

Synopsis

This post will describe the steps required to complete the initial configuration of the PostgreSQL DBMS on a system running FreeBSD. Configuring the basic configuration of PostgreSQL and creating an initial super-user will be covered; performance tuning, database administration, and Structured Query Language (SQL) will not be covered in this post.

Installation

Configure and install the PostgreSQL v8.4.x DBMS from the ports collection. The default port configuration will work just fine, although you may consider enabling the *OPTIMIZED_CFLAGS* option for better performance.

```
# cd /usr/ports/databases/postgresql84-server      sprawdź w portach czy mamy najnowszą bazę
# make config                                     postgresql xxx-server
# make install clean                               i dopiero tą instalować
```

Configuration

Enable PostgreSQL to start at system boot in */etc/rc.conf*.

```
# echo 'postgresql_enable="YES"' >> /etc/rc.conf
```

Initialize the PostgreSQL database cluster for the first time. NOTE: The following command will create the initial database cluster in the */usr/local/pgsql/data* directory by default.

```
# /usr/local/etc/rc.d/postgresql initdb
/usr/local/etc/rc.d/postgresql initdb
The files belonging to this database system will be owned by user "pgsql".
This user must also own the server process.
```

```
The database cluster will be initialized with locale C.
The default text search configuration will be set to "english".
```

```
fixing permissions on existing directory /usr/local/pgsql/data ... ok
creating subdirectories ... ok
selecting default max_connections ... 40
selecting default shared_buffers ... 28MB
creating configuration files ... ok
creating template1 database in /usr/local/pgsql/data/base/1 ... ok
initializing pg_authid ... ok
initializing dependencies ... ok
creating system views ... ok
loading system objects' descriptions ... ok
creating conversions ... ok
```

```
creating dictionaries ... ok
setting privileges on built-in objects ... ok
creating information schema ... ok
vacuuming database template1 ... ok
copying template1 to template0 ... ok
copying template1 to postgres ... ok
```

WARNING: enabling "trust" authentication for local connections
You can change this by editing `pg_hba.conf` or using the `-A` option the next time you run `initdb`.

Success. You can now start the database server using:

```
/usr/local/bin/postgres -D /usr/local/pgsql/data
or
/usr/local/bin/pg_ctl -D /usr/local/pgsql/data -l logfile start
```

Configure PostgreSQL to listen for database connections on all system IP addresses by adding the following line to `/usr/local/pgsql/data/postgresql.conf`.

```
listen_addresses = '*'
```

Configure PostgreSQL to use password hash authentication for all hosts and users connecting from the local network by adding the following line to the `/usr/local/pgsql/data/pg_hba.conf` file. NOTE: Replace `10.0.1.0/24` with your own network.

```
host all all 10.0.1.0/24 md5
```

Tu zmienić na własną podsieć gdzie hosty będą ją osiągały

Start the PostgreSQL DBMS for the first time and add a new super-user (with database and role creation rights) by executing the following commands.

```
# /usr/local/etc/rc.d/postgresql start
# su postgres
$ createuser -sdrP username
Enter password for new role: *****
Enter it again: *****
```

```
$ exit
```

Testing

The PostgreSQL DBMS should now be up and running with the newly created super-user. Using a PostgreSQL client, such as *pgadmin3* connect to the PostgreSQL server from another system using the username and password of the role previously created. New database schemas, roles, procedures, etc can now be created using the super-user.

References

- [PostgreSQL: Documentation Manuals: PostgreSQL 8.4.](#)
- [PGAdmin – PostgreSQL Administrator.](#) Zainstalować program do zarządzania naszą bazą.