



Postgres Enterprise Manager

リリース **7.13**

PEM Installation Guide

2020 年 03 月 01 日

目次

第 1 章	新機能	2
第 2 章	Postgres Enterprise Manager-概要	3
2.1	ハードウェアの前提条件	5
2.2	ソフトウェアの前提条件	6
第 3 章	Postgres Enterprise Manager のインストール	7
3.1	Windows への PEM サーバーのインストール	8
3.1.1	PEM サーバーと PEM-HTTPD を同じホストにインストールする	10
3.1.2	PEM サーバーと PEM-HTTPD を別々のホストにインストールする	18
3.1.2.1	データベースホストの指定	22
3.1.2.2	Web サービスのインストール	28
3.1.3	PEM サーバーを既存の Postgres サーバーにインストールする	33
3.1.3.1	Postgres サーバーの準備	33
3.1.3.1.1	sslutils 拡張機能のインストール	33
3.1.3.1.2	サービススクリプトの作成またはサービスの登録	35
3.1.3.2	PEM サーバーインストーラーの呼び出し	36
3.1.4	コマンドラインからサーバーインストーラーを呼び出す	45
3.1.4.1	無人モードでの PEM サーバーインストーラーの呼び出し	45
3.2	Linux での PEM サーバーのインストール	48
3.2.1	RHEL または CentOS HOST の追加の前提条件	48
3.2.1.1	CentOS または RHEL ホストへの PEM サーバーのインストール	50
3.2.1.2	Debian または Ubuntu ホストへの PEM サーバーのインストール	50
3.2.1.3	SLES ホストへの PEM サーバーのインストール	51
3.2.1.4	PEM サーバーの構成	52
3.3	隔離されたネットワークでの PEM リポジトリの作成	54
3.4	Windows での PEM エージェントのインストール	56

3.4.1	コマンドラインからのエージェントインストーラーの呼び出し	63
3.4.1.1	無人モードでの PEM エージェントインストーラーの呼び出し	63
3.5	Linux での PEM エージェントのインストール	65
3.5.1	CentOS または RHEL ホストへの PEM エージェントのインストール	65
3.5.2	Debian または Ubuntu ホストへの PEM エージェントのインストール	67
3.5.3	SLES ホストへの PEM エージェントのインストール	67
3.5.4	エージェントの登録	68
第 4 章	PEM Web インターフェイス	72
第 5 章	SQL プロファイラープラグインのインストール	75
5.1	グラフィカルインストーラーを使用した SQL プロファイラーのインストール	77
5.2	RPM を使用した SQL プロファイラーのインストール	81
5.3	SQL プロファイラーの構成	81
第 6 章	PEM コンポーネントのアップグレードとアンインストール	83
第 7 章	リファレンス-Linux サービススクリプト	87
第 8 章	結論	94
索引		95

Postgres Enterprise Manager (PEM) は、PostgreSQL および 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を行う際に、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ように設計されています。PEM は、単一のコンソールから単一のサーバーまたは複数のサーバーを管理および監視し、監視対象データベースを完全に制御できるように設計されています。

このドキュメントは、Postgres Enterprise Manager のインストールをガイドするステップバイステップの手順を提供します。

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 EnterpriseDB Web 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>

このガイド全体で、* Postgres * という用語は、PostgreSQL または Advanced Server のインストールのいずれかを指します（どちらか適切な場合）。

言語パックインストーラーには、EDB Postgres Advanced Server および EnterpriseDB PostgreSQL データベースインストーラーで利用できるサポートされている言語が含まれています。言語パックインストーラーを使用すると、サードパーティベンダーのサポートソフトウェアをインストールせずに、Perl、TCL / TK、および Python をインストールできます。言語パックのインストールと使用の詳細については、EnterpriseDB Web サイトから入手できる* EDB Postgres 言語パックガイド*を参照してください。

新機能

Postgres Enterprise Manager 7.13 を作成するために、次の機能が追加されました。

- コア使用状況レポート:コア使用状況レポートは、EDM Postgres Advanced Server または PEM によって管理されている Postgres データベースで使用するコアの総数などのメトリックを提供します。このレポートは、コアベースのライセンスガイドラインに準拠しているかどうかを理解するのに役立ちます。
- SMMP v3 サポート:SNMP v3 サポートにより、PEM サーバーおよび PEM エージェントは、ネットワーク上でデータパケットを認証および暗号化することにより、安全な通信のために SNMP バージョン 3 を使用できます。
- スキーマ差分ツール（ベータ）:スキーマ差分ツールを使用すると、2 つのデータベーススキーマオブジェクトを比較して、2 つのスキーマを同期できます。このツールは違いをリストし、2 つのスキーマの同期スクリプトも生成します。
- 非アクティブなユーザーの自動ログアウト:非アクティブなユーザーセッションは、一定の非アクティブ期間が経過すると自動的にログアウトされます。このタイムアウト間隔は config.py ファイルで設定できます。この機能は、データベースサーバーへの意図しないアクセスを減らすことにより、セキュリティを向上させます。
- アクセシビリティの改善:スクリーンリーダーは、非テキスト要素のラベルと説明を読み取り、アラートエラーを識別し、ネストされた要素の関係属性を読み取ることができます。
- その他の機能は次のとおりです。
 - システム構成レポートの機能強化。
 - マルチレベルパーティションテーブルのサポート。

第 2

Postgres Enterprise Manager-概要

Postgres Enterprise Manager (PEM) は、PEM の管理および分析機能を提供する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、データを監視するためのデータリポジトリとして、およびエージェントとクライアントの両方が接続するサーバーとして使用されます。 PEM サーバーは、PostgreSQL のインスタンスと、監視データを保存するための関連データベース、および Web サービスを提供するサーバーで構成されています。
- **PEM Web インターフェイス:** PEM Web インターフェイスを使用すると、Postgres サーバーを管理および監視し、PEM 拡張機能を利用できます。 Web インターフェースソフトウェアは PEM サーバーインストーラーでインストールされ、選択した Web ブラウザーからアクセスします。
- **PEM エージェント:** PEM エージェントは、タスクの実行と、エージェントホストおよび監視対象の Postgres インスタンスから PEM サーバーへの統計のレポートを担当します。単一の PEM エージェントは、1 つ以上のホストに存在する Postgres の複数のインストール済みインスタンスを監視できます。

PEM エージェントインストーラーは、PEM ワーカー (pemworker.exe) と PEM エージェント (pemagent.exe) の 2 つの実行可能ファイルを作成します。各 PEM ワーカーには、PEM ワーカーの起動または停止に使用できる対応する PEM エージェントがあります。 PEM エージェントは、予期せず終了した場合にも PEM ワーカーを再起動します。

PEM ワーカーのログファイルには、PEM ワーカーのアクティビティ (プローブアクティビティ、ハートビート応答など) に関連する情報が含まれ、 /var/log/pem/worker.log に保存されます。

- **SQL Profiler プラグイン:** Postgres サーバーへのこのプラグインは、SQL Profiler ツールで使用される監視データを生成するために使用されます。 SQL Profiler プラグインのインストールはオプションですが、プラ

グインはプロファイルする Postgres の各インスタンスにインストールする必要があります。SQL Profiler は、PostgreSQL サーバーまたは Advanced Server（PEM サーバーで管理されているものだけでなく）の EnterpriseDB ディストリビューションのサポートされているバージョンで使用できます。SQL プロファイラプラグインのインストールを参照してください。詳細およびサポートされているバージョン。

次のアーキテクチャ図は、一般的な PEM インストールに関係するさまざまなサーバーとワークステーションの関係を示しています。

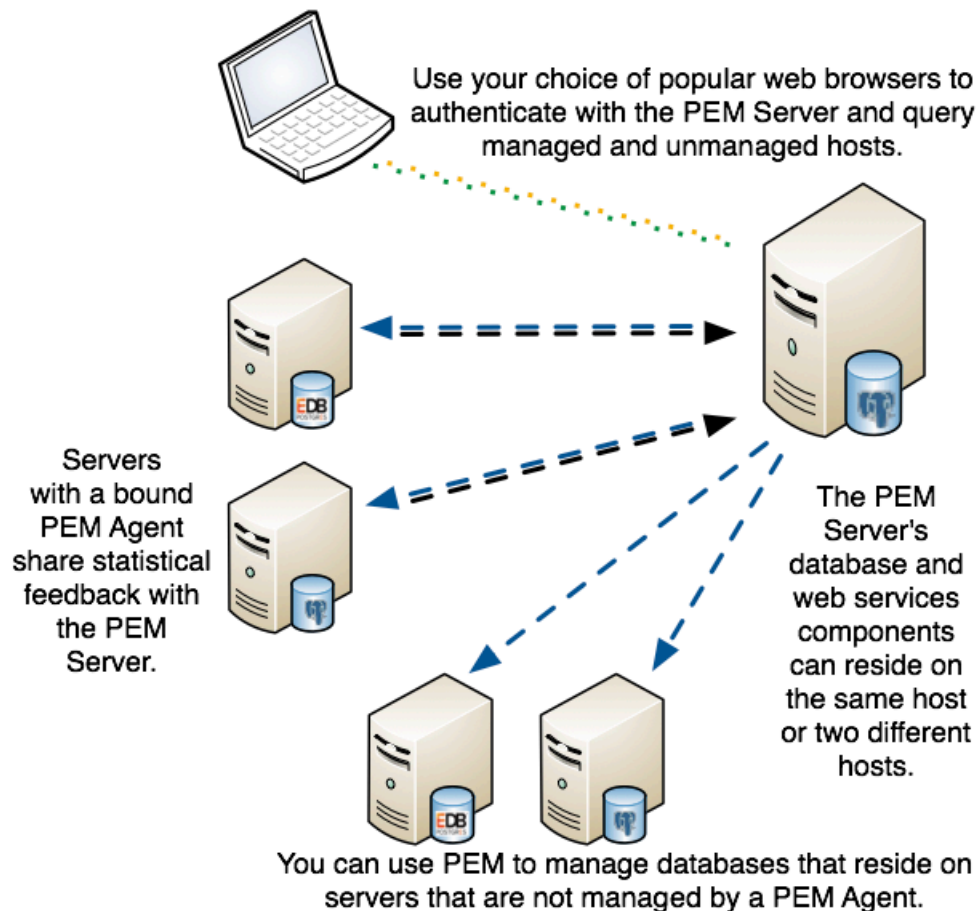


図 1 典型的な PEM インストール

2.1 ハードウェアの前提条件

サーバーを監視し、ダッシュボードをレンダリングするときに最適な速度を得るには、少なくとも次のシステムに PEM をインストールすることをお勧めします。

- 4 つの CPU コア
- 8 GB の RAM
- 100 GB のストレージ

データストレージには追加のディスク領域が必要です。リソースの使用量は、どのプローブが定義および有効化されているか、監視対象データベースのアクティビティレベルに応じて異なることに注意してください。（PEM を使用して）サーバーリソースを監視すると、初期システム構成を拡張する必要がある場合に通知されます。

2.2 ソフトウェアの前提条件

pg_hba.conf ファイルの変更

pg_hba.conf ファイルは Postgres サーバーの接続を管理します。各監視対象サーバー上の pg_hba.conf ファイルが PEM サーバー、監視 PEM エージェント、および PEM-HTTPD サーバー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。

pg_hba.conf ファイルの変更については、次の Web サイトで入手可能な* PEM 管理者ガイド*を参照してください。

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

認証の管理に関する情報は、次の場所にある Postgres のコアドキュメントでも入手できます。

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

ファイアウォールの制限

PEM コンポーネント間の通信を許可するには、ファイアウォールを調整する必要があることに注意してください。

PEM エージェントのインストール-RHEL または CentOS 7.x の前提条件

CentOS 7.x ホストに PEM エージェントをインストールする前に、ホストに epel-release パッケージが含まれていることを確認する必要があります。パッケージをインストールするには、コマンドラインを開き、root 権限を引き受けて、コマンドを呼び出します:

```
yum install epel-release
```

Windows の権限

PEM コンポーネントを Windows にインストールする場合、Administrative 特権で PEM インストーラーを呼び出す必要があります。管理者権限を使用してインストーラーを起動するには、インストーラーアイコンを右クリックし、コンテキストメニューから Run as Administrator を選択します。

サポートされているロケール

現在、PEM サーバーと Web インターフェースは English (US) en_US のロケールと言語区切り文字としてのピリオド (.) の使用をサポートしています。代替ロケール、またはピリオド以外の区切り文字を使用すると、エラーが発生する場合があります。

Windows Server IE セキュリティ

Windows Server ホストで Internet Explorer を使用して PEM Web インターフェイスで監視対象サーバーにアクセスする場合、PEM 機能を利用するには Internet Explorer のセキュリティ強化を無効にする必要があります。IE セキュリティ強化を無効にたくない場合は、別のブラウザ (Chrome など) を使用することをお勧めします。

第 3

Postgres Enterprise Manager のインストール

グラフィカルインストーラー、StackBuilder Plus、または RPM パッケージを使用して、Postgres Enterprise Manager サーバーまたはエージェントをホストに追加できます。インストーラーは “EnterpriseDB Web サイト <<https://www.enterprisedb.com/download-postgres-enterprise-manager>>”。

Windows 用の PEM サーバーグラフィカルインストーラーは、PEM サーバー、PEM エージェント、および選択したブラウザで PEM Web インターフェイスに接続するために必要なソフトウェアをインストールおよび構成します。PEM サーバーのグラフィカルインストーラーの使用に関する詳細については、“[Windows での PEM サーバーのインストール](#)”を参照してください。。

RPM パッケージを使用して、Linux ホストに PEM サーバーをインストールできます。インストーラーの使用に関する詳細については、[Linux への PEM サーバーのインストール](#)を参照してください。。

Windows 用の PEM エージェントグラフィカルインストーラーは、PEM エージェントをインストールして登録します。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、同じホストまたはリモートホストにある複数のサーバーを監視できます。リモートエージェントによって監視されるサーバー上の PEM 機能が制限される場合があることに注意してください。

PEM エージェントグラフィカルインストーラーの使用の詳細については、“[Windows での PEM エージェントのインストール](#)”を参照してください。。

RPM パッケージを使用して、Linux に PEM エージェントをインストールできます。PEM エージェント RPM パッケージの詳細については、[Linux への PEM エージェントのインストール](#)を参照してください。。

3.1 Windows への PEM サーバーのインストール

各 PEM インストールの中心にあるのはサーバーです。実稼働環境では、サーバーは通常、専用のマシンであり、多数の Postgres サーバーまたは少数のビジーなサーバーを監視します。

PEM サーバーバックエンドデータベースは、PostgreSQL または Advanced Server データベースサーバーの EnterpriseDB ディストリビューション、または別のソースからインストールされた既存の Postgres サーバーです。Postgres バックアップデータベースサーバーはバージョン 9.4 以降である必要があります、PEM サーバーによってリポジトリとして使用される pem というデータベースが含まれます。

- 既存の Postgres サーバーを使用して PEM サーバーをホストする場合、PEM サーバーのインストーラーは Postgres ホストに pem データベースを作成できます。既存のサーバーを使用する場合は、ソフトウェアの前提条件を手動で満たす必要があります。

既存の Postgres サーバーを使用して PEM サーバーバックエンドデータベースをホストする方法の詳細については、“[既存の Postgres サーバーへの PEM サーバーのインストール](#)”を参照してください。セクション。

- Postgres の既存のインストールを PEM サーバーホストとして使用したくない場合、PEM サーバーインストーラーは PostgreSQL をインストールし、サーバーホストのソフトウェア前提条件を満たし、“[PEM サーバーインストール](#)”を含むインスタンス (PostgreSQL データベースクラスター) 0 '' データベース。

これは最も単純な PEM サーバーのインストールオプションです。

PEM-HTTPD は、PEM サーバーインストーラーまたは StackBuilder ユーティリティを使用して Postgres のインストールに使用できます。PEM-HTTPD がすでにホストにインストールされている場合、PEM サーバーインストーラーは必要に応じて既存のインストールを確認および更新します。PEM サーバーホストに既存の PEM-HTTPD インストールが含まれていない場合、PEM サーバーインストーラーはそれを追加します。

PEM サーバーをインストールする前に、PostgreSQL と PEM-HTTPD を同じホストで実行するか、別々のホストで実行するかを決定する必要があります。PostgreSQL データベースサーバーと PEM-HTTPD を異なるホストで実行する場合は、[PEM サーバーと PEM-HTTPD を別々のホストにインストールする](#)で説明されているように、[PEM サーバーインストーラー](#)を各ホストで 2 回実行する必要があります セクション。

PEM サーバーインストーラーは、PEM Web インターフェイスを介してサーバーにアクセスするために必要なソフトウェアもインストールします。選択したブラウザのサポートされているバージョンで Web インターフェイスにアクセスできます。

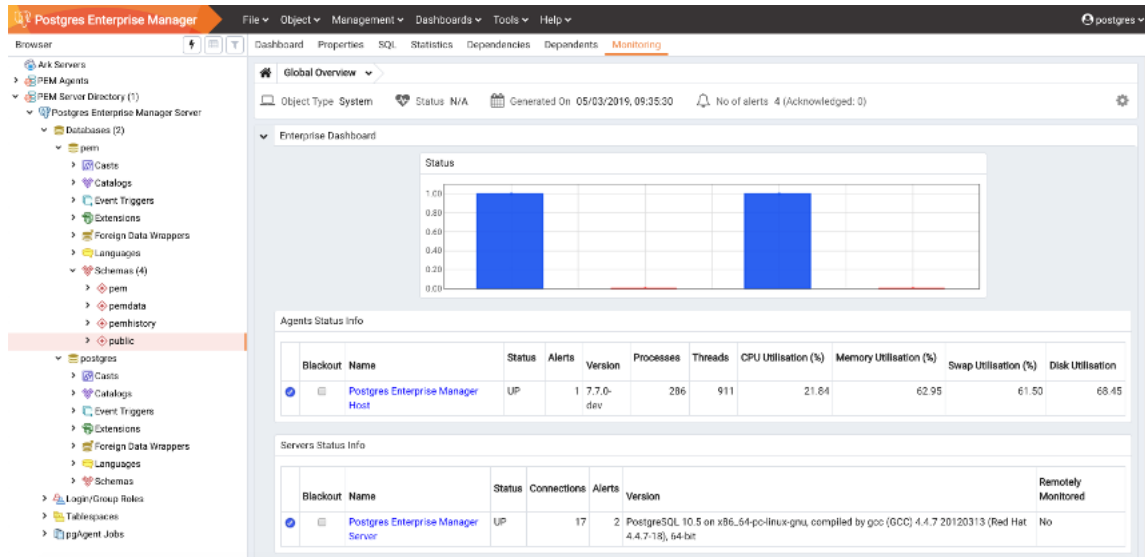


図 1 The PEM web interface

Web インターフェイスを使用して、監視対象サーバーにあるオブジェクトに関する情報を確認したり、監視対象サーバーにあるデータベースとデータベースオブジェクトを管理したり、PEM サーバーが収集した統計情報を確認したりできます。インターフェイスは、登録済みの EDB Ark コンソールにあるクラスターへのアクセスも提供します。

3.1.1 PEM サーバーと PEM-HTTPD を同じホストにインストールする

最も簡単な PEM サーバーのインストール構成は、同じホストにある PEM バックエンドデータベースサーバー（PEM サーバーインストーラーでインストールされた PostgreSQL データベースでホストされる）と PEM-HTTPD サービスで構成されます。この構成では、PEM サーバーインストーラーはバックエンドホストに必要なソフトウェアを提供し、サービススクリプトを作成（Linux の場合）またはサービスを登録（Windows の場合）します。

Windows システムで PEM サーバーインストーラーを起動するには、インストーラーアイコンを右クリックし、Run as Administrator を選択します。インストーラーは Welcome ダイアログを表示します。

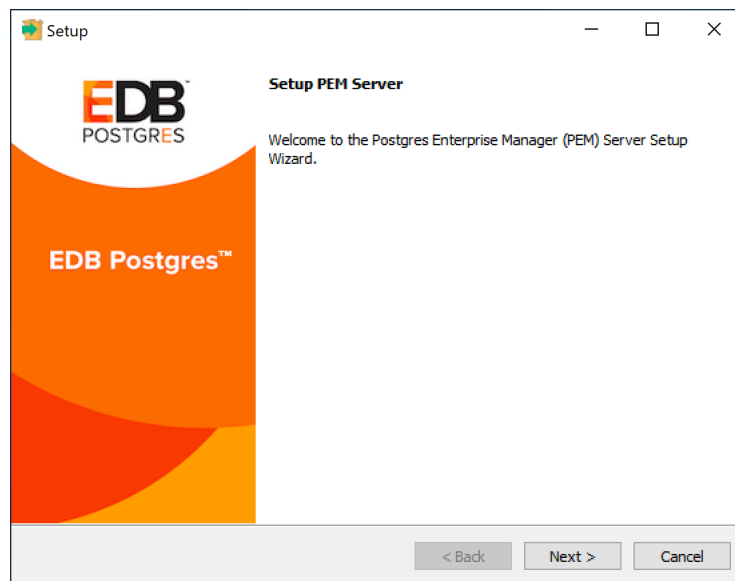


図 2 The PEM Server Installer's Welcome dialog

Next をクリックして、License Agreement ダイアログに進みます。



図 3 ライセンス契約

適切なラジオボタンを強調表示して契約に同意する前に、ライセンス契約を注意深く確認してください。Next をクリックして、Installation Directory ダイアログに進みます。

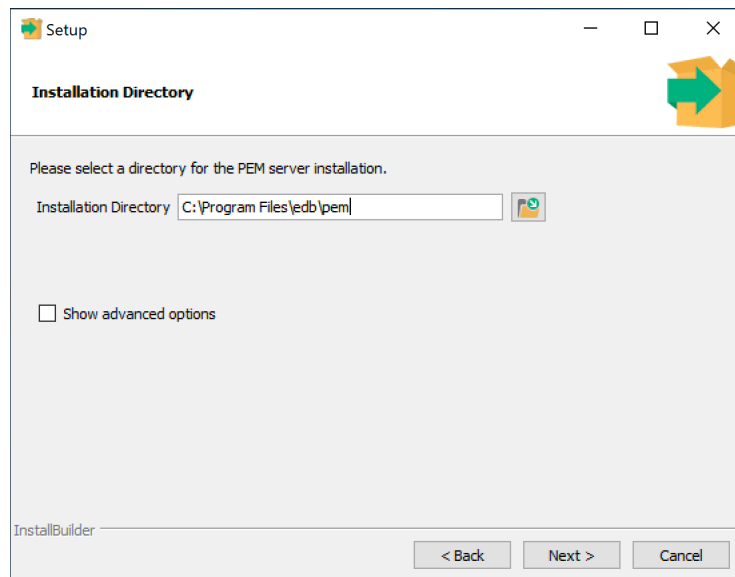


図 4 インストールディレクトリを指定

Installation Directory ダイアログを使用して、PEM サーバーの場所を指定します。

- デフォルトでは、PEM サーバーは Windows の C:/Program Files/edb/pem にインストールされます。デフォルトの場所を受け入れるか、Installation Directory ボタンを使用してブラウザダイアログを開き、PEM サーバーをインストールするディレクトリを選択します。

- Show advanced options チェックボックスを使用して、Advanced options ダイアログを開くようインストーラーに指示します。
- Postgres データベースサーバーと PEM-HTTPD を異なるホストにインストールする場合、または PEM サーバーを既存の Postgres サーバーインストールに常駐させる場合は、Advanced options ダイアログを使用します。

インストーラーと PEM-HTTPD でパッケージ化された PostgreSQL サーバーを同じホストにインストールするには、Show advanced options ボックスをオフのままにして Next をクリックします。

PEM サーバーインストーラーは、PEM-HTTPD、Language Pack、および PostgreSQL 10 のインストール前チェックを実行します。インストーラーがこれらのパッケージを見つけられない場合、Dependency missing ダイアログで通知されます。

注釈: デフォルトでは、EDB Language Pack は C:/edb/languagepack/v1 にインストールされます。

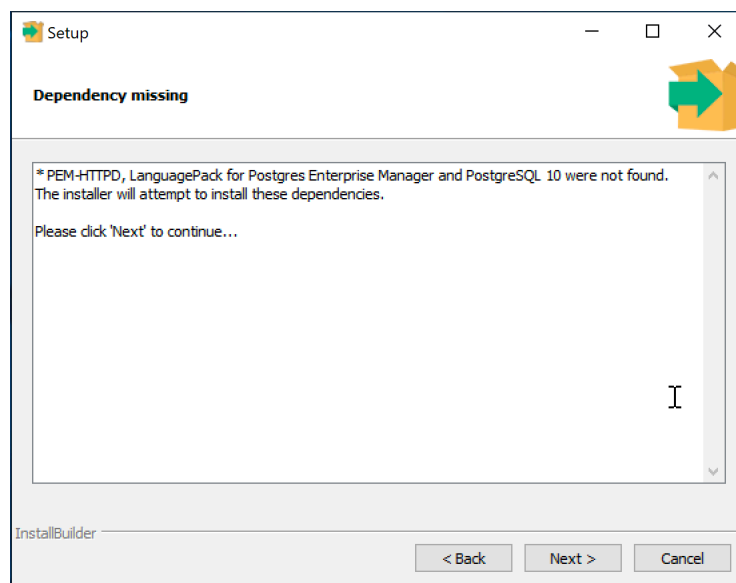


図 5 インストーラーは前提条件を確認します

依存関係が欠落している場合、PEM サーバーインストーラーはそれぞれのインストールウィザードを起動します。各パッケージのインストールウィザードに表示される画面上の指示に従います。

不足している依存関係をインストールした後、Database Server Installation Details ダイアログを表示してインストールプロセスを続行します。

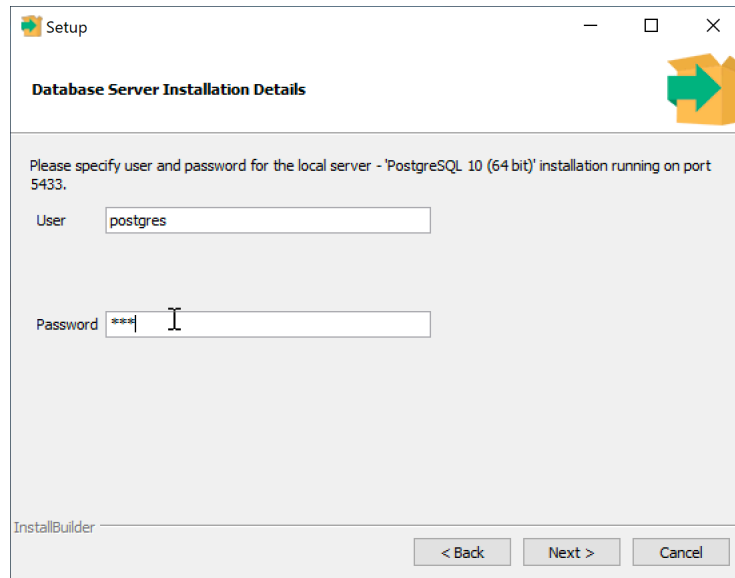


図 6 Enter the name and password of the PostgreSQL superuser

Database Server Installation Details ダイアログで提供される情報により、インストーラーは PostgreSQL サーバーに接続できます。データベースのスーパーユーザーの User name と Password を指定します。要求された情報を提供した後、Next をクリックして pemAgent Service Account ダイアログに進みます。

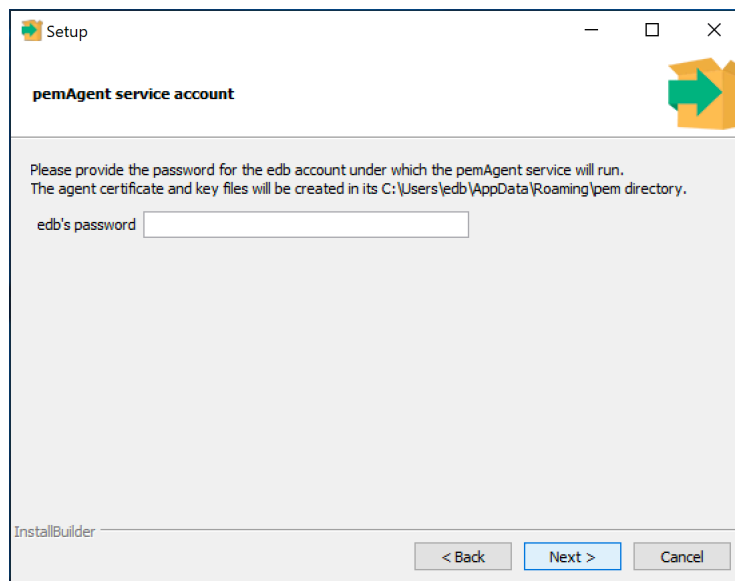


図 7 pemAgent Service Account Password

Postgres データベースのスーパーユーザーの名前とパスワードを入力した後、PEM エージェントを実行するユーザーアカウントのパスワードを求められる場合があります。プロンプトが表示されたら、パスワードを入力し、Next を押して Network Details ダイアログに進みます。

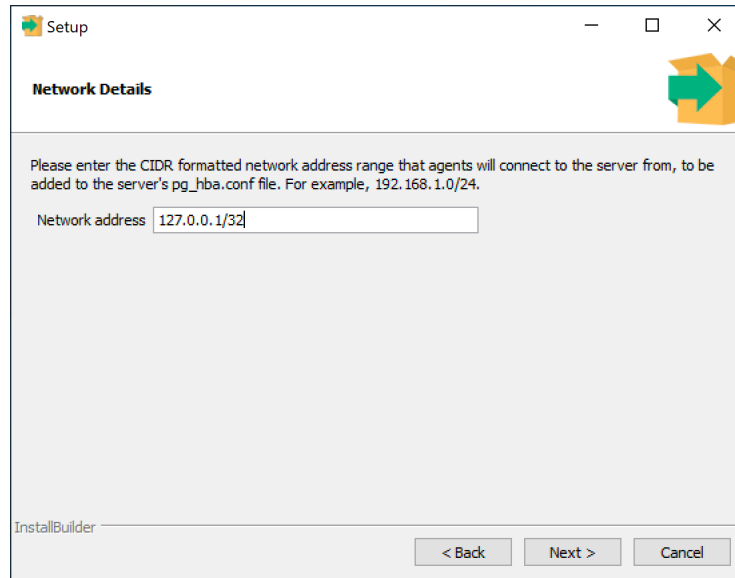


図 8 エージェントが接続するネットワークアドレスを指定します

Network Details ダイアログを使用して、**PEM** エージェントがサーバーに接続する **CIDR** スタイルのネットワークアドレス（*クライアント側*アドレス）を指定します。

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 のデータベースサーバーを監視する場合は、192.168.10.0/24 と入力して、そのネットワーク内のホストとの接続を許可します。

指定されたアドレスはサーバーの `pg_hba.conf` ファイルに追加されます。必要に応じて、初期エントリをテンプレートとして使用して、PostgreSQL サーバー上の `pg_hba.conf` ファイルにエントリを手動で追加することにより、追加のネットワークアドレスを指定できます。

Network address を追加したら、Next をクリックして Agent Details ダイアログに進みます。

PEM サーバーインストーラーは、サーバーが存在するホストに **PEM** エージェントをインストールして、サーバーを監視し、アラート処理およびガベージコレクションサービスを提供します。証明書は、Agent certificate path フィールドで指定された場所にもインストールされます。

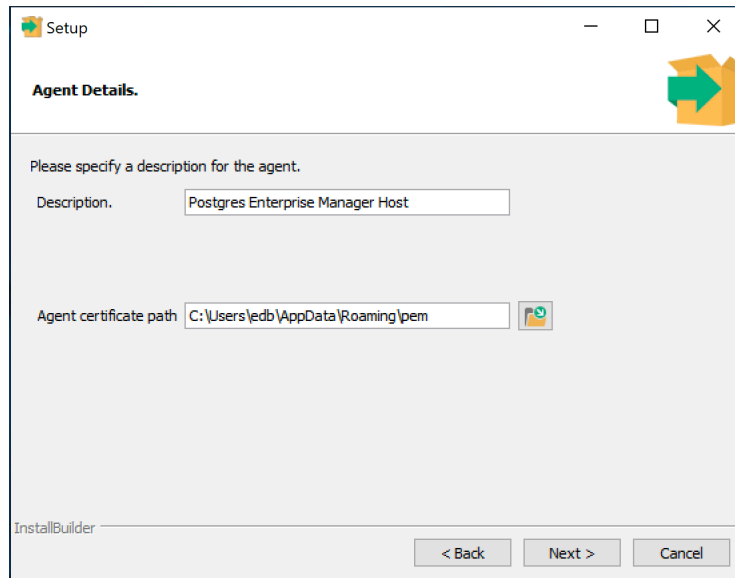


図 9 サーバーに存在するエージェントの説明を提供します

代替の説明を入力するか、**PEM** エージェントの代替エージェント証明書パスを選択するか、デフォルトを受け入れます。Next をクリックして、Ready to Install ダイアログに進みます。

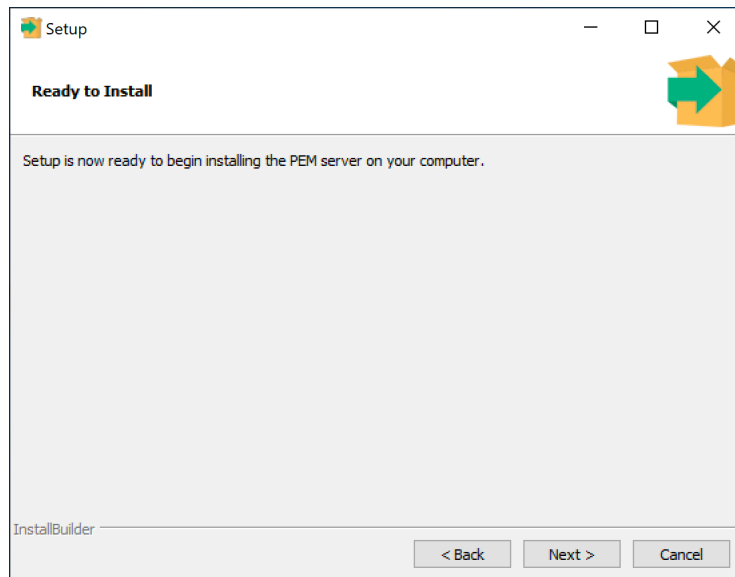


図 10 インストールを開始する準備ができました

これで、ウィザードで **PEM** サーバーをインストールする準備ができました。

以前に選択したオプションを変更するには Back をクリックし、インストールを続行するには Next をクリックします。

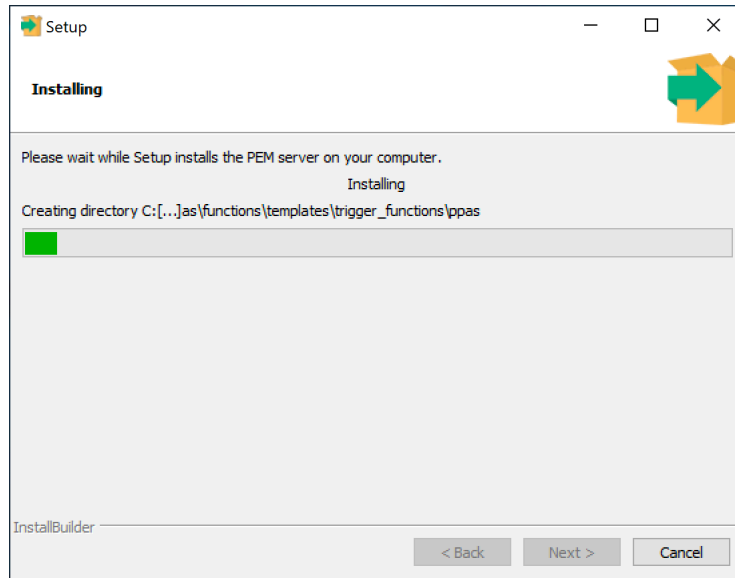


図 11 インストール中

インストールプロセス中、インストーラーはファイルをシステムにコピーし、PEM の実行に必要なデータベースと Web サービスをセットアップします。インストールが完了すると、ポップアップダイアログが開き、ウェブサービスが設定され、ポート 8443 でリッスンしていること、および pem データベースが作成され設定されていることを確認します。

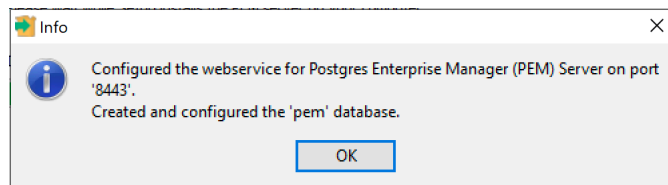


図 12 ポップアップでインストールの詳細を確認します

OK をクリックして、Web サービスが設定されたこと、および pem データベースが作成されたことを確認し、Completed …ダイアログに進みます。

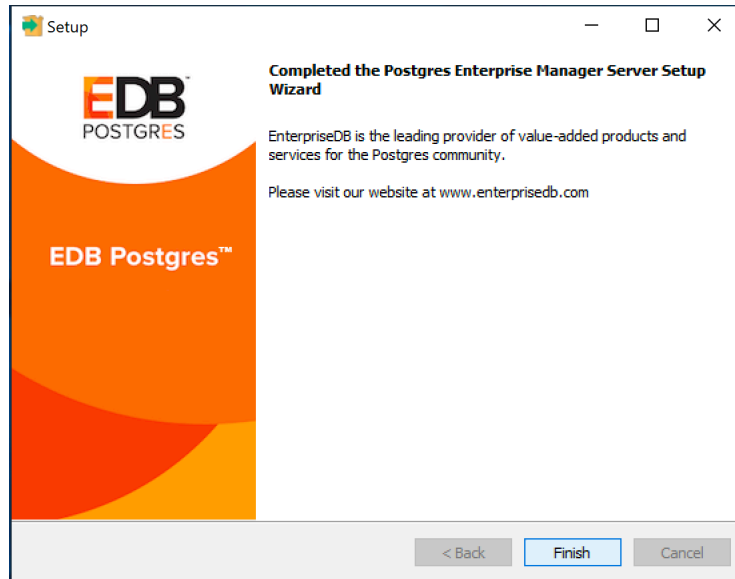


図 13 *The PEM server installation is complete*

3.1.2 PEM サーバーと PEM-HTTPD を別々のホストにインストールする

PEM サーバーバックエンドデータベースと PEM-HTTPD に別々のホストを使用するには、次のことを行う必要があります。

1. pem データベースを含む Postgres サーバーのホストで PEM サーバーインストーラーを起動します。インストール中に、Advanced options ダイアログで Database オプションを選択し、Postgres サーバーの接続情報を指定します。
2. PEM サーバー（および pem データベース）が存在する Postgres インストールの pg_hba.conf ファイルを変更し、PEM-HTTPD サーバーのホストからの接続を許可します。
3. PEM-HTTPD サーバーのホストで PEM サーバーインストーラーを起動し、Installation Type ダイアログで Web Services オプションを選択します。

Windows システムで PEM サーバーインストーラーを起動するには、インストーラーアイコンを右クリックし、Run as Administrator を選択します。インストーラーは Welcome ダイアログを表示します。

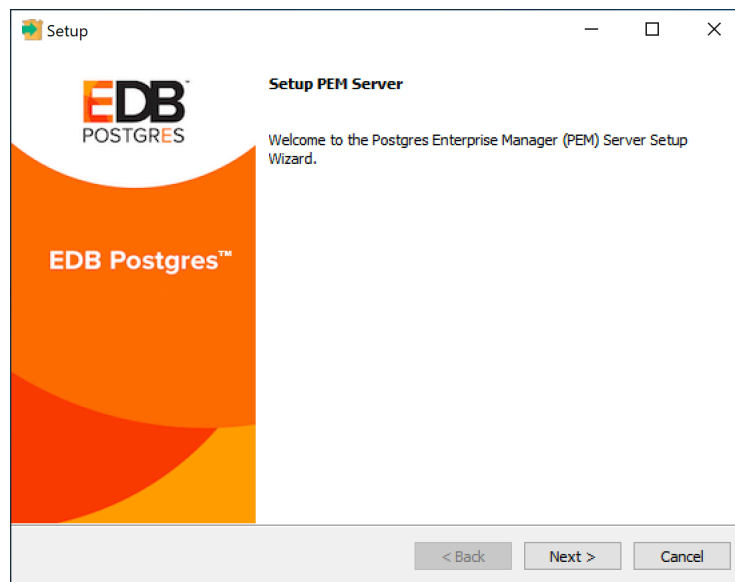


図 14 The PEM Server Installer's Welcome dialog

Next をクリックして、License Agreement ダイアログに進みます。

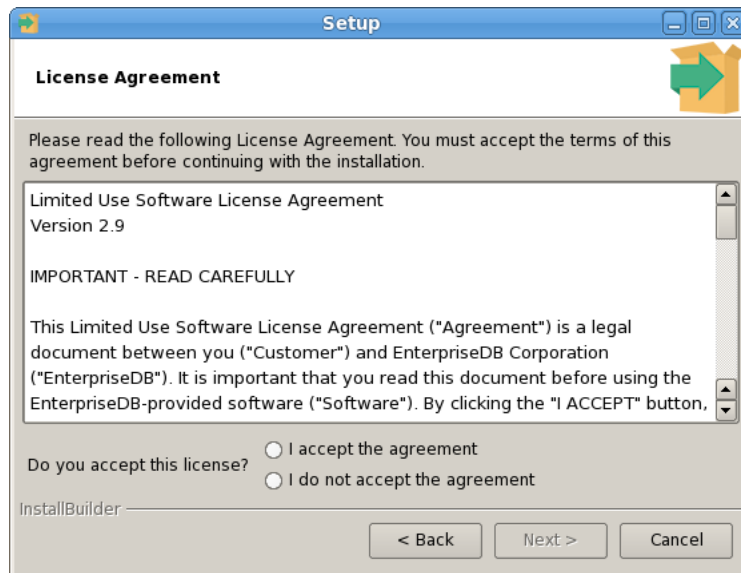


図 15 ライセンス契約

適切なラジオボタンを強調表示して契約に同意する前に、ライセンス契約を注意深く確認してください。Next をクリックして続行します。

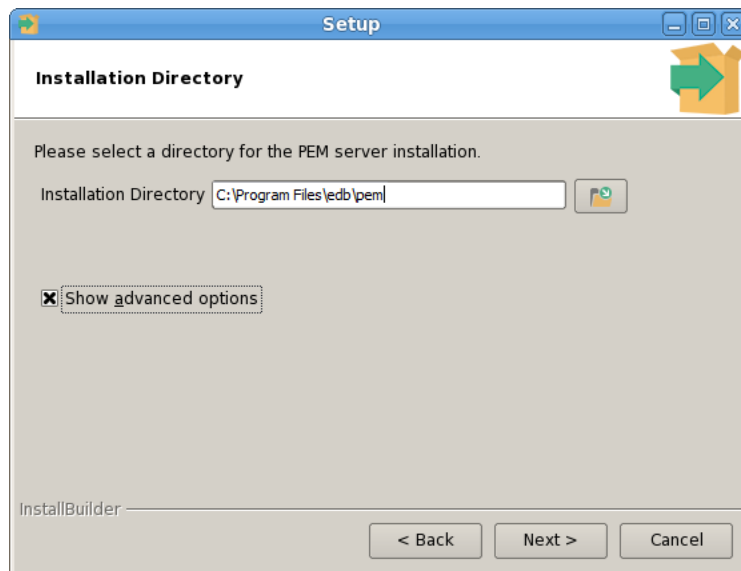


図 16 インストールディレクトリダイアログ

Installation Directory ダイアログのフィールドを使用して、PEM サーバーが存在するディレクトリを指定し、Advanced options ダイアログにアクセスします。

- デフォルトでは、PEM サーバーは Windows の C:/Program Files/edb/pem にインストールされます。デフォルトの場所を受け入れるか、Installation Directory フィールドを使用してブラウザダイアログを開き、PEM サーバーがインストールされるディレクトリを選択します。

- PEM サーバーと PEM-HTTPD を別々のホストにインストールするには、Advanced options ダイアログを使用してインストールタイプ（Web Services または Database）を指定する必要があります。Show advanced options チェックボックスを選択して、インストールプロセスに Advanced options ダイアログを含めるようインストーラーに指示します。

Next をクリックして、Advanced Options ダイアログに進みます。

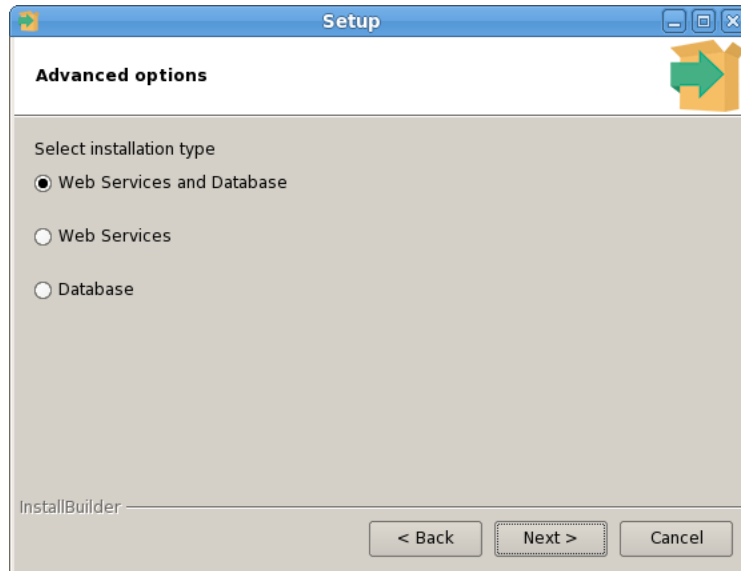


図 17 詳細オプションダイアログ

Advanced options ダイアログのラジオボタンを使用して、インストールするコンポーネントを指定します:

- Web Services and Database を選択して、Postgres サーバーと PEM-HTTPD の両方が現在のホストに存在することを示します。Web Services and Database オプションを選択すると、PEM サーバーインストーラーは、PEM-HTTPD インストールをチェックする前に、PEM サーバーに使用する Postgres サーバーを指定できます。
- Web Services を選択して PEM-HTTPD を現在のホストにインストールし、別のホストにある Postgres データベースサーバーを使用して PEM サーバーと pem データベースをホストします。

注釈: PEM サーバーのホスト（および pem バックエンドデータベース）で PEM サーバーのインストールプロセスを完了し、Advanced options ダイアログで Database を選択し、pem 4“オプションを使用して PEM-HTTPD をインストールする前に、PEM サーバーに 3 ‘'' ファイルを作成します。

このオプションは、[Installing Web Services](#) に記載されているインストール手順を呼び出します。セクション。

- 既存の Postgres サーバー（バージョン 9.4 以降）を使用する場合、または PEM サーバーインストーラーで配布されるデータベースサーバーのみをインストールする場合は、Database を選択します。このオプションは、データベースホストの指定に記載されているインストール手順を呼び出します。。

インストールオプションを選択したら、Next をクリックして続行します。

3.1.2.1 データベースホストの指定

Advanced options ダイアログで Database オプションを選択すると、PEM サーバーバックエンドデータベース（pem という名前）が存在するホストの接続情報を指定できます。

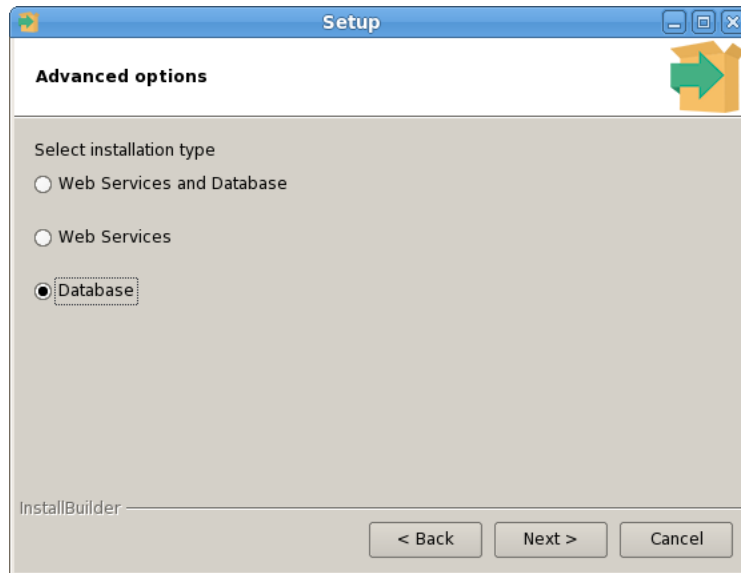


図 18 詳細オプションダイアログ

Next をクリックして、Database Server Selection ダイアログに進みます。

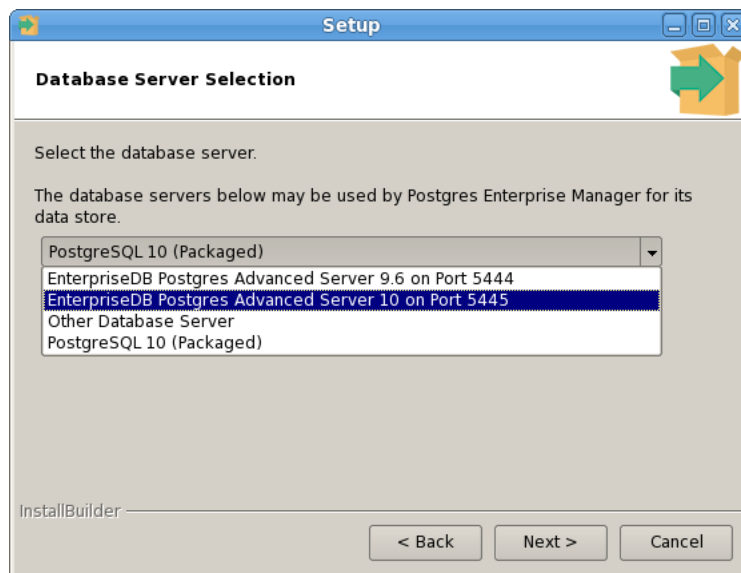


図 19 データベースサーバーの選択

Database Server Selection ダイアログのドロップダウンリストボックスを使用して、PEM サーバーバックエンドデータベースのホストを選択します。次のことができます。

- 現在のホストに存在する既存の Postgres インストールからホストを選択します。

注釈: インストールに `sslutils` パッケージを追加する必要がある場合があります。

- PostgreSQL x オプションを選択して、PEM サーバーインストーラーで配布される Postgres サーバーをインストールします。x は PostgreSQL データベースサーバーのバージョンです。PEM サーバーインストーラーにバンドルされている PostgreSQL のバージョンを使用する場合は、EnterpriseDB のワンクリック PostgreSQL インストーラーが開き、インストールの手順が示されます。
- EnterpriseDB のワンクリックグラフィカルインストーラーを使用してインストールされなかった Postgres サーバーの接続情報を指定するには、Other Database Server を選択します。PEM サーバーデータベースホストのソフトウェア前提条件については、[Postgres サーバーの準備を参照してください](#)。セクション。

Next をクリックして続行します。

PEM サーバーが既存の Postgres サーバーに常駐する場合、Database Server Installation Details ダイアログが表示されます。

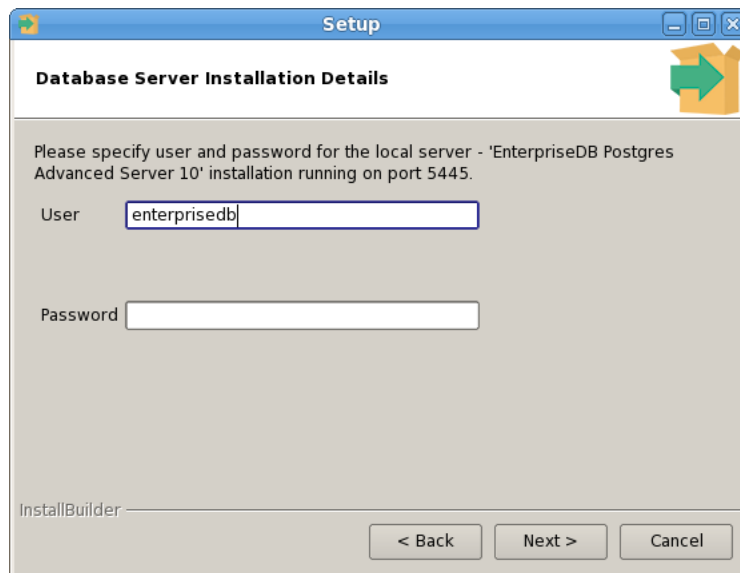


図 20 データベースサーバーのインストールの詳細ダイアログ

Database Server Installation Details ダイアログで必要な情報は異なる場合があります。PEM サーバーインストーラーは、選択できないインストールに関する情報のみを提供するように求めます。

- User フィールドに Postgres データベースのスーパーユーザーの名前を指定します。
- そのユーザーに関連付けられているパスワードを Password フィールドに指定します。

Next をクリックして続行します。

プロンプトが表示されたら、**PEM** エージェントを実行するサービスアカウントのシステムパスワードを入力し、**Next** をクリックして **Network Details** ダイアログに進みます。

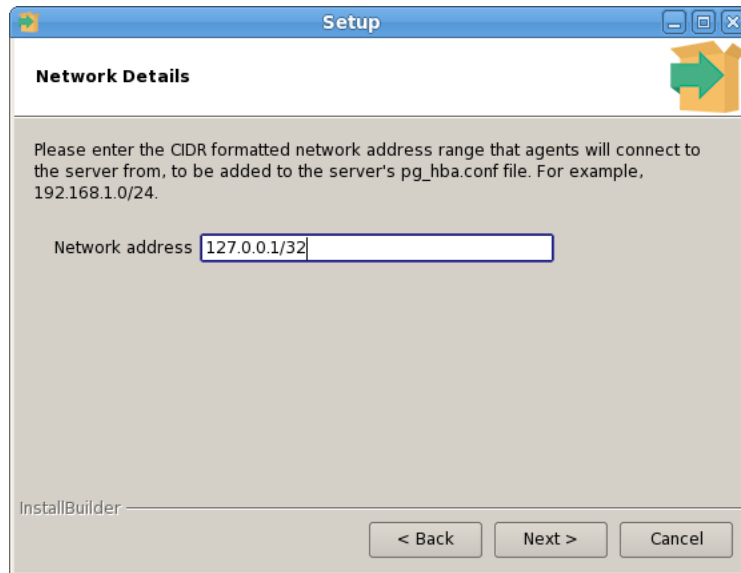


図 21 エージェントが接続するネットワークアドレスを指定します

Network Details ダイアログを使用して、**PEM** エージェントがサーバーに接続する **CIDR** スタイルのネットワークアドレス（*クライアント側*アドレス）を指定します。指定されたアドレスはサーバーの `pg_hba.conf` ファイルに追加されます。

注釈: PostgreSQL サーバー上の `pg_hba.conf` ファイルにエントリを手動で追加することにより、追加のネットワークアドレスを指定できます。

デフォルト（`localhost` を指定）を受け入れるか、**Network address** の範囲を指定し、**Next** をクリックして **Agent Details** ダイアログに進みます。

PEM サーバーインストーラーは、サーバーが存在するホストに **PEM** エージェントをインストールして、サーバーを監視し、アラート処理およびガベージコレクションサービスを提供します。証明書は、**Agent certificate path** フィールドで指定された場所にもインストールされます。

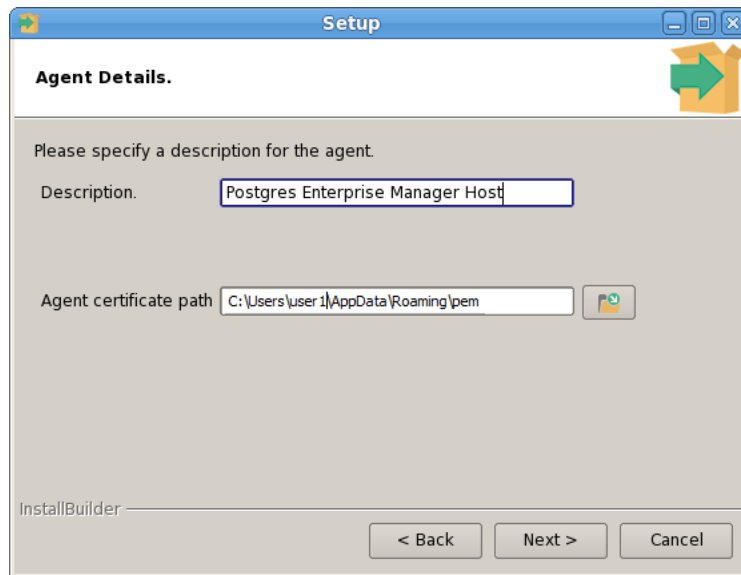


図 22 エージェントの説明を提供

PEM エージェントの代替説明または代替エージェント証明書パスを入力するか、デフォルトを受け入れることができます。Next をクリックして続行します。

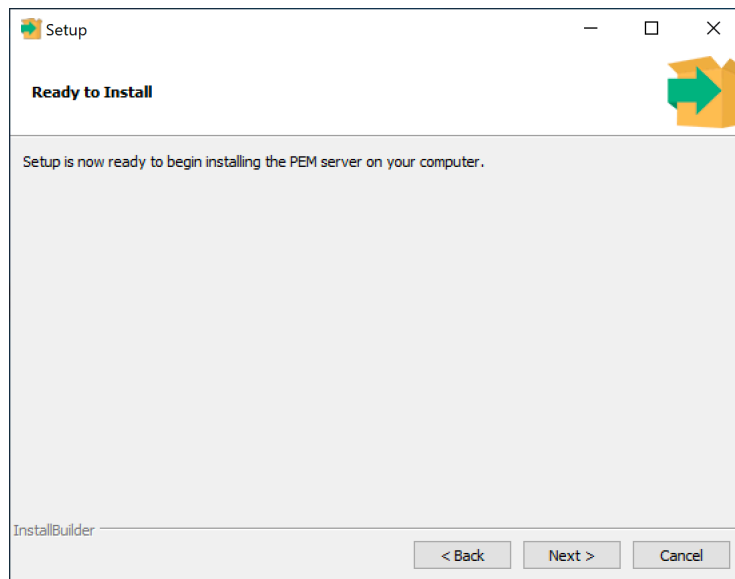


図 23 インストールを開始する準備ができました

これで、ウィザードで PEM サーバーをインストールする準備ができました。Back をクリックして以前に選択したオプションを変更するか、Next をクリックしてインストールを続行します。

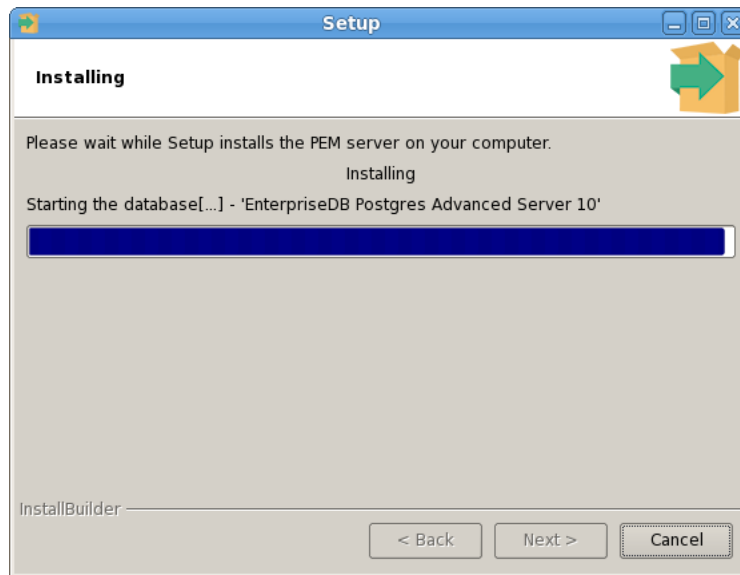


図 24 インストール中

インストールプロセス中に、インストーラーはシステムにファイルをコピーし、PEM サーバーのバックエンドデータベースをセットアップします。pem データベースが作成および設定されたことを確認するポップアップダイアログが開きます。

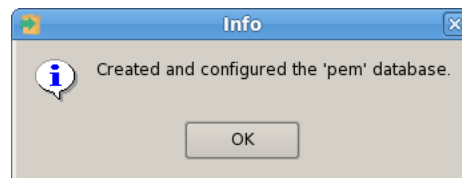


図 25 Click OK to confirm that the pem database has been created

OK をクリックして pem データベースが作成されたことを確認し、Completed… ダイアログに進みます。

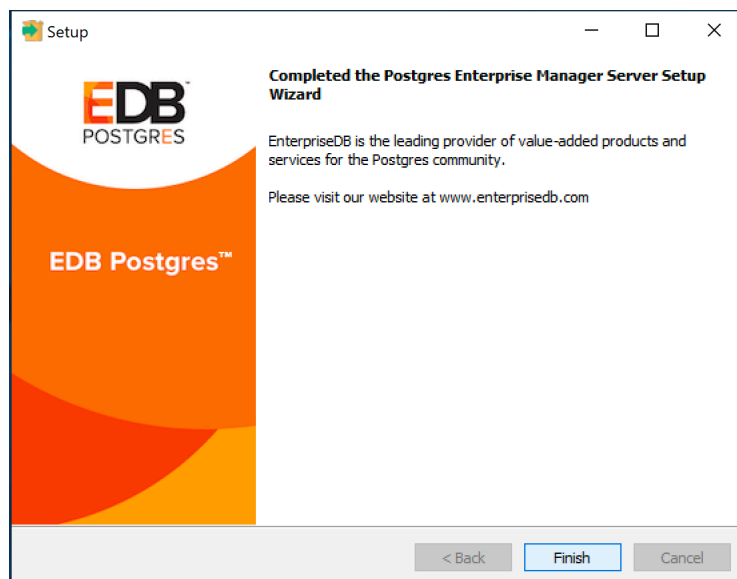


図 26 The PEM server installation is complete

PEM サーバーのインストールのデータベース部分が完了したら、別のホストで PEM サーバーを呼び出して、PEM-HTTPD をインストール（またはアップグレード）できます。

3.1.2.2 Web サービスのインストール

Advanced options ダイアログで Web Services ラジオボタンを選択すると、PEM サーバーのインストーラーに、現在のホストに PEM-HTTPD をインストールするか、既存の PEM-HTTPD インストールを更新するよう指示されます。

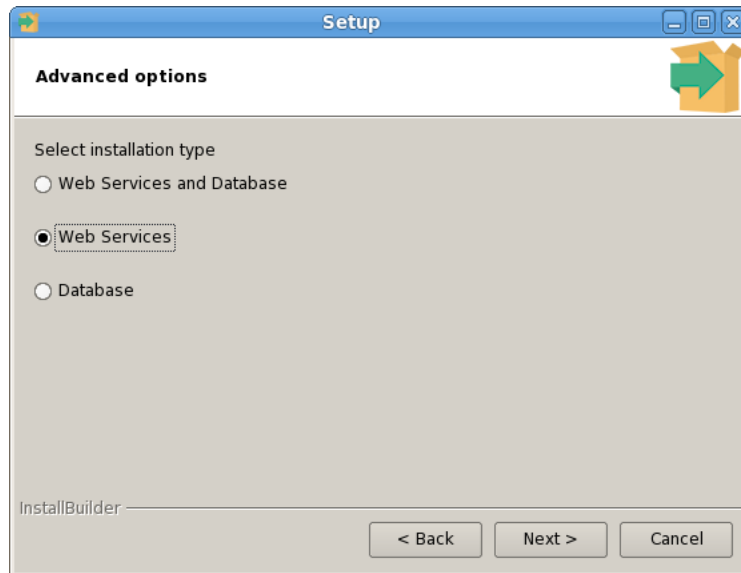


図 27 Selecting the Web Services option on the Advanced options dialog

注釈: 現在のホストは、PEM バックアップデータベースのホストではない可能性があります。

このオプションを選択する前に、次のものがが必要です。

- ホストシステムへの PEM サーバーインストーラーのインストールを完了しました。その間に、PEM サーバーのバックアップデータベースを指定しました。
- PEM サーバーデータベースホスト上の `pg_hba.conf` ファイルを変更して、PEM-HTTPD ホストからの接続を許可し、データベースサーバーを再起動しました。

Web Services オプションを選択して Next をクリックすると、PEM サーバーのインストーラーは、現在のホストで既存の PEM-HTTPD および LanguagePack のインストールを確認します。

インストーラーがコンポーネントを見つけられない場合、インストーラーは 1 つ以上の依存関係が欠落していることを通知します。

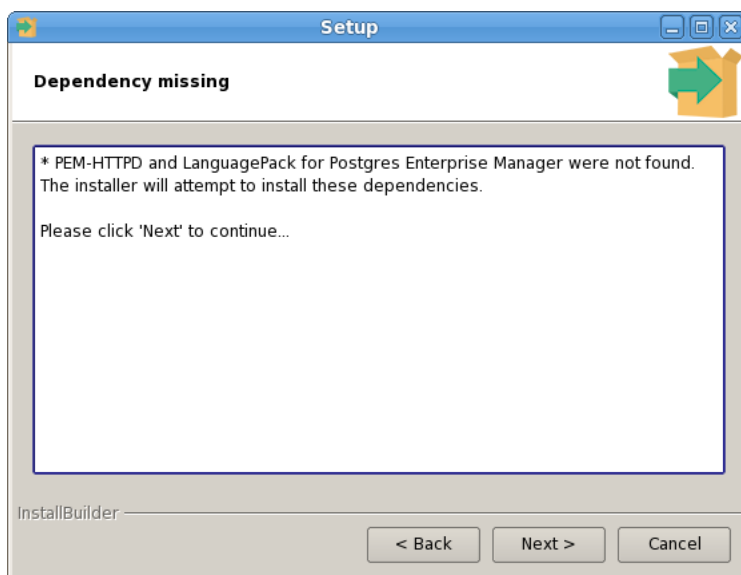


図 28 インストーラーが欠落している依存関係を検出しました

Next をクリックして、LanguagePack をインストールするようサーバーに指示します

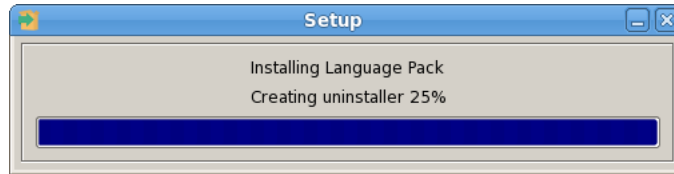


図 29 インストーラーが欠落している依存関係を検出しました

言語パックをインストールした後、インストーラーは PEM-HTTPD セットアップウィザードを呼び出します。

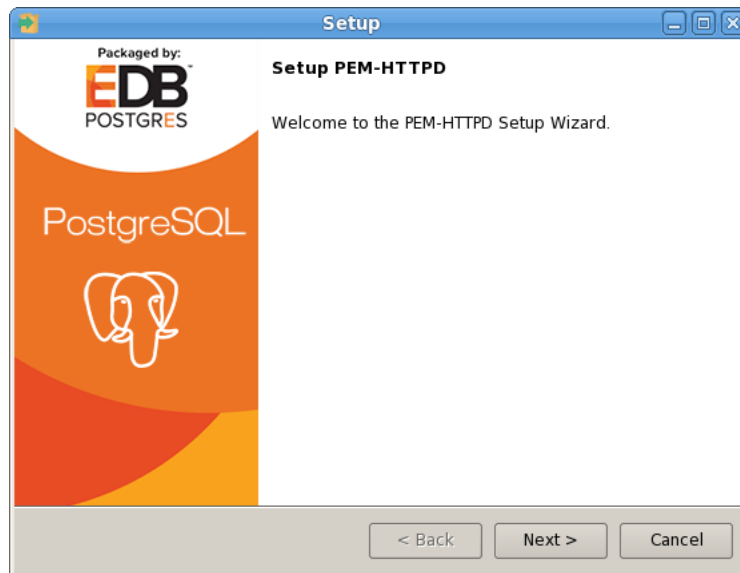


図 30 *The PEM-HTTPD installation wizard*

PEM-HTTPD Setup Wizard の画面上の指示に従います。ウィザードがインストールを完了したら、Finish をクリックして Database Server Installation Details ダイアログを開きます。

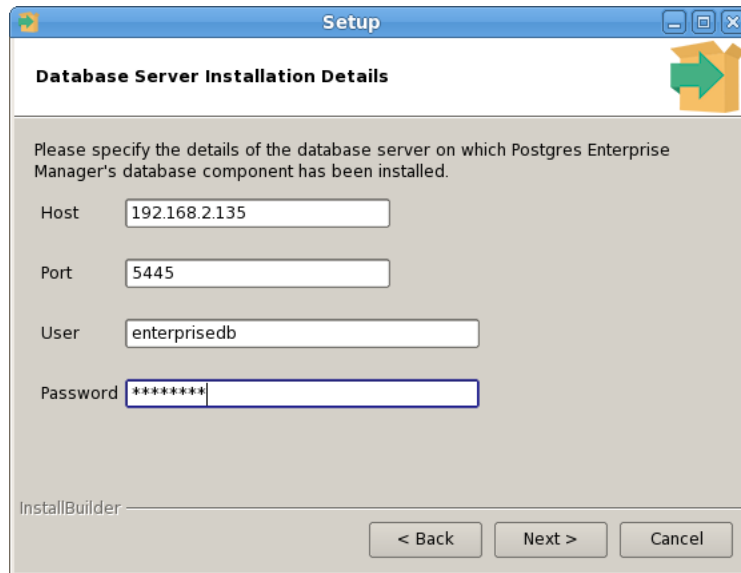


図 31 データベースサーバーのインストールの詳細ダイアログ

Database Server Installation Details ダイアログのフィールドを使用して、**PEM** サーバーのインストールをホストしている **Postgres** インストールの接続情報を提供します。

- **Host** フィールドに **PEM** サーバーホストの名前または **IP** アドレスを入力します。
- **Postgres** サーバーがリッスンしているポート番号を **Port** フィールドに入力します。
- **User** フィールドに **Postgres** データベースのスーパーユーザーの名前を入力します。
- **Password** フィールドに **Postgres** スーパーユーザーに関連付けられたパスワードを入力します。

Next をクリックして続行します。 **PEM** サーバーのインストールを完了する前に、インストーラーはデータベースホストに接続します。 **PEM** データベースホスト上の `pg_hba.conf` ファイルは、**httpd** サーバーのホストからの接続を受け入れるように設定する必要があり、ファイアウォールはインストールを続行するために接続を許可する必要があります。 **PEM** サーバーインストーラーは **PEM** サーバーのインストールを完了し、**PEM-HTTPD** サーバーのホストに存在する必要があるアイテムのみを追加します。

3.1.3 PEM サーバーを既存の Postgres サーバーにインストールする

既存の Postgres サーバー（バージョン 9.4 以降）を使用して、PEM サーバーと pem データベースをホストできます。Postgres インストーラーおよび前提条件のソフトウェア拡張機能は、次の pem で無料で入手できます。
<<http://www.enterprisedb.com/>>‘_

このセクションでは、PEM サーバーのインストール用に既存の Postgres サーバーを構成する方法について説明します。

注釈：以下の手順は、ガイドラインのみと見なされる必要があります。Postgres インストールの構成に必要な実際の手順は、Postgres サーバーの構成によって異なります。

Postgres の以下のバージョンは、sslutils 拡張子とサービススクリプトを含むように事前設定されています。次のバージョンの Postgres を PEM バックエンドデータベースサーバーとして使用するために、追加の準備は必要ありません。

- PostgreSQL 9.4 以降（PEM サーバーインストーラーにバンドル）
- Advanced Server 9.4 以降

3.1.3.1 Postgres サーバーの準備

PEM サーバーを既存の Postgres サーバーにインストールする前に、以下を行う必要があります。

- Postgres サーバーに sslutils 拡張機能のインストールが含まれていることを確認してください。詳細については、sslutils を参照してください。<installing_the_sslutils_extension>‘セクション。
- サービススクリプトを作成する（Linux 上）か、サーバーを Windows サービスマネージャーに登録します。詳細については、“[サービススクリプトの作成またはサービスの登録](#)”を参照してください。セクション。

サーバーを準備したら、PEM サーバーインストーラーを使用して、既存の Postgres サーバーに PEM をインストールできます。

3.1.3.1.1 sslutils 拡張機能のインストール

PEM サーバーが常駐する Postgres サーバーには、sslutils 拡張子が含まれている必要があります。sslutils パッケージは、[EnterpriseDB Web](#) サイトから無料でダウンロードできます。

Web ページが開いたら、SRC- SSL Utils 1.2 パッケージのリンクを選択します。ダウンロードが完了したら、ファイルを抽出し、Postgres インストールディレクトリにコピーします。

Linux で

Postgres サーバーが Linux システムにある場合は、gcc コンパイラーを使用してスラッティルをビルドする必要があります。

1. yum を使用して gcc をインストールします。

```
yum install gcc
```

2. PATH の値を設定して、pg_config プログラムを見つけられるようにします。

```
export PATH=$PATH:/usr/pgsql-x/bin ここで x は PostgreSQL データベースサーバーのバージョンです
```

3. 次に、yum を使用して sslutil の依存関係をインストールします。

```
yum install openssl-devel
```

4. sslutils フォルダーに移動し、次を入力します:

```
make USE_PGXS=1
make USE_PGXS=1 install
```

5. psql を使用して、sslutils 拡張子を作成します。

```
CREATE EXTENSION sslutils
```

Windows で

要確認: 次の Postgres インストールを使用する場合、sslutils 拡張子を手動で追加する必要はありません。

- PostgreSQL 9.4 以降 (PEM サーバーインストーラーと共に配布)
- Advanced Server 9.4 以降

sslutils は、バックエンドの Postgres インストールをコンパイルするのに使用したのと同じコンパイラーでビルドする必要があります。PostgreSQL ワンクリックインストーラー (EnterpriseDB から) または Advanced Server インストーラーを使用して Windows プラットフォームにインストールされたバックエンド Postgres データベースを使用している場合、Visual Studio を使用して sslutils をビルドする必要があります。

インストールプロセスの具体的な詳細はプラットフォームとコンパイラによって異なりますが、基本的な手順は同じです。あなたがする必要があります:

1. sslutils パッケージを Postgres インストールディレクトリにコピーします。
2. 適切なコンパイラのコマンドラインを開き、sslutils ディレクトリに移動します。
3. 次のコマンドを使用して、sslutils を作成します:

```
SET USE_PGXS=1

SET GETTEXT_PATH=<path_to_gettext>
```

```
SET OPENSSL_PATH=<path_to_openssl>

SET PGPATH=<path_to_pg_installation_dir>

SET ARCH=x86

REM Set ARCH x64 for 64 bit

msbuild sslutils.proj /p:Configuration=Release
```

どこで:

path_to_gettext は GETTEXT ライブラリとヘッダーファイルの場所を指定します。

path_to_openssl は openssl ライブラリとヘッダーファイルの場所を指定します。

path_to_pg_installation_dir は Postgres インストールの場所を指定します。

4. コンパイル済みの sslutils ファイルをインストールに適したディレクトリにコピーします。 sslutils ディレクトリには以下のファイルが含まれます:

```
sslutils--1.1.sql

sslutils--unpackaged--1.1.sql

sslutils--pemagent.sql.in

sslutils.dll
```

.dll ライブラリと .sql ファイルを所定の場所にコピーします。

```
COPY sslutils*.sql* /"%PGPATH%/share/extension/"

COPY sslutils.dll /"%PGPATH%/lib/"
```

3.1.3.1.2 サービススクリプトの作成またはサービスの登録

サービススクリプトを使用すると、構成管理、証明書管理、またはその他の管理タスクを実行するときに、必要に応じて PEM サーバーがサーバーを起動、停止、または再起動できます。

EnterpriseDB のインストーラー (PostgreSQL ワンクリックインストーラーなど) を使用して PostgreSQL または Advanced Server データベースをインストールすると、インストーラーがサービススクリプトを作成するか、Windows でサービスを登録します。ソースから Postgres インストールをビルドした場合は、サービススクリプトを手動で作成する必要があります。

PEM サーバーインストーラーは、サービススクリプトの存在をチェックしますが、スクリプト自体の整合性はチェックしません。PEM が正しく機能するためには、サービススクリプトが期待どおりに機能することを確認する必要があります。

Linux サービススクリプトの作成

Linux では、サービススクリプトは `/etc/init.d` ディレクトリに存在する必要があります。サービススクリプトは、データベースサーバーを起動、停止、および再起動できる必要があります。サービススクリプトはプラットフォーム固有です。Linux のサンプルサービススクリプトは、[Linux サービススクリプト \(サンプル\)](#) にあります。セクション。Postgres サービスのカスタマイズについては、次をご覧ください。

‘<<https://www.postgresql.org/docs/current/static/server-start.html>>’_

Windows でのサービスの登録

Windows を使用して PEM バックエンドデータベースをホストしている場合、Postgres サーバーの名前を Windows サービスマネージャーに登録する必要があります。EnterpriseDB インストーラーを使用して作成された Postgres サーバーを使用している場合、サービスは自動的に登録されます。インストールを手動でビルドする場合、Postgres の `pg_ctl` コマンドの `register` 句を使用してサービスを登録できます。コマンドの構文は次のとおりです。

```
pg_ctl register [-N <service_name>] [-U <user_name>] | [-P <password>]
[-D <data_directory>]
```

どこで:

`service name` は Postgres クラスターの名前を指定します。

`user_name` は、Postgres インストールディレクトリにアクセスして Postgres サービスを開始するのに十分な特権を持つオペレーティングシステムユーザーの名前を指定します。

`password` は、ユーザーに関連付けられたオペレーティングシステムのパスワードを指定します。

`data_directory` は Postgres データディレクトリの場所を指定します。

`pg_ctl` コマンドと使用可能なコマンドオプションの使用の詳細については、

‘Postgres のコアドキュメント<<http://www.postgresql.org/docs/current/static/app-pg-ctl.html>>’_

3.1.3.2 PEM サーバーインストーラーの呼び出し

既存の Postgres サーバーを準備した後、PEM サーバーインストーラーを起動します。スーパーユーザー（または Windows では管理）権限を想定し、インストーラーを含むディレクトリに移動します。次に、次のコマンドを使用してインストーラーを起動します。

```
./pem_server-7.<x>.<x>-<x>-<platform>
```

ここで、`*x*` は PEM のメジャーバージョンとマイナーバージョンであり、プラットフォームはプラットフォームです。

インストーラーは Welcome ダイアログを表示します。

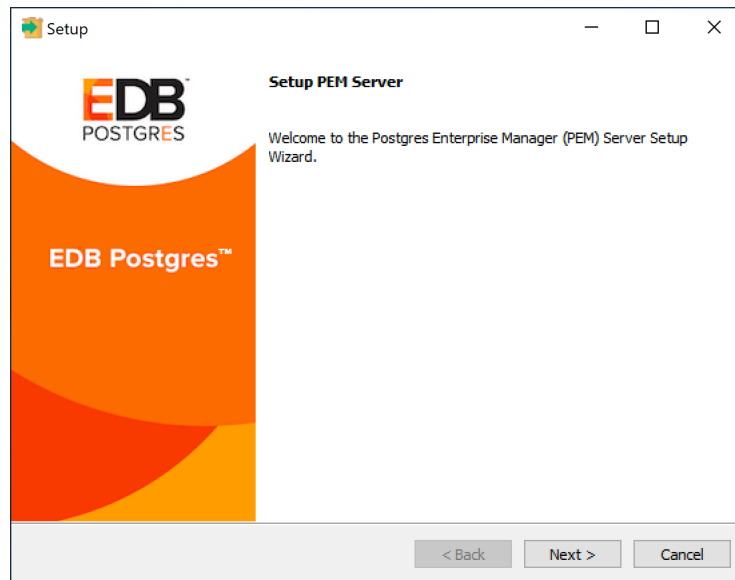


図 32 The PEM server Installer's Welcome Dialog

Next をクリックして、License Agreement ダイアログに進みます。



図 33 ライセンス契約

適切なラジオボタンを強調表示して契約に同意する前に、ライセンス契約を注意深く確認してください。Next をクリックして、Installation Directory ダイアログに進みます。

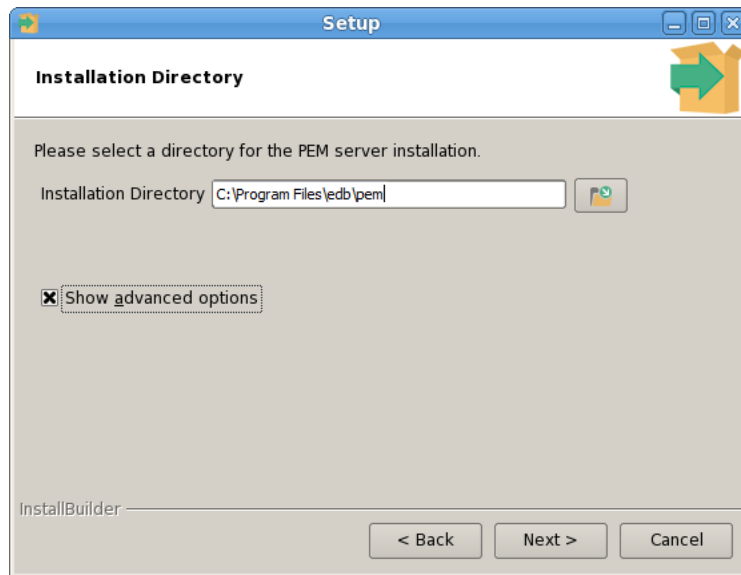


図 34 インストールディレクトリダイアログ

Installation Directory ダイアログを使用して PEM サーバーの場所を指定し、Advanced options ダイアログにアクセスします。

- Installation Directory フィールドを使用してブラウザーダイアログを開き、PEM サーバーをインストールするディレクトリを選択します。
- PEM サーバーを既存のサーバーにインストールする場合は、Show advanced options の隣のボックスをオンにして、インストールプロセスに Advanced options ダイアログを含めるようインストーラーに指示します。

Next をクリックして続行します。

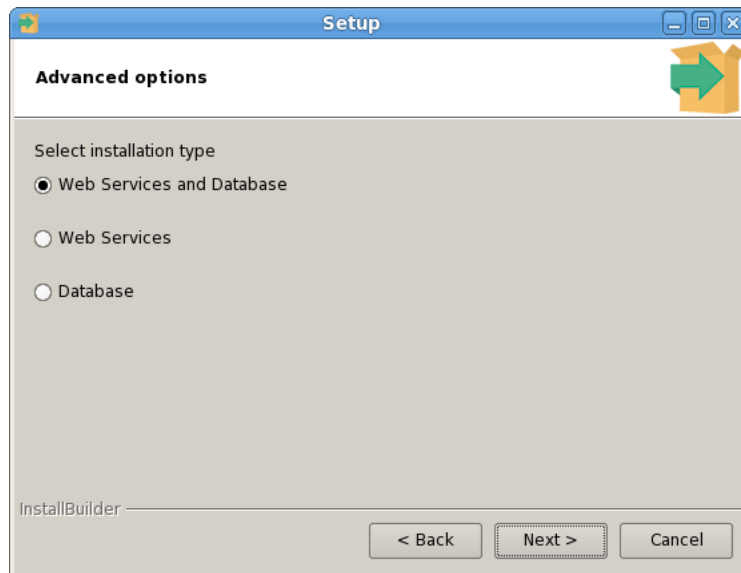


図 35 詳細オプションダイアログ

Advanced options ダイアログのラジオボタンを使用して、インストールタイプを指定します。選択してください:

- Postgres サーバーと PEM-HTTPD サーバーの両方が現在のホストに存在する場合は Web Services and Database。このオプションは、既存の Postgres サーバーを使用して PEM サーバーをホストする場合、または PEM サーバーインストーラーを使用して PEM サーバーが常駐する Postgres サーバーをインストールする場合に有効です。

Web Services and Database を選択すると、PEM サーバーインストーラーは現在のホストで PEM-HTTPD のインストールを確認し、必要に応じて PEM-HTTPD をアップグレードまたはインストールします。

- PEM-HTTPD サーバーのみが現在のホストに存在する場合は Web Services。 [Web サービスのインストールを参照してください](#) このオプションの呼び出しに関する詳細情報のセクション。
- 現在のホストに PEM サーバーのみをインストールする場合（および pem バックエンドデータベースを作成する場合） Database。このオプションは、既存の Postgres サーバーを使用して PEM サーバーをホストする場合、または PEM サーバーインストーラーを使用して PEM が常駐する PostgreSQL サーバーをインストールする場合に有効です。

インストールオプションを選択したら、Next をクリックして続行します。

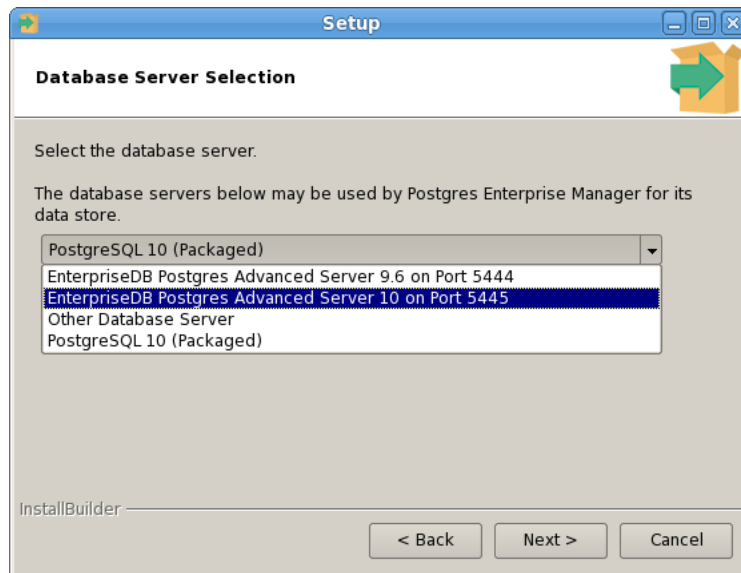


図 36 データベースサーバーの選択ダイアログ

Database Server Selection ダイアログのドロップダウンリストボックスを使用して、PEM サーバーのバックエンドデータベースを選択します。

- Postgres ワンクリックインストーラーまたは Advanced Server インストーラーを使用してインストールされた現在のホスト上の Postgres サーバーの名前を選択します。
- PostgreSQL x (Packaged) オプションを選択して、PEM サーバーインストーラーにパッケージされている PostgreSQL サーバーをインストールして使用するようインストールウィザードに指示します。ここで、x は PostgreSQL データベースサーバーのバージョンです。
- EnterpriseDB インストーラー以外のソースからインストールされた（つまり、rpm から、またはソースからビルドされた）Postgres データベースを使用するように PEM サーバーインストーラーに指示するには、Other Database Server を選択します。

注釈: 選択したデータベースサーバーには `sslutils contrib` モジュールのインストールが含まれており、起動スクリプト（Linux の場合）または登録済みサービス（Windows の場合）が必要です。

Postgres サーバーの準備については、[このセクションを参照してください](#)。

Advanced options ダイアログで Web Services and Database を選択した場合、インストールウィザードは既存の PEM-HTTPD インストールの現在のホストを確認し、必要に応じてサービスをアップグレードまたはインストールします。

Advanced options ダイアログで Database を選択した場合、Database Server Installation Details ダイアログが開きます。

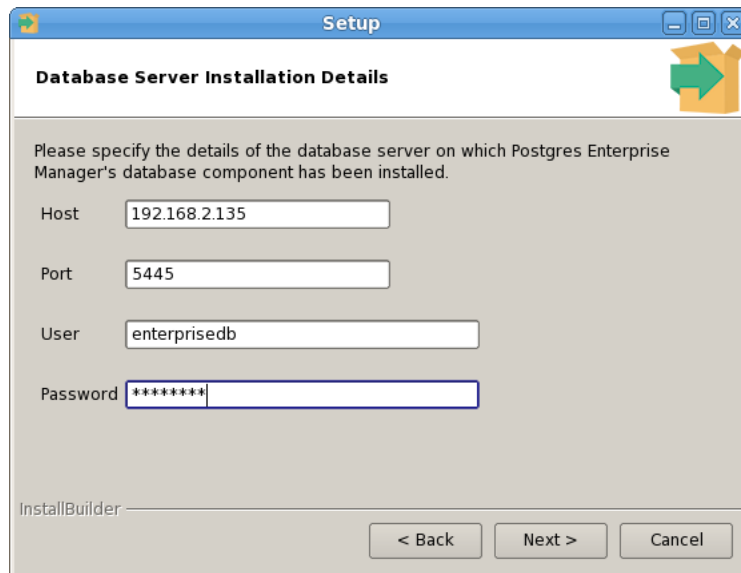


図 37 データベースサーバーのインストールの詳細ダイアログ

Database Server Installation Details ダイアログのフィールドを使用して、PEM サーバーをホストする Postgres サーバーへの接続を記述します。

- Postgres サーバーがリッスンするポート番号を Port フィールドに入力します。
- User フィールドにデータベースのスーパーユーザーの名前を入力します。
- スーパー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。
- Service Name フィールドに Postgres サーバーのサービススクリプトの名前を入力します。

注釈: サービススクリプトの準備については、[このセクションを参照してください](#)。

Next をクリックして続行します。

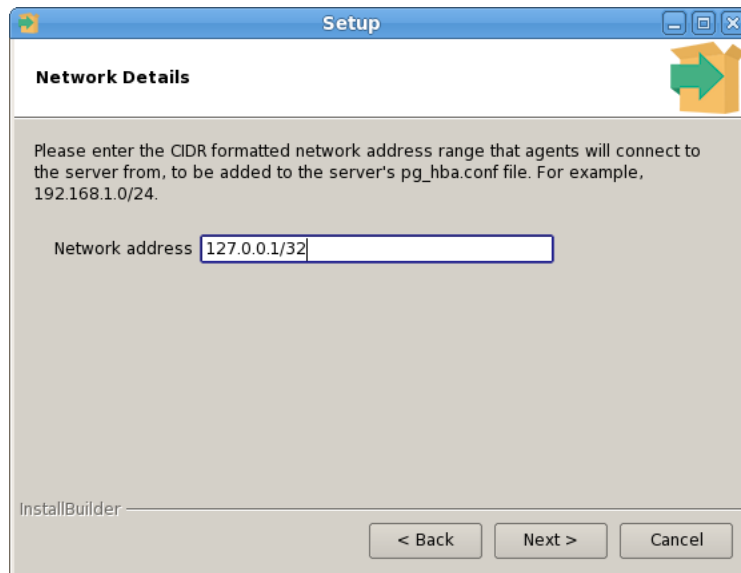


図 38 エージェントが接続するネットワークアドレスを指定します

Network Details ダイアログを使用して、**PEM** エージェントがサーバーに接続する **CIDR** スタイルのネットワークアドレス（client-side アドレス）を指定します。指定されたアドレスがサーバーの `pg_hba.conf` file. に追加されます

必要に応じて、初期エントリをテンプレートとして使用して、**PostgreSQL** サーバー上の `pg_hba.conf` ファイルにエントリを手動で追加することにより、追加のネットワークアドレスを指定できます。

Network address を追加したら、Next をクリックして Agent Details ダイアログに進みます。

PEM サーバーインストーラーは、サーバーが存在するホストに **PEM** エージェントをインストールして、サーバーを監視し、アラート処理およびガベージコレクションサービスを提供します。証明書は、Agent certificate パスフィールドで指定された場所にもインストールされます。

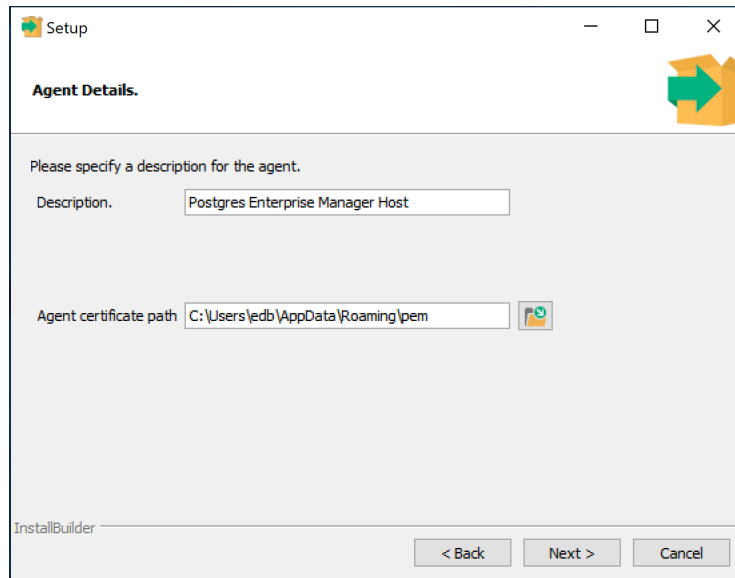


図 39 存在するエージェントの説明を提供します

PEM エージェントの代替説明または代替エージェント証明書パスを入力するか、デフォルトを受け入れることができます。Next をクリックして、Ready to Install ダイアログに進みます。

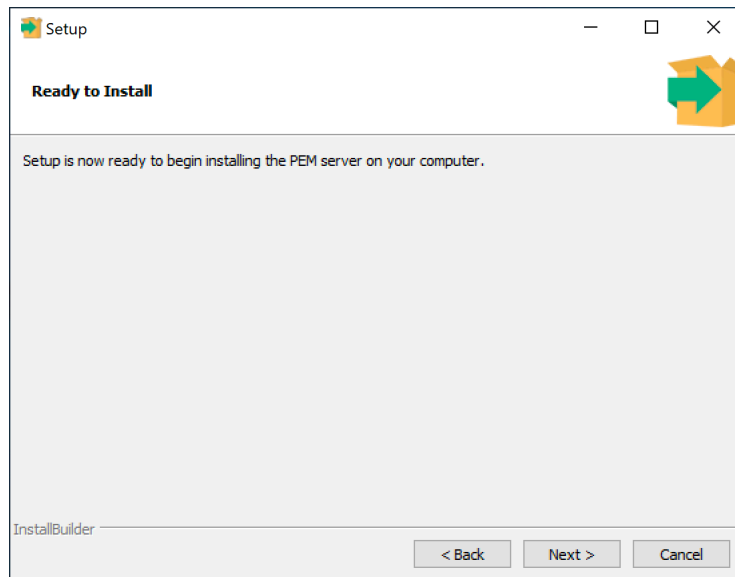


図 40 インストールを開始する準備ができました

これで、ウィザードで PEM サーバーをインストールする準備ができました。以前に選択したオプションを変更するには Back をクリックし、インストールを続行するには Next をクリックします。

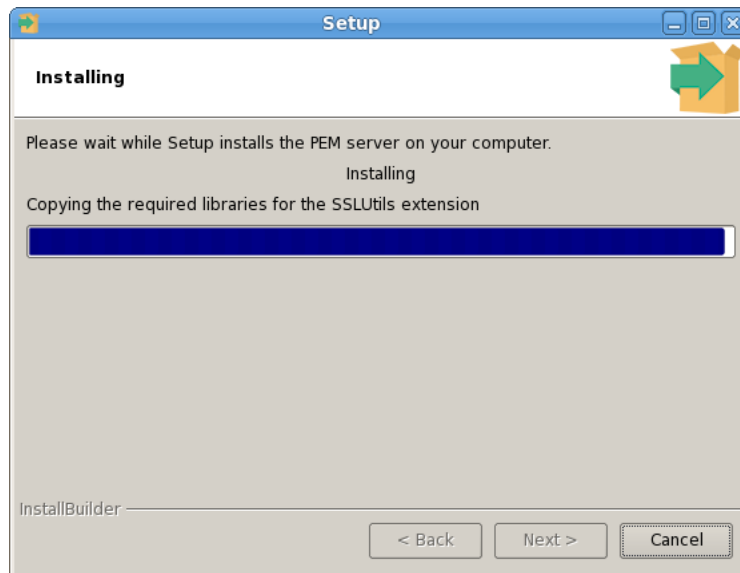


図 41 インストール中

インストールプロセス中に、インストーラーはシステムにファイルをコピーし、PEM サーバーのバックエンドデータベースをセットアップします。pem データベースが作成および設定されたことを確認するポップアップダイアログが開きます。

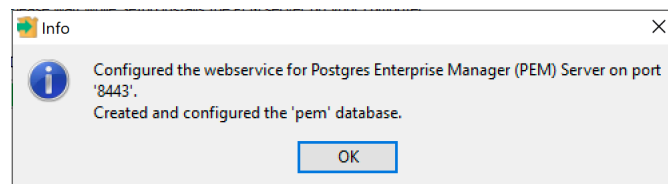


図 42 Click OK to confirm that the pem database has been created

OK をクリックして、pem データベースが作成されたことを確認し、Completed... ダイアログに進みます。

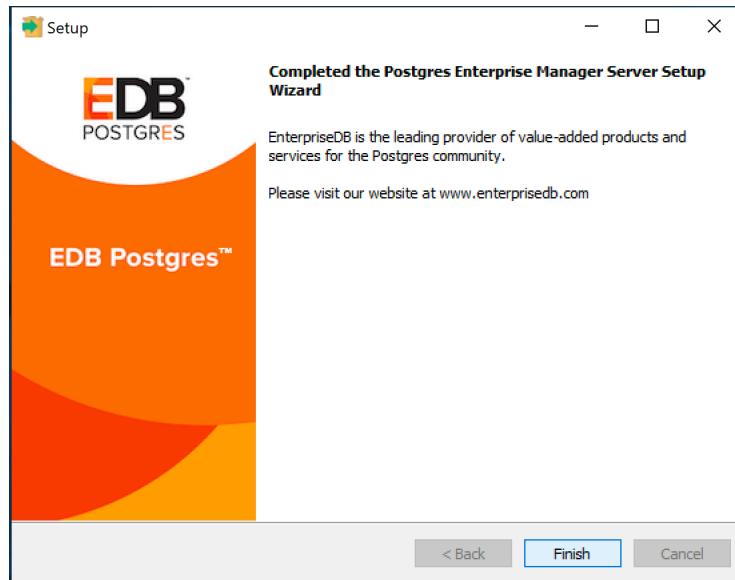


図 43 The PEM server installation is complete

別のホストにある PEM-HTTPD サービスを使用している場合は、次を行う必要があります。

- Postgres サーバーホストの `pg_hba.conf` ファイルを変更して、ホスト間の接続を許可します。
- PEM-HTTPD サーバーのホストで PEM サーバーインストーラーを起動します。 [Web サービスのインストールを参照してください](#) PEM-HTTPD のインストールに関する詳細情報のセクション。

3.1.4 コマンドラインからサーバーインストーラーを呼び出す

PEM サーバーおよび PEM エージェントインストーラーのコマンドラインオプションは、リソースまたはシステム構成が制限されているためにグラフィカルインストールが機能しない場合に機能を提供します。次のことができます。

- インストーラーを呼び出して追加のユーザー入力なしでインストールを実行する場合は、`--mode unattended` オプションを含めます。

すべてのコマンドラインオプションがすべてのプラットフォームに適しているわけではありません。コマンドラインオプションの完全なリファレンスガイドについては、インストーラーを起動するときに `--help` オプションを含めてください。

3.1.4.1 無人モードでの PEM サーバーインストーラーの呼び出し

インストーラーを呼び出すときにコマンドラインでインストール設定を指定することにより、PEM サーバーの無人インストールを実行できます。PEM サーバーをインストールするシステムは、インターネットにアクセスできる必要があることに注意してください。

PEM サーバーをインストールするには管理者権限が必要です。PEM サーバーインストーラーを呼び出す前に、次の依存関係をインストールする必要があります。

- PostgreSQL
- pem-httpd
- 言語パック

PEM サーバーのインストーラーを使用して、PEM サーバーの依存関係を満たすことができます。次のコマンドを使用して、依存関係を抽出します。インストーラーの場所に移動し、次のコマンドを使用して依存関係を抽出します。

```
pem-server-7.<x>.<x>-windows-x64.exe --extract-dependents C:/
```

この例では、ファイルは C:/ ディレクトリに抽出されます。ファイルを抽出した後、各プログラムをインストールする必要があります。ファイルが含まれるディレクトリ（この例では C:/）に移動し、次のように入力します：

```
edb-languagepack-<version>-windows-x64.exe --mode unattended  
pem-httpd-<version>-windows-x64.exe --mode unattended  
postgresql- <version> -windows-x64.exe --mode unattended
```

次に、PEM サーバーインストーラーを起動できます。

```
pem-server-7.<x>.<x>-windows-x64.exe --mode unattended  
--existing-user <registered_edb_user> --existing-password  
<edb_user_password> --pgport <port> --pguser postgres  
--pgpassword postgres --cidr-address <cidr_address_range>  
--agent_description pem-agent --systempassword <windows_password>  
--agent-crt-path C:/edb
```

どこで:

registered_edb_user は、登録済みの EnterpriseDB ユーザーの名前を指定します。登録するには、**‘EDB ウェブサイトにアクセスしてください<<https://www.enterprisedb.com/user/register>>’** _ _

edb_user_password は、EDB ユーザーアカウントに関連付けられたパスワードを指定します。

port は、バックアップ PostgreSQL データベースが使用するポートを指定します。デフォルトでは、PostgreSQL データベースはポート 5432 を使用します。

`cidr_address_range` は、PEM サーバーのバックアップデータベースの `pg_hba.conf` ファイルに追加されるアドレス範囲を指定して、サーバーによって監視されるエージェントからの接続を許可します。ネットワーク範囲（たとえば、192.168.2.0 / 24）を指定して、同じネットワーク上にあるエージェントにサーバーアクセスを提供することができます。

`windows_password` は、Windows 管理者のアカウントに関連付けられたパスワードを指定します。

注釈: 無人モードで起動すると、PostgreSQL インストーラーは、パスワードが `postgres` の `postgres` という名前のユーザーを作成します。

EnterpriseDB は、Postgres コミュニティ向けの付加価値製品およびサービスのリーディングプロバイダーです。当社のウェブサイト www.enterprisedb.com をご覧ください

3.2 Linux での PEM サーバーのインストール

RHEL、CentOS、SLES、Debian、または Ubuntu ホストに PEM サーバーをインストールする場合、以下を確認する必要があります。

1. RPM パッケージを使用して PEM サーバーをインストールする場合、最初にバックアップデータベースを手動でインストールし、データベースクラスターを作成する必要があります。サーバーのバックアップデータベースは、RPM パッケージまたは BitRock を介してインストールする必要があります。データベースは、次のバージョンのいずれかである必要があります。

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

Advanced Server または PostgreSQL データベースのインストールの詳細については、製品のドキュメントを参照してください。

2. 接続に信頼認証を使用するには、バックアップデータベースの `pg_hba.conf` ファイルを構成する必要があります。 `pg_hba.conf` ファイルの変更については、次を参照してください。

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

3. PostgreSQL データベースを使用している場合は、`hstore contrib` モジュールもインストールする必要があります。詳細については、次をご覧ください。

<https://www.postgresql.org/docs/current/static/hstore.html>

4. ファイアウォールを使用している場合、ポート 8443 へのアクセスを許可する必要があります。次のコマンドを使用します。

```
firewall-cmd --permanent --zone=public --add-port=8443/tcp  
  
firewall-cmd --reload
```

3.2.1 RHEL または CentOS HOST の追加の前提条件

上記の前提条件に加えて、RHEL または CentOS ホストを使用している場合、次の前提条件が適用されます。

1. 次のいずれかのコマンドを実行して、ホストに `epel-release` パッケージをインストールする必要があります。

- `yum -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm`
- `yum install epel-release`

注釈: CentOS-Base.repo ファイル (/etc/yum.repos.d にある) で [extras] リポジトリ定義を有効にする必要があるかもしれません。

Red Hat Network ユーザーの場合、EPEL パッケージを使用するには `rhel-<x>-server-optional-rpms` リポジトリも有効にする必要があります。* **x** *はホスト上の RHEL のバージョンを指定します。RHN-Classic の RHEL optional subchannel を有効にすることで、リポジトリにアクセスできるようになります。証明書ベースのサブスクリプションをお持ちの場合は、Red Hat Subscription Management Guide をご覧ください。

2. また、`rhel-<x>-server-extras-rpms` リポジトリを有効にする必要があります。x は、ホスト上の RHEL のバージョンを指定します。
3. RHEL 6.x または CentOS 6.x ホストを使用している場合、PEM サーバーをインストールする前にデフォルトの `httpd` サーバーを停止し、コマンドを実行して `rh-python36` をインストールする必要があります。PEM サーバーのインストールの一部として、RHEL または CentOS 6.x ホストを使用している場合、`httpd24` サーバーがインストールされます。

1. 次のコマンドを使用して、デフォルトの `httpd` サーバーを停止します。

```
service httpd stop
```

2. 次のコマンドを実行して、`rh-python36` をインストールします:

```
yum install centos-release-scl
```

```
yum install rh-python36
```

4. EnterpriseDB リポジトリへのアクセスを許可する資格情報も必要です。資格情報のリクエストについては、次をご覧ください。

<https://info.enterprisedb.com/rs/069-ALB-339/images/Repository%20Access%2004-09-2019.pdf>

リポジトリの資格情報を受け取った後、次のことができます。

1. リポジトリ構成ファイルを作成します。
2. ユーザー名とパスワードを指定して、ファイルを変更します。
3. RHEL または CentOS ホストに `edb-pem` をインストールします。

リポジトリ設定ファイルの作成

リポジトリ構成ファイルを作成するには、スーパーユーザー特権を引き受けて、次のコマンドを呼び出します。

```
yum -y install https://yum.enterprisedb.com/edb-repo-rpms/
edb-repo-latest.noarch.rpm
```

リポジトリ設定ファイルの名前は `edb.repo` です。ファイルは `/etc/yum.repos.d` にあります。

ファイルの変更、ユーザー名とパスワードの入力

`edb.repo` ファイルを作成した後、エディターの選択を使用して、有効なパラメーターの値が 1 であることを確認し、`edb.repo` の `username` および `password` プレースホルダーを置き換えます 4 '' 登録済みの EnterpriseDB ユーザーの名前とパスワードを含む仕様。

```
[edb]
name=EnterpriseDB RPMs $releasever - $basearch
baseurl=https://<username>:<password>@yum.enterprisedb.com/edb/redhat/rhel-$releasever-
↪$basearch
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/ENTERPRISEDB-GPG-KEY
```

3.2.1.1 CentOS または RHEL ホストへの PEM サーバーのインストール

`yum` を使用して PEM サーバーをインストールできます。

```
yum install edb-pem
```

システムで認識されないソースによって署名された RPM パッケージをインストールすると、`yum` はローカルサーバーにキーをインポートする許可を求める場合があります。プロンプトが表示され、パッケージが信頼できるソースからのものであることに満足したら、`y` を入力し、Return を押して続行します。

インストール中に、`yum` は解決できない依存関係に遭遇する場合があります。存在する場合、手動で解決する必要がある依存関係のリストが提供されます。

分離されたネットワークにあるマシンに PEM サーバーをインストールする場合、最初にそのマシンに PEM リポジトリを作成する必要があります。分離されたネットワークでの PEM リポジトリの作成の詳細については、“分離されたネットワークでの PEM リポジトリの作成”を参照してください。

3.2.1.2 Debian または Ubuntu ホストへの PEM サーバーのインストール

Debian または Ubuntu ホストに PEM をインストールするには、EnterpriseDB リポジトリへのアクセスを許可する資格情報が必要です。リポジトリの資格情報を要求するには、“EnterpriseDB <<https://www.enterprisedb.com/repository-access-request>>”。

次の手順では、EnterpriseDB apt リポジトリを使用して Debian パッケージをインストールする手順を説明します。コマンドを使用する場合、ユーザー名とパスワードを EnterpriseDB が提供する資格情報に置き換えます。

1. <https://apt.enterprisedb.com/> にアクセスし、ルートとしてログインします。

```
sudo su -
```

2. EnterpriseDB リポジトリを構成します。

```
sh -c 'echo /"deb https://username:password@apt.enterprisedb.com
$(lsb_release -cs)-edb/ $(lsb_release -cs) main/" > /etc/apt/sources.
list.d/edb-$(lsb_release -cs).list'
```

3. 安全な APT リポジトリのサポートをシステムに追加します。

```
apt-get install apt-transport-https
```

4. EDB 署名キーを追加します。

```
wget -q -O -https://username:password@apt.enterprisedb.com/edb-deb.gpg.
key | apt-key add -
```

5. リポジトリのメタデータを更新します。

```
apt-get update
```

6. 次のコマンドを使用して、PEM サーバーの Debian パッケージをインストールします。

```
apt-get install edb-pem
```

3.2.1.3 SLES ホストへの PEM サーバーのインストール

Advanced Server のインストールと SLES ホストへのサポートコンポーネントの詳細については、次の場所にある* 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/edb-docs/p/edb-postgres-enterprise-manager>

SLES パッケージは以下から入手できます。 <https://zypp.enterprisedb.com>

PEM をインストールする前に、前提条件パッケージをインストールする必要があります。次のコマンドを呼び出して、* sp_no *を使用しているサービスパック（SP4 など）に置き換えます。

```
SUSEConnect -p sle-module-legacy/12/x86_64

SUSEConnect -p sle-sdk/12/x86_64

zypper addrepo https://download.opensuse.org/repositories/Apache:Modules/
SLE_12_<sp_no>/Apache:Modules.repo

zypper addrepo http://download.opensuse.org/repositories/Cloud:/
OpenStack:/Newton:/cisco-apic:/2.3.1/SLE_12_<sp_no>/ pem_opensuse_boost

zypper refresh

zypper install edb-pem
```

3.2.1.4 PEM サーバーの構成

PEM サーバーを設定する前に、バックアップデータベースに `sslutils` 拡張機能がインストールされていることを確認してください。

- **Advanced Server** バックアップデータベースの場合、デフォルトでは `sslutils` 拡張子が **Advanced Server** とともにインストールされます。
- **PostgreSQL** バックアップデータベースを使用している場合は、**PostgreSQL** コミュニティリポジトリにアクセスできることを確認し、次のコマンドを使用します。

```
yum install sslutils_<x> postgresql<X>-contrib
```

ここで、`x` はサーバーのバージョンです。

PEM サーバーインストーラーには、**RPM** インストールの構成プロセスを自動化するためのスクリプト（`configure-pem-server.sh`）が含まれています。スクリプトは `/usr/edb/pem/bin` ディレクトリにインストールされます。スクリプトを呼び出すには、次のコマンドを使用します。

```
/usr/edb/pem/bin/configure-pem-server.sh
```

スクリプトを呼び出すときに、構成プロパティを指定するコマンドラインオプションを含めることができます。スクリプトは、コマンドラインで省略した値の入力を求めます。受け入れられるオプションは次のとおりです。

オプション	説明
<code>-acp</code>	PEM エージェントの証明書パスを定義します。デフォルトは <code>/root/.pem</code> です。
<code>-ci</code>	サーバーの <code>pg_hba.conf</code> ファイルに追加される、エージェントがサーバーに接続する CIDR 形式のネットワークアドレス範囲。たとえば、 <code>192.168.1.0/24</code> 。デフォルトは <code>0.0.0.0/0</code> です。
<code>-dbi</code>	データベースサーバーのインストールディレクトリ。たとえば、 Advanced Server の場合は <code>/usr/edb/as10</code> 、 PostgreSQL の場合は <code>/usr/pgsql-10</code> です。
<code>-ds</code>	PEM データベースサーバーのユニットファイル名。 Advanced Server の場合、デフォルトのファイル名は <code>edb-as-10</code> です。 PostgreSQL の場合、 <code>postgresql-10</code> です。
<code>-ho</code>	PEM データベースサーバーのホストアドレス。
<code>-p</code>	PEM データベースサーバーのポート番号。
<code>-ps</code>	<code>pemagent</code> のサービス名。デフォルト値は <code>pemagent</code> です。
<code>-sp</code>	PEM データベースサーバーのスーパーユーザーパスワード。この値は必須です。
<code>-su</code>	PEM データベースサーバーのスーパーユーザー名。
<code>-t</code>	インストールの種類:構成が Web サービスとバックアップデータベース用の場合は 1、 Web サービスを構成する場合は 2、バックアップデータベースを構成する場合は 3 を指定します。3 を指定する場合、データベースはローカルホストに存在する必要があることに注意してください。

コマンドラインで構成プロパティを指定しない場合、スクリプトによって値の入力が求められます。スクリプトを呼び出すとき、次から選択します。

1. **Web Services and Database-** Select this option if the web server and database both reside on the same host as the PEM server.
2. **Web Services-** Select this option if the web server resides on a different host than the PEM server.
3. **Database-** Select this option to configure the PEM backing database for use by the PEM server. Please note that the specified database must reside on the local host.

注釈: **Web** サーバーとバックアップデータベースが別々のホストにある場合は、最初にデータベースサーバーを構成し（オプション 3）、次に **Web** サービスを構成します（オプション 2）。**Web** サービスの前にバックアップデータベースが構成されていない場合、スクリプトは終了します。

構成オプションを選択すると、スクリプトは構成プロパティの入力を求めるプロンプトに進みます。スクリプトが完了すると、**PEM** サーバーが必要とするオブジェクトを作成するか、必要な構成手順を実行します。

スクリプト関連のヘルプを表示するには、次のコマンドを使用します。

```
/usr/edb/pem/bin/configure-pem-server.sh -help
```

RHEL または CentOS 6.x ホストを使用している場合は、**PEM** サーバーを構成した後に httpd24 サーバーを再起動します。次のコマンドを使用します。

```
service httpd24-httpd restart
```

PEM サーバーを構成したら、ブラウザで **PEM Web** インターフェイスにアクセスできます。以下に移動します:

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

3.3 隔離されたネットワークでの PEM リポジトリの作成

PEM をインストールするサーバーが EnterpriseDB リポジトリに直接アクセスできない場合、PEM RPM パッケージのホストとして機能するローカルリポジトリを作成できます。これは、必要な手順の概要であることに注意してください。個々のネットワークのプロセスを変更する必要がある場合があります。ローカルリポジトリを作成して使用するには、以下を行う必要があります。

1. 次のコマンドを使用して、インターネットアクセスが可能なマシンに PEM に必要なすべての依存関係をダウンロードします。

```
yum install yum-plugin-downloadonly

mkdir /tmp/<pem_dir>

yum install --downloadonly --downloadaddir=/tmp/<pem_dir>/ edb-pem-server

mkdir /tmp/<epel_dir>

yum install --downloadonly --downloadaddir=/tmp/<epel_dir>/ epel-release*
```

ここで、<pem_dir> と <epel_dir> は RPM をダウンロードするために作成するローカルディレクトリです。

2. ディレクトリ /tmp/<pem_dir> および /tmp/<epel_dir> を分離されたネットワーク内のマシンにコピーします。
3. リポジトリを作成します。

```
yum install createrepo

createrepo /tmp/<pem_dir>

createrepo /tmp/<epel_dir>
```

5. 以下を指定する接続情報を使用して、/etc/yum.repos.d/pem.repo というリポジトリ構成ファイルを作成します。

```
[pemrepo]
name=PEM Repository
baseurl=file:///tmp//<pem_dir>/
enabled=1
gpgcheck=0
```

6. 以下を指定する接続情報を使用して、/etc/yum.repos.d/epel.repo というリポジトリ構成ファイルを作成します。

```
[epelrepo]
name=epel Repository
baseurl=file:///tmp/<epel_dir>/
enabled=1
gpgcheck=0
```

ローカルリポジトリの場所と接続情報を指定したら、yum コマンドを使用して PEM サーバーをインストールまたはアップグレードできます。

- PEM サーバーをインストールするには:

```
yum install edb-pem
```

- PEM サーバーをアップグレードするには:

```
yum upgrade edb-pem
```

ローカル yum リポジトリの作成の詳細については、<https://wiki.centos.org/HowTos/CreateLocalRepos> にアクセスしてください。

3.4 Windows での PEM エージェントのインストール

PEM エージェントインストーラーを起動するには、Administrative 特権を想定し、インストーラーが含まれるディレクトリに移動します。次に、次のコマンドを使用してインストーラーを起動します。

```
pem_agent-7.<x>.<x>-<x>-platform.exe
```

Setup… ページが開き、PEM エージェントインストーラーが表示されます。

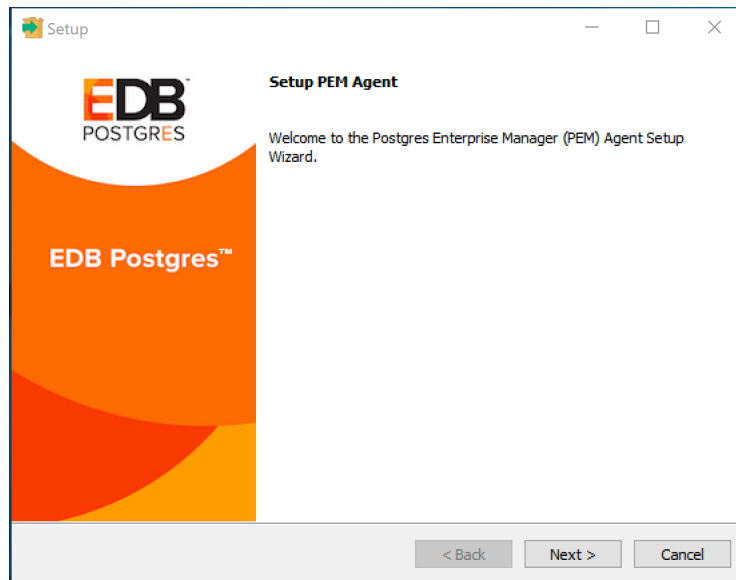


図 44 The PEM Agent Installer's Welcome dialog

Next をクリックして、License Agreement に進みます。



図 45 The PEM License Agreement

適切なラジオボタンを強調表示して契約に同意する前に、ライセンス契約を注意深く確認してください。Next をクリックして、Installation Directory ダイアログに進みます。

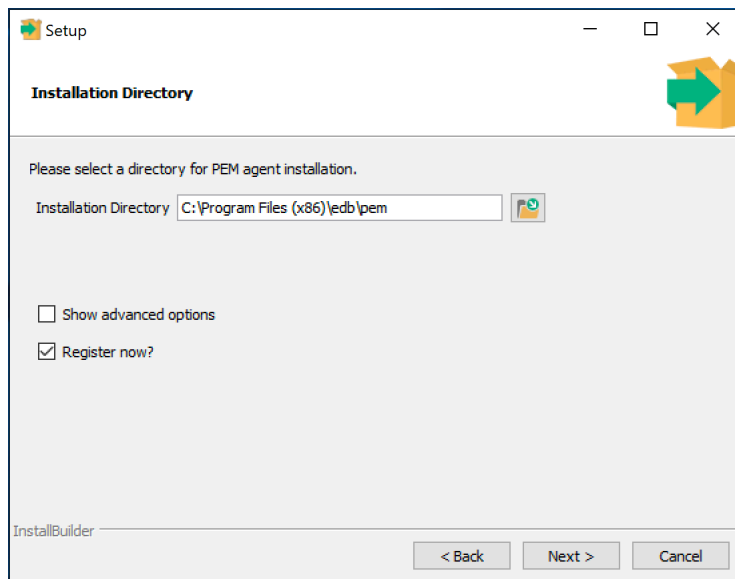


図 46 インストールディレクトリの指定

デフォルトでは、PEM エージェントは C:/Program Files (x86)/edb/pem ディレクトリにインストールされます。デフォルトのインストールディレクトリを受け入れるか、Installation Directory フィールドの内容を変更して、PEM エージェントの代替インストールディレクトリを指定できます。

デフォルトでは、PEM エージェントインストーラーは管理者の %APPDATA%/pem ディレクトリに証明書を配置します。Show advanced options ボックスをチェックして、証明書ファイルの代替パスを指定できるダイア

ログを PEM エージェントインストーラーに含めることを示します。

Register now? の隣のボックスをチェックして、新しくインストールされた PEM エージェントを PEM サーバーに登録するようインストーラーに指示します。

Next をクリックして、PEM Server Installation Details ダイアログに進みます。

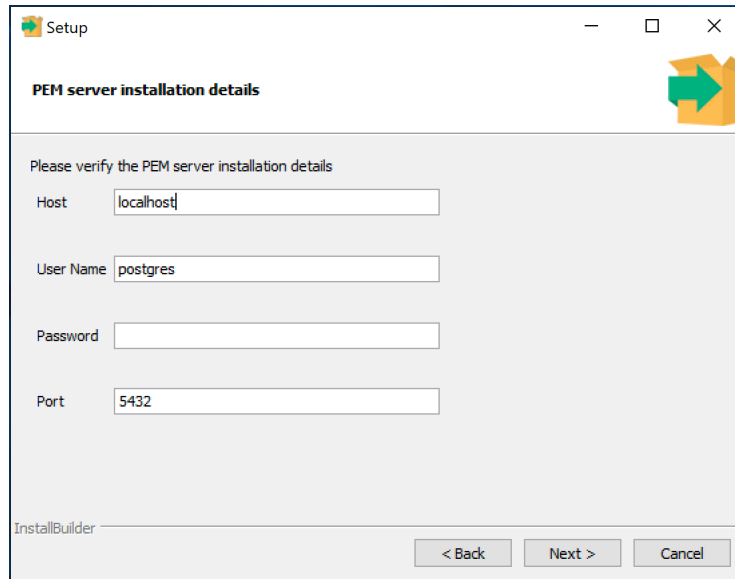


図 47 Provide PEM server installation details

PEM server installation details ダイアログで PEM サーバーの接続の詳細を入力します:

- PEM データベースサーバーが存在するシステムの名前または IP アドレスを Host フィールドに指定します。注:PEM-HTTPD Web サーバーと PEM データベースが異なるシステムでホストされている場合は、* PEM データベースのホスト*を指定する必要があります。
- User Name フィールドにデータベースのスーパーユーザーの名前を指定します。
- Password フィールドでデータベースのスーパーユーザーに関連付けられたパスワードを指定します。
- PostgreSQL が監視しているポートを Port フィールドに指定します。

Next をクリックして、pemAgent Service Account に進みます。インストーラーはサーバーに接続して、詳細が正しいことを確認しようとします。

注釈: PEM サーバーは、PEM エージェントインストーラーからの接続を許可する必要があります。接続エラーが発生した場合は、PEM Server Installation Details ダイアログで指定された接続プロパティが正しいことを確認し、pg_hba.conf ファイル（PEM サーバー上）がエラーで説明されているサーバーへの接続を許可することを確認しますメッセージ。

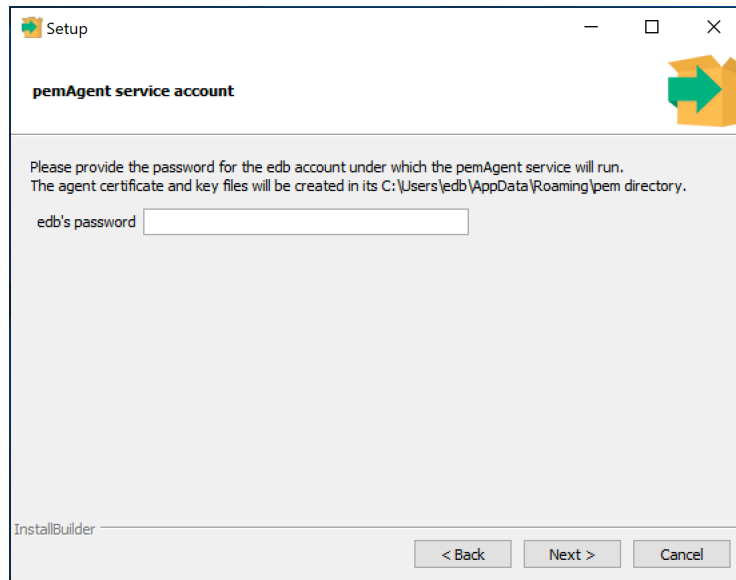


図 48 pemAgent Service Account password

pemAgent サービスを実行する edb アカウントのパスワードを提供します。エージェント証明書とキーファイルは C:/Users/edb/App に作成され、pemAgent サービスを実行する edb アカウントのパスワードを指定します。エージェント証明書とキーファイルは C:/Users/edb/AppData/Roaming/pem ディレクトリに作成されます。Next をクリックして、Agent Details ダイアログに進みます。

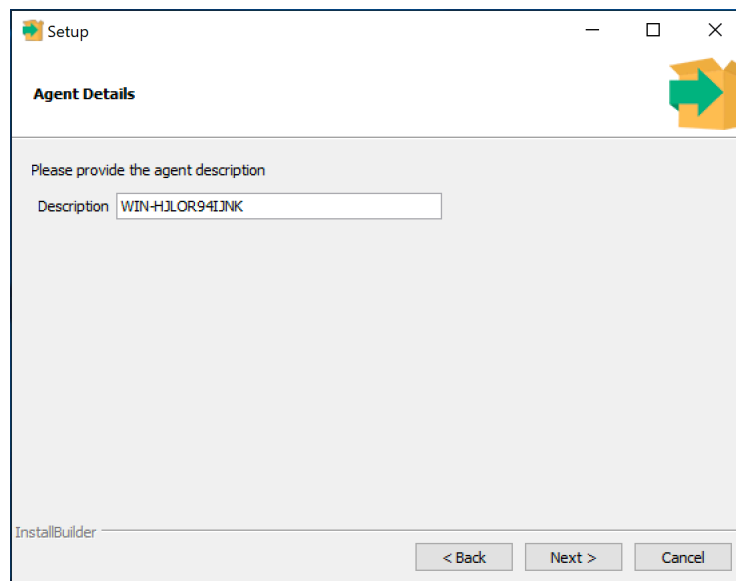


図 49 Specify a descriptive name for the PEM agent

PEM Web インターフェースの Browser パネルに表示されるツリーコントロールには、Description フィールドに入力された値が表示され、PEM エージェントを識別します。エージェントがインストールされているマシンのホスト名など、エージェントの説明的な名前、またはホストの機能を反映する名前を指定します。

わかりやすい名前を指定するか、PEM エージェントホストによって提供されるデフォルトを受け入れ、Next をクリックして続行します。

Show advanced options チェックボックスをオンにすると、Advanced options ダイアログが開きます。

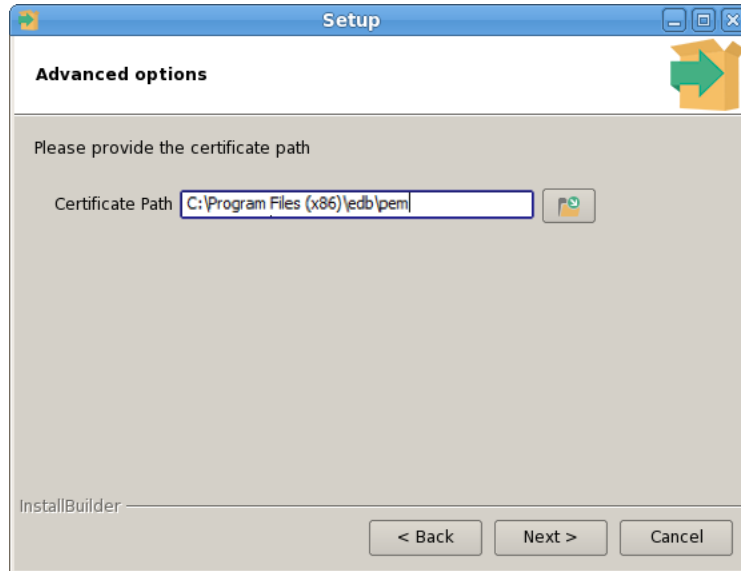


図 50 証明書のパスを提供

デフォルトでは、PEM エージェントインストーラーは証明書を C:/Program Files (x86)/edb/pem ディレクトリに配置します。証明書の代替パスを指定するか、デフォルトを受け入れて Next をクリックします。

これで、ウィザードで PEM エージェントをインストールする準備ができました。Back をクリックしてインストールディレクトリを修正するか、Next をクリックして続行します。

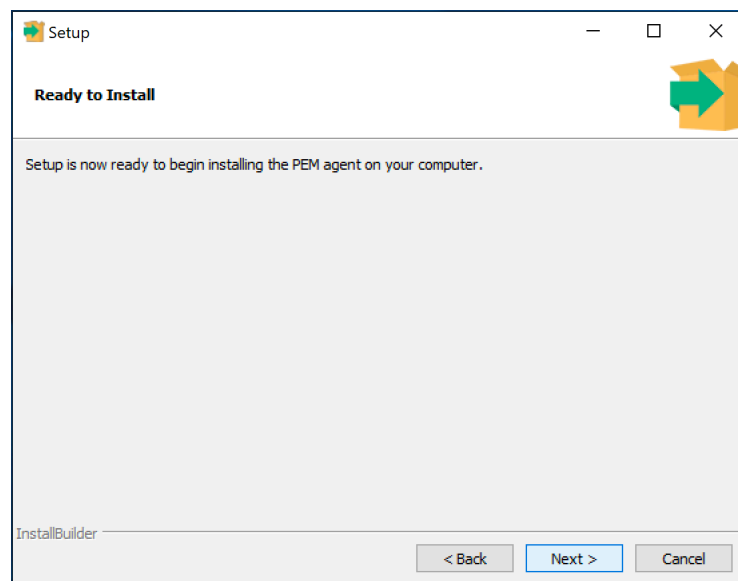


図 51 The PEM Agent installation is ready to begin

Ready to Install ダイアログで Next をクリックして、インストーラーにファイルをシステムにコピーし、PEM サーバーにエージェントを登録するよう指示します。

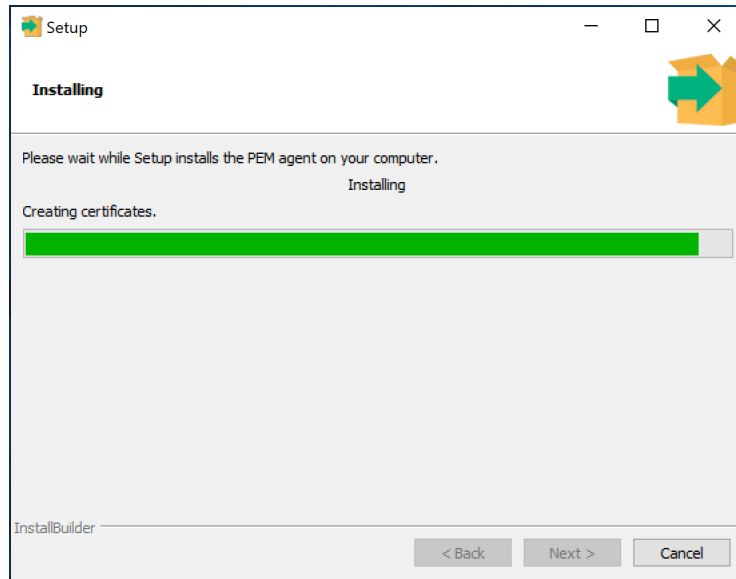


図 52 進行状況バーはインストールの進行状況を示します

PEM エージェントインストーラーには、PEM エージェントのインストールの進行状況を示す進行状況バーが表示されます。

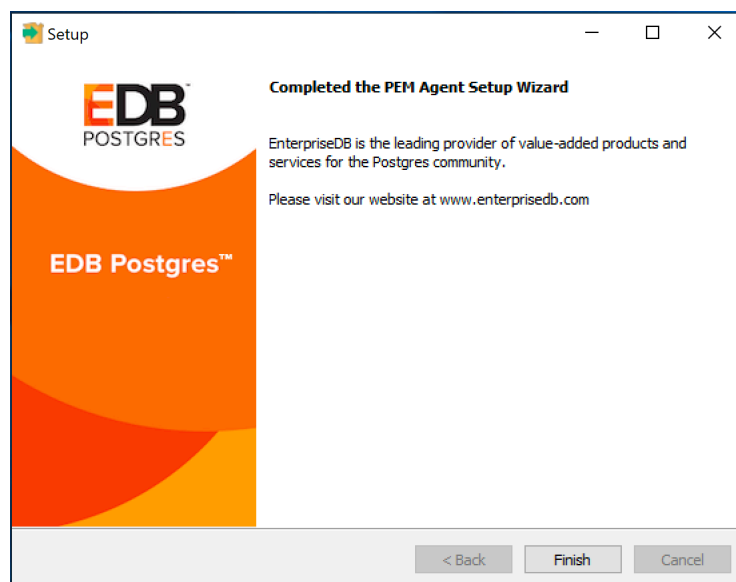


図 53 The PEM Agent installation is complete

インストールが完了すると、PEM エージェントが実行され、オペレーティングシステムとホストデータが PEM サーバーに報告されます。PEM エージェントのホストで Postgres インスタンスの監視を開始するには、PEM の

エンタープライズディレクトリに追加し、エージェントにバインドする必要があります。

3.4.1 コマンドラインからのエージェントインストーラーの呼び出し

PEM エージェントインストーラーのコマンドラインオプションは、リソースまたはシステム構成が制限されているためにグラフィカルインストールが機能しない場合に機能を提供します。次のことができます。

- インストーラーを呼び出して追加のユーザー入力なしでインストールを実行する場合は、`--mode unattended` オプションを含めます。

すべてのコマンドラインオプションがすべてのプラットフォームに適しているわけではありません。コマンドラインオプションの完全なリファレンスガイドについては、インストーラーを起動するときに `--help` オプションを含めてください。

3.4.1.1 無人モードでの PEM エージェントインストーラーの呼び出し

インストーラーを呼び出すときにコマンドラインでインストール設定を指定することにより、PEM サーバーの無人インストールを実行できます。PEM サーバーをインストールするシステムは、インターネットにアクセスできる必要があることに注意してください。

無人モードで PEM エージェントインストーラーを呼び出す前に、以下を行う必要があります。

- PEM サーバーをインストールします。PEM サーバーの `pg_hba.conf` ファイルは、PEM エージェントのホストからの接続を許可する必要があります。
- PEM エージェントのホストで SE Linux を無効にします。詳細については、セクション [ソフトウェアの前提条件を参照してください](#)。
- 監視対象の Postgres データベースで SSL が有効になっており、接続を受け入れていることを確認してください。

PEM エージェントをインストールするには管理者権限が必要です。次のコマンドを使用して、無人モードで PEM エージェントインストーラーを呼び出します。

```
pem-agent-7.<x>.<x>-windows-x64.exe --mode unattended
--pghost <pem_server_host_address> --pgport <pem_server_port>
--pguser postgres --pgpassword <pguser_password>
--agent_description <agent_name>
```

どこで:

`pem_server_host_address` は、PEM サーバーのホストの IP アドレスを指定します。

`pem_server_port` は、バックアップ PEM データベースが使用するポートを指定します。デフォルトでは、データベースはポート 5432 を使用します。

`pguser_password` は、PEM データベースのスーパーユーザーに関連付けられたパスワードを指定します。

`agent_name` は PEM エージェントの説明的な名前を指定します。

EnterpriseDB is the leading provider of value-added products and services for the ↪Postgres community.
Please visit our website at www.enterprisedb.com.

注釈: PEM 7.11 以降のバージョンがインストールされている Windows エージェントによって実行されるシェル/バッチスクリプトを構成する場合、`agent.cfg` ファイルで `AllowBatchJobSteps` パラメーターを `True` に設定する必要があります。 `pemagent` は、デフォルトではバッチ/シェルスクリプトを実行しません。

3.5 Linux での PEM エージェントのインストール

3.5.1 CentOS または RHEL ホストへの PEM エージェントのインストール

Linux システムでは、yum パッケージマネージャーを使用して PEM エージェントをインストールできます。パッケージマネージャーを使用してホストに PEM エージェントをインストールする前に、以下を行う必要があることに注意してください。

- 次のコマンドのいずれかを実行して、ホストに epel-release パッケージをインストールします。
- `yum -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm`
- `yum install epel-release`

注釈: CentOS-Base.repo ファイル（/etc/yum.repos.dにある）で [extras] リポジトリ定義を有効にする必要があるかもしれません。

- EnterpriseDB リポジトリへのアクセスを許可する資格情報も必要です。資格情報のリクエストについては、次をご覧ください。

<https://info.enterprisedb.com/rs/069-ALB-339/images/Repository%20Access%2004-09-2019.pdf>

リポジトリの資格情報を受け取った後、次のことができます。

1. リポジトリ構成ファイルを作成します。
2. ユーザー名とパスワードを指定して、ファイルを変更します。
3. edb-pem-agent をインストールします。

リポジトリ設定ファイルの作成

リポジトリ構成ファイルを作成するには、スーパーユーザー特権を引き受けて、次のコマンドを呼び出します。

```
yum -y install https://yum.enterprisedb.com/edb-repo-rpms/
edb-repo-latest.noarch.rpm
```

リポジトリ設定ファイルの名前は edb.repo です。ファイルは /etc/yum.repos.d にあります。

ファイルの変更、ユーザー名とパスワードの入力

edb.repo ファイルを作成した後、エディターの選択を使用して、有効なパラメーターの値が 1 であることを確認し、edb.repo の username および password プレースホルダーを置き換えます 4 '' 登録済みの EnterpriseDB ユーザーの名前とパスワードを含む仕様。

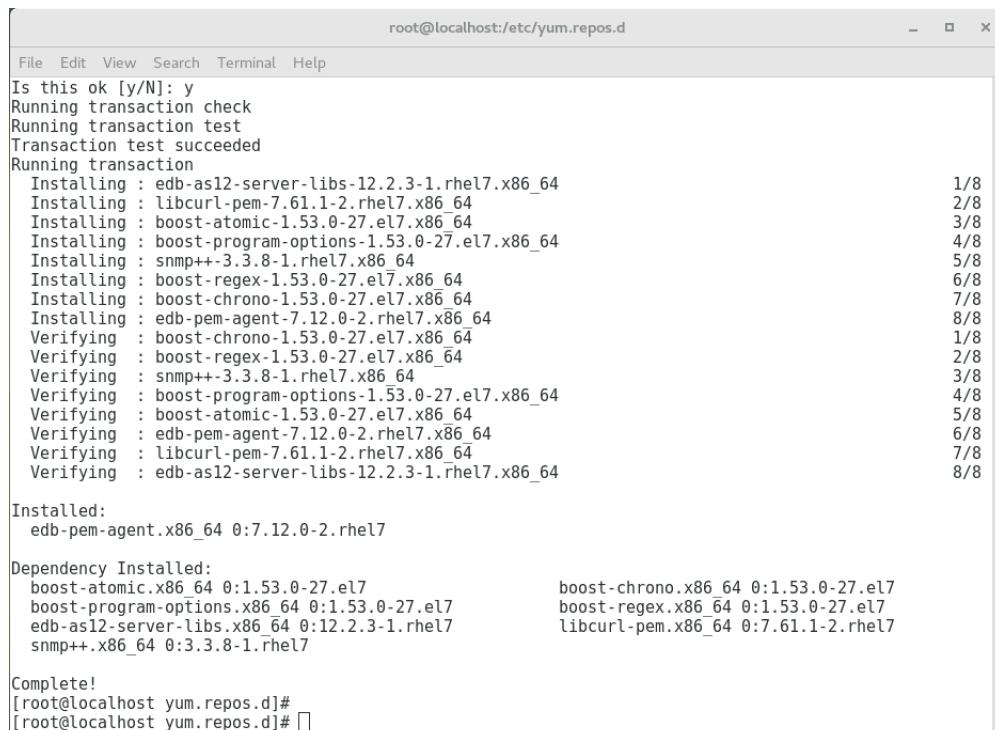
```
[edb]
name=EnterpriseDB RPMs $releasever - $basearch
baseurl=https://<username>:<password>@yum.enterprisedb.com/edb/redhat/rhel-$releasever-
↪$basearch
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/ENTERPRISEDB-GPG-KEY
```

PEM エージェントのインストール

設定ファイルへの変更を保存した後、`yum install` コマンドを使用して `edb-pem-agent` をインストールできます。たとえば、次のコマンドは PEM エージェントをインストールします。

```
yum install edb-pem-agent
```

インストールが完了すると、`yum` はインストールされたパッケージと依存関係のリストを表示します。



```
root@localhost:/etc/yum.repos.d
File Edit View Search Terminal Help
Is this ok [y/N]: y
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : edb-as12-server-libs-12.2.3-1.rhel7.x86_64 1/8
Installing : libcurl-pem-7.61.1-2.rhel7.x86_64 2/8
Installing : boost-atomic-1.53.0-27.el7.x86_64 3/8
Installing : boost-program-options-1.53.0-27.el7.x86_64 4/8
Installing : snmp++-3.3.8-1.rhel7.x86_64 5/8
Installing : boost-regex-1.53.0-27.el7.x86_64 6/8
Installing : boost-chrono-1.53.0-27.el7.x86_64 7/8
Installing : edb-pem-agent-7.12.0-2.rhel7.x86_64 8/8
Verifying : boost-chrono-1.53.0-27.el7.x86_64 1/8
Verifying : boost-regex-1.53.0-27.el7.x86_64 2/8
Verifying : snmp++-3.3.8-1.rhel7.x86_64 3/8
Verifying : boost-program-options-1.53.0-27.el7.x86_64 4/8
Verifying : boost-atomic-1.53.0-27.el7.x86_64 5/8
Verifying : edb-pem-agent-7.12.0-2.rhel7.x86_64 6/8
Verifying : libcurl-pem-7.61.1-2.rhel7.x86_64 7/8
Verifying : edb-as12-server-libs-12.2.3-1.rhel7.x86_64 8/8

Installed:
  edb-pem-agent.x86_64 0:7.12.0-2.rhel7

Dependency Installed:
  boost-atomic.x86_64 0:1.53.0-27.el7          boost-chrono.x86_64 0:1.53.0-27.el7
  boost-program-options.x86_64 0:1.53.0-27.el7  boost-regex.x86_64 0:1.53.0-27.el7
  edb-as12-server-libs.x86_64 0:12.2.3-1.rhel7  libcurl-pem.x86_64 0:7.61.1-2.rhel7
  snmp++.x86_64 0:3.3.8-1.rhel7

Complete!
[root@localhost yum.repos.d]#
[root@localhost yum.repos.d]#
```

図 54 Using an RPM package to install the PEM agent

システムで認識されないソースによって署名された RPM パッケージをインストールすると、`yum` はローカルサーバーにキーをインポートする許可を求める場合があります。プロンプトが表示され、パッケージが信頼できるソースからのものであることに満足したら、`y` を入力し、Return を押して続行します。

インストール中に、`yum` は解決できない依存関係に遭遇する場合があります。存在する場合、手動で解決する必要がある依存関係のリストが提供されます。

3.5.2 Debian または Ubuntu ホストへの PEM エージェントのインストール

Debian または Ubuntu ホストに PEM をインストールするには、EnterpriseDB リポジトリへのアクセスを許可する資格情報が必要です。リポジトリの資格情報を要求するには、**“EnterpriseDB に連絡してください<<https://www.enterprisedb.com/repository-access-request>>”**。

次の手順では、EnterpriseDB apt リポジトリを使用して Debian パッケージをインストールする手順を説明します。コマンドを使用する場合、ユーザー名とパスワードを EnterpriseDB が提供する資格情報に置き換えます。

1. <https://apt.enterprisedb.com/>にアクセスし、ルートとしてログインします。

```
sudo su -
```

2. EnterpriseDB リポジトリを構成します。

```
sh -c 'echo /"deb https://username:password@apt.enterprisedb.com/
$(lsb_release -cs)-edb/ $(lsb_release -cs) main/" > /etc/apt/sources.
list.d/edb-$(lsb_release -cs).list'
```

3. 安全な APT リポジトリのサポートをシステムに追加します。

```
apt-get install apt-transport-https
```

4. EDB 署名キーを追加します。

```
wget -q -O -https://username:password@apt.enterprisedb.com/edb-deb.gpg.
key | apt-key add -
```

5. リポジトリのメタデータを更新します。

```
apt-get update
```

6. 次のコマンドを使用して、PEM エージェントの Debian パッケージをインストールします。

```
apt-get install edb-pem-agent
```

3.5.3 SLES ホストへの PEM エージェントのインストール

Advanced Server のインストールおよび SLES ホストへのサポートコンポーネントの詳細については、

“EDB Postgres Advanced Server インストールガイド<<https://www.enterprisedb.com/edb-docs/p/edb-postgres-enterprise-manager>>”

SLES パッケージは以下から入手できます。

```
<https://zypp.enterprisedb.com>
```

PEM をインストールする前に、前提条件パッケージをインストールする必要があります。* `sp_no` *を使用して
いるサービスパック（SP2 または SP3）に置き換えて、次のコマンドを使用します。

```
SUSEConnect -p sle-module-legacy/12/x86_64

SUSEConnect -p sle-sdk/12/x86_64

zypper addrepo https://download.opensuse.org/repositories/Apache:Modules/SLE_12_<sp_no>
↪/Apache:Modules.repo

zypper addrepo http://download.opensuse.org/repositories/Cloud:/OpenStack:/Newton:/
↪cisco-apic:/2.3.1/SLE_12_<sp_no>/ pem_opensuse_boost

zypper refresh

zypper install edb-pem-agent
```

3.5.4 エージェントの登録

各 PEM エージェントは、PEM サーバーに*登録*する必要があります。登録プロセスは、エージェントと通信する
ために必要な情報を PEM サーバーに提供します。PEM エージェントのグラフィカルインストーラーはエージェ
ントの自己登録をサポートしますが、グラフィカルインストール中に PEM エージェントの登録をスキップするか、
RPM パッケージを使用して PEM エージェントをインストールする場合、`pemworker` ユーティリティを使用し
てエージェントを登録できます。

RPM インストーラーは PEM エージェントを `/usr/edb/pem/agent/bin` ディレクトリに配置します。エー
ジェントを登録するには、`pemworker` ユーティリティを呼び出すときに、`--register-agent` キーワードと
登録の詳細を含めます。

```
pemworker --register-agent
```

`pemworker` ユーティリティを呼び出すときに、コマンド文字列にコマンドラインオプションを追加します。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-server</code>	PEM バックエンドデータベースサーバーの IP アドレ スを指定します。このパラメーターは必須です。
<code>--pem-port</code>	PEM バックエンドデータベースサーバーのポートを 指定します。デフォルト値は 5432 です。
<code>--pem-user</code>	PEM バックエンドデータベースサーバーのデー タベースユーザー（スーパーユーザー特権を持つ）の名 前を指定します。このパラメーターは必須です。

次のページに続く

表 1 – 前のページからの続き

オプション	説明
<code>--pem-agent-user</code>	PEM サーバーバックエンドデータベースサーバーに接続するエージェントユーザーを指定します。
<code>--cert-path</code>	証明書が作成されるディレクトリへの完全なパスを指定します。パスを指定しない場合、証明書は次の場所に作成されます。Linux では <code>~/ .pem</code> Windows では <code>% APPDATA %/ pem</code>
<code>--config-dir</code>	構成ファイルが見つかるディレクトリパスを指定します。デフォルトは <code><pemworker path> /../etc</code> 。
<code>--display-name</code>	PEM Browser ツリーコントロールに表示されるエージェントのわかりやすい名前を指定します。デフォルトはシステムのホスト名です。
<code>--force-registration</code>	<code>force_registration</code> 句を含めて、指定された引数でエージェントを登録するように PEM サーバーに指示します。この句は、既存のエージェント構成をオーバーライドする場合に役立ちます。デフォルト値は “はい” です。
<code>--group</code>	エージェントが表示されるグループの名前。
<code>--team</code>	監視対象データベースサーバーにアクセスする必要がある PEM バックエンドデータベースサーバー上のデータベースロールの名前。
<code>--owner</code>	エージェントを所有する PEM バックエンドデータベースサーバー上のデータベースユーザーの名前。
<code>--allow_server_restart</code>	<code>allow-server_restart</code> パラメーターを有効にして、PEM が監視対象サーバーを再起動できるようにします。デフォルト値は <code>True</code> です。
<code>--allow-batch-probes</code>	<code>allow-batch-probes</code> パラメーターを有効にして、PEM がこのエージェントでバッチプローブを実行できるようにします。デフォルト値は <code>False</code> です。
<code>--batch-script-user</code>	バッチ/シェルスクリプトの実行に使用するオペレーティングシステムユーザーを指定します。デフォルト値は <code>none</code> です。このパラメーターを空白のままにした場合、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。
<code>--enable-heartbeat-connection</code>	<code>enable-heartbeat-connection</code> パラメーターを有効にして、PEM エージェントとサーバー間に専用のハートビート接続を作成し、アクティブステータスを更新します。デフォルト値は <code>False</code> です。

次のページに続く

表 1 – 前のページからの続き

オプション	説明
<code>--enable-smtp</code>	<code>enable-smtp</code> パラメーターを有効にして、PEM エージェントが PEM サーバーの代わりに電子メールを送信できるようにします。デフォルト値は <code>False</code> です。
<code>--enable-snmp</code>	<code>enable-snmp</code> パラメーターを有効にして、PEM エージェントが PEM サーバーに代わって SNMP トラップを送信できるようにします。デフォルト値は <code>False</code> です。
<code>-o</code>	構成ファイルのオプションをオーバーライドするかどうかを指定します。

Audit Manager、Log Manager、Tuning Wizard など、`pemagent` でデータベースサーバーの再起動が必要な PEM 機能を使用する場合は、`allow_server_restart` の値を `true` として設定する必要があります `agent.cfg` ファイル。

注釈: PEM 7.11 以降のバージョンがインストールされている PEM エージェントによって実行されるシェル/バッチスクリプトを構成する場合、`batch_script_user` parameter のユーザーを指定する必要があります。スクリプトの実行には、非 `root` ユーザーを使用することを強くお勧めします。 `root` ユーザーを使用すると、データのセキュリティとオペレーティングシステムのセキュリティが低下する可能性があります。ただし、スクリプトを実行するために `root` ユーザーを使用して `pemagent` を元の設定に復元する場合は、`batch_script_user` パラメーター値を `root` に設定する必要があります。

PEM データベースで変更が行われる前に、接続エージェントは PEM データベースサーバーで認証されます。 `pemworker` ユーティリティを呼び出すときは、PEM サーバー管理ユーザーロールに関連付けられたパスワード (`postgres`) を提供する必要があります。管理パスワードを指定するには 3 つの方法があります。できること:

- `PEM_MONITORED_SERVER_PASSWORD` 環境変数を設定します。
- コマンドラインで `PGPASSWORD` キーワードを使用してパスワードを指定します。
- `.pgpass` ファイルにエントリを作成します。

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。他の必須だが省略された情報の入力を求められます。登録が完了すると、サーバーはエージェントが正常に登録されたことを確認します。

PEM エージェント構成パラメーターの設定

PEM エージェント RPM インストーラーは、`/usr/edb/pem/agent/etc` ディレクトリに `agent.cfg.sample` という名前のサンプル構成ファイルを作成します。 PEM エージェントを登録すると、`pemworker` プログラムが実際のエージェント構成ファイル (`agent.cfg` という名前) を作成します。 `agent.cfg` ファイルを修正し、次の設定パラメーターを追加する必要があります:

```
heartbeat_connection = true
```

ca-bundle.crt ファイル（認証局）の場所も追加する必要があります。デフォルトでは、インストーラーは agent.cfg.sample ファイルで指定された場所に ca-bundle.crt ファイルを作成します。サンプルファイルからデフォルトのパラメーター値をコピーするか、別の場所に保存されている ca-bundle.crt ファイルを使用する場合は、その値を ca_file パラメーターに指定します。

```
ca_file=/usr/libexec/libcurl-pem7/share/certs/ca-bundle.crt
```

次に、プラットフォーム固有のコマンドを使用して、PEM エージェントサービスを開始します。サービスの名前は pemagent です。たとえば、CentOS または RHEL 6.x システムでは、次のコマンドを使用します。

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

CentOS または RHEL 7.x ホストで、systemctl を使用してサービスを開始します。

```
systemctl start pemagent
```

サービスは、エージェントを開始していることを確認します。エージェントが登録および起動されると、PEM Web インターフェースの Global Overview および Object browser に表示されます。

pemworker ユーティリティを使用してサーバーを登録する方法については、pemworker を参照してください <<https://www.enterprisedb.com/resources/product-documentation>>‘_ _

PEM Web インターフェイス

PEM サーバーとエージェントをインストールした後、PostgreSQL または Advanced Server インスタンスの監視と管理を開始するように PEM を構成できます。PEM サーバーインストーラーは、PEM Web インターフェイスをインストールします。インターフェイスを使用して、監視対象サーバーに存在するオブジェクトに関する情報を確認したり、PEM サーバーによって収集された統計情報を確認したりできます。

PEM をインストールして構成したら、ブラウザを使用して PEM Web インターフェイスにアクセスできます。ブラウザを開き、次の場所に移動します。

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

Where *ip_address_of_PEM_host* specifies the IP address of the host of the PEM server. The Postgres Enterprise Manager Web Login window opens:

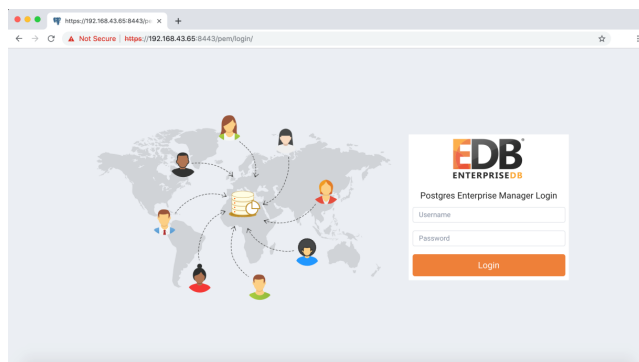


図 1 The PEM Web Login page

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。

Login ボタンをクリックして、PEM サーバーに接続します。

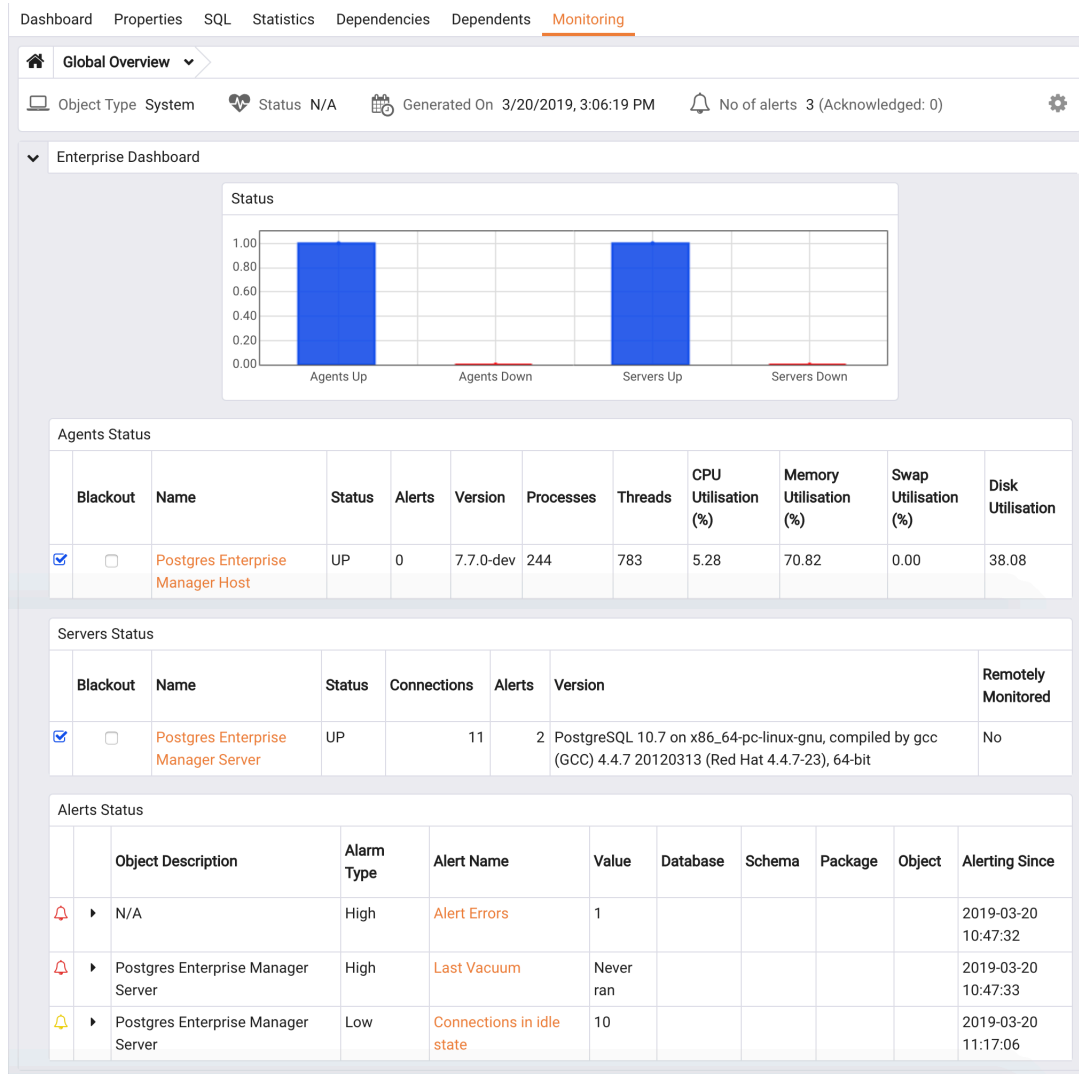


図2 The PEM Web interface

PEM Web インターフェイスを使用してデータベースサーバーを管理または監視する前に、サーバーを PEM サーバーに*登録*する必要があります。サーバーを登録するとき、サーバーへの接続を記述し、接続の認証情報を提供し、管理設定を指定します（オプションでエージェントをバインドします）。

サーバーは管理されている場合と管理されていない場合があります。

- managed サーバーは PEM エージェントにバインドされています。PEM エージェントは、バインドされているサーバーを監視し、PEM ダッシュボードに表示するためにタスクを実行したり統計をレポートします。管理対象サーバーは、パッケージ管理やカスタムアラートなどの拡張 PEM 機能にアクセスできます。サーバーを登録するときに、必要に応じて管理対象サーバーを PEM で再起動することもできます。
- unmanaged サーバーは PEM エージェントにバインドされていません。非管理対象サーバーでデータベースオブジェクトを作成できますが、拡張 PEM 機能（パッケージ管理やカスタムアラートなど）は非管理対象サーバーではサポートされていません。

また、登録するサーバーの `pg_hba.conf` ファイルが PEM Web インターフェイス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。

PEM Web インターフェイスに関するオンラインヘルプ情報にアクセスするには、メニューバーから Help を選択します。追加情報は、Help から `.pdf` および `.html` 形式で入手できます。 <<https://www.enterprisedb.com/resources/product-documentation>>‘__

- PEM Administrator's Guide には、サーバー、エージェント、ユーザーの登録と管理に関する情報が含まれています。
- PEM Enterprise Features Guide には、Web インターフェイスの一部であるツールとウィザードの使用に関する情報が含まれています。
- PEM Agent User Guide には、PEM エージェントの管理に関する有用な情報が含まれています。
- PEM Upgrade and Migration Guide には、以前のバージョンから PEM 7.8 へのアップグレードに関する情報が含まれています。
- PEM PgBouncer Configuration Guide には、PEM インストールでの PgBouncer の使用に関する情報が含まれています。
- PEM EDB Ark Management Guide には、PEM を使用して Advanced Server と PostgreSQL のクラウドインストールを管理することに関する情報が含まれています。

第 5

SQL プロファイラープラグインのインストール

SQL プロファイラープラグインを使用すると、サーバーのワークロードをプロファイルできます。SQL プロファイラーを使用する各サーバーにプラグインをインストールする必要があります。たとえば、PostgreSQL 9.6 および PostgreSQL 10 を実行しているホストがある場合は、各サーバーに 1 つずつ、2 つのバージョンのプラグインをインストールする必要があります。

SQL プロファイラーは、PostgreSQL バージョン 9.4 以降および Advanced Server バージョン 9.4 以降の EnterpriseDB ディストリビューションでのみ公式にサポートされています。プラグインは StackBuilder を介して配布されるか、**‘EnterpriseDB Web サイトから入手できます<<https://www.enterprisedb.com/advanced-downloads>>’**

—

プラグインは、Advanced Server インストーラーのサーバーコンポーネントとともに配布およびインストールされます。

SQL Profiler プラグインは、PEM エージェントの有無にかかわらずサーバーにインストールできますが、トレースは非管理対象サーバーでアドホックモードでのみ実行でき、管理対象サーバーでのみスケジュールできます。

Configuration セクションに進む前に、以下のインストール手順に従って PostgreSQL のプラグインをインストールしてください。Advanced Server を使用している場合は、Configuration セクションまでスキップできます。

グラフィカルインストーラーを使用して、Windows プラットフォームに任意のバージョンの SQL プロファイラーをインストールできます。

Linux では、グラフィカルインストーラーを使用してデータベースサーバーをインストールした場合、グラフィカルインストーラーを使用して SQL プロファイラーをインストールする必要があります。RPM パッケージを使用

してデータベースサーバーをインストールした場合は、RPM パッケージを使用して SQL プロファイラーをインストールする必要があります。

5.1 グラフィカルインストーラーを使用した SQL プロファイラーのインストール

SQL Profiler グラフィカルインストーラーを起動するには、スーパーユーザー特権（または Windows の場合は Administrator 特権）を想定し、インストーラーを含むディレクトリに移動し、次のコマンドでインストーラーを起動します:

```
postgresql<pg_version>-sqlprofiler-<sql_profiler_version>
```

ここで、pg_version は postgres のバージョン、sql_profiler_version は SQL プロファイラーのバージョンです。

SQL Profiler インストーラーがセットアップウィザードにようこそ。



図 1 The SQL Profiler Installer Welcome dialog

Next をクリックして、License Agreement に進みます。



図 2 The SQL Profiler License Agreement

適切なラジオボタンを強調表示して契約に同意する前に、ライセンス契約を注意深く確認してください。Next をクリックして、Installation Directory ダイアログに進みます。

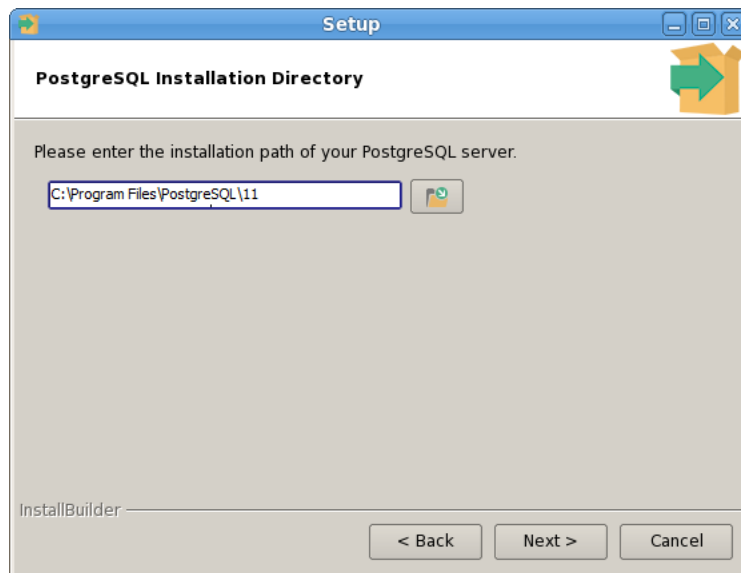


図 3 The PostgreSQL Installation Directory

インストールディレクトリの別の場所を指定するか、デフォルトの場所を受け入れて Next をクリックして続行します。

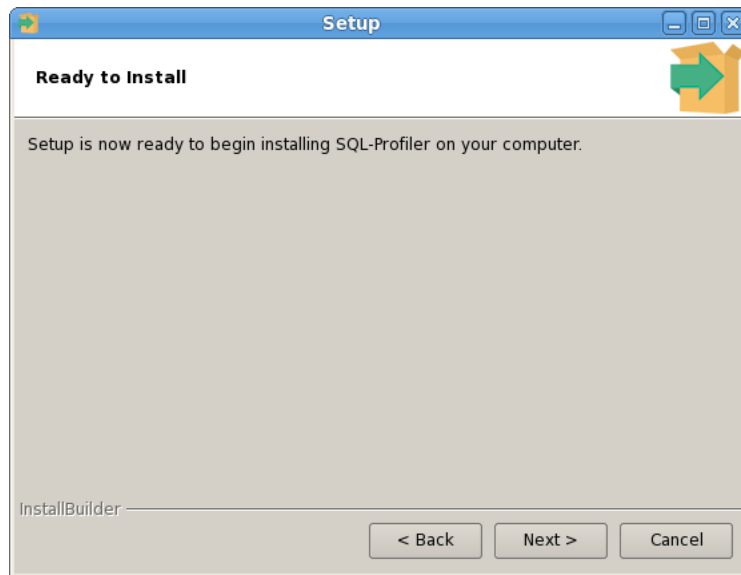


図 4 インストールの準備

これで、ウィザードで SQL Profiler プラグインをインストールする準備ができました。Next をクリックして続行します。

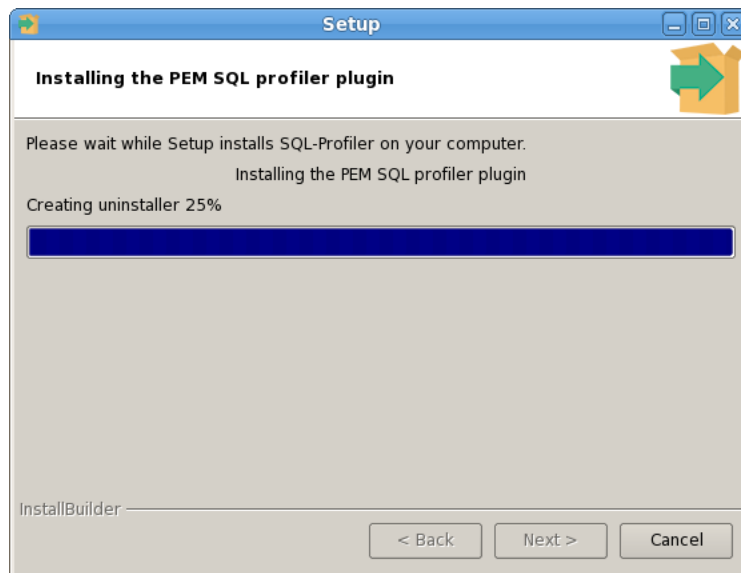


図 5 Installing the SQL Profiler plugin

SQL Profiler プラグインインストーラーは、システムにファイルをコピーするときに進行状況バーを表示します。

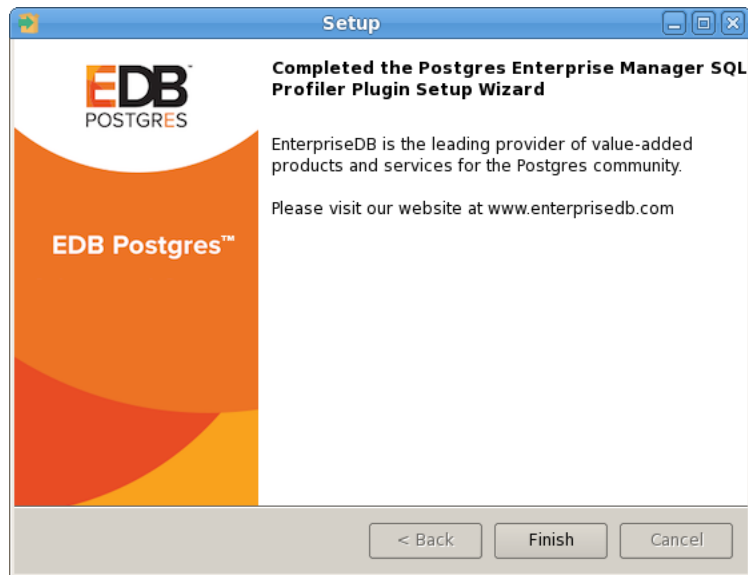


図 6 The SQL Profiler installation is complete

インストールが完了すると、SQL Profiler プラグインを構成する準備が整います。

5.2 RPM を使用した SQL プロファイラーのインストール

root ユーザーとして yum コマンドを使用して、RHEL または Centos 6 または 7 で rpm を使用して SQL プロファイラーをインストールできます。

```
yum install postgresql<pg_version>-sqlprofiler-<sql_profiler_version>
```

ここで、pg_version は postgres のバージョン、sql_profiler_version は SQL プロファイラーのバージョンです。

5.3 SQL プロファイラーの構成

インストール処理が完了しても、SQL Profiler プラグインは自動的に有効になりません。これにより、都合の良いときにサーバーを再起動でき、継続的に必要とされないシステムでプラグインが不必要にロードされるのを防ぎます。

プラグインを有効にするには、次の手順を使用します。

1. プロファイルするサーバー上の postgresql.conf ファイルを編集し、以下に示すように shared_preload_libraries パラメーターを変更します。

```
shared_preload_libraries = '$libdir/sql-profiler'
```

2. Postgres サーバーを再起動します。
3. Query Tool または psql コマンドラインインターフェースを使用して、プロファイリングしたいサーバーで Maintenance Database として指定されたデータベースで sql-profiler.sql スクリプトを実行します。使用している場合:
 - PostgreSQL、デフォルトのメンテナンスデータベースは postgres です。
 - Advanced Server、デフォルトのメンテナンスデータベースは edb です。

PEM クエリツールを使用してスクリプトを実行するには、Browser ツリーコントロールでメンテナンスデータベースの名前を強調表示し、Tools メニューをナビゲートして Query tool を選択します。クエリツールが開いたら、Files メニューの Open オプションを使用して Web ブラウザーを開き、sql-profiler.sql スクリプトに移動します。デフォルトでは、sql-profiler.sql スクリプトは Postgres インストールの下 contrib フォルダーにあります。

クエリツールの SQL Editor パネルでスクリプトが開いたら、SQL エディターでスクリプトの内容を強調表示し、Query メニューから Execute オプションを選択します（または “1 '' アイコン）を押してスクリプトを呼び出し、SQL プロファイラーを設定します。

psql コマンドラインを使用して、構成スクリプトを呼び出すこともできます。次のコマンドは、psql を使用して、Linux システム上の Advanced Server データベースで sql-profiler.sql スクリプトを呼び出します。

```
$ /usr/edb/as10/bin/psql -U postgres postgres <
/usr/edb/as10/share/contrib/sql-profiler.sql
```

SQL プロファイラーを構成すると、サーバー上にあるすべてのデータベースで使用できるようになります。

SQL プロファイラー機能にアクセスするには、**PEM** の **Browser ツリーコントロール** でデータベースの名前を強調表示します。 **Tools** メニューの **Server オプション** から **SQL Profiler プルアサイドメニュー** に移動します。メニューオプションを使用すると、**SQL** トレースを管理できます。

- **Create trace …** を選択して、新しいトレースを定義します。
- **Open trace …** を選択して、既存のトレースを開きます。
- 1 つ以上のトレースを削除するには、**Delete trace(s) …** を選択します。
- **View scheduled trace(s) …** を選択して、スケジュールされたトレースのリストを確認します。

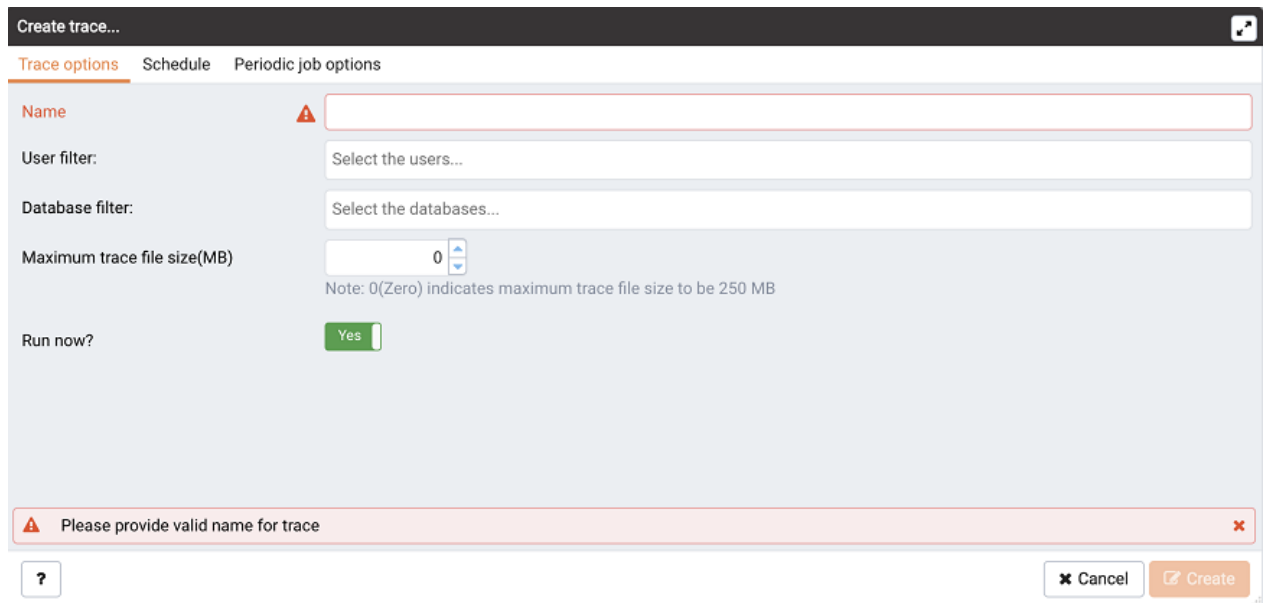


図 7 新しいトレースの作成

SQL プロファイラーの使用の詳細については、**SQL** プロファイラーのオンラインヘルプテキスト（**Help** メニューからアクセス）または *** PEM 管理者ガイド *** を参照してください。

第 6

PEM コンポーネントのアップグレードとアンインストール

PEM コンポーネントのアップグレードと移動の詳細については、* PEM アップグレードおよび移行ガイド*を参照してください。 PEM のドキュメントと PEM および SQL Profiler インストーラーへのリンクは、“EnterpriseDB Web サイト<<https://www.enterprisedb.com/resources/product-documentation>>’_ _

Windows™ での Postgres Enterprise Manager のアップグレード

Windows で PEM のバージョンをアップグレードするには、PEM コンポーネントインストーラーを起動するだけです。各インストーラーは、以前のインストールを検出すると通知し、PEM のより新しいバージョンにアップグレードします。

次の順序で PEM インストーラーを呼び出す必要があります。

1. PEM サーバーノードを除く各監視対象ノードで PEM エージェントインストーラーを呼び出します。
2. PEM サーバーインストーラーを起動します。このインストーラーは、PEM サーバーと PEM サーバーノードにある PEM エージェントの両方をアップグレードします。

以前のバージョンからのアップグレードの詳細については、* PEM Upgrade and Migration Guide *を参照してください。

Linux ホストでの SQL プロファイラーのアップグレード

Linux ホストにある SQL Profiler インストールをアップグレードするには:

1. `uninstall-sql-profiler.sql` スクリプトを呼び出して、各ノードの既存の SQL プロファイラーのエリセットを削除します。
- デフォルトでは、Linux ホストで Advanced Server を使用している場合、スクリプトは次の場所にあります。

`/opt/PostgresPlus/x.x/share/contrib`

- Linux ホストで PostgreSQL インストールを使用している場合、スクリプトは次の場所にあります。

`/opt/PostgreSQL/x.x/share/postgresql/contrib`

ここで、*xx* はインストールされている Postgres バージョンを指定します。

2. 次に、プロファイリングする各ノードで新しい SQL Profiler インストーラーを呼び出します。

Windows ホストでの SQL プロファイラーのアップグレード

Windows ホストで SQL プロファイラーを使用している場合、Windows は実行またはメモリにロードされたファイルをロックします。ロックされたファイルを解放するには、アップグレードを実行する前に Postgres サーバーを停止する必要があります。

Postgres サーバーを停止した後:

1. `uninstall-sql-profiler.sql` スクリプトを呼び出して、各ノードの既存の SQL プロファイラーのエリセットを削除します。

- Windows ホストで Advanced Server を使用している場合、スクリプトは次の場所にあります。

`C:/Program Files/PostgresPlus/x.x/share/contrib`

- Windows ホストで PostgreSQL インストールを使用している場合、スクリプトは次の場所にあります。

`C:/Program Files/PostgreSQL/x.x/share/contrib`

ここで、*x* はインストールされた Postgres バージョンを指定します。

2. プロファイリングする各ノードで新しい SQL Profiler インストーラーを呼び出します。

次に、Postgres サーバーを再起動して、ノードのプロファイリングを再開します。

トラブルシューティング

Windows ホストでの SQL プロファイラーのアップグレード

SQL プロファイラーの新しいバージョンにアップグレードした後、次のエラーが発生した場合:

```
An error has occurred:
ERROR: function return row and query-specified return row do not
      match
DETAIL: Returned row contains 11 attributes, but query expects 10.
```

このエラーを修正するには、既存のクエリセットを新しいクエリセットに置き換える必要があります。まず、`uninstall-sql-profiler.sql` スクリプトを呼び出して SQL プロファイラーをアンインストールし、次に `sql-profiler.sql` スクリプトを呼び出して SQL プロファイラーを再インストールします。

Postgres Enterprise Manager のアンインストール

PEM サーバーまたはエージェントをアンインストールするプロセスはプラットフォーム固有です。PEM サーバーのパッケージ名は `edb-pem-server` で、PEM エージェントのパッケージ名は `edb-pem-agent` です。

ホストから PEM サーバーパッケージをアンインストールする場合、同じホストにインストールされている PEM エージェントパッケージはアンインストールされません。ただし、PEM エージェントパッケージをアンインストールすると、同じホストにインストールされている PEM サーバーパッケージもアンインストールされます。

Windows ホストからの PEM のアンインストール

PEM インストールが Windows ホストにある場合、Windows の Add/Remove Programs アプリケーションを使用して PEM コンポーネントを削除できます。Windows の Control Panel から Add/Remove Programs オプションを選択します。control panel が開いたら、プログラムリストで PEM コンポーネントの名前を見つけてみます。コンポーネントを削除するには、Remove ボタンをクリックします。

次の場所にあるアンインストーラーを呼び出すこともできます。

```
C:/Program Files/edb/pem/server
```

CentOS または RHEL ホストからの PEM のアンインストール

`rpm`、`yum remove`、または “`yum erase`” コマンドのバリエーションを使用して、インストールされたパッケージを削除できます。パッケージを削除しても、PEM データディレクトリが破損することはありません。

- `rpm` コマンドを呼び出してインストール済みパッケージを削除する場合は、`-e` オプションを含めます。コマンド構文は次のとおりです。

```
rpm -e <package_name>
```

- `yum remove` コマンドを使用して、`yum` によってインストールされた `pem` サーバーまたはエージェントパッケージを削除できます。パッケージを削除するには、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。

```
yum remove <package_name>
```

- `yum erase` コマンドを使用して、`edb-pem` および `edb-pem-docs` の依存関係とともに `pem` サーバーまたはエージェントパッケージを削除できます。パッケージを削除するには、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。

```
yum erase <package_name>
```

Where *package_name* is the name of the package that you would like to remove.

Debian または Ubuntu ホストからの PEM のアンインストール

`apt-get remove` または `apt-get purge` コマンドを使用して、Debian または Ubuntu ホストから PEM サーバーまたはエージェントパッケージをアンインストールできます。

- 構成ファイルとデータディレクトリに影響を与えずに Debian または Ubuntu ホストから PEM サーバーまたはエージェントをアンインストールするには、次のコマンドを呼び出します。

```
apt-get remove <package_name>
```

- 構成ファイルとデータディレクトリとともに PEM サーバーまたはエージェントをアンインストールするには、次のコマンドを呼び出します。

```
apt-get purge <package_name>
```

Where *package_name* is the name of the package that you would like to remove.

SLES ホストからの **PEM** のアンインストール

SLES ホストから PEM サーバーまたはエージェントをアンインストールするには、次のコマンドを呼び出します。

```
zypper remove <package_name>
```

Where *package_name* is the name of the package that you would like to remove.

第 7

リファレンス-Linux サービススクリプト

PEM サーバーが存在する Postgres サーバーには、サービススクリプトが含まれている必要があります。EnterpriseDB によって生成された Postgres インストーラーがサービススクリプトを作成します。別のソースから Postgres サーバーを使用している場合は、サービススクリプトを提供する必要があります。

提供されていないソースからインストールまたはビルドされた Postgres インストール用のスクリプトを開発する際の出発点として、Linux サービススクリプトの次の例を使用できます。（この例からコピーして貼り付ける場合）改行が正しくコピーされることを確認してください。

```
| #!/bin/bash
| # chkconfig: 2345 85 15
| # description: Starts and stops the PostgreSQL/Postgres Plus Advanced
Server database server
| # PostgreSQL/Postgres Plus Advanced Server Service script template for
Linux
| # Please modify the values accordingly
| DB_DESC="Database Server - PostgreSQL 9.5"
| DB_INSTALL_DIR=/opt/PostgreSQL/9.5
| DB_BIN_DIR=${DB_INSTALL_DIR}/bin
| DB_LIB_DIR=${DB_INSTALL_DIR}/lib
| DB_DATA_DIR=${DB_INSTALL_DIR}/data
| DB_HBA_FILE=${DB_DATA_DIR}/pg_hba.conf
| DB_CONF_FILE=${DB_DATA_DIR}/postgresql.conf
| DB_PID_FILE=${DB_DATA_DIR}/postmaster.pid
| DB_STARTUP_LOG=${DB_DATA_DIR}/pg_log/startup.log
| DB_SERVICE_USER=postgres
| _die()
```

(次のページに続く)

(前のページからの続き)

```

| {
|     echo ""
|     echo "FATAL ERROR: $*"
|     echo ""
|     exit 1
| }
| if [ `id -u` != 0 ]; then
|     _die "You must run this script as the root."
| fi
|
| # Source function library.
| if [ -f /etc/rc.d/functions ];
| then
|     . /etc/init.d/functions
| fi
start()
| {
|     STARTDBSERVER=0
|     if [ -e "${DB_PID_FILE}" ]
|     then
|         PIDOFDB=`head -n 1 "${DB_PID_FILE}"`
|         PIDALIVEDB=""
|         if [ -n "${DB_PID_FILE}" ]; then
|             PIDALIVEDB=`ps -p "${PIDOFDB}" | grep
"${PIDOFDB}"`
|             fi
|             if [ -n "${PIDALIVEDB}" ]
|             then
|                 echo "The '${DB_DESC}' is already running.
|                     PID(${PIDALIVEDB})."
|                     exit
|             else
|                 STARTDBSERVER=1
|             fi
|             else
|                 STARTDBSERVER=1
|             fi
|         fi
|     if [ "${STARTDBSERVER}" != 0 ]
|     then
|         echo "Starting ${DB_DESC}..."
|         su - "${DB_SERVICE_USER}" -c
|         "LD_LIBRARY_PATH=\"${DB_LIB_DIR}:\$LD_LIBRARY_PATH\"
|         \"${DB_BIN_DIR}/pg_ctl\" -w start -D \"${DB_DATA_DIR}\" -l
|         \"${DB_STARTUP_LOG}\" -o \"${DB_STARTUP_OPTIONS}\""
|         if [ $? -eq 0 ];
|         then
|             echo "${DB_DESC} started successfully."
|             exit 0

```

(次のページに続く)

(前のページからの続き)

```

| else
| echo "${DB_DESC} did not start in a timely fashion, please see
| "${DB_STARTUP_LOG}" for details."
| exit 1
| fi
| fi
| }
| stop()
| {
| if [ -e "${DB_PID_FILE}" ]
| then
| PIDOFDB=`head -n 1 "${DB_PID_FILE}"`
| PIDALIVEDB=""
| if [ -n "${DB_PID_FILE}" ]; then
| PIDALIVEDB=`ps -p "${PIDOFDB}" | grep "${PIDOFDB}"`
| fi
| if [ -n "${PIDALIVEDB}" ]
| then
| echo "Stopping ${DB_DESC}..."
| su - "${DB_SERVICE_USER}" -c
| "LD_LIBRARY_PATH=\"${DB_LIB_DIR}:${LD_LIBRARY_PATH}\"
| "${DB_BIN_DIR}/pg_ctl" stop -m fast -D \"${DB_DATA_DIR}\"
| -l \"${DB_STARTUP_LOG}\" -o \"${DB_STARTUP_OPTIONS}\""
| else
| echo "The '${DB_DESC}' is not running."
| fi
| else
| echo "The '${DB_DESC}' is not running."
| fi
| }
| reload()
| {
| echo "Reloading '${DB_DESC}'..."
| su - "${DB_SERVICE_USER}" -c
| "LD_LIBRARY_PATH=\"${DB_LIB_DIR}:${LD_LIBRARY_PATH}\"
| "${DB_BIN_DIR}/pg_ctl" reload -D \"${DB_DATA_DIR}\" -l
| \"${DB_STARTUP_LOG}\" -o \"${DB_STARTUP_OPTIONS}\""
| }
| restart()
| {
| echo "Restarting '${DB_DESC}'..."
| su - "${DB_SERVICE_USER}" -c
| "LD_LIBRARY_PATH=\"${DB_LIB_DIR}:${LD_LIBRARY_PATH}\"
| "${DB_BIN_DIR}/pg_ctl" restart -m fast -w -D
| \"${DB_DATA_DIR}\" -l \"${DB_STARTUP_LOG}\" -o
| \"${DB_STARTUP_OPTIONS}\""
| if [ $? -eq 0 ];
| then

```

(次のページに続く)

(前のページからの続き)

```

| echo "${DB_DESC}" restarted successfully."
| exit 0
| else
| echo "${DB_DESC}" did not start in a timely fashion, please see
| "${DB_STARTUP_LOG}" for details."
| exit 1
| fi
| }
| _die_incomplete_requirement()
| {
| echo "One or more required configuration variables are not set:"
| _die $*
| }
| _validate_script()
| {
| if [ -z "${DB_INSTALL_DIR}" ]; then
| _die_incomplete_requirement "Missing installation directory";
| fi
| if [ ! -d "${DB_INSTALL_DIR}" ]; then
| _die_incomplete_requirement "The specified - '${DB_INSTALL_DIR}'
is not a valid installation directory. It is not present on the
system.";
| fi
| if [ -z "${DB_BIN_DIR}" ]; then
DB_BIN_DIR=${DB_INSTALL_DIR}/bin; fi
| if [ ! -d "${DB_BIN_DIR}" ]; then
| _die_incomplete_requirement "The specified - '${DB_BIN_DIR}' is
not a valid bin directory. It is not present on the system.";
| fi
| if [ ! -f "${DB_BIN_DIR}/pg_config" -o ! -f
"${DB_BIN_DIR}/pg_ctl" ]; then
| _die_incomplete_requirement "The specified - '${DB_BIN_DIR}' does
not contain the database server binaries.";
| fi
| if [ -z "${DB_LIB_DIR}" ]; then
DB_LIB_DIR=${DB_INSTALL_DIR}/lib; fi
| if [ -z "${DB_DESC}" ]; then DB_DESC=`"${DB_BIN_DIR}/pg_config
--version`; fi
| if [ -z "${DB_DATA_DIR}" ]; then
| _die_incomplete_requirement "Missing data directory settings in the
script. Please set 'DB_DATA_DIR' variable in the script.";
| fi
| if [ ! -d "${DB_DATA_DIR}" ]; then
| _die_incomplete_requirement "The specified - '${DB_DATA_DIR}' is
not a valid. It is not present on the system.";
| fi
| if [ ! -f "${DB_DATA_DIR}/PG_VERSION" -o ! -d
"${DB_DATA_DIR}/base" -o ! -d "${DB_DATA_DIR}/global" ]; then

```

(次のページに続く)

(前のページからの続き)

```

| _die_incomplete_requirement "The directory - '${DB_DATA_DIR}'
does not look like a valid PostgreSQL/Postgres Plus Advanced Server
data directory."
| fi
| if [ -z "${DB_SERVICE_USER}" ]; then
| _die_incomplete_requirement "The service-user is not specified in
the service script. Please set 'DB_SERVICE_USER' variable in the
script."
| fi
| DB_VALID_SERVICE_USER=`cat /etc/passwd | grep
"^${DB_SERVICE_USER}:"`
| if [ -z "${DB_VALID_SERVICE_USER}" ]; then
| _die_incomplete_requirement "The service-user
'${DB_SERVICE_USER}' is not present on the system. Please specify
the correct information."
| fi
| DB_DATA_DIR_OWNER=`ls -l ${DB_DATA_DIR}/PG_VERSION | awk
'{print $3}'`
| if [ x"${DB_DATA_DIR_OWNER}" != x"${DB_SERVICE_USER}" ]; then
| _die_incomplete_requirement "The specified user -
'${DB_SERVICE_USER}' does not own the data directory -
'${DB_DATA_DIR}'. The data directory is owned by the user -
'${DB_DATA_DIR_OWNER}'."
| fi
| if [ -z "${DB_HBA_FILE}" ]; then
DB_HBA_FILE=${DB_DATA_DIR}/pg_hba.conf; fi
| if [ ! -f "${DB_HBA_FILE}" ]; then
| _die_incomplete_requirement "The hba-file - '${DB_HBA_FILE}' does
not exist."
| fi
| if [ -z "${DB_CONF_FILE}" ]; then
DB_CONF_FILE=${DB_DATA_DIR}/postgresql.conf; fi
| if [ ! -f "${DB_CONF_FILE}" ]; then
| _die_incomplete_requirement "The config-file - '${DB_CONF_FILE}'
does not exist."
| fi
| if [ -z "${DB_PID_FILE}" ]; then
DB_PID_FILE=${DB_DATA_DIR}/postmaster.pid; fi
| if [ -z "${DB_STARTUP_LOG}" ]; then
DB_STARTUP_LOG=${DB_DATA_DIR}/pg_log/startup.log; fi
| DB_STARTUP_OPTIONS=""
| if [ x"${DB_CONF_FILE}" != x"${DB_DATA_DIR}/postgresql.conf" ];
then
| DB_STARTUP_OPTIONS="-c 'config_file=${DB_CONF_FILE}'"
| fi
| if [ x"${DB_HBA_FILE}" != x"${DB_DATA_DIR}/pg_hba.conf" ]; then
| DB_STARTUP_OPTIONS="${DB_STARTUP_OPTIONS} -c
'hba_file=${DB_HBA_FILE}'"

```

(次のページに続く)

(前のページからの続き)

```
fi

if [ x"${DB_PID_FILE}" != x"${DB_DATA_DIR}/postmaster.pid" ]; then

| DB_STARTUP_OPTIONS="${DB_STARTUP_OPTIONS} -c
'external_pid_file=${DB_PID_FILE}'"
| fi
| if [ x"${DEBUG_VALIDATION}" = x"1" ]; then
| echo "Using these values in the scripts:"
| echo ""
| echo "DB_DESC : ${DB_DESC}"
| echo ""
| echo "DB_INSTALL_DIR : ${DB_INSTALL_DIR}"
| echo "DB_BIN_DIR : ${DB_BIN_DIR}"
| echo "DB_LIB_DIR : ${DB_LIB_DIR}"
| echo ""
| echo "DB_DATA_DIR : ${DB_DATA_DIR}"
| echo "DB_HBA_FILE : ${DB_HBA_FILE}"
| echo "DB_CONF_FILE : ${DB_CONF_FILE}"
| echo "DB_PID_FILE : ${DB_PID_FILE}"
| echo "DB_STARTUP_LOG : ${DB_STARTUP_LOG}"
| echo ""
| echo "DB_SERVICE_USER : ${DB_SERVICE_USER}"
| echo "DB_STARTUP_OPTIONS : ${DB_STARTUP_OPTIONS}"
| echo ""
| fi
| }
| DEBUG_VALIDATION=0
| # See how we were called.
| case "$1" in
| start)
| _validate_script
| start
| ;;
| stop)
| _validate_script
| stop
| ;;
| reload)
| _validate_script
| reload
| ;;
| restart)
| _validate_script
| restart
| ;;
| condrestart)
```

(次のページに続く)

(前のページからの続き)

```

| _validate_script

if [ -e "${DB_PID_FILE}" ]
then
PIDOFDB=`head -n 1 "${DB_PID_FILE}"`
PIDALIVEDB=""
if [ -n "${DB_PID_FILE}" ]; then
PIDALIVEDB=`ps -p "${PIDOFDB}" | grep
"${PIDOFDB}"`
fi
if [ -n "${PIDALIVEDB}" ]
then
restart
else
echo "The '${DB_DESC}' is not running."
fi
else
echo "The '${DB_DESC}' is not running."
fi
;;
status)
_validate_script
su - "${DB_SERVICE_USER}" -c
"LD_LIBRARY_PATH=\"${DB_LIB_DIR}:\$LD_LIBRARY_PATH\"
\"${DB_BIN_DIR}/pg_ctl\" status -D \"${DB_DATA_DIR}\" -l
\"${DB_STARTUP_LOG}\" -o \"${DB_STARTUP_OPTIONS}\""
| ;;
| validate)
| DEBUG_VALIDATION=1
| _validate_script
| exit 0
| ;;
| *)
| echo "Usage: $0
{start|stop|restart|condrestart|reload|status|validate}"
| exit 1
| esac

```

EDB Postgres Enterprise Manager インストールガイド

Copyright©2007-2020 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、EDB 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、および入力検証の検証を行います。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

Configuring SQL Profiler, 81
 Creating a Service Script or Registering the Service, 35

Hardware Prerequisites, 5

installing agent on Linux, 65
 installing agent on windows, 56
 installing PEM, 7
 Installing PEM Agent on a SLES Host, 67
 installing PEM Server and PEM-HTTPD on same Host, 10
 installing PEM Server and PEM-HTTPD on separate Hosts, 18
 installing PEM server on Linux, 48
 installing PEM server on Windows, 8
 installing PEM server through graphical installer, 8
 Installing SQL Profiler using graphical installer, 77
 Installing SQL Profiler using RPMs, 81
 Installing the PEM Server on an Existing Postgres Server, 33
 Installing the SQL Profiler Plugin, 75
 Installing the sslutils Extension, 33
 Installing the sslutils Extension on Linux, 33
 Installing the sslutils Extension on Windows, 34
 invoking agent Installers from command line, 63
 invoking PEM server installer in unattended mode, 45
 invoking server installers from command line, 45
 Invoking the PEM Agent Installer in Unattended Mode, 63
 Invoking the PEM Server Installer, 36

Postgres Enterprise Manager-概要, 3
 Postgres Enterprise Manager のインストール, 7

Reference - Linux Service Script, 87
 Registering a Service on Windows, 36
 Registering an Agent, 68

Software Pre-Requisites, 6

The PEM Web Interface, 72

Uninstalling Postgres Enterprise Manager, 84
 Upgrading and Uninstalling PEM Components, 83

Upgrading Postgres Enterprise Manager on Linux, 83
 Upgrading Postgres Enterprise Manager on windows, 83
 Upgrading SQL Profiler on a Linux Host, 83
 Upgrading SQL Profiler on a Windows Host, 84
 Upgrading SQL Profiler on a Windows Host - Troubleshooting, 84

Writing a Linux Service Script, 35

隔離されたネットワークでの PEM リポジトリの作成, 54

結論, 94
 新機能, 2