**EJERCICIO BACKUP RESTORE POSTGRE PARA TP2**

**EJERCICIO 1**

1. ELEGIR UNA BASE DE DATOS A (Investigar)
2. BACKUP DE A (investigar): 2 formas
   1. **pg\_dump a > a.sql O**
   2. **pg\_dump -p (si hay mas de un cluster (mas de un server) debo decir a que puerto)**
3. **CREAR TABLA LABOSABADO;**
4. ELIMINAR BASE DE DATOS A
5. CREAR BASE DATOS A;
6. Restaurar la base de datos A con el comando
   1. **psql a < a.sql**

**EJERCICIO 2 (Mover tabla de un cluster a otro)**

**Mover tabla t1 de 9.1 a 9.3**

1. **Crear tabla t1 en 9.1 (Cluster) y cargar 2 registros**
2. **pg\_dump -t t1 > t1.sql**
3. **Con psql ejecutar el t1.sql en el cluster 9.3**
4. **Verificar que t1 exista y tenga 2 registros en 9.3**

**EJERCICIO 3 (BACKUP CONTINUO)**

1. Crear carpeta /tmp/repo\_wal
2. Hacer
   1. Cual es mi archivo conf: psql show\_all (parametro config\_file): **/etc/postgresql/9.5/main/postgresql.conf**
   2. Editar el archivo -> sudo nano postgresql.conf
      1. archive\_mode = on
      2. archive\_command = **‘cp %p /tmp/repo\_wal/%f’**
      3. wal\_level = ‘archive’
   3. Reiniciar servicio
      1. /etc/init.d/postgresql restart
3. Insertar datos con generate\_series (muchos 2.000.000)
4. Verificar Mirar carpeta -> /tmp/repo\_wal (**Verificar que tenga permisos de escritura para todos -> chmod 777 /tmp/repo\_wal)**

**EJERICIO 4 (BACKUP FULL)**

1. PSQL SELECT PG\_START\_BACKUP(‘sabado’);
2. Backup tar
   1. tar cvf mibackup.tar [Carpeta PG DATA] (PARA SABER CARPETA PG\_DATA MIRAR PSQL show all o show data\_directory)
      1. show data\_directory: **/var/lib/postgresql/9.5/main**
3. psql select pg\_stop\_backup();

**Si insertamos datos entre el punto 2 y 3 es decir despues del TAR, esos registros estarán en el WAL BACKUP**

**2.5 Restore del backup continuo**

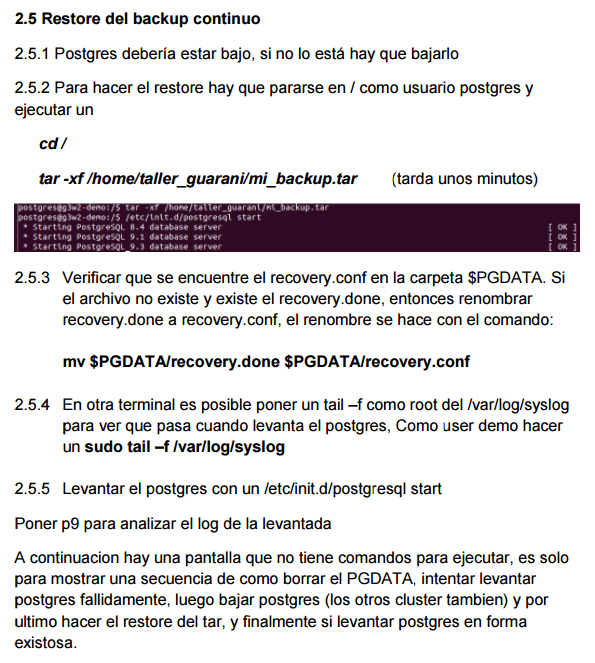
**2.5.1 Postgres debería estar bajo, si no lo está hay que bajarlo**

**2.5.2 Para hacer el restore hay que pararse en / como usuario postgres y ejecutar un cd / tar -xf /home/taller\_guarani/mi\_backup.tar (tarda unos minutos)**

**2.5.3 Verificar que se encuentre el recovery.conf en la carpeta $PGDATA. Si el archivo no existe y existe el recovery.done, entonces renombrar recovery.done a recovery.conf, el renombre se hace con el comando: mv $PGDATA/recovery.done $PGDATA/recovery.conf**

**2.5.4 En otra terminal es posible poner un tail –f como root del /var/log/syslog para ver que pasa cuando levanta el postgres, Como user demo hacer un sudo tail –f /var/log/syslog**

**2.5.5 Levantar el postgres con un /etc/init.d/postgresql start Poner p9 para analizar el log de la levantada A continuacion hay una pantalla que no tiene comandos para ejecutar, es solo para mostrar una secuencia de como borrar el PGDATA, intentar levantar postgres fallidamente, luego bajar postgres (los otros cluster tambien) y por ultimo hacer el restore del tar, y finalmente si levantar postgres en forma existosa.**

****

**psql -U username -W -h iphost basename**

**Parámetros:**

**-U es el usuario de la base**

**-W mostrará el prompt de solicitud de password**

**-h IP del servidor de la base de datos en caso nos conectemos remotamente sino bastaría con poner localhost**

**psql \l : Mostrar todas las bases de datos**

**psql \c o \connect DBNAME: Se conecta a la base de datos DBNAME**

**psql \d : Mostrar tablas**

<https://www.postgresql.org/docs/devel/static/app-pgdump.html>

## **pg\_dump (ejecutar fuera del postgres)**

pg\_dump — extract a PostgreSQL database into a script file or oth er archive file

pg\_dump [*connection-option*...] [*option*...] [*dbname*]

**-p *port***

**--port=*port***

**Specifies the TCP port or local Unix domain socket file extension on which the server is listening for connections. Defaults to the PGPORT environment variable, if set, or a compiled-in default.**

**ls -ld . (quien es el owner)**

**id (usuario actual)**

**chmod 777 . (carpeta actual)**

[**https://www.postgresql.org/docs/9.5/static/storage-file-layout.html**](https://www.postgresql.org/docs/9.5/static/storage-file-layout.html)

\d : Mostrar tablas

Mostrar procesos

*ps -ef*

*ps -ef | grep firefox*

*kill -9 PID (dónde PID, són los números del proceso que queremos detener)*

*kill -9 11070 11082 11109*

*/usr/local/9.1/bin*

*/usr/local/9.3/bin*

*Variable*

*$PATH (va a estar el pg\_dump y el psql)*

*tengo que ver con que version de pg\_dump esta corriendo y usar la misma version de la BD*

*locate pg\_dump (busca todas las carpetas donde se encuentre pg\_dump)*

*Saber que version estoy usando*

*psql select version();*