

xCAT 2 Setup PostgreSQL

07/21/10, 03:19:20 PM

Table of Contents

1.0 Switching to PostgreSQL Database on Management Node	1
1.1 Install PostgreSQL	1
1.1.1 Install Postgresql on Linux	1
1.1.2 Install PostgreSQL on AIX MN (xCAT 2.5 release or later)	2
1.2 Setup PostgreSQL on AIX and Linux	2
1.2.1 Using the pgsqsetup script (xCAT 2.5 or later)	2
1.2.2 Manually setup PostgreSQL	3
1.3 Migrate your database to PostgreSQL	5
1.4 Using Postgresql (psql command line interface)	6

1.0 Switching to PostgreSQL Database on Management Node

One reason to migrate from the default SQLite database to PostgreSQL with xCAT is for xCAT hierarchy using Service Nodes. PostgreSQL provides the ability for remote access to the xCAT database on the Management node which is required by Service Nodes. PostgreSQL also support IPV6.

To set up the postgresql database on the Management Node follow these steps.

This example assumes:

- 11.16.0.1: IP of management node (cluster-facing NIC)
- xcatdb: database name
- xcatadm: database role (aka user)
- cluster: database password
- 11.16.1.230 & 11.16.2.230: service nodes (mgmt node facing NIC)

Substitute your addresses and desired userid , password and database name as appropriate.

1.1 Install PostgreSQL

1.1.1 Install Postgresql on Linux

The postgresql rpms are part of the base Linux OS. You should find for example the following rpms installed.

```
postgresql-libs-*  
postgresql-server-*
```

postgresql-*

In addition you need to install the Perl-DBD code from the OS.

Perl-DBD-Pg*

1.1.2 Install PostgreSQL on AIX MN (xCAT 2.5 release or later)

Before you install make sure you have enough space, PostgreSQL will be installed in /var/lib/pgsql and needs about 90 mbytes for the code + the size needed for the xCAT database.

As root:

- Down load PostgreSQL rpms package from the following location:http://sourceforge.net/projects/xcat/files/xcat-dep/2.x_AIX
- Unzip and untar in the location of your choice.

```
gunzip xcat-postgresql*.gz  
tar -xvf xcat-postgresql*.tar
```

Read the README file for installation instructions, and install the two rpms on the AIX Management Node that are appropriate for your OS level.

1.2 Setup PostgreSQL on AIX and Linux

1.2.1 Using the pgsqsetup script (xCAT 2.5 or later)

You may use the pgsqsetup script to setup xCAT on PostgreSQL instead of following the manual steps in the Manually setup PostgreSQL.

See man pgsqsetup for information on running the script. The script will complete all actions in the Manually setup PostgreSQL section , including the Management Nodes ip address in the pg_hba.conf file.

After the automatic setup is complete, you need to

- add additional ip addresses to the pg_hba.conf file for each Service Node.
- add set listen_addresses = '*' to the postgresql.conf file.
- Stop and start postgresQL, if you edit those files.
Look in the “#setup by xCAT” section of the postgresql.conf file to change.
See Setup the PostgreSQL configuration files .

1.2.2 Manually setup PostgreSQL

Note: you can use `pgsqlsetup` to do all this work, if you are running 2.5 or later release. See Using the `pgsqlsetup` script (xCAT 2.5 or later).

As root: Stop the `xcatd` daemon during the database migration:

```
AIX:
    stopsrc -s xcatd
```

```
Linux:
    service xcatd stop
```

On AIX create the needed PostgreSQL ids:

- Create the postgres id that will administer the PostgreSQL server
`mkgroup postgres`
`mkuser pgrp=postgres home=/var/lib/pgsql postgres`
- `passwd postgres` (assign a password this is optional)
- Create the xcatadm id that will own the xcatdb in PostgreSQL
`mkuser xcatadm`
- `passwd xcatadm` (assign temp password with root)
`su - xcatadm`
- `passwd` (assign permanent password that will be used in the `/etc/xcat/cfgloc` file)
- Create the directory for the databases and make postgres the owner
as root:
`mkdir /var/lib/pgsql/data`
`chown postgres /var/lib/pgsql/data`
`chgrp postgres /var/lib/pgsql/data`
- `su - postgres`
- `pwd` (are you in `/var/lib/pgsql`)
- Setup `.profile`
Add paths needing to run DB commands
`MANPATH=/usr/local/pgsql/man:$MANPATH`
`export MANPATH`
`PATH=/usr/local/pgsql/bin:$PATH`
`export PATH`

On AIX as postgres Create a database installation by running the following:

```
/var/lib/pgsql/bin/initdb -D /var/lib/pgsql/data
```

You should get the following message “Success. You can now start the database...”

On Linux as root run the following to create the Database installation:

service postgresql initdb to initialize the database

Setup the PostgreSQL configuration files

On AIX or Linux as root:

vi /var/lib/pgsql/data/pg_hba.conf

Lines should look like this (with your IP addresses substituted). Add all nodes that need to access the database.

```
local all all ident sameuser
# IPv4 local connections:
host all all 127.0.0.1/32 md5
host all all 11.16.0.1/32 md5
host all all 11.16.1.230/32 md5
host all all 11.16.2.230/32 md5
```

For example, where 11.16.0.1 is the MN and 11.16.1.230 and 11.16.2.230 are service nodes.

vi /var/lib/pgsql/data/postgresql.conf

set listen_addresses = '*' # This allows remote access from all ips

Note: be sure and uncomment the line.

The following logging setup is the default on Linux, but should be set on AIX also.

```
logging_collector = on
log_directory = 'pg_log'
log_filename = 'postgresql-%a.log'
log_truncate_on_rotation = on
log_rotation_age = 1d
log_rotation_size = 0
log_min_messages = notice
```

If you are working on large systems, you may need to set the max_connections attribute in the file. This is the number of connections that can be make to the database at one time. If you are using service nodes, it is recommended that you

```
set max_connections = 1000
```

- Start/Stop the PostgreSQL server

On AIX:

su – postgres

start the server:

```
/var/lib/pgsql/bin/pg_ctl -D /var/lib/pgsql/data start
If you need to stop the server
/var/lib/pgsql/bin/pg_ctl -D /var/lib/pgsql/data stop
```

Note: you can get the message “\$ LOG: could not bind IPv6 socket: Address already in use
HINT: Is another postmaster already running on port 5432? If not, wait a few seconds and
retry after setting listen_addresses = '*' , it can be ignored.

On Linux as root:

```
service postgresql start
```

To stop/stop postgresql:

```
service postgresql start
service postgresql stop
```

On AIX and Linux:

```
su – postgres:
```

Create the xcatadm userid in the database and set to own xcatdb

```
/var/lib/pgsql/bin/createuser -SDRP xcatadm
```

(Will prompt for a password, use the same one that you input for the AIX xcatadm id. Note:
this xcatadm unix id does not have to exist on Linux, only in the database.)

Create the xcatdb database owned by xcatadm

```
/var/lib/pgsql/bin/createdb -O xcatadm xcatdb
```

```
exit ( back to root)
```

1.3 Migrate your database to PostgreSQL

Note: the postgresqlsetup script will do this for you also, if you choose to use it.

Backup your database to migrate to the new database. (This is required even if you have not added anything to your xCAT database yet. Required default entries were created when the xCAT RPMs were installed on the management node which, and they must be migrated to the

new postgresql database.)

```
mkdir -p ~/xcat-dbback
XCATBYPASS=1 dumpxCATdb -p ~/xcat-dbback
```

1. /etc/xcat/cfgloc file should contain the following line, substituting your specific info. This points the xCAT database access code to the new database.

```
Pg:dbname=xcatdb;host=11.16.0.1|xcatadm|cluster
```

change to allow only root access:

```
chmod 0600 /etc/xcat/cfgloc
```

2. Restore your database to postgresql (bypass mode runs the command without xcatd):

```
XCATBYPASS=1 restorexCATdb -p ~/xcat-dbback
```

3. Start the xcatd daemon using the postgresql database

AIX:

```
startsrc -s xcatd
```

Linux:

```
service xcatd start
```

1.4 Using Postgresql (psql command line interface)

If you want to access the database through the Postgresql (psql) command, to check the database, enter the following:

su – postgres:

```
psql -h <hostname> -U xcatadm -d xcatdb ( note hostname must match ip in the pg_hba.conf file)
and you will be prompted for the password ( cluster).
```

You can then run sql commands on the database.

Run \h for a list of commands

Run \g so SQL commands can end in ;

Then

```
select * from nodelist; to see table entries
```

```
\dt list all tables;
```

\q to quit