

# PostgreSQL (Linux)

Deploy PostgreSQL database (Linux)

Version 8.0



This documentation is provided under restrictions on use and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this documentation, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

# Table of Contents

<b>Deploy PostgreSQL database (Linux)</b>	<b>4</b>
I. Install PostgreSQL	4
II. Create PostgreSQL user	5
III. Restore PostgreSQL database	6
IV. Change the database owner (optional)	8

# Deploy PostgreSQL database (Linux)

PRODUCTS: [ALL CREATIO PRODUCTS](#)

Use one of two database configurations to deploy Creatio:

- Use a remote DBMS (recommended)
- Use a local PostgreSQL server.

If you already have a PostgreSQL server running on the intended machine, skip to step II.

If you have set up sysadmin (with privileges to log in, create and modify databases) and public (unprivileged) user roles, skip to step III.

## I. Install PostgreSQL

PostgreSQL is unavailable in most standard repositories. To install PostgreSQL on Linux:

1. Log in as root:

```
sudo su
```

2. Add the PostgreSQL repository:

```
echo -e "deb http://apt.postgresql.org/pub/repos/apt/ $(lsb_release -sc)-pgdg main" > /etc/ap
```

3. Import the signing key of the PostgreSQL repository:

```
wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | apt-key add -
```

4. Update the package lists:

```
apt-get update
```

5. Install PostgreSQL:

```
apt-get install -y postgresql-12
```

6. Log out from your root session:

```
exit
```

**Note.** Please refer to the [PostgreSQL documentation](#) for details on PostgreSQL clustering.

## II. Create PostgreSQL user

A fresh installation of PostgreSQL is not ready for deploying Creatio immediately. If you plan to use a fresh installation of PostgreSQL, you need to create a user that can log in to the database using a password and has sufficient privileges to create and update a database. By default, no such user will be available.

We recommend to create two database users in PostgreSQL:

- A user with the “**sysadmin**” role, who has maximum access privileges on the database server level. This user will restore the Creatio database from a backup file and assign access permissions.
- A user with the “**public**” role, whose permissions are limited. You will need this user to set up a secure connection to the restored Creatio database using PostgreSQL authentication.

If your PostgreSQL instance already has sysadmin (privileged) and public (unprivileged) user roles, skip this step.

To create PostgreSQL users:

1. Log in as **postgres**:

```
sudo su - postgres
```

2. Open PostgreSQL shell:

```
psql
```

3. Create a sysadmin user:

```
CREATE USER pg_sysadmin;
```

**pg\_sysadmin** – user who will be granted sysadmin privileges. This user will restore the Creatio database from a backup file and assign access permissions

4. Make **pg\_sysadmin** a system administrator:

```
ALTER ROLE pg_sysadmin WITH SUPERUSER;
```

5. Allow **pg\_sysadmin** to log in:

```
ALTER ROLE pg_sysadmin WITH LOGIN;
```

6. Set a password for **pg\_sysadmin**:

```
ALTER ROLE pg_sysadmin WITH PASSWORD 'pg_syspassword';
```

**pg\_password** – sysadmin user password for connecting to the PostgreSQL server.

7. Create a public user:

```
CREATE USER pg_user;
```

**pg\_user** – public user for connecting to the PostgreSQL server. You will need this user to set up a connection to the restored Creatio database.

8. Allow **pg\_user** to log in:

```
ALTER ROLE pg_user WITH LOGIN;
```

9. Set a password for **pg\_user**:

```
ALTER ROLE pg_user WITH PASSWORD 'pg_password';
```

**pg\_password** – public user password for connecting to the PostgreSQL server.

10. Exit the PostgreSQL shell:

```
\q
```

11. Log out from your postgres session:

```
exit
```

### III. Restore PostgreSQL database

To restore a PostgreSQL database from a backup file, you will need **psql** and **pg\_restore** utilities. Both are part of the **postgresql-client-common** package.

If you install **postgresql-12** locally using **apt-get**, APT will install **postgresql-client-common** as a dependency for **postgresql-12**.

If you plan to use a remote PostgreSQL database without installing the PostgreSQL DBMS on your server, install the **postgresql-client-common** package manually by running:

```
sudo apt-get install postgresql-client-common
```

To restore the Creatio database from a backup file:

1. Enter DB connection password in the environment variable:

```
export PGPASSWORD=pg_syspassword
```

**pg\_syspassword** – pg\_sysadmin user password for connecting to the PostgreSQL server.

2. Create a database where the backup data will be restored:

```
psql --host=pg_server_address --port=pg_server_port --username=pg_sysadmin --dbname=pg_dbname
```

**pg\_server\_address** – PostgreSQL server address

**pg\_server\_port** – PostgreSQL server port

**pg\_sysadmin** – sysadmin user for connecting to the PostgreSQL server

**pg\_dbname** – name of the PostgreSQL DB where the instructions will be executed

**Note.** If you have not created any databases yet or an attempt to connect to a database triggers the “FATAL: database “pg\_dbname” does not exist” error, use the default database “template1”.

**pg\_dbname\_creatio** – name of the PostgreSQL DB which will host Creatio tables

**pg\_user** – the “public” user who will be granted permission to use and update the Creatio database

3. If you are using AWS RDS:
  - a. Download the [ChangeTypesOwner.sql](#) script.
  - b. In the script, replace the “postgres” value with a valid Postgres username.
  - c. Run the updated ChangeTypesOwner.sql script.
4. Navigate to the application directory:

```
cd /path/to/application/directory/
```

**/path/to/application/directory/** – the directory with Creatio setup files.

5. Navigate to the database directory:

```
cd db
```

- Restore the database from the backup file:

```
pg_restore --host pg_server_ip --port pg_server_port --username=pg_sysadmin --dbname=pg_dbnam
```

**pg\_server\_address** - PostgreSQL server address

**pg\_server\_port** - PostgreSQL server port

**pg\_sysadmin** - sysadmin user for connecting to the PostgreSQL server

**pg\_dbname\_creatio** - name of the PostgreSQL DB to insert backup tables. Use the name you specified in the "CREATE DATABASE" command on step 2.

- Download the [CreateTypeCastsPostgreSql.sql file](#).

- Execute type conversion:

```
psql --host=pg_server_address --port=pg_server_port --username=pg_sysadmin --dbname=pg_dbname
```

**pg\_server\_ip** - PostgreSQL server address

**pg\_server\_port** - PostgreSQL server port

**pg\_sysadmin** - sysadmin user for connecting to the PostgreSQL server

**pg\_dbname\_creatio** - name of the PostgreSQL DB where the instructions will be executed

**/path/to/CreateTypeCastsPostgreSql.sql** - path to the CreateTypeCastsPostgreSql.sql file.

## IV. Change the database owner (optional)

Creatio lets you replace the owner of the database and its objects to a non-administrator user (not a superuser) during the restoration. Use the ChangeDbObjectsOwner script for that. For Postgres version 10 and earlier: [Download the script](#). For Postgres version 11 and later: [Download the script](#).

To restore the database on behalf of a non-administrator user:

- Replace the database owner:

```
psql --host pg_server_address --port pg_server_port --username=pg_sysadmin --dbname=pg_dbname
```

**pg\_server\_address** - PostgreSQL server address

**pg\_server\_port** - PostgreSQL server port

**pg\_sysadmin** - sysadmin user for connecting to the PostgreSQL server This user must be an administrator (superuser) or have the "ALTER DATABASE" privileges.

**pg\_user** - the placeholder to replace with the actual username of the new database owner. You will need this user to set up a connection to the Creatio database.

**pg\_dbname\_creatio** - the name of the database whose owner is being changed.

- Replace the owner of the database objects:



```
psql --host pg_server_address --port pg_server_port --username=pg_sysadmin --dbname=pg_dbname
```

**pg\_server\_address** – PostgreSQL server address

**pg\_server\_port** – PostgreSQL server port

**pg\_sysadmin** – sysadmin user for connecting to the PostgreSQL server This user must be an administrator (superuser) or the Creatio database owner.

**pg\_user** – the placeholder to replace with the actual username of the new database owner. You will need this user to set up a connection to the Creatio database.

**pg\_dbname\_creatio** – the name of the database whose owner is being changed.

**/path/toChangeDbObjectsOwner.sql** – the path to the previously saved toChangeDbObjectsOwner.sql file.

You can ignore this step. In that case, the user who ran the pg\_restore command will remain the owner of the database and its objects. Normally, this is the postgres user.