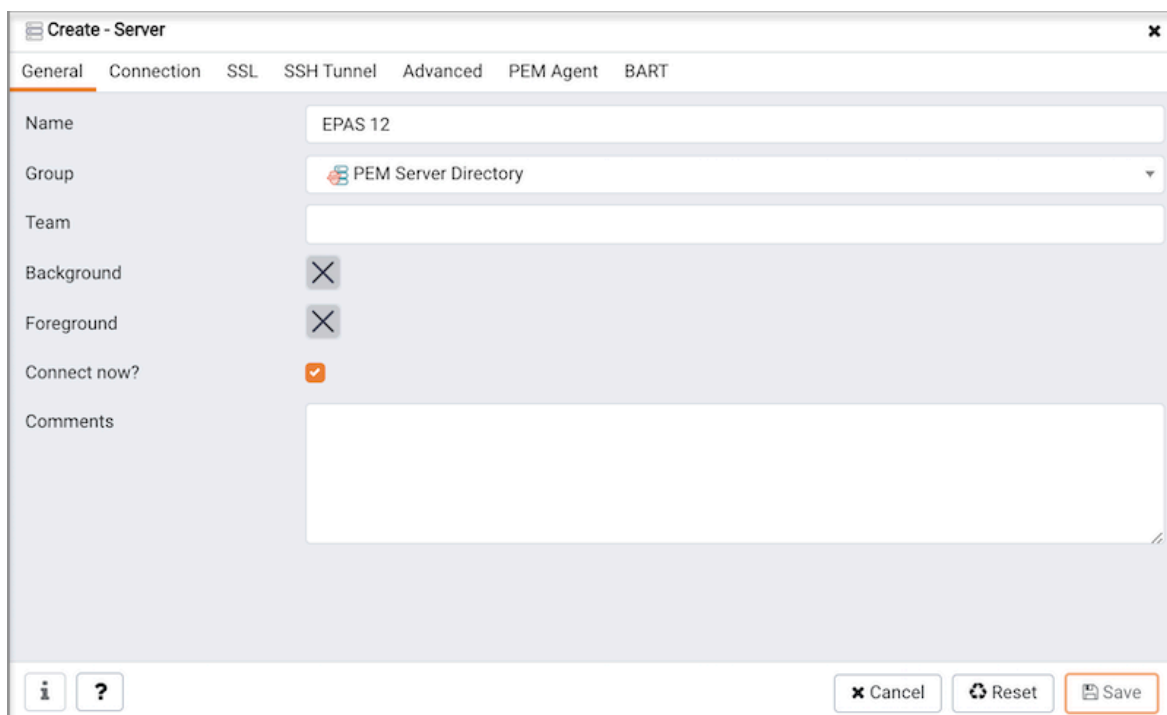


図2 [作成-サーバー] ダイアログの [全般] タブ

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

- 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、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。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、運用グループ、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。
- Check the box next to `Connect now?` to instruct PEM to attempt a server connection when you click the Save button. Leave `Connect now?` unchecked if you do not want the PEM client to validate the specified connection parameters until a later connection attempt.
- Comments フィールドにサーバーに関するメモを入力します。

The screenshot shows the 'Create - Server' dialog box with the 'Connection' tab selected. The fields are as follows:

Field	Value
Host name/address	192.168.1.135
Port	5444
Maintenance database	postgres
Username	enterprisedb
Password	...
Save password?	☒
Role	

Buttons at the bottom: `?`, `Cancel`, `Reset`, `Save`.

図3 [作成-サーバー] ダイアログの [接続] タブ

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 PostgreSQLUnix 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。
- Port フィールドにホストのポート番号を指定します。
- Use the Maintenance database field to specify the name of the initial database that PEM will connect to, and that will be expected to contain pgAgent schema and adminpack objects installed (both optional). On PostgreSQL 8.1 and above, the maintenance DB is normally called postgres; on earlier versions template1 is often used, though it is preferable to create a postgres database to avoid cluttering the template database.
- 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たパスワードを Password フィールドに入力します。
- Save password? の横のボックスをオンにして、後で再利用できるように、パスワードを暗号化形式で PEM バックエンドデータベースに保存するように PEM に指示します。パスワードはサーバーごとにユーザーごとに保存されるため、他のチームメンバーとは共有されません。PEM は保存されたパスワードを使用して、次回データベースサーバーに接続します。保存したパスワードを削除するには、最初にデータベースサーバーを切断してから、[オブジェクト/コンテキスト] メニューの [保存したパスワードをクリア] メニュー項目を使用します。
- 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。

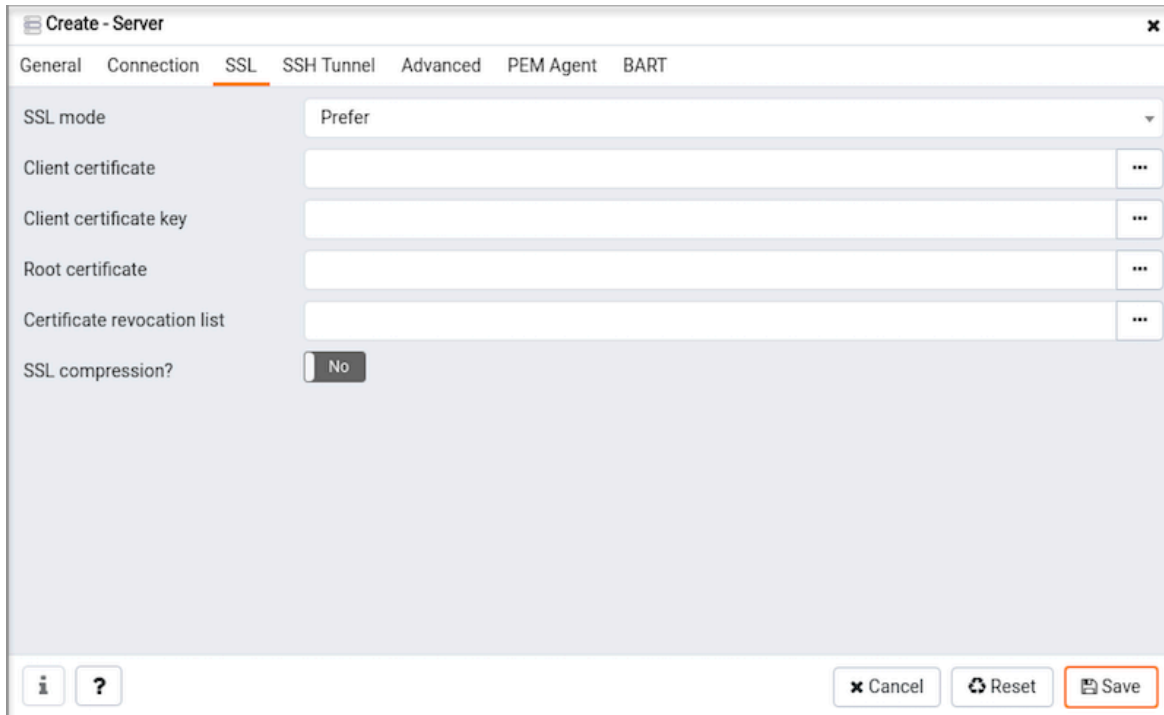


図 4 [作成-サーバー] ダイアログの [SSL] タブ

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。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、このファイルはデフォルトの `~/.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。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、

このファイルはデフォルトの `~/.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は圧縮されます。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 [作成-サーバー] ダイアログの [SSH トンネル] タブ

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（クライアント接続が転送される）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。
- SSH ホストに接続するときに使用される認証のタイプを Authentication フィールドに指定します。
- 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 [作成-サーバー] ダイアログの [詳細設定] タブ

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`<http://www.postgresql.org/docs/current/interactive/catalog-pg-database.html>>`\_\_テーブルに対して使用される SQL 制限を指定します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`<http://www.postgresql.org/docs/current/interactive/catalog-pg-namespace.html>>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。__。
- Use the Password file field to specify the location of a password file (.pgpass). The .

pgpass file allows a user to login without providing a password when they connect. It must be present on the PEM Server. For more information, see the Postgres documentation at:

<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 libpqv10.0 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。EnterpriseManager 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、サーバーの起動に使用される init スクリプトの名前は /etc/init.d で、サーバーを起動する systemd スクリプトの名前は systemctl です。たとえば、AdvancedServer10 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、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/pgsql-x/bin/pg_ctl -D /var/lib/pgsql/x/data ここで、x は PostgreSQL データベースサーバーのバージョンです。
```

- サーバーが FailoverManager 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、FailoverManager 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 FailoverManager 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 FailoverManager バイナリファイルの場所を指定します。デフォルトでは、フェールオーバーマネージャーのバイナリファイルは /usr/

edb/efm-x.x/bin にインストールされます。x.x はフェイルオーバーマネージャーのバージョンを指定します。

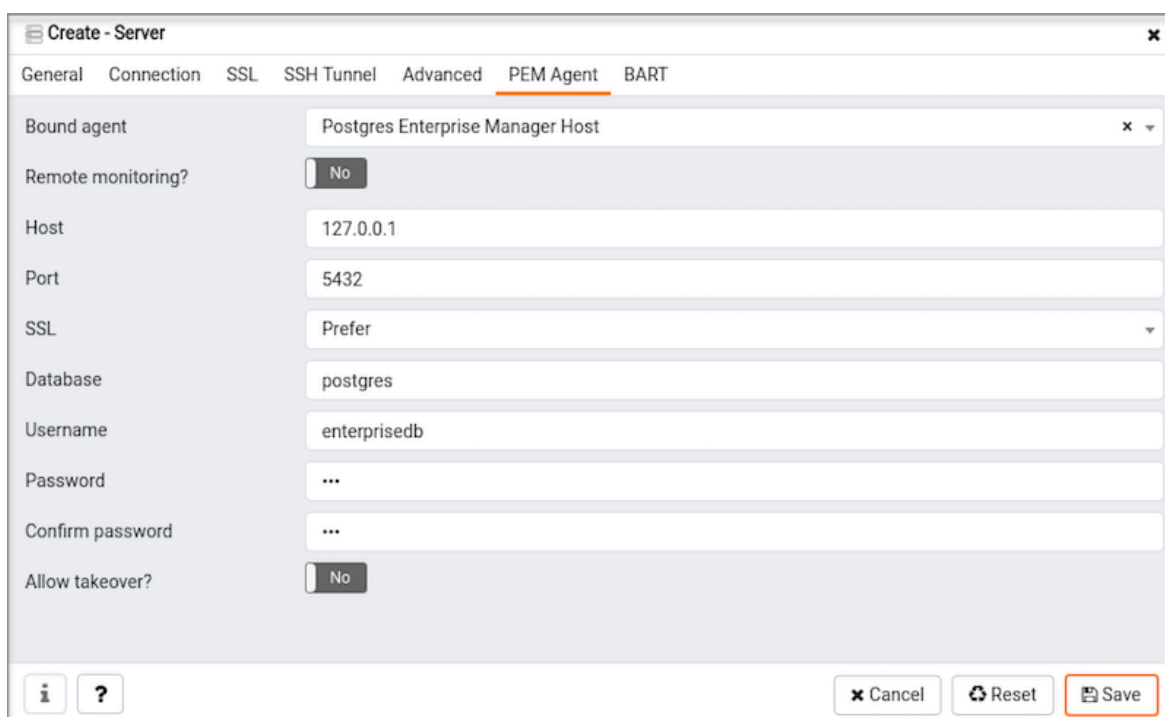


図 7 [作成-サーバー] ダイアログの [PEM エージェント] タブ

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。

- 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、EnterpriseManager 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。
- 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 (AuditManager、CapacityManager、LogManager、PostgresExpert 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)。
- 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、ソケットパスを指定できます。 /tmp
- 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。

<https://www.enterprisedb.com/edb-docs/d/postgresql/reference/manual/12.1/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 フィールドに、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。指定し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に機能しない機能のリストについては、[.ref: エージェント権限 <non_root_user>](#) を参照してください。

If you are using Postgres version 10 or above, you can use the `pg_monitor` role to grant the required privileges to a non-superuser. For information about `pg_monitor` role, see:

<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。このファイルは、エージェントがインストールされているシステム上に存在し、アクセス可能でなければなりません。
- 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ように指定します。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。

The screenshot shows the 'Postgres Enterprise Manager Server' window with the 'BART' tab selected. The 'General' sub-tab is active. A message at the top states: 'For BART configuration, you need to install a PEM agent on the database server if Remote Monitoring is disabled for the agent. BART supports database server version 9.5 and above.' The configuration fields include: 'BART server' (a dropdown menu), 'Server name' (a text field with a description: 'Database server name that uniquely identifies an entry for database server in the server section of the configuration file'), 'Backup name' (a text field with a description: 'Template for backup name (may include %year, %month, %day, %hour, %minute, and %second)'), 'Host address' (a text field with a description: 'IP address of the database server to be configured for backup'), 'Port' (a text field), 'User' (a text field), 'Password' (a text field), 'Cluster owner' (a text field with a description: 'Operating system user that owns the database cluster'), 'Override archive command?' (a radio button set to 'No' with a description: 'Overrides the archive_command settings in the postgresql.conf file.'), 'Archive command' (a text field with a description: 'Parameters for archive command (%p, %h, %a, %f). Ensure to bind a PEM agent and provide Service ID to reload or restart the database server.'), 'Allow incremental backup?' (a radio button set to 'No'), and 'Setup passwordless SSH?' (a radio button set to 'No' with a description: 'Ensure to bind a PEM agent to setup passwordless SSH authentication between BART host and database server.'). At the bottom right, there are 'Cancel', 'Reset', and 'Save' buttons.

図 8 [サーバーの作成] ダイアログ (BART-[全般] タブ)

BART タブの下の General タブのフィールドを使用して、PEM サーバーにマップする BART サーバーの一般的なプロパティを説明します。

- BART server フィールドを使用して、BART サーバー名を選択します。PEM コンソールで設定されたすべての BART サーバーがこのドロップダウンリストにリストされます。
- Server name フィールドを使用して、BART サーバーを使用してバックアップするデータベースサーバーの名前を指定します。この名前は、BART 構成ファイルに保存されます。
- データベースサーバーのバックアップに割り当てるユーザー定義名のテンプレートを指定するには、Backup name フィールドを使用します。バックアップ名テンプレートを指定しない場合、バックアップは、BART に割り当てられた整数のバックアップ識別子によってのみ、BART サブコマンドで参照できます。
- Host address フィールドを使用して、バックアップ用に構成するデータベースサーバーの IP アドレスを指定します。
- Port フィールドを使用して、バックアップするデータベースに使用するポートを指定します。

- User フィールドを使用して、PEM コンソールから BART を使用してバックアップするデータベースのユーザーを指定します。このデータベースサーバーの増分バックアップを有効にする場合、ユーザーはスーパーユーザーである必要があります。
- Password フィールドを使用して、バックアップするデータベースのユーザーのパスワードを指定します。
- Cluster Owner フィールドを使用して、データベースクラスターを所有する Linux オペレーティングシステムのユーザーアカウントを指定します。これは通常、Oracle データベース互換モードでインストールされた AdvancedServer データベースクラスターの場合は `enterprisedb`、PostgreSQL データベース互換モードでインストールされた PostgreSQL データベースクラスターおよび AdvancedServer データベースクラスターの場合は `postgres` です。
- Archive command フィールドを使用して、`bart.cfg` ファイルで使用するアーカイブコマンド文字列の形式を指定します。アーカイブコマンドに入力すると、データベースサーバーの `Postgresql.conf` ファイルが上書きされます。サーバーが追加されると、データベースサーバーが再起動されるか、データベース構成が再読み込みされます。
- Allow incremental backup? スイッチを使用して、このデータベースサーバーで増分バックアップを有効にするかどうかを指定します。
- Setup passwordless SSH? スイッチを使用して、データベースサーバーと BART サーバー間のパスワードレスログインを許可する SSH 証明書を作成するかどうかを指定します。パスワードレス SSH 認証をセットアップする前に、PEM エージェントをバインドしてください。パスワードなしの SSH は、PEM エージェントによってリモートで監視されているデータベースサーバーでは機能しません。

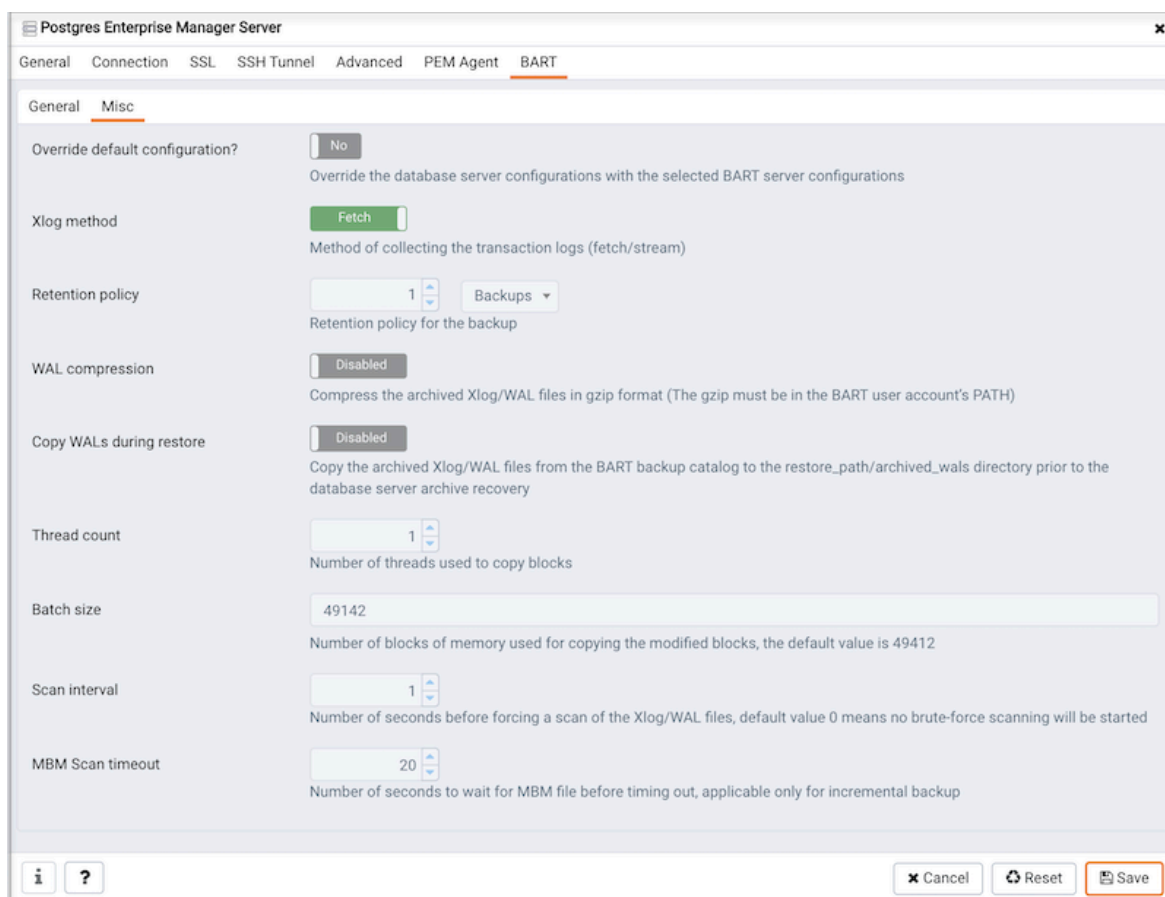


図9 [作成-サーバー] ダイアログ (BART-[その他] タブ)

BART タブの下の Misc タブのフィールドを使用して、BARTServer のその他のプロパティを説明します。

- `Override default configuration?` スイッチを使用して、特定のデータベースサーバー構成で BART サーバー構成を上書きするかどうかを指定します。
- `Xlog method` を使用して、`pg_basebackup` の実行中にトランザクションログを収集する方法を指定します。
- `Retention policy` フィールドを使用して、バックアップの保持ポリシーを指定します。これにより、アクティブなバックアップが廃止としてマークされ、削除の候補となるタイミングが決定されます。保持ポリシーは、バックアップの数または期間（日、週、または月）で指定できます。
- `WAL compression` スイッチを使用して、アーカイブされた Xlog/WAL ファイルを Gzip 形式で圧縮するかどうかを指定します。WAL 圧縮を有効にするには、gzip 圧縮プログラムが BART

ユーザーアカウントの `PATH` に存在する必要があります。増分バックアップを作成する必要があるデータベースサーバーでは、`wal_compression` 設定を有効にしないでください。

- `Copy WALs during restore` フィールドを使用して、**RESTORE** 操作を呼び出すときにアーカイブされた **WAL** ファイルを収集する方法を指定します。データベースサーバーのアーカイブリカバリの前に、アーカイブされた **WAL** ファイルを **BART** バックアップカタログから `<restore_path>/archived_wals` ディレクトリにコピーするには、有効に設定します。無効に設定すると、データベースサーバーのアーカイブリカバリ中に **BART** バックアップカタログからアーカイブされた **WAL** ファイルを直接取得します。
- `Thread count` フィールドを使用して、ブロックをコピーするスレッドの数を指定します。`pg_basebackup` ユーティリティでバックアップを取りたい場合は、1 を 1 に設定する必要があります。
- `Batch size` フィールドを使用して、変更されたブロックのコピーに使用されるメモリのブロック数を指定します。これは増分バックアップにのみ適用されます。
- `Scan interval` フィールドを使用して、**WAL** スキャナーが新しい **WAL** ファイルをスキャンするまでの秒数を指定します。
- `MBM scan timeout` フィールドを使用して、タイムアウトまで **MBM** ファイルを待機する秒数を指定します。これは増分バックアップにのみ適用されます。

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。ManageProbes 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 10 PEM 自動検出ダイアログ

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。

表示されたすべてのサーバーの横にあるボックスを選択するには Check All ボタンを使用し、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するには Uncheck All を使用します。

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。
- By default, the Maintenance database field indicates that the selected server uses a Postgres maintenance database. Customize the content of the Maintenance database field for your installation.

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。

Postgres Enterprise Manager, リリース 7.14

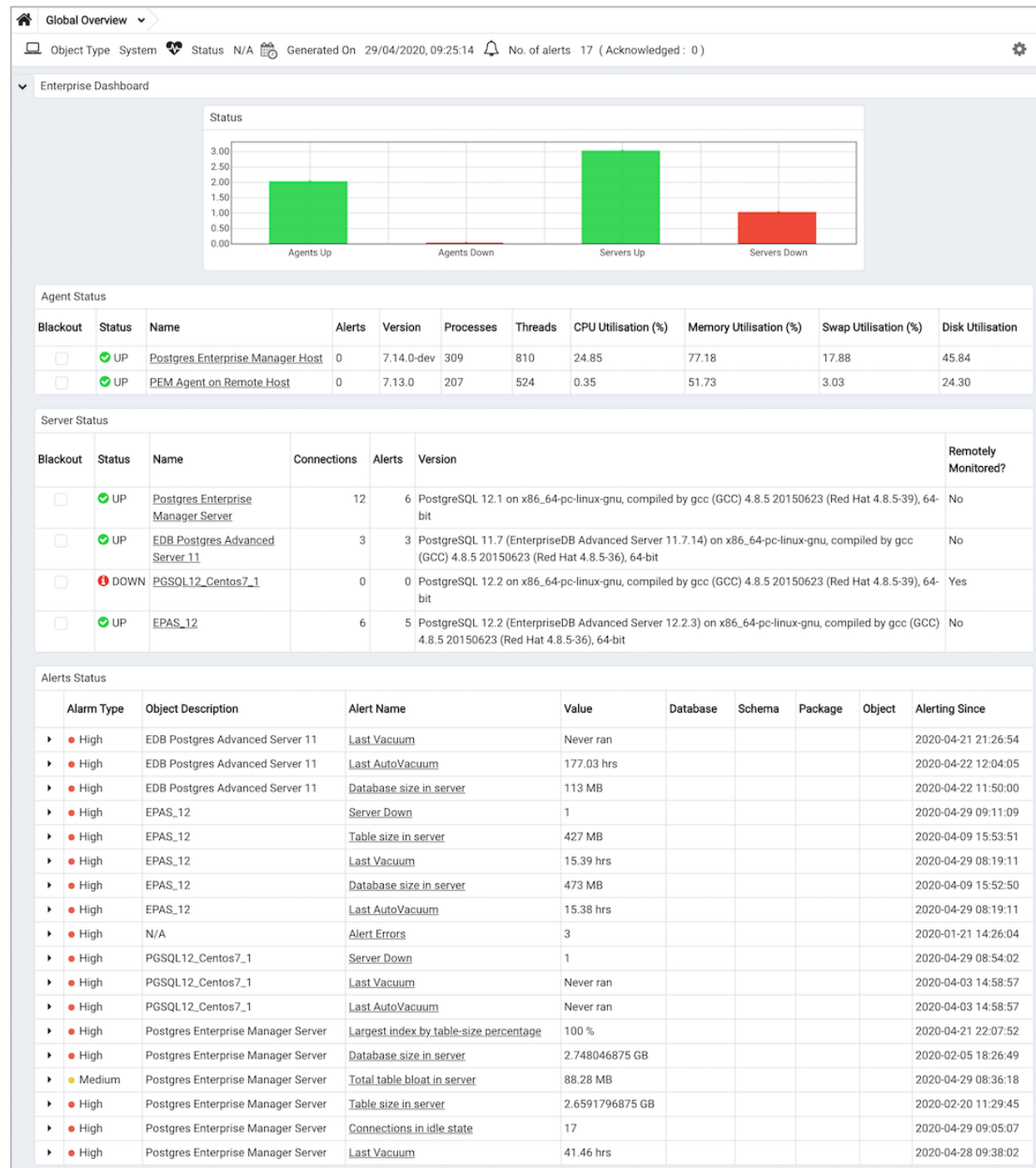


図 11 登録済みサーバー

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

You can use the `pemworker` utility to register a server for monitoring by the PEM server or to unregister a database server. During registration, the `pemworker` utility will bind the new server to the agent that resides on the system from which you invoked the registration command. To register a server:

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 PostgreSQLUnix 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（STOP、START、RESTART など）を制御するデータベースサービスの名前を指定します。オプション。
--remote-monitoring	--remote-monitoring 句と no（デフォルト）の値を含めて、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（yes）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（AuditManager、CapacityManager、LogManager、PostgresExpertandTuningWizard）。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	データベースで認証するとき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。
42--owner	監視対象サーバーを所有する役割の名前を指定第 2 章。サーバーの登録

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るデータベースサーバーのパスワードを指定します。

パスワードの入力に失敗すると、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

You can use the `pemworker` utility to unregister a database server; to unregister a server, invoke the `pemworker` utility:

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
<code>--pem-user</code>	PEM 管理ユーザーの名前を指定します。必須です。
<code>--server-addr</code>	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 PostgreSQLUnix 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。
<code>--server-port</code>	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。

パスワードの入力に失敗すると、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

登録すると、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。

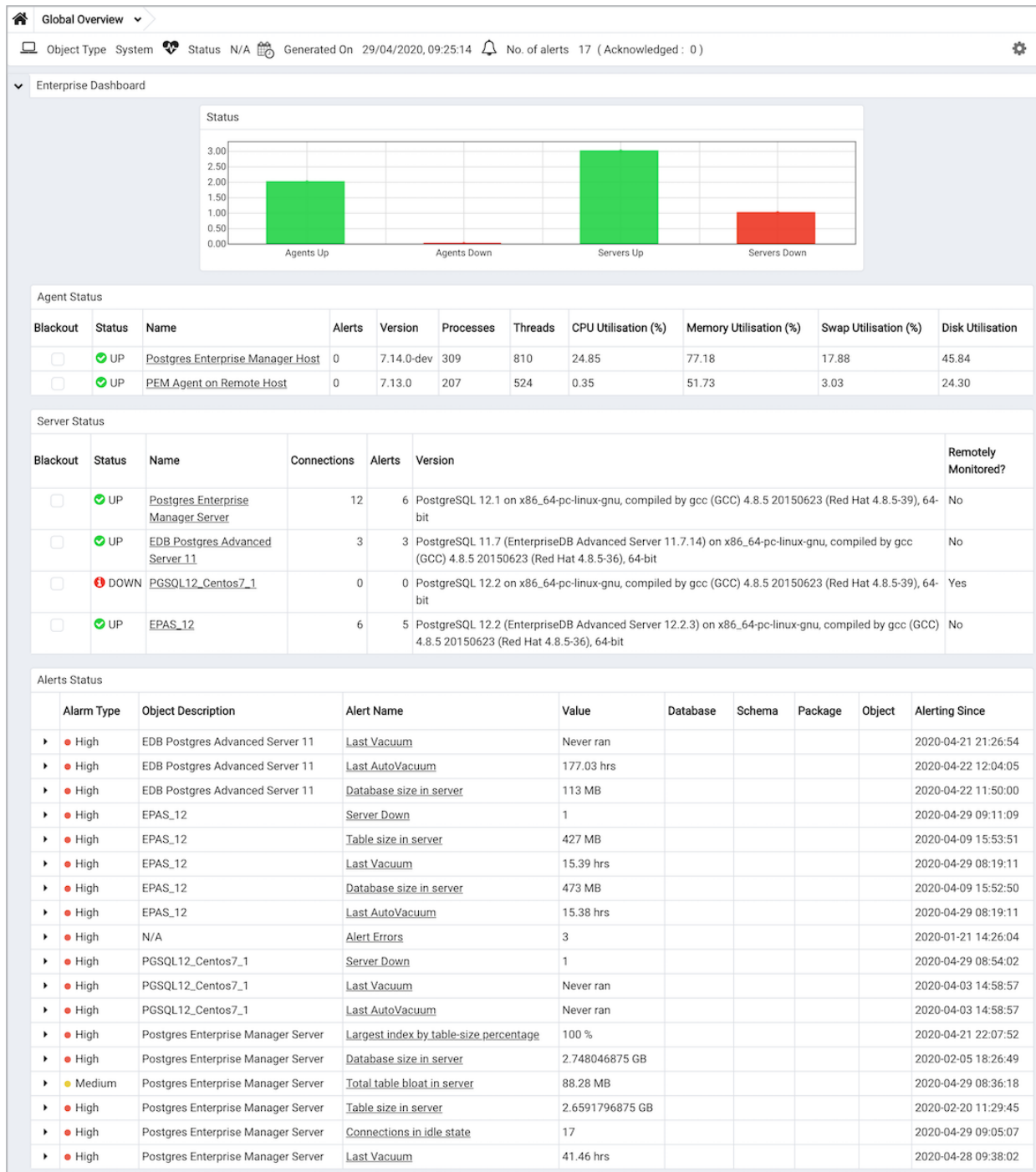


図 12 グローバル概要ダッシュボード

When initially connecting to a newly bound server, the Global Overview dashboard may display the new server with a status of “unknown” in the server list; before recognizing the server, the bound agent must execute a number of probes to examine the server, which may take a few minutes to complete depending on network availability.

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. 実行中のすべての PEM エージェントを、最初にサーバーホストで、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。

Linux で `init.d` を使用する場合（例:Centos6）

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

Linux で `systemd` を使用している場合（例:Centos7）

```
systemctl stop pemagent
```

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 `PostgresEnterpriseManagerAgent` 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 インストールの下の `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。

```
/var/lib/pgsql/x/data
```

 ここで `x` は PostgreSQL データベースのバージョンです。

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:

```
/usr/pgsql-x.x/bin/psql -U postgres -d pem --no-psqlrc -t -A -c  
/"SELECT public.openssl_rsa_generate_key(1024)/" > /var/lib/pgsql/  
x/data/ca_key.key
```

After creating the `ca_key.key` file, cat the contents to the variable `CA_KEY` for use when generating the `ca_certificate.crt` file and modify the privileges on the `ca_key.key` file:

```
CA_KEY=$(cat /var/lib/pgsql/x/data/ca_key.key)
```

```
chmod 600 /var/lib/pgsql/x/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 クエリを一意の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, '/var/lib/pgsql/x/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。

```
/usr/pgsql-x.x/bin/psql -U postgres -d pem --no-psqlrc -t -A -f /  
tmp/_random.$$ > /var/lib/pgsql/x/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 /var/lib/pgsql/x/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 /var/lib/pgsql/x/data/ca_certificate.crt /var/lib/pgsql/x/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 /var/lib/pgsql/x/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。

```
/usr/pgsql-x/bin/psql -U postgres -d pem --no-psqlrc -t -A  
-c /"SELECT openssl_rsa_generate_crl('/var/lib/pgsql/x/data/  
ca_certificate.crt', '/var/lib/pgsql/x/data/ca_key.key')/" > /  
var/lib/pgsql/x/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 /var/lib/pgsql/x/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:

```
/usr/pgsql-x/bin/psql -U postgres -d pem --no-psqlrc -t -A -c /  
"SELECT public.openssl_rsa_generate_key(1024)/" >> /var/lib/pgsql/  
x/data/server.key
```

After creating the server.key file, cat the contents to the variable SSL_KEY for use when generating the server.crt file and modify the privileges on the server.key file:

```
SSL_KEY=$(cat /var/lib/pgsql/x/data/server.key)
```

```
chmod 600 /var/lib/pgsql/x/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 クエリを一意の名前を持つ一時ファイルに配置します。

```
echo /"SELECT openssl_csr_to_cert(openssl_rsa_key_to_csr('${SSL_KEY}',
'PEM','US','MA','Bedford','Postgres Enterprise Manager',
'support@enterprisedb.com'), '/var/lib/pgsql/x/data/
ca_certificate.crt', '/var/lib/pgsql/x/data/ca_key.key')/" > /
tmp/_random.$$
```

```
/usr/pgsql-x/bin/psql -U postgres -d pem --no-psqlrc -t -A -f /
tmp/_random.$$ >> /var/lib/pgsql/x/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 /var/lib/pgsql/x/data/server.crt
```

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

10. Postgres サーバーを再起動します。

Linux で init.d を使用する場合 (例:Centos6)

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

Linux で systemd を使用している場合 (例:Centos7)

```
systemctl restart postgresql-x
```

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。

```
/usr/pgsql-x/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。

```
/usr/pgsql-x/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。

```
/usr/pgsql-x/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'), '/var/lib/pgsql/x/data/
ca_certificate.crt', '/var/lib/pgsql/x/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。

- init.d を使用する 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
```

systemd を使用する 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。

```
systemctl start pemagent
```

- On a Windows host, you can use the Services applet to start the PEM agent. The PEM agent service is named Postgres Enterprise Manager Agent; highlight the service name in the Services dialog, and click Start the service.

設定の管理

複数の設定ファイルは、起動時に `PostgresEnterpriseManager` によって読み取られます。ファイルは次のとおりです。

- `config.py`: これはメインの構成ファイルであり、変更しないでください。次のファイルのいずれかで上書きされる可能性のある構成設定の参照として使用できます。
- `config_distro.py`: This file is read after `config.py` and is intended for packagers to change any settings that are required for their Postgres Enterprise Manager distribution. This may typically include certain paths and file locations. This file is optional, and may be created by packagers in the same directory as `config.py` if needed.
- `config_local.py`: このファイルは `config_distro.py` の後に読み取られ、エンドユーザーがローカルの設定や標準に合わせて調整したいデフォルトまたはパッケージ固有の設定を変更するためのものです。このファイルはオプションであり、必要に応じて、`config.py` と同じディレクトリにユーザーが作成します。

デフォルトの `config.py` ファイルのコピーは、参照用に PEM オンラインヘルプに含まれています。

第 5

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

5.1 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、または再起動すると、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、プラットフォームと 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。

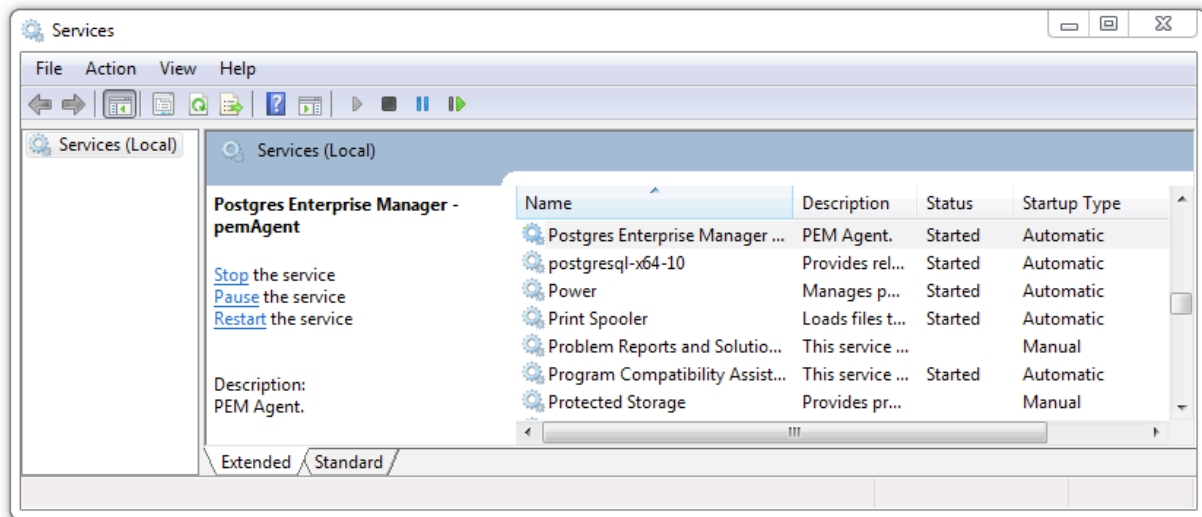


図 1 Windows サービスウィンドウの PEM サービス

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）は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 EnterpriseDBWeb サイトから入手できる*EDBPostgresAdvancedServer 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>

5.2 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:
 - AdvancedServer 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。

ここで、*x* はサーバーのバージョン番号を示します。

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、Tools 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。

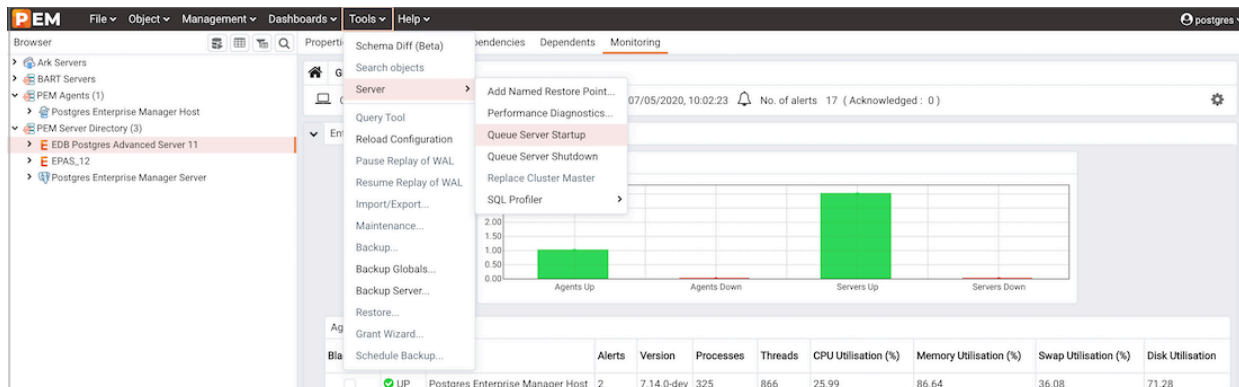


図 2 管理対象サーバーの [ツール] メニュー

5.3 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、次を制御するサービススクリプトの名前:

- AdvancedServer 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

ここで、***x***はサーバーのバージョン番号を示します。

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はサービスの名前です。

actionは、サービスによって実行されるアクションを指定します。指定してください:

- `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。

5.4 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
- AdvancedServer 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

ここで、`*x*`はサーバーのバージョン番号を示します。

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。

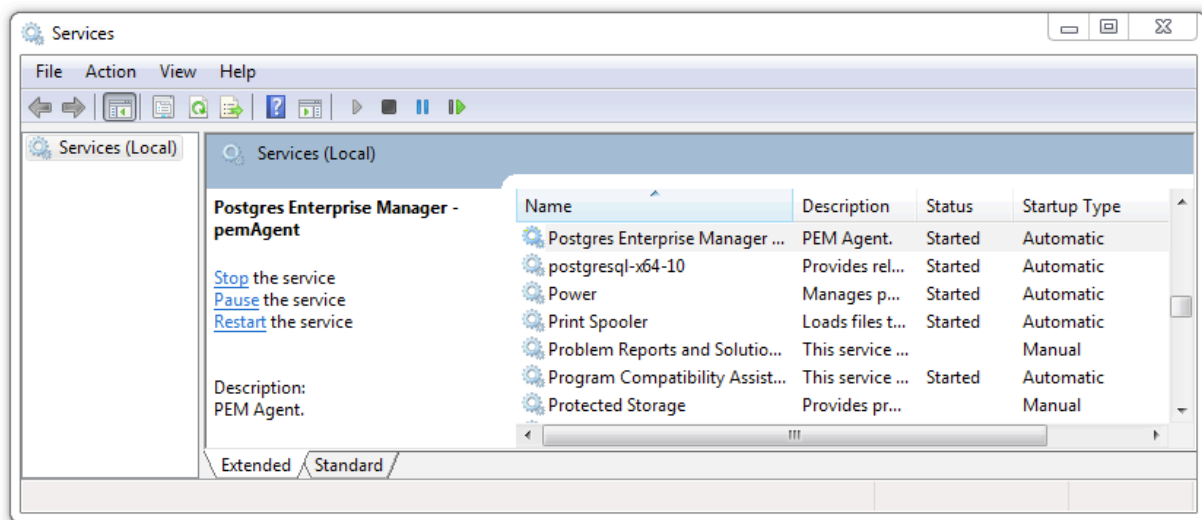


図 3 Windows サービスウィンドウ

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。サービスを停止すると、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）は突然切断されます。

Use the `Pause the service` option to instruct Postgres to reload a service's configuration parameters. The `Pause the service` option is an effective way to reset parameters without disrupting user sessions for many of the configuration parameters.

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。

5.5 HTTPD サーバーの制御

Linux では、コマンドラインを開いて次のコマンドを入力することで、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 RHEL6.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 RHEL7.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
```

On Windows, you can use the Services applet to check the status of the PEM HTTPD service. After opening the Services applet, scroll through the list to locate the PEM HTTPD service.

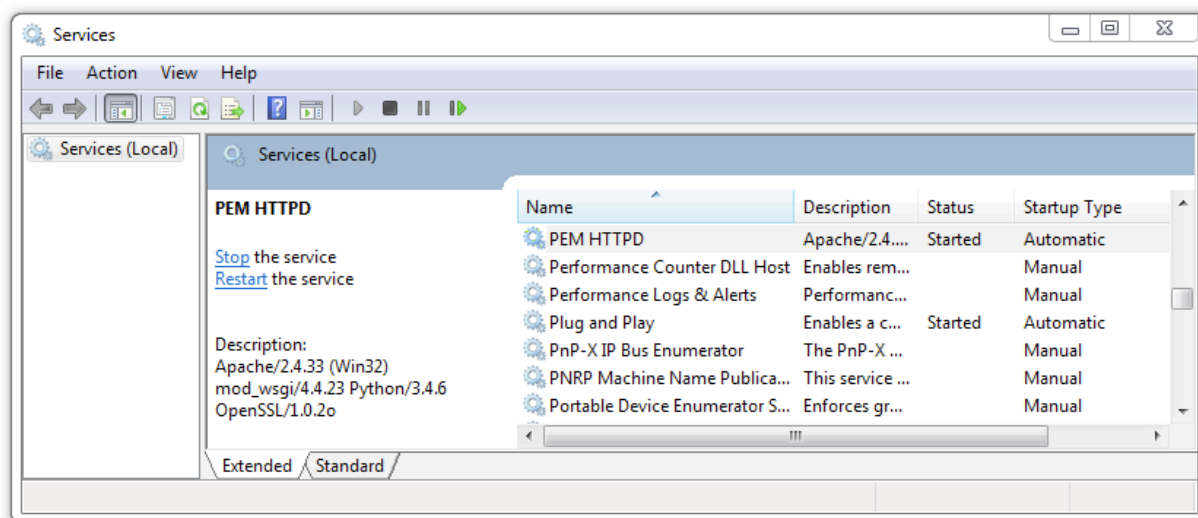


図 4 PEMHTTPDWindows サービス

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。

5.6 pg_hba.conf ファイルの変更

Entries in the `pg_hba.conf` file control network authentication and authorization. The `pg_hba.conf` file on the PEM server host must allow connections between the PEM server and PEM-HTTPD, the PEM agent, and the monitored servers.

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。

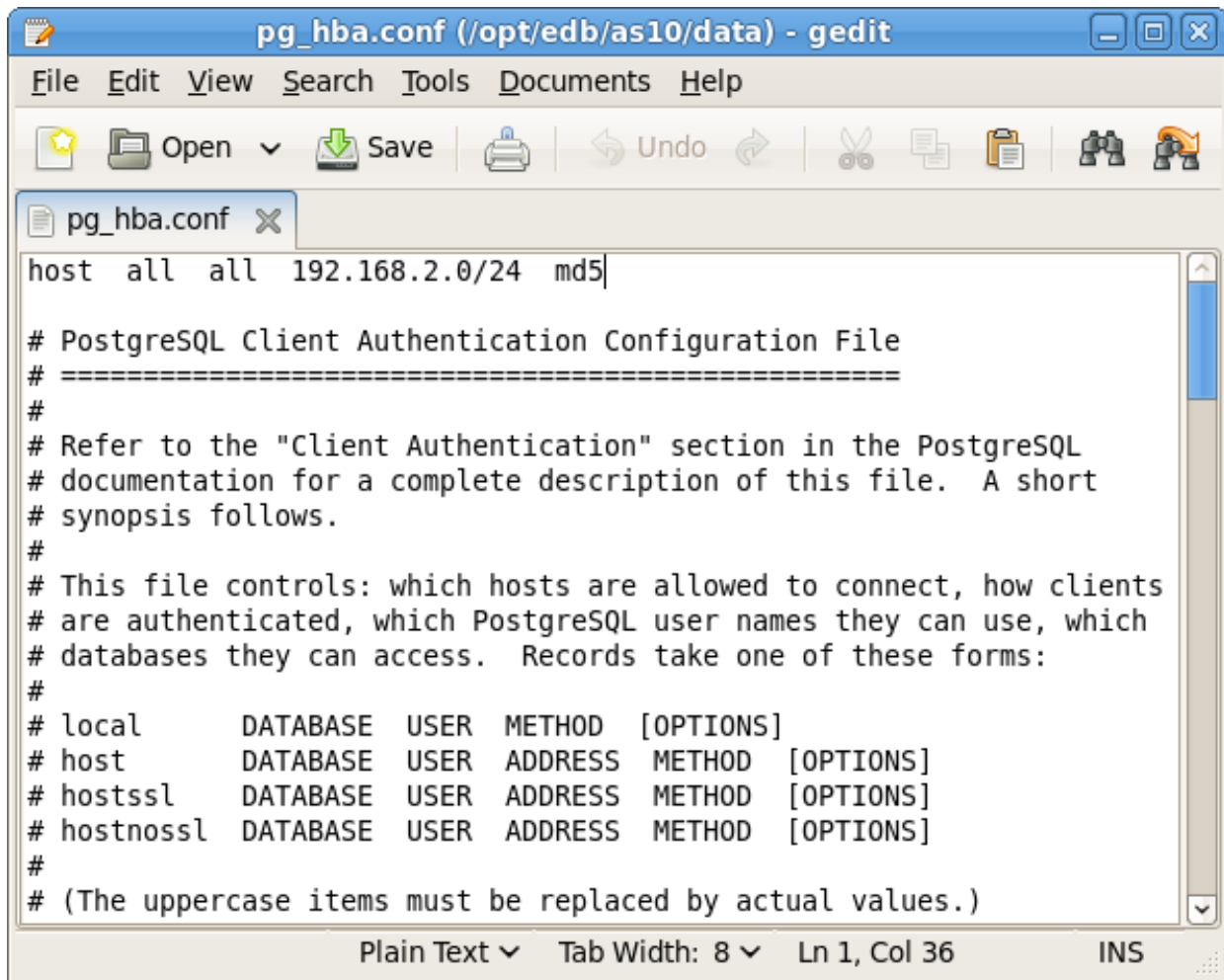


図 5 `pg_hba.conf` ファイルの PEM エントリ

You may also need to manually modify the `pg_hba.conf` file to allow connections between the PEM server and other components. For example, if your PEM-HTTPD installation does not reside on the same host as the PEM server, you must modify the `pg_hba.conf` file on the PEM server host to allow PEM-

HTTPD to connect to the server.

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、AdvancedServer10 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。

```
/var/lib/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/current/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。

5.7 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）のすべてを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。

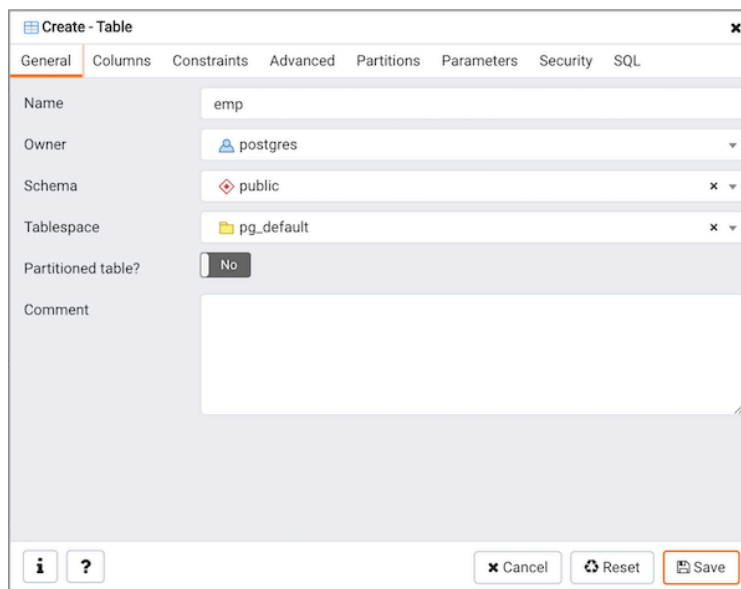


図 6 テーブルの作成-一般タブ

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。

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

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

5.9 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	☒ True	t/f
auto_create_server_alerts	☒ True	t/f
bart_log_retention_time	30	days
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_connovervw_timeout	300	seconds
dash_db_eventlag_span	7	days
dash_db_eventlag_timeout	1800	seconds
dash_db_hottable_rows	25	rows
dash_db_hottable_timeout	300	seconds
dash_db_io_span	168	hours
dash db io timeout	1800	seconds

Buttons: ? Cancel Reset Save

図 7 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。

5.10 セキュリティの管理

PEM は、Postgres のロール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。

5.10.1 ログインロール

PEM サーバーに接続するとき、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、PEMWeb 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。

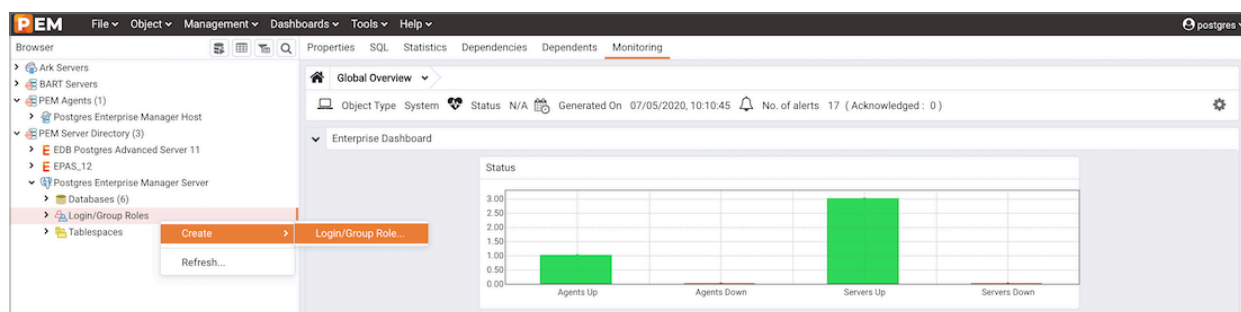


図 8 ログインロールノードのコンテキストメニュー

役割を定義するには、Create - Login/Group 役割ダイアログのタブのフィールドを使用します。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。

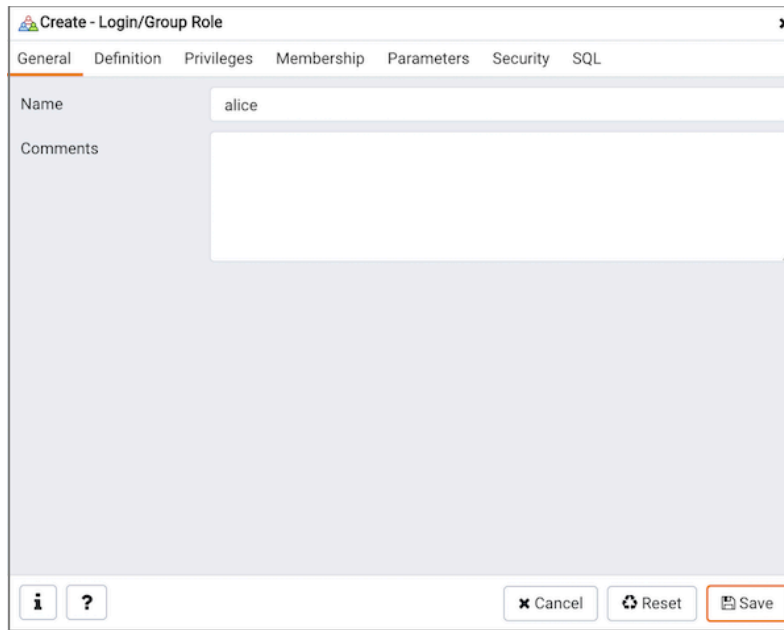
The image shows a window titled "Create - Login/Group Role" with a close button (X) in the top right corner. Below the title bar is a tabbed interface with tabs labeled "General", "Definition", "Privileges", "Membership", "Parameters", "Security", and "SQL". The "General" tab is currently active. In this tab, there is a "Name" label followed by a text input field containing the text "alice". Below the "Name" field is a "Comments" label followed by a large, empty text area. At the bottom of the window, there are three buttons: an information icon (i) and a question mark icon (?) on the left, and "Cancel", "Reset", and "Save" buttons on the right. The "Save" button is highlighted with a red border.

図 9 ログインロールの作成-[全般] タブ

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/current/sql-createrole.html>

5.10.2 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/current/sql-createrole.html>

5.10.3 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。

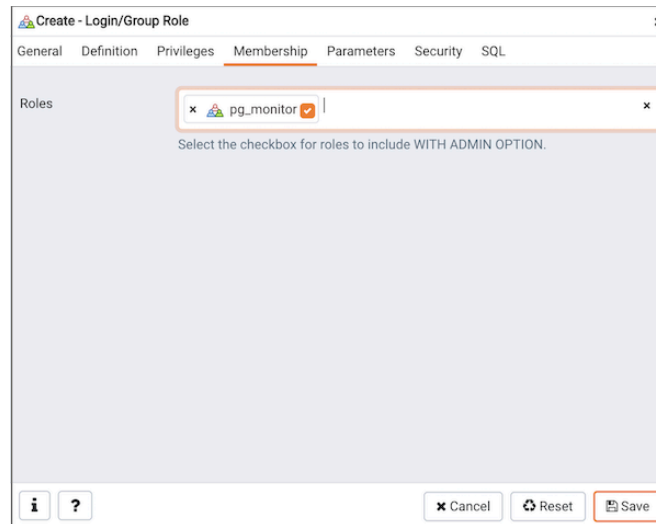


図 10 ログインロールの作成-[メンバーシップ] タブ

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。

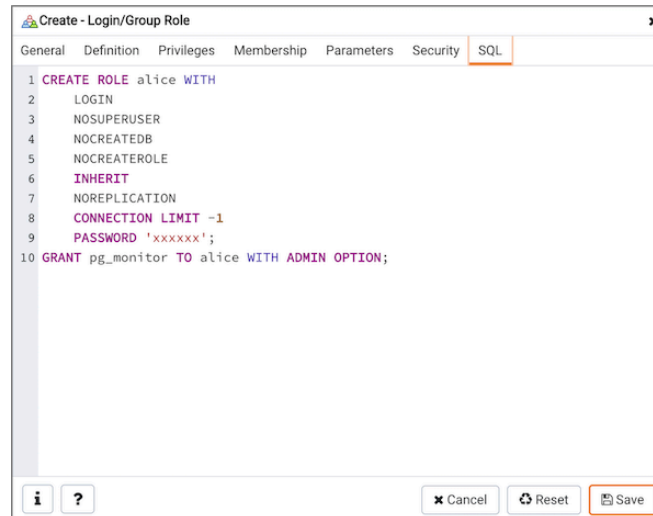


図 11 ログインロールの作成-[SQL] タブ

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。役割は、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		PostgresEnterpriseManager 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	PostgresEnterpriseManager 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	RESTAPI 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のプローブ設定 (履歴保持、実行頻度、プローブの有効化/無効化) の役割。
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	PostgresExpert 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	CapacityManager を実行する役割。
pem_comp_log_manager	pem_component	LogManager を実行するロール。

5.10.4 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ロール
- 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。

5.10.5 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。

オブジェクトの権限は、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。

5.11 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 スケジュール済みジョブの作成`](#)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）

5.11.1 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.

Buttons: Cancel, Reset, Save

図 12 ジョブ通知の設定-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。

- [通知を送信] 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。
- ***Emailgroup***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。

5.11.2 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。

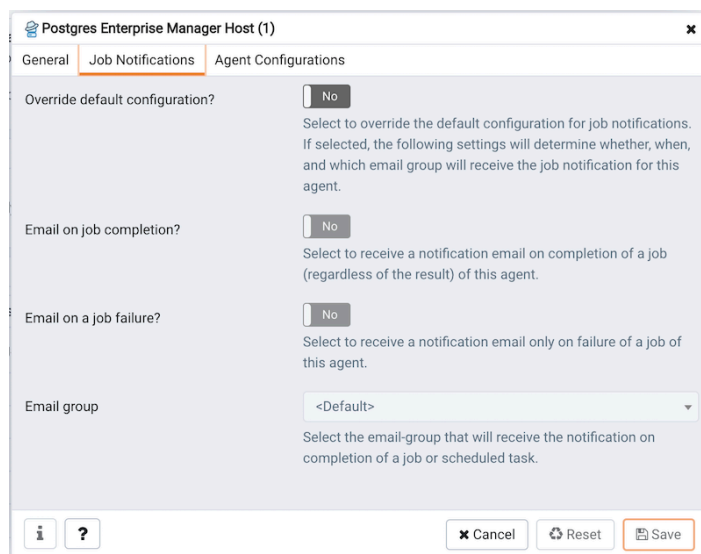


図 13 ジョブ通知の設定-エージェントレベル

[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。

- ***デフォルト設定のオーバーライド？***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オーバーライドするかどうかを指定します。このスイッチで [はい] を選択した場合、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、***Overridedefaultconfiguration？***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、ジョブ完了時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。

5.11.3 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 at the top right labeled 'Search by parameter name'. Below the search bar is a table of configuration parameters. The 'job_status_change_notification' row is highlighted with a red border. At the bottom of the dialog are buttons for '?', 'Cancel', 'Reset', and 'Save'.

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	True	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
nagios_medium_alert_as_critical	False	t/f
nagios_spool_retention_time	7	days
probe_log_retention_time	30	days
reminder_notification_interval	24	hours

図 14 ジョブ通知の設定-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。

- ***job_failure_notification***スイッチを使用して、各ジョブの失敗後に電子メール通知を送信するかどうかを指定します。
- ***job_notification_email_group***パラメーターを使用して、メール通知を受信するメールグループを指定します。
- ***job_retention_time** パラメーター*を使用して、非定期的なスケジュール済みタスクをシステムに保持する日数を指定します。
- ***job_status_change_notification***スイッチを使用して、ジョブのステータスが変更された後に、そのステータスが失敗、成功、中断のいずれであっても、電子メール通知を送信するかどうかを指定します。

5.12 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。ビジネス要件に合わせてこれらのジョブを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。

When the `Create - Agent Job` dialog opens, use the tabs on the `Create - Agent Job` dialog to define the steps and schedule that make up a PEM scheduled job.

The screenshot shows the 'Create - Agent Job' dialog box with the 'General' tab selected. The dialog has five tabs: 'General', 'Steps', 'Schedules', 'Notifications', and 'SQL'. The 'General' tab contains the following fields:

- Name:** A text input field containing 'Job_backup_pem'.
- Enabled?:** A toggle switch set to 'Yes'.
- Comment:** A text area containing 'Job for taking backup of pem database'.

At the bottom of the dialog, there are three buttons: 'Cancel', 'Reset', and 'Save' (highlighted with an orange border). There are also information and help icons on the bottom left.

図 15 エージェントジョブの作成-[全般] タブ

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 位置に移動し、ジョブを無効にするには No スイッチを移動します。
- 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。

The screenshot shows the 'Create - Agent Job' dialog box with the 'Steps' tab selected. A table lists the job steps, with 'Step1_backup' being the only one. Below the table, the 'General' sub-tab is active, displaying configuration fields for the selected step. The fields are: Name (Step1_backup), Enabled? (Yes), Kind (SQL), On error (Success), Server (Postgres Enterprise Manager Server (192.168.1.19:5432)), Database (pem), and Comment (empty). At the bottom of the dialog are buttons for 'Cancel', 'Reset', and 'Save'.

図 16 エージェントジョブの作成-[ステップ] タブ

Steps タブを使用して、ジョブが実行するステップを定義および管理します。[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。

The screenshot shows a 'Create - Agent Job' window with the following details:

- General Tab:**
 - Name:** Job_backup_pem_db
 - Enabled?:** Yes (checked)
 - Comment:** Job for taking backup of pem database
- Buttons:** Cancel, Reset, Save (highlighted with an orange border)

図 17 エージェントジョブの作成-[ステップ] タブ-[全般] タブ

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) を呼び出すかを指定します。
 - 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。

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 the 'Create - Agent Job' dialog box with the 'Steps' tab selected. A table lists the steps, with 'step1_backup' selected. Below the table, the 'Code' sub-tab is active, showing configuration fields for the selected step.

Name	Enabled?	Kind	On error
step1_backup	True	SQL	Success

Below the table, the 'Code' sub-tab is active, showing the following fields:

- Name: step1_backup
- Enabled?: Yes
- Kind: SQL
- On error: Success
- Server: Postgres Enterprise Manager Server (192.168.1.19:5432)
- Database: pem
- Comment: (empty text area)

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

図 18 エージェントジョブの作成-[ステップ]タブ-[コード]タブ

- 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。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、任意のシェルスクリプトを使用できます (例: `#!/bin/sh`)。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。

Linux システムでスクリプトを呼び出すには、agent.cfg ファイルの `*batch_script_user*` パラ

メーターのエントリを変更し、スクリプトの実行に使用するユーザーを指定する必要があります。このパラメーターには、非 root ユーザーまたは root を指定できます。ユーザーを指定しない場合、または指定したユーザーが存在しない場合、スクリプトは実行されません。ファイルを変更した後、エージェントを再起動します。

Windows システムでスクリプトを呼び出すには、***AllowBatchJobSteps***のレジストリエントリを **true** に設定し、**PEM** エージェントを再起動します。**PEM** レジストリエントリは `HKEY_LOCAL_MACHINE/Software/Wow6432Node/EnterpriseDB/PEM/agent` にあります。

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。

Create - Agent Job

General Steps **Schedules** Notifications SQL

Name	Enabled?	Start	End
Backup_pem_db_schedule	☒ True	2020-04-01 23:30 +05:30	2020-04-02 23:30 +05:30

General Repeat Exceptions

Name: Backup_pem_db_schedule

Enabled?: ☒ Yes

Start: 2020-04-01 23:30 +05:30

End: 2020-04-02 23:30 +05:30

Comment:

Cancel Reset **Save**

図 19 エージェントジョブの作成-[スケジュール] タブ-[全般] タブ

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。
- ***Enabled***スイッチを使用して、pgAgent がスケジュールを使用する（Yes）か、スケジュールを無効にする（No）ことを示します。
- Start フィールドのカレンダーセレクトターを使用して、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ターを使用して、スケジュールの終了日時を指定します。
- スケジュール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。

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。

Create - Agent Job

General Steps **Schedules** Notifications SQL

Name	Enabled?	Start	End
Backup_pem_db_schedule	True	2020-04-01 23:30 +05:30	2020-04-02 23:30 +05:30

General **Repeat** Exceptions

Schedules are specified using a **cron-style** format.
 For each selected time or date element, the schedule will execute.
 e.g. To execute at 5 minutes past every hour, simply select '05' in the Minutes list box.
 Values from more than one field may be specified in order to further control the schedule.
 e.g. To execute at 12:05 and 14:05 every Monday and Thursday, you would click minute 05, hours 12 and 14, and weekdays Monday and Thursday.
 For additional flexibility, the Month Days check list includes an extra Last Day option. This matches the last day of the month, whether

Days

Week Days: Select the weekdays...

Month Days: × 1st ×

Months: Select the months...

Times

Hours: Select the hours...

Minutes: Select the minutes...

Cancel Reset Save

図 20 エージェントジョブの作成-[スケジュール] タブ-[繰り返し] タブ

Use the fields on the Repeat tab to specify the details about the schedule in a cron-style format. The job will execute on each date or time element selected on the Repeat tab.

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、ジョブが月の最後の日に実行されることを示すには、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。

The screenshot shows the 'Create - Agent Job' dialog box with the 'Schedules' tab selected. Below the main schedule table, the 'Exceptions' sub-tab is active, displaying a table with one exception entry.

Name	Enabled?	Start	End
☒ Backup_pem_db_schedule	☒ True	2020-04-01 23:30 +05:30	2020-04-02 23:30 +05:30

Exceptions	
Date	Time
☒ 2020-04-01	00:00

図 21 エージェントジョブの作成-[スケジュール] タブ-[例外] タブ

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、国民の祝日にはジョブを実行しないようにすることができます。

[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てから、次の操作を行います。

- Click within the `Date` column to open a calendar selector, and select a date on which the job will not execute. Specify `<Any>` in the `Date` column to indicate that the job should not execute on any day at the time selected.
- Click within the `Time` column to open a time selector, and specify a time on which the job will not execute. Specify `<Any>` in the `Time` column to indicate that the job should not execute at any time on the day selected.

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:

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.

Buttons: **Cancel** **Reset** **Save**

図 22 エージェントジョブの作成-[通知] タブ

`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。

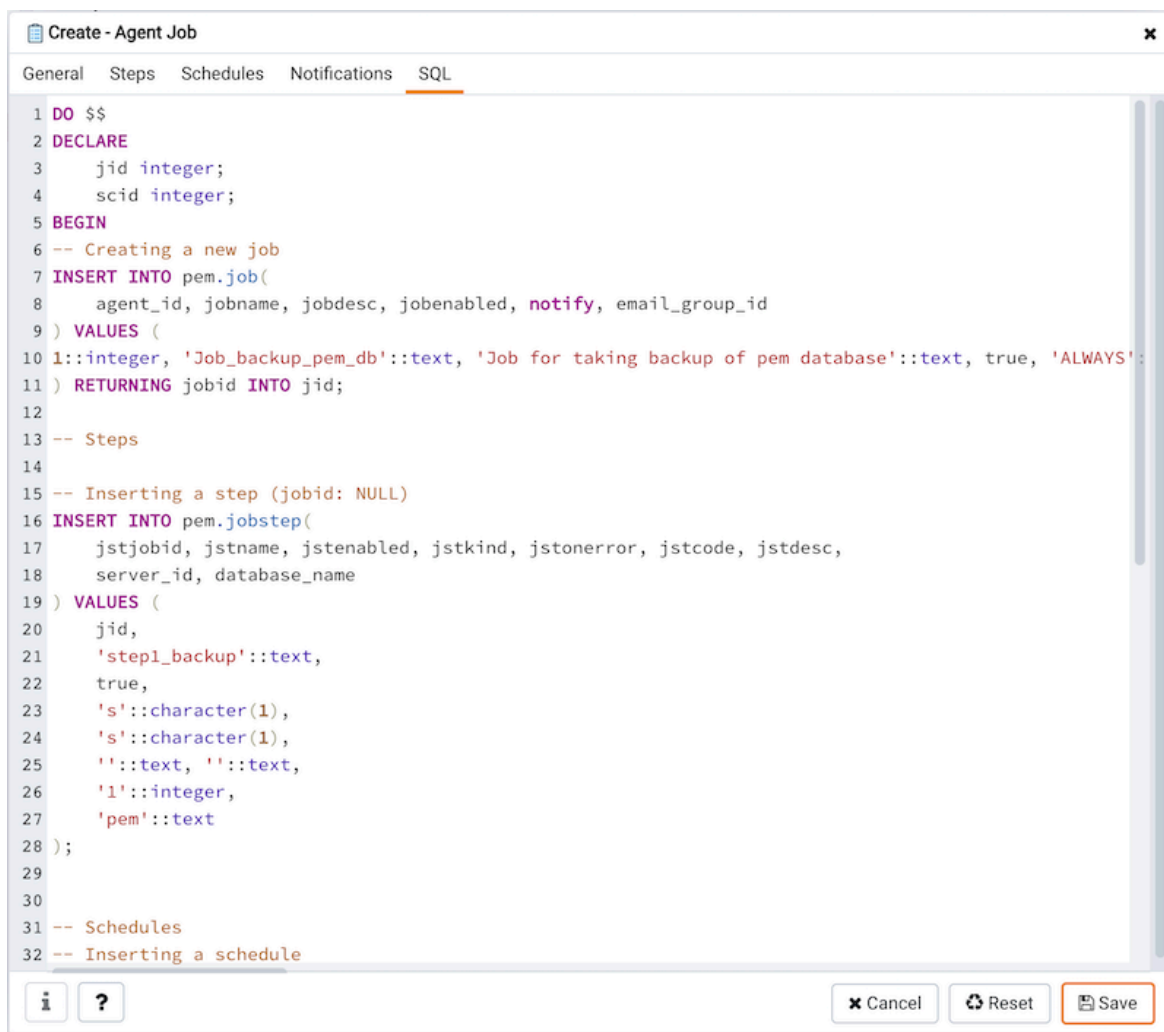


図 23 エージェントジョブの作成-[SQL] タブ

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。

After saving a job, the job will be listed under the Jobs node of the PEM tree control of the server on which it was defined. The Properties tab in the PEM console will display a high-level overview of the

selected job, and the Statistics tab will show the details of each run of the job. To modify an existing job or to review detailed information about a job, right-click on a job name, and select `Properties` from the context menu.

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。

6.1 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無制限のプローブを呼び出し、PEM サーバーに情報を取得して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 エキスパート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。
PGHBA	Linux/Windows	Postgres エキスパート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、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は表示されません。
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 関連のチャート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。

プローブが不十分な権限で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権限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する場合があります。

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 ワーカーログファイルの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。

6.2 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。一部の PEM 機能を有効にするには、PEM エージェントのパラメーター設定を変更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。

On Linux systems, PEM configuration options are stored in the `agent.cfg` file, located in `/usr/edb/pem/agent/etc`. The `agent.cfg` file contains the following entries:

パラメータ名	説明	デフォルト値
<code>pem_host</code>	PEM サーバーの IP アドレスまたはホスト名。	127.0.0.1。
<code>pem_port</code>	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。
<code>pem_agent</code>	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。
<code>agent_ssl_key</code>	PEM エージェントのキーファイルへの完全なパス。	<code>/root/.pem/agent.key</code>
<code>agent_ssl_cert</code>	PEM エージェントの証明書ファイルへの完全なパス。	<code>/root/.pem/agent.crt</code>
<code>agent_flag_dir</code>	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。
<code>log_level</code>	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
<code>log_location</code>	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存在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、複数の PEM エージェントおよび PEM バックエンドデータベースサーバーが 9.4 以下の場合、重複した電子メールを送信する可能性があります（PEM エージェント 7.14 以降でも）。また、少なくとも 1 つの PEM エージェントが 7.14 以上である複数のエージェントに対して true に設定した場合、重複した電子メールを送信する可能性があります。	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、複数の PEM エージェントおよび PEM バックエンドデータベースサーバーが 9.4 以下の場合、重複した SNMP トラップを送信する場合があります（PEM エージェント 7.14 以降でも）。また、少なくとも 1 つの PEM エージェントが 7.14 以上である複数のエージェントに対して true に設定した場合、重複したトラップを送信する可能性があります。	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。

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
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
max_connections	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）
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（接続は、エージェントの処理ループの後にアイドル状態になると切断されます）。
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接続が使用されます。	偽
batch_script_dir	スクリプトファイル（アラート用）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。	デフォルトでは設定されていません。

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
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存在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
MaxConnections	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）
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（接続は、エージェントの処理ループの後にアイドル状態になると切断されます）。
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接続が使用されます。	偽
BatchScriptDir	スクリプトファイル（アラート用）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。

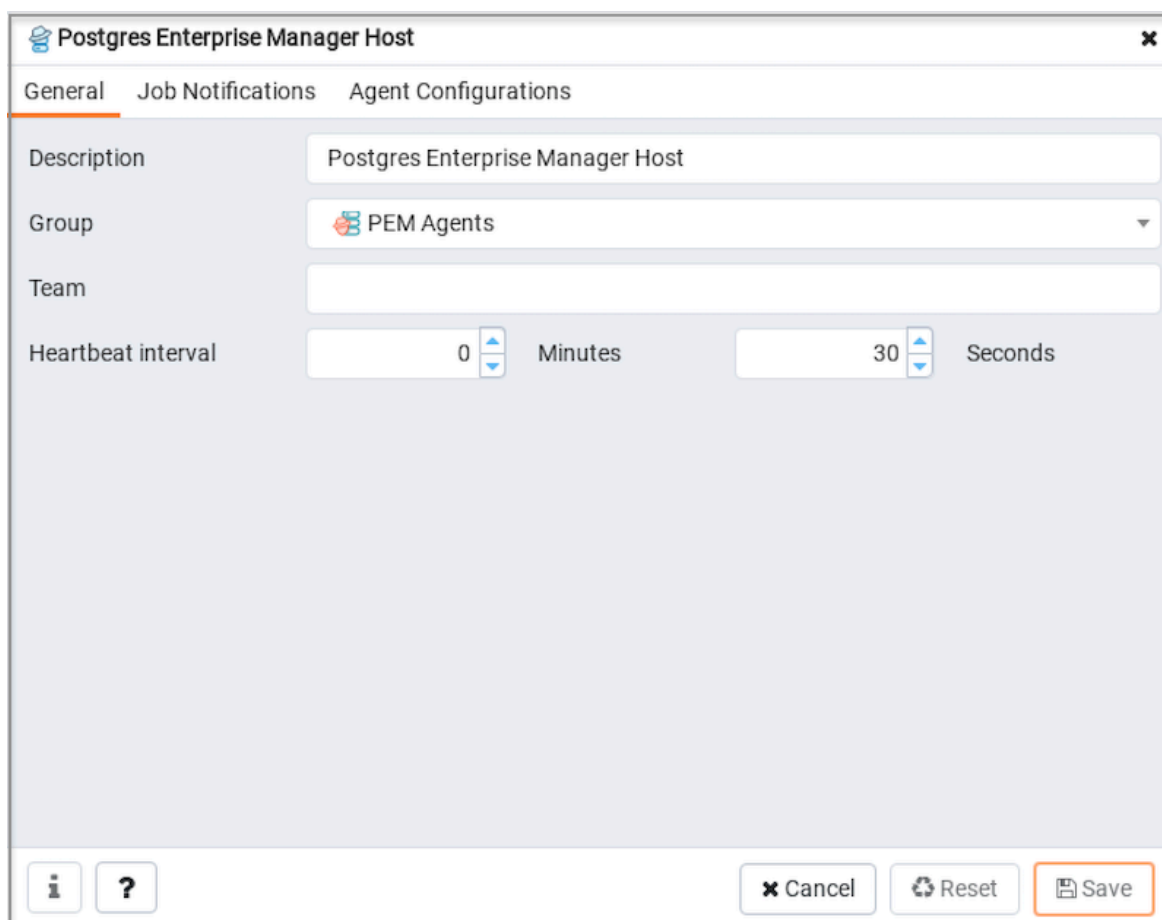
次のページに続く

表 2 – 前のページからの続き

ca_file	CA 証明書が存在するパスを指定します。	デフォルトでは設定されていません。
---------	----------------------	-------------------

6.3 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。



The screenshot shows a dialog box titled "Postgres Enterprise Manager Host". It has three tabs: "General", "Job Notifications", and "Agent Configurations". The "General" tab is selected. The dialog contains the following fields:

- Description:** A text field containing "Postgres Enterprise Manager Host".
- Group:** A dropdown menu showing "PEM Agents".
- Team:** An empty text field.
- Heartbeat interval:** Two spinners. The first is set to "0" and is labeled "Minutes". The second is set to "30" and is labeled "Seconds".

At the bottom of the dialog, there are three buttons: "Cancel", "Reset", and "Save". There are also information and help icons on the left.

図 1 PEM エージェントプロパティダイアログ

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で経過する時間の長さを表示します。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 7

結論

PostgresEnterpriseManager 入門ガイド

Copyright©2013-2020EnterpriseDBCorporation。無断複写・転載を禁じます。

EnterpriseDB®Corporation34CrosbyDrive, Suite201, Bedford, MA01730, USA

T+17813573390F+19784671307Einfo@enterprisedb.comwww.enterprisedb.com

- EDB は、PostgresEnterpriseManager 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, 95
Agent Privileges, 92
Agent Properties, 101
agent ssl certificate, 52

control server remotely, 56
Control service on Linux, 57
Control service on Windows, 58
Creating PEM scheduled jobs, 80

Group Roles, 69

HTTPD server, 59

Login Roles, 68

PEM
 architecture, 5
 installlation, 7
 overview, 1
PEMOverview, 2
PEMWeb インターフェイスの使用, 8
PEM エージェントの管理, 92
PEM サーバーの管理, 55
PEM のインストール-概要, 7
pg_hba.conf file, 60

replacing, 48
replacing ssl certificates, 48
restart PEM agent, 55
restart PEM service, 55

Security, 67
Sending email notifications for a job, 76
ssl certificates, 48
starting service, 55

update agent ssl certificate, 52

サーバーの登録, 23

一般的なアーキテクチャ, 5
結論, 103
証明書の管理, 47
設定の管理, 54