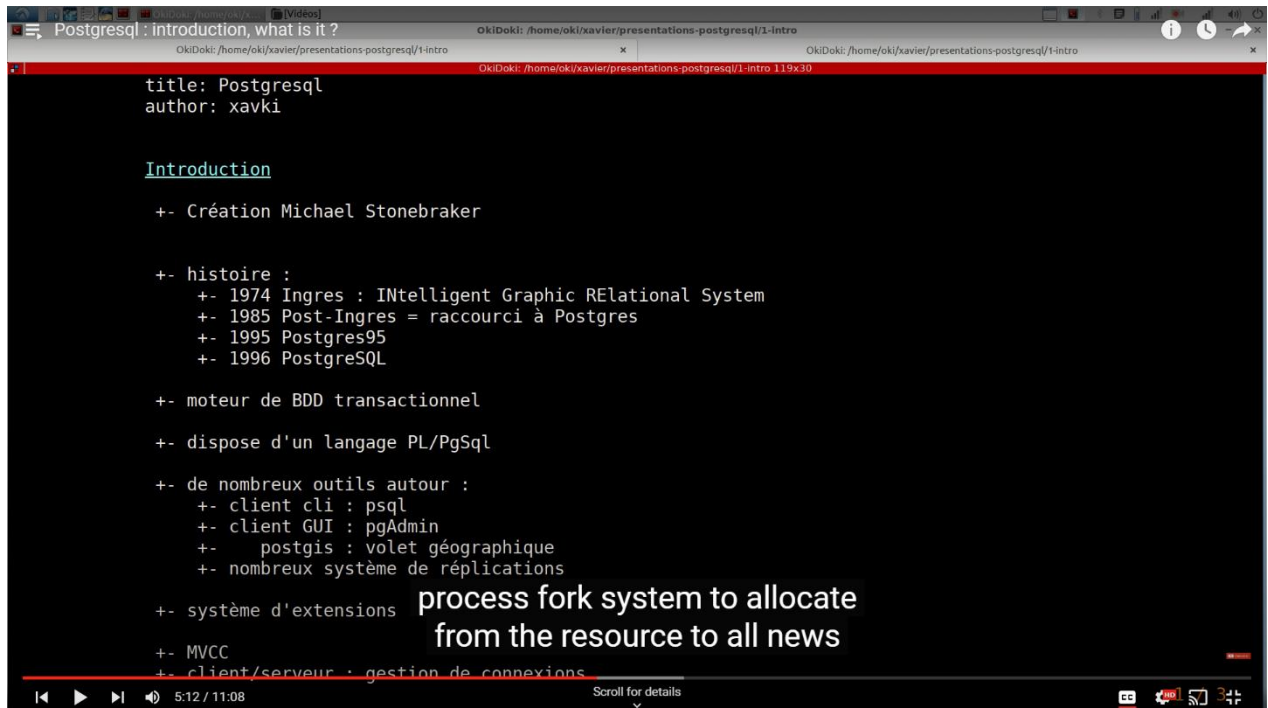


POSTGRESQL



Installation :

\$apt-get install postgresql-11



Principaux répertoires

```
+ configuration : /etc/postgresql/<version>/<clusterpg>
+ datas : /var/lib/postgresql/<version>/<clusterpg>
+ binaire principal : /usr/lib/postgresql/11/bin/postgres
+ binaires pg : /usr/bin/ (pg_*)
```

Processus forkés

```
postgres -D /var/lib/postgresql/11/main -c config_file=/etc/postgresql/11/main/postgresql.conf
\_ postgres: 11/main: checkpoint process
\_ postgres: 11/main: writer process
\_ postgres: 11/main: wal writer process
\_ postgres: 11/main: autovacuum launcher process
\_ postgres: 11/main: stats collector process
\_ postgres: 11/main: bgworker: logical replication launcher
```

```
root@postgres:/usr/lib/postgresql/11/bin# service postgresql start
root@postgres:/usr/lib/postgresql/11/bin# service postgresql status
● postgresql.service - PostgreSQL RDBMS
   Loaded: loaded (/lib/systemd/system/postgresql.service; enabled; vendor preset: enabled)
   Active: active (exited) since Sun 2019-08-04 16:13:30 GMT; 3min 31s ago
 Main PID: 2513 (code=exited, status=0/SUCCESS)
    Tasks: 0 (limit: 1150)
   Memory: 0B
    CGroup: /system.slice/postgresql.service

Aug 04 16:13:30 postgres systemd[1]: Starting PostgreSQL RDBMS...
Aug 04 16:13:30 postgres systemd[1]: Started PostgreSQL RDBMS.
root@postgres:/usr/lib/postgresql/11/bin# ps auxf | grep postgres
root      3949  0.0  0.0  6076  892 pts/0    S+   16:17   0:00          \_ grep postgres
postgres  2843  0.0  2.7 213444 27440 ?        S    16:13   0:00 /usr/lib/postgresql/11/bin/postgres -D /var/lib
/postgresql/11/main -c config_file=/etc/postgresql/11/main/postgresql.conf
postgres  2845  0.0  0.4 213444  4068 ?        Ss   16:13   0:00 \_ postgres: 11/main: checkpointer
postgres  2846  0.0  0.5 213584  5760 ?        Ss   16:13   0:00 \_ postgres: 11/main: background writer
postgres  2847  0.0  0.9 213444  9684 ?        Ss   16:13   0:00 \_ postgres: 11/main: walwriter
postgres  2848  0.0  0.6 213872  6704 ?        Ss   16:13   0:00 \_ postgres: 11/main: autovacuum launcher
postgres  2849  0.0  0.4  68504  4128 ?        Ss   16:13   0:00 \_ postgres: 11/main: stats collector
postgres  2850  0.0  0.6 213872  6828 ?        Ss   16:13   0:00 \_ postgres: 11/main: logical replication laun
```

Postgresql : binaires

```
+ pg_ctl :
+ gestion de l'instance/cluster
+ start/stop/kill
+ init: création autre espace de datas
+ promote : promotion du standby

+ psql :
+ client de connexion à un cluster
+ précision utilisateur et/ou db
+ passage de sql en cli ou script sql
```

```
+ - pg_createcluster :  
    +- création d'un cluster (une instance PG)  
    +- création des répertoires (/etc /var/lib/)  
  
+ - pg_dropcluster :  
    +- suppression d'un cluster  
    +- cluster arrêté  
  
+ - pg_lscluster :  
    +- lister les cluster  
  
+ - pg_ctlcluster :  
    +- équivalent du pg_ctl  
    +- contrôle du cluster (stop, start, restart, reload, status)
```

that's the last two

Spécifiques Sauvegardes

```
+ - pg_dump:  
    +- sauvegarde d'une instance  
    +- différents formats : plain text, binaire...  
    +- différents niveaux d'objets (cluster/db/table/schéma)  
  
+ - pg_dumpall:  
    +- sauvegarde intégrale en format binaire  
  
+ - pg_restore:  
    +- restauration à partir d'une sauvegarde (pg_dumpall)
```

Pour restaurer une sauvegarde avec pg_dump, on peut utiliser psql

Wrappers

- + - équivalent de commandes sql
- + - createdb :
 - + - création d'une base de données
- + - dropdb :
 - + - suppression d'une base de données
- + - createuser :
 - + - création d'un utilisateur
- + - dropuser :
 - + - suppression d'un user

Maintenance

- + - reindexdb :
 - + - réindexation des index avec des paramètres pour restreindre le périmètre
- + - vacuumdb :
 - + - tâche de maintenance (ménage)
- + - vacuumlo :
 - + - suppression de large objects

Spécifiques système avancées

- + - pg_controldata :
 - + - vérifie l'état du serveur et des infos critiques
- + - pg_resetwal :
 - + - en cas de crash avec pb de WAL (Write Ahead Logging)
 - + - attention : datas inconsistantes (dernier recours)
- + - pg_receive_wal :
 - + - récupération des WAL d'une autre DB
- + - pg_basebackup:
 - + - récupération de datas par une connexion à une autre DB (ex : init réplication)

Postgresql : les différents processus

- + - processus père : postgresql -D <data_dir> -c <conf_dir>
- + - principe de fork
- + - 2 type de fils :
 - + - les processus du moteur
 - + - les processus affectés aux connexions

Processus

- + - le checkpoint : permet poser des jalons mémoire/disque
 - + - écriture des dirty pages (mémoire) sur les data files (disque)
 - + - paramétrage de cette fréquence d'écriture
 - + - soit en durée et/ou en nombre de WAL switch
 - + - configuration dans postgresql.conf
 - + - param : checkpoint_segments et checkpoint_timeout
 - + - important pour le recovery notamment (coût en perf)
- + - background writer : écriture des dirty pages (mémoire) sur data files (disque)
- + - wal writer : écriture de wal buffer sur disque
- + - autovacuum launcher : en charge de l'autovacuum
- + - stats collector : calcule les statistiques
 - + - hyper important pour le fonctionnement d'un moteur
 - | (traitement efficace des requêtes)
 - + - écriture dans pg_stats_temp
- + - logical replication launcher :
 - + - en cas de réplication logique
 - + - lance un process pour les standby qui se connectent
- + - postgres postgres : allocation de la ressource
 - + - work memory

say see you soon on Gaddafi

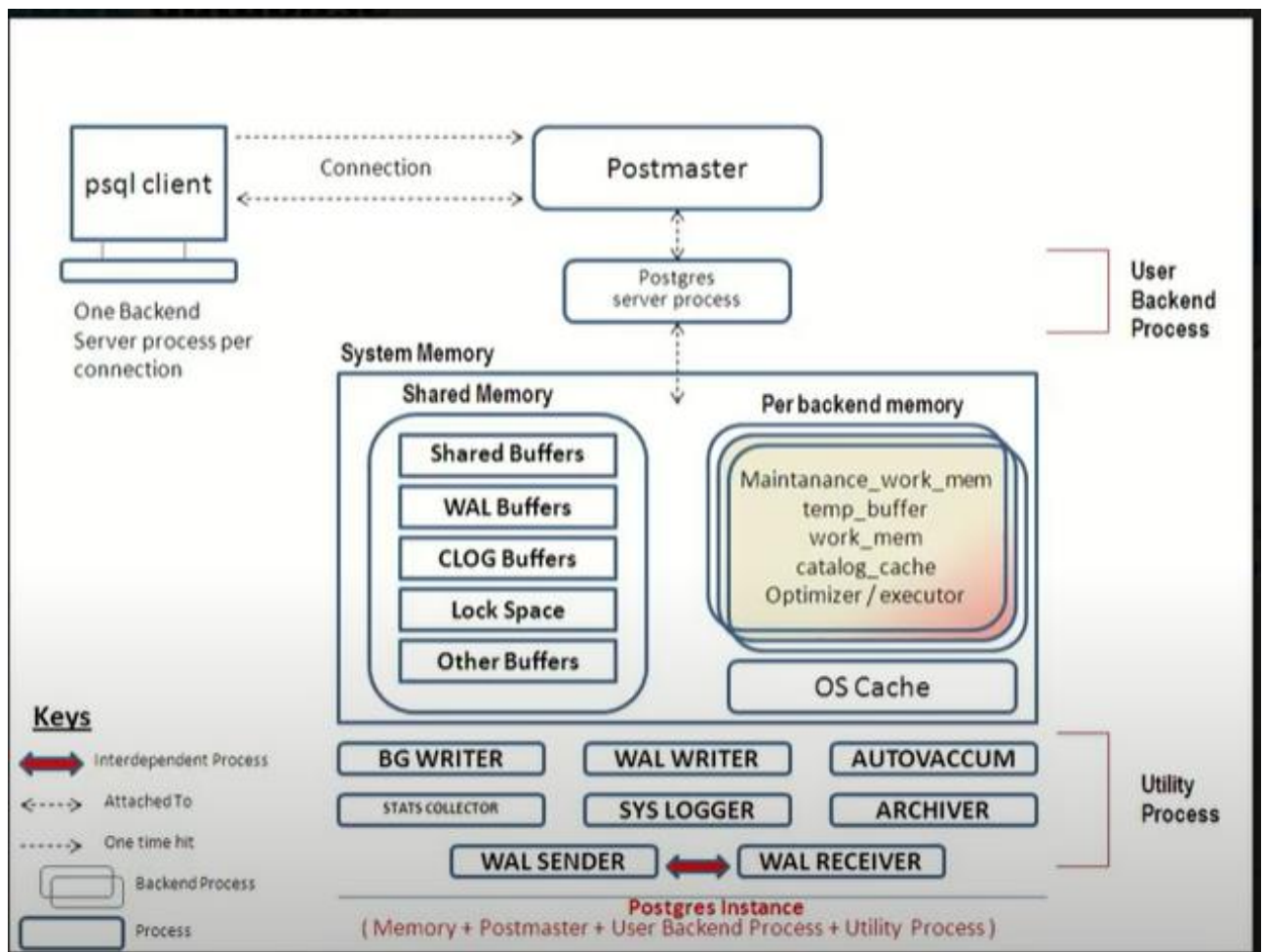
Le param work_mem dans le fichier de config définit la mémoire allouée pour chaque connexion utilisateur

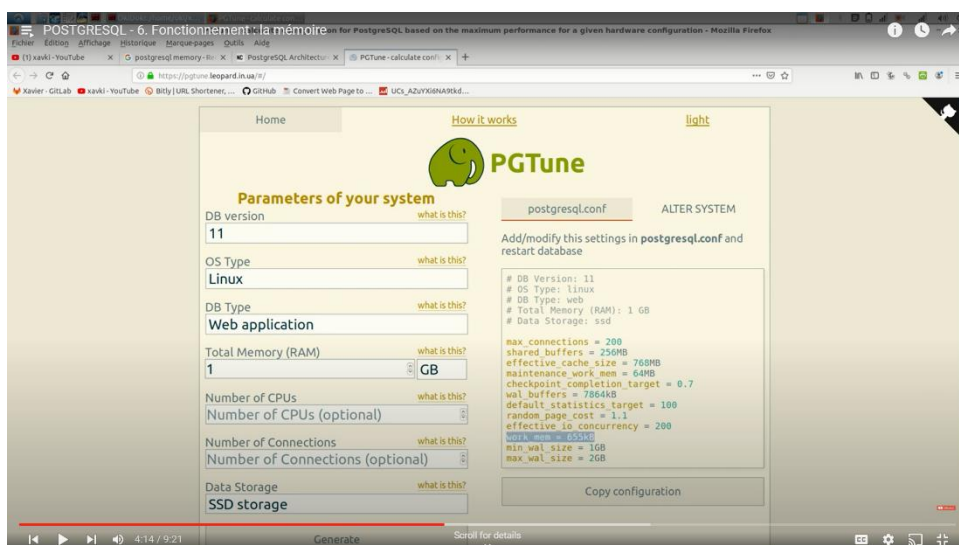
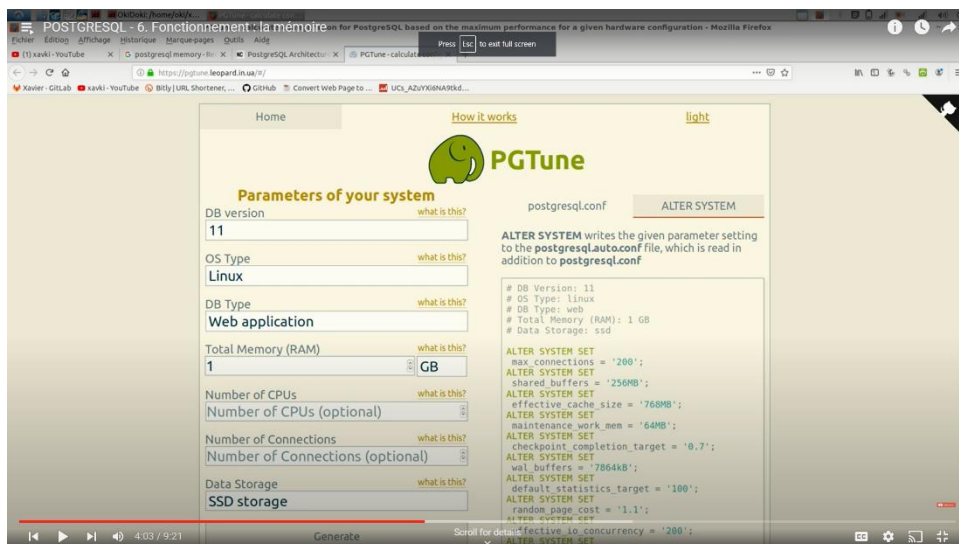
```
root@postgres:/home/vagrant# less /etc/postgresql/11/main/postgresql.conf
root@postgres:/home/vagrant#
```

```
#work_mem = 4MB          # min 64kB
#maintenance_work_mem = 64MB # min 1MB
#autovacuum_work_mem = -1  # min 1MB, or -1 to use maintenance_work_mem
#max_stack_depth = 2MB    # min 100kB
dynamic_shared_memory_type = posix
                          # the default is the first option
                          # supported by the operating system:
                          #   posix
                          #   sysv
                          #   windows
                          #   mmap
                          # use none to disable dynamic shared memory
                          # (change requires restart)
```


Postgresql : la mémoire

- + très important pour un moteur de BDD
- + accessibilité rapide à la donnée
- + la mémoire n'est pas pérenne :
 - + cas d'arrêt : prévu > système d'écriture
 - + indicent : imprévu > perte de la mémoire / recovery
- + postgres scinde en 4 la mémoire disponible :
 - + shared buffer : mémoire partagée
 - + wal buffers
 - + work mem
 - + maintenance work mem
- + on appelle shared memory = shared buffer + wal buffer + c-log-buffer
- + aide pour configurer : <https://pgtune.leopard.in.ua/>





Shared Buffer

+ - par défaut 128 Mo

+ - au mini 128ko + 16 ko par max connexions

+ - si plus 25% de celle du système

+ - cette partie de la mémoire est ensuite écrite sur datafile

+ - data plus rapide que sur datafile

+ - background writer : process chargé de la copie

WAL Buffer

- + en charge des archives des transactions
- + cette partie est ensuite écrite dans les WAL files puis archivée
- + wal writer : en charge de la copie
- + archiver : process en charge de l'archivage

Work mem

- + mémoire allouée pour chaque utilisateur
- + permet les tri, classement, filtre, jointures... des requêtes clientes
- + affectation pour chaque connexion
- + cf postgresql.conf > reload nécessaire

Maintenance work mem

- + mémoire dédiée aux opérations de maintenance
- + vacuum, create index, reindex...
- + valeur qui peut jouer sur la vitesse de restauration

Postgresql : principaux répertoires

- + 2 essentiels

```
/var/lib/postgresql/11/main/  
/etc/postgresql/11/main/
```

```
root@postgres:/var/lib/postgresql/11/main# ls  
base          pg_dynshmem   pg_notify     pg_snapshots  pg_subtrans  PG_VERSION    postgresql.auto.conf  
global        pg_logical    pg_replslot   pg_stat        pg_tblspc    pg_wal        postmaster.opts  
pg_commit_ts  pg_multixact  pg_serial     pg_stat_tmp    pg_twophase  pg_xact       postmaster.pid  
root@postgres:/var/lib/postgresql/11/main#
```

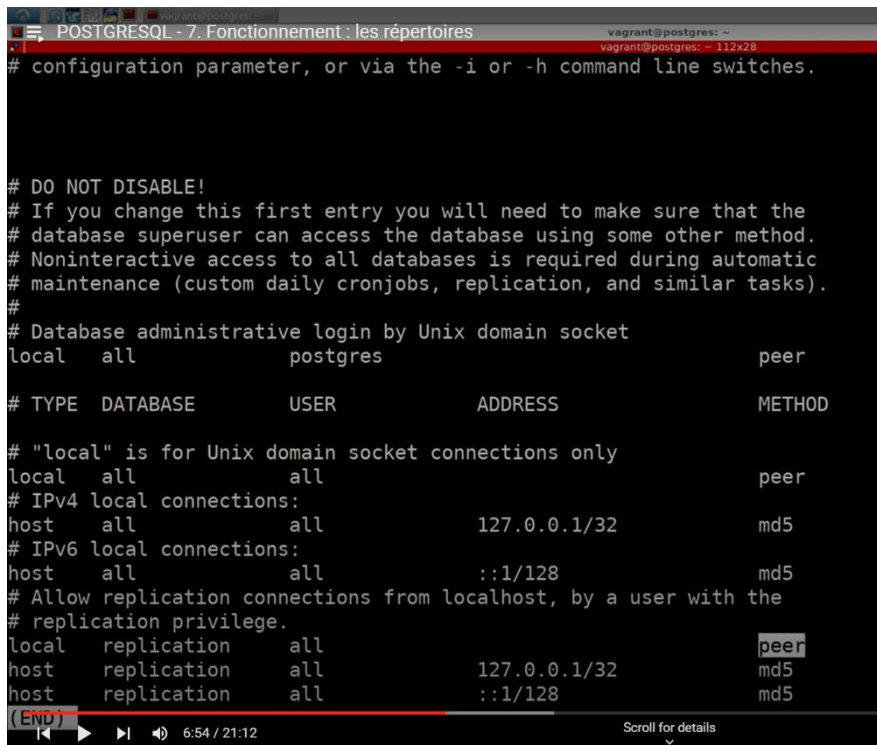
```
root@postgres:/etc/postgresql/11/main# ls  
conf.d  environment  pg_ctl.conf  pg_hba.conf  pg_ident.conf  postgresql.conf  start.conf  
root@postgres:/etc/postgresql/11/main# █
```

Fichier pg_hba : permet de définir les connexions possibles/autorisées à la bd

Le fichier pg_ctl.conf : permet de passer des options supplémentaires au lancement de la bd /service postgre

```
# Automatic pg_ctl configuration
# This configuration file contains cluster specific options to be passed to
# pg_ctl(1).

pg_ctl_options = ''
pg_ctl.conf (END)
```



```
# configuration parameter, or via the -i or -h command line switches.

# DO NOT DISABLE!
# If you change this first entry you will need to make sure that the
# database superuser can access the database using some other method.
# Noninteractive access to all databases is required during automatic
# maintenance (custom daily cronjobs, replication, and similar tasks).
#
# Database administrative login by Unix domain socket
local  all             postgres                    peer

# TYPE  DATABASE  USER  ADDRESS  METHOD

# "local" is for Unix domain socket connections only
local  all             all                          peer
# IPv4 local connections:
host   all             all             127.0.0.1/32    md5
# IPv6 local connections:
host   all             all             ::1/128         md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
local  replication     all                          peer
host   replication     all             127.0.0.1/32    md5
host   replication     all             ::1/128         md5
(END)
```

Le fichier start.conf : permet à postgre de démarrer automatiquement les clusters si la machine redemarre avec la valeur 'auto'. Sinon démarrer manuellement le cluster avec la commande indiquée

```
# Automatic startup configuration
# auto: automatically start the cluster
# manual: manual startup with pg_ctlcluster/postgresql@.service only
# disabled: refuse to start cluster
# See pg_createcluster(1) for details. When running from systemd,
# invoke 'systemctl daemon-reload' after editing this file.

auto
start.conf (END)
```

Le fichier pg_ident.conf : permet de faire un mapping entre user os et user postgre

```

# MAPNAME is the (otherwise freely chosen) map name that was used in
# pg_hba.conf. SYSTEM-USERNAME is the detected user name of the
# client. PG-USERNAME is the requested PostgreSQL user name. The
# existence of a record specifies that SYSTEM-USERNAME may connect as
# PG-USERNAME.
#
# If SYSTEM-USERNAME starts with a slash (/), it will be treated as a
# regular expression. Optionally this can contain a capture (a
# parenthesized subexpression). The substring matching the capture
# will be substituted for \1 (backslash-one) if present in
# PG-USERNAME.
#
# Multiple maps may be specified in this file and used by pg_hba.conf.
#
# No map names are defined in the default configuration. If all
# system user names and PostgreSQL user names are the same, you don't
# need anything in this file.
#
# This file is read on server startup and when the postmaster receives
# a SIGHUP signal. If you edit the file on a running system, you have
# to SIGHUP the postmaster for the changes to take effect. You can
# use "pg_ctl reload" to do that.
#
# Put your actual configuration here
# -----
# MAPNAME          SYSTEM-USERNAME          PG-USERNAME
# (END)

```

Répertoires

- + base
 - + localisation des bases
 - + template 0 et template 1 : appelé lors de la création de DB
 - + postgresql
 - + classement par oid (tables de correspondance entre oid et objets)
 - + SELECT relname, oid, relfilenode FROM pg_class;
 - + ex : SELECT relname, oid, relfilenode FROM pg_class where relname='xavier';
 - + ou SELECT pg_relation_filepath('xavier');
 - + Free Space Map (FSM) et Visibility Map (VM) : espace dispo , lien avec Vacuum
- + global :
 - + control files : pg_control appelés au démarrage
 - + connaître l'état du moteur
- + pg_wal :
 - + write ahead log (WAL)
 - + wal files : défaut 16 Mb
 - + perf = déplacement sur disques performants (ssd)
 - + recovery (checkpoint) : rejoue les fichiers wal

```

root@postgres:/var/lib/postgresql/11/main# ls
base          pg_dynshmem  pg_notify    pg_snapshots  pg_subtrans  PG_VERSION  postgresql.auto.conf
global        pg_logical   pg_replslot  pg_stat        pg_tblspc    pg_wal      postmaster.opts
pg_commit_ts  pg_multixact pg_serial     pg_stat_tmp    pg_twophase  pg_xact     postmaster.pid
root@postgres:/var/lib/postgresql/11/main# less v
v: No such file or directory
root@postgres:/var/lib/postgresql/11/main# less PG_VERSION
root@postgres:/var/lib/postgresql/11/main#

```

PG_VERSION : fichier qui indique la version majeure de postgres

Le fichier de control : fichier non lisible

```

root@postgres:/var/lib/postgresql/11/main/global# ls pg_control
pg_control
root@postgres:/var/lib/postgresql/11/main/global# cat pg_control
)0t+@G]L0\
root@postgres:/var/lib/postgresql/11/main/global#

```

Répertoire pg_wal : des logs de transaction

```

root@postgres:/var/lib/postgresql/11/main/pg_wal# ls
00000001000000000000000000000000 00000001000000000000000000000000 00000001000000000000000000000000 archive_status
00000001000000000000000000000000 00000001000000000000000000000000 00000001000000000000000000000000
00000001000000000000000000000000 00000001000000000000000000000000 00000001000000000000000000000000
00000001000000000000000000000000 00000001000000000000000000000000 00000001000000000000000000000000
root@postgres:/var/lib/postgresql/11/main/pg_wal#

```

Répertoires

```
+~ pg_act :  
  +- statut des transactions non s rializable commit es (type d'isolation des transactions)  
  
+~ pg_serial :  
  +- idem mais s rializable  
  
+~ pg_stat:  
  +- statistiques : important pour  laboration des plans des requ tes  
  
+~ pg_tblspc :  
  +- pointeur vers les tablespaces  
  
+~ pg_repslot:  
  +- informations sur la r plication par slot
```

CONNEXIONS

Postgresql : autorisation de connexion

```
+~ ip d' coute : listen_adresses  
  +- ou ip  
  +- plusieurs s par s par virgule ","  
  +- changement restart  
  
+~ pg_hba.conf :  
  +- Host Based Authentication  
  +- /etc/postgresql/...  
  +- lu de haut en bas  
  +- reload pour prise en compte  
  
+~ filtre multifacteurs :  
  +- type  
  +- database  
  +- user  
  +- ip/net  
  +- m thode d'authentification
```

Modifier le param tre listen_adresses dans pg_hba.conf - hba : host based authentication - pour autoriser les machines   se connecter   postgre :

* : Pour autoriser toutes les ips

'xx,xx,xx,..' : Pour autoriser seulement une liste de ips

```
#-----  
# CONNECTIONS AND AUTHENTICATION  
#-----  
# - Connection Settings -  
listen_addresses = '*'          # what IP address(es) to listen on;  
                                # comma-separated list of addresses;  
                                # defaults to 'localhost'; use '*' for all  
                                # (change requires restart)  
port = 5432                     # (change requires restart)
```

```

root@postgres:~# vim /etc/postgresql/11/main/postgresql.conf
root@postgres:~# service postgresql restart
root@postgres:~# service postgresql status
● postgresql.service - PostgreSQL RDBMS
   Loaded: loaded (/lib/systemd/system/postgresql.service; enabled; vendor preset: enabled)
   Active: active (exited) since Wed 2019-08-28 06:47:11 GMT; 1s ago
     Process: 11386 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
    Main PID: 11386 (code=exited, status=0/SUCCESS)

Aug 28 06:47:11 postgres systemd[1]: Starting PostgreSQL RDBMS...
Aug 28 06:47:11 postgres systemd[1]: Started PostgreSQL RDBMS.
root@postgres:~#

```

Avant redemarrage :

```

root@postgres:~# netstat -ntaup
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN      383/sshd
tcp        0      0 127.0.0.1:5432         0.0.0.0:*               LISTEN      390/postgres
tcp        0      0 10.0.2.15:22          10.0.2.2:53712         ESTABLISHED 781/sshd: vagrant [
tcp6       0      0 :::22                  :::*                     LISTEN      383/sshd
tcp6       0      0 :::5432                 :::*                     LISTEN      390/postgres
udp        0      0 0.0.0.0:68             0.0.0.0:*               ESTABLISHED 349/dhclient
udp6       0      0 :::52846                :::1:52846              ESTABLISHED 390/postgres
root@postgres:~#

```

Après redémarrage : 0.0.0.0 indique toutes les ips autorisées

```

root@postgres:~# netstat -ntaup
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN      383/sshd
tcp        0      0 0.0.0.0:5432           0.0.0.0:*               LISTEN      11368/postgres
tcp        0      0 10.0.2.15:22          10.0.2.2:53712         ESTABLISHED 781/sshd: vagrant [
tcp6       0      0 :::22                  :::*                     LISTEN      383/sshd
tcp6       0      0 :::5432                 :::*                     LISTEN      11368/postgres
udp        0      0 0.0.0.0:68             0.0.0.0:*               ESTABLISHED 349/dhclient
udp6       0      0 :::1:45266              :::1:45266              ESTABLISHED 11368/postgres
root@postgres:~#

```

Les facteurs

Les facteurs

```

+- type : local ou host (distant)

+- database : spécifique ou global (all)

+- user : spécifique ou global (all)

+- adresse :
  +- une ip, un masques de sous réseau (CIDR)
  +- ipv4 ou ipv6

+- méthode :
  +- trust : sans login
  +- peer : user de l'OS (exemple postgres)
  +- password : user/password simple
  +- md5 : avec chiffrement
  +- reject : refus (utile en cas de maintenance)
  +- ldap...

```

Rq : pour postgres > passwd postgres

Connexion par défaut dans pg_hba.conf : par peer (avec le user os 'postgresql')

```
# Database administrative login by Unix domain socket
local  all                postgres                peer

# TYPE  DATABASE      USER      ADDRESS      METHOD

root@postgres:/etc/postgresql/11/main# su - postgres
postgres@postgres:~$ id
uid=106(postgres) gid=113(postgres) groups=113(postgres),112(ssl-cert)
postgres@postgres:~$
```

Connexion avec psql :

```
postgres@postgres:~$ psql
psql (11.4 (Debian 11.4-1))
Type "help" for help.

postgres=# ^C
postgres=# \q
postgres@postgres:~$
```

Créer et se connecter avec psql avec un user particulier et en remote (host ou option h):

```
# Database administrative login by Unix domain socket
local  all                postgres                peer

# TYPE  DATABASE      USER      ADDRESS      METHOD

# "local" is for Unix domain socket connections only
local  all                all                peer
# IPv4 local connections:
host   all                all                127.0.0.1/32    md5
# IPv6 local connections:
host   all                all                ::1/128         md5

postgres=# create user xavier encrypted password 'password';

postgres@postgres:~$ psql -h 127.0.0.1 -U xavier postgres
Password for user xavier:
psql (11.4 (Debian 11.4-1))
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, bits: 256, compression: off)
Type "help" for help.

postgres=>
```

Dans cette commande :

-h 127.0.0.1 : on se connecte à partir de cette adresse

-U xavier : le user de connexion

Postgres : nom de la bd à laquelle on se connecte

NB : Fichier log de postgresql :

```
postgres@postgres:~$ less /var/log/postgresql/postgresql-11-main.log
```

Exemple 1: autoriser connexion d'un seul user :


```
# IPv4 local connections:
host    all             xavier          127.0.0.1/32          md5
# IPv6 local connections:

postgres=# create user paul encrypted password 'password';

postgres@postgres:~$ logout
root@postgres:/etc/postgresql/11/main# psql -h 127.0.0.1 -U paul postgres
psql: FATAL: no pg_hba.conf entry for host "127.0.0.1", user "paul", database "postgres", SSL on
FATAL: no pg_hba.conf entry for host "127.0.0.1", user "paul", database "postgres", SSL off
```

Exemple 2 :

On crée une bd “xavier” avec postgres

```
postgres@postgres:~$ psql
psql (11.4 (Debian 11.4-1))
Type "help" for help.

postgres=# create database xavier;
CREATE DATABASE
postgres=# \q
postgres@postgres:~$ vim /etc/postgresql/11/main/pg_hba.conf
```

On autorise ‘paul’ à se connecter uniquement sur la bd ‘xavier’:

```
# IPv4 local connections:
host    xavier          paul          127.0.0.1/32          md5
host    postgres        xavier        127.0.0.1/32          md5
```

On vérifie que ‘Paul’ ne peut pas se connecter à la bd ‘postgres’ mais peut se connecter sur la bd ‘xavier’ :

```
root@postgres:/etc/postgresql/11/main# service postgresql reload
root@postgres:/etc/postgresql/11/main# psql -h 127.0.0.1 -U paul postgres
psql: FATAL: no pg_hba.conf entry for host "127.0.0.1", user "paul", database "postgres", SSL on
FATAL: no pg_hba.conf entry for host "127.0.0.1", user "paul", database "postgres", SSL off
root@postgres:/etc/postgresql/11/main# psql -h 127.0.0.1 -U paul xavier
Password for user paul:
psql (11.4 (Debian 11.4-1))
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, bits: 256, compression: off)
Type "help" for help.

xavier=>
```

```
Connexions : memory

+- max_connexion : 400 bytes par connexion

Rq : si elles ne sont pas établies

+- connexions établies : paramètres work_mem par connexion établie
```

PSQL

PSQL : les débuts

+ - connexion par défaut en connexion local

```
psql
```

Remarque :

+ - cluster : main

+ - port : 5432

+ - serveur : localhost

+ - user : postgres

+ - avec des options et en passant par la socket (pas local)

```
psql -h 127.0.0.1 -p 5432 -U postgres -d postgres
```

Rq : --cluster dans le cas de plusieurs cluster ou nom différent

NB : Lorsqu'on ne se connecte pas en localhost mais qu'on passe par l'adresse de la machine cible, c'est le fichier pg_hba.conf (cf tnsnames.ora) qui est utilisé pour valider les connexions

Quelques options utiles

+ - -c : passer des commandes à partir de la CLI

+ - -L : logger la CLI

+ - -f : passer des fichiers de scripts sql

+ - -H : export en html

+ - -s : marque un temps d'arrêt à chaque requête (passage de fichier -f)

+ - -q : mode silencieux sans message

+ - -t : uniquement les lignes de résultats

Pour se connecter à la bd postgres (option -d) avec le user xavier (-U) au host 127.0.0.1 (-h) avec le port 5432 (-p) :

```
postgres@postgres:~$ psql -p 5432 -h 127.0.0.1 -U xavier -d postgres
Password for user xavier:
psql (11.4 (Debian 11.4-1))
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, bits: 256, compression: off)
Type "help" for help.

postgres=>
```

S'il y a plusieurs clusters sur le serveur, il faut utiliser l'option --cluster suivie du numéro de version et le nom de l'instance (ici 'main'):

```
postgres@postgres:~$ psql --cluster 11/main -p 5432 -h 127.0.0.1 -U xavier -d postgres
Password for user xavier:
psql (11.4 (Debian 11.4-1))
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, bits: 256, compression: off)
Type "help" for help.

postgres=>
```

Option -c : Permet de passer des commandes à psql

Ex: Pour lister les bds : -c "\l" . Avec "l+" on a plus de détails sur les tailles des bds

```
postgres@postgres:~$ psql -c "\l+"

```

List of databases					
Name	Owner	Encoding	Collate	Ctype	Access privileges
Size	Tablespace	Description			
-----+-----+-----+-----+-----+-----					
postgres	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	
7773 kB	pg_default	default administrative connection database			
template0	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	=c/postgres
7569 kB	pg_default	unmodifiable empty database			
					postgres=CTc/postgres
template1	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	=c/postgres
7569 kB	pg_default	default template for new databases			
					postgres=CTc/postgres
xavier	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	
7569 kB	pg_default				
(4 rows)					

(END)

Pour lister toutes les tables : "\dt"

```
postgres@postgres:~$ psql -c "\l+"
postgres@postgres:~$ psql -c "\dt"
      List of relations
 Schema |   Name   | Type  | Owner
-----+-----+-----+-----
 public | capitals | table | postgres
 public | cities   | table | postgres
 public | titi     | table | postgres
 public | toto     | table | postgres
 public | xavier   | table | postgres
(5 rows)

postgres@postgres:~$ psql -c "SELECT * FROM xavier;"
 id
----
  1
(1 row)

postgres@postgres:~$
```

Option -L : redirige le résultat dans un fichier qui suit l'option -L

```
postgres@postgres:~$ psql -c "SELECT * FROM xavier;" -L monlog.log
 id
----
  1
(1 row)

postgres@postgres:~$ ls
ll monlog.log
postgres@postgres:~$ less monlog.log
```

Option -f : permet de passer un fichier script de commandes sql en param :

```
postgres@postgres:~$ psql -f monsql.sql
count
-----
      1
(1 row)

 id
----
  1
(1 row)

postgres@postgres:~$
```

Option -html : Output en format html

```
postgres@postgres:~$ psql -f monsql.sql --html -L output.html
<table border="1">
  <tr>
    <th align="center">count</th>
  </tr>
  <tr valign="top">
    <td align="right">1</td>
  </tr>
</table>
<p>(1 row)<br />
</p>
<table border="1">
  <tr>
    <th align="center">id</th>
  </tr>
  <tr valign="top">
    <td align="right">1</td>
  </tr>
</table>
<p>(1 row)<br />
</p>
```

Option -t : retourne uniquement les résultats

```
postgres@postgres:~$ psql -f monsql.sql -t
1

1

postgres@postgres:~$
```

Commandes utiles une fois connecté à psql :

```
Quelques commandes

+- lister les databases
\l+

Rq: + > volumétrie

+- lister les users
\du

+- lister les tables
\dt+

+- lister les schémas
\dn

+- lister les champs d'une table
\dS <nom_table>

postgres@postgres:~$ psql
psql (11.4 (Debian 11.4-1))
Type "help" for help.

postgres=# \l+
```

\l+ : Pour lister les dbs et certains détails:

List of databases								
Name	Owner	Encoding	Collate	Ctype	Access privileges	Size	Tablespace	Description
postgres	postgres	UTF8	en_US.UTF-8	en_US.UTF-8		7773 kB	pg_default	default administrative connection database
template0	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	=c/postgres postgres=CTc/postgres	7569 kB	pg_default	unmodifiable empty database
template1	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	=c/postgres postgres=CTc/postgres	7569 kB	pg_default	default template for new databases
xavier	postgres	UTF8	en_US.UTF-8	en_US.UTF-8		7569 kB	pg_default	
(4 rows)								

\du : Pour lister les users :

```
postgres=# \du
```

List of roles		
Role name	Attributes	Member of
paul		{}
postgres	Superuser, Create role, Create DB, Replication, Bypass RLS	{}
xavier		{}

Superuser ~ DBA en oracle

Option \dt+ : Liste les tables avec les tailles et owners

```
postgres=# \dt+
```

List of relations					
Schema	Name	Type	Owner	Size	Description
public	capitals	table	postgres	8192 bytes	
public	cities	table	postgres	8192 bytes	
public	titi	table	postgres	0 bytes	
public	toto	table	postgres	0 bytes	
public	xavier	table	postgres	8192 bytes	
(5 rows)					

```
postgres=#
```