Postgres Enterprise Manager

Release 7.13

PEM Installation Guide

Feb 25, 2020

Contents

Postgres Enterprise Manager (PEM) is designed to assist database administrators, system architects, and performance analysts when administering, monitoring, and tuning PostgreSQL and Advanced Server database servers. PEM has been designed to manage and monitor a single server or multiple servers from a single console, allowing complete control over monitored databases.

This document provides step-by-step instructions to guide you through the installation of Postgres Enterprise Manager.

For information about the platforms and versions supported by PEM, visit the EnterpriseDB website at:

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

Throughout this guide, the term *Postgres* refers to either a PostgreSQL or an Advanced Server installation, where either is appropriate.

Language pack installers contain supported languages that may be used with EDB Postgres Advanced Server and EnterpriseDB PostgreSQL database installers. The language pack installer allows you to install Perl, TCL/TK, and Python without installing supporting software from third party vendors. For more information about installing and using Language Pack, please see the *EDB Postgres Language Pack Guide*, available from the EnterpriseDB Website.

CHAPTER 1

What's New

The following features have been added to create Postgres Enterprise Manager 7.13:

- Core Usage Report: The Core Usage report provides you with metrics such as total number of cores used by a EDB Postgres Advanced Server or Postgres database that is being managed by PEM. This report can help you understand if you are in compliance with core-based licensing guidelines.
- SMMP v3 support: SNMP v3 support enables the PEM Server and PEM Agent to use SNMP Version 3 for secure communication by authenticating and encrypting data packets over the network.
- Schema Diff tool (Beta): The Schema Diff tool allows you to compare two database schema objects and synchronize the two schemas. The tool lists the differences and also generates the synchronization scripts for the two schemas.
- Auto logout inactive users: An inactive user session is automatically logged out after a certain period of inactivity. This timeout interval can be configured in the `config.py` file. This feature improves security by reducing the unintended access to the database server.
- Accessibility Improvements: Enables screen-readers to read labels and descriptions of non-text elements, to identify the alert errors, and to read relationship attributes in nested elements.
- Other features include:
  - Enhancements to the System Configuration Report.
  - Support for a multi-level partitioned table.

CHAPTER 2

Postgres Enterprise Manager - Overview

Postgres Enterprise Manager (PEM) consists of components that provide the management and analytical features of PEM:

- **PEM Server:** The PEM server is used as the data repository for monitoring data and as a server to which both agents and clients connect. The PEM server consists of an instance of PostgreSQL and an associated database for storage of monitoring data, and a server that provides web services.
- **PEM web interface:** The PEM web interface allows you to manage and monitor Postgres servers and utilize PEM extended functionality. The web interface software is installed with the PEM server installer, and is accessed via your choice of web browser.
- **PEM Agent:** The PEM agent is responsible for executing tasks and reporting statistics from the agent host and monitored Postgres instances to the PEM server. A single PEM agent can monitor multiple installed instances of Postgres that reside on one or many hosts.

  The PEM Agent installer creates two executables: the PEM worker (`pemworker.exe`) and the PEM agent (`pemagent.exe`). Each PEM worker has a corresponding PEM agent that you can use to start or stop the PEM worker. The PEM agent will also restart the PEM worker should it terminate unexpectedly.

  The PEM worker log file contains information related to PEM worker activity (probe activities, heartbeat responses, etc.), and is stored in `/var/log/pem/worker.log`.
- **SQL Profiler plugin:** This plugin to the Postgres server is used to generate the monitoring data used by the SQL Profiler tool. Installation of the SQL Profiler plugin is optional, but the plugin must be installed into each instance of Postgres you wish to profile. The SQL Profiler may be used with any supported version of an EnterpriseDB distribution of a PostgreSQL server or an Advanced Server (not just those managed through the PEM server). See *Installing the SQL Profiler Plugin* for details and supported versions.

The architectural diagram below illustrates the relationship between the various servers and workstations involved in a typical PEM installation.

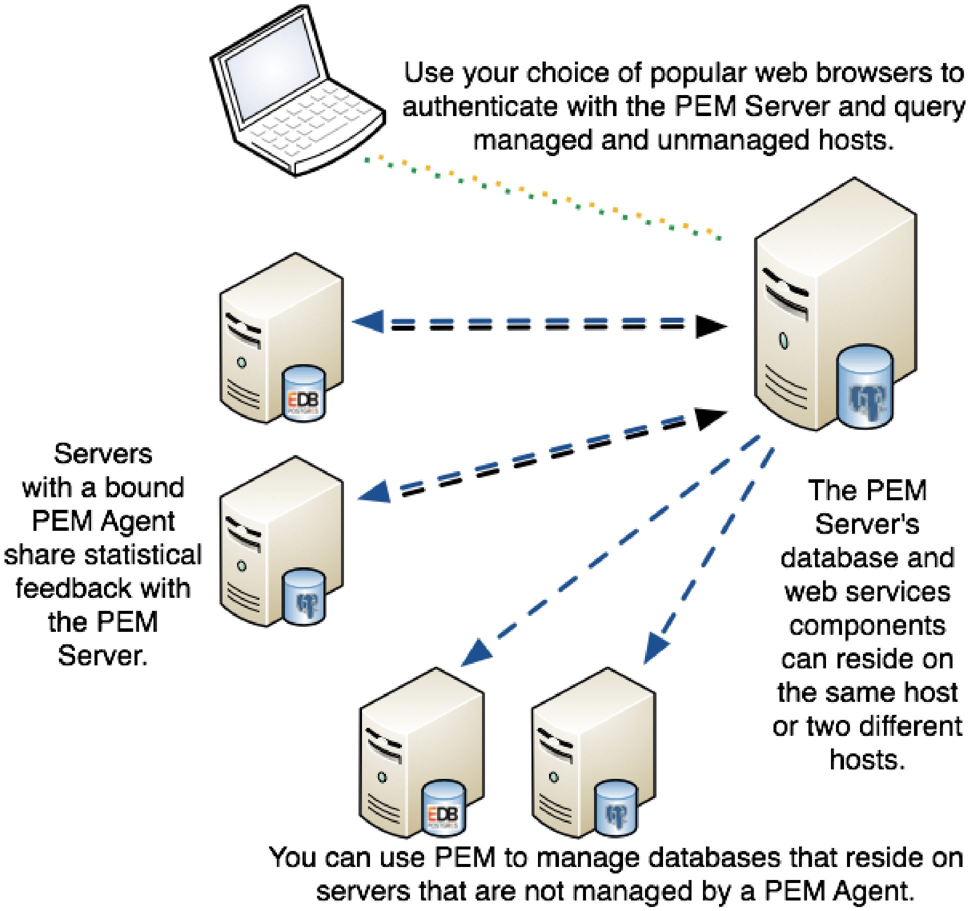

Use your choice of popular web browsers to authenticate with the PEM Server and query managed and unmanaged hosts.
Servers with a bound PEM Agent share statistical feedback with the PEM Server.
EDB
The PEM Server's database and web services components can reside on the same host or two different hosts.
EDB
You can use PEM to manage databases that reside on servers that are not managed by a PEM Agent.

Fig. 2.1: *A typical PEM installation*

2.1 Hardware Prerequisites

For optimum speed when monitoring servers and rendering dashboards, we recommend installing PEM on a system with at least:

- 4 CPU cores
- 8 GB of RAM
- 100 GB of Storage

Additional disk space is required for data storage. Please note that resource usage will vary based on which probes are defined and enabled, and the activity level on the monitored databases. Monitoring server resources (as you use PEM) will let you know when you need to expand your initial system configuration.

2.2 Software Prerequisites

Modifying the pg_hba.conf File

The `pg_hba.conf` file manages connections for the Postgres server. You must ensure that the `pg_hba.conf` file on each monitored server allows connections from the PEM server, the monitoring PEM agent, and the host of the PEM-HTTPD server.

For information about modifying the `pg_hba.conf` file, see the *PEM Administrator's Guide* available at:

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

Information about managing authentication is also available in the Postgres core documentation available at:

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

Firewall Restrictions

Please note that you must adjust your firewall to allow communication between PEM components.

PEM Agent Installation- Prerequisites for RHEL or CentOS 7.x

Before installing the PEM agent on a CentOS 7.x host, you must ensure that the host contains the `epel-release` packages. To install the packages, open a command line, assume `root` privileges, and invoke the commands:

```
yum install epel-release
```

Windows Permissions

If you are installing PEM components on Windows, you may be required to invoke the PEM installers with `Administrative` privileges. To invoke an installer using Administrator privileges, right click on the installer icon and select `Run as Administrator` from the context menu.

Supported Locales

Currently, the PEM server and web interface support a locale of `English(US) en_US` and use of a period (.) as a language separator character. Using an alternate locale, or a separator character other than a period may result in errors.

Windows Server IE Security

If you are using Internet Explorer on a Windows Server host to access monitored servers with the PEM web interface, you must disable Internet Explorer's Enhanced Security to take advantage of PEM functionality. If you do not wish to disable IE Enhanced Security, we recommend that you use an alternate browser (such as Chrome).

CHAPTER 3

Installing Postgres Enterprise Manager

You can use a graphical installer, StackBuilder Plus, or an RPM package to add the Postgres Enterprise Manager server or agent to a host. Installers are available on the EnterpriseDB website.

The PEM server graphical installer for Windows installs and configures the PEM server, a PEM agent, and the software required to connect to the PEM web interface with your choice of browser. For detailed information about using the PEM server graphical installer, see *Installing the PEM Server on Windows*.

You can use an RPM package to install the PEM server on a Linux host; for detailed information about using the installer, see *Installing the PEM Server on Linux*.

The PEM agent graphical installer for Windows installs and registers the PEM agent. The PEM agent that is installed with the PEM server is capable of monitoring multiple servers that reside on the same host, or on remote hosts. Please note that the PEM functionality on servers monitored by a remote agent may be limited.

For more information about using the PEM agent graphical installer, see *Installing a PEM Agent on Windows*.

You can use an RPM package to install the PEM agent on Linux; for more information about the PEM agent RPM package, see *Installing a PEM Agent on Linux*.

3.1 Installing the PEM Server on Windows

At the heart of each PEM installation is the server. In a production environment, the server will typically be a dedicated machine, monitoring a large number of Postgres servers or a smaller number of busy servers.

The PEM server backend database may be an EnterpriseDB distribution of the PostgreSQL or Advanced Server database server, or an existing Postgres server installed from another source. The Postgres backing database server must be version 9.4 or later, and will contain a database named pem, which is used by the PEM server as a repository.

- If you would like to use an existing Postgres server to host the PEM server, the PEM server installer can create the `pem` database on the Postgres host. You must manually satisfy the software pre-requisites if you choose to use an existing server.

  For more information about using an existing Postgres server to host the PEM server backend database, see *Installing the PEM Server on an Existing Postgres Server* section.

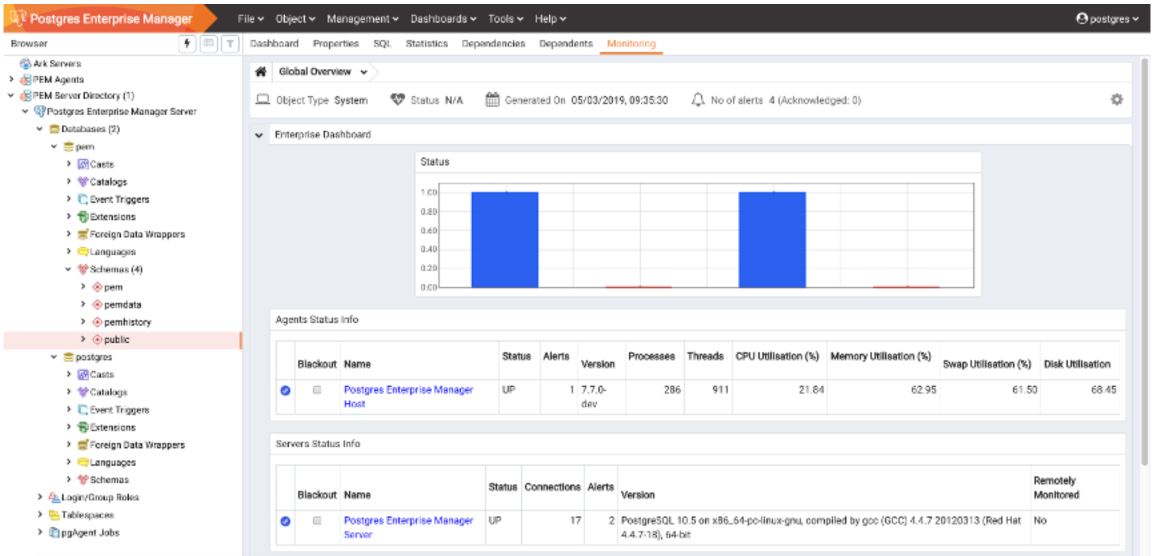
- If you do not wish to use an existing installation of Postgres as the PEM server host, the PEM server installer can install PostgreSQL, satisfy the server host's software pre-requisites, and create an instance (a PostgreSQL database cluster) that contains the `pem` database.

  This is the simplest PEM server installation option.

PEM-HTTPD is made available for Postgres installations through the PEM server installer or the StackBuilder utility. If PEM-HTTPD is already installed on the host, the PEM server installer will review and update the existing installation if required. If the PEM server host does not contain an existing PEM-HTTPD installation, the PEM server installer will add it.

Before installing the PEM server, you must decide if you wish to run PostgreSQL and PEM-HTTPD on the same host or on separate hosts. If you intend to run the PostgreSQL database server and PEM-HTTPD on different hosts, then you must run the PEM server installer twice – once on each host, as detailed in *Installing the PEM Server and PEM-HTTPD on Separate Hosts* section.

The PEM server installer will also install the software required to access the server via the PEM web interface. You can access the web interface with a supported version of your browser of choice.



Object Type System   Status N/A   Generated On 05/03/2019, 09:35:30   No of alerts 4 (Acknowledged: 0)

Agents Status Info

	Blackout	Name	Status	Alerts	Version	Processes	Threads	CPU Utilisation (%)	Memory Utilisation (%)	Swap Utilisation (%)	Disk Utilisation
●	□	Postgres Enterprise Manager Host	UP	1	7.7.0-dev	286	911	21.84	62.95	61.50	68.45

Servers Status Info

	Blackout	Name	Status	Connections	Alerts	Version	Remotely Monitored
●	□	Postgres Enterprise Manager Server	UP	17	2	PostgreSQL 10.5 on x86_64-pc-linux-gnu, compiled by gcc (GCC) 4.4.7 20120313 (Red Hat 4.4.7-18), 64-bit	No

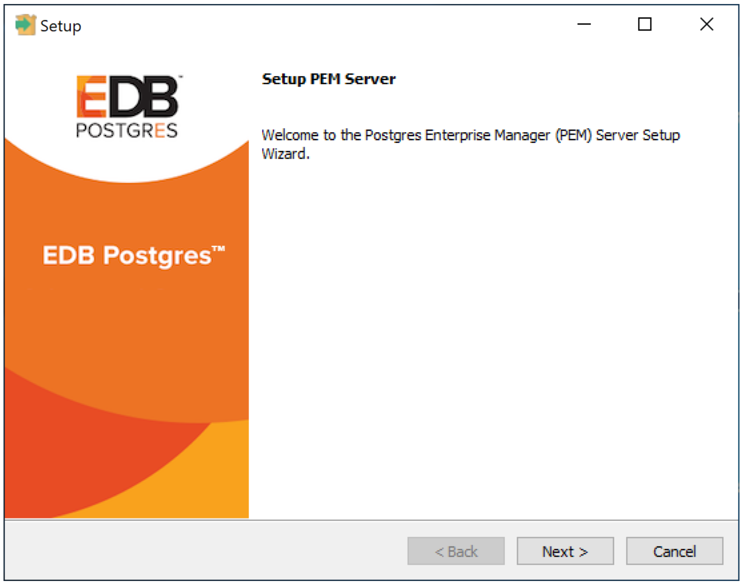
Fig. 3.1: *The PEM web interface*

You can use the web interface to review information about objects that reside on monitored servers, manage databases and database objects that reside on monitored servers, or review statistical information gathered by the PEM server. The interface also provides access to clusters that reside on registered EDB Ark consoles.

3.1.1 Installing the PEM Server and PEM-HTTPD on the Same Host

The easiest PEM server installation configuration consists of a PEM backend database server (hosted on a PostgreSQL database installed with the PEM server installer) and a PEM-HTTPD service that reside on the same host. In this configuration, the PEM server installer will provide the pre-requisite software for the backend host and create a service script (on Linux) or register the service (on Windows).

To invoke the PEM server installer on a Windows system, right click the installer icon and select `Run as Administrator`. The installer displays a `Welcome` dialog.



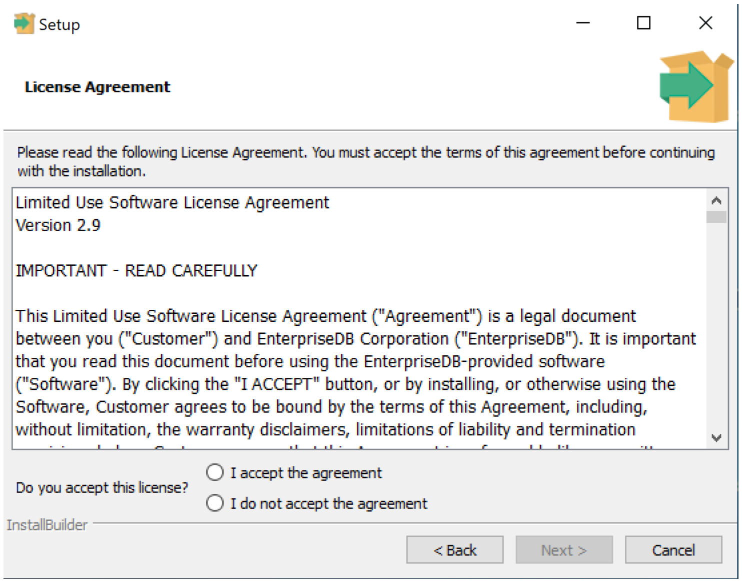
Setup

Setup PEM Server

Welcome to the Postgres Enterprise Manager (PEM) Server Setup Wizard.

Fig. 3.2: *The PEM Server Installer's Welcome dialog*

Click `Next` to continue to the `License Agreement` dialog.



Setup

License Agreement

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

Limited Use Software License Agreement
Version 2.9

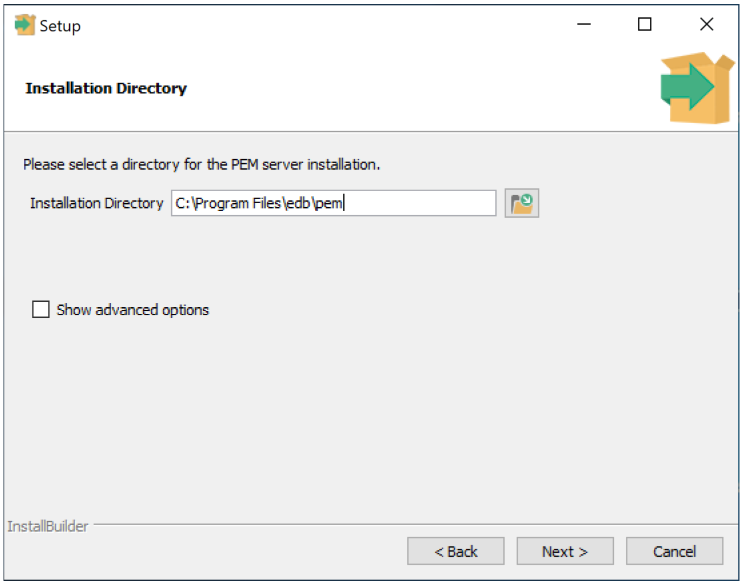
IMPORTANT - READ CAREFULLY

This Limited Use Software License Agreement ("Agreement") is a legal document between you ("Customer") and EnterpriseDB Corporation ("EnterpriseDB"). It is important that you read this document before using the EnterpriseDB-provided software ("Software"). By clicking the "I ACCEPT" button, or by installing, or otherwise using the Software, Customer agrees to be bound by the terms of this Agreement, including, without limitation, the warranty disclaimers, limitations of liability and termination

Do you accept this license? ○ I accept the agreement ○ I do not accept the agreement

Fig. 3.3: *The License Agreement*

Carefully review the license agreement before highlighting the appropriate radio button and accepting the agreement. Click Next to continue to the Installation Directory dialog.



Please select a directory for the PEM server installation.

Installation Directory C:\Program Files\edb\pem

□ Show advanced options

Fig. 3.4: *Specify an installation directory*

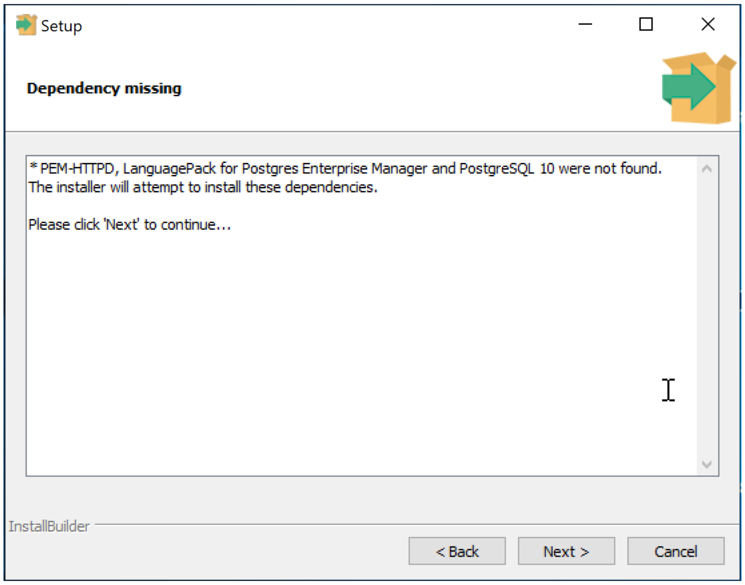
Use the Installation Directory dialog to specify the location of the PEM server:

- By default, the PEM server is installed in C:\Program Files\edb\pem on Windows. Accept the default location, or use the Installation Directory button to open a browser dialog and select the directory in which the PEM server will be installed.
- Use the Show advanced options check box to instruct the installer to open the Advanced options dialog.
- Use the Advanced options dialog when installing the Postgres database server and the PEM-HTTPD on different hosts, or if you wish the PEM server to reside on an existing Postgres server installation.

To install the PostgreSQL server packaged with the installer and PEM-HTTPD on the same host, leave the Show advanced options box unchecked and click Next.

The PEM server installer will perform a pre-installation check for PEM-HTTPD, Language Pack, and PostgreSQL 10. If the installer does not locate these packages, it will inform you in the Dependency missing dialog.

Note: By default EDB Language Pack is installed in C:\edb\languagepack\v1.



Setup

Dependency missing

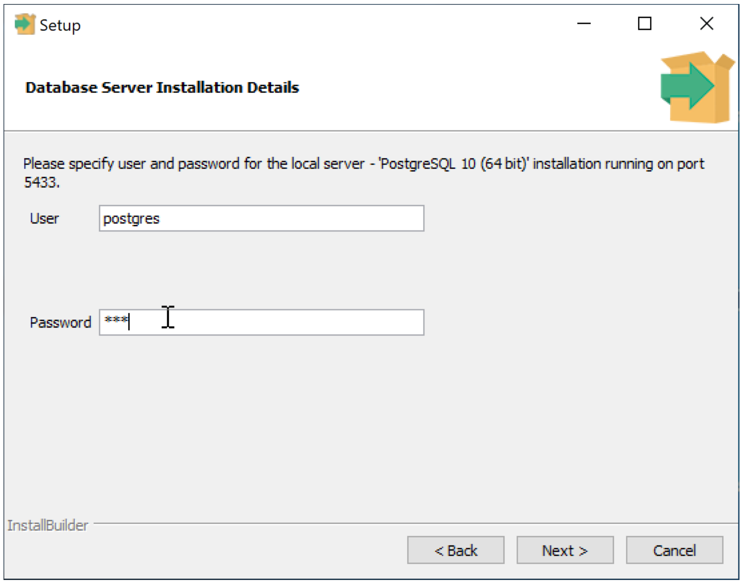
* PEM-HTTPD, LanguagePack for Postgres Enterprise Manager and PostgreSQL 10 were not found.
The installer will attempt to install these dependencies.

Please click 'Next' to continue...

Fig. 3.5: *The installer checks for pre-requisites*

If the dependencies are missing, the PEM server installer will launch the respective installation wizards; follow the onscreen directions presented by the installation wizards for each package.

After installing any missing dependencies, the installation process continues by displaying the `Database Server Installation Details` dialog.



Setup

Database Server Installation Details

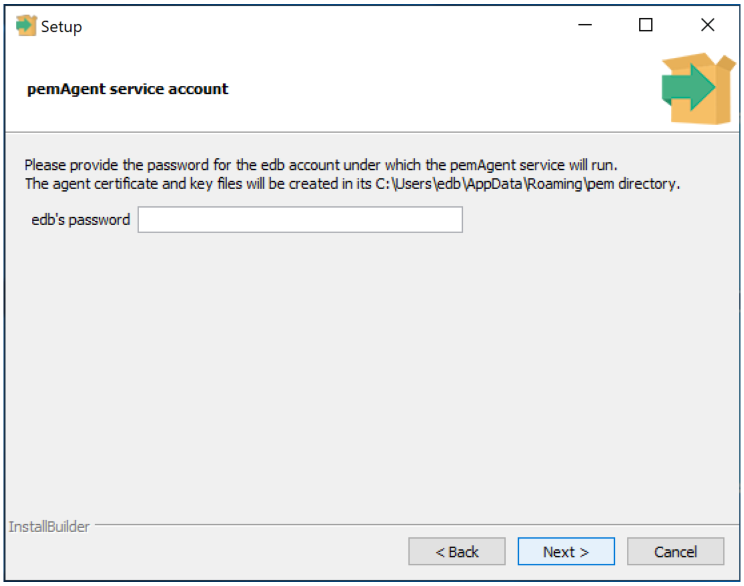
Please specify user and password for the local server - 'PostgreSQL 10 (64 bit)' installation running on port 5433.

User postgres

Password ***

Fig. 3.6: *Enter the name and password of the PostgreSQL superuser*

The information provided on the `Database Server Installation Details` dialog enables the installer to connect to the PostgreSQL server. Provide the `User name` and `Password` of a database superuser. After supplying the requested information, click `Next` to continue to the `pemAgent Service Account` dialog.

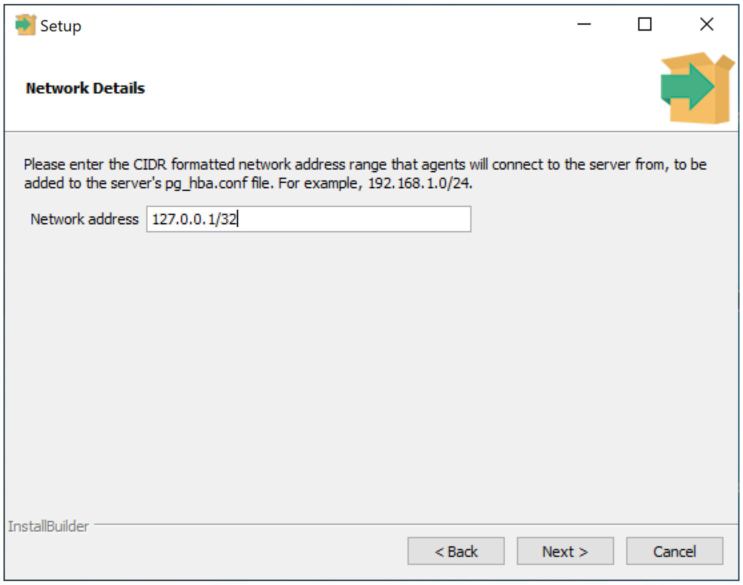


Please provide the password for the edb account under which the pemAgent service will run.
The agent certificate and key files will be created in its C:\Users\edb\AppData\Roaming\pem directory.

edb's password

Fig. 3.7: *pemAgent Service Account Password*

After providing the name and password of the Postgres database superuser, you may be prompted for the password to the user account under which the PEM agent will run. If prompted, provide the password, and press `Next` to continue to the `Network Details` dialog.



Please enter the CIDR formatted network address range that agents will connect to the server from, to be added to the server's pg_hba.conf file. For example, 192.168.1.0/24.

Network address 127.0.0.1/32

Fig. 3.8: *Supply the network address from which the agent will connect*

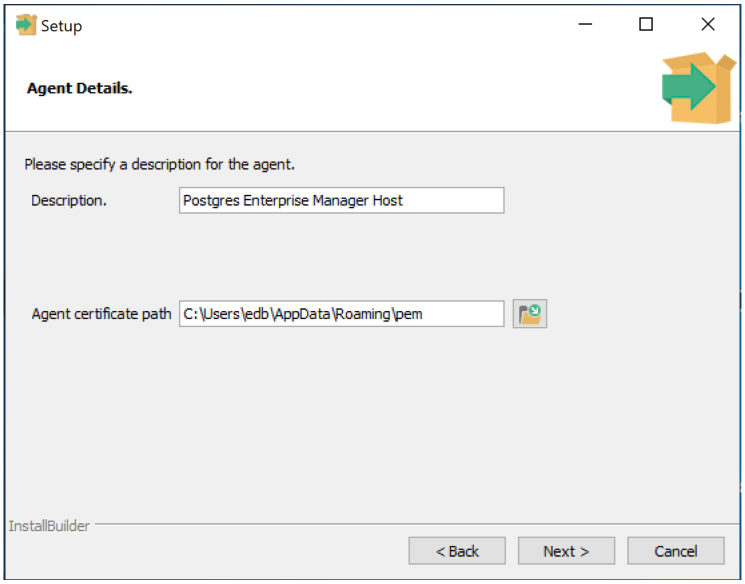
Use the `Network Details` dialog to specify the CIDR-style network address from which the PEM agents will connect to the server (the *client-side* address).

You may specify the address of a network host, or a network address range. For example, if you wish to monitor database servers with the addresses `192.168.10.23`, `192.168.10.76` and `192.168.10.184`, enter `192.168.10.0/24` to allow connections with hosts in that network.

The specified address will be added to the server's `pg_hba.conf` file. You can specify additional network addresses by manually adding entries to the `pg_hba.conf` file on the PostgreSQL server if required, using the initial entry as a template.

When you've added the `Network address`, click `Next` to continue to the `Agent Details` dialog.

The PEM server installer will install a PEM agent on the host on which the server resides, to monitor the server and provide alert processing and garbage collection services. A certificate will also be installed in the location specified in the `Agent certificate path` field.



Agent Details.

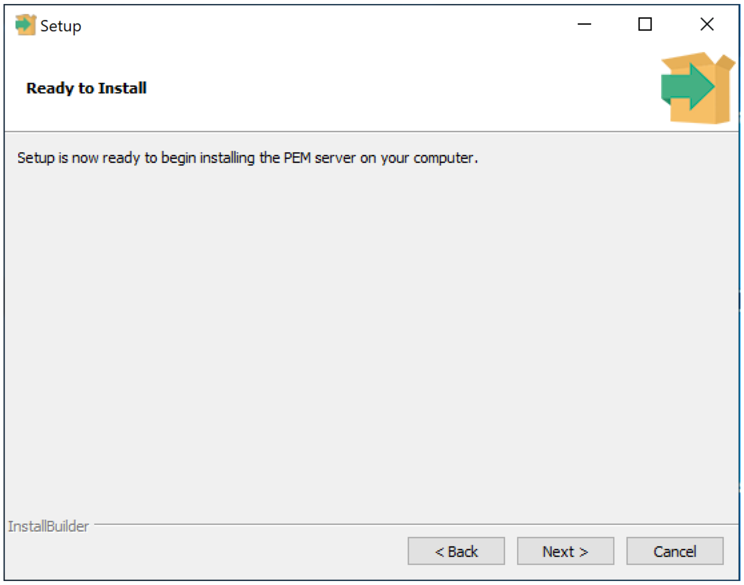
Please specify a description for the agent.

Description. Postgres Enterprise Manager Host

Agent certificate path C:\Users\edb\AppData\Roaming\pem

Fig. 3.9: *Provide a Description for the agent that resides on the server*

Enter an alternate description or select an alternate agent certificate path for the PEM agent, or accept the defaults. Click `Next` to continue to the `Ready to Install` dialog.



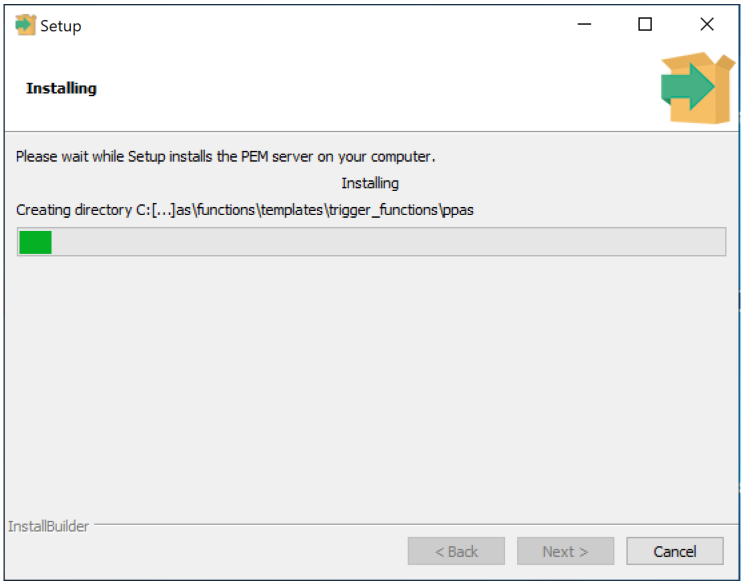
Ready to Install

Setup is now ready to begin installing the PEM server on your computer.

Fig. 3.10: *The installation is ready to begin*

The wizard is now ready to install the PEM server.

Click `Back` to modify any of the options previously selected, or `Next` to continue with the installation.



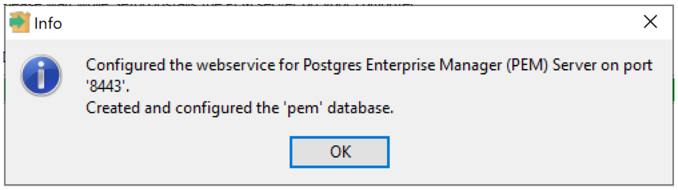
Please wait while Setup installs the PEM server on your computer.

Installing

Creating directory C:[...]as\functions\templates\trigger_functions\ppas

Fig. 3.11: *The installation in progress*

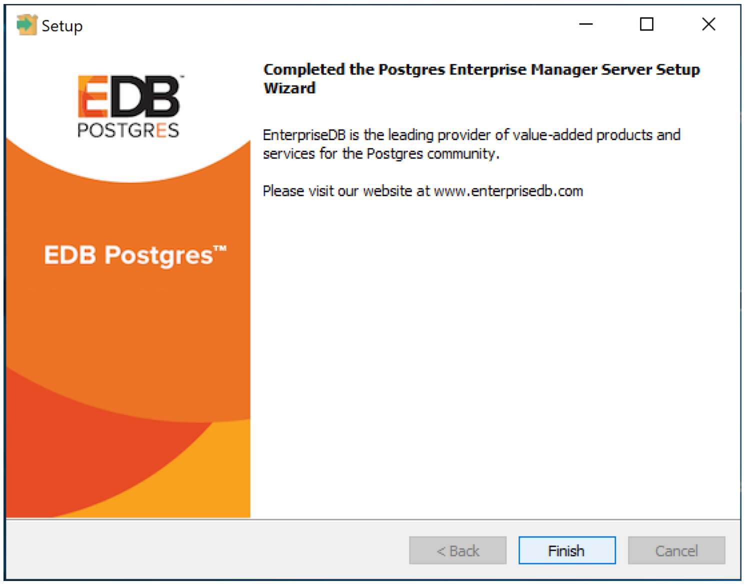
During the installation process, the installer will copy files to the system, and set up the database and web services required to run PEM. When the installation completes, a popup dialog opens confirming that the webservice has been configured, and is listening on port `8443`, and that the pem database has been created and configured.



Configured the webservice for Postgres Enterprise Manager (PEM) Server on port '8443'.
Created and configured the 'pem' database.

Fig. 3.12: *A popup confirms the installation details*

Click `OK` to acknowledge that the webservice has been configured, and that the `pem` database has been created, and continue to the `Completed...` dialog.



Completed the Postgres Enterprise Manager Server Setup Wizard

EnterpriseDB is the leading provider of value-added products and services for the Postgres community.

Please visit our website at www.enterprisedb.com

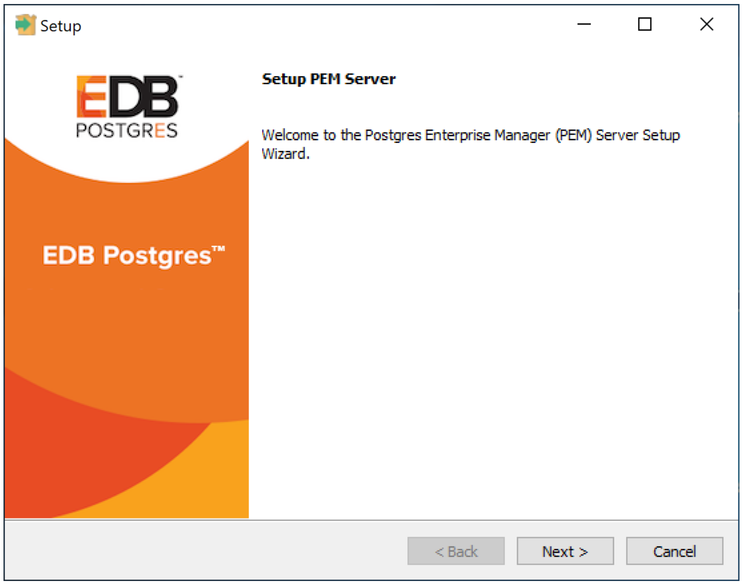
Fig. 3.13: *The PEM server installation is complete*

3.1.2 Installing the PEM Server and PEM-HTTPD on Separate Hosts

To use separate hosts for the PEM server backend database and PEM-HTTPD, you must:

1. Invoke the PEM server installer on the host of the Postgres server that will contain the `pem` database. During the installation, select the `Database` option on the `Advanced options` dialog, and provide connection information for the Postgres server.
2. Modify the `pg_hba.conf` file of the Postgres installation on which the PEM server (and `pem` database) resides, allowing connections from the host of the PEM-HTTPD server.
3. Invoke the PEM server installer on the host of the PEM-HTTPD server, selecting the `Web Services` option on the `Installation Type` dialog.

To invoke the PEM server installer on a Windows system, right click the installer icon and select `Run as Administrator`. The installer displays a `Welcome` dialog.

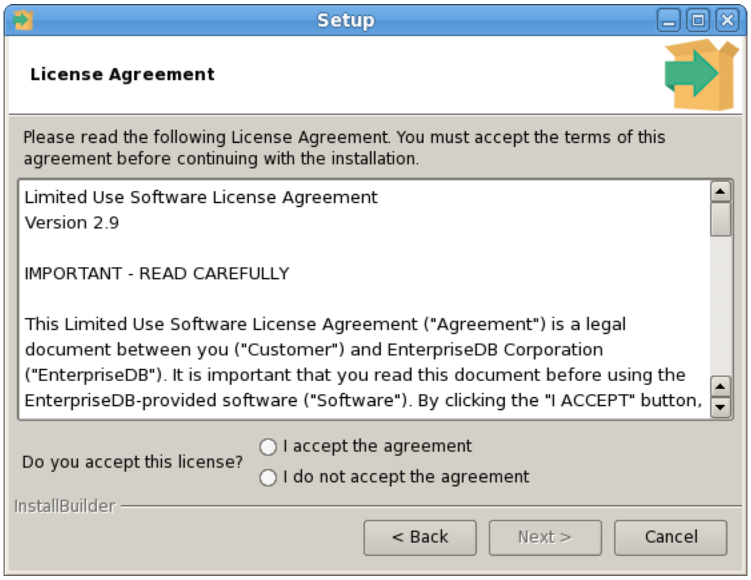


Setup PEM Server

Welcome to the Postgres Enterprise Manager (PEM) Server Setup Wizard.

Fig. 3.14: *The PEM Server Installer's Welcome dialog*

Click `Next` to continue to the `License Agreement` dialog.


Setup

License Agreement

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

Limited Use Software License Agreement
Version 2.9

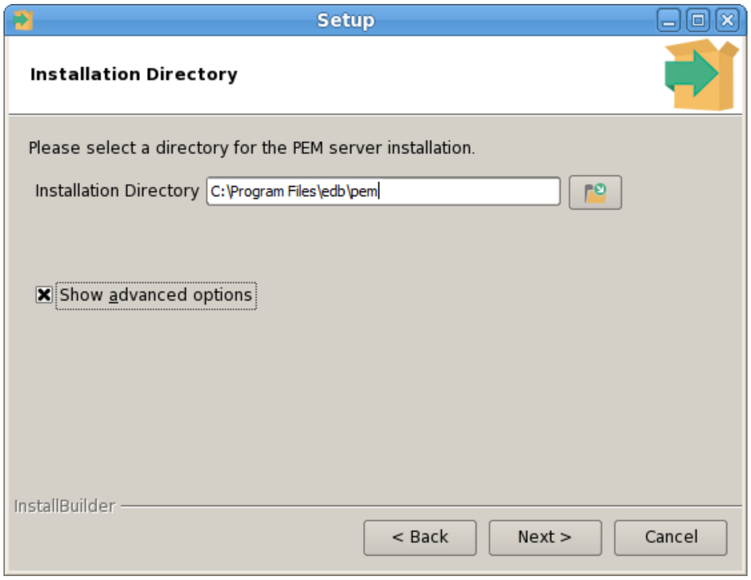
IMPORTANT - READ CAREFULLY

This Limited Use Software License Agreement ("Agreement") is a legal document between you ("Customer") and EnterpriseDB Corporation ("EnterpriseDB"). It is important that you read this document before using the EnterpriseDB-provided software ("Software"). By clicking the "I ACCEPT" button,

Do you accept this license? ○ I accept the agreement ○ I do not accept the agreement

Fig. 3.15: *The License Agreement*

Carefully review the license agreement before highlighting the appropriate radio button and accepting the agreement; click Next to continue.


Setup

Installation Directory

Please select a directory for the PEM server installation.

Installation Directory C:\Program Files\edb\pem

☒ Show advanced options

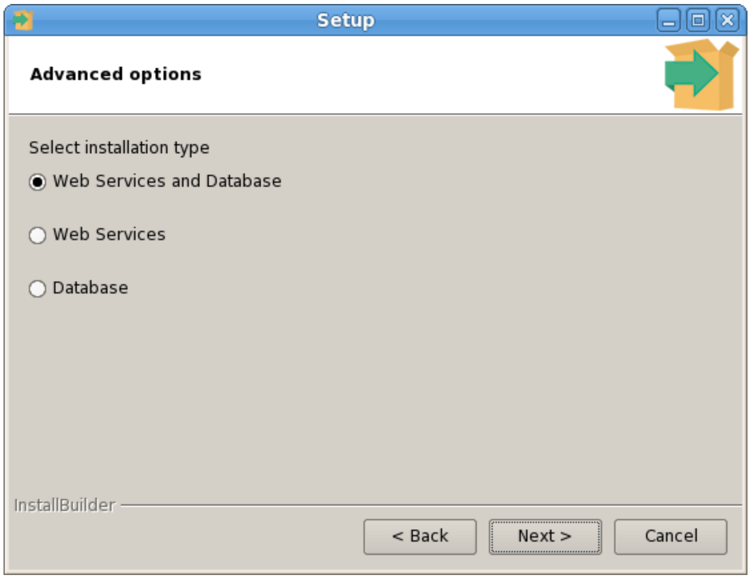
Fig. 3.16: *The Installation Directory dialog*

Use fields on the Installation Directory dialog to specify the directory in which the PEM server will reside, and to access the Advanced options dialog:

- By default, the PEM server is installed in the C:\Program Files\edb\pem on Windows. Accept the default location, or use the Installation Directory field to open a browser dialog and select the directory in which the PEM server will be installed.
- To install the PEM server and PEM-HTTPD on separate hosts, you must use the Advanced options dialog to specify the installation type (Web Services or Database). Select the Show advanced options

check box to instruct the installer to include the Advanced options dialog in the installation process.

Click Next to continue to the Advanced Options dialog.



Advanced options

Select installation type

● Web Services and Database

○ Web Services

○ Database

Fig. 3.17: *The Advanced Options dialog*

Use the radio buttons on the Advanced options dialog to specify the component or components that you would like to install:

- Select Web Services and Database to indicate that the Postgres server and PEM-HTTPD will both reside on the current host. If you select the Web Services and Database option, the PEM server installer will allow you to specify which Postgres server you wish to use for the PEM server before checking for a PEM-HTTPD installation.
- Select Web Services to install PEM-HTTPD on the current host, while using a Postgres database server that resides on another host to host the PEM server and pem database.

Note: You must complete the PEM server installation process on the host of the PEM server (and pem backend database), selecting Database on the Advanced options dialog, and modifying the connection properties of the pg_hba.conf file on the PEM server before using the Web Services option to install PEM-HTTPD.

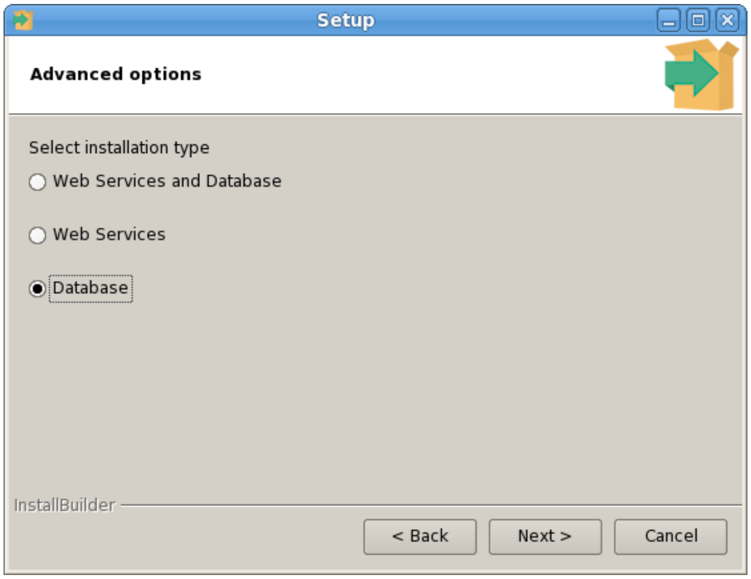
This option invokes the installation steps documented in *Installing Web Services* section.

- Select Database to use an existing Postgres server (version 9.4 or greater), or to install only the database server that is distributed with the PEM server installer. This option invokes the installation steps documented in *Specifying a Database Host*.

After selecting an installation option, click Next to continue.

3.1.2.1 Specifying a Database Host

Selecting the Database option on the Advanced options dialog allows you to specify connection information for the host on which the PEM server backend database (named pem) will reside.



Select installation type

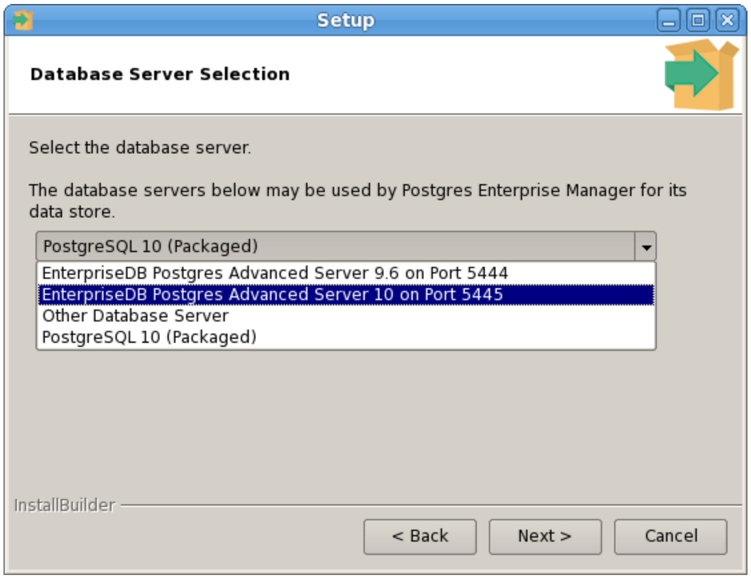
○ Web Services and Database

○ Web Services

● Database

Fig. 3.18: *The Advanced options dialog*

Click Next to continue to the Database Server Selection dialog.



Select the database server.

The database servers below may be used by Postgres Enterprise Manager for its data store.

PostgreSQL 10 (Packaged)

EnterpriseDB Postgres Advanced Server 9.6 on Port 5444
EnterpriseDB Postgres Advanced Server 10 on Port 5445
Other Database Server
PostgreSQL 10 (Packaged)

Fig. 3.19: *Selecting a database server*

Use the drop-down listbox on the Database Server Selection dialog to select a host for the PEM server backend database. You can:

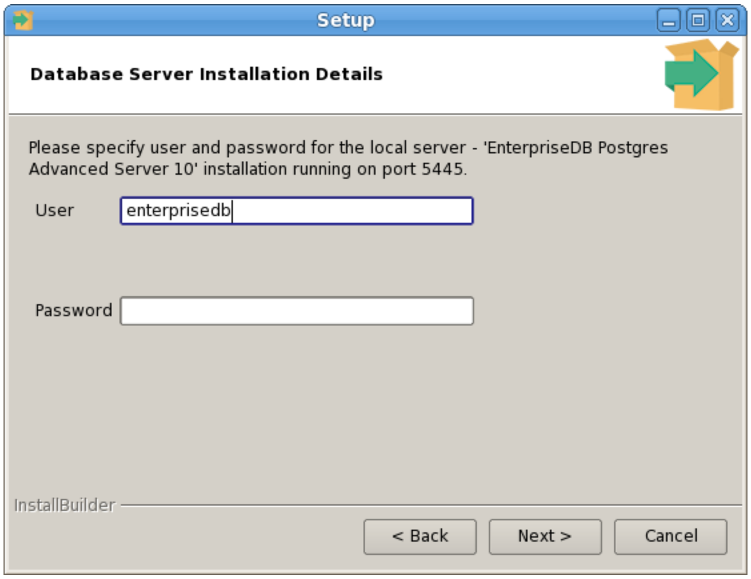
- Select a host from existing Postgres installations that reside on the current host.

Note: You may be required to add the sslutils package to your installation.

- Select the PostgreSQL x option to install the Postgres server that is distributed with the PEM server installer where x is the PostgreSQL database server version. If you decide to use the version of PostgreSQL that is bundled with the PEM server installer, the EnterpriseDB one-click PostgreSQL installer will open and walk you through the installation.
- Select Other Database Server to specify connection information for a Postgres server that was not installed using a one-click graphical installer from EnterpriseDB. For information about the software pre-requisites for the PEM server database host, please see *Preparing the Postgres Server* section.

Click Next to continue.

If the PEM server will reside on an existing Postgres server, the Database Server Installation Details dialog shown in opens.



Database Server Installation Details

Please specify user and password for the local server - 'EnterpriseDB Postgres Advanced Server 10' installation running on port 5445.

User enterprisedb

Password

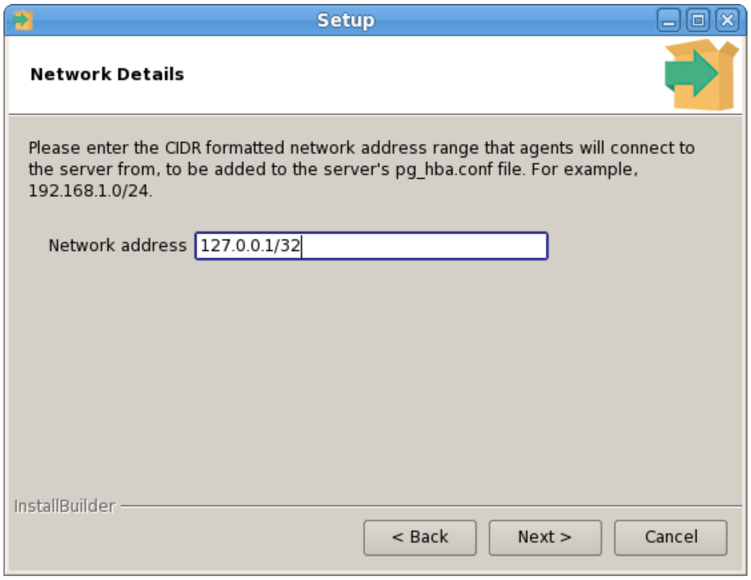
Fig. 3.20: *The Database Server Installation Details dialog*

The information required on the Database Server Installation Details dialog may vary; the PEM server installer will ask you to provide only the information about the selected installation that it cannot locate:

- Specify the name of a Postgres database superuser in the User field.
- Specify the password associated with that user in the Password field.

Click Next to continue.

If prompted, provide the system password for the service account under which the PEM agent will run, and click Next to continue to the Network Details dialog.



Please enter the CIDR formatted network address range that agents will connect to the server from, to be added to the server's pg_hba.conf file. For example, 192.168.1.0/24.

Network address 127.0.0.1/32

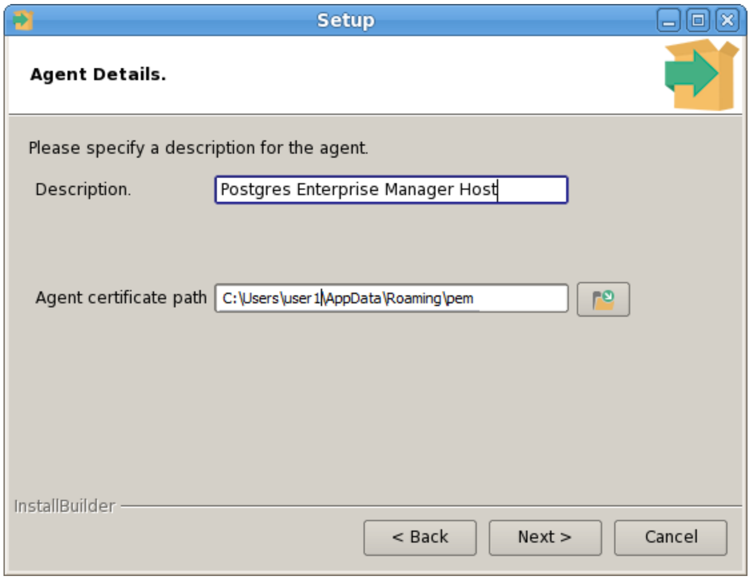
Fig. 3.21: *Supply the network address from which the agent will connect*

Use the `Network Details` dialog to specify the CIDR-style network address from which PEM agents will connect to the server (the *client-side* address). The specified address will be added to the server's `pg_hba.conf` file.

Note: You can specify additional network addresses by manually adding entries to the `pg_hba.conf` file on the PostgreSQL server.

Accept the default (specifying the localhost), or specify a `Network address` range, and click `Next` to continue to the `Agent Details` dialog.

The PEM server installer will install a PEM agent on the host on which the server resides, to monitor the server and provide alert processing and garbage collection services. A certificate will also be installed in the location specified in the `Agent certificate path` field.


Setup

Agent Details.

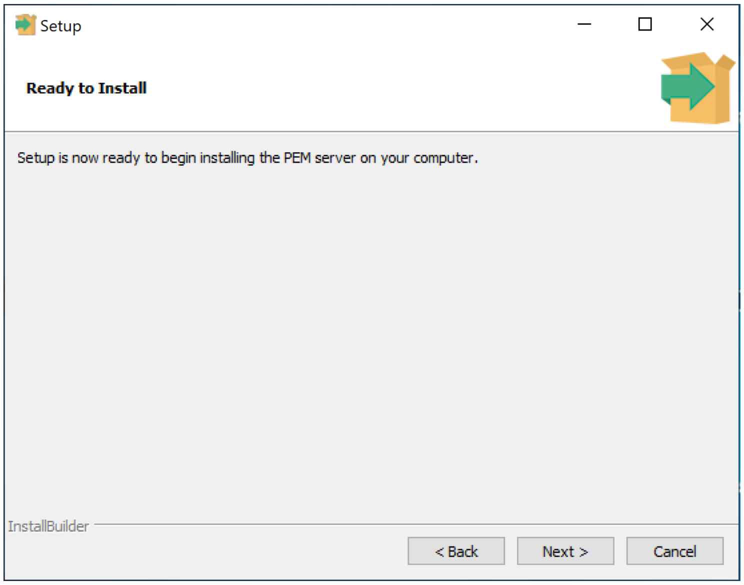
Please specify a description for the agent.

Description. Postgres Enterprise Manager Host

Agent certificate path C:\Users\user 1\AppData\Roaming\pem

Fig. 3.22: *Provide a Description for the agent*

You can enter an alternate description or an alternate agent certificate path for the PEM agent, or accept the defaults. Click `Next` to continue.

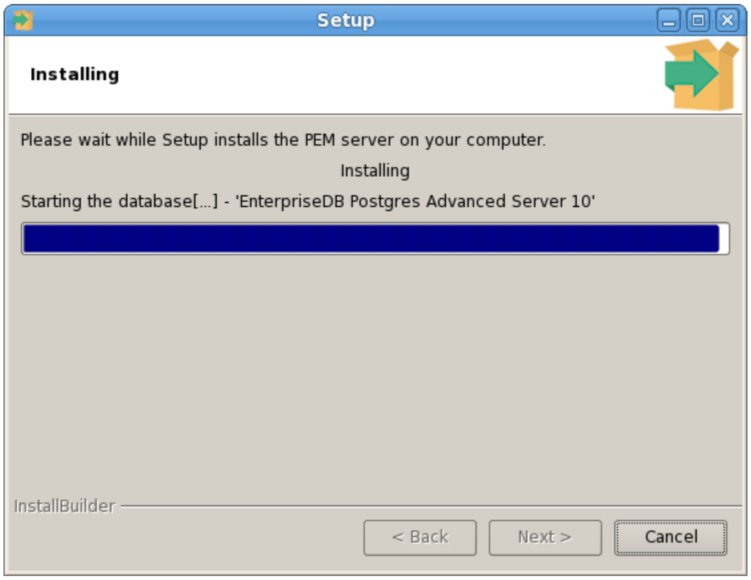

Setup

Ready to Install

Setup is now ready to begin installing the PEM server on your computer.

Fig. 3.23: *The installation is ready to begin*

The wizard is now ready to install the PEM server. Click `Back` to modify any of the options previously selected, or `Next` to proceed with the installation.



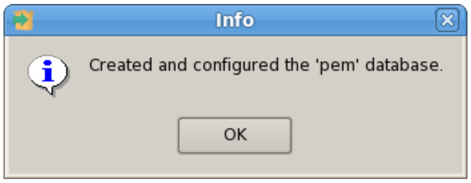
Please wait while Setup installs the PEM server on your computer.

Installing

Starting the database[...] - 'EnterpriseDB Postgres Advanced Server 10'

Fig. 3.24: *The installation in progress*

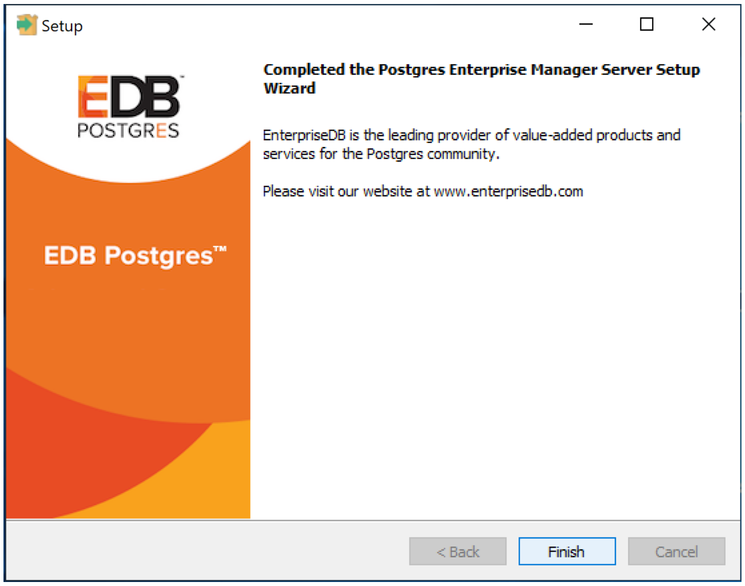
During the installation process, the installer will copy files to the system, and set up the PEM server's backend database. A popup dialog opens confirming that the `pem` database has been created and configured.



Created and configured the 'pem' database.

Fig. 3.25: *Click OK to confirm that the pem database has been created*

Click `OK` to acknowledge that the `pem` database has been created, and continue to the `Completed...` dialog.



Completed the Postgres Enterprise Manager Server Setup Wizard

EnterpriseDB is the leading provider of value-added products and services for the Postgres community.

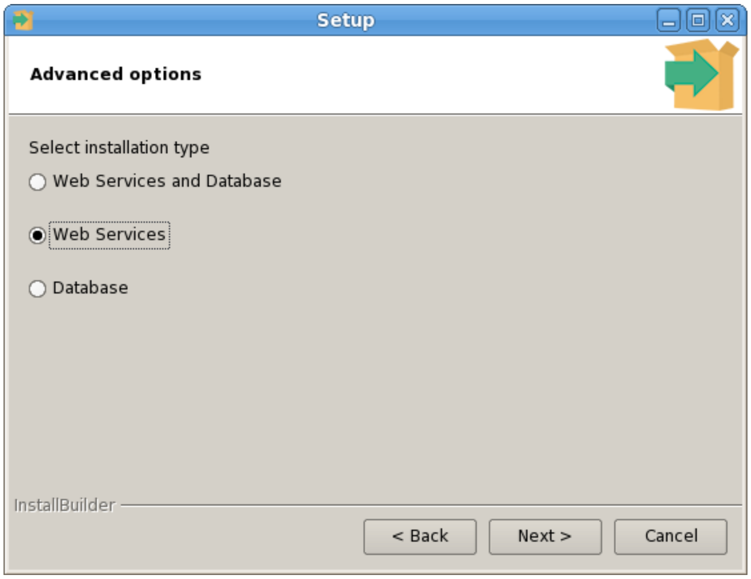
Please visit our website at www.enterprisedb.com

Fig. 3.26: *The PEM server installation is complete*

When the database portion of the PEM server installation is complete, you can invoke the PEM server on another host to install (or upgrade) PEM-HTTPD.

3.1.2.2 Installing Web Services

Selecting the `Web Services` radio button on the `Advanced options` dialog instructs the PEM server installer to either install PEM-HTTPD on the current host or update an existing PEM-HTTPD installation.



Advanced options

Select installation type

○ Web Services and Database

● Web Services

○ Database

Fig. 3.27: *Selecting the Web Services option on the Advanced options dialog*

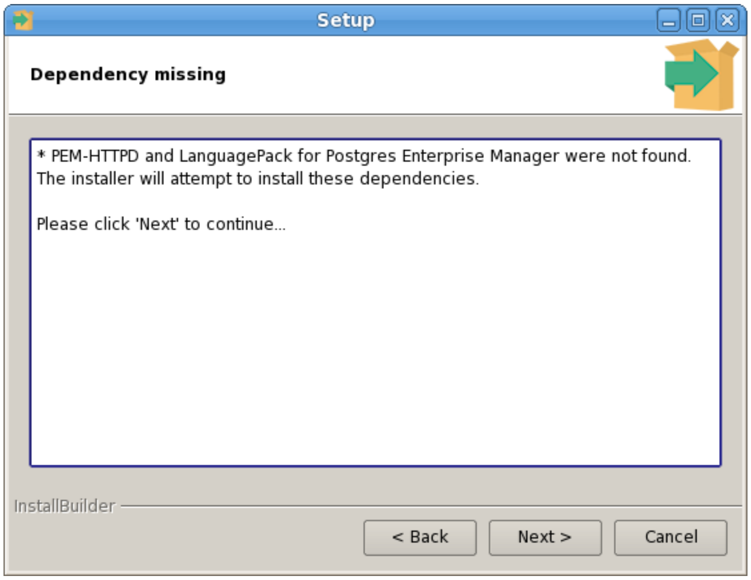
Note: The current host may not be the host of the PEM backing database.

Before selecting this option, you must have:

- Completed an installation of the PEM server installer on a host system, during which you specified a backing database for the PEM server.
- Modified the `pg_hba.conf` file on the PEM server database host to allow connections from the PEM-HTTPD host, and restarted the database server.

When you select the `Web Services` option and click `Next`, the PEM server installer will check the current host for existing PEM-HTTPD and LanguagePack installations.

If the installer does not locate the components, the installer will inform you that one or more dependencies are missing.

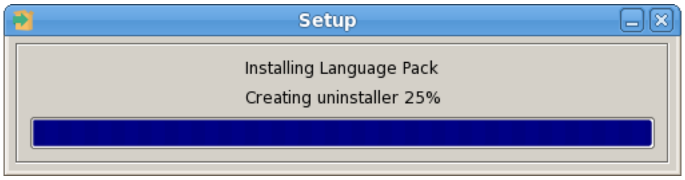


* PEM-HTTPD and LanguagePack for Postgres Enterprise Manager were not found.
The installer will attempt to install these dependencies.

Please click 'Next' to continue...

Fig. 3.28: *The installer has detected missing dependencies*

Click `Next` to instruct the server to install LanguagePack



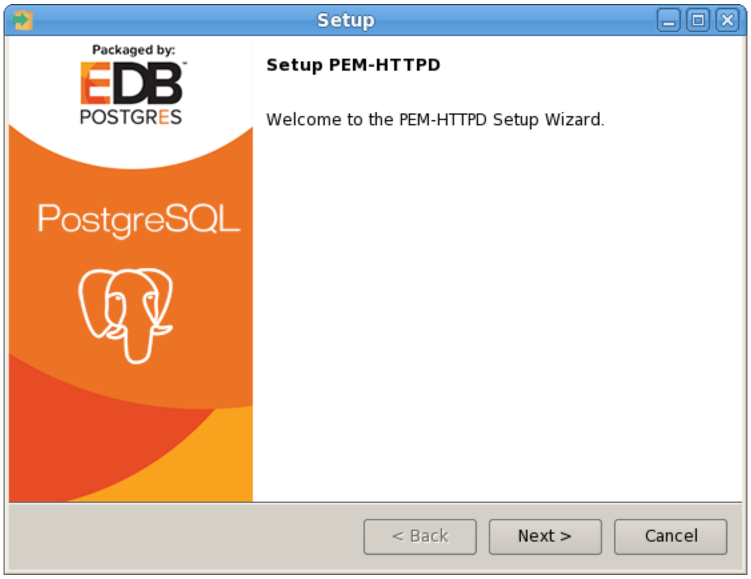
Setup

Installing Language Pack

Creating uninstaller 25%

Fig. 3.29: *The installer has detected missing dependencies*

After installing language pack, the installer will invoke the PEM-HTTPD setup wizard.

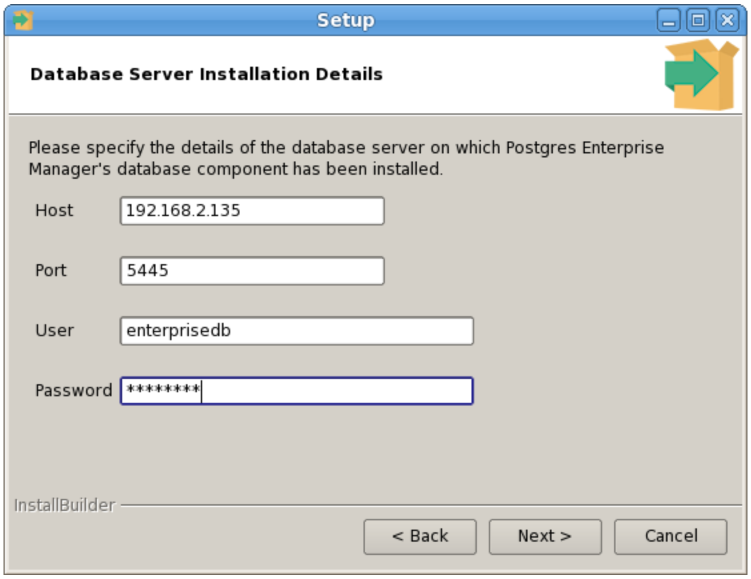


Setup PEM-HTTPD

Welcome to the PEM-HTTPD Setup Wizard.

Fig. 3.30: *The PEM-HTTPD installation wizard*

Follow the onscreen instructions of the PEM-HTTPD Setup Wizard. When the wizard completes the installation, click Finish to open the Database Server Installation Details dialog.



Please specify the details of the database server on which Postgres Enterprise Manager's database component has been installed.

Host 192.168.2.135

Port 5445

User enterprisedb

Password ********

Fig. 3.31: *The Database Server Installation Details dialog*

Use the fields on the `Database Server Installation Details` dialog to provide connection information for the Postgres installation that is hosting the PEM server installation:

- Enter the name or IP address of the PEM server host in the `Host` field.
- Enter the port number on which the Postgres server is listening in the `Port` field.
- Enter the name of a Postgres database superuser in the `User` field.
- Enter the password associated with the Postgres superuser in the `Password` field.

Click `Next` to continue. Before completing the PEM server installation, the installer will contact the database host. The `pg_hba.conf` file on the PEM database host must be configured to accept connections from the host of the httpd server and the firewall must allow a connection for the installation to continue. The PEM server installer will complete the PEM server installation, adding only those items that must reside on the host of the PEM-HTTPD server.

3.1.3 Installing the PEM Server on an Existing Postgres Server

You can use an existing Postgres server (version 9.4 or later) to host the PEM server and the `pem` database. Postgres installers and pre-requisite software extensions are freely available on the EnterpriseDB website at

This section provides information about configuring an existing Postgres server for a PEM server installation.

Note: The steps that follow should be considered guidelines only; the actual steps required to configure your Postgres installation will vary depending on the configuration of your Postgres server.

The following versions of Postgres are pre-configured to contain the `sslutils` extension and a service script; no additional preparation is required to use the following Postgres versions as a PEM backend database server:

- PostgreSQL 9.4 or later (as bundled with the PEM Server installer)
- Advanced Server 9.4 or later

3.1.3.1 Preparing the Postgres Server

Before installing the PEM server on an existing Postgres server, you must:

- Ensure that the Postgres server contains an installation of the `sslutils` extension. For more information, see *Installing the sslutils Extension* section.
- Create a service script (on Linux) or register the server with the Windows service manager. For more information, see *Creating a Service Script or Registering the Service* section.

After preparing the server, you can use the PEM server installer to install PEM on the existing Postgres server.

3.1.3.1.1 Installing the sslutils Extension

The Postgres server on which the PEM server will reside must contain the `sslutils` extension. The `sslutils` package is freely available for download from the EnterpriseDB website

When the web page opens, select the link for the `SRC- SSL Utils 1.2` package. When the download completes, extract the file, and copy it into the Postgres installation directory.

On Linux

If the Postgres server resides on a Linux system, you must use the gcc compiler to build sslutils.

1. Use yum to install gcc:

   ```
   yum install gcc
   ```

2. Set the value of `PATH` so it can locate the `pg_config` program:

   `export PATH=$PATH:/usr/pgsql-x/bin` where `x` is the version of PostgreSQL database server

3. Then, use `yum` to install the `sslutil` dependencies:

   ```
   yum install openssl-devel
   ```

4. Move into the `sslutils` folder, and enter:

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

5. Use psql to create the `sslutils` extension:

```
CREATE EXTENSION sslutils
```

On Windows

Remember: You are *not* required to manually add the sslutils extension when using the following Postgres installations:

- PostgreSQL 9.4 or later (as distributed with the PEM server installer)
- Advanced Server 9.4 or later

sslutils must be built with the same compiler that was used to compile the backend Postgres installation. If you are using a backend Postgres database that was installed on a Windows platform using a PostgreSQL one-click installer (from EnterpriseDB) or an Advanced Server installer, you must use Visual Studio to build sslutils.

While specific details of the installation process will vary by platform and compiler, the basic steps are the same. You must:

1. Copy the sslutils package to the Postgres installation directory.
2. Open the command line of the appropriate compiler, and navigate into the sslutils directory.
3. Use the following commands to build sslutils:

   ```
   SET USE_PGXS=1
   SET GETTEXTPATH=<path_to_gettext>
   SET OPENSSLPATH=<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
   ```

   Where:

   path_to_gettext specifies the location of the GETTEXT library and header files.

   path_to_openssl specifies the location of the openssl library and header files.

   path_to_pg_installation_dir specifies the location of the Postgres installation.

4. Copy the compiled sslutils files to the appropriate directory for your installation. The sslutils directory will contain the following files:

   ```
   sslutils--1.1.sql
   sslutils--unpackaged--1.1.sql
   sslutils--pemagent.sql.in
   sslutils.dll
   ```

   Copy the .dll libraries and .sql files into place:

   ```
   COPY sslutils*.sql* "%PGPATH%\share\extension\"
   COPY sslutils.dll "%PGPATH%\lib\"
   ```

3.1.3.1.2 Creating a Service Script or Registering the Service

A service script allows the PEM server to start, stop or restart the server if necessary when performing configuration management, certificate management, or other administrative tasks.

When you install a PostgreSQL or an Advanced Server database using an installer from EnterpriseDB (such as the PostgreSQL one-click installer), the installer will create a service script, or on Windows, register the service for you. If you have built the Postgres installation from source, you are required to manually create a service script.

While the PEM server installer checks for the presence of the service script, it does not check the integrity of the script itself; for PEM to function properly, you must ensure that the service script works as expected.

Writing a Linux Service Script

On Linux, the service script must reside in the `/etc/init.d` directory. The service script must be able to start, stop and restart the database server. Service scripts are platform-specific; you can find a sample service script for Linux in *Linux Service Script (Sample)* section. For information about customizing a Postgres service, visit:

> https://www.postgresql.org/docs/current/static/server-start.html

Registering a Service on Windows

If you are using Windows to host the PEM backend database, you must register the name of the Postgres server with the Windows service manager. If you are using a Postgres server that was created using an EnterpriseDB installer, the service will be registered automatically. If you are manually building the installation, you can use the `register` clause of the Postgres `pg_ctl` command to register the service. The syntax of the command is:

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

Where:

`service name` specifies the name of the Postgres cluster.

`user_name` specifies the name of an operating system user with sufficient privileges to access the Postgres installation directory and start the Postgres service.

`password` specifies the operating system password associated with the user.

`data_directory` specifies the location of the Postgres data directory.

For more information about using the `pg_ctl` command and the available command options, see the

> Postgres core documentation

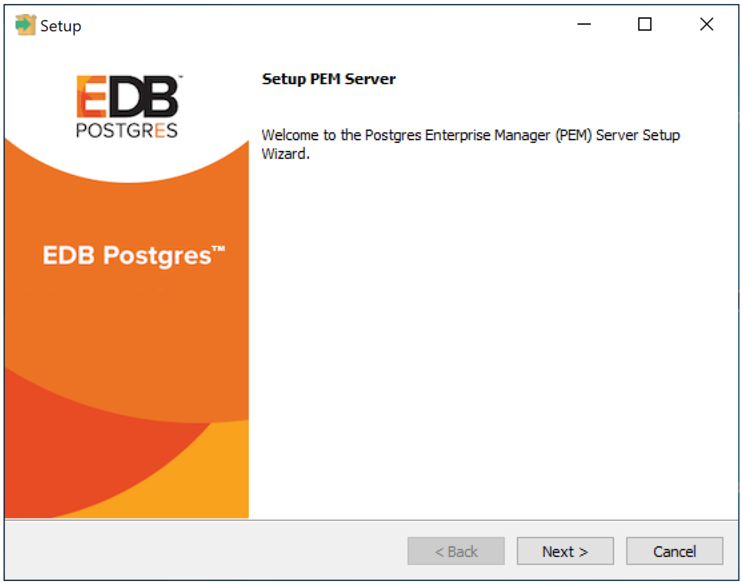
3.1.3.2 Invoking the PEM Server Installer

After preparing the existing Postgres server, invoke the PEM server installer. Assume superuser (or, on Windows, Administrative) privileges and navigate into the directory that contains the installer. Then, invoke the installer with the command:

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

Where *x* is the major and minor versions of PEM and platform is the platform.

The installer displays a `Welcome` dialog.

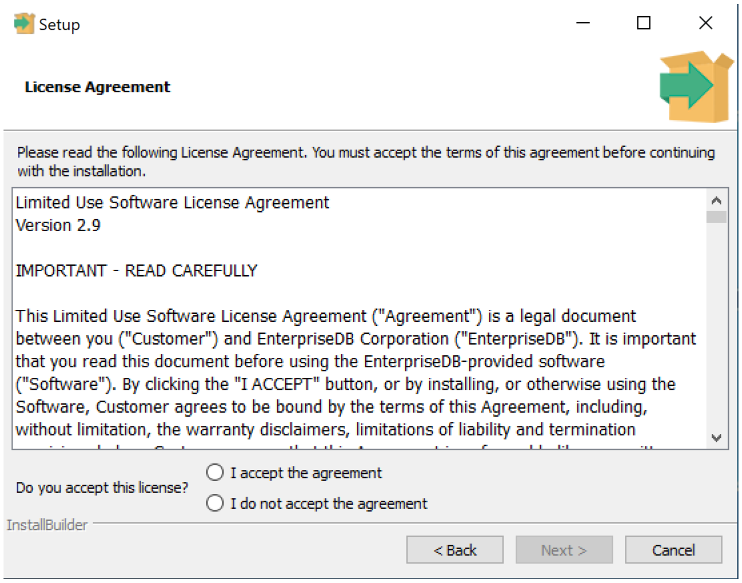


Setup PEM Server

Welcome to the Postgres Enterprise Manager (PEM) Server Setup Wizard.

Fig. 3.32: *The PEM server Installer's Welcome Dialog*

Click `Next` to continue to the `License Agreement` dialog.



License Agreement

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

Limited Use Software License Agreement
Version 2.9

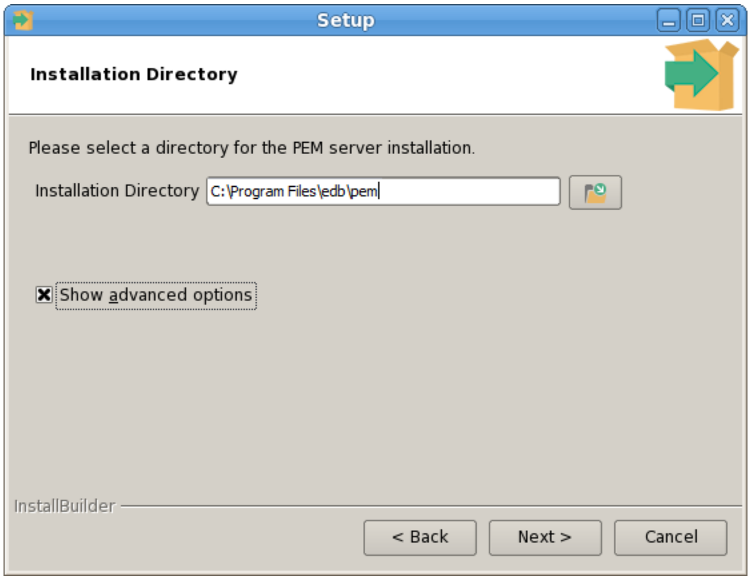
IMPORTANT - READ CAREFULLY

This Limited Use Software License Agreement ("Agreement") is a legal document between you ("Customer") and EnterpriseDB Corporation ("EnterpriseDB"). It is important that you read this document before using the EnterpriseDB-provided software ("Software"). By clicking the "I ACCEPT" button, or by installing, or otherwise using the Software, Customer agrees to be bound by the terms of this Agreement, including, without limitation, the warranty disclaimers, limitations of liability and termination

Do you accept this license?
○ I accept the agreement
○ I do not accept the agreement

Fig. 3.33: *The License Agreement*

Carefully review the license agreement before highlighting the appropriate radio button and accepting the agreement; click `Next` to continue to the `Installation Directory` dialog.



Installation Directory

Please select a directory for the PEM server installation.

Installation Directory C:\Program Files\edb\pem

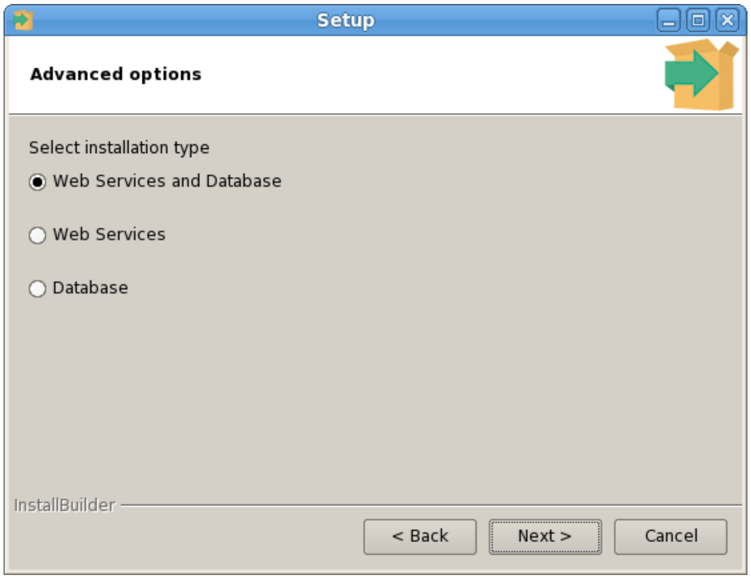
☒ Show advanced options

Fig. 3.34: *The Installation Directory dialog*

Use the Installation Directory dialog to specify the location of the PEM server and access the Advanced options dialog:

- Use the Installation Directory field to open a browser dialog and select the directory in which the PEM server will be installed.
- If you are installing the PEM server on an existing server, check the box next to Show advanced options to instruct the installer to include the Advanced options dialog in the installation process.

Click Next to continue.



Advanced options

Select installation type

● Web Services and Database

○ Web Services

○ Database

Fig. 3.35: *The Advanced Options dialog*

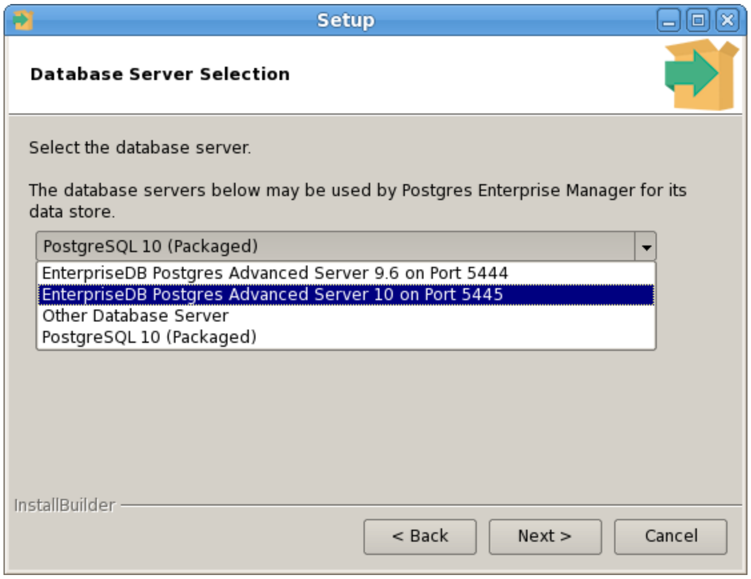
Use the radio buttons on the Advanced options dialog to specify an installation type. Select:

- Web Services and Database if both the Postgres server and the PEM-HTTPD server will reside on the current host. This option is valid if you are using an existing Postgres server to host the PEM server, or using the PEM server installer to install the Postgres server on which the PEM server will reside.

  If you select Web Services and Database, the PEM server installer will check the current host for a PEM-HTTPD installation, and upgrade or install PEM-HTTPD if necessary.

- Web Services if only the PEM-HTTPD server will reside on the current host. See *Installing Web Services* section for more information about invoking this option.
- Database if you are installing only the PEM server (and creating the pem backend database) on the current host. This option is valid if you are using an existing Postgres server to host the PEM server, or using the PEM server installer to install the PostgreSQL server on which PEM will reside.

After selecting an installation option, click Next to continue.



Database Server Selection

Select the database server.

The database servers below may be used by Postgres Enterprise Manager for its data store.

PostgreSQL 10 (Packaged)

Setup
EnterpriseDB Postgres Advanced Server 9.6 on Port 5444
EnterpriseDB Postgres Advanced Server 10 on Port 5445
Other Database Server
PostgreSQL 10 (Packaged)
InstallBuilder
< Back
Next >
Cancel

Fig. 3.36: *The Database Server Selection dialog*

Use the drop-down listbox on the Database Server Selection dialog to select a backend database for the PEM server:

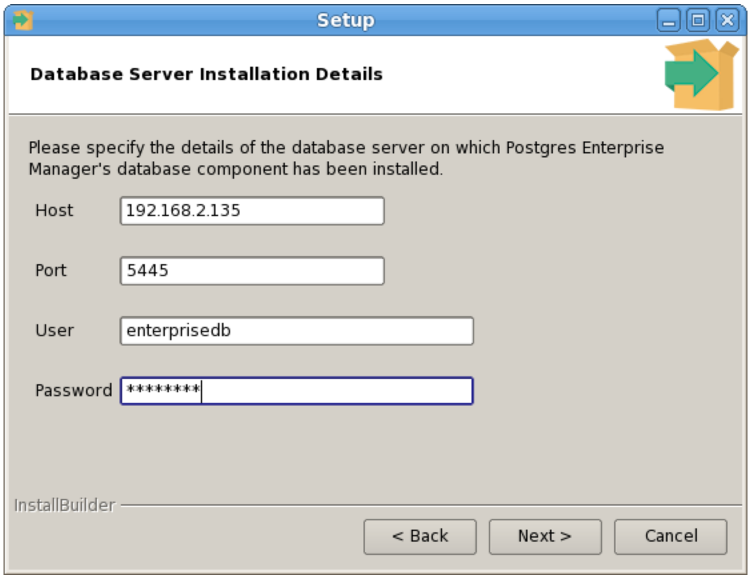
- Select the name of a Postgres server on the current host that was installed using a Postgres one-click installer or Advanced Server installer.
- Select the PostgreSQL x (Packaged) option to instruct the installation wizard to install and use the PostgreSQL server that is packaged with the PEM server installer. Where x is the version of the PostgreSQL database server.
- Select Other Database Server to instruct the PEM server installer to use a Postgres database that was installed from a source other than an EnterpriseDB installer (i.e. from an rpm, or built from source).

Note: The selected database server must include an installation of the sslutils contrib module, and have a startup script (on Linux) or a registered service (on Windows).

For information about Preparing the Postgres Server, please see *this section*.

If you selected Web Services and Database on the Advanced options dialog, the installation wizard will check the current host for an existing PEM-HTTPD installation, and upgrade or install the service as needed.

If you selected Database on the Advanced options dialog, the Database Server Installation Details dialog opens.



Please specify the details of the database server on which Postgres Enterprise Manager's database component has been installed.

Host 192.168.2.135

Port 5445

User enterprisedb

Password ********

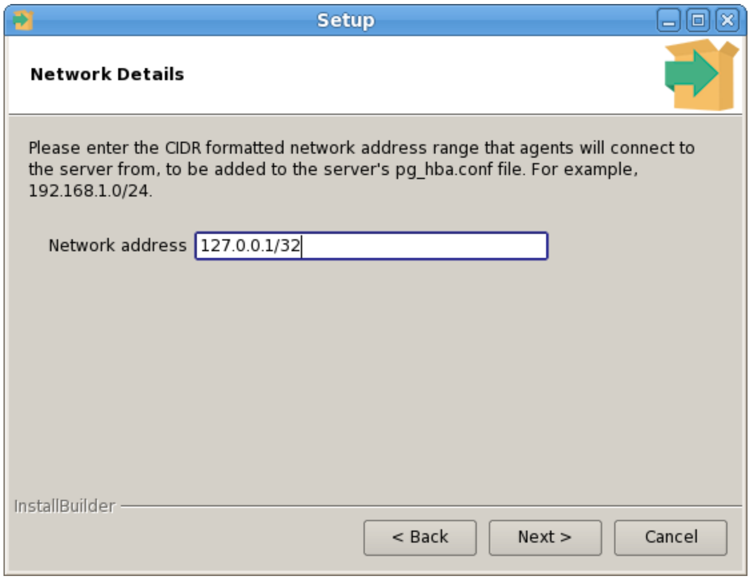
Fig. 3.37: *The Database Server Installation Details dialog*

Use the fields on the Database Server Installation Details dialog to describe the connection to the Postgres server that will host the PEM server:

- Enter the port number on which the Postgres server listens in the Port field.
- Enter the name of a database superuser in the User field.
- Enter the password associated with the superuser in the Password field.
- Enter the name of the service script for the Postgres server in the Service Name field.

Note: For information about preparing a service script, please see *this section*

Click Next to continue.



Please enter the CIDR formatted network address range that agents will connect to the server from, to be added to the server's pg_hba.conf file. For example, 192.168.1.0/24.

Network address 127.0.0.1/32

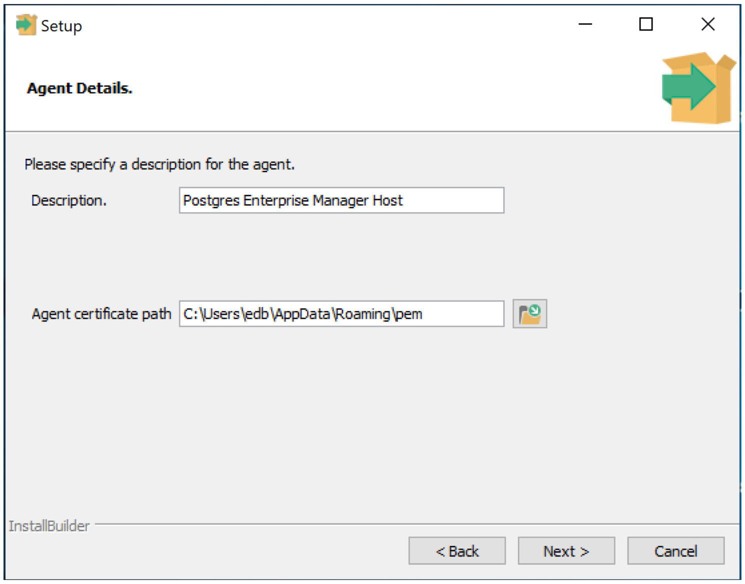
Fig. 3.38: *Supply the network address from which the agent will connect*

Use the `Network Details` dialog to specify the CIDR-style network address from which the PEM agents will connect to the server (the `client-side` address). The specified address will be added to the server's `pg_hba.conf file.`

You can specify additional network addresses by manually adding entries to the `pg_hba.conf` file on the PostgreSQL server if required, using the initial entry as a template.

When you've added the `Network address`, click `Next` to continue to the `Agent Details` dialog.

The PEM server installer will install a PEM agent to the host on which the server resides, to monitor the server and provide alert processing and garbage collection services. A certificate will also be installed in the location specified in the `Agent certificate` path field.



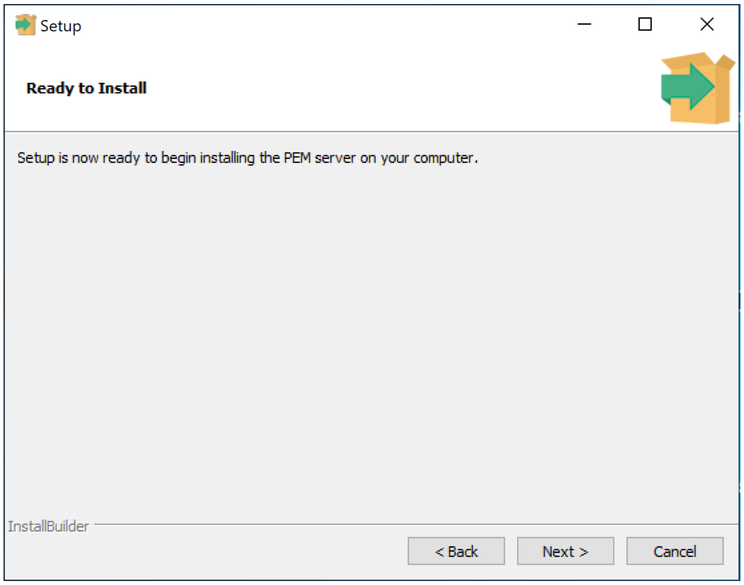
Please specify a description for the agent.

Description. Postgres Enterprise Manager Host

Agent certificate path C:\Users\edb\AppData\Roaming\pem

Fig. 3.39: *Provide a Description for the agent that resides*

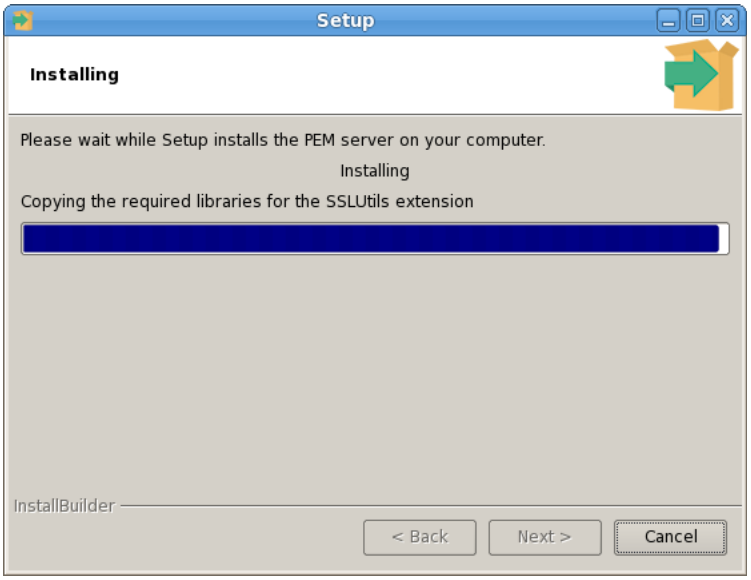
You can enter an alternate description or an alternate agent certificate path for the PEM agent, or accept the defaults. Click `Next` to continue to the `Ready to Install` dialog.



Setup is now ready to begin installing the PEM server on your computer.

Fig. 3.40: *The installation is ready to begin*

The wizard is now ready to install the PEM server. Click `Back` to modify any of the options previously selected, or `Next` to continue with the installation.



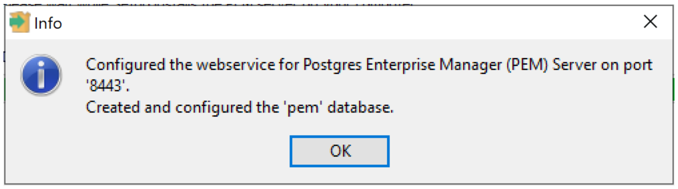
Please wait while Setup installs the PEM server on your computer.

Installing

Copying the required libraries for the SSLUtils extension

Fig. 3.41: *The installation in progress*

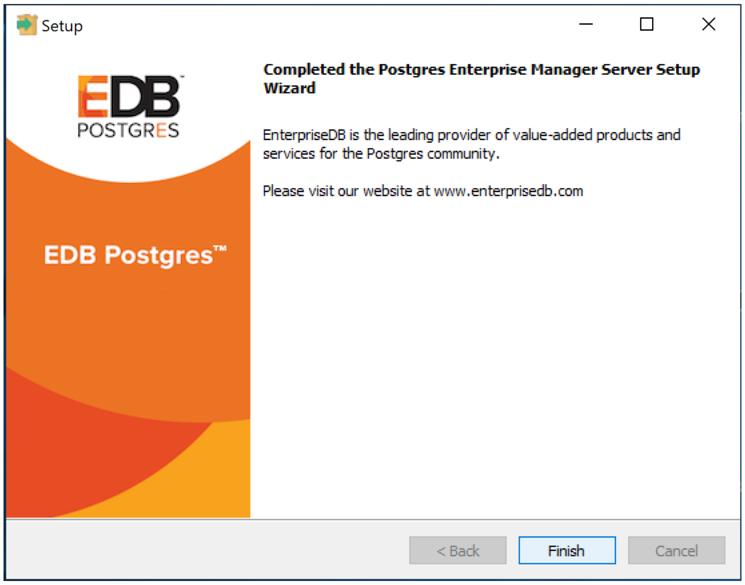
During the installation process, the installer will copy files to the system, and set up the PEM server's backend database. A popup dialog opens confirming that the `pem` database has been created and configured.



Configured the webservice for Postgres Enterprise Manager (PEM) Server on port '8443'.
Created and configured the 'pem' database.

Fig. 3.42: *Click OK to confirm that the pem database has been created*

Click `OK` to acknowledge that the pem database has been created, and continue to the `Completed...` dialog.



Completed the Postgres Enterprise Manager Server Setup Wizard

EnterpriseDB is the leading provider of value-added products and services for the Postgres community.

Please visit our website at www.enterprisedb.com

Fig. 3.43: *The PEM server installation is complete*

If you are using a PEM-HTTPD service that resides on a separate host, you must:

- Modify the `pg_hba.conf` file on the Postgres server host to allow connections between the hosts.
- Invoke the PEM server installer on the host of the PEM-HTTPD server. See *Installing Web Services* section for more information about installing PEM-HTTPD.

3.1.4 Invoking the Server Installer from Command Line

The command line options of the PEM server and PEM agent installers offer functionality in situations where a graphical installation may not work because of limited resources or system configuration. You can:

- Include the `--mode unattended` option when invoking the installer to perform an installation without additional user input.

Not all command line options are suitable for all platforms. For a complete reference guide to the command line options, include the `--help` option when you invoke the installer.

3.1.4.1 Invoking the PEM Server Installer in Unattended Mode

You can perform an unattended PEM server installation by providing installation preferences on the command line when invoking the installer. Please note that the system on which you are installing the PEM server must have internet access.

You must have Administrative privileges to install the PEM server. Before invoking the PEM server installer, you must install the following dependencies:

- PostgreSQL
- pem-httpd
- Language Pack

You can use the PEM server installer to satisfy the dependencies of the PEM server; use the following command to extract the dependencies. Navigate to the location of the installer, and use the following command to extract the dependencies:

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

In our example, the files are extracted to the `C:\` directory. After extracting the files, you must install each program. Navigate into the directory that contains the files (in our example, `C:\`), and enter:

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

postgresql-<version>-windows-x64.exe –mode unattended

Then, you can invoke the PEM server installer:

```
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
```

Where:

`registered_edb_user` specifies the name of a registered EnterpriseDB user. To register, visit the EDB website

`edb_user_password` specifies the password associated with the EDB user account.

`port` specifies the port used by the backing PostgreSQL database; by default, the PostgreSQL database uses port `5432`.

`cidr_address_range` specifies the address range that will be added to the `pg_hba.conf` file of the PEM server's backing database to allow connections from the agents that will be monitored by the server. You may wish to specify a network range (for example, 192.168.2.0/24) to provide server access to agents that reside on the same network.

`windows_password` specifies the password associated with the Windows Administrator's account.

Note: when invoked in unattended mode, the PostgreSQL installer creates a user named `postgres`, with a password of `postgres`.

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

3.2 Installing the PEM Server on Linux

When installing a PEM server on a RHEL, CentOS, SLES, Debian, or Ubuntu host, you must ensure the following:

1. When using an RPM package to install the PEM server, you must first manually install a backing database and create the database cluster. The server's backing database must be installed via an RPM package or via BitRock. The database must be one of the following versions:

- EDB Postgres Advanced Server version 9.4 or above
- PostgreSQL version 9.4 or above

For detailed information about installing an Advanced Server or PostgreSQL database, please see the product documentation.

2. The `pg_hba.conf` file on the backing database must be configured to use trust authentication for connections. For information about modifying the pg_hba.conf file, visit:

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

3. If you are using a PostgreSQL database, you must also install the `hstore contrib` module; for more information, visit:

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

4. If you are using a firewall, you must allow access to port `8443`; use the commands:

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

   ```
   firewall-cmd --reload
   ```

3.2.1 Additional Prerequisites for RHEL or CentOS HOST

In addition to the above listed prerequisites, the following prerequisites are applicable if you are using a RHEL or CentOS host:

1. You must install the `epel-release` package on the host by running any one of the following commands:

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

Note: You may need to enable the `[extras]` repository definition in the `CentOS-Base.repo` file (located in `/etc/yum.repos.d`).

If you are a Red Hat Network user you must also enable the `rhel-<x>-server-optional-rpms` repository to use EPEL packages, where *x* specifies the version of RHEL on the host. You can make the repository accessible by enabling the `RHEL optional subchannel` for `RHN-Classic`. If you have a certificate-based subscription, please see the `Red Hat Subscription Management Guide`.

2. You must also enable the `rhel-<x>-server-extras-rpms` repository, where `x` specifies the version of the RHEL on the host.

3. If you are using a `RHEL 6.x` or `CentOS 6.x` host, you need to stop the default httpd server and also run commands to install `rh-python36` before installing the PEM server. As part of PEM server installation, an httpd24 server will be installed if you are using a RHEL or CentOS 6.x host.

   (a) Stop the default httpd server using the command:

      `service httpd stop`

   (b) Run the following commands to install `rh-python36`:

      `yum install centos-release-scl`

      `yum install rh-python36`

4. You must also have credentials that allow access to the EnterpriseDB repository. For information about requesting credentials, visit:

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

After receiving your repository credentials you can:

1. Create the repository configuration file.
2. Modify the file, providing your user name and password.
3. Install `edb-pem` on RHEL or CentOS host.

Creating a Repository Configuration File

To create the repository configuration file, assume superuser privileges, and invoke the following command:

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

The repository configuration file is named `edb.repo`. The file resides in `/etc/yum.repos.d`.

Modifying the file, providing your user name and password

After creating the `edb.repo` file, use your choice of editor to ensure that the value of the enabled parameter is `1`, and replace the `username` and `password` placeholders in the `baseurl` specification with the name and password of a registered EnterpriseDB user.

```
[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 Installing the PEM Server on a CentOS or RHEL Host

You can use yum to install the PEM server:

```
yum install edb-pem
```

When you install an RPM package that is signed by a source that is not recognized by your system, yum may ask for your permission to import the key to your local server. If prompted, and you are satisfied that the packages come from a trustworthy source, enter `y`, and press `Return` to continue.

During the installation, yum may encounter a dependency that it cannot resolve. If it does, it will provide a list of the required dependencies that you must manually resolve.

If you want to install PEM server on a machine that is in isolated network, you must first create PEM repository on that machine. For more information about creating PEM repository on an isolated network, see *Creating a PEM repository in an Isolated Network*.

3.2.1.2 Installing the PEM Server on a Debian or Ubuntu Host

To install PEM on a Debian or Ubuntu host, you must have credentials that allow access to the EnterpriseDB repository. To request credentials for the repository, contact EnterpriseDB .

The following steps will walk you through using the EnterpriseDB apt repository to install a Debian package. When using the commands, replace the username and password with the credentials provided by EnterpriseDB.

1. Go to https://apt.enterprisedb.com/ and log in as root:

   ```
   sudo su -
   ```

2. Configure the EnterpriseDB repository:

   ```
   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. Add support to your system for secure APT repositories:

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

4. Add the EBD signing key:

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

5. Update the repository metadata:

   ```
   apt-get update
   ```

6. Use the following command to install the Debian package for the PEM server:

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

3.2.1.3 Installing PEM Server on a SLES Host

For detailed information about installing Advanced Server and supporting components on a SLES host, please consult the *EDB Postgres Advanced Server Installation Guide*,available at:

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

SLES packages are available from: https://zypp.enterprisedb.com

Before installing PEM, you must install prerequisite packages. Invoke the following commands, replacing *sp_no* with the service pack that you are using (i.e. SP4):

```
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 Configuring the PEM Server

Before configuring the PEM server, ensure that the sslutils extension is installed for your backing database.

- For an Advanced Server backing database, sslutils extension is by default installed along with Advanced Server.
- If you are using a PostgreSQL backing database, ensure you have access to the PostgreSQL community repository, and use the command:

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

Where, x is the server version.

The PEM server installer includes a script (configure-pem-server.sh) to help automate the configuration process for RPM installations. The script is installed in the /usr/edb/pem/bin directory. To invoke the script, use the command:

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

When invoking the script, you can include command line options to specify configuration properties; the script will prompt you for values that you omit on the command line. The accepted options are:

Option	Description
-acp	Defines PEM Agent certificate path. The default is /root/.pem.
-ci	CIDR formatted network address range that agents will connect to the server from, to be added to the server's pg_hba.conf file. For example, 192.168.1.0/24. The default is 0.0.0.0/0.
-dbi	The directory for the database server installation. For example, /usr/edb/as10 for Advanced Server or /usr/pgsql-10 for PostgreSQL.
-ds	The unit file name of the PEM database server. For Advanced Server, the default file name is edb-as-10; for PostgreSQL, it is postgresql-10.
-ho	The host address of the PEM database server.
-p	The port number of the PEM database server.
-ps	The service name of the pemagent; the default value is pemagent.
-sp	The superuser password of the PEM database server. This value is required.
-su	The superuser name of the PEM database server.
-t	The installation type: Specify 1 if the configuration is for web services and backing database, 2 if you are configuring web services, or 3 if you are configuring the backing database. If you specify 3, please note that the database must reside on the local host.

If you do not provide configuration properties on the command line, you will be prompted for values by the script. When you invoke the script, choose from:

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.

Note: If the web server and the backing database reside on separate hosts, configure the database server first (option 3), and then web services (option 2). The script will exit if the backing database is not configured before web services.

After selecting a configuration option, the script will proceed to prompt you for configuration properties. When the script completes, it will create the objects required by the PEM server, or perform the configuration steps required.

To view script-related help, use the command:

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

If you are using a RHEL or CentOS 6.x host, restart the httpd24 server after configuring the PEM server; use the command:

```
service httpd24-httpd restart
```

After configuring the PEM server, you can access the PEM web interface in your browser. Navigate to:

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

3.3 Creating a PEM Repository in an Isolated Network

You can create a local repository to act as a host for PEM RPM packages if the server on which you wish to install PEM cannot directly access the EnterpriseDB repository. Please note that this is a high-level overview of the steps required; you may need to modify the process for your individual network. To create and use a local repository, you must:

1. Download all the dependencies required for PEM on a machine with internet access using the following commands:

```
yum install yum-plugin-downloadonly

mkdir /tmp/<pem_dir>

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

mkdir /tmp/<epel_dir>

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

Where, `<pem_dir>` and `<epel_dir>` are the local directories that you create for downloading the RPMs.

2. Copy the directories `/tmp/<pem_dir>` and `/tmp/<epel_dir>` to the machine that is in the isolated network.

3. Create the repositories:

```
yum install createrepo

createrepo /tmp/<pem_dir>

createrepo /tmp/<epel_dir>
```

5. Create a repository configuration file called `/etc/yum.repos.d/pem.repo` with connection information that specifies:

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

6. Create a repository configuration file called `/etc/yum.repos.d/epel.repo` with connection information that specifies:

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

After specifying the location and connection information for your local repository, you can use yum commands to install or upgrade PEM server:

- To install PEM server:

  `yum install edb-pem`

- To upgrade PEM server:

```
yum upgrade edb-pem
```

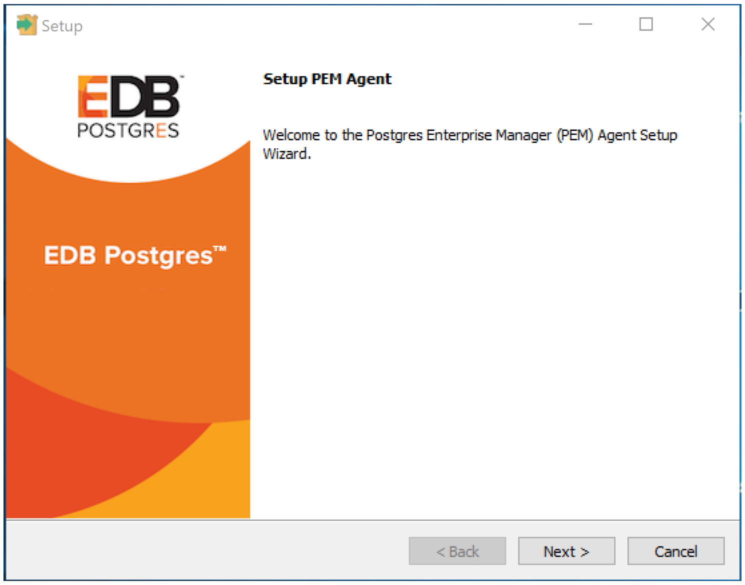
For more information about creating a local yum repository, visit: https://wiki.centos.org/HowTos/CreateLocalRepos

3.4 Installing a PEM Agent on Windows

To invoke the PEM agent installer, assume `Administrative` privileges and navigate into the directory that contains the installer. Then, invoke the installer with the command:

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

The `Setup...` page opens, welcoming you to the PEM Agent installer.

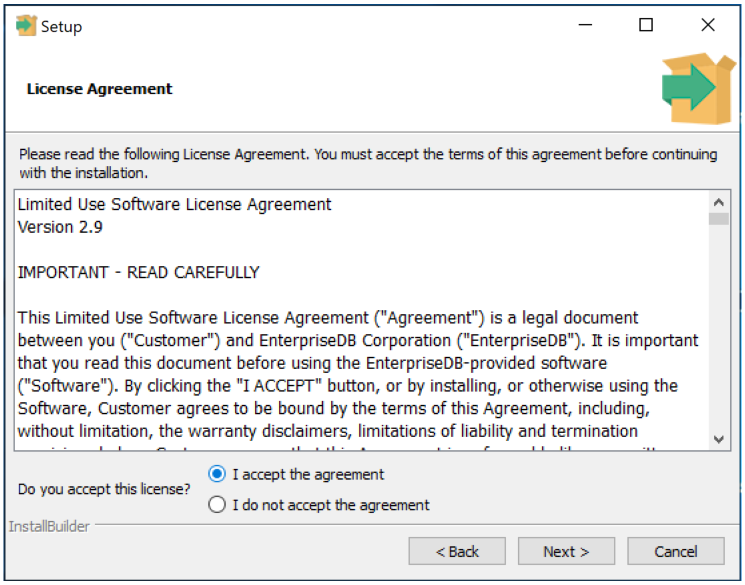


Setup PEM Agent

Welcome to the Postgres Enterprise Manager (PEM) Agent Setup Wizard.

Fig. 3.44: *The PEM Agent Installer's Welcome dialog*

Click `Next` to continue to the `License Agreement`.



License Agreement

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

Limited Use Software License Agreement
Version 2.9

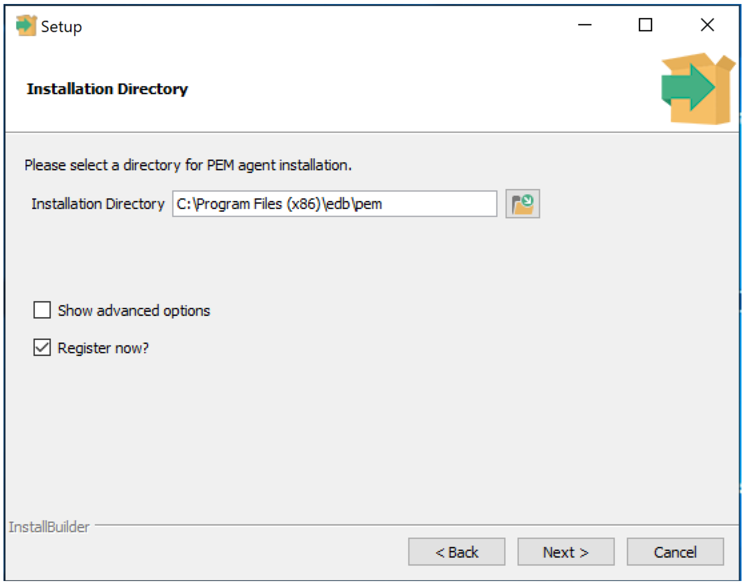
IMPORTANT - READ CAREFULLY

This Limited Use Software License Agreement ("Agreement") is a legal document between you ("Customer") and EnterpriseDB Corporation ("EnterpriseDB"). It is important that you read this document before using the EnterpriseDB-provided software ("Software"). By clicking the "I ACCEPT" button, or by installing, or otherwise using the Software, Customer agrees to be bound by the terms of this Agreement, including, without limitation, the warranty disclaimers, limitations of liability and termination

Do you accept this license?
● I accept the agreement
○ I do not accept the agreement

Fig. 3.45: *The PEM License Agreement*

Carefully review the license agreement before highlighting the appropriate radio button and accepting the agreement; click `Next` to continue to the `Installation Directory` dialog.



Please select a directory for PEM agent installation.

Installation Directory: C:\Program Files (x86)\edb\pem

☐ Show advanced options

☑ Register now?

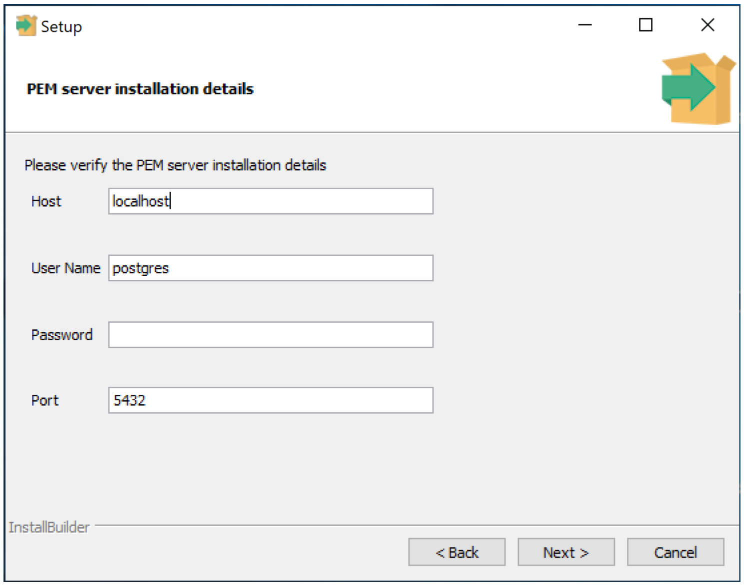
Fig. 3.46: *Specify an Installation Directory*

By default, the PEM agent is installed in the `C:\Program Files (x86)\edb\pem` directory.You can accept the default installation directory, or modify the contents of the `Installation Directory` field, specifying an alternate installation directory for the PEM agent.

By default, the PEM agent installer places a certificate in the Administrator's `%APPDATA%\pem` directory. Check the `Show advanced options` box to indicate that you would like the PEM agent installer to include a dialog that allows you to specify an alternate path for the certificate file.

Check the box next to `Register now?` to instruct the installer to register the newly installed PEM agent with the PEM server.

Click `Next` to continue to the `PEM Server Installation Details` dialog.



Setup

PEM server installation details

Please verify the PEM server installation details

Host: localhost

User Name: postgres

Password:

Port: 5432

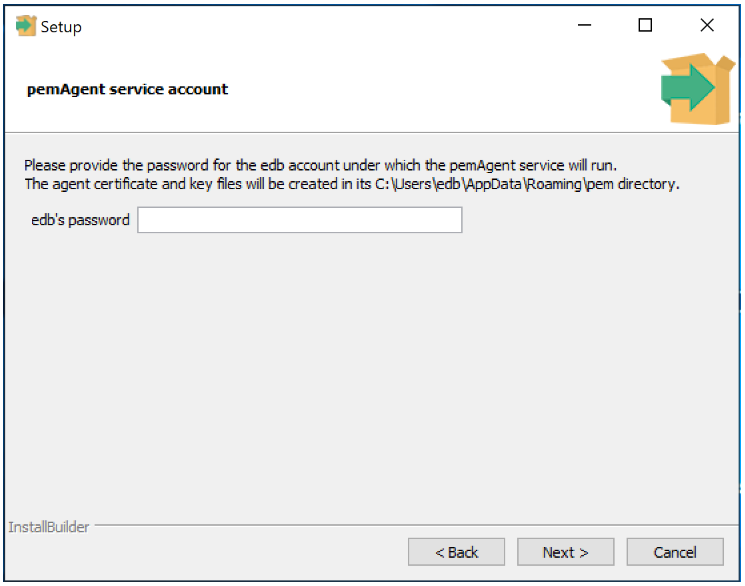
Fig. 3.47: *Provide PEM server installation details*

Enter the connection details for the PEM server on the PEM server installation details dialog:

- Specify the name or IP address of the system on which the PEM database server resides in the Host field. Please note: If the PEM-HTTPD web server and PEM database are hosted on different systems, you must specify *the host of the PEM database*.
- Specify the name of the database superuser in the User Name field.
- Specify the password associated with the database superuser in the Password field.
- Specify the port that PostgreSQL is monitoring in the Port field.

Click Next to continue to pemAgent Service Account. The installer will attempt to connect to the server to verify that the details are correct.

Note: The PEM server must allow connections from the PEM agent installer. If you encounter a connection error, confirm the connection properties specified on the PEM Server Installation Details dialog are correct, and confirm that the pg_hba.conf file (on the PEM server) will allow a connection to the server described in the error message.

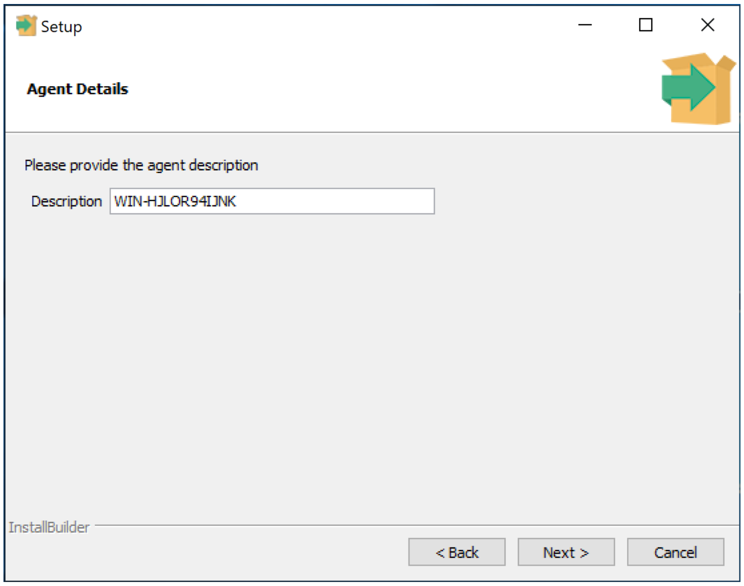


Please provide the password for the edb account under which the pemAgent service will run.
The agent certificate and key files will be created in its C:\Users\edb\AppData\Roaming\pem directory.

edb's password

Fig. 3.48: *pemAgent Service Account password*

Provide the password for the edb account under which the pemAgent service will run. The agent certificate and key files will be created in `C:\Users\edb\App` Provide the password for the edb account under which the pemAgent service will run. The agent certificate and key files will be created in `C:\Users\edb\AppData\Roaming\pem` directory. Click `Next` to continue to `Agent Details` dialog.



Please provide the agent description

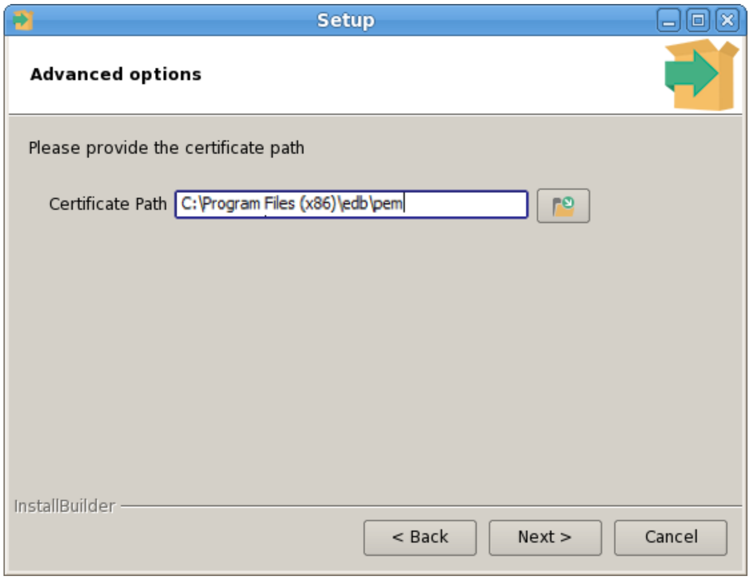
Description WIN-HJLOR94IJNK

Fig. 3.49: *Specify a descriptive name for the PEM agent*

The tree control displayed in the `Browser` panel of the PEM web interface displays the value entered in the `Description` field to identify the PEM agent. Specify a descriptive name for the agent, such as the hostname of the machine the agent is installed on, or a name that reflects the host's functionality.

Provide a descriptive name, or accept the default provided by the PEM agent host, and click `Next` to continue.

If you checked the `Show advanced options` checkbox, the `Advanced options` dialog opens.



Setup

Advanced options

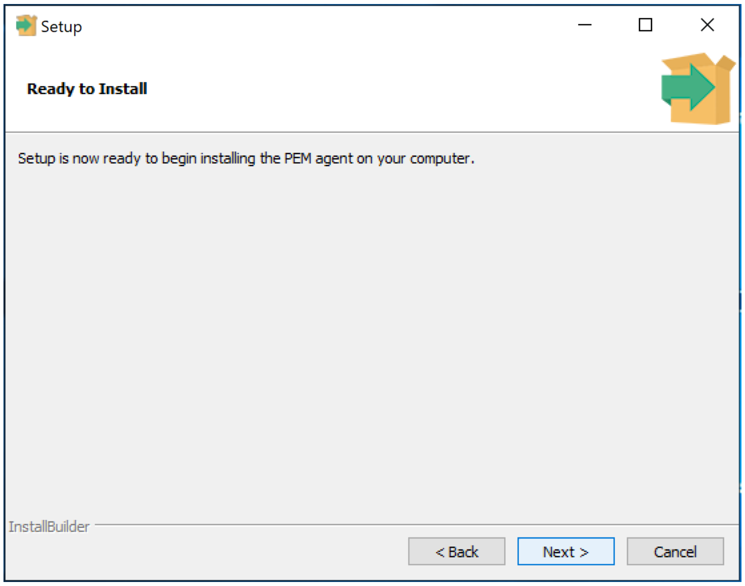
Please provide the certificate path

Certificate Path: C:\Program Files (x86)\edb\pem

Fig. 3.50: *Provide the certificate path*

By default, the PEM agent installer places the certificate in the `C:\Program Files (x86)\edb\pem` directory. Specify an alternate path for the certificate or accept the default and click `Next`.

The wizard is now ready to install the PEM agent; click `Back` to amend the installation directory, or `Next` to continue.



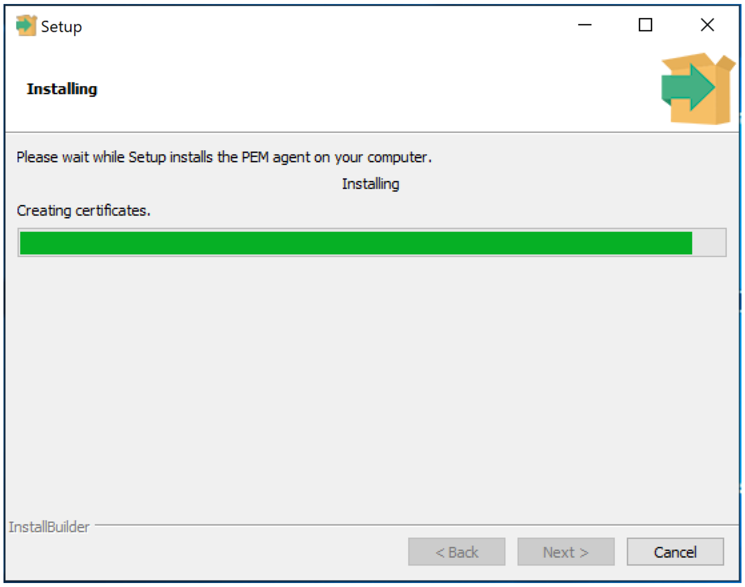
Setup

Ready to Install

Setup is now ready to begin installing the PEM agent on your computer.

Fig. 3.51: *The PEM Agent installation is ready to begin*

Click `Next` on the `Ready to Install` dialog to instruct the installer to copy files to the system and register the agent on the PEM server.



Setup

Installing

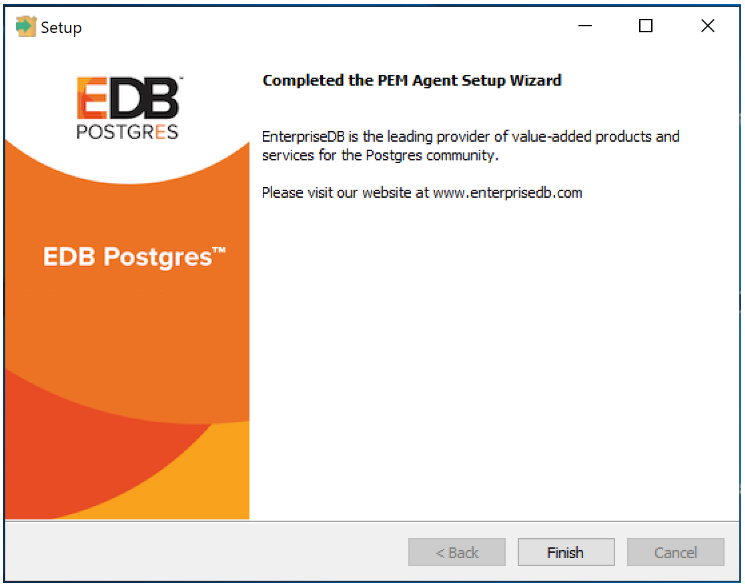
Please wait while Setup installs the PEM agent on your computer.

Installing

Creating certificates.

Fig. 3.52: *Progress bars mark the installation's progress*

The PEM agent installer displays progress bars to mark the PEM agent's installation progress.



Setup

Completed the PEM Agent Setup Wizard

EnterpriseDB is the leading provider of value-added products and services for the Postgres community.

Please visit our website at www.enterprisedb.com

Fig. 3.53: *The PEM Agent installation is complete*

When the installation has completed, the PEM agent will be running and reporting operating system and host data to the PEM server. To start monitoring Postgres instances on the host of the PEM agent, they must now be added to PEM's enterprise directory and bound to the agent.

3.4.1 Invoking the Agent Installer from the Command Line

The command line options of the PEM agent installers offer functionality in situations where a graphical installation may not work because of limited resources or system configuration. You can:

- Include the `--mode unattended` option when invoking the installer to perform an installation without additional user input.

Not all command line options are suitable for all platforms. For a complete reference guide to the command line options, include the `--help` option when you invoke the installer.

3.4.1.1 Invoking the PEM Agent Installer in Unattended Mode

You can perform an unattended PEM server installation by providing installation preferences on the command line when invoking the installer. Please note that the system on which you are installing the PEM server must have internet access.

Before invoking the PEM agent installer in unattended mode, you must:

- install the PEM server; the `pg_hba.conf` file of the PEM server must allow connections from the host of the PEM agent.
- disable SE Linux on the host of the PEM agent; for more information, see section *Software Prerequisites*.
- ensure that the monitored Postgres database has SSL enabled, and is accepting connections.

You must have Administrator privileges to install the PEM agent. Use the following command to invoke the PEM agent installer in unattended mode:

```
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>
```

Where:

`pem_server_host_address` specifies the IP address of the host of the PEM server.

`pem_server_port` specifies the port used by the backing PEM database; by default, the database uses port `5432`.

`pguser_password` specifies the password associated with the PEM database superuser.

`agent_name` specifies a descriptive name for the PEM agent.

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

Note: When configuring a shell/batch script run by a Windows agent that has PEM 7.11 or later version installed, the `AllowBatchJobSteps` parameter must be set to `True` in the `agent.cfg` file. The pemagent will not execute any batch/shell script by default.

3.5 Installing a PEM Agent on Linux

3.5.1 Installing a PEM agent on a CentOS or RHEL host

On a Linux system, you can use the `yum` package manager to install a PEM agent. Please note that before using a package manager to install the PEM agent on a host, you must:

- Install the `epel-release` package on the host by running any one of the following commands:
- `yum -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm`
- `yum install epel-release`

> **Note:** You may need to enable the `[extras]` repository definition in the `CentOS-Base.repo` file (located in `/etc/yum.repos.d`).

- You must also have credentials that allow access to the EnterpriseDB repository. For information about requesting credentials, visit:

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

After receiving your repository credentials you can:

1. Create the repository configuration file.
2. Modify the file, providing your user name and password.
3. Install `edb-pem-agent`.

Creating a Repository Configuration File

To create the repository configuration file, assume superuser privileges, and invoke the following command:

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

The repository configuration file is named `edb.repo`. The file resides in `/etc/yum.repos.d`.

Modifying the file, providing your user name and password

After creating the `edb.repo` file, use your choice of editor to ensure that the value of the enabled parameter is `1`, and replace the `username` and `password` placeholders in the `baseurl` specification with the name and password of a registered EnterpriseDB user.

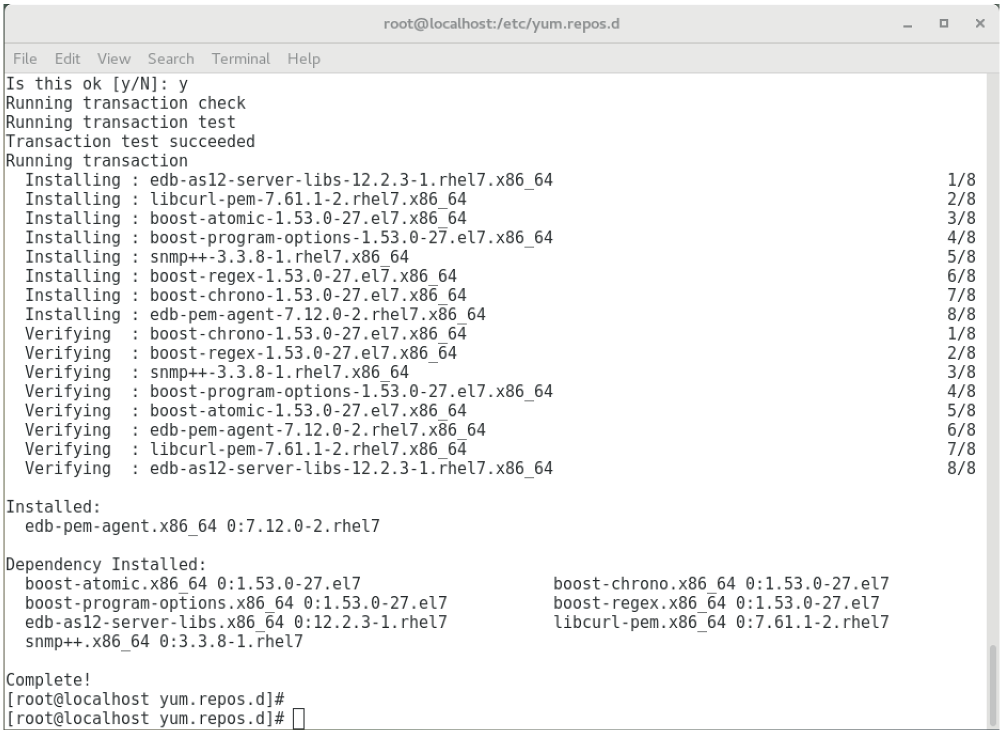
```
[edb]
name=EnterpriseDB RPMs $releasever - $basearch
baseurl=https://<username>:<password>@yum.enterprisedb.com/edb/redhat/rhel-
↪$releasever-$basearch
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/ENTERPRISEDB-GPG-KEY
```

Installing PEM Agent

After saving your changes to the configuration file, you can use the yum install command to install `edb-pem-agent`. For example, the following command installs PEM Agent:

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

When the installation is complete, `yum` will display a list of the installed packages and dependencies.


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]# 
```

Fig. 3.54: *Using an RPM package to install the PEM agent*

When you install an RPM package that is signed by a source that is not recognized by your system, yum may ask for your permission to import the key to your local server. If prompted, and you are satisfied that the packages come from a trustworthy source, enter `y`, and press `Return` to continue.

During the installation, yum may encounter a dependency that it cannot resolve. If it does, it will provide a list of the required dependencies that you must manually resolve.

3.5.2 Installing a PEM Agent on a Debian or Ubuntu Host

To install PEM on a Debian or Ubuntu host, you must have credentials that allow access to the EnterpriseDB repository. To request credentials for the repository, contact EnterpriseDB .

The following steps will walk you through using the EnterpriseDB apt repository to install a Debian package. When using the commands, replace the username and password with the credentials provided by EnterpriseDB.

1. Go to https://apt.enterprisedb.com/ and log in as root:

   ```
   sudo su -
   ```

2. Configure the EnterpriseDB repository:

   ```
   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. Add support to your system for secure APT repositories:

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

4. Add the EBD signing key:

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

5. Update the repository metadata:

   ```
   apt-get update
   ```

6. Use the following command to install the Debian package for the PEM agent:

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

3.5.3 Installing a PEM Agent on a SLES Host

For detailed information about installing Advanced Server and supporting components on a SLES host, please consult the

EDB Postgres Advanced Server Installation Guide

SLES packages are available from:

> https://zypp.enterprisedb.com

Before installing PEM, you must install prerequisite packages. Use the following commands replacing *sp_no* with the service pack that you are using (i.e. SP2 or SP3):

```
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 Registering an Agent

Each PEM agent must be *registered* with the PEM server. The registration process provides the PEM server with the information it needs to communicate with the agent. The PEM agent graphical installer supports agent self-registration, but you can use the `pemworker` utility to register the agent if you skip PEM agent registration during a graphical installation or use an RPM package to install a PEM agent.

The RPM installer places the PEM agent in the `/usr/edb/pem/agent/bin` directory. To register an agent, include the `--register-agent` keywords along with registration details when invoking the `pemworker` utility:

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

Append command line options to the command string when invoking the pemworker utility. Each option should be followed by a corresponding value:

Option	Description
`--pem-server`	Specifies the IP address of the PEM backend database server. This parameter is required.
`--pem-port`	Specifies the port of the PEM backend database server. The default value is 5432.
`--pem-user`	Specifies the name of the Database user (having superuser privileges) of the PEM backend database server. This parameter is required.
`--pem-agent-user`	Specifies the agent user to connect the PEM server backend database server.
`--cert-path`	Specifies the complete path to the directory in which certificates will be created. If you do not provide a path, certificates will be created in: On Linux, ~/.pem On Windows, %APPDATA%/pem
`--config-dir`	Specifies the directory path where configuration file can be found. The default is the <pemworker path>/../etc.
`--display-name`	Specifies a user-friendly name for the agent that will be displayed in the PEM Browser tree control. The default is the system hostname.
`--force-registration`	Include the force_registration clause to instruct the PEM server to register the agent with the arguments provided; this clause is useful if you are overriding an existing agent configuration. The default value is Yes.
`--group`	The name of the group in which the agent will be displayed.
`--team`	The name of the database role, on the PEM backend database server, that should have access to the monitored database server.
`--owner`	The name of the database user, on the PEM backend database server, who will own the agent.
`--allow_server_restart`	Enable the allow-server_restart parameter to allow PEM to restart the monitored server. The default value is True.
`--allow-batch-probes`	Enable the allow-batch-probes parameter to allow PEM to run batch probes on this agent. The default value is False.
`--batch-script-user`	Specifies the operating system user that should be used for executing the batch/shell scripts. The default value is none; the scripts will not be executed if you leave this parameter blank or the specified user does not exist.
`--enable-heartbeat-connection`	Enable the enable-heartbeat-connection parameter to create a dedicated heartbeat connection between PEM Agent and server to update the active status. The default value is False.
`--enable-smtp`	Enable the enable-smtp parameter to allow the PEM agent to send the email on behalf of the PEM server.The default value is False.
`--enable-snmp`	Enable the enable-snmp parameter to allow the PEM agent to send the SNMP traps on behalf of the PEM server.The default value is False.
`-o`	Specify if you want to override the configuration file options.

If you want to use any PEM feature for which database server restart is required by the pemagent such as Audit Manager, Log Manager, or Tuning Wizard, then you must set the value for `allow_server_restart` as `true` in the `agent.cfg` file.

Note: When configuring a shell/batch script run by a PEM agent that has PEM 7.11 or later version installed, the user for the `batch_script_user parameter` must be specified. It is strongly recommended that a non-root user is used to run the scripts. Using the root user may result in compromising the data security and operating system security. However, if you want to restore the pemagent to its original settings using `root` user to run the scripts, then the `batch_script_user` parameter value must be set to `root`.

Before any changes are made on the PEM database, the connecting agent is authenticated with the PEM database server. When invoking the pemworker utility, you must provide the password associated with the PEM server administrative user role (`postgres`). There are three ways to specify the administrative password; you can:

- set the `PEM_MONITORED_SERVER_PASSWORD` environment variable.
- provide the password on the command line with the `PGPASSWORD` keyword.
- create an entry in the `.pgpass` file.

Failure to provide the password will result in a password authentication error; you will be prompted for any other required but omitted information. When the registration is complete, the server will confirm that the agent has been successfully registered.

Setting PEM Agent Configuration Parameters

The PEM agent RPM installer creates a sample configuration file named `agent.cfg.sample` in the `/usr/edb/pem/agent/etc` directory. When you register the PEM agent, the `pemworker` program creates the actual agent configuration file (named `agent.cfg`). You must modify the `agent.cfg` file, adding the following configuration parameter:

```
heartbeat_connection = true
```

You must also add the location of the `ca-bundle.crt` file (the certificate authority). By default, the installer creates a `ca-bundle.crt` file in the location specified in your `agent.cfg.sample` file. You can copy the default parameter value from the sample file, or, if you use a `ca-bundle.crt` file that is stored in a different location, specify that value in the `ca_file` parameter:

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

Then, use a platform-specific command to start the PEM agent service; the service is named `pemagent`. For example, on a CentOS or RHEL 6.x system, you would use the command:

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

On a CentOS or RHEL 7.x host, use systemctl to start the service:

```
systemctl start pemagent
```

The service will confirm that it is starting the agent; when the agent is registered and started, it will be displayed on the `Global Overview` and in the `Object browser` of the PEM web interface.

For information about using the `pemworker` utility to register a server, please see the PEM Administrator's Guide

CHAPTER 4

The PEM Web Interface

After installing a PEM server and agent, you can configure PEM to start monitoring and managing PostgreSQL or Advanced Server instances. The PEM server installer installs the PEM web interface. You can use the interface to review information about objects that reside on monitored servers, or to review statistical information gathered by the PEM server.

After installing and configuring PEM, you can use your browser to access the PEM web interface. Open your browser, and navigate to:

```
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:

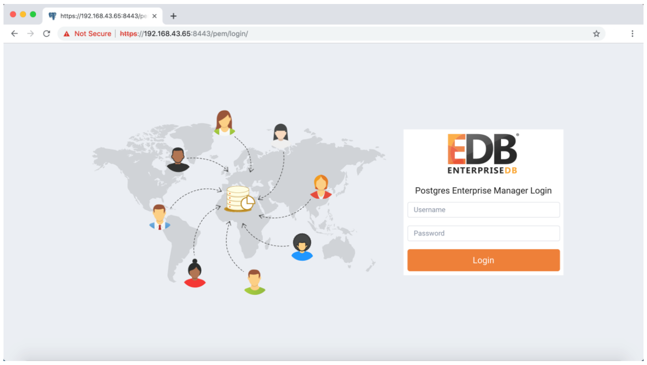
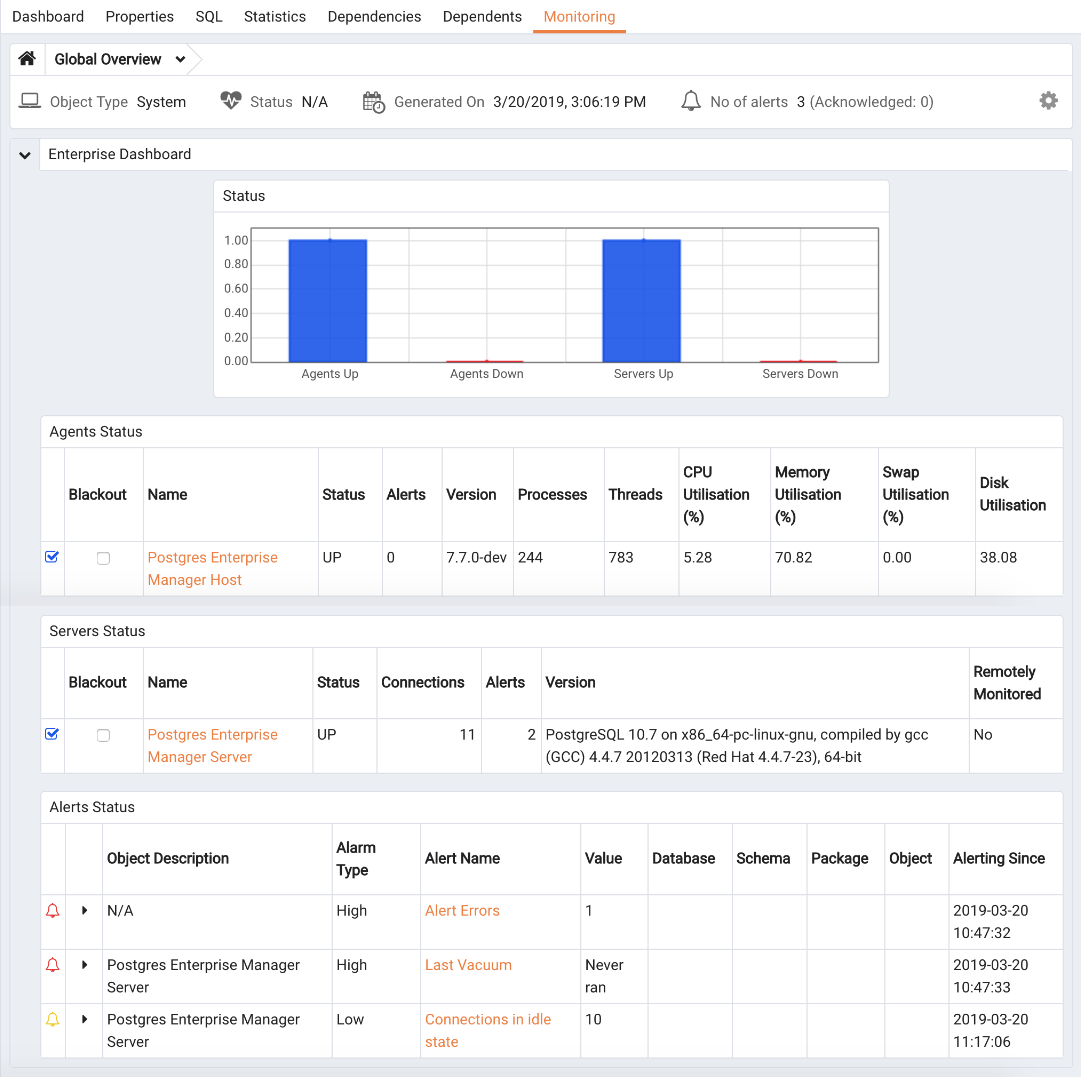

https://192.168.43.65:8443/pe
Not Secure
https://192.168.43.65:8443/pem/login/
EDB
ENTERPRISEDB
Postgres Enterprise Manager Login
Username
Password
Login

Fig. 4.1: *The PEM Web Login page*

Use the fields on the `Postgres Enterprise Manager Login` window to authenticate yourself with the PEM server:

- Provide the name of a `pem` database user in the `Username` field. For the first user connecting, this will be the name provided when installing the PEM server.
- Provide the password associated with the user in the `Password` field.

Click the Login button to connect to the PEM server.



Object Type System | Status N/A | Generated On 3/20/2019, 3:06:19 PM | No of alerts 3 (Acknowledged: 0)

Agents Status

	Blackout	Name	Status	Alerts	Version	Processes	Threads	CPU Utilisation (%)	Memory Utilisation (%)	Swap Utilisation (%)	Disk Utilisation
☑	□	Postgres Enterprise Manager Host	UP	0	7.7.0-dev	244	783	5.28	70.82	0.00	38.08

Servers Status

	Blackout	Name	Status	Connections	Alerts	Version	Remotely Monitored
☑	□	Postgres Enterprise Manager Server	UP	11	2	PostgreSQL 10.7 on x86_64-pc-linux-gnu, compiled by gcc (GCC) 4.4.7 20120313 (Red Hat 4.4.7-23), 64-bit	No

Alerts Status

		Object Description	Alarm Type	Alert Name	Value	Database	Schema	Package	Object	Alerting Since
🔔	▸	N/A	High	Alert Errors	1					2019-03-20 10:47:32
🔔	▸	Postgres Enterprise Manager Server	High	Last Vacuum	Never ran					2019-03-20 10:47:33
🔔	▸	Postgres Enterprise Manager Server	Low	Connections in idle state	10					2019-03-20 11:17:06

Fig. 4.2: *The PEM Web interface*

Before you can use the PEM web interface to manage or monitor a database server, you must *register* the server with the PEM server. When you register a server, you describe the connection to the server, provide authentication information for the connection, and specify any management preferences (optionally binding an agent).

A server may be managed or unmanaged:

- A managed server is bound to a PEM agent. The PEM agent will monitor the server to which it is bound, and perform tasks or report statistics for display on the PEM dashboards. A managed server has access to extended PEM functionality such as Package Management or Custom Alerting; when registering a server, you can also allow a managed server to be restarted by PEM as required.
- An unmanaged server is not bound to a PEM agent; you can create database objects on an unmanaged server, but extended PEM functionality (such as Package Management or Custom Alerting) is not supported on an unmanaged server.

You must also ensure the pg_hba.conf file of the server that you are registering allows connections from the host of the PEM web interface.

To access online help information about the PEM web interface, select `Help` from the menu bar. Additional information is available in .pdf and .html format from the EnterpriseDB website

- The `PEM Administrator's Guide` contains information about registering and managing servers, agents, and users.
- The `PEM Enterprise Features Guide` contains information about using the tools and wizards that are part of the web interface.
- The `PEM Agent User Guide` contains helpful information about managing your PEM agents.
- The `PEM Upgrade and Migration Guide` contains information about upgrading to PEM 7.8 from a previous version.
- The `PEM PgBouncer Configuration Guide` contains information about using PgBouncer with your PEM installation.
- The `PEM EDB Ark Management Guide` contains information about using PEM to manage cloud installations of Advanced Server and PostgreSQL.

CHAPTER 5

Installing the SQL Profiler Plugin

The SQL Profiler Plugin allows you to profile a server's workload. You must install the plugin on each server on which you wish to use SQL Profiler. For example, if you have a host running PostgreSQL 9.6 and PostgreSQL 10, you must install two versions of the plugin, one for each server.

SQL Profiler is officially supported only on the EnterpriseDB distributions of PostgreSQL version 9.4 or above and Advanced Server version 9.4 or above. The plugin is distributed via StackBuilder, or is available from the EnterpriseDB website

The plugin is also distributed and installed with the server component of the Advanced Server installer.

The SQL Profiler plugin may be installed on servers with or without a PEM agent, however traces can only be run in ad-hoc mode on unmanaged servers, and may only be scheduled on managed servers.

Follow the installation steps listed below to install the plugin for PostgreSQL before continuing to the `Configuration` section. If you are using Advanced Server, you can skip ahead to the `Configuration` section.

You can use the graphical installer to install any version of SQL Profiler on Windows platform.

On Linux, if you have installed your database server through graphical installer then you must use the graphical installer to install the SQL Profiler. If you have installed your database server using the RPM package, then you must use the RPM package to install the SQL Profiler.

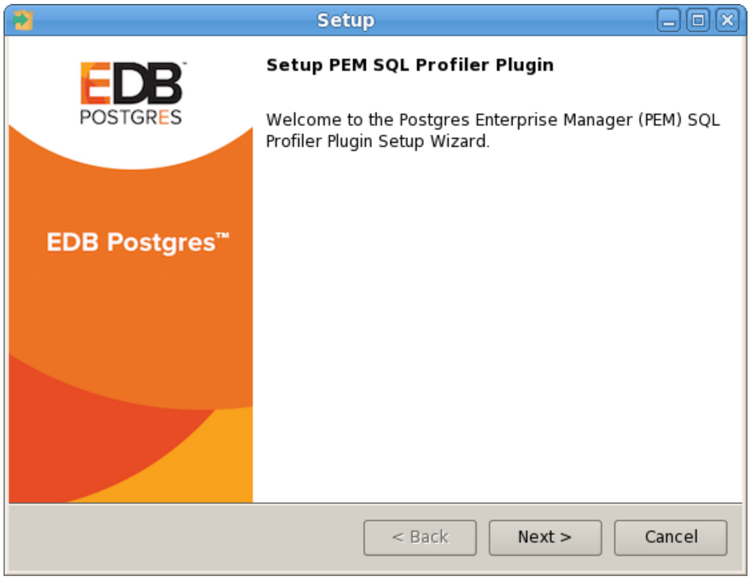
5.1 Installing SQL Profiler using Graphical Installer

To invoke the SQL Profiler graphical installer, assume superuser privileges (or `Administrator` privileges on Windows), navigate into the directory that contains the installer, and invoke the installer with the command:

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

Where, `pg_version` is the version of your postgres and `sql_profiler_version` is the version of SQL Profiler.

The SQL Profiler installer welcomes you to the Setup Wizard.

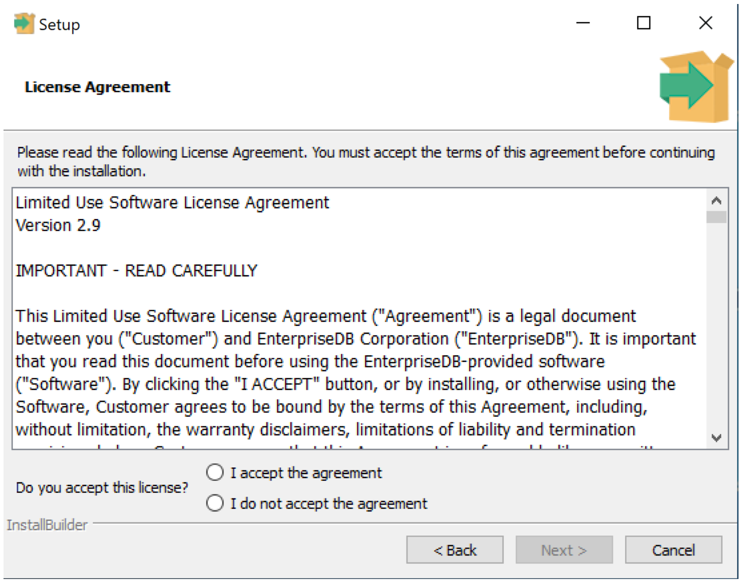


Setup PEM SQL Profiler Plugin

Welcome to the Postgres Enterprise Manager (PEM) SQL Profiler Plugin Setup Wizard.

Fig. 5.1: *The SQL Profiler Installer Welcome dialog*

Click `Next` to continue to the `License Agreement`.



License Agreement

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

Limited Use Software License Agreement
Version 2.9

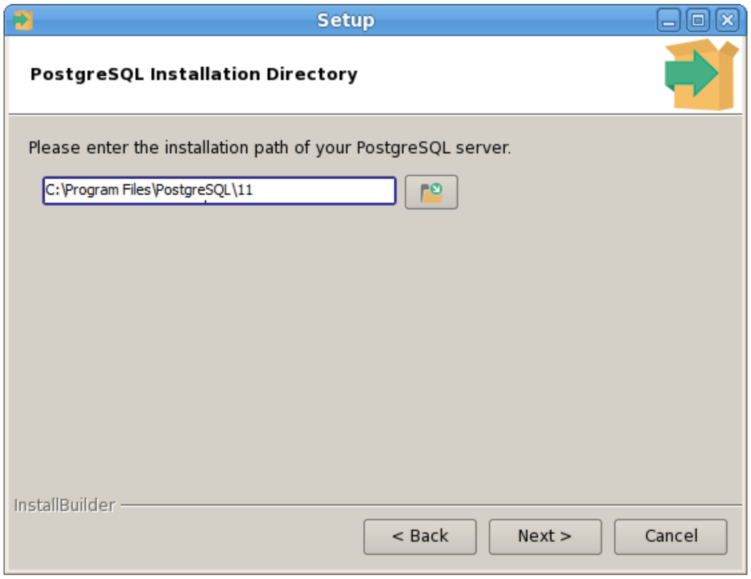
IMPORTANT - READ CAREFULLY

This Limited Use Software License Agreement ("Agreement") is a legal document between you ("Customer") and EnterpriseDB Corporation ("EnterpriseDB"). It is important that you read this document before using the EnterpriseDB-provided software ("Software"). By clicking the "I ACCEPT" button, or by installing, or otherwise using the Software, Customer agrees to be bound by the terms of this Agreement, including, without limitation, the warranty disclaimers, limitations of liability and termination

Do you accept this license?
○ I accept the agreement
○ I do not accept the agreement

Fig. 5.2: *The SQL Profiler License Agreement*

Carefully review the license agreement before highlighting the appropriate radio button and accepting the agreement; click `Next` to continue to the `Installation Directory` dialog.



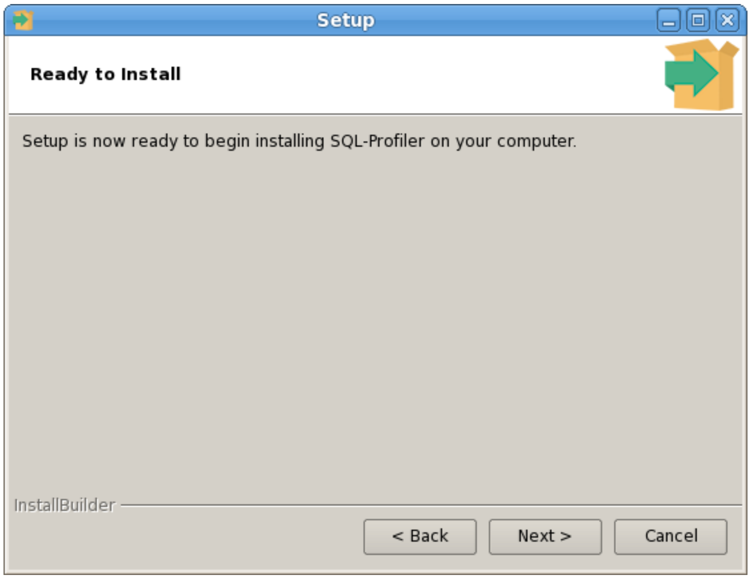
PostgreSQL Installation Directory

Please enter the installation path of your PostgreSQL server.

C:\Program Files\PostgreSQL\11

Fig. 5.3: *The PostgreSQL Installation Directory*

Specify an alternate location for the installation directory, or accept the default location and click `Next` to continue.

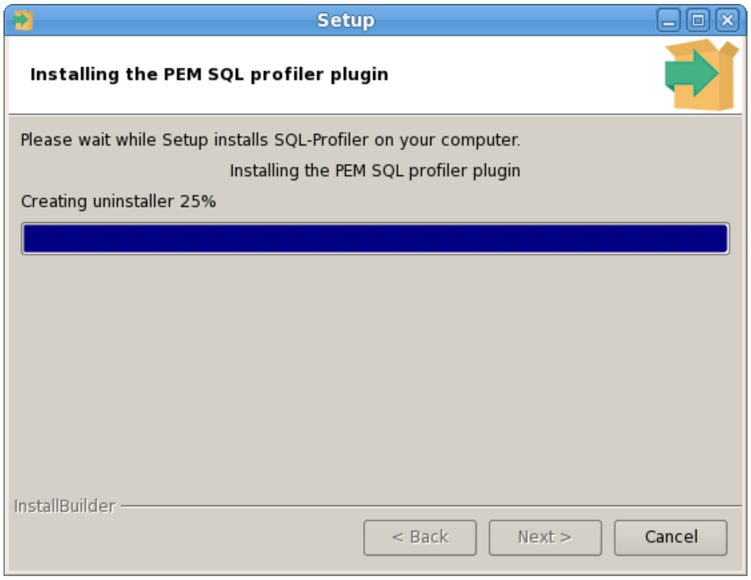


Ready to Install

Setup is now ready to begin installing SQL-Profiler on your computer.

Fig. 5.4: *Ready to Install*

The wizard is now ready to install the SQL Profiler plugin. Click `Next` to continue.



Installing the PEM SQL profiler plugin

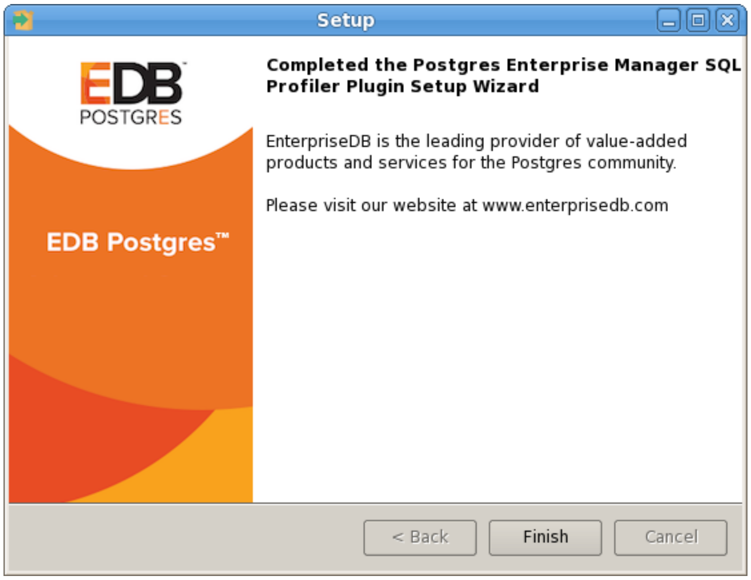
Please wait while Setup installs SQL-Profiler on your computer.

Installing the PEM SQL profiler plugin

Creating uninstaller 25%

Fig. 5.5: *Installing the SQL Profiler plugin*

The SQL Profiler plugin installer displays progress bars as it copies files to your system.



Completed the Postgres Enterprise Manager SQL Profiler Plugin Setup Wizard

EnterpriseDB is the leading provider of value-added products and services for the Postgres community.

Please visit our website at www.enterprisedb.com

Fig. 5.6: *The SQL Profiler installation is complete*

When the installation is complete, the SQL Profiler plugin is ready to be configured.

5.2 Installing SQL Profiler using RPMs

You can install SQL Profiler using rpm on RHEL or Centos 6 or 7, using yum command as root user:

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

Where, `pg_version` is the version of your postgres and `sql_profiler_version` is the version of SQL Profiler.

5.3 Configuring SQL Profiler

The SQL Profiler plugin is not automatically enabled when the installation process completes. This allows you to restart the server at a convenient time, and prevents the plugin from being loaded unnecessarily on systems where it is not required on a continual basis.

Use the following steps to enable the plugin:

1. Edit the `postgresql.conf` file on the server you wish to profile, modifying the `shared_preload_libraries` parameter as shown below:

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

2. Restart the Postgres server.
3. Using the `Query Tool` or the `psql` command line interface, run the `sql-profiler.sql` script in the database specified as the `Maintenance Database` on the server you wish to profile. If you are using:

- PostgreSQL, the default maintenance database is `postgres`.
- Advanced Server, the default maintenance database is `edb`.

To use the PEM Query Tool to run the script, highlight the name of the maintenance database in the `Browser` tree control, and navigate through the `Tools` menu to select `Query tool`. When the Query Tool opens, use the `Open` option on the `Files` menu to open a web browser and navigate to the `sql-profiler.sql` script. By default, the `sql-profiler.sql` script is located in the `contrib` folder, under your Postgres installation.

When the script opens in the `SQL Editor` panel of the Query Tool, highlight the content of the script in the SQL Editor and select the `Execute` option from the `Query` menu (or click the `Execute` icon) to invoke the script and configure SQL Profiler.

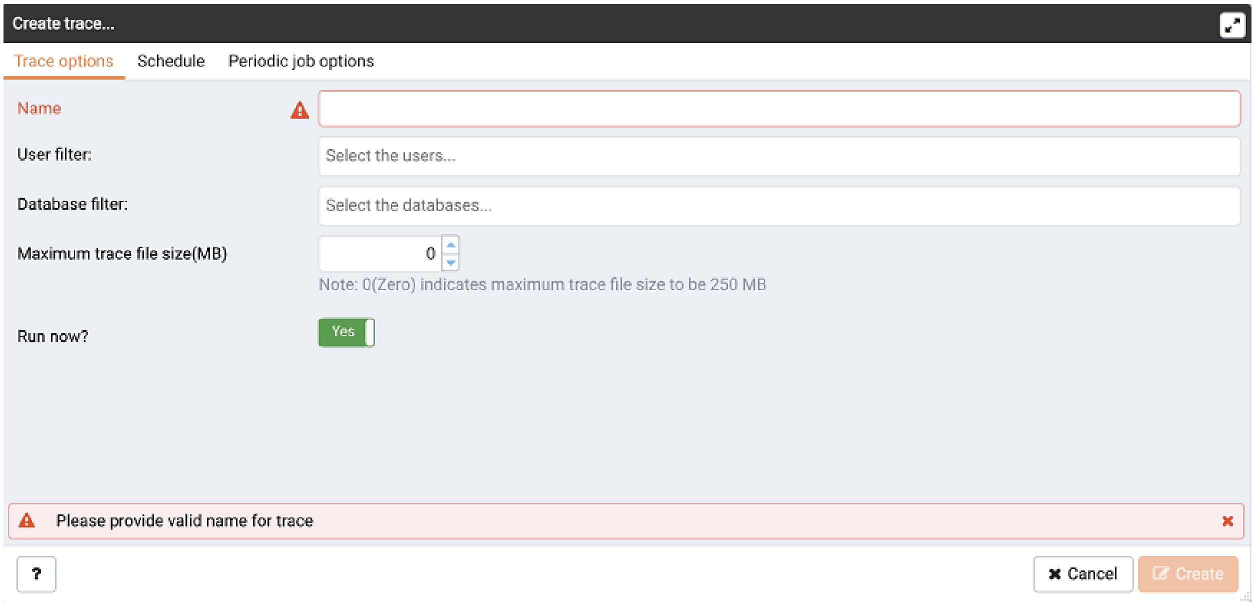
You can also use the psql command line to invoke the configuration script. The following command uses psql to invoke the `sql-profiler.sql` script on an Advanced Server database on a Linux system:

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

After configuring SQL Profiler, it is ready to use with all databases that reside on the server.

To access SQL Profiler functionality, highlight the name of the database in the PEM `Browser` tree control; navigate through `Server` option under `Tools` menu to the `SQL Profiler` pull-aside menu. Menu options allow you to manage your SQL traces:

- Select `Create trace...` to define a new trace.
- Select `Open trace...` to open an existing trace.
- Select `Delete trace(s)...` to delete one or more traces.
- Select `View scheduled trace(s)...` to review a list of scheduled traces.



Create trace...

Name:

User filter: Select the users...

Database filter: Select the databases...

Maximum trace file size(MB): 0

Note: 0(Zero) indicates maximum trace file size to be 250 MB

Run now? Yes

Please provide valid name for trace

Fig. 5.7: *Creating a new trace*

For more information about using SQL Profiler, consult the online help text for SQL Profiler (accessed through the `Help` menu) or the *PEM Administrator's Guide*.

CHAPTER 6

Upgrading and Uninstalling PEM Components

For detailed information about upgrading and moving PEM components, please see the *PEM Upgrade and Migration Guide*. PEM documentation and links to PEM and SQL Profiler installers are available from the EnterpriseDB website

Upgrading Postgres Enterprise Manager on Windows™

To upgrade your version of PEM on Windows, simply invoke the PEM component installers. Each installer will notify you if it detects a previous installation, and will upgrade to the more recent version of PEM.

You should invoke the PEM installers in the following order:

1. Invoke the PEM agent installer on each monitored node *except* the PEM server node.
2. Invoke the PEM server installer; this installer will upgrade *both* the PEM server and the PEM agent that resides on the PEM server node.

For detailed information about upgrading from a previous version, see the *PEM Upgrade and Migration Guide*.

Upgrading SQL Profiler on a Linux Host

To upgrade a SQL Profiler installation that resides on a Linux host:

1. Delete the existing SQL Profiler query set on each node by invoking the `uninstall-sql-profiler.sql` script.

- By default, if you are using Advanced Server on a Linux host, the script resides in:

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

- If you are using a PostgreSQL installation on a Linux host, the script resides in:

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

  Where *x.x* specifies the installed Postgres version.

2. Then, invoke the new SQL Profiler installer on each node you wish to profile.

Upgrading SQL Profiler on a Windows Host

If you are using SQL Profiler on a Windows host, Windows will lock any files that have been executed or loaded into memory. To release any locked files, you must stop the Postgres server before performing an upgrade.

After stopping the Postgres Server:

1. Delete the existing SQL Profiler query set on each node by invoking the `uninstall-sql-profiler.sql` script.

- If you are using Advanced Server on a Windows host, the script resides in:

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

- If you are using a PostgreSQL installation on a Windows host, the script resides in:

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

  Where *x* specifies the installed Postgres version.

2. Invoke the new SQL Profiler installer on each node you wish to profile.

   Then, restart the Postgres Server, to resume profiling the node.

Troubleshooting

Upgrading SQL Profiler on a Windows Host

After upgrading to a newer version of SQL Profiler, if you encounter the following error:

```
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.
```

To correct this error, you must replace the existing query set with a new query set. First, uninstall SQL Profiler by invoking the `uninstall-sql-profiler.sql` script, and then reinstall SQL Profiler by invoking the `sql-profiler.sql` script.

Uninstalling Postgres Enterprise Manager

The process of uninstalling the PEM server or agent is platform-specific. The name of the package for PEM server is `edb-pem-server` and for PEM agent is `edb-pem-agent`.

If you uninstall the PEM server package from a host, the PEM agent package installed on the same host doesn't get uninstalled. But if you uninstall the PEM agent package, then the PEM server package installed on the same host also gets uninstalled.

Uninstalling PEM from Windows hosts

If the PEM installation resides on a Windows host, you can use the Windows `Add/Remove Programs` application to remove PEM components. Select the `Add/Remove Programs` option from the Windows `Control Panel`. When the `control panel` opens, locate the name of the PEM component in the program list. Click the `Remove` button to remove the component.

You can also invoke the uninstaller that resides at the following location:

> `C:\\Program Files\edb\pem\server`

Uninstalling PEM from CentOS or RHEL hosts

You can use variations of the `rpm`, `yum remove`, or ‘‘ yum erase‘‘ commands to remove the installed packages. Note that removing a package does not damage the PEM data directory.

- Include the `-e` option when invoking the rpm command to remove an installed package; the command syntax is:

  `rpm -e <package_name>`

- You can use the `yum remove` command to remove the pem server or agent package installed by yum. To remove a package, open a terminal window, assume superuser privileges, and enter the command:

  ```
  yum remove <package_name>
  ```

- You can use the `yum erase` command to remove the pem server or agent package along with the `edb-pem` and `edb-pem-docs` dependencies. To remove a package, open a terminal window, assume superuser privileges, and enter the command:

  ```
  yum erase <package_name>
  ```

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

Uninstalling PEM from Debian or Ubuntu hosts

You can use `apt-get remove` or `apt-get purge` command to uninstall the PEM server or agent package from a Debian or Ubuntu host:

- To uninstall PEM server or agent from a Debian or Ubuntu host without impacting the configuration files and data directories, invoke the following command:

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

- To uninstall PEM server or agent along with the configuration files and data directory, invoke the following command:

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

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

Uninstalling PEM from SLES hosts

To uninstall PEM server or agent from a SLES host, invoke the following command:

```
zypper remove <package_name>
```

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

CHAPTER 7

Reference - Linux Service Script

The Postgres server on which the PEM server resides must contain a service script. Postgres installers generated by EnterpriseDB create a service script for you; if you are using a Postgres server from another source, you must provide a service script.

You can use the following example of a linux service script as a starting point when developing a script for a Postgres installation that was installed or built from a source that does not provide one. Please ensure (if you copy and paste from this example) that the line breaks are copied correctly.

```
| #!/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()
| {
|   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(${PIDALIVEDB})."
|                                         exit
|                         else
|                                         STARTDBSERVER=1
|                         fi
|                          else
|                         STARTDBSERVER=1
|                             fi
| if [ "${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
| 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 [ $? -eq 0 ];
| then
| echo "'${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 "${DB_DATA_DIR}/global" ]; then
| _die_incomplete_requirement "The directory - '${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}'"

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)
| _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
```

CHAPTER 8

Conclusion

EDB Postgres Enterprise Manager Installation Guide

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 designs, establishes coding best practices, reviews, and verifies input validation for the logon UI for EDB Postgres Enterprise Manager where present. EDB follows the same approach for additional input components, however the nature of the product may require that it accepts freeform SQL, WMI or other strings to be entered and submitted by trusted users for which limited validation is possible. In such cases it is not possible to prevent users from entering incorrect or otherwise dangerous inputs.
- EDB reserves the right to add features to products that accept freeform SQL, WMI or other potentially dangerous inputs from authenticated, trusted users in the future, but will ensure all such features are designed and tested to ensure they provide the minimum possible risk, and where possible, require superuser or equivalent privileges.
- EDB does not that warrant that we can or will anticipate all potential threats and therefore our process cannot fully guarantee that all potential vulnerabilities have been addressed or considered.

Index