

# TP3 PostgreSQL

<http://tutoriels.meddeb.net/category/postgresql/>

## Partie 1 Installation de base de PostgreSQL

1) Installation de base de PostgreSQL, on liste les versions de PostgreSQL disponibles pour le système

```
q20305: root /home/ayman# apt-cache search postgresql | grep "^postgresql-[0-9].[0-9] "
```

postgresql-9.6 - object-relational SQL database, version 9.6 server

2) Installer une des versions listées (par exemple postgresql-9.6)

```
q20305: root /home/ayman# sudo apt-get install postgresql-9.6
```

3) Vérifier l'installation

On obtient un retour:

```
q20305: root /home/ayman# sudo service postgresql
Usage: /etc/init.d/postgresql {start|stop|restart|reload|force-reload|status} [version ..]
q20305: root /home/ayman#
```

Ce retour nous renseigne sur les commandes utilisables pour contrôler le fonctionnement du serveur, notamment la commande « status » qui permet de se renseigner sur l'état de fonctionnement: **online** (fonctionnement) ou **down** (arrêt).

4. Configurer l'accès au serveur

```
q20305: root /home/ayman# sudo -u postgres psql
psql (9.6.24)
Saisissez « help » pour l'aide.

postgres=#
```

Permet de se connecter en tant que « postgres » qui est le super administrateur du serveur par défaut. Une fois connecté créer un utilisateur avec des droits suffisants:

(remarque = Il faut utiliser maintenant CREATE ROLE à la place de USER)

```
postgres=# CREATE USER admin WITH PASSWORD 'admin' CREATEDB CREATEROLE;
CREATE ROLE
postgres=#
```

Crée l'utilisateur « admin » avec le mot de passe « admin » et qui a les droits de création de bases de données et de création d'autres utilisateurs:

Vérifie la réussite de la commande précédente, l'utilisateur « admin » doit apparaître dans la liste. Désormais, vous pouvez utiliser ce login et mot de passe pour vous connecter au serveur:

```
postgres=# SELECT * FROM pg_catalog.pg_user;
username | usesysid | usecreatedb | usesuper | userepl | usebypassrls | passwd | valuntil | useconfig
-----+-----+-----+-----+-----+-----+-----+-----+-----
postgres |      10 | t           | t        | t        | t            | ***** |          | 
admin    |    16387 | t           | f        | f        | f            | ***** |          | 
(2 lignes)
```

```
postgres=#
```

Se déconnecter du serveur.

```
postgres=# \q
q20305: root /home/ayman#
```

pgAdmin 4: Utilisation courante par la pratique

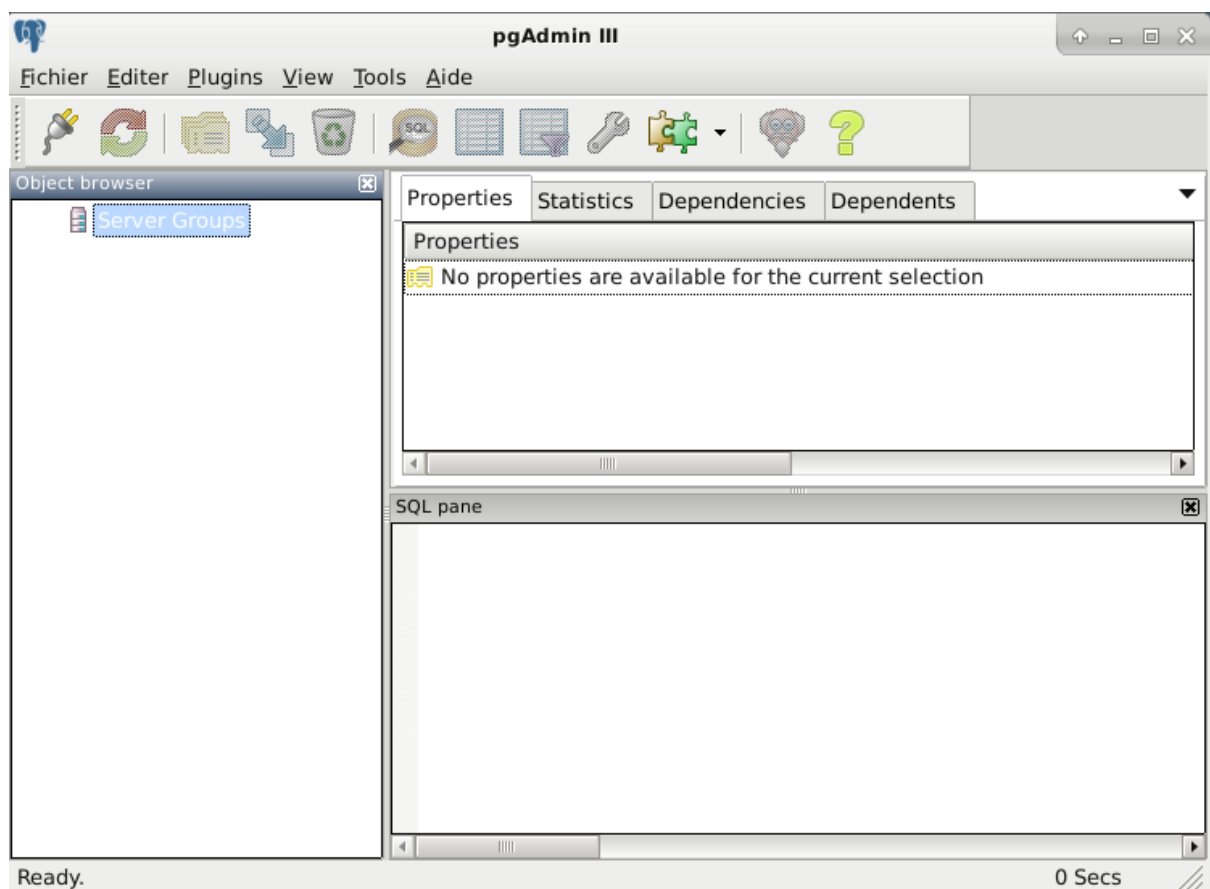
## Partie 2 Installation Postgre pgAdmin3

1) installer Postgres pgAdmin 3

```
q20305: root /home/ayman# sudo apt-get install pgadmin3
```

2) Paramétrage de base de Postgres pgAdmin

Le premier lancement de pgAdmin donne l'écran:



Utiliser le menu: Fichier / Ajouter un serveur...

**Nom:** *Serveur local*, ne sert que pour l'affichage, choix libre.

**Hôte:** *localhost* si le serveur est installé sur la même machine que pgAdmin. Sinon adresse (IP ou DNS) de la machine distante.

**Port:** 5432 par défaut

**Base de maintenance:** *postgres* créée et existe par défaut

**Nom utilisateur:** *admin* Celui qui a été créée à l'installation du serveur (cf. l'article à ce sujet).

**New Server Registration**

Properties | SSL | SSH Tunnel | Advanced

Nom: Serveur local

Host: localhost

Port: 5432

Service:

Maintenance DB: postgres

Username: admin

Password:

Store password: ☒

Colour:

Group: Servers

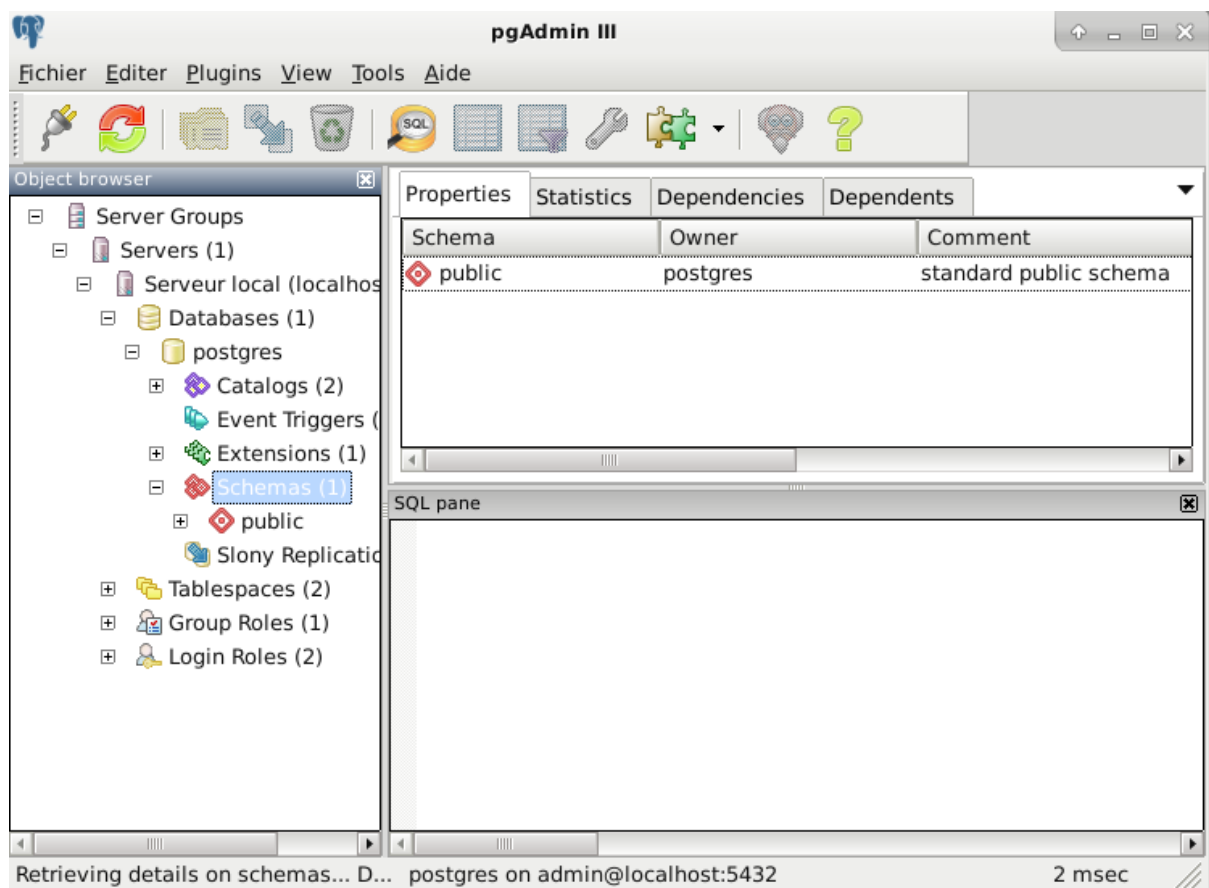
Aide Valider Annuler

A la validation l'accès au serveur est disponible

On remarquera la présence de l'unique base de donnée **postgres** et de l'utilisateur (rôle de connexion) **postgres**.

Ce sont les deux objets créés par défaut lors de l'installation du serveur.

On remarquera aussi la présence de l'utilisateur **admin** qui a été créée après l'installation du serveur (cf. l'article à ce sujet).



3) Création d'une base de données

- Se positionner sur le nœud « *Base de données (1)* »
- Clic k **droit** et utiliser le menu contextuel « *Ajouter une base de données...* »

**Nom:** Nom significatif, unique sur le même serveur.

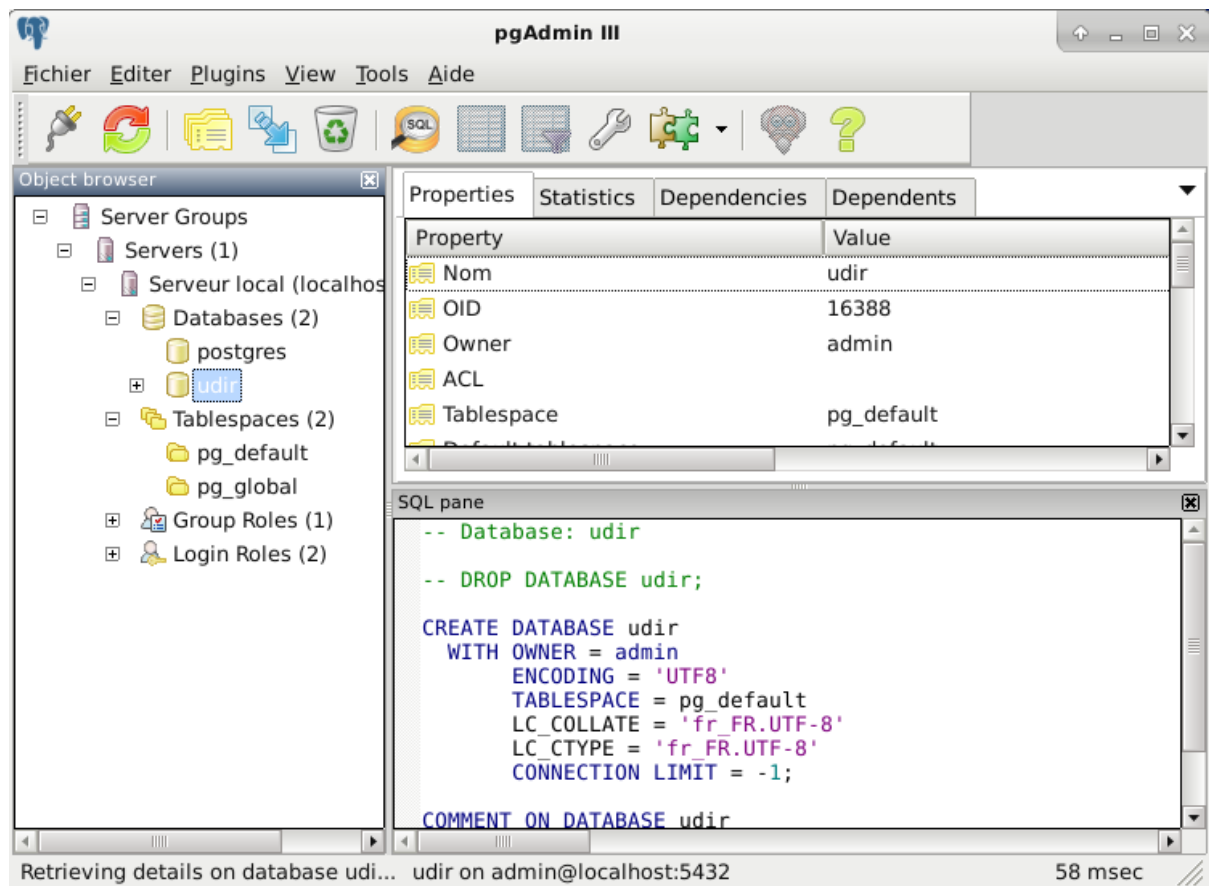
**Propriétaire:** Choisir l'utilisateur dans la liste.

**Commentaire:** Facultatif mais très utile pour la documentation

The image shows a 'New Database...' dialog box with the following elements:

- Tabbed Interface:** Properties (selected), Definition, Variables, Privileges, Security Labels, SQL.
- Nom:** Text field containing 'udir'.
- OID:** Empty text field.
- Owner:** Dropdown menu showing 'admin'.
- Comment:** Large text area containing 'Base de données du système uDir'.
- Buttons:** 'Aide' (Help), 'Valider' (Validate), and 'Annuler' (Cancel).

A la validation, la base de données est créée.



## Partie 3 Spécificité de l'installation de PostgreSQL 10

La version 10 de PostgreSQL a constitué une évolution majeure de ce serveur de bases de données objets, relationnelles et noSQL. Elle a marqué également une rupture avec la politique historique de la progression des numéros de version, spécialement en relation avec la compatibilité avec les données antérieures.

Avant la version 10 (jusqu'à la version 9.6 donc), PostgreSQL n'assurait pas la compatibilité des données quand le numéro de version mineur s'incrémente. A titre d'exemple, la version 9.6 ne peut pas et ne doit pas gérer des données créées avec la version 9.5. Ceci explique l'insertion automatique du numéro de version dans sa forme **majeur.mineur** dans tous les chemins qui concernent l'installation du serveur. Cette insertion permettrait de minimiser le risque de confusion entre les versions binaires vs données. Cela explique également la présence de ce format de numéro de version dans les paquets d'installation des distributions Debian et Ubuntu. Pour visualiser la version disponible pour votre système, exécuter :

```
q20305: root /home/ayman# sudo apt-get update
Atteint:1 http://security.debian.org stretch/updates InRelease
Ign:2 http://ftp.fr.debian.org/debian stretch InRelease
Atteint:3 http://ftp.fr.debian.org/debian stretch-updates InRelease
Atteint:4 http://ftp.fr.debian.org/debian stretch Release
Atteint:5 http://download.virtualbox.org/virtualbox/debian stretch InRelease
Lecture des listes de paquets... Fait
q20305: root /home/ayman#
```

En réponse à cette commande voici le résultat obtenu pour la dernière version stable de Debian (Le 10/03/2018 – Debian 9.3, Stretch):

```
q20305: root /home/ayman# apt-cache search postgresql | egrep "^postgresql-[0-9]+([.][0-9])? "
postgresql-9.6 - object-relational SQL database, version 9.6 server
q20305: root /home/ayman#
```

Cette contrainte nécessitait, si on voudrait passer à une nouvelle version, de migrer les données en passant par un format texte du *dump* (*dump* sous forme d'ordres SQL). Or, cette opération est très lente et devient vite contraignante à partir de quelques dizaines de Go de données. Dans ce cas également, un arrêt du service de l'application qui utilise le serveur, et qui peut être excessivement long, doit être envisagé.

A partir de la version 10, PostgreSQL assurera la compatibilité des données entre les versions mineurs. En d'autres termes, aucune migration de données n'est nécessaire entre les versions 10.1 et 10.2 par exemple. Ceci permettra une plus grande flexibilité pour la mise à jour du serveur. Si on sait que le passage entre la première version 9.0 stable et la 10 stable s'est fait en 7 ans, contre 10 mois entre la 9.5 et la 9.6, il est évident que le gain en terme de stabilité du format des données est substantiel. Pour cette raison, le numéro de

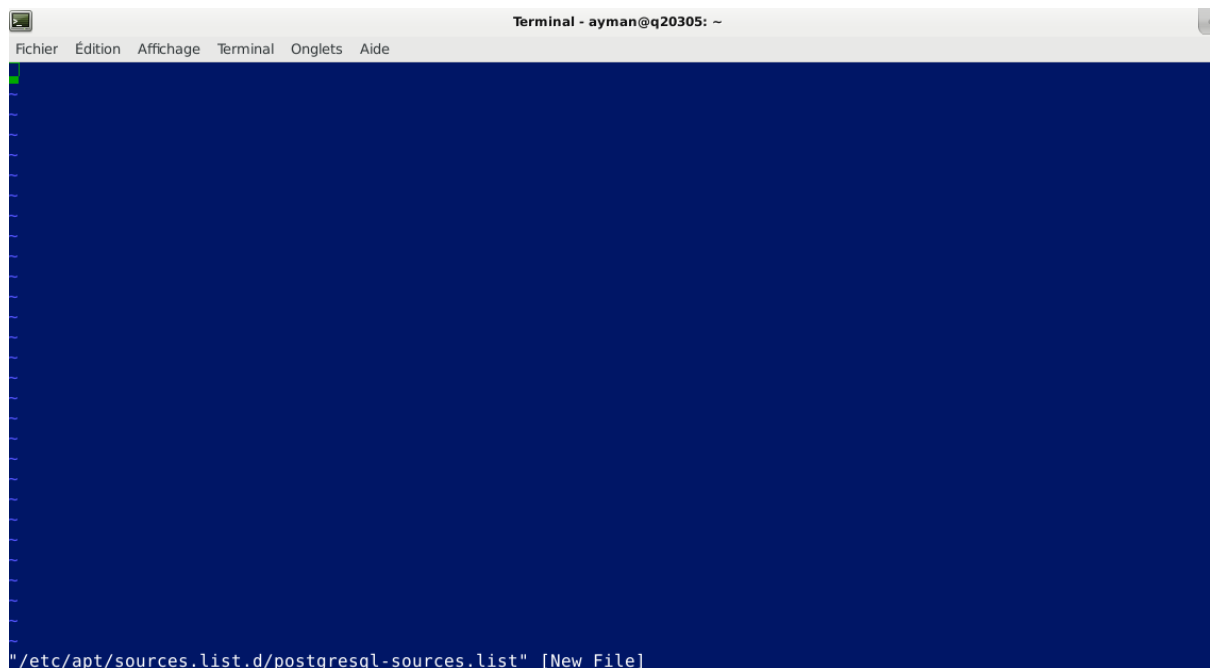


version inclut par défaut dans les chemins d'installation du serveur ne comporte plus que le premier chiffre. C'est également le cas pour le numéro qui accompagne les paquets d'installation des distributions Debian et Ubuntu.

## Versions disponibles pour le système installé

Même avec la dernière version stable du système (Debian 9.3, Stretch ou Ubuntu 16.04 LTS, Xenial au 10/03/2018), on ne dispose pas encore de la version 10 de PostgreSQL. Ceci a été vérifié avec la commande de visualisation des version disponibles exécutée précédemment. En tout cas vous n'aurez toujours pas la totalité des versions stables publiées de ce serveur.

En réalité ceci est vrai quand on ne référence que le dépôt officiel du système. Pour pouvoir accéder à l'ensemble des versions stables publiées, il faut ajouter le dépôt PostgreSQL à la liste des sources de paquets de la machine. Pour cela il faut créer un fichier de configuration de ce dépôt suivant la version installée de l'OS:



Le contenu de ce fichier doit être pour un système Debian 9 – Stretch: