

ApsaraDB for RDS

クイックスタート (PostgreSQL)

クイックスタート (PostgreSQL)

ご利用案内

制限事項

ApsaraDB for PostgreSQL は、インスタンスの安定性とセキュリティを確保するために下記示すように一部使用制限があります。

操作	RDS の使用に関する制限
データベースパラメーターの設定変更	当面はサポートされていません。
データベースの root 権限	スーパーユーザー権限は利用できません。
データベースのバックアップ	データのバックアップは pg_dump でのみ実行できます。
データの移行	pg_dump でバックアップしたデータは psql でのみ復元できます。
データベースレプリケーションの構築	HA モードは、PostgreSQL のストリーミングレプリケーションに基づいて自動的に構築されるため、手動で構築する必要はありません。PostgreSQL のスタンバイノードはユーザーに表示されないため、直接アクセスできません。
RDS インスタンスの再起動	インスタンスの再起動は、RDS コンソールまたは OPEN API から行う必要があります。

RDS利用の一般手順

クイックスタートの目的

このドキュメントでは、RDSインスタンスの購入から使用までの手順を説明します。また、RDSインスタン

スのApsaraDBを作成する基本的な設定と、インスタンスデータベースへの接続方法について詳しく説明します。

対象となる方

初めてRDSインスタンスを使用するユーザ

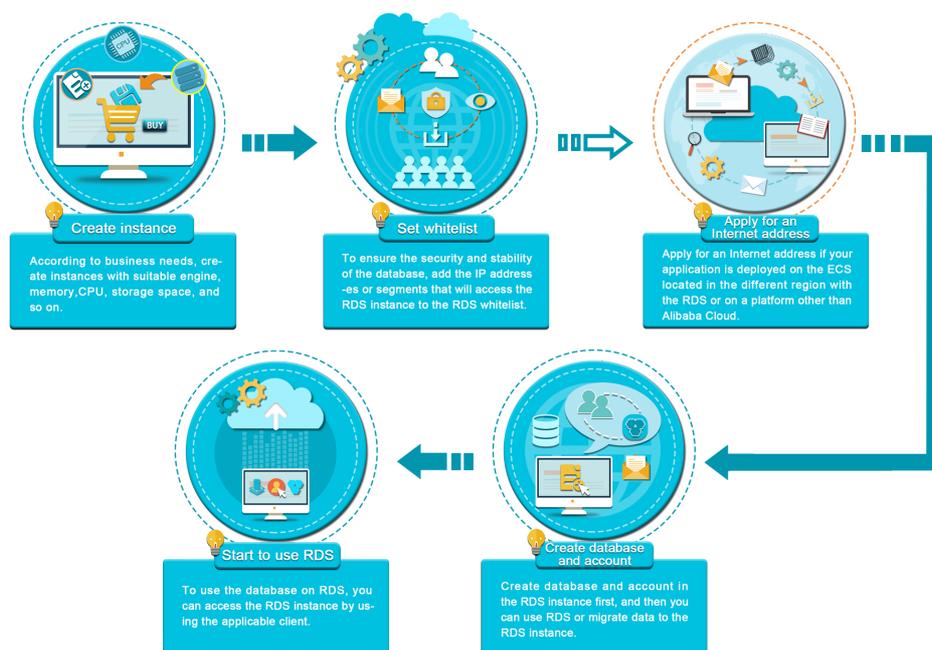
基本的なパフォーマンスチューニングをするユーザ

RDSインスタンスへ接続したいユーザ

クイックスタートのフローチャート

初めてAlibaba Cloud ApsaraDB for RDSを使用する場合は、「ApsaraDB for RDS 制限事項」と「ApsaraDB for RDSコンソール」を参照してください。

次の図は、インスタンスの作成から使用までの手順の説明になります。



インスタンスの作成

インスタンスの作成

Alibaba Cloud ApsaraDB for RDS コンソールまたは API を使用して、ApsaraDB for RDS インスタンスを作成できます。インスタンス価格の詳細については、「ApsaraDB for RDS の価格設定」を参照してください。

このドキュメントでは、ApsaraDB for RDS コンソールを使用してインスタンスを作成する方法について説明します。API を使用してインスタンスを作成する方法の詳細については、「RDS インスタンスの作成」を参照してください。

前提条件

Alibaba Cloud アカウントに登録済みであること。

操作手順

RDS コンソールにログインします。

インスタンスリスト ページで、**インスタンスの作成** をクリックして、**インスタンスの作成** ページに入ります。

サブスクリプション または **従量課金** を選択してください。課金方法の詳細については、**価格の説明** を参照してください。

インスタンス構成を選択します。パラメータは次のように記述されます。

基本設定

リージョンとゾーン：選択リージョンとインスタンスが配置されているゾーン。単一のゾーンまたはマルチゾーンをサポートするリージョンもあれば、単一のゾーンのみをサポートするリージョンもあります。詳細については **リージョンとゾーン** を参照してください。

注：異なる地域の製品はイントラネットを介して相互に通信することはできず、インスタンスを購入した後でインスタンス領域を変更することはできません。そのため、リージョン選択するときは注意してください。

データベースエンジン：RDS は、MySQL、SQL Server、PostgreSQL、PPAS 9.3 をサポートします。異なる地域では、異なるデータベースタイプが

サポートされています。このドキュメントを使用する場合は、実際のインターフェイスを参照してください。

バージョン：データベースのバージョン。現在、RDS は MySQL 5.6、SQL Server 2008 R2、および PostgreSQL 9.4 をサポートしています。異なる地域で異なるデータベースのバージョンがサポートされています。このドキュメントを使用する場合は、実際のインターフェイスを参照してください。

MySQL データベースの場合、MySQL 5.6 を選択することをお勧めします。これは、データファイルの占有スペースを大幅に削減してストレージコストを削減できる TokuDB ストレージエンジンをサポートするためです。

SQL Server 2008 R2 および SQL Server 2012 では、さまざまな機能がサポートされています。詳細は SQL Server 2008 R2 と SQL Server 2012 の機能の違いを参照してください。

シリーズ：RDS インスタンスは、Basic Edition、High Availability Edition、および Finance Edition をサポートします。異なるデータベースバージョンは異なるシリーズをサポートします。このドキュメントを使用する場合は、実際のインターフェイスを参照してください。

ネットワークタイプ：RDS は従来のネットワークと仮想プライベートクラウド (VPC) をサポートします。あらかじめ VPC を作成する必要があります。また、インスタンスの作成後にネットワークタイプを変更することもできます。詳細は、ネットワークタイプの設定を参照してください。

タイプ：インスタンスが占有する CPU およびメモリ。接続数と最大IOPS（読み書きの場合はそれぞれ測定され、混合読み書きの場合はベンチマークの最大 2 倍）は、タイプによって異なります。インスタンスタイプの詳細については、インスタンスタイプ一覧を参照してください。

ストレージ：ストレージスペースには、データ、システムファイル、bin ログファイル、およびトランザクションファイル用のスペースが含まれています。

サブスクリプション期間：サブスクリプションインスタンスの期間を設定します。

数量：購入する同じ構成のインスタンスの数。

今すぐ購入 をクリックして **注文の確認** ページに入ります。

[サービス利用規約とサービスレベル通知に同意します] の横にあるチェックボックスをオンにしてから：

インスタンスの請求方法がサブスクリプションの場合は、**支払い** をクリックします。

インスタンスの請求方法が従量課金の場合は、**有効化** をクリックします。

初期設定

ホワイトリストの設定

ホワイトリストは、指定されたIPアドレスおよび特定のIPセグメントへのアクセスを制限するために使用されます。ホワイトリストが設定されていない限り、データベースインスタンスにアクセスすることはできません。RDSのセキュリティを維持するために、要件に応じてホワイトリストを定期的を確認して調整することをお勧めします。この章では、主にホワイトリストを設定する方法を紹介します。

背景

イントラネット、インターネット、またはイントラネットとインターネットの両方を介してRDSインスタンスにアクセスできます。各接続タイプ（イントラネットとインターネット）の適用シナリオの詳細については、イントラネットアドレスとインターネットアドレスの設定を参照してください。

接続タイプを設定する前に、アプリケーションサービスのIPアドレスまたはIPセグメントまたはECSインスタンスをRDSインスタンスのホワイトリストに追加する必要があります。ホワイトリストが設定されると、システムは自動的にRDSインスタンスのイントラネットアドレスを生成します。インターネットアドレスが必要な場合は、インターネットアドレスの申請の詳細な手順を参照してください。

注意：アプリケーションサービスのIPアドレスをホワイトリストに追加した後にRDSインスタンスに接続できない場合は、アプリケーションサービスの実際のIPアドレスを取得する必要があります。

注意

システムは、新しく作成したRDSインスタンスごとに**デフォルト**ホワイトリストグループを自動的に

に作成します。この**デフォルト**ホワイトリストグループは、変更または削除のみできますが、削除することはできません。

新しく作成されたRDSインスタンスごとに、ローカルループバックIPアドレス127.0.0.1がデフォルトの**ホワイトリストグループ**にデフォルトで追加されます。これは、すべてのIPアドレスまたはIPセグメントがこのRDSインスタンスにアクセスすることを禁止されていることを意味します。したがって、他のIPアドレスまたはIPセグメントをRDSホワイトリストに追加する前に、まず**デフォルト**ホワイトリストグループから127.0.0.1を削除する必要があります。

%または0.0.0.0/0は、IPアドレスがRDSインスタンスへのアクセスを許可されていることを示します。この構成では、データベースのセキュリティが大幅に低下するため、お勧めしません。

操作手順

RDSコンソールにログオンします。

リージョンターゲット・インスタンスが配置されている場所を選択します。

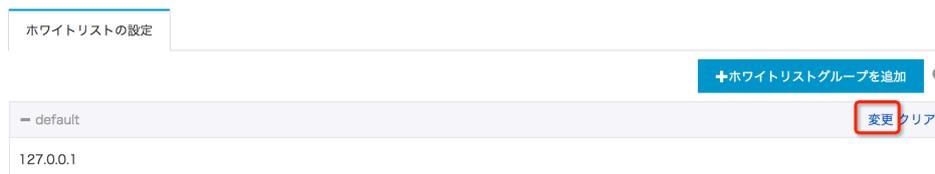
ターゲットインスタンスの名前をクリックすると、**基本情報**ページに移動します。

左側のナビゲーションバーで **セキュリティコントロール** を選択して **セキュリティコントロール** ページにアクセスしてください。

ホワイトリスト設定タブページで、**デフォルト**ホワイトリストグループの**変更**をクリックします (次の図を参照)。

注意：カスタムホワイトリストグループをRDSインスタンスに追加する場合は、**デフォルト**ホワイトリストグループの**クリア**をクリックして、IPアドレス127.0.0.1を最初に削除してから、**ホワイトリストグループを追加**をクリックします。カスタムホワイトリストの設定手順は、次の手順と似ています。

セキュリティコントロール



ホワイトリストの設定

+ ホワイトリストグループを追加

- default	変更 クリア
127.0.0.1	

注意: IP ホワイトリストは、0.0.0.0/0 を指定するとすべてのアクセスを許可してしまいます。また、127.0.0.1 アドレスを指定すると、このアドレス以外の外部アクセスを拒否することができます。 [ホワイトリスト設定の説明](#)

グループの変更ページで、ホワイトリストフィールドのRDSインスタンスにアクセスするIPアドレスまたはIPセグメントを追加します。ECSイントラネットIPアドレスを追加する必要がある場合は、次の図に示すように、[ECSイントラネットIPアドレスのアップロード]をクリックし、プロンプトに従ってIPアドレスを選択します。

グループを変更する
×

グループ名:

ホワイトリスト:

127.0.0.1

ECS の IP アドレスをアップロード
最大のホワイトリスト数 999

コンマ区切りの IP アドレス (例: 192.168.0.1,192.168.0.2)
[ローカル IP を取得する方法](#)

The whitelist takes effect in 1 minute.

OK

キャンセル

パラメータの説明：

グループ名：小文字、数字またはアンダースコアを含む2～32文字を含むことができます。グループ名は、小文字で始まり、文字または数字で終わらなければなりません。この名前は、ホワイトリストグループが正常に作成されたときは変更できません。

ホワイトリスト：RDSインスタンスにアクセスできるカスタムIPアドレスまたはIPセグメントを入力できます。

10.10.10.0/24などのIPセグメントを入力すると、10.10.10.Xという形式のIPアドレスがRDSインスタンスにアクセスできることを示します。

複数のIPアドレスまたはIPセグメントを入力する必要がある場合は、192.168.0.1,172.16.213.9など、コンマで区切ります（空白を追加しないでください）。

- ホワイトリストグループごとに、MySQL、PostgreSQLインスタンスに対して最大1,000のIPアドレスまたはIPセグメントを設定できます。SQL Serverインスタンスには最大800まで設定できます。

ECSイントラネットIPアドレスのアップロード：このボタンをクリックすると、ECSイントラネットIPアドレスを追加する簡単な方法である、RDSインスタンスと同じアカウントのECSインスタンスのイントラネットIPアドレスを選択できます。

2. **OK** をクリックします。

ホワイトリストグループを変更または削除する

ビジネス要件に応じて、ホワイトリストグループを変更または削除することができます。詳細な操作手順は次のとおりです。

RDSコンソールにログオンします。

リージョンターゲット・インスタンスが配置されている場所を選択します。

ターゲットインスタンスの名前をクリックすると、**基本情報**ページに移動します。

左側のナビゲーションバーで **セキュリティコントロール** を選択して **セキュリティコントロール** ページにアクセスしてください。

ホワイトリストの設定 タブページで、対象のホワイトリストグループの **変更** または **削除** をクリックします。

IPアドレスまたはIPセグメントを変更したら、**[OK]** をクリックします。または、削除するホワイトリストグループであることを確認する場合は、**[確認]** をクリックします。

接続モードの設定

同じリージョンに配置され、RDSインスタンスと同じネットワークタイプを持つECSインスタンスにアプリケーションをデプロイする場合は、インターネットアドレスは必要ありません。アプリケーションが異なる地域にあるECS、またはRDSインスタンスと異なるネットワークタイプ、またはAlibaba Cloud以外のプラットフォームに配置されている場合は、RDSインスタンスへのアクセスにインターネットアドレスが必要です。

注：インスタンスが同じリージョンにある場合（ゾーンが異なる場合があります）、イントラネットを介してインスタンスにアクセスできます。

背景情報

RDS には、イントラネットアドレスとインターネットアドレスの 2 種類の接続アドレスがあります。

- イントラネットアドレスまたはインターネットアドレスは、[接続モード] が [Standard Mode] である場合にのみ使用できます。
 - アプリケーションが同じリージョン内の ECS にデプロイされている場合は、イントラネットアドレスを使用できます。イントラネットアドレスはデフォルトで提供され、接続アドレスは直接変更できます。
 - アプリケーションが Alibaba Cloud 以外のシステムにデプロイされている場合は、インターネットアドレスを使用する必要があります。[インターネットアドレスの取得] をクリックして、イントラネットアドレスをリリースしてインターネットアドレスを生成します。
- イントラネットアドレスおよびインターネットアドレスは、[接続モード] が [高セキュリティモード] である場合にのみ同時に使用できます。アプリケーションが同じリージョン内の ECS および Alibaba Cloud 以外のシステムに同時にデプロイされている場合は、イントラネットアドレスとインターネットアドレスの両方を使用する必要があります。

注意

- インターネットアドレスのトラフィックに対して料金が発生します。
- 伝送速度とセキュリティレベルの向上のために、アプリケーションを RDS と同じリージョンの ECS に移行することをお勧めします。

操作手順

この例では、イントラネットアドレスとインターネットアドレスの両方を使用します。RDS を使用する際には、実際のシステム計画に基づいて接続モードを設定してください。

1. RDS コンソールにログインして、ターゲットインスタンスを選択します。
2. メニューで [データベースの接続] を選択します。

[データベースの接続] で [インターネットアドレスの取得] をクリックし、表示される確認インターフェイスで [OK] をクリックして、インターネットアドレスを生成します (下図を参照)。インターネットアドレスでは、トラフィックで料金が発生したり、インスタンスのセキュリティが低下したりする可能性があります。使用する際には注意してください。



[接続アドレスの変更] をクリックし、表示されるウィンドウでイントラネットとインターネットの接続アドレスおよびポート番号を設定して、[OK] をクリックします (下図を参照)。

接続アドレスの変更 ×

接続タイプ: 1

接続アドレス: 2

アルファベットと数字で構成され、最初の文字には小文字のアルファベットを使用します。長さの範囲は 8 ~ 64 文字です。

ポート: 3

ポート番号の範囲: 3200 ~ 3999

4

- 接続タイプ: 変更する接続タイプに応じて、[イントラネットアドレス] または [インターネットアドレス] を選択します。
- 接続アドレス: アドレス形式は、xxx.pg.rds.aliyuncs.com です。xxx は、8 ~ 64 文字 (サポートされているのはアルファベットと数字のみ) のユーザー定義フィールドです。先頭の文字は小文字のアルファベットである必要があります (例: *extranet4example*)。
- ポート: RDS が外部サービスを提供するポートの番号を指定します。3,200 ~ 3,999 の整数を指定できます。

データベースとアカウントの作成 (PostgreSQL)

RDS を使用する前に、RDS インスタンス用にデータベースとアカウントを作成する必要があります。PostgreSQL インスタンスでは、RDS コンソールに初期アカウントを作成する必要があります。そしてあなたはクリアできます

背景情報

同一インスタンス内のデータベースは、このインスタンスのすべてのリソースを共有します。各 PostgreSQL インスタンスは、1 つの初期アカウント、無制限の一般アカウント、無制限のデータベースをサポートしています。SQL 文を使用して一般アカウントとデータベースを作成して管理する必要があります。

ローカルデータベースを RDS に移行するには、RDS インスタンスにローカルデータベースと同じデータベースおよびアカウントを作成する必要があります。

各データベースのアカウントに権限を割り当てる際は、最小権限の原則とサービスロールに従ってアカウントを作成し、読み取り専用および読み取り/書き込み権限を合理的に割り当てます。必要に応じて、アカウントとデータベースをさらに小さく分割して、各アカウントが自身のサービスに関連するデータにのみアクセスできるようにすることができます。アカウントがデータをデータベースに書き込む必要がない場合は、読み取り専用権限を割り当てます。

データベースを安全に保護するために、アカウントに強力なパスワードを設定し、パスワードを定期的に変更します。

操作手順

[RDS コンソール]にログインします。

対象のインスタンスが存在するリージョンを選択します。

インスタンスの ID をクリックして、[基本情報] ページに移動します。

左側のナビゲーションバーで [アカウント管理] をクリックして、[アカウント管理] ページに移動します。

次の図に示された [アカウント作成] をクリックします。



作成するアカウントの情報を入力します (下図を参照)。

| アカウントを作成する [アカウント管理に戻る](#)

データベースアカウント:

小文字のアルファベット、数字、アンダースコアで構成され、先頭にアルファベットを使用し、末尾にアルファベットまたは数字を使用します。使用できる文字数は最大 16 文字です。

許可済みデータベース:

未許可のデータベース	許可済みデータベース
データなし	一時的にデータが存在しません

権限 [すべて設定](#) [読み取り/書き込み](#)

許可 >
< 削除

*パスワード:

長さは 8 ~ 32 文字で、大文字、小文字、数字、特殊記号の三種類以上が必要です。特殊記号には下記が含まれます:
@#\$\$%^&*(_+)=!

*パスワードの確認:

パラメーターの説明:

Database Account: 初期アカウント名を表します。2 ~ 16 文字 (小文字のアルファベット、数字、アンダースコアを含む) で指定できます。先頭はアルファベット、末尾はアルファベットか数字にする必要があります。

Password: 初期アカウントのパスワードを指定します。少なくとも次の 3 種類の文字を含む 8 ~ 32 文字で指定できます。

大文字のアルファベット

小文字のアルファベット

数字

特殊文字 (!@#\$\$%^&*()_+)=)

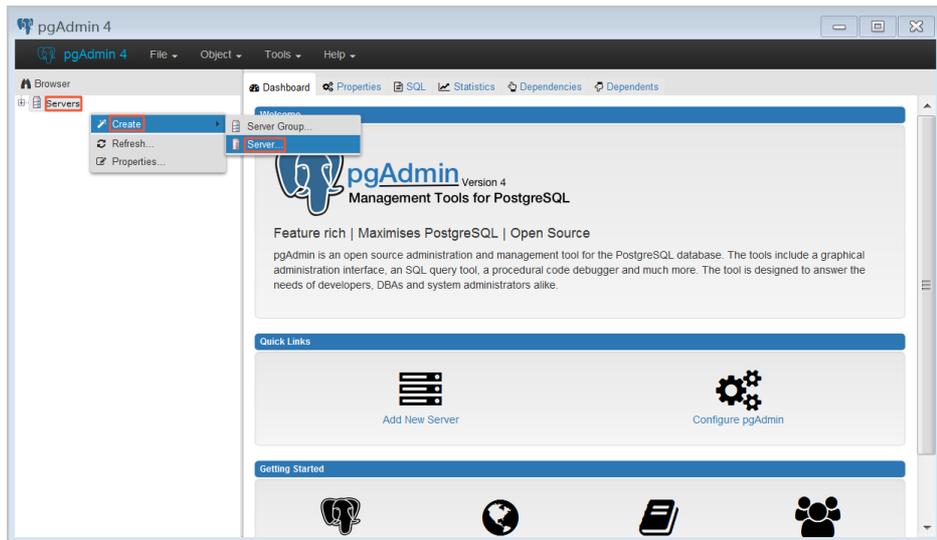
Re-enter Password: 正しく入力されていることを確認するためにパスワードをもう一度入力します。

[OK] をクリックして、初期アカウントの作成は終了です。

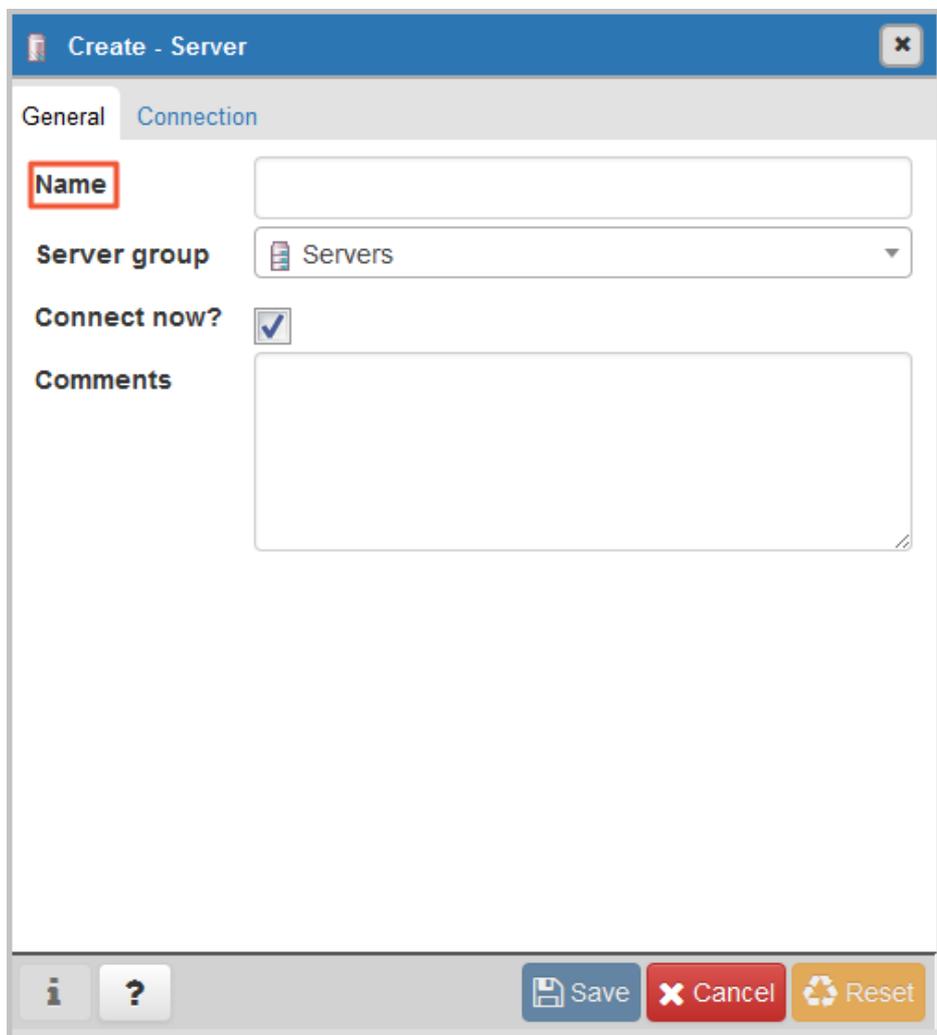
RDS インスタンスにアクセスする IP アドレスを、RDS のホワイトリストに追加します。ホワイトリストの設定方法については、「[ホワイトリストの設定](#)」を参照してください。

pgAdmin 4 クライアントを起動します。

[Servers] を右クリックし、[Create] > [Server] をクリックします (下図を参照)。



[Create - Server] ウィンドウの [General] タブで、サーバー名を入力します (下図を参照)。



[Connection] タブを選択し、接続するインスタンスの情報を入力します (下図を参照)。

The screenshot shows a 'Create - Server' dialog box with the 'Connection' tab selected. The fields are as follows:

- Host name/address: [Empty text box]
- Port: [Empty text box]
- Maintenance database: postgres
- Username: [Empty text box]
- Password: [Empty text box]
- Save password?:
- Role: [Empty text box]
- SSL mode: Prefer (dropdown menu)

A red error message at the bottom reads: "Port" must be greater than or equal to 1024.

Buttons at the bottom: Save, Cancel, Reset.

パラメーターの説明:

Host name/address: RDS インスタンスの接続アドレスを参照します。アプリケーションがイントラネットを使用して RDS インスタンスにアクセスする場合は、RDS インスタンスのイントラネットアドレスを入力します。アプリケーションがインターネットを使用して RDS インスタンスにアクセスする場合は、RDS インスタンスのインターネットアドレスを入力します。次の手順は、RDS インスタンスの接続アドレスとポート番号を見つける方法を示しています。

Port: RDS インスタンスのポート番号を指定します。アプリケーションがイントラネットを使用して RDS インスタンスにアクセスする場合は、RDS インスタンスのイントラネットポート番号を入力します。アプリケーションがインターネットを使用して RDS インスタンスにアクセスする場合は、RDS インスタンスのインターネットポート番号を入力します。

Username: RDS インスタンスの初期アカウント名を指定します。

Password: RDS インスタンスの初期アカウントに対応するパスワードを指定します。

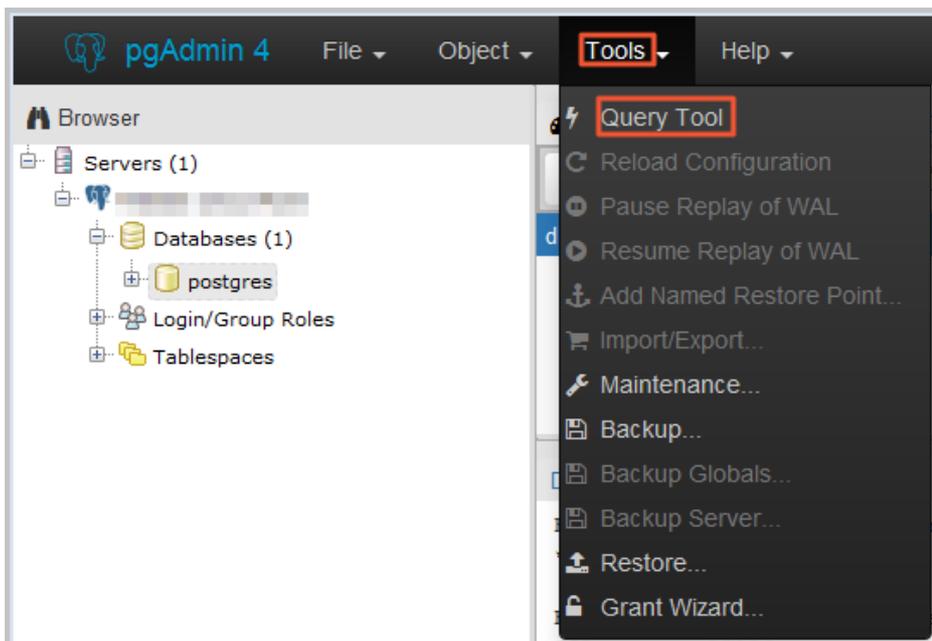
[Save] をクリックします。

接続情報が正しい場合は、[Servers] > [サーバー名] > [Databases] > [postgres] をクリックすると、RDS インスタンスへの接続が正常に行われたことを示す、次のインターフェイスが表示されます。

注意: postgres は、RDS インスタンスのデフォルトのシステムデータベースです。このデータベースで操作を実行しないでください。

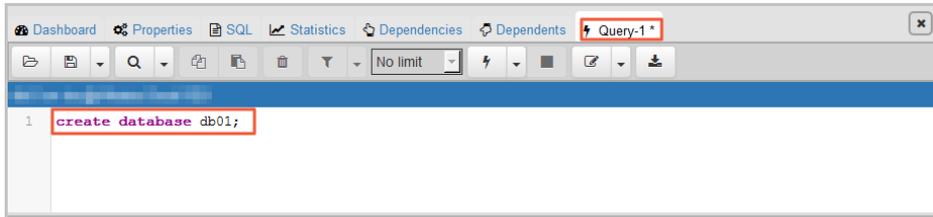


[postgres] をクリックし、[Tools] > [Query Tool] をクリックします (下図を参照)。

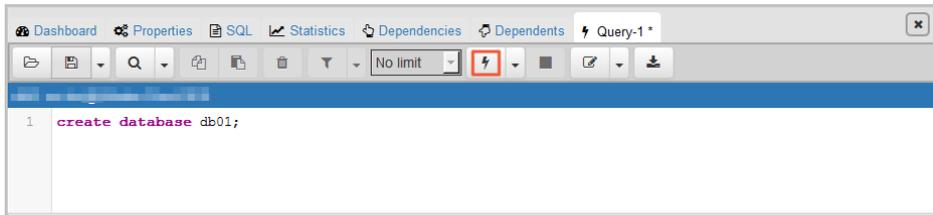


[Query-1] タブページで次のコマンドを入力し、データベースを作成します (下図を参照)。

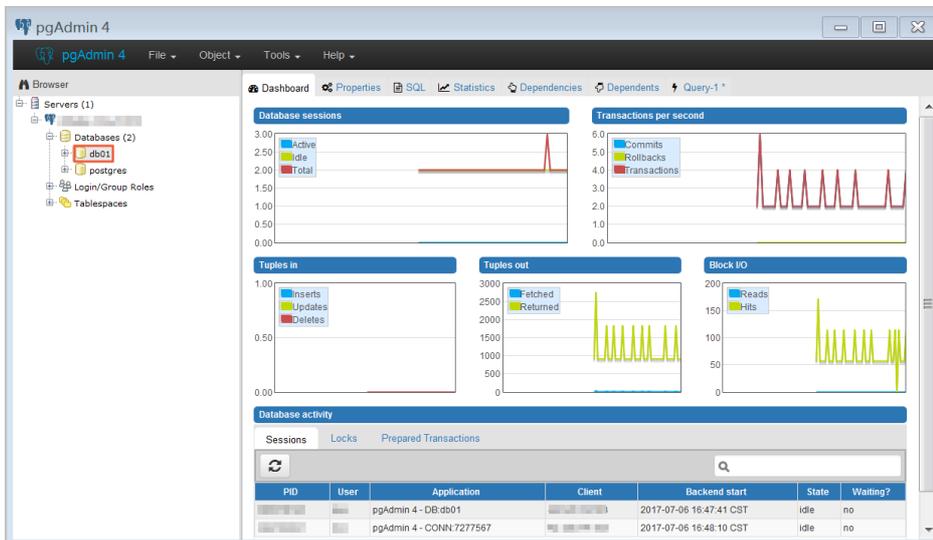
```
create database <database name>;
```



[Execute/Refresh] アイコンをクリックします (下図を参照)。

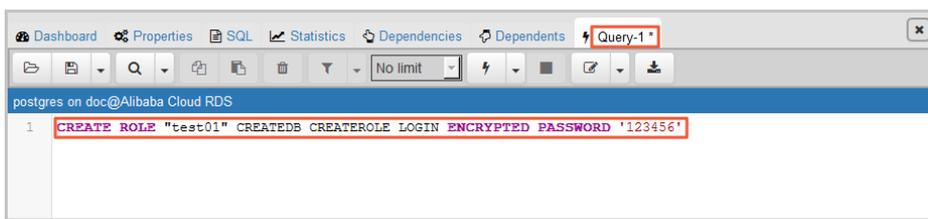


実行が正常に終了すると、新しいデータベースの作成が正常に終了したと表示されます。[Databases] を右クリックして [Refresh] をクリックすると、新しく作成されたデータベースが表示されます (下図を参照)。

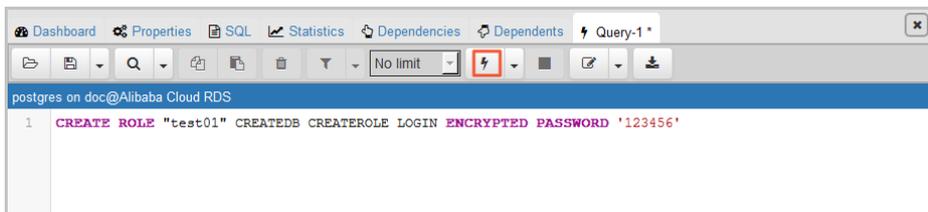


[Query-1] タブページで次のコマンドを入力し、アカウントを作成します (下図を参照)。

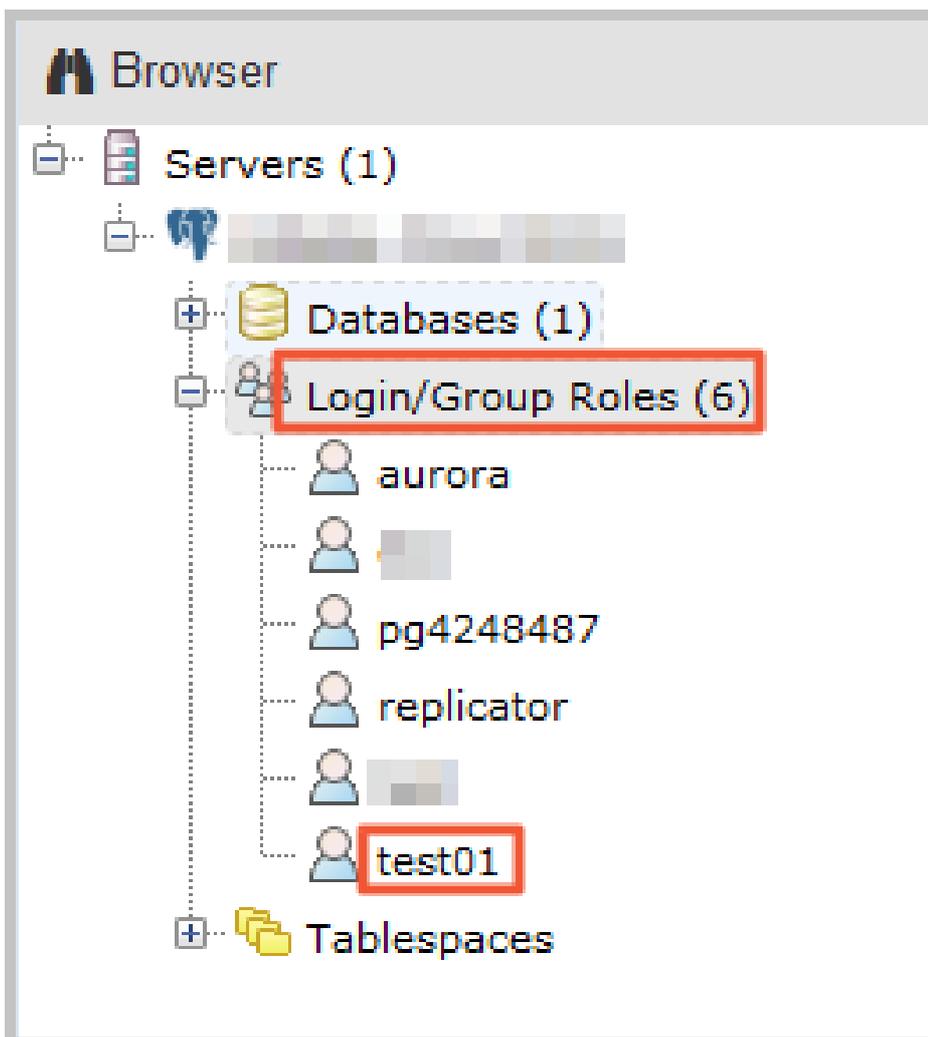
```
CREATE ROLE "username" CREATEDB CREATEROLE LOGIN ENCRYPTED PASSWORD 'password';
```



[Execute/Refresh] アイコンをクリックします (下図を参照)。



実行が正常に終了すると、新しいアカウントの作成が正常に終了したと表示されます。 [Login/Group Roles] を右クリックして [Refresh] をクリックすると、新しく作成されたアカウントが表示されます (下図を参照)。



インスタンスへの接続

PostgreSQL クライアントを通じて RDS インスタンスに接続できます。この記事では、pgAdmin 4 クライアントを例として使用して、接続手順を示します。

背景情報

RDS for PostgreSQL は PostgreSQL と完全に互換性があるため、同じ方法を使用してデータベースに接続できます。この記事では、pgAdmin 4 クライアントを例として使用して、RDS インスタンスに接続します。この方法は、他のクライアントを使用する場合にも利用できます。クライアントを通じて RDS インスタンスに接続する場合は、イントラネットアドレスとインターネットアドレスの設定に関する次の情報に注意してください。

クライアントのインストール先が接続先の RDS インスタンスと同じリージョン内にあり、その RDS インスタンスと同じネットワークタイプの設定を持つ場合は、イントラネットアドレスを使用します。

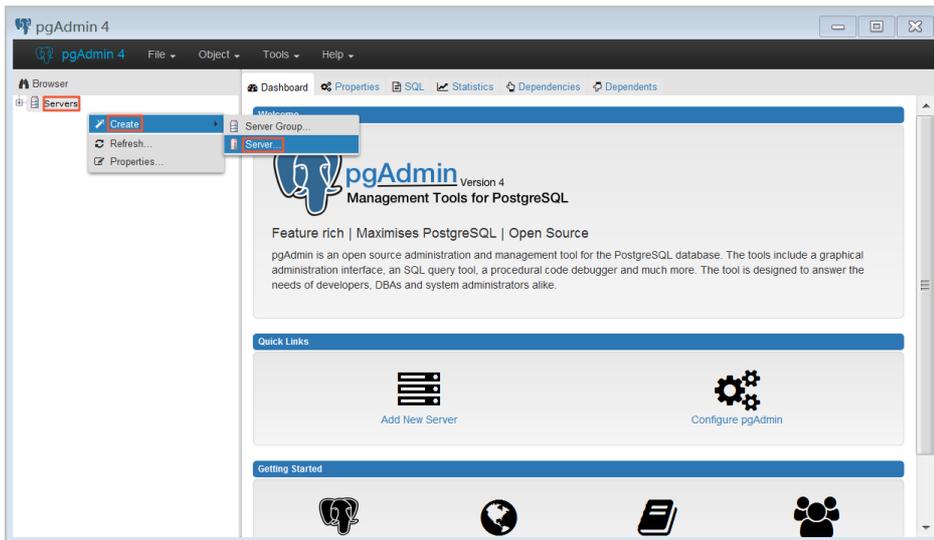
その他の状況では、インターネットアドレスを使用します。

操作手順

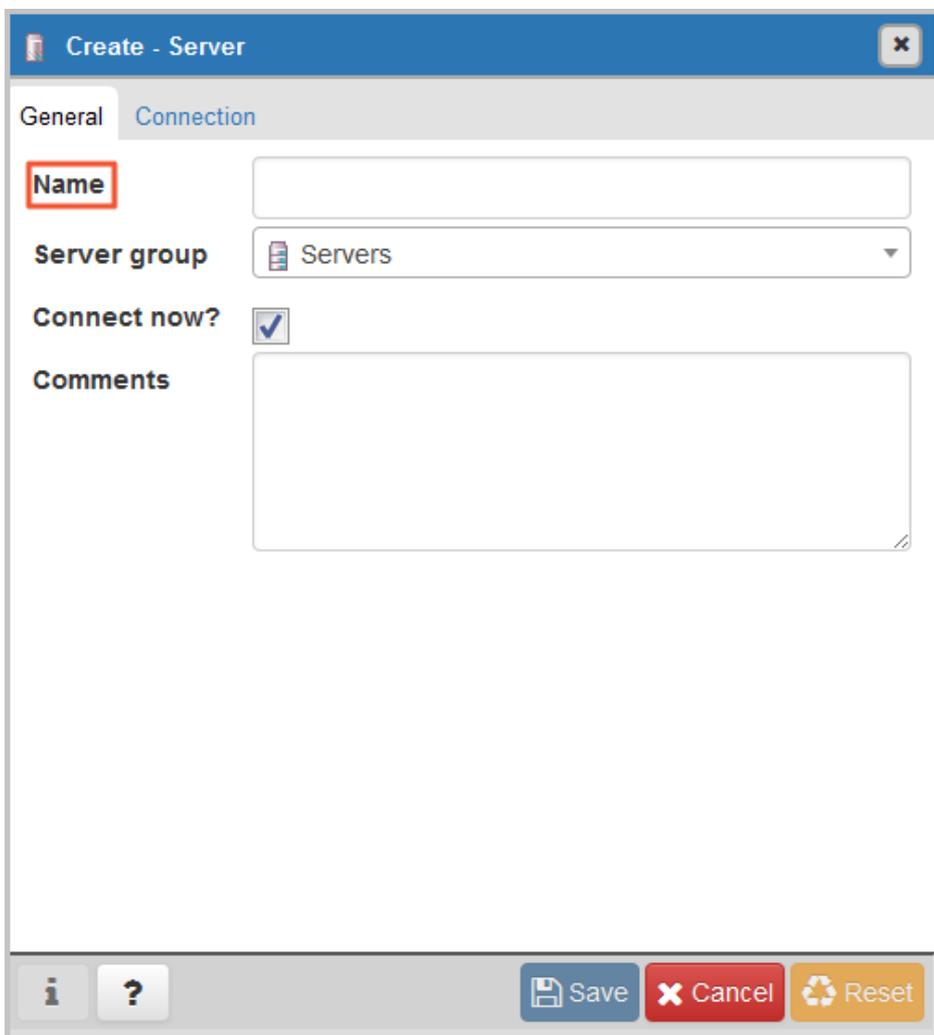
RDS インスタンスにアクセスする IP アドレスを、RDS のホワイトリストに追加します。ホワイトリストの設定方法については、「[ホワイトリストの設定](#)」を参照してください。

pgAdmin 4 クライアントを起動します。

[Servers] を右クリックし、[Create]、[Server] の順に選択します (下図を参照)。



[Create - Server] ウィンドウの [General] タブで、サーバー名を入力します (下図を参照)。



[Connection] タブを選択し、接続するインスタンスの情報を入力します (下図を参照)。

Host name/address

Port

Maintenance database postgres

Username

Password

Save password?

Role

SSL mode Prefer

'Port' must be greater than or equal to 1024.

Save Cancel Reset

パラメーターの説明:

[Host name/address]: RDS インスタンスの接続アドレスを指定します。アプリケーションがイントラネットを使用して RDS インスタンスにアクセスする場合は、RDS インスタンスのイントラネットアドレスを入力します。アプリケーションがインターネットを使用して RDS インスタンスにアクセスする場合は、RDS インスタンスのインターネットアドレスを入力します。次の手順は、RDS インスタンスの接続アドレスとポート番号を見つける方法を示しています。

RDS コンソールにログインします。

対象のインスタンスが存在するリージョンを選択します。

インスタンスの ID をクリックして、[Basic Information] ページに移動します。

[Basic Information] 領域で、RDS インスタンスの接続アドレスとポート番号を見つけることができます (下図を参照)。

[Port]: RDS インスタンスのポート番号を指定します。アプリケーションがイントラネットを使用して RDS インスタンスにアクセスする場合は、RDS インスタンスのイントラネットポート番号を入力します。アプリケーションがインターネットを使用して RDS インスタンスにアクセスする場合は、RDS インスタンスのインターネットポート番号を入力します。

[Username]: RDS インスタンスの初期アカウント名を指定します。

[Password]: RDS インスタンスの初期アカウント名に対応するパスワードを指定します。
。

[Save] をクリックします。

接続情報が正しい場合は、[Servers]、[server name]、[Databases]、[postgres] の順に選択すると、RDS インスタンスへの接続に成功したことを示す、次のインターフェイスが表示されます。
。

注意: postgres は、RDS インスタンスのデフォルトのシステムデータベースです。このデータベースで操作を実行しないでください。



oss_fdwを使用して外部データファイルを読み

書きする

Alibaba Cloud では、OSS 上のデータを PostgreSQL および PPAS データベースにロードするために `oss_fdw` プラグインを使用できます。また、データベース内のデータを OSS に書き込むこともできます。

oss_fdw パラメータ

他の `fdw` インタフェースと同様に、`oss_fdw` は OSS（外部データソース）に格納されたデータをカプセル化して、テーブルからデータを読み取るような OSS 上のファイルを読み取ることができます。`oss_fdw` は、OSS 上のファイル・データへの接続および解析に使用される一意のパラメータを提供します。

CREATE SERVER の主なパラメータ

注：パラメータ値は、不要なスペースを含まない引用符 (") で囲む必要があります。

パラメータ	説明
<code>ossendpoint</code>	イントラネットから OSS にアクセスするために使用されたアドレス（ホスト）。
<code>id</code>	OSS アカウント ID。
<code>key</code>	OSS アカウントキー。
<code>bucket</code>	OSS バケットは、OSS アカウントが作成された後に割り当てられます。

CREATE SERVER の補助パラメータ

注意：

パラメータ値は、不要なスペースなしで引用符 (") で囲む必要があります。

ファイルパスとディレクトリは、`OPTIONS` パラメータに含める必要があります。

`filepath` または `dir` のいずれかを指定する必要がありますが、両方を設定することはできません。

現在、エクスポートモードは `dir`（仮想ディレクトリで一致）のみをサポートしますが、ファイルパスはサポートしません。

パラメータ	説明
-------	----

filepath	<p>OSS 上のパスを示すファイル名。</p> <ul style="list-style-type: none"> - ファイル名にはパスが含まれていますが、バケット名は含まれていません。 - このパラメータは、OSS 上の対応するパスにある複数のファイルと一致し、データベースへのファイルロードをサポートします。 - filepath および filepath.x の形式で指定されたファイルは、データベースにインポートできます。filepath.x のxは 1 から始まり連続していなければなりません。たとえば、filepath、filepath.1、filepath.2、filepath.3、および filepath.5 です。最初の 4 つのファイルは一致してインポートされますが、filepath.5 という名前のファイルはありません。
dir	<p>OSS の仮想ディレクトリ。</p> <ul style="list-style-type: none"> - dir はスラッシュ (/)で終わらなければなりません。 - dir で指定された仮想ディレクトリ内のすべてのファイル（サブフォルダとサブフォルダ内のファイルを除く）は、データベースと照合され、データベースにインポートされます。
format	ファイル形式は現在 CSV のみ可能です。
encoding	ファイルデータのエンコーディング形式。UTF-8 などの一般的な PostgreSQL エンコーディング形式をサポートします。
parse_errors	エラー許容モードでの解析。ファイルの解析処理中に発生するエラーは、行ごとに見捨てられます。
delimiter	列に指定された区切り文字。
quote	指定されたファイルの引用符文字。
escape	指定したファイルのエスケープ文字。
null	指定された文字列に一致する列を無効にするために使用されます。たとえば、ヌルテストは、test 値でカラムを無効にするために使用されます。
force_not_null	1 つ以上の列の値を無効にするために使用されません。例えば、force_not_null'id'は空文字列の id という名前のヌル列を入力するために使用されません。

CREATE FOREIGN TABLE のエクスポートモードパラメータ

oss_flush_block_size および oss_file_max_size がエクスポート・モードに追加されます。

注：以下の 2 つのパラメータはインポートモードで無効です。

パラメータ	説明
oss_flush_block_size	一度に OSS に書き込まれるデータのバッファサイズ。デフォルト値：32 MB。値の範囲：1 MB～128 MB。
oss_file_max_size	OSS に書き込まれるデータの最大ファイルサイズ（最大ファイルサイズを超えると、後続のデータが別のファイルに書き込まれます）。デフォルト値：1,024 MB;値の範囲：8 MB～4,000 MB。

CREATE FOREIGN TABLE のその他の一般的なパラメータ

次のパラメータは、インポートおよびエクスポートモードでのエラー許容値に関連しています。

パラメータ	説明
oss_connect_timeout	接続タイムアウト時間（秒単位）。デフォルト値：10 秒。
oss_dns_cache_timeout	DNS タイムアウト時間（秒単位）。デフォルト値：60秒。
oss_speed_limit	最低許容率。デフォルト値：1,024 バイト/秒（1 Kbps）。
oss_speed_time	最大許容時間。デフォルト値は 15 秒です。

デフォルトのパラメータ値を使用すると、伝送速度が1秒間に 15 秒間連続して低下するとタイムアウトエラーが発生します。詳細については、この記事の末尾にある参照リンクを参照してください。4 つのパラメータは、サーバーオブジェクトで指定する必要があります。

oss_fdw インスタンス

```
# Create the plugin
create extension oss_fdw;

# Create a server instance
CREATE SERVER ossserver FOREIGN DATA WRAPPER oss_fdw OPTIONS
(host 'oss-cn-hangzhou.aliyuncs.com', id 'xxx', key 'xxx', bucket 'mybucket');

# Create an OSS external table
CREATE FOREIGN TABLE ossexample
```

```
(date text, time text, open float,
high float, low float, volume int)
SERVER ossserver
OPTIONS (filepath 'osstest/example.csv', delimiter ',',
format 'csv', encoding 'utf8', PARSE_ERRORS '100');

# Create a table, to which data is loaded
create table example
(date text, time text, open float,
high float, low float, volume int);

# Load data from ossexample to example.
insert into example select * from ossexample;

# As you can see
# oss_fdw estimates the file size on OSS and formulates a query plan correctly.
explain insert into example select * from ossexample;
QUERY PLAN
-----
Insert on example (cost=0.00..1.60 rows=6 width=92)
-> Foreign Scan on ossexample (cost=0.00..1.60 rows=6 width=92)
Foreign OssFile: osstest/example.csv.0
Foreign OssFile Size: 728
(4 rows)

# Write the data in the example table to OSS.
insert into ossexample select * from example;

explain insert into ossexample select * from example;
QUERY PLAN
-----
Insert on ossexample (cost=0.00..16.60 rows=660 width=92)
-> Seq Scan on example (cost=0.00..16.60 rows=660 width=92)
(2 rows)
```

oss_fdw 使い方のヒント

oss_fdw は、PostgreSQL FOREIGN TABLE フレームワークに基づいて開発された外部テーブルプラグインです。

データインポートのパフォーマンスは、PostgreSQL クラスタリソース（CPU IO MEM MET）および OSS に関連しています。

目的のデータのインポートのパフォーマンスのために、ossprotocol で ossendpoint は一致している必要がありますリージョンの PostgreSQL をクラウドに配置されている場所。詳細については、この記事の末尾にある[参照リンク](#)を参照してください。

エラー処理

PostgreSQL CREATE FOREIGN TABLEマニュアル

OSSエラー応答

付録

付録: ユーザーとスキーマの管理

通常 RDS では PostgreSQL スーパーユーザーを利用できないため、データベースを使用するには別のユーザーを作成し、そのユーザーのプライベートスペースをスキーマで管理することをお勧めします。

手順

注意: この例では、myuser はインスタンスと一緒に作成された管理アカウント、newuser はここで作成するアカウントです。

ログイン権限を持つユーザーを作成します。

```
CREATE USER newuser LOGIN PASSWORD 'password';
```

パラメーターは次のとおりです。

- USER: 作成するユーザーの名前 (例: *newuser*)
- password: ユーザー名に対応するパスワード (例: *password*)

新しいユーザーのスキーマを作成します。

```
CREATE SCHEMA newuser;  
GRANT newuser to myuser;  
ALTER SCHEMA newuser OWNER TO newuser;  
REVOKE newuser FROM myuser;
```

注意:

ALTER SCHEMA newuser OWNER TO newuserの前に newuser が myuser ロールに追加されていないと、次のような権限の問題が発生します。

```
ERROR: must be member of role "newuser"
```

- セキュリティを考慮して、オーナーの権限が付与されたら、newuser を myuser ロールから移動してください。

newuser を使用してデータベースにログインします。

```
psql -U newuser -h intranet4example.pg.rds.aliyuncs.com -p 3433 pg001
Password for user newuser:
psql.bin (9.4.4, server 9.4.1)
Type "help" for help.
```

リリースノート

リリースノート 20160801

PostGISは 2.1.7 から 2.2.2 にアップグレードされました。新しい PostGIS プラグインのデフォルトバージョンは 2.2.2 です。

次のコマンドを使用して、既存の PostGIS 2.1.7 プラグインをアップグレードできます。

注意：新しい PostGIS バージョンとアプリケーションとの間の互換性を避けるため、アップグレード前にアプリケーションテストを実行することをお勧めします。

```
-- Upgrade PostGIS (includes raster)
ALTER EXTENSION postgis UPDATE TO "2.2.2";
-- Upgrade Topology
ALTER EXTENSION postgis_topology UPDATE TO "2.2.2";
-- Upgrade US Tiger Geocoder
ALTER EXTENSION postgis_tiger_geocoder UPDATE TO "2.2.2";
```

リリースノート 20160701

構文

set は、set par1=val1, par2=val2 のような複数の変数をサポートしました。

rds discard all 構文がサポートされました（プロキシトランスペアレント接続プールのサポート、virtual pid と virtual cancel key のクリア）。

rds_superuser 作成のための新しい構文が追加されました。

```
CREATE ROLE | ALTER ROLE | CEATE GROUP xxx [WITH] RDS_SUPERUSER
```

高可用性

HA 透過スイッチ。再接続は必要ありません。

プロキシの透過性。

ストリームレプリケーション

複数のインスタンスの xlog データを LAN カードに同期させる際に、発生した競争問題を解決するために、WAL Sender レート制限機能が導入されました。

alidecode は、RDS から他のデータベースへの増分レプリケーション、または MySQL から RDS PG への完全なレプリケーションをサポートしました。

管理

logger ログの最大長は 2 KB に制限されたため、頻繁かつ長い SQL 文に起因するパフォーマンスへの影響を軽減しました。

RDS SUPERUSER は、プラグイン作成のために CREATE EXTENSION を実行することができました。

max_connect ソフトスイッチは、データベースクラスタを再起動せずに接続数を動的に調整するために導入されました。

PG インスタンスのメモリ使用を非同期に監視するために、OOM 信号が追加されました。メモリオーバーヘッドの低減ができます。

rds_superuser 権限を持つユーザーは、REASSIGN OWNED BY と他のコマンドを実行できます。

rds_superuser 権限のないユーザーが、データベースの作成中に pg_default として表領域を指定すると、エラーは返されません。

OOM 発生率は低減します。

ログによるストレージ不足問題は回避できます。

セキュリティ

PostgreSQL レプリケーションメカニズムによる HA 切り替え後のデータ損失を防止するため、hash index は b-tree index に自動的に変更され、unlogged table は共通 table に変更されます。

トリガ、ルール、または関数がトリガされた場合、一般ユーザーは rds_superuser パーミッションなしで CREATE EXTENSION または ALTER EXTENSION を実行します。

security definer トラップ（トリガーとルール）が修正されました。

暗号化されていないパスワードと pg_hba.conf パスワードの使用が無効になり、パスワードの複雑さの要件が増加します。

pg_authid MD5 コードセキュリティの脆弱性が修正されました。

パフォーマンス

データベースの最適化とデータファイルの事前配布がサポートされています。Inode 書き込みおよび I/O ハング時間が短縮されます。

チェックポイントが最適化されています。fsync 中に更新された dirty pages の量が減少します。data=ordered によるメタデータの書き込みの減少に従い、dirty pages 更新による I/O ハングの確率が低下します。

clog が最適化されています。clog buffer が増加しました。fsync がチェックポイント作成時に実行されます。

プラグイン

プラグインリストは下記となります。

コミュニティ版のプラグイン

```
plpgsql,  
pg_stat_statements,  
btree_gin,  
btree_gist,  
chkpass,  
citext,  
cube,  
dblink,  
dict_int,  
earthdistance,  
hstore,intagg,  
intarray,  
isn,  
ltree,  
pgcrypto,  
pgrowlocks,  
pg_prewarm,  
pg_trgm,  
postgres_fdw,  
sslinf,  
tablefunc,  
tsearch2,  
unaccent,  
pgstattuple,  
"uuid-oss" NOTE: uuid-oss must be enclosed by the double quotation mark (" ").
```

新しいプラグイン

```
postgis,  
postgis_topology,  
fuzzystrmatch,  
postgis_tiger_geocoder,  
plperl,  
pltcl,  
plv8,  
plls,  
plcoffee,  
zhparser, which supports custom word segmentation  
pgrouting,  
rdkit,  
pg_hint_plan,  
jsonb,  
www_fdw,  
oss_fdw,  
pg_rewind
```

dblink と postgres_fdw を使用して、このインスタンスの他のデータベースにアクセスします。

モニターリング

エラー

- Database error log

スペース

- Available space, data directory space, XLOG directory space (archived and unarchived)

ジャンクデータ

Table expansion

Index expansion

Deadtuple

Unreferenced large object

実行環境

Database age

Long transaction and 2PC

Sequence depletion

Unlogged table

Hash index

パフォーマンスビュー

Standby database delay

Stream replication SLOT delay

Cache hit rate

Transaction rollback percentage

Lock wait

Slow SQL

TOP SQL

Connections

Instance memory usage

Instance CPU usage

Instance IOPS usage

配置

Password expiration time

Master configuration, backup configuration inconsistent

Master configuration file, backup configuration file inconsistent