Accessing PostgreSQL on Amazon Web Services RDS

Summary: information about accessing a PostgreSQL instance hosted on Amazon Web Services RDS.

Introduction

CAST supports PostgreSQL instances deployed via **Amazon Web Services RDS** - some recommendations for accessing these instances for use with CAST AIP are listed below.

postgres database

CAST products all require the CAST AIP schemas to be installed within a database called "postgres" - this is therefore also a requirements for PostgreSQL instances deployed on **Amazon Web Services RDS**.

Users

By default the **rds_superuser** will be made available within the Amazon Web Services RDS PostgreSQL instance (see **https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Appendix.PostgreSQL.CommonDBATasks.html**) - therefore, based on that you can create additional users. The equivalent CAST Storage Service provided by CAST includes two default users as follows:

Username	Password	Permissions	Notes
operator	CastAIP	SUPERUSER	-
guest	WelcomeToAIP	-	Note that in the CAST AIP **8.3.11**, the "guest" user is no longer used.

If you would like to create these users in the Amazon Web Services RDS environment, use the following commands:

```
create user operator with rds_superuser password 'CastAIP';
create user guest with password 'WelcomeToAIP';
grant postgres to operator;
```

You can also create users without the **rds_superuser** role, for example:

```
create user css_administrator with password 'CastAdminAIP' nosuperuser;
grant create, connect, temporary on database postgres to css_administrator;
```

SSL access

You should also read **SSL encrypted mode configuration for CAST Storage Service and PostgreSQL**.

Just like an on premises PostgreSQL/CAST Storage Service instance, Amazon Web Services RDS supports Secure Socket Layer (SSL) encryption for PostgreSQL instances. Using SSL, you can encrypt a PostgreSQL connection between your applications and your PostgreSQL DB instances. You can also force all connections to your PostgreSQL DB instance to use SSL. The configuration

To connect to a PostgreSQL DB instance over SSL:

1. Download the certificate. For information about downloading certificates, see **Using SSL/TLS to encrypt a connection to a DB instance** (third-party information).
2. Import the certificate into your operating system. For sample scripts that import certificates, see **Sample script for importing certificates into your trust store** (third-party information).
3. Connect to your PostgreSQL DB instance over SSL.

When you connect using SSL, your client can choose whether to verify the certificate chain. If your connection parameters specify `sslmode=verify-ca` or `sslmode=verify-full`, then your client requires the RDS CA certificates to be in their trust store or referenced in the connection URL. This requirement is to verify the certificate chain that signs your database certificate. Use the `sslrootcert` parameter to reference the certificate, for example `sslrootcert=rds-ssl-ca-cert.pem`

For example, the entry for AWS SSL RDS in the **SSLParameters.ini** looks like this:

```
[<AWS_RDS_HOST>:<AWS_RDS_PORT>]
ssl=true
sslmode=verify-ca
sslrootcert=<Local_path>\rds-ssl-ca-cert.pem
ssljdbckey=
sslkey=
sslcert=
sslrootcer=
sslcer=
```