



Postgres Enterprise Manager

リリース 7.16

Agent User Guide

2020 年 10 月 10 日

目次

第 1 章	PEMArchitecture	2
1.1	PEM サーバー	4
1.2	PEM エージェント	6
1.3	PEMWeb クライアント	7
1.4	SQL プロファイラープラグイン	7
第 2 章	エージェントの登録	9
2.1	PEM エージェント構成パラメーターの設定	12
2.2	非ルートユーザーアカウントを使用して PEM エージェントを登録する	13
第 3 章	PEM エージェントの管理	15
3.1	エージェント権限	15
3.2	エージェント構成	18
3.3	エージェントのプロパティ	23
第 4 章	PEM エージェントのトラブルシューティング	26
4.1	削除された PEM エージェントの復元	26
4.2	コマンドラインを使用して、ダウンまたは不明な状態の PEM エージェントを削除する	27
第 5 章	結論	28
	索引	29

PEM は、PEM サーバー、PEM エージェント、および PEMWeb インターフェ이스の 3 つの主要コンポーネントで構成されています。PEM エージェントは、各管理対象マシンでタスクを実行し、データベースサーバーとオペレーティングシステムの統計を収集します。

PEM がサポートするプラットフォームとバージョンの詳細については、次の EDBWeb サイトにアクセスしてください。

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

PEM エージェントのインストール、アンインストール、またはアップグレードの詳細については、次の EDBWeb サイトにアクセスしてください。

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

このドキュメントでは、PEM エージェントを使用するために必要な情報を提供します。このガイドでは、エージェントの基本的な登録、構成、および管理について説明します。このガイドは、次のコアセクションに分かれています。

- **PostgresEnterpriseManager-概要**-このセクションでは、PEM アーキテクチャの概要を説明します。
- **PEM エージェントの登録**-このセクションでは、PEM エージェントの登録に関する情報を提供します。
- **PEM エージェントの管理**-このセクションでは、PEM エージェントの構成と管理に関する情報を提供します。
- **PEM エージェントのトラブルシューティング**-このセクションでは、PEM エージェントのトラブルシューティングに関する情報を提供します。

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

PEMArchitecture

PostgresEnterpriseManager (PEM) は、単一の GUI インターフェイスを介して複数の Postgres サーバーを監視および管理するために設計されたツールです。PEM は、インフラストラクチャの次の領域を監視できます。

注:Postgres という用語は、PostgreSQL または EDBPostgresAdvancedServer を指します。

- ホスト-1 つまたは複数のサーバー（物理または仮想）およびそれらのオペレーティングシステム。
- サーバー-ホストで実行されている PostgreSQL または EDBPostgresAdvancedServer の 1 つ以上のインスタンス。
- データベース-1 つまたは複数のデータベースと、その中のスキーマオブジェクト（テーブル、インデックスなど）。

PEM は、多数の個々のソフトウェアコンポーネントで構成されています。個々のコンポーネントについて以下に説明します。

- **PEM サーバー**-PEM サーバーは、データを監視するためのデータリポジトリとして、およびエージェントとクライアントの両方が接続するサーバーとして使用されます。PEM サーバーは、PostgreSQL のインスタンスと、監視データを保存するための関連データベース、および Web サービスを提供するサーバーで構成されます。
- **PEM エージェント**-PEM エージェントは、タスクを実行し、エージェントホストおよび監視対象の Postgres インスタンスから PEM サーバーに統計を報告します。単一の PEM エージェントは、1 つ以上のホストに存在する Postgres の複数のインストール済みインスタンスを監視できます。
- **PEMWeb クライアント**-PEMWeb インターフェイスを使用すると、Postgres サーバーを管理および監視し、PEM 拡張機能を利用できます。Web インターフェイスソフトウェアは PEM サーバーとともにインストールされ、サポートされている Web ブラウザーを介してアクセスされます。

- SQLProfiler**-SQLProfiler は、PEM の SQLProfiler ツールで分析される監視データとクエリプランを記録する Postgres サーバープラグインです。これは PEM のオプションコンポーネントですが、SQL プロファイラツールを使用する Postgres の各インスタンスにプラグインをインストールする必要があります。SQL プロファイラは、PostgreSQL サーバーまたは AdvancedServer（PEM サーバーで管理されているものだけでなく）の EDB ディストリビューションのサポートされているバージョンで使用できます。詳細およびサポートされているバージョンについては、[“PEMSQL プロファイラ構成ガイド”](https://www.enterprisedb.com/edb-docs/p/edb-postgres-enterprise-manager) <<https://www.enterprisedb.com/edb-docs/p/edb-postgres-enterprise-manager>>‘_’を参照してください。

PEM アーキテクチャ

次のアーキテクチャ図は、PEM サーバー、クライアント、管理対象および非管理対象の Postgres サーバー間の関係を示しています。

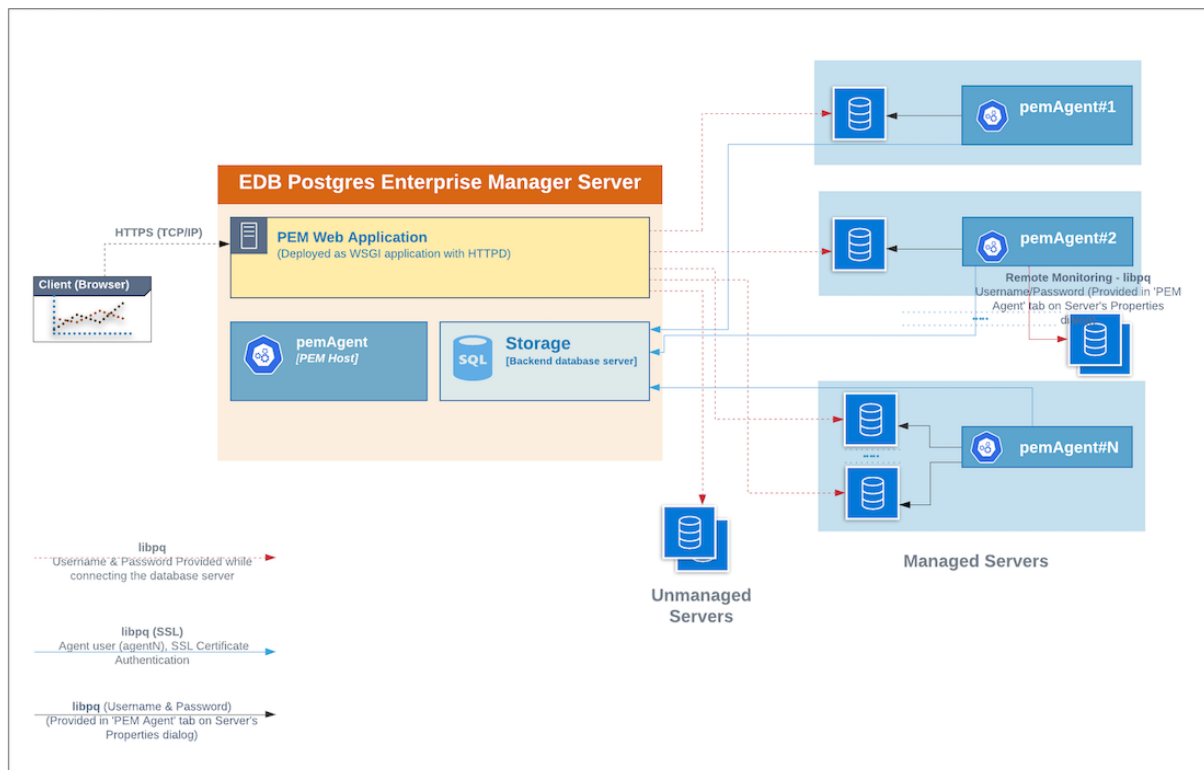


図 1 PEM アーキテクチャ

1.1 PEM サーバー

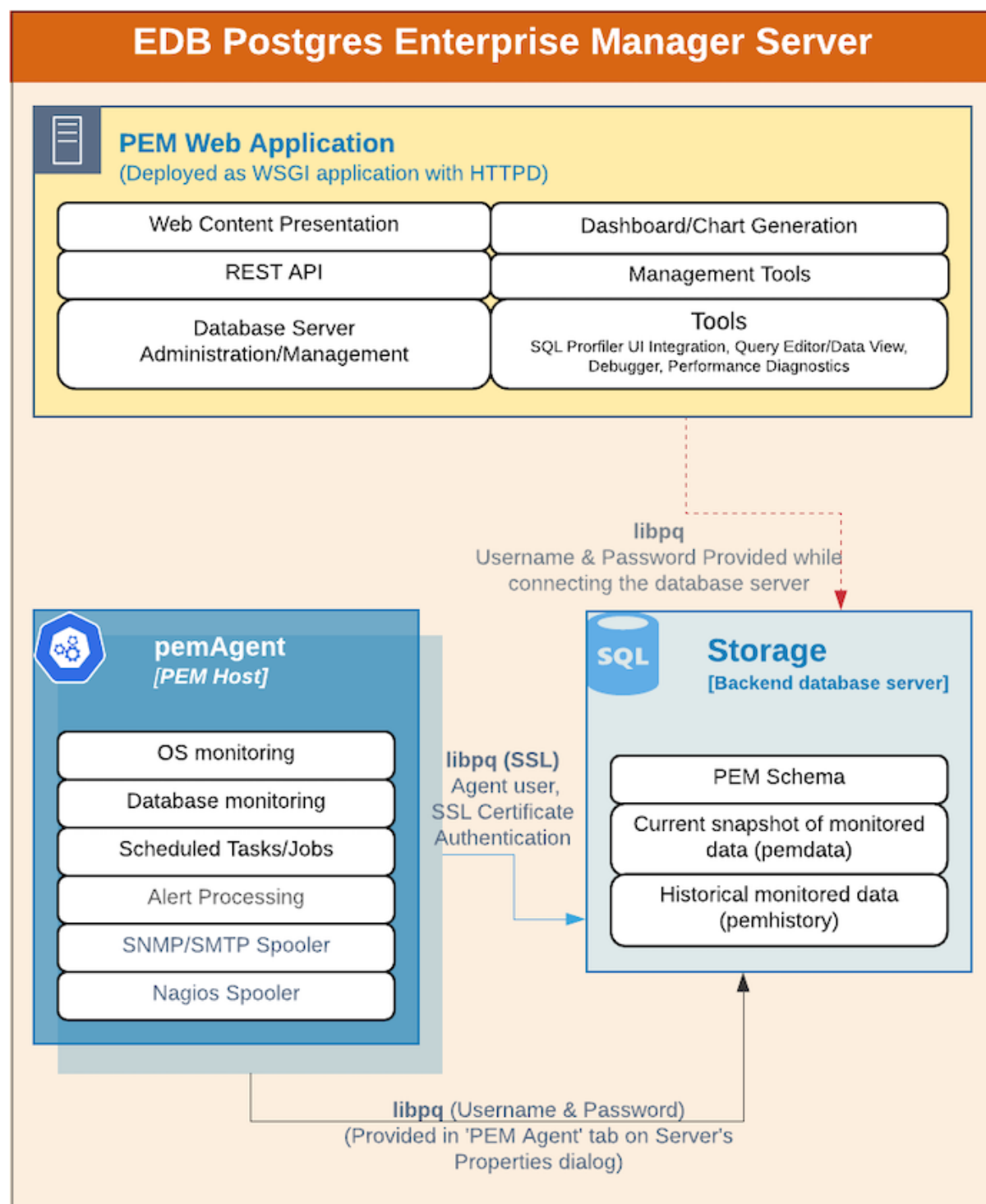


図2 PEM サーバー

PEM サーバーは、Postgres のインスタンス、クライアントに Web サービスを提供する ApacheWeb サーバーのインスタンス、および PEM エージェントで構成されます。PEM は、サーバー側の暗号化プラグインを使用して認証証明書を生成します。

Postgres（データベースサーバー）のインスタンスと ApacheWeb サーバー（HTTPD）のインスタンスは、同じホスト上にあっても、別々のホスト上にあってもかまいません。

- **Postgres インスタンス（データベースサーバー）**-これはバックエンドデータベースサーバーです。PEM サーバーのリポジトリとして機能する**pem**という名前のデータベースをホストします。****pem**** データベースには、監視対象の各ホスト、サーバー、およびデータベースから収集されたメトリックデータを格納するいくつかのスキーマが含まれています。
 - **pem**-このスキーマは PEM アプリケーションの中核です。これには、アプリケーションに必要な構成関数、テーブル、またはビューの定義が含まれています。
 - **pemdata**-このスキーマには、監視対象データの現在のスナップショットが格納されます。
 - **pemhistory**-このスキーマには、監視対象の履歴データが格納されます。
- **ApacheWeb サーバー（HTTPD）**-PEMWeb アプリケーションは、HTTPD を使用して WSGI アプリケーションとしてデプロイされ、クライアントに Web サービスを提供します。次のもので構成されます。
 - **Web コンテンツプレゼンテーション**-プレゼンテーションレイヤーは、Web アプリケーション（ブラウザ、ログインページなど）によって作成されます。
 - **RESTAPI**-RESTAPI を使用すると、他のアプリやサービスと統合できます。
 - **データベースサーバーの管理/管理**-CREATE、ALTER、DROP などのデータベースサーバーの管理および管理アクティビティは、管理対象サーバーと非管理対象サーバーの両方で実行できます。
 - **ダッシュボード/チャート生成**-内部的に、Web アプリケーションにはダッシュボードとチャートを生成する機能が含まれています。
 - **管理ツール**-AuditManager、CapacityManager、LogManager、PostgresExpert、PostgresLogAnalysisExpert、TuningWizard が Web アプリケーションで利用可能になりました。
 - その他のツールは、管理対象サーバーまたは非管理対象サーバーで機能を提供します。
 - * **SQLProfilerUI の統合**-SQLProfiler は、セッションコンテンツの分析が容易なトレースを生成します。
 - * **クエリエディタ/データビュー**-クエリエディタを使用すると、データをクエリ、編集、および表示できます。
 - * **デバッガ**-デバッガはクエリのデバッグに役立ちます。
 - * **パフォーマンス診断**-パフォーマンス診断は、AdvancedServer のパフォーマンス分析に役立ちます。

PEM バックエンドデータベースの実稼働インスタンスをホストするには、専用のマシンを使用することをお勧めします。ホストは、監視されているデータベースサーバーの数とサーバーが処理しているワークロードに応じて、高レベルのデータスループットの影響を受ける場合があります。

1.2 PEM エージェント

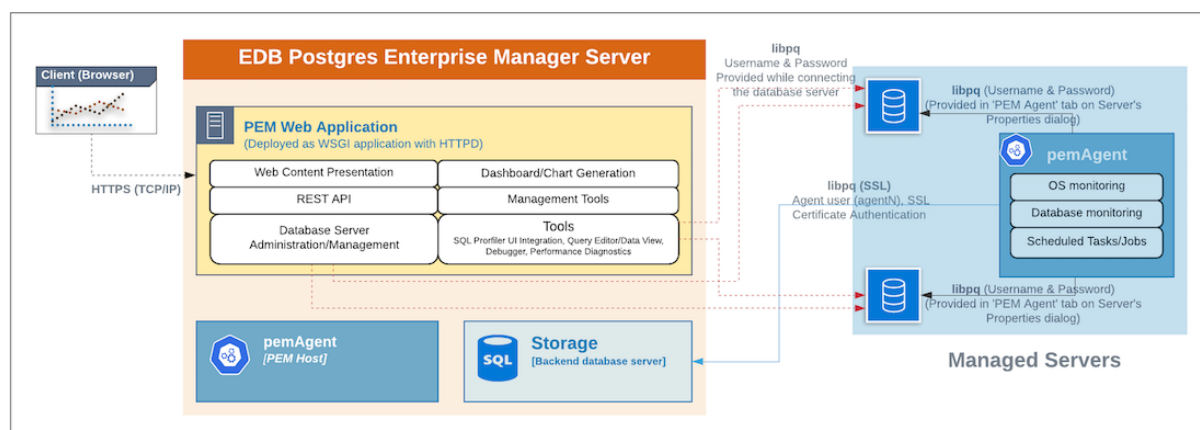


図 3 PEM エージェント

PEM エージェントは、マシンとオペレーティングシステム、およびそれらがバインドされている各 Postgres インスタンスからの監視データの収集を担当します。各 PEM エージェントは、1 台の物理マシンまたは仮想マシンを監視でき、複数のデータベースサーバーをローカルに（同じシステムにインストールされているか、リモートに）他のシステムにインストールされているかを監視できます。また、ユーザーがスケジュールする他のタスク（サーバーのシャットダウン、SQL プロファイラートレース、ユーザー定義ジョブなど）の実行も担当します。

PEM エージェントは、PEM サーバーのインストールとともに PEM サーバーにデフォルトでインストールされます。通常、PEM ホスト上の PEM エージェントと呼ばれます。別途、PEM を使用して監視する Postgres インスタンスをホストする他のサーバーに PEM エージェントをインストールすることもできます。

ローカルで監視するかリモートで監視するかにかかわらず、PEM エージェントは、SSL 証明書ベースの認証を使用して、PostgreSQL の libpq を使用して PEM サーバーに接続します。Windows の PEM エージェントインストーラーと Linux の pemworkerCLI は、各エージェントを PEM サーバーに登録し、必要な証明書を生成およびインストールします。

PEM エージェントと PEM サーバーの間には一方向のトラフィックしかありません。PEM エージェントは常に PEM サーバーに接続します。

PEM エージェントは、監視する各データベースサーバーに接続する必要があります。この接続は TCP/IP 接続（またはオプションで Unix ホストの Unix ドメインソケット）を介して行われ、オプションで SSL を使用できます。ユーザーは、監視対象サーバーへの接続と認証を構成する必要があります。

構成が完了すると、各エージェントは、ホストと、監視対象の各データベースサーバーとデータベースに関する統計情報やその他の情報を収集します。各情報は**メトリック**と呼ばれ、**プローブ**によって収集されます。ほとんどのプローブは、効率のために一度に複数のメトリックを収集します。収集されるメトリックの例は次のとおりです。

- ディスク I/O 統計
- ネットワーク統計

- データベースサーバーのバージョン文字列
- データベースサーバー構成オプション（GUC）値
- テーブルアクセス統計
- テーブルとインデックスのサイズ

PEM プローブのリストは ‘こちら<https://www.enterprisedb.com/edb-docs/d/edb-postgres-enterprise-manager/reference/online-documentation/latest/pem_probes.html#pem-probes> にあります。>’_。

デフォルトでは、データベースサーバーにバインドされた PEM エージェントは OS/データベース監視統計を収集し、その特定のデータベースサーバーのスケジュールされたタスク/ジョブも実行し、PEM サーバーの pem データベースにデータを保存します。

アラート処理、SNMP/SMTP スプーラー、および Nagios スプーラーデータは、PEM サーバー上の pem データベースに保存され、デフォルトで PEM ホスト上の PEM エージェントによって処理されます。ただし、PEM エージェントの SNMP/SMTP および Nagios パラメーターを調整することにより、他の PEM エージェントによる処理を有効にすることができます。

これらのパラメーターの詳細については、**“サーバー構成**<https://www.enterprisedb.com/edb-docs/d/edb-postgres-enterprise-manager/reference/online-documentation/latest/pem_config_options.html#pem-**“を参照してください。config-options>’_。**

1.3 PEMWeb クライアント

PEM クライアントは、サポートされているブラウザで実行される Web ベースのアプリケーションです。クライアントの Web インターフェイスは PEM サーバーに接続し、管理対象サーバーまたは管理対象外サーバー、およびそれらに存在するデータベースとスキーマを直接管理できます。

クライアントでは、ダッシュボード、PostgresLogAnalysisExpert、CapacityManager などの機能を介してサーバーに記録されたデータを利用する PEM 機能を使用できます。

1.4 SQL プロファイラープラグイン

すべてのサーバーに SQL プロファイラープラグインをインストールする必要はありませんが、SQL プロファイラーを使用する各サーバーにプラグインをインストールして構成する必要があります。監視されていない開発サーバーに SQL プロファイラーをインストールして構成することもできます。アドホック使用の場合も、SQLProfiler プラグインを一時的にインストールできます。

プラグインは EDBPostgresAdvancedServer ディストリビューションとともにインストールされますが、PostgreSQL で使用するには別途インストールする必要があります。SQLProfiler インストーラーは、EDBWeb サイト<<http://www.enterprisedb.com/download-postgres-enterprise-manager>>’_から入手できます。

SQLProfiler は、PEM で管理されていないサーバーで使用できますが、スケジュールされたトレースを実行するには、サーバーにプラグインをインストールし、インストールおよび構成済みの PEM エージェントで管理する必要があります。

SQL プロファイラーの使用の詳細については、“PEMSQL プロファイラー構成ガイド<<https://www.enterprisedb.com/edb-docs/p/edb-postgres-enterprise-manager>>__”を参照してください。

第 2

エージェントの登録

各 PEM エージェントは、PEM サーバーに*登録*する必要があります。登録プロセスは、エージェントと通信するために必要な情報を PEM サーバーに提供します。Windows 用の PEM エージェントグラフィカルインストーラーは、エージェントの自己登録をサポートしています。エージェントが Linux ホスト上にある場合、エージェントを登録するには pemworker ユーティリティを使用する必要があります。

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

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

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

オプション	説明
--pem-server	PEM バックエンドデータベースサーバーの IP アドレスを指定します。このパラメーターは必須です。
--pem-port	PEM バックエンドデータベースサーバーのポートを指定します。デフォルト値は 5432 です。
--pem-user	PEM バックエンドデータベースサーバーのデータベースユーザー（スーパーユーザー特権を持つ）の名前を指定します。このパラメーターは必須です。
--pem-agent-user	PEM サーバーバックエンドデータベースサーバーに接続するエージェントユーザーを指定します。

次のページに続く

表 1 – 前のページからの続き

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

次のページに続く

表 1 – 前のページからの続き

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

データベースサーバーの再起動が必要な PEM 機能（監査マネージャー、ログマネージャー、チューニングウィザードなど）を使用する前に、最初に `allow_server_restart` の値を `true` に設定する必要があります `agent.cfg` ファイル。

注釈: PEM7.11 以降のバージョンがインストールされている PEM エージェントによって実行されるシェル/バッチスクリプトを構成する場合、`batch_script_user` パラメーターのユーザーを指定する必要があります。スクリプトの実行には、非 `root` ユーザーを使用することを強くお勧めします。`root` ユーザーを使用すると、データのセキュリティとオペレーティングシステムのセキュリティが低下する可能性があります。ただし、`root` ユーザーを使用してスクリプトを実行して `pemagent` を元の設定に復元する場合は、`batch_script_user` パラメーター値を `root` に設定する必要があります。

`PEM_SERVER_PASSWORD` 環境変数を使用して、PEM 管理ユーザーのパスワードを設定できます。`PEM_SERVER_PASSWORD` が設定されていない場合、サーバーは PEM データベースサーバーへの接続時に `PGPASSWORD` または `.pgpass` ファイルを使用します。

パスワードを提供しないと、パスワード認証エラーが発生します。他の必須だが省略された情報の入力を求められます。登録が完了すると、サーバーはエージェントが正常に登録されたことを確認します。

2.1 PEM エージェント構成パラメーターの設定

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
```

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

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

次に、プラットフォーム固有のコマンドを使用して、PEM エージェントサービスを開始します。サービスの名前は `pemagent` です。たとえば、CentOS または RHEL6.x システムでは、次のコマンドを使用します。

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

RHEL または CentOS7.x または 8.x ホストで、`systemctl` を使用してサービスを開始します。

```
systemctl start pemagent
```

サービスは、エージェントを開始していることを確認します。エージェントが登録および起動されると、Global Overview ダッシュボードおよび PEMWeb インターフェイスのオブジェクトブラウザツリーコントロールに表示されます。

`pemworker` ユーティリティを使用してサーバーを登録する方法については、次の Web サイトで入手可能な*PEM 管理者ガイド*を参照してください。

<https://www.enterprisedb.com/edb-docs>

2.2 非ルートユーザーアカウントを使用して PEM エージェントを登録する

非ルートユーザーアカウントを使用して PEM エージェントを登録するには、最初に PEM エージェントをルートユーザーとしてインストールする必要があります。インストール後、非ルートユーザーの ID（たとえば、edb）を想定して、次の手順を実行します。

1. .pem ディレクトリと logs ディレクトリを作成し、ファイルに読み取り、書き込み、実行のアクセス許可を割り当てます。

```
mkdir /home/<edb>/.pem
mkdir /home/<edb>/.pem/logs
chmod 700 /home/<edb>/.pem
chmod 700 /home/<edb>/.pem/logs
```

2. エージェントを PEM サーバーに登録します。

```
./pemworker --register-agent --pem-server <172.19.11.230> --pem-user
<postgres> --pem-port <5432> --display-name <non_root> --cert-path /home/
<edb> --config-dir /home/<edb>

The above command creates agent certificates and an agent configuration file
(`agent.cfg`) in the ``/home/edb/.pem`` directory. Use the following
command to assign read and write permissions to these files:

``chmod -R 600 /home/edb/.pem/agent*``
```

3. agent.cfg ファイルのパラメーターを変更します:

```
agent_ssl_key=/home/edb/.pem/agent<id>.key
agent_ssl_cert=/home/edb/.pem/agent<id>.crt
log_location=/home/edb/.pem/worker.log
agent_log_location=/home/edb/.pem/agent.log
```

4. 構成ファイルのパスと pemagent サービスファイルのユーザーの値を更新します。

- RHEL または CentOS6 を使用している場合は、pemagent サービスファイルを更新して agent.cfg ファイルへの正しいパスを反映し、ユーザーを su から su edb に変更します。
- RHEL または CentOS7 または 8 を使用している場合は、次のようにパラメーターを更新します。

```
User=edb
ExecStart=/usr/edb/pem/agent/bin/pemagent -c /home/edb/.pem/agent.cfg
```

5. エージェントプロセスを停止してから、非 root ユーザーを使用してエージェントサービスを再起動します。

- RHEL または CentOS6 を使用している場合、`sudo /etc/init.d/pemagent start/stop/restart`
- RHEL または CentOS7 または 8 を使用している場合、`sudo systemctl start/stop/restart pemagent`

6. PEM ダッシュボードでエージェントのステータスを確認します。

第 3

PEM エージェントの管理

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

3.1 エージェント権限

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

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

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

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

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

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

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

プローブが不十分な権限でオペレーティングシステムを照会している場合、プローブは `permission denied` エラーを返すことがあります。

プローブが不十分な権限でデータベースをクエリしている場合、プローブは `permission denied` エラーを返すか、返されたデータを空の値として **PEM** チャートまたはグラフに表示します。

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

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

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

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

3.2 エージェント構成

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

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

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

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

次のページに続く

表 1 – 前のページからの続き

パラメータ名	説明	デフォルト値
long_wait	最初の接続試行が失敗した場合に、PEM エージェントが PEM サーバーへの接続を試行する前に待機する最大時間 (秒)。	30 秒
short_wait	PEM エージェントがキュー内に次にある (実行を待機している) プロブをチェックする前に待機する最小時間 (秒単位)。	10 秒
alert_threads	エージェントによって生成されるアラートスレッドの数。	PEM サーバーのホストに常駐するエージェントに対して 1 に設定します。他のすべてのエージェントの場合は 0。
enable_smtp	複数の PEM エージェント (7.13 以下) および PEM バックエンドデータベース (9.4 以下) に対して true に設定すると、より多くの重複メールを送信する場合があります。一方、PEM エージェント (7.14 以降) および PEM バックエンドデータベース (9.5 以降) の場合、送信する重複メールの数は少なくなります。	PEM サーバーホストの場合は true。他のすべての場合は false。
enable_snmp	複数の PEM エージェント (7.13 以下) および PEM バックエンドデータベース (9.4 以下) に対して true に設定すると、より多くの重複トラップが送信される場合があります。一方、PEM エージェント (7.14 以降) および PEM バックエンドデータベース (9.5 以降) の場合、送信する重複トラップが少なくなる場合があります。	PEM サーバーホストの場合は true。他のすべての場合は false。
enable_nagios	true に設定すると、Nagios アラートが有効になります。	PEM サーバーホストの場合は true。他のすべての場合は false。
connect_timeout	エージェントが接続を待機する最大時間 (秒) (10 進整数文字列)。	デフォルトでは設定されていません。0 に設定すると、エージェントは無期限に待機する必要があります。

次のページに続く

表 1 – 前のページからの続き

パラメータ名	説明	デフォルト値
allow_server_restart	TRUE に設定すると、エージェントは、監視するデータベースサーバーを再起動できます。このパラメータの値に応じて、一部の PEM 機能が有効/無効になる場合があります。	偽
max_connections	接続スロットルで使用されるプローブ接続の最大数。	0（無制限の数）
connection_lifetime	ConnectionLifetime（または connection_lifetime）を使用して、開いているがアイドル状態の接続が保持される最小秒数を指定します。MaxConnections で指定された値に達し、待機中の要求を満たすために（別のデータベースへの）新しい接続が必要な場合、このパラメーターは無視されます。	デフォルトでは、0 に設定されます（接続は、エージェントの処理ループの後にアイドル状態になると切断されます）。
allow_batch_probes	TRUE に設定すると、ユーザーはカスタムプローブ機能を使用してバッチプローブを作成できます。	偽
heartbeat_connection	TRUE に設定すると、ハートビートの送信に専用の接続が使用されます。	偽
batch_script_dir	スクリプトファイル（アラート用）が保存されるパスを指定します。	/tmp
connection_custom_setup	監視対象サーバーとの新しい接続が確立されたときに呼び出される SQL コードを提供するために使用します。	デフォルトでは設定されていません。
ca_file	CA 証明書が存在するパスを指定します。	デフォルトでは設定されていません。
batch_script_user	バッチ/シェルスクリプトの実行に使用するユーザーの名前を指定します。	なし

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

HKEY_LOCAL_MACHINE/Software/Wow6432Node/EnterpriseDB/PEM/agent

レジストリには次のエントリが含まれます。

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

次のページに続く

表 2 – 前のページからの続き

AllowServerRestart	TRUE に設定すると、エージェントは、監視するデータベースサーバーを再起動できます。このパラメータの値に応じて、一部の PEM 機能が有効/無効になる場合があります。	本当
MaxConnections	接続スロットルで使用するブローブ接続の最大数。	0（無制限の数）
ConnectionLifetime	ConnectionLifetime（または connection_lifetime）を使用して、開いているがアイドル状態の接続が保持される最小秒数を指定します。MaxConnections で指定された値に達し、待機中の要求を満たすために（別のデータベースへの）新しい接続が必要な場合、このパラメーターは無視されます。	デフォルトでは、0 に設定されません（接続は、エージェントの処理ループの後にアイドル状態になると切断されます）。
AllowBatchProbes	TRUE に設定すると、ユーザーはカスタムブローブ機能を使用してバッチブローブを作成できます。	偽
HeartbeatConnection	TRUE に設定すると、ハートビートの送信に専用の接続が使用されます。	偽
BatchScriptDir	スクリプトファイル（アラート用）が保存されるパスを指定します。	/tmp
ConnectionCustomSetup	監視対象サーバーとの新しい接続が確立されたときに呼び出される SQL コードを提供するために使用します。	デフォルトでは設定されていません。
ca_file	CA 証明書が存在するパスを指定します。	デフォルトでは設定されていません。
AllowBatchJobSteps	true に設定すると、バッチ/シェルスクリプトは管理者ユーザーアカウントを使用して実行されます。	なし

3.3 エージェントのプロパティ

The PEM Agent Properties dialog provides information about the PEM agent from which the dialog was opened; to open the dialog, right-click on an agent name in the PEM client tree control, and select Properties from the context menu.

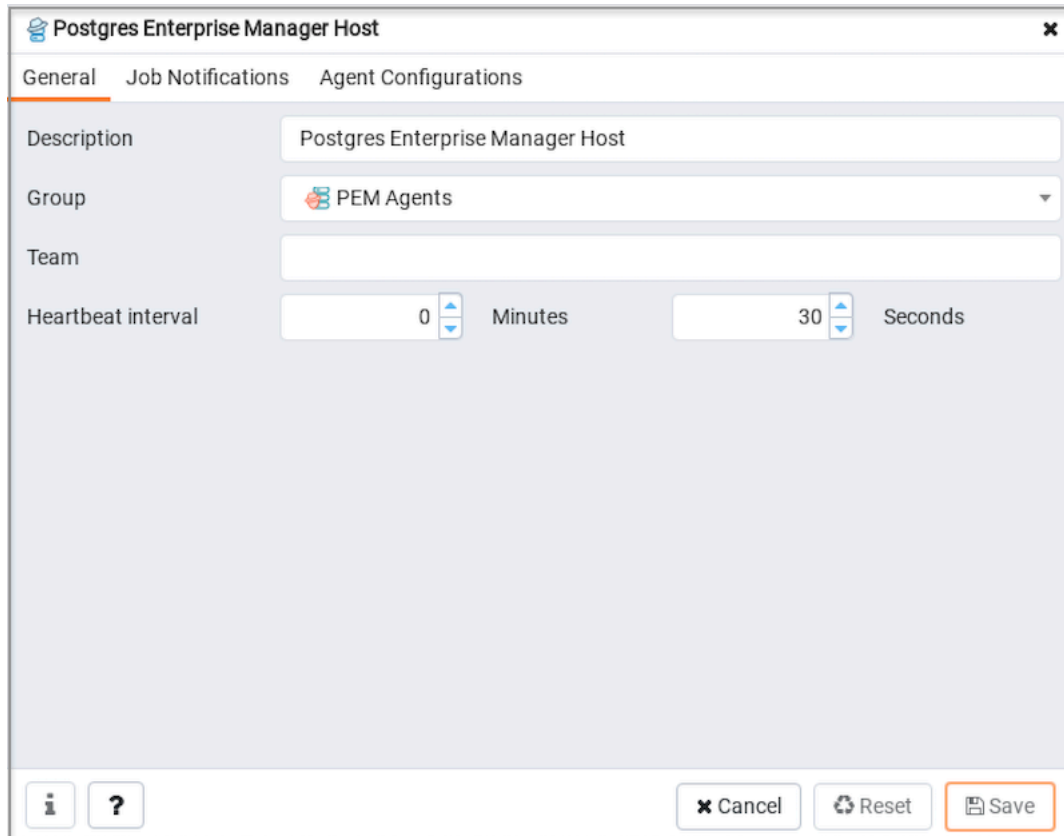


図 1 PEM エージェントプロパティダイアログ-[全般] タブ

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

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

時間を表示します。間隔を変更するには、Minutes または Seconds フィールドの横にあるセレクトターを使用します。

The screenshot shows the 'Postgres Enterprise Manager Host' dialog box with the 'Job Notifications' tab selected. The settings are as follows:

- Override default configuration?**: A toggle switch set to 'No'. Description: 'Select to override the default configuration for job notifications. If selected, the following settings will determine whether, when, and which email group will receive the job notification for this agent.'
- Email on job completion?**: A toggle switch set to 'No'. Description: 'Select to receive a notification email on completion of a job (regardless of the result) of this agent.'
- Email on a job failure?**: A toggle switch set to 'No'. Description: 'Select to receive a notification email only on failure of a job of this agent.'
- Email group**: A dropdown menu set to '<Default>'. Description: 'Select the email-group that will receive the notification on completion of a job or scheduled task.'

At the bottom of the dialog are three buttons: 'Cancel', 'Reset', and 'Save'.

図 2 PEM エージェントプロパティダイアログ-ジョブ通知タブ

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

- Use the `Override default configuration?` switch to specify if you want the agent level job notification settings to override the default job notification settings. If you select `Yes` for this switch, you can use the rest of the settings on this dialog to define when and to whom the job notifications should be sent. Please note that the rest of the settings on this dialog work only if you enable the `Override default configuration?` switch.
- `Email on job completion?` スイッチを使用して、ジョブが正常に完了したときにジョブ通知を送信するかどうかを指定します。
- `Email on a job failure?` スイッチを使用して、ジョブの失敗時にジョブ通知を送信するかどうかを指定します。
- `Email group` フィールドを使用して、ジョブ通知の送信先の電子メールグループを指定します。

Parameter	Value	Category
Agent Id	1	configuration
Running as root?	true	capability
Running as User	root	capability
Platform	"CentOS Linux 7 (Core)"	capability
Architecture	x64	capability
PEM host	127.0.0.1	configuration
PEM port	5444	configuration
Log level	debug1	configuration
Agent SSL key path	/root/.pem//agent1.key	configuration
Agent SSL crt path	/root/.pem//agent1.crt	configuration
Long wait	30	configuration
Short wait	10	configuration
Alert threads	1	configuration

図3 PEM エージェントプロパティダイアログ-エージェント設定タブ

Agent Configurations タブには、エージェントの現在の設定と機能がすべて表示されます。

- Parameter 列にはパラメーターのリストが表示されます。
- Value 列には、対応するパラメーターの現在の値が表示されます。
- Category 列には、対応するパラメーターのカテゴリが表示されます。configuration または capability のいずれかです。

PEM エージェントのトラブルシューティング

4.1 削除された PEM エージェントの復元

エージェントが `pem.agent` テーブルから削除されている場合、復元することはできません。エージェントを再登録するには、`pemworker` ユーティリティを使用する必要があります。

エージェントが PEMWeb クライアントから削除されたが、値が `active = f` の `pem.agent` テーブルにまだエントリがある場合は、次の手順を使用してエージェントを復元できます。

1. 次のコマンドを使用して、`id` フィールドと `active` フィールドの値を確認します。

```
pem=# SELECT * FROM pem.agent;
```

2. `pem.agent` テーブルのエージェントのステータスを `true` に更新します。

```
pem=# UPDATE pem.agent SET active=true WHERE id=<x>;
```

ここで、`x` は、手順 1 で使用したクエリの出力に表示された識別子です。

3. PEMWeb クライアントを更新します。

削除されたエージェントが再び復元されます。ただし、その特定のエージェントにバインドされたサーバーはダウンしているように見える場合があります。この問題を解決するには、サーバーの PEM エージェントプロパティを変更して、バインドされたエージェントを再度追加する必要があります。変更が成功すると、サーバーは正常に実行されていると表示されます。

4.2 コマンドラインを使用して、ダウンまたは不明な状態の PEM エージェントを削除する

Using the PEM web interface to delete PEM agents with Down or Unknown status may be difficult if the number of such agents is large. In such a situation, you might want to use the command line interface to delete Down or Unknown agents.

1. 次のクエリを使用して、*N*時間以上 Down であるエージェントを削除します。

```
UPDATE pem.agent SET active=false WHERE id IN
(SELECT a.id FROM pem.agent
a JOIN pem.agent_heartbeat b ON (b.agent_id=a.id)
WHERE a.id IN
(SELECT agent_id FROM pem.agent_heartbeat WHERE (EXTRACT (HOUR FROM now()) -
EXTRACT (HOUR FROM last_heartbeat)) > <N> ));
```

2. 次のクエリを使用して、ステータスが Unknown のエージェントを削除します。

```
UPDATE pem.agent SET active=false WHERE id IN
(SELECT id FROM pem.agent WHERE id NOT IN
(SELECT agent_id FROM pem.agent_heartbeat));
```

EDBPostgresEnterpriseManager エージェントユーザーガイド

著作権 ©2007-2020EnterpriseDBCorporation。無断複写・転載を禁じます。

EnterpriseDB@Corporation34CrosbyDrive、Suite201、Bedford、MA01730、USA

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

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

索引

Agent Configuration, 18
Agent Privileges, 15
Agent Properties, 23

deleting agents with Down status, 27
deleting agents with Unknown status, 27

PEM
 architecture, 1
PEMArchitecture, 2
PEM エージェントのトラブルシューティング, 26
PEM エージェントの管理, 15

restoring a deleted agent, 26

エージェントの登録, 9

結論, 28