PostgreSQL Migration - New

Last updated by | Eduardo Santana | Feb 24, 2022 at 4:48 AM PST

Migration to Azure Database for PostgreSQL can be done with different methods:

1. Dump/restore:

https://docs.microsoft.com/en-us/azure/postgresql/howto-migrate-using-dump-and-restore

2. Export/ Import:

https://docs.microsoft.com/en-us/azure/postgresql/howto-migrate-using-export-and-import

3. DMS:

https://docs.microsoft.com/en-us/azure/dms/tutorial-azure-postgresql-to-azure-postgresql-online-portal

https://docs.microsoft.com/en-us/azure/dms/tutorial-postgresql-azure-postgresql-online-portal

https://docs.microsoft.com/en-us/azure/dms/tutorial-postgresql-azure-postgresql-online

pg_dump / pg_restore

https://www.postgresql.org/docs/11/app-pgdump.html

https://www.postgresql.org/docs/11/app-pgrestore.html

pg_dump is used to dump the database using the following command:

pg_dump -h source-server-name.postgres.database.azure.com -U source-server-username@source-server-name -Fc -d source-databasename -f path_to_file ex: Z:\Data\Backups\database-backup.dump

pg_restore is used to restore the database from a dump using the following command:

pg_restore -h target-server-name.postgres.database.azure.com -U target-server-username@target-server-name -Fc -j 4 -d target-databasename path_to_file ex: Z:\Data\Backups\database-backup.dump

pg_dump / psql

pg_dump -h source-server-name.postgres.database.azure.com -U source-server-username@source-server-name -d source-databasename -f path_to_file ex: Z:\Data\Backups\database-backup.sql

psql --host=target-server-name.postgres.database.azure.com --username=target-server-username@target-server-name --dbname=target-databasename --file="path_to_file" ex: "Z:\Data\Backups\database-backup.sql"

Restoring a database might need the following two grants to work with no issues. Please check if the customer is getting an error to use superuser to continue the restore process:

GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO your_username; GRANT ALL PRIVILEGES ON ALL SEQUENCES IN SCHEMA public TO your_username;

Recommendation on data restore:
https://www.postgresql.org/docs/current/static/populate.html

Please keep in mind that with pg_dump we are only migrating databases data. However there are global objects that are common to all databases (database roles and tablespaces) which are not backup with pg_dump. If you need to save and restore this objects you should use pg_dumpall and psql.

https://www.postgresql.org/docs/11/app-pg-dumpall.html

https://www.postgresql.org/docs/11/app-psql.html

pg_dumpall -v --host=source-server-name.postgres.database.azure.com --username=source-server-username@source-server-name --globals-only -f path_to_file ex: Z:\Data\Backups\pg_dumpall_globals.sql

You can choose to dump only the roles by choosing below option: -r, --roles-only dump only roles, no databases or tablespaces

psql –host=target-server-name.postgres.database.azure.com –username=target-server-username@target-server-name --dbname=postgres --file=="path_to_file" ex: "Z:\Data\Backups\pg_dumpall_globals.sql"

The dump script is expected to run with some errors because it will execute CREATE ROLE for every role existing in the source server. So, you will get a “role already exists” error for the Admin user unless the destination server was created with a different Admin user name, and also for azure roles (ex: superuser).

ERROR: role "azure_pg_admin" already exists ERROR: role "azure_superuser" already exists ERROR: role " your_user_name" already exists

You can also edit the .sql file and remove the already existing roles.