



Postgres Enterprise Manager

Release 7.15

Installation Guide for Linux

Jul 30, 2020

Contents

1	What's New	2
2	Postgres Enterprise Manager - Overview	3
2.1	Hardware Prerequisites	5
2.2	Software Prerequisites	5
3	Installing Postgres Enterprise Manager	6
3.1	Installing the PEM Server on Linux Platforms	6
3.1.1	Prerequisites for installing the PEM Server on Linux Platforms	6
3.1.2	Installing PEM Server and Apache Web Server(PEM-HTTPD) preferences	7
3.1.3	Installing the PEM Server on a CentOS or RHEL Host	8
3.1.4	Installing the PEM Server on a Debian or Ubuntu Host	10
3.1.5	Installing the PEM Server on a SLES Host	11
3.2	Creating a Repository in an Isolated Network	12
3.3	Configuring the PEM Server on Linux Platforms	14
3.4	Installing a PEM Agent on Linux Platforms	16
3.4.1	Installing a PEM Agent on a CentOS or RHEL host	16
3.4.2	Installing a PEM Agent on a Debian or Ubuntu Host	18
3.4.3	Installing a PEM Agent on a SLES Host	19
3.5	Registering a PEM Agent	20
4	The PEM Web Interface	23
5	Uninstalling Postgres Enterprise Manager Components	26
5.1	Uninstalling PEM components from CentOS or RHEL Hosts	26
5.2	Uninstalling PEM components from Debian or Ubuntu hosts	27
5.3	Uninstalling PEM components from SLES hosts	27
6	Reference - Linux Service Script	28
7	Conclusion	29
	Index	30

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 on a Linux host.

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

The following features have been added to Postgres Enterprise Manager 7.15:

- **Enhanced EDB Backup and Recovery Tool management:** Support for two new configuration parameters `archive_path` and `bart_socket_directory` enhances PEM's integration with BART 2.5.4.
- **Row Security Policy:** You can now CREATE, UPDATE, DELETE, ENABLE, and disable row-level security policies for the database tables via the PEM Web Client.
- **Improved User Interface (UI) for Performance diagnostics tool:** You can now see the number of active sessions at a sample time, and can also see a session's wait event or CPU utilization at a specific sample time.
- **Improved the EDB Failover Manager (EFM) monitoring:** PEM now provides the capability to monitor the `xLogReceive` parameter set by EFM.
- Other features and changes include:
 - Added support to monitor PostgreSQL/EDB Postgres Advanced Server 13.
 - Added documentation explaining how the SNMP trap OID is generated by PEM.
 - Added support for schema level restriction.
 - Added RLS Policy support in the Schema Diff tool.

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, 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`) and the PEM Agent (`pemagent`). 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 the [PEM SQL Profiler User's Guide](#) for details and supported versions.

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

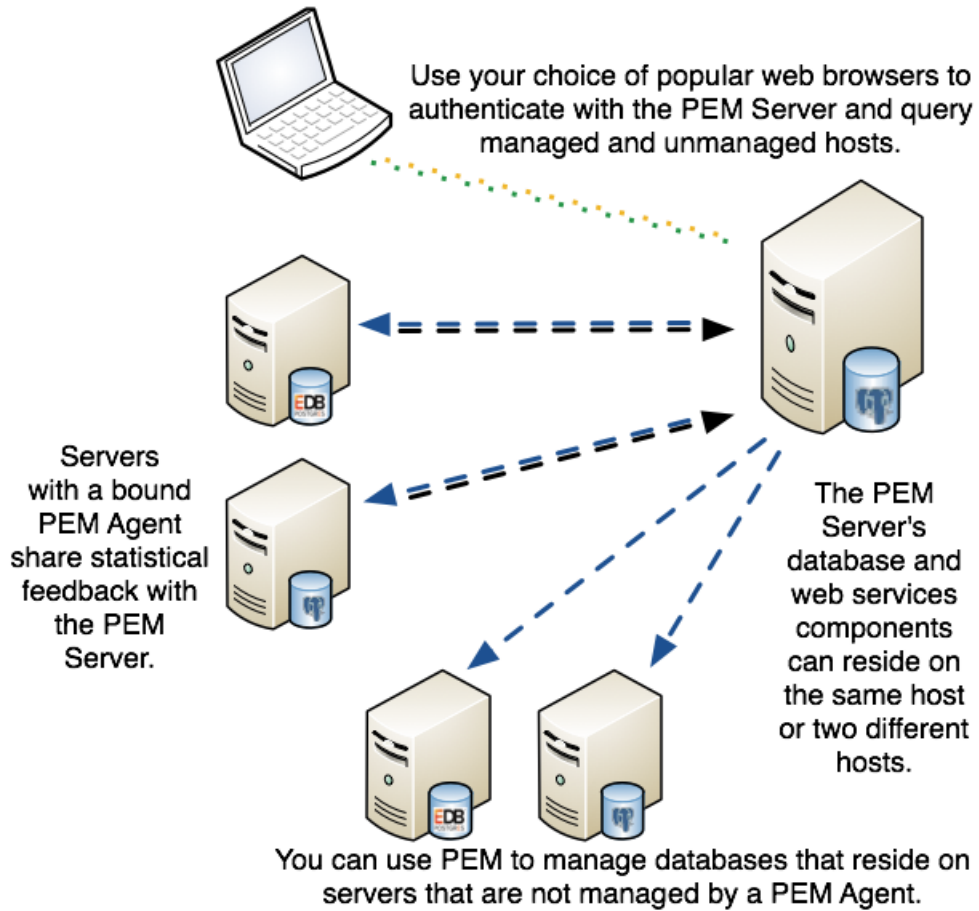


Fig. 1: A typical PEM installation

2.1 Hardware Prerequisites

For optimum performance 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

Platforms and Versions Support

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>

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/edb-docs>

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.

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.

Installing Postgres Enterprise Manager

The `edb-pem` package for Linux platforms installs the PEM Server, a PEM Agent, and the required software to connect to the PEM web interface with a supported browser.

The PEM server uses a Postgres installation and backing database to manage data. The `pem` backing database is created when you configure PEM.

For detailed information about installing the PEM Server, see *Installing the PEM Server on Linux*. For information about configuring a PEM Server see *Configuring the PEM Server on Linux*.

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 detailed information about remote monitoring functionality see the [PEM Agent Privileges](#).

For detailed information about installing and configuring a PEM Agent, see *Installing the PEM Agent on Linux*.

3.1 Installing the PEM Server on Linux Platforms

Before following the detailed instructions that install the PEM server on your specific platform, you must perform the prerequisite steps detailed in Section 3.1.1.

3.1.1 Prerequisites for installing the PEM Server on Linux Platforms

1. Install a backing database.

When installing a PEM server on a Linux host, you must first install a backing database cluster which will hold the `pem` database. The PEM server's backing database may be installed via package for Linux. The backing database must be one of the following versions:

- EDB Postgres Advanced Server version 9.5 or above
- PostgreSQL version 9.5 or above

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

2. Configure Postgres authentication on the backing database.

The `pg_hba.conf` file on the backing database can be configured to use any supported authentication methods (for example: `md5`, `trust`,...) for connections. For information about modifying the `pg_hba.conf` file, see the [PostgreSQL core documentation](#).

3. Install the `hstore` contrib module (PostgreSQL users only).

- If you are using a PostgreSQL database, use the following command to install the `hstore contrib` module:

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

Where, `x` is the server version.

4. Ensure that the `sslutils` extension is installed.

- On an Advanced Server backing database, the `sslutils` extension is installed by default.
- 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>
```

Where, `x` is the server version.

5. Adjust your firewall restrictions.

If you are using a firewall, you must allow access to port `8443` on the PEM backing database:

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

6. Request credentials that allow you to access the EnterpriseDB repositories:

To install the PEM Server, you must have credentials that allow access to the EnterpriseDB repository. To request credentials for the repository, contact [EnterpriseDB](#). When using commands in the sections that follow, replace the `username` and `password` placeholders with the credentials provided by EnterpriseDB.

3.1.2 Installing PEM Server and Apache Web Server(PEM-HTTPD) preferences

During the PEM server installation, you can specify your hosting preferences for the Apache Web Server(PEM-HTTPD):

To install the PEM Server and Apache Web Server(PEM-HTTPD) on the same host

Follow the installation steps; while running the configuration script, select the `Web Services` and `Database` option to install PEM Server and Apache Web Server on the same host.

To install the PEM Server and Apache Web Server(PEM-HTTPD) on separate hosts

Follow the installation steps on both the hosts. While running the configuration script, first configure PEM Server host by selecting the `Database` option on first host and then configure Apache Web Server (PEM-HTTPD) host by selecting the `Web Services` option on the second host.

For detailed information regarding Configuration of the PEM Server, see [Configuring the PEM Server on Linux Platforms](#).

3.1.3 Installing the PEM Server on a CentOS or RHEL Host

In addition to the above considerations, the following prerequisites are applicable if you are using a CentOS or RHEL 6.x, 7.x or 8.x host:

1. Install the `epel-release` package and prepare the host.

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

```
yum install epel-release

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

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, then you must also enable `rhel-<x>-server-eus-optional-rpms` repository to use EPEL packages or please see the Red Hat Subscription Management Guide for the required repository.
- You must also enable the `rhel-<x>-server-extras-rpms` repository, where `x` specifies the version of the RHEL on the host.

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

- Stop the default `httpd` server using the command:

```
service httpd stop
```

- Run the following commands to install `rh-python36`:

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

2. Install and configure the `edb.repo` file.

- a. To create a 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`.

- b. After creating the `edb.repo` file, invoke the following command to replace the `username` and `password` placeholders in the `baseurl` specification with the name and password of a registered EnterpriseDB user:

```
sed -i "s@<username>:<password>@USERNAME:PASSWORD@" /etc/yum.repos.d/edb.repo
```

3. Install the PEM Server.

- You can use `yum` to install the PEM server on CentOS or RHEL 6.x, 7.x or 8.x:

```
yum install edb-pem
```

- You can use `dnf` to install PEM Server on CentOS or RHEL 8.x:

```
dnf 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](#).

After installing PEM Server using `yum` or `dnf`, you need to configure the PEM Server. For more detailed information see [Configuring the PEM Server on Linux platforms](#).

3.1.4 Installing the PEM Server on a Debian or Ubuntu Host

The following steps will walk you through using the EnterpriseDB `apt` repository to install a Debian package.

Note: You can also visit <https://repos.enterprisedb.com/> and select the platform and product to view the steps for installation.

1. Log in as root on your Debian or Ubuntu host:

```
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'
```

Where `username:password` is to be replaced by the credentials provided by EnterpriseDB.

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
```

When the installation completes, you must configure the PEM Server. For detailed information see *Configuring the PEM Server on Linux Platforms*.

3.1.5 Installing the PEM Server on a SLES Host

Use the following command to add the EnterpriseDB repository configuration files to your SLES host:

```
zypper addrepo https://zypp.enterprisedb.com/suse/edb-sles.repo
```

The command creates a repository configuration file named `edb.repo` in the `/etc/zypp/repos.d` directory. Modify the repository configuration file, adding the username and password of a registered EnterpriseDB user.

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
```

Then, refresh the repository and install the PEM server:

```
zypper refresh

zypper install edb-pem
```

After installing PEM Server using `zypper`, you must configure the PEM Server. For detailed information see *Configuring the PEM Server on Linux Platforms*.

3.2 Creating a Repository in an Isolated Network

You can create a local repository to act as a host for 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. Use a system with Internet access to download all of the required dependencies:

```
yum install yum-plugin-downloadonly

mkdir /<pem_dir>

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

mkdir /<epel_dir>

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

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

2. Copy the /<pem_dir> and /<epel_dir> directories to a machine on the isolated network.
3. Create the repositories:

```
yum install createrepo

createrepo /<pem_dir>

createrepo /<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:///<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:///<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 the PEM server:

- To install a PEM server:

```
yum install edb-pem
```

- To upgrade a PEM server:

```
yum upgrade edb-pem
```

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

3.3 Configuring the PEM Server on Linux Platforms

Before configuring the PEM server, ensure that the `sslutils` extension and `hstore` contrib module is installed for your backing database.

- For an Advanced Server backing database, `sslutils` extension and `hstore` contrib module are by default installed along with Advanced Server.
- For a PostgreSQL backing database, ensure you have access to the PostgreSQL community repository, and then install `sslutils` extension and `hstore` contrib module using the command:

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

Where, `x` is the server version.

The PEM server package includes a script (`configure-pem-server.sh`) to help automate the configuration process for Linux Platform 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
<code>-acp</code>	Defines PEM Agent certificate path. The default is <code>/root/.pem</code> .
<code>-ci</code>	CIDR formatted network address range that Agents will connect to the server from, to be added to the server's <code>pg_hba.conf</code> file. For example, <code>192.168.1.0/24</code> . The default is <code>0.0.0.0/0</code> .
<code>-dbi</code>	The directory for the database server installation. For example, <code>/usr/edb/as12</code> for Advanced Server or <code>/usr/pgsql-12</code> for PostgreSQL.
<code>-ds</code>	The unit file name of the PEM database server. For Advanced Server, the default file name is <code>edb-as-12</code> ; for PostgreSQL, it is <code>postgresql-12</code> .
<code>-ho</code>	The host address of the PEM database server.
<code>-p</code>	The port number of the PEM database server.
<code>-ps</code>	The service name of the pemagent; the default value is <code>pemagent</code> .
<code>-sp</code>	The superuser password of the PEM database server. This value is required.
<code>-su</code>	The superuser name of the PEM database server.
<code>-t</code>	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.
<code>-un</code>	To unregister the PEM Server.

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 (PEM-HTTPD) and the backing database (PEM Server) 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. *I* To view script-related help, use the command:

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

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
```

3.4 Installing a PEM Agent on Linux Platforms

A PEM Agent may monitor one or more servers on one or more hosts. For comprehensive information about managing a PEM Agent, see the [PEM Agent User Guide](#).

3.4.1 Installing a PEM Agent on a CentOS or RHEL host

On a CentOS or RHEL system, you can use the `yum` package manager or `dnf` command to install a PEM Agent depending on the version of the operating system.

The following prerequisites are applicable if you are using a CentOS or RHEL 6.x, 7.x or 8.x host:

1. Install epel-release package

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

```
yum install epel-release

yum -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-<X>.
↪noarch.rpm
```

Where `X` is the OS version.

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, then you must also enable `rhel-<x>-server-eus-optional-rpms` repository to use EPEL packages or please see the [Red Hat Subscription Management Guide](#) for the required repository.
 - You must also enable the `rhel-<x>-server-extras-rpms` repository, where `x` specifies the version of the RHEL on the host.
-

2. Install and configure edb.repo

- a. You must also have credentials that allow access to the EnterpriseDB repository. To request credentials, visit:

[EnterpriseDB Repository Access Steps](#).

- b. Create a 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`.

- c. After creating the `edb.repo` file, to ensure that the value of the `enabled` parameter is `1`, and the `username` and `password` placeholders in the `baseurl` specification are replaced with the name and password of a registered EnterpriseDB user, run the following command:

```
sed -i "s@<username>:<password>@USERNAME:PASSWORD@" /etc/yum.repos.d/edb.repo
```

If you want to install PEM Agent 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](#).

Installing the PEM Agent

- Use the `yum install` command to install PEM Agent on CentOS or RHEL 6.x, 7.x or 8.x:

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

- Also, you can use `dnf` command to install PEM Agent on CentOS or RHEL 8.x:

```
dnf install edb-pem-agent
```

When the installation is complete, it 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. 1: 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.

After installing PEM Agent using `yum` or `dnf`, you need to register the PEM Agent. For more detailed information see [Registering a PEM Agent](#).

3.4.2 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. 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
```

After installing PEM Agent using `apt-get`, you need to register the PEM Agent. For more detailed information see [Registering a PEM Agent](#).

3.4.3 Installing a PEM Agent on a SLES Host

Use the following command to add the EnterpriseDB repository configuration files to your SLES host:

```
zypper addrepo https://zypp.enterprisedb.com/suse/edb-sles.repo
```

The command creates a repository configuration file named `edb.repo` in the `/etc/zypp/repos.d` directory. Modify the repository configuration file, adding the username and password of a registered EnterpriseDB user.

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
```

Then, you can refresh the repository and add a PEM agent:

```
zypper refresh

zypper install edb-pem-agent
```

After installing the PEM Agent, you must register the agent. For more detailed information see [Registering a PEM Agent](#).

3.5 Registering a PEM 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. You can use the `pemworker` utility to register the Agent if you use the package to install a PEM Agent.

The PEM Agent package 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
<code>--pem-server</code>	Specifies the IP address of the PEM backend database server. This parameter is required.
<code>--pem-port</code>	Specifies the port of the PEM backend database server. The default value is 5432.
<code>--pem-user</code>	Specifies the name of the Database user (having super-user privileges) of the PEM backend database server. This parameter is required.
<code>--pem-agent-user</code>	Specifies the Agent user to connect the PEM server backend database server.
<code>--cert-path</code>	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, <code>~/pem</code> On Windows, <code>%APPDATA%/pem</code>
<code>--config-dir</code>	Specifies the directory path where configuration file can be found. The default is the <code><pemworker path>/../etc</code> .
<code>--display-name</code>	Specifies a user-friendly name for the Agent that will be displayed in the PEM Browser tree control. The default is the system hostname.
<code>--force-registration</code>	Include the <code>force_registration</code> 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.
<code>--group</code>	The name of the group in which the Agent will be displayed.
<code>--team</code>	The name of the database role, on the PEM backend database server, that should have access to the monitored database server.
<code>--owner</code>	The name of the database user, on the PEM backend database server, who will own the Agent.
<code>--allow_server_restart</code>	Enable the <code>allow_server_restart</code> parameter to allow PEM to restart the monitored server. The default value is True.
<code>--allow_batch_probes</code>	Enable the <code>allow_batch_probes</code> parameter to allow PEM to run batch probes on this Agent. The default value is False.

Continued on next page

Table 1 – continued from previous page

Option	Description
<code>--batch-script-user</code>	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.
<code>--enable-heartbeat-connection</code>	Enable the <code>enable-heartbeat-connection</code> parameter to create a dedicated heartbeat connection between PEM Agent and server to update the active status. The default value is False.
<code>--enable-smtp</code>	When set to true for multiple PEM Agents (7.13 or lesser) and PEM backend database (9.4 or lesser) then it may send more duplicate emails. Whereas for PEM Agents (7.14 or higher) and PEM backend database (9.5 or higher) then it may send lesser duplicate emails.
<code>--enable-snmp</code>	When set to true for multiple PEM Agents (7.13 or lesser) and PEM backend database (9.4 or lesser) then it may send more duplicate traps. Whereas for PEM Agents (7.14 or higher) and PEM backend database (9.5 or higher) then it may send lesser duplicate traps.
<code>-o</code>	Specify if you want to override the configuration file options.

If you want to use any PEM feature for which a database server `restart` is required by the `pemagent` (such as Audit Manager, Log Manager, or the Tuning Wizard), then you must set the value for `allow_server_restart` to `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 native package 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`). Modify the `agent.cfg` file, changing the following configuration parameter value to `true`:

```
heartbeat_connection = true
```

By default, `heartbeat_connection` value is `false` but you can override the value during pemagent registration with `pemworker` utility using the `--enable-heartbeat-connection` option.

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 or 8.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](#)

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 PostgreSQL Enterprise Manager Web Login window opens:

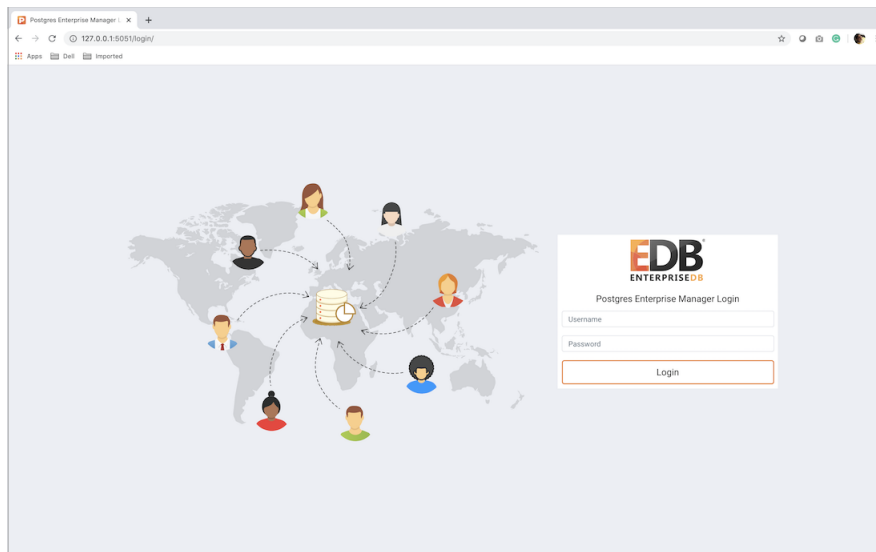


Fig. 1: *The PEM Web Login page*

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.

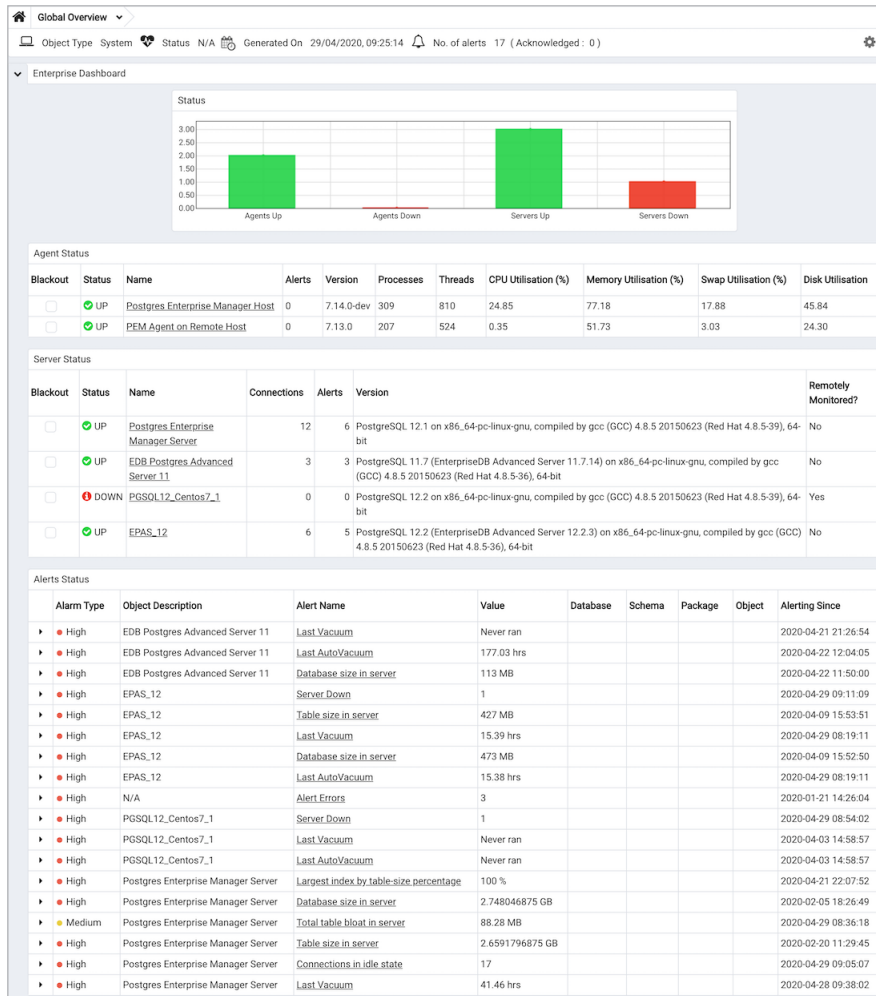


Fig. 2: The Global Overview Dashboard

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 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 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`, `.epub`, and `.html` format from the [EnterpriseDB website](#)

Uninstalling Postgres Enterprise Manager Components

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

5.1 Uninstalling PEM components from CentOS or RHEL Hosts

You can use variations of the `rpm`, `yum remove`, or `yum erase`, commands to remove the installed packages from CentOS or RHEL 6.x, 7.x or 8.x hosts. Also you use `dnf remove` command to remove the installed package from CentOS or RHEL 8.x host. 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.

- You can use `dnf remove` command to remove the pem server or Agent along with the `edb-pem` and `edb-pem-docs` dependencies on CentOS or RHEL 8.x hosts. To remove a package, open a terminal window, assume superuser privileges, and enter the command:

```
dnf remove package_name
```

5.2 Uninstalling PEM components 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.

5.3 Uninstalling PEM components 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.

Reference - Linux Service Script

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

On **CentOS or RHEL 6.x**, the service script resides in the `/etc/init.d` directory.

On **CentOS or RHEL 7.x or 8.x**, the service script resides in the `/usr/lib/systemd/system` directory.

- Service scripts are platform-specific.
- For information about customizing a Postgres service, visit:

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

EDB Postgres Enterprise Manager Installation Guide for Linux

Copyright © 2013 - 2020 EnterpriseDB Corporation. All rights reserved.

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

- EnterpriseDB and Postgres Enterprise Manager are registered trademarks of EnterpriseDB Corporation. EDB and EDB Postgres are trademarks of EnterpriseDB Corporation. Oracle is a registered trademark of Oracle, Inc. Other trademarks may be trademarks of their respective owners.
- 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 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.

C

Conclusion, 29
configuring the PEM Server on Linux, 14
Creating a Repository in an Isolated
Network, 12

H

Hardware Prerequisites, 5

I

installing agent on Linux, 16
installing PEM, 6
Installing PEM Agent on a SLES Host, 19
installing PEM server on Linux, 6
Installing Postgres Enterprise
Manager, 6

P

Postgres Enterprise Manager -
Overview, 3

R

Reference - Linux Service Script, 28
Registering a PEM Agent, 20

S

Software Pre-Requisites, 5

T

The PEM Web Interface, 23

U

Uninstalling Postgres Enterprise
Manager Components, 26

W

What's New, 2