

Laboratorio

En este laboratorio comenzarás a practicar con cambios en los ficheros de configuración para mejorar el servidor PostgreSQL.

Nota:

Las tareas se realizará sobre una máquina virtual VMWare y un sistema operativo Ubuntu o en un entorno con PostgreSQL ya instalado.

Práctica 1: Configuración de PostgreSQL

Objetivos

El objetivo de la práctica es aprender a realizar cambios en la configuración de PostgreSQL.

Notas

Para facilitar las prácticas, se ofrece una máquina virtual ya creada.
La máquina virtual descargada tiene la siguiente información:

- Usuario: postgresql96
- Contraseña: postgresql

Para utilizar la máquina virtual puede utilizar el software gratuito VMWare Player.

Todas las tareas deberán ser ejecutadas en la consola de Linux (Aplicación Terminal).

Tareas

- 1.- Crear otro cluster, cuyo PGDATA sea /curso/postgresql/config/. Cambiar el puerto donde escucha Postgres al 5434. Arrancar el servidor y verificar que se puede conectar a dicho puerto.
- 2.- Ajustar la memoria compartida de postgres a $\frac{1}{4}$ de la RAM de la máquina. Verificar el cambio con un comando SHOW.
- 3.- Mostrar la memoria que postgres reserva para operaciones de vacuum e indexado. Cambiamos el valor del parámetro en el fichero postgresql.conf a 32MB.
- 4.- Supongamos que el servidor que estamos configurando puede ser accedido por hasta 8 servidores de aplicaciones, cada uno de los cuales tendrá un pooler local configurado con un máximo de 50 conexiones cada uno. Ajustar la configuración de postgres para asegurar el correcto funcionamiento en este escenario.

- 5.- En una base de datos con alta tasa de inserciones/actualizaciones/borrados se detecta en el log de forma recurrente el mensaje “checkpoints are occurring too frequently”. Modificar el parámetro de configuración adecuado para solucionarlo. ¿Qué implicaciones tiene?
- 6.- Ajustar la configuración de postgres para que genere los logs en un fichero en disco llamado “logs/pglog-YYYY-MM-DD-HH-MM.log” (‘pglog-%Y-%m-%d-%H-%M.log’). En esta ocasión realiza el cambio con el comando `alter system`. Comprueba los cambio (**tail postgresql.auto.conf**) y reinicia el servidor cuando hayas terminado.
- 7.- Cambiar el mínimo nivel de emisión de logs del servidor a “notice”.
- 8.- Asegurar que en los logs se incluyen:
 - 1.- Los checkpoints.
 - 2.- Conexiones y desconexiones.
 - 3.- Consultas que duren más de 5 segundos.
 - 4.- En cada log, un prefijo (formato libre) con el host remoto y nombre de la aplicación remota.

Soluciones

1. Crear otro cluster, cuyo PGDATA sea /curso/postgresql/config/. Cambiar el puerto donde escucha Postgres al 5434. Arrancar el servidor y verificar que se puede conectar a dicho puerto.

```
initdb -D /curso/postgresql/config -E UTF8
nano /curso/postgresql/config/postgresql.conf
port = 5434                # (change requires restart)
pg_ctl -D /curso/postgresql/config/ start
psql -p 5434 postgres ó
psql -p 5434 -h localhost postgres
```

2. Ajustar la memoria compartida de postgres a 1/4 de la RAM de la máquina. Verificar el cambio con un comando SHOW.

```
shared_buffers = 512MB      # ó 1/4 de la RAM de la máquina
pg_ctl -D /curso/postgresql/config/ restart
psql -p 5434 postgres
SHOW shared_buffers;
```

3. Cambiar a 32MB la memoria que postgres reserva para operaciones de vacuum e indexado.

```
maintenance_work_mem = 32MB
pg_ctl -D /curso/postgresql/config/ reload
```

4. Supongamos que el servidor que estamos configurando puede ser accedido por hasta 8 servidores de aplicaciones, cada uno de los cuales tendrá un pooler local configurado con un máximo de 50 conexiones cada uno. Ajustar la configuración de postgres para asegurar el correcto funcionamiento en este escenario.

```
max_connections = 410      # o alguna cantidad superior a 8*50=400, para
dejar para conexiones locales/consola/etc
pg_ctl -D /curso/postgresql/config/ restart
```

5. En una base de datos con alta tasa de inserciones/actualizaciones/borrados se detecta en el log de forma recurrente el mensaje "checkpoints are occurring too frequently". Modificar el parámetro de configuración adecuado para solucionarlo. ¿Qué implicaciones tiene?

```
checkpoint_timeout = 30s          # el valor debería probarse y medirse, logeando cada
cuántos segundos se producen checkpoints
pg_ctl -D /curso/postgresql/config/ reload
```

6. Ajustar la configuración de postgres para que genere los logs en un fichero en disco llamado "logs/pglog-YYYY-MM-DD-HH-MM.log" ('pglog-%Y-%m-%d-%H-%M.log'). En esta ocasión realiza el cambio con el comando alter system. Comprueba los cambios (**tail postgresql.auto.conf**) y reinicia el servidor cuando hayas terminado.

Utilizamos el comando ALTER SYSTEM, la sintaxis podemos extraerla escribiendo \h ALTER SYSTEM

```
postgres=# \h alter system
Command:      ALTER SYSTEM
Description:  change a server configuration parameter
Syntax:
ALTER SYSTEM SET configuration_parameter { TO | = } { value | 'value' | DEFAULT }
ALTER SYSTEM RESET configuration_parameter
ALTER SYSTEM RESET ALL
```

Por último ejecutamos los cambios.

```
postgres=# alter system set logging_collector = on ;
ALTER SYSTEM
postgres=# alter system set log_directory = 'logs';
ALTER SYSTEM
postgres=# alter system set log_filename = 'pglog-%Y-%m-%d-%H-%M.log';
ALTER SYSTEM
```

Comprobamos que en el fichero postgresql.auto.conf se encuentran los cambios indicados por el comando alter system

```
postgres196@ubuntu:/curso/postgresql/config$ tail postgresql.auto.conf
# Do not edit this file manually!
# It will be overwritten by ALTER SYSTEM command.
logging_collector = 'on'
log_directory = 'logs'
log_filename = 'pglog-%Y-%m-%d-%H-%M.log'
```

También podemos observar que en fichero postgresql.conf no se encuentra cambiado:

```
# This is used when logging to stderr:
#logging_collector = off
```

Cuando reiniciamos podemos comprobar que el parámetro ha sido cambiado:

```
postgres=# \c
You are now connected to database "postgres" as user "postgresl96".
postgres=# show logging_collector;
 logging_collector
-----
 on
(1 row)

postgres=#
server stopped
server starting
postgresl96@ubuntu:/curso/postgresql/config$ LOG:  redirecting log output to logging collector process
HINT:  Future log output will appear in directory "logs".
```

Además, se ha creado un nuevo directorio llamado logs donde se están guardando los ficheros logs con el formato establecido.

```
postgresl96@ubuntu:/curso/postgresql/config$ cd logs/
postgresl96@ubuntu:/curso/postgresql/config/logs$ ls
pglog-2017-05-15-13-22.log
```

7. Cambiar el mínimo nivel de emisión de logs del servidor a "notice".

```
log_min_messages = notice
pg_ctl -D /curso/postgresql/config/ reload
```

8. Asegurar que en los logs se incluyen:

1. Los checkpoints.
2. Conexiones y desconexiones.
3. Consultas que duren más de 5 segundos.
4. En cada log, un prefijo (formato libre) con el host remoto y nombre de la aplicación remota.

```
log_checkpoints = on
log_connections = on
log_disconnections = on
log_min_duration_statement = 5000
log_line_prefix = '%h - %a'
```