



EDB Postgres Enterprise Manager Configuring pgBouncer for Use with PEM Agents

リリース *8.3.0*

EnterpriseDB

2021 年 12 月 26 日

目次

第 1 章	The PEM Server - PEM Agent Connection Management Mechanism	3
第 2 章	Preparing the PEM Database Server	5
第 3 章	Configuring PgBouncer	7
第 4 章	Configuring the PEM Agent	11
4.1	新しい PEM エージェントの設定 (RPM 経由でインストール)	11
4.2	既存の PEM エージェントの設定 (RPM 経由でインストール)	12

このドキュメントは、Windows 以外のマシン上の PEM エージェントから Postgres Enterprise Manager (PEM) サーバーへの接続数を制限するためのコネクションプーラとしての pgBouncer の使用に関する詳細情報を提供します。

- PEM データベースサーバーの準備-pgBouncer で使用する PEM データベースサーバの準備に関する情報を提供します。
- pgBouncer の設定-PEM データベースサーバと連携する make の pgBouncer の設定に関する詳細情報を提供します。
- PEM エージェントの設定-pgBouncer に接続するための PEM エージェントの設定に関する詳細情報を提供します。

PEM Web インタフェースの使用に関する詳細については、[PEM Administrator's Guide](#) を参照してください。

このドキュメントでは、Postgres という用語を使用して、PostgreSQL または Advanced Server データベースを意味します。

第1章 The PEM Server - PEM Agent Connection Management Mechanism

各 PEM エージェントは、各ユーザの SSL 証明書を使用して PEM データベースサーバに接続します。例、ID#1 を持つエージェントは、agent1 ユーザを使用して PEM データベースサーバに接続します。

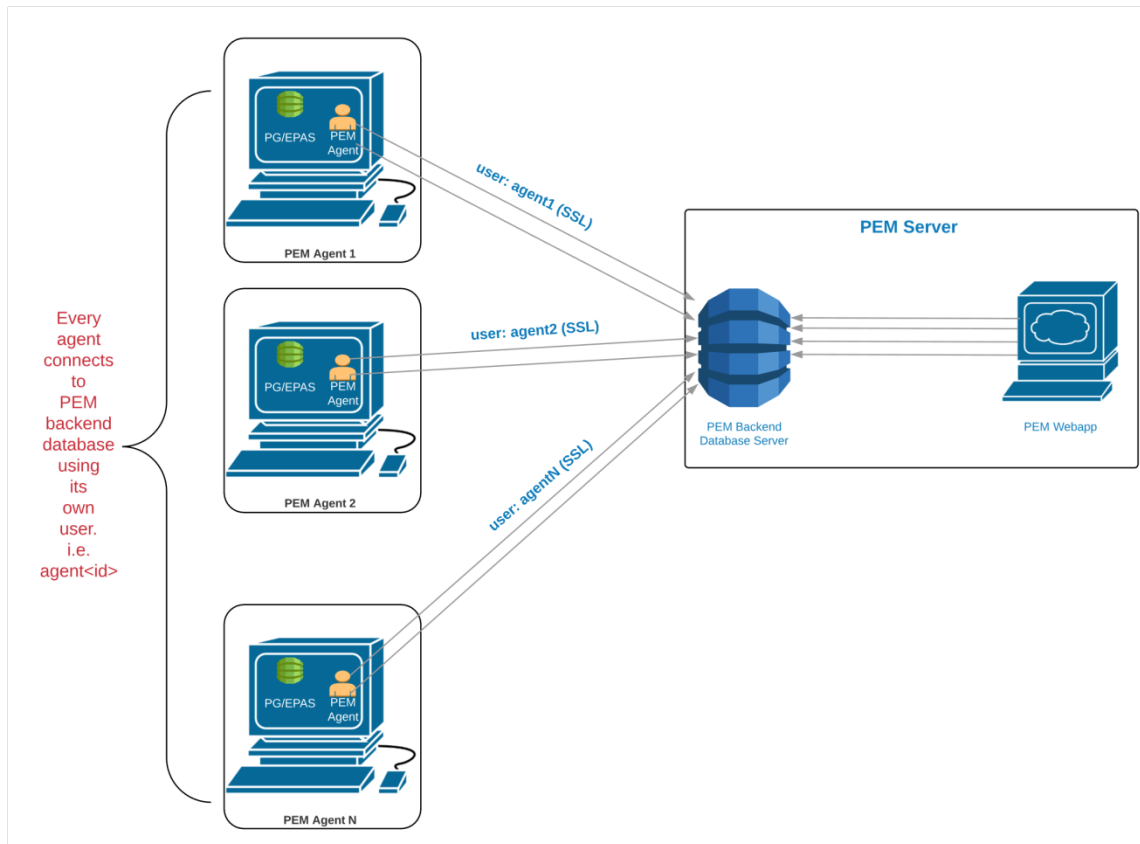


図 1: Connecting to the PEM database without pgBouncer

PEM バージョン 7.5 より前のバージョンでは、次の制限により、PEM サーバーと PEM エージェント間のコネクションプーラの使用が許可されていませんでした。

- PEM エージェントは SSL 証明書を使用して PEM データベースサーバに接続します。
- PEM データベースサーバに接続するときに、個々のユーザ識別子を使用します。

EDB は PEM エージェントを変更して、エージェントが（専用のエージェントユーザーではなく）共通のデータベースユーザを使用して PEM データベースサーバに接続できるようにしました。

コネクションプーラとしてバージョン 1.9.0 以降の PgBouncer バージョンを使用することをお勧めします。バージョン 1.9.0 以降は cert 認証をサポートしています。PEM エージェントは、SSL 証明書を使用して

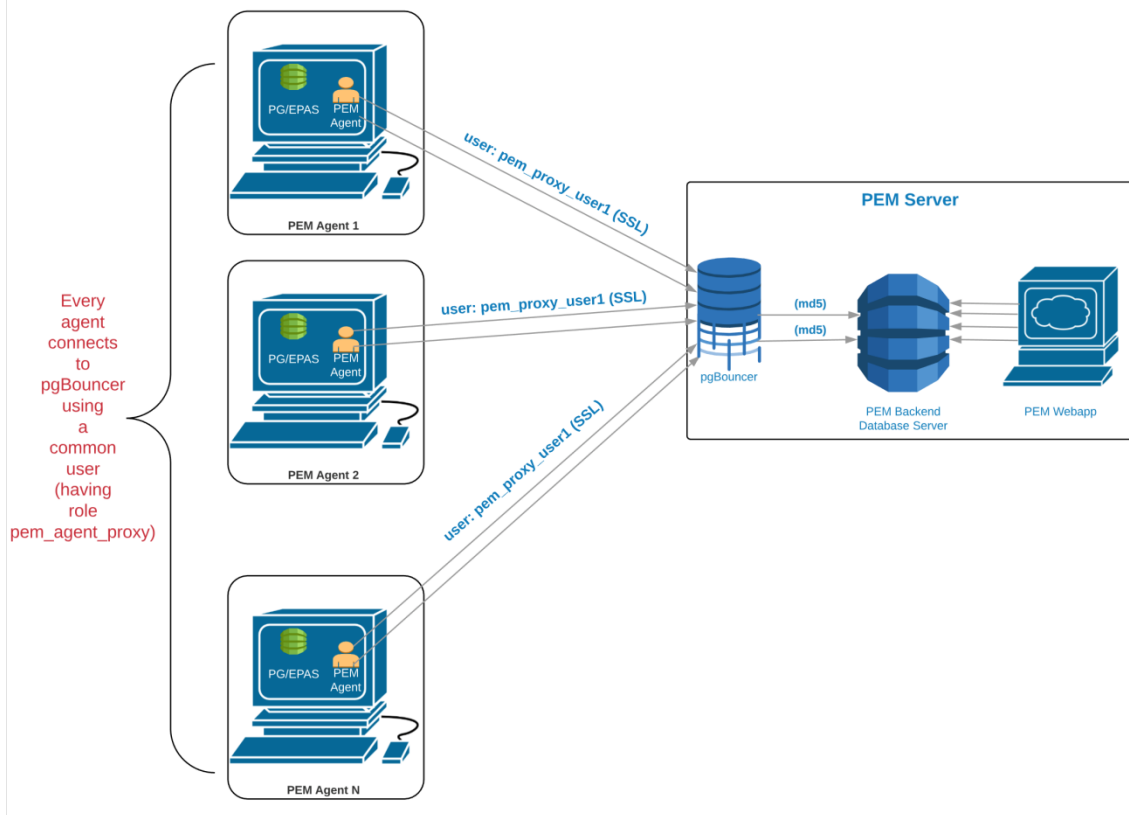


図 2: Connecting to pgBouncer.

pgBouncer に接続できます。

第2章 Preparing the PEM Database Server

PgBouncer と連携するように PEM データベースサーバを構成する必要があります。次の例は、PEM データベースサーバの構成に必要な手順を示しています。

1. PEM データベースサーバに `pgbouncer` 名前付けの専用ユーザを作成します。例：

```
* pem=# CREATE USER pgbouncer PASSWORD 'ANY_PASSWORD' LOGIN;  
* CREATE ROLE
```

2. PEM データベースサーバで、`pem_admin` および `pem_agent_pool` role メンバーシップを持つ `pem_admin1` 名前付けのユーザ（非スーパーユーザ）を作成します。例：

```
* pem=# CREATE USER pem_admin1 PASSWORD 'ANY_PASSWORD' LOGIN  
* CREATEROLE;  
* CREATE ROLE  
* pem=# GRANT pem_admin, pem_agent_pool TO pem_admin1;  
* GRANT ROLE
```

3. `pem` データベースの `pgbouncer` ユーザに `CONNECT` 特権を付与します。例：

```
* pem=# GRANT CONNECT ON DATABASE pem TO pgbouncer ;GRANT USAGE ON  
* SCHEMA pem TO pgbouncer;  
* GRANT
```

4. `pem` データベースの `pem` スキーマの `pgbouncer` ユーザに `USAGE` 特権を付与します。例：

```
* pem=# GRANT USAGE ON SCHEMA pem TO pgbouncer;  
* GRANT
```

5. `pem` データベースの `pem.get_agent_pool_auth(text)` ファンクションの `pgbouncer` ユーザに `EXECUTE` 特権を付与します。例：

```
* pem=# GRANT EXECUTE ON FUNCTION pem.get_agent_pool_auth(text) TO  
* pgbouncer;  
* GRANT
```

6. `pem.create_proxy_agent_user(varchar)` ファンクションを使用して、PEM データベースサーバに `pem_agent_user1` 名前付けのユーザを作成します。例：

```
* pem=# SELECT pem.create_proxy_agent_user('pem_agent_user1');  
* create_proxy_agent_user
```

(次のページに続く)

(前のページからの続き)

```
* -----  
* (1 row)
```

- このファンクションは、ランダムパスワードを使用して同じ名前のユーザを作成し、そのユーザーに `pem_agent` および `pem_agent_pool` ロールを付与しユーザ。これにより、pgBouncer はエージェントに代わってプロキシユーザを使用できます。

7. PEM データベースサーバの `pg_hba.conf` ファイルのスタートに次のエントリを追加します。これにより、pgBouncer ユーザは md5 認証メソッドを使用して pem データベースに接続できます。例：

```
* # Allow the PEM agent proxy user (used by  
* # pgbouncer) to connect the to PEM server using  
* # md5  
* local pem pgbouncer,pem_admin1 md5
```

第3章 Configuring PgBouncer

PEM データベースサーバと連携するように PgBouncer を構成する必要があります。この例では、enterprisedb システムユーザとして PgBouncer を実行します。次の手順は、pgBouncer (バージョン = 1.9) を構成するプロセスの概要を示しています。

1. ターミナルウィンドウを開き、pgBouncer ディレクトリに移動します。
2. pgBouncer (pgbouncer.ini が存在する) の etc ディレクトリの所有者を enterprisedb に変更し、ディレクトリ権限を 0700 に変更します。例：

```
* $ chown enterprisedb:enterprisedb /etc/edb/pgbouncer1.9
* $ chmod 0700 /etc/edb/pgbouncer1.9
```

3. pgbouncer.ini または edb-pgbouncer.ini ファイルの内容を次のように変更します。

```
* [databases]
* ;; Change the pool_size according to maximum connections allowed
* ;; to the PEM database server as required.
* ;; 'auth_user' will be used for authenticate the db user (proxy
* ;; agent user in our case)
* pem = port=5444 host=/tmp dbname=pem auth_user=pgbouncer
* pool_size=80 pool_mode=transaction
* * = port=5444 host=/tmp dbname=pem auth_user=pgbouncer
* pool_size=10
* [pgbouncer]
* logfile = /var/log/edb/pgbouncer1.9/edb-pgbouncer-1.9.log
* pidfile = /var/run/edb/pgbouncer1.9/edb-pgbouncer-1.9.pid
* listen_addr = *
* ;; Agent needs to use this port to connect the pem database now
* listen_port = 6432
* ;; Require to support for the SSL Certificate authentications
* ;; for PEM Agents
* client_tls_sslmode = require
* ;; These are the root.crt, server.key, server.crt files present
* ;; in the present under the data directory of the PEM database
* ;; server, used by the PEM Agents for connections.
* client_tls_ca_file = /var/lib/edb/as11/data/root.crt
* client_tls_key_file = /var/lib/edb/as11/data/server.key
* client_tls_cert_file = /var/lib/edb/as11/data/server.crt
* ;; Use hba file for client connections
```

(次のページに続く)

```
* auth_type = hba
* ;; Authentication file, Reference:
* ;; https://pgbouncer.github.io/config.html#auth_file
* auth_file = /etc/edb/pgbouncer1.9/userlist.txt
* ;; HBA file
* auth_hba_file = /etc/edb/pgbouncer1.9/hba_file
* ;; Use pem.get_agent_pool_auth(TEXT) function to authenticate
* ;; the db user (used as a proxy agent user).
* auth_query = SELECT * FROM pem.get_agent_pool_auth($1)
* ;; DB User for administration of the pgbouncer
* admin_users = pem_admin1
* ;; DB User for collecting the statistics of pgbouncer
* stats_users = pem_admin1
* server_reset_query = DISCARD ALL
* ;; Change based on the number of agents installed/required
* max_client_conn = 500
* ;; Close server connection if its not been used in this time.
* ;; Allows to clean unnecessary connections from pool after peak.
* server_idle_timeout = 60
```

4. 次のコマンドを使用して、PgBouncer の/etc/edb/pgbouncer1.9/userlist.txt 認証ファイルを作成および更新します。

```
* pem=# COPY (
* SELECT 'pgbouncer'::TEXT, 'pgbouncer_password'
* UNION ALL
* SELECT 'pem_admin1'::TEXT, 'pem_admin1_password'
* TO '/etc/edb/pgbouncer1.9/userlist.txt'
* WITH (FORMAT CSV, DELIMITER ' ', FORCE_QUOTE *);
* COPY 2
```

- !!! Note
- スーパーユーザは、PEM 認証クエリー関数 pem.get_proxy_auth(text) を呼び出すことができません。pem_admin ユーザがスーパーユーザの場合、パスワードを認証ファイル (enterprisedb in the above example) に追加する必要があります。

5. 次のコンテンツを含む PgBouncer の HBA ファイル (/etc/edb/pgbouncer1.9/hba_file) を作成します。

```
* # Use authentication method md5 for the local connections to
* # connect pem database & pgbouncer (virtual) database.
* local pgbouncer all md5
* # Use authentication method md5 for the remote connections to
* # connect to pgbouncer (virtual database) using enterprisedb
* # user.
* host pgbouncer,pem pem_admin1 0.0.0.0/0 md5
```

```
* # Use authentication method cert for the TCP/IP connections to
* # connect the pem database using pem_agent_user1
* hostssl pem pem_agent_user1 0.0.0.0/0 cert
```

6. HBA ファイル (/etc/edb/pgbouncer1.9/hba_file) の所有者を **enterprisedb** に変更し、ディレクトリ権限を **0600** に変更します。例：

```
* $ chown enterprisedb:enterprisedb /etc/edb/pgbouncer1.9/hba_file
* $ chmod 0600 /etc/edb/pgbouncer1.9/hba_file
```

7. PgBouncer サービスを有効にして、サービスをスタートします。例：

```
* $ systemctl enable edb-pgbouncer-1.9
* Created symlink from
* /etc/systemd/system/multi-user.target.wants/edb-pgbouncer-1.9.service
* to /usr/lib/systemd/system/edb-pgbouncer-1.9.service.
* $ systemctl start edb-pgbouncer-1.9
```


第4章 Configuring the PEM Agent

RPM パッケージを使用して PEM エージェントをインストールできます。詳細なインストール情報については、[PEM Linux Installation Guide](#) を参照してください。

SNMP 通知の送信を担当する PEM エージェントは、ノートで設定しないでください。例、PEM サーバーと一緒にインストールされたデフォルトの PEM エージェントが SNMP 通知に使用される場合、pgBouncer で構成しないでください。

4.1 新しい PEM エージェントの設定（RPM 経由でインストール）

RPM パッケージを使用して PEM エージェントをインストールした後、特定の PEM データベースサーバに対して動作するように構成する必要があります。次のコマンドを使用します。

```
$ PGSSLMODE=require PEM_SERVER_PASSWORD=pem_admin1_password
/usr/edb/pem/agent/bin/pemworker --register-agent --pem-server
pem_agent_user1 --display-name "Agent Name"
```

```
Postgres Enterprise Manager Agent registered successfully!
```

上記のコマンドで、`--pem-agent-user` 引数は、`/root/.pem` ディレクトリに `pem_agent_user1` データベースユーザの SSL 証明書とキーを作成するようにエージェントに指示します。

例：

```
/root/.pem/pem_agent_user1.crt
```

```
/root/.pem/pem_agent_user1.key
```

キーは、PEM エージェントが `pem_agent_user1` として PEM データベースサーバに接続するために使用されます。また、`/usr/edb/pem/agent/etc/agent.cfg`. 名前付けのエージェント構成ファイルを作成します

`agent.cfg` 構成ファイルで使用する `agent-user` に言及する行があります。

例：

```
$ cat /usr/edb/pem/agent/etc/agent.cfg
[PEM/agent]
pem_host=172.16.254.22
pem_port=6432
```

(次のページに続く)

```
agent_id=12
agent_user=pem_agent_user1
agent_ssl_key=/root/.pem/pem_agent_user1.key
agent_ssl_cert=/root/.pem/pem_agent_user1.crt
log_level=warning
log_location=/var/log/pem/worker.log
agent_log_location=/var/log/pem/agent.log
long_wait=30
short_wait=10
alert_threads=0
enable_smtp=false
enable_snmp=false
enable_webhook=false
max_webhook_retries=3
allow_server_restart=true
allow_package_management=false
allow_streaming_replication=false
max_connections=0
connect_timeout=-1
connection_lifetime=0
allow_batch_probes=false
heartbeat_connection=false
```

4.2 既存の PEM エージェントの設定（RPM 経由でインストール）

既存の PEM エージェントを使用している場合は、SSL 証明書とキーファイルをターゲットマシンにコピーして、ファイルを再利用できます。ファイルを変更し、新しいパラメータを追加して、既存の `agent.cfg` ファイルのいくつかのパラメーターを置き換える必要があります。

エージェントに使用する `agent_user` の行を追加します。例：

```
agent_user=pem_agent_user1
```

ポートを更新して pgBouncer ポートを指定します。例：

```
pem_port=6432
```

証明書とキーの場所を更新します。例：

```
agent_ssl_key=/root/.pem/pem_agent_user1.key
agent_ssl_cert=/root/.pem/pem_agent_user1.crt
```

ノート：別の方法として、エージェントの自己登録スクリプトを実行できますが、新しいエージェント ID が作成されます。エージェントの自己登録スクリプトを実行する場合は、新しいエージェント ID を既存

の ID に置き換え、pem.agent テーブルの新しいエージェント ID のエントリーを無効にする必要があります。例：

```
pem=# UPDATE pem.agent SET active = false WHERE id = <new_agent_id>;
```

```
UPDATE 1
```

!!! Note * 既存の SSL 証明書、キーファイル、およびエージェント構成ファイルのバックアップを保持します。