

# Administration de PostgreSQL

Fabien Coelho  
fabien@coelho.net  
MINES ParisTech

## Résumé

Cette séance pratique aborde l'administration de PostgreSQL.

Vous rédigerez au fur et à mesure un compte-rendu de vos travaux : questions abordées, commandes exécutées, résultats obtenus, commentaires éventuels. Vous enverrez ce compte-rendu au plus tard le soir du TP.

N'hésitez pas à consulter la documentation en ligne sur le site de PostgreSQL : [www.postgresql.org](http://www.postgresql.org)

## 1 Initialisation et lancement

Vous allez utiliser les binaires déjà installés dans `/usr/linux/postgres/pg803` pour créer<sup>1</sup> un nouveau *cluster* dans un répertoire de votre choix sur votre compte<sup>2</sup>, avec un compte administrateur de la base nommé `root` dont le mot de passe est *secret*.

La base tournera sous votre identité. Configurez le contrôle des accès de manière à pouvoir accéder à votre *cluster* en tant qu'administrateur par une connexion réseau locale (numéro ip `127.0.0.1`).

Démarrer votre serveur avec `pg_ctl` accompagné des bons paramètres. En particulier, vous utiliserez comme fichier de log un fichier `trace` situé sur votre compte. Vous consulterez régulièrement ce fichier pour voir ce qui s'y passe, par exemple en ouvrant dans un terminal un `tail -f trace`.

## 2 Comptes utilisateurs et contrôle des accès

Créez quatre utilisateurs normaux : `calvin`, `hobbes`, `suzy` et *votre-login-unix* avec des mots de passe de votre choix. Le mot de passe de `hobbes` devra être stocké en clair.

---

1. Il peut être nécessaire d'éliminer quelques variables d'environnement (`unsetenv LANG, LC_...`) lors de la création.

2. Créer un *cluster* basé sur un système de fichier distant accessible via NFS (*Network File System*) est une très mauvaise idée en terme de performance.

Configurez le contrôle des accès de manière à ce que ces nouveaux comptes soient accessibles par mots de passe à partir de votre machine et des machines de vos voisins.

Rechargez votre configuration dans votre serveur et vérifiez que les accès distants fonctionnent. Consultez le contenu de la table système `pg_shadow` en tant qu'administrateur.

### 3 Création de bases

Créez deux catalogues `beaudelaire` et `rimbaud` appartenant respectivement aux utilisateurs `calvin` et `hobbes`.

Créez un groupe `family` regroupant `calvin` et `hobbes`. Permettez également aux membres du groupe `family` d'accéder à ces deux bases.

Configurez les accès pour que le compte base de donnée de *votre-login-unix* ait accès via `ident` à partir de la machine locale à tous les catalogues.

Rechargez votre configuration et vérifiez les droits d'accès.

### 4 Droits spéciaux utilisateurs

Modifiez votre installation afin que :

- *votre-login-unix* soit un administrateur
- `calvin` puisse créer de nouvelles bases

Créez un nouveau compte administrateur `fabien` de mot de passe *FaBieN* qui puisse accéder à **toutes** les bases à partir de la station `amphi-coelho.paris.ensmp.fr`.

Rechargez votre configuration et vérifiez les accès.

### 5 Extensions diverses

Vous souhaitez pouvoir développer des fonctions PL/pgSQL dans la base `beaudelaire`.

Créez et testez une fonction de test avec les commandes suivantes :

```
CREATE FUNCTION one() RETURNS INTEGER AS $$  
  BEGIN RETURN 1; END;  
$$ LANGUAGE plpgsql;
```

```
SELECT one(); -- doit retourner 1
```

Vous souhaitez ajouter à la base `rimbaud` les extensions `dbsize` et `isbn_issn` dont les documentations sont dans le sous répertoire `doc/contrib` de l'installation des binaires et les scripts SQL de chargement sont dans le sous-répertoire `share/contrib`.

Quels nouveaux types ont été ajoutés par l'extension `isbn_issn` ? Créez une table `livre` utilisant un des nouveaux type.

Quelles fonctions ont été ajoutées par l'extension `dbsize`? Quelle est la taille de la base `beaudelaire`?

## 6 Sauvegarde et restauration

Sauver le catalogue `beaudelaire` dans un fichier compressé `charles.dbz` avec la commande `pg_dump`.

Le restaurer dans un nouveau catalogue nommé `charles` avec la commande `pg_restore`. Vérifier que la relation `livre` et la fonction `one` ont bien été installées dans la nouvelle base.

## 7 Tablespace

Vous souhaitez utiliser un répertoire supplémentaire `/tmp/espace` pour stocker des bases supplémentaires sur un autre disque.

Créez un *tablespace* vers ce répertoire, et utilisez-le pour y mettre un nouveau catalogue `apolinaire`.

## 8 Extensions automatiques

Vous souhaitez que toute nouvelle base créée dispose automatiquement du langage PL/pgSQL et de l'extension `dbsize`. Que proposez-vous?

## 9 Perte du mot de passe administrateur

Vous avez perdu le mot de passe administrateur qui était secret. Comment le remplacer sans le connaître?

## 10 Pgpool

Vous souhaitez mettre en place un proxy intermédiaire `pgpool` pour votre machine.

Celui-ci est installé sous `/usr/linux/postgres/misc` : exécutable dans `bin`, documentation dans `doc` et exemple de fichier de configuration dans `etc`.

Cela remet-il en cause le contrôle des accès que vous avez mis en place?

## 11 Maintenance périodique

Vous souhaitez utiliser le démon `cron` accessible avec la commande `crontab` pour maintenir quotidiennement vos bases en opérant les nettoyages utiles.