

Révision	Auteur	Commentaires	Date
1	Jérôme Despatis	Création de la documentation	13/04/2012
2	Jérôme Despatis	2 fixes ajoutés non impactant	16/04/2012
3	Jérôme Despatis	Coquilles sur la numérotation + 2 ajouts mineurs sur la documentation	17/04/2012

Paris, le 13 avril 2012

Installation et Maintenance Réplication PostgreSQL



I - Table des matières

[I - Table des matières](#)

[II - Contexte](#)

[III - Architecture globale](#)

[A - Schéma global](#)

[a - Dans le cas où SQL 1 est le Master](#)

[b - Dans le cas où SQL 2 est le Master](#)

[IV - Installation des serveurs PostgreSQL](#)

[A - Paramétrage réseau](#)

[B - Installation des paquets nécessaires](#)

[C - Utilisateur système postgres](#)

[V - Configuration de chaque serveur PostgreSQL](#)

[A - Configuration des accès SSH](#)

[B - Configuration de PostgreSQL](#)

[VI - Initialisation de la réplication](#)

[A - Pré-Initialisation du Master et du Slave](#)

[B - Initialisation du Master](#)

[C - Initialisation du Slave](#)

[VII - Maintenance de la réplication - Switch Master <-> Slave](#)

[A - Actions à effectuer sur le Master sqlX.domain.com, afin de le désactiver en tant que Master](#)

[a - \(1er cas\) le Master sqlX.domain.com est fonctionnel et accessible](#)

[b - \(2ème cas\) le Master sqlX.domain.com n'est pas fonctionnel ou n'est pas accessible](#)

[B - Actions à effectuer sur le Slave sqlY.domain.com, afin de le rendre autonome](#)

[C - Remise en état de fonctionnement du Master sqlX.domain.com, s'il était précédemment down](#)

[D - Réactivation de la réplication](#)

[VIII - Retour à Zero](#)

[IX - Personnalisation du document](#)



Installation et maintenance d'une réplique de 2 serveurs PostgreSQL en mode Master / Slave

II - Contexte

La présente documentation a pour but de configurer une réplique PostgreSQL entre un Master et un Slave, et d'assurer sa maintenance en faisant passer indifféremment un des serveurs en Master et l'autre en Slave.

Les 2 serveurs, devant faire office de Master et de Slave, sont tous deux basés sur **Ubuntu 10.04 LTS**

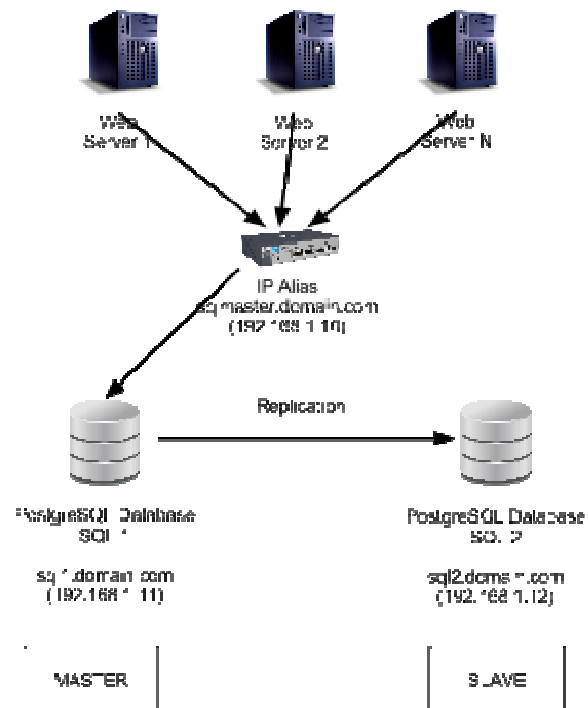
III - Architecture globale

A - Schéma global

Le schéma architectural est le suivant :

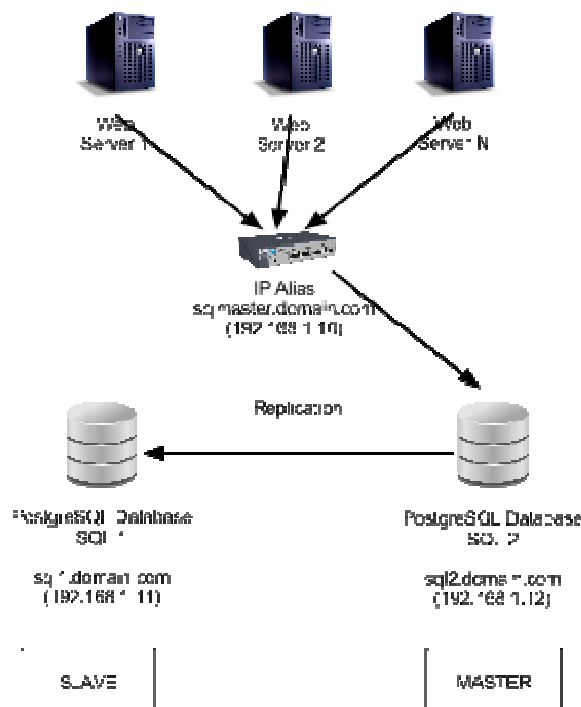
a - Dans le cas où SQL 1 est le Master





b - Dans le cas où SQL 2 est le Master





A savoir un (ou plusieurs) frontaux Web attaquent tous la même IP, db-server-postgres.apps.valeo.com, qui ne change jamais, afin de ne jamais avoir à modifier la configuration SQL des frontaux Web.

Cette IP fixe, db-server-postgres.apps.valeo.com, est affectée à un des 2 serveurs de bases de données PostgreSQL, qui devient, dès lors, le Master. L'autre serveur PostgreSQL devenant alors, de facto, le Slave.

IV - Installation des serveurs PostgreSQL

Procédure à effectuer sur les 2 serveurs sql1.so.apps.valeo.com et sql1.lc.apps.valeo.com

A - Paramétrage réseau

Il est nécessaire de définir un plan d'adresse dédié que l'on reportera dans le fichier /etc/hosts.



Les 'domain.com', ainsi que les IP sont à paramétrer.

1. Lignes à ajouter dans le fichier /etc/hosts du serveur sql1.so.apps.valeo.com :


```
172.26.3.17 db-server-postgres.apps.valeo.com # IP Alias of the Master database
172.26.3.9 sql1.so.apps.valeo.com # Static IP of the SQL 1, which
is the local IP
172.26.3.10 sql1.lc.apps.valeo.com db-slave-postgres.apps.valeo.com # Static IP of the
SQL 2, which is a friend of SQL 1
```
2. Lignes à ajouter dans le fichier /etc/hosts du serveur sql1.lc.apps.valeo.com :


```
172.26.3.17 db-server-postgres.apps.valeo.com # IP Alias of the Master
database
172.26.3.9 sql1.so.apps.valeo.com db-slave-postgres.apps.valeo.com # Static IP of the
SQL 1, which is a friend of SQL2
172.26.3.10 sql1.lc.apps.valeo.com # Static IP of the SQL 2, which
is the local IP
```
3. Vérification: avec cette configuration les 2 serveurs doivent répondre au ping sur :
 - a. sql1.so.apps.valeo.com et sql1.lc.apps.valeo.com
 - b. Sur sql1.so.apps.valeo.com, un ping sur db-slave-postgres.apps.valeo.com doit aller sur l'IP de sql1.so.apps.valeo.com
Sur sql1.lc.apps.valeo.com, un ping sur db-slave-postgres.apps.valeo.com doit aller sur l'IP de sql1.so.apps.valeo.com

B - Installation des paquets nécessaires

1. Pour chacun des serveurs sql1.so.apps.valeo.com et sql1.lc.apps.valeo.com, installer les paquets suivants :


```
root@sqlX:~# apt-get install postgresql postgresql-client postgresql-contrib rsync ssh
pitertools language-pack-fr
```

(Attention au copier/coller sur 2 lignes)
2. Prendre soin d'installer aussi, toujours sur sql1.so.apps.valeo.com et sur sql1.lc.apps.valeo.com, **xqueries** (L'archive est disponible sur le serveur admin1.so.apps.valeo.com via l'url http://admin1.so.apps.valeo.com/prod/roles/xqueries_for_pg84_64x.tar.gz). Ceci consistant à:
 - ☐ copier le fichier xqueries.so dans /usr/lib/postgresql/8.4/lib
 - ☐ copier les fichiers xqueries.sql et uninstall_xqueries.sql dans /usr/share/postgresql/8.4/contrib

C - Changement du répertoire de donnée postgresql



1. Pour chacun des serveurs sql1.so.apps.valeo.com et sql1.lc.apps.valeo.com, faire :

```
root@sqlX:~# mkdir /opt/data/postgresql
root@sqlX:~# chown -R postgres:postgres /opt/data/postgresql
root@sqlX:~# /etc/init.d/postgresql-8.4 stop
root@sqlX:~# usermod -d /opt/data/postgresql/ postgres
root@sqlX:~# su - postgres
root@sqlX:~# /usr/lib/postgresql/8.4/bin/initdb -D /opt/data/postgresql/8.4/main --locale
fr_FR.UTF-8
root@sqlX:~# exit
root@sqlX:~# cd /opt/data/postgresql/8.4/main
root@sqlX:~# ln -s /etc/ssl/certs/ssl-cert-snakeoil.pem server.crt
root@sqlX:~# ln -s /etc/ssl/private/ssl-cert-snakeoil.key server.key
root@sqlX:~# rm postgresql.conf pg_hba.conf pg_ident.conf
root@sqlX:~# mv /var/lib/postgresql /var/lib/postgresql.old
root@sqlX:~# /etc/init.d/postgres-8.4 start
```

D - Utilisateur système postgres

1. Pour chacun des serveurs sql1.so.apps.valeo.com et sql1.lc.apps.valeo.com, affecter un mot de passe à l'utilisateur système postgres (même mot de passe sur les 2 machines) :

```
root@sqlX:~# passwd postgres
```

E - Configuration de l'alias IP

1. Dans le fichier /etc/network/interface, rajouter :

```
#auto eth0:1
iface eth0:1 inet static
    address 172.26.3.17
    netmask 255.255.255.192
```

V - Configuration de chaque serveur PostgreSQL

ATTENTION: L'actuelle procédure (étape V) est à effectuer sur les 2 serveurs sql1.so.apps.valeo.com et sql1.lc.apps.valeo.com APRES avoir effectué intégralement l'étape précédente (étape IV) (concernant les 2 serveurs)

Toutes les instructions présentes dans toute cette partie V sont à appliquer, et ceci de façon rigoureusement identique, sur les 2 serveurs SQL :

- ☐ sql1.so.apps.valeo.com
- ☐ sql1.lc.apps.valeo.com

En effet, chaque serveur peut être indifféremment un Master ou un Slave.



Il est donc nécessaire d'apporter toute la configuration pour que le serveur en question puisse prendre chacun de ces 2 états.

Aussi, pour toute la suite de cette partie V, nous utiliserons le terme sql1.X.apps.valeo.com (pour indiquer séquentiellement sql1.so.apps.valeo.com, puis sql1.lc.apps.valeo.com)

A - Configuration des accès SSH

L'utilisateur système 'postgres' des serveurs sql1.X.apps.valeo.com doit pouvoir se connecter, de façon directe, sans avoir à taper de mot de passe sur le compte 'postgres' de db-slave-postgres.apps.valeo.com

Cette opération passe par les étapes suivantes :

1. Se connecter avec l'utilisateur 'postgres' :

```
admin@sqlX:~$ sudo su - postgres
```
2. Créer une clef privée ssh en tant qu'utilisateur système postgres :
(Tout accepter par défaut et ne pas fournir de passphrase)

```
postgres@sqlX:~$ ssh-keygen
```
3. Copier la clé public sur db-slave-postgres.apps.valeo.com :
(Fournir le mot de passe de l'utilisateur système postgres de db-slave-postgres.apps.valeo.com)

```
postgres@sqlX:~$ ssh-copy-id db-slave-postgres.apps.valeo.com
```
4. Vérifier que la connection ssh s'effectue désormais sans demander de mot de passe :

```
postgres@sqlX:~$ ssh db-slave-postgres.apps.valeo.com
```

=> On doit se retrouver sur db-slave-postgres.apps.valeo.com sans avoir à effectuer quoi que ce soit (ni taper un mot de passe, ni rien taper sur son clavier d'ailleurs...)

B - Configuration de PostgreSQL

Note : la configuration par défaut de PostgreSQL est utilisée ici.

1. Le serveur sqlX doit pouvoir être interrogé par tous les frontaux Web.
 Il est donc nécessaire d'ajouter la plage d'IP des frontaux Web à la configuration de PostgreSQL
=> Cette plage d'IP est à paramétrer
 Ajouter la ligne suivante, tout à la fin du fichier



- ```
/etc/postgresql/8.4/main/pg_hba.conf :
```
- ```
host    all    all    172.26.3.0/26    md5
```
2. Il faut également redéfinir le mot de passe de l'utilisateur bdd postgres :
(Mot de passe à paramétrer. Il faut cependant prendre le **MÊME** mot de passe sur sql1 et sur sql2 !)

```
postgres@sqlX:~$ psql postgres
```

```
--
```

```
ALTER USER postgres WITH ENCRYPTED PASSWORD 'UaQHhGp8';
```
 3. Modifier sur sql1.X.apps.valeo.com le fichier
/etc/postgresql/8.4/main/postgresql.conf comme suit :

```
listen_addresses = '*'
```

et

```
archive_mode = off
```

```
archive_command = '/usr/bin/cmd_archiver -C /etc/postgresql/8.4/archiver.ini -F %p'
```

```
archive_timeout = 600 #10mn
```
 4. Créer le fichier /etc/postgresql/8.4/archiver.ini, à partir de celui fourni en annexe
 5. Paramétrer dans ce fichier le paramètre db-slave-postgres.apps.valeo.com
 6. Puis modifier les droits de ce fichier comme suit :

```
root@sqlX:~# chown postgres:postgres /etc/postgresql/8.4/archiver.ini
```
 7. Créer le fichier /etc/postgresql/8.4/standby.ini, à partir de celui fourni en annexe.
 8. Paramétrer dans ce fichier le paramètre db-server-postgres.apps.valeo.com
 9. Puis modifier les droits de ce fichier comme suit :

```
root@sqlX:~# chown postgres:postgres /etc/postgresql/8.4/standby.ini
```
 10. Créer le script failover.sh, comme suit :

```
root@sqlX:~# echo "exit 0" > /opt/data/postgresql/8.4/failover.sh
```

```
root@sqlX:~# chown postgres:postgres /opt/data/postgresql/8.4/failover.sh
```

```
root@sqlX:~# chmod 755 /opt/data/postgresql/8.4/failover.sh
```
 11. En tant qu'utilisateur système postgres, ajouter les fonction pitrtools :

```
postgres@sqlX:~$ psql -f /usr/share/pitrtools/cmd_standby.sql postgres
```

VI - Initialisation de la réplication

Cette étape consiste à décider (de façon arbitraire ou non), le serveur parmi SQL 1 et SQL 2 qui fera office de Master (l'autre étant alors celui qui fera office de Slave).

Cette décision est laissée libre à l'opérateur.

Afin de garder une procédure générique, appelons donc sqlV.domain.dom le serveur choisi comme Master et sqlW.domain.com celui choisi comme Slave.

A - Pré-Initialisation du Master et du Slave



Remarque : Il est important d'effectuer les opérations de cette rubrique **AVANT** d'attaquer les parties suivantes B et C.

1. Créer le répertoire de stockage des journaux d'échange, en tant qu'utilisateur postgres, sur le Master :

```
postgres@sqlV:~$ cmd_archiver -C /etc/postgresql/8.4/archiver.ini -I
```
2. Créer le répertoire de stockage des journaux d'échange, en tant qu'utilisateur postgres, sur le Slave :
(Aucun problème si la fonction retourne une erreur dans le cas où le répertoire existe déjà)

```
postgres@sqlW:~$ cmd_archiver -C /etc/postgresql/8.4/archiver.ini -I
```

B - Initialisation du Master

1. Mettre en place l'IP Alias sur sqlV.domain.com :

```
root@sqlV:~# perl -pi -e 's/\#auto eth0:1/auto eth0:1/g' /etc/network/interfaces
root@sqlV:~# ifup eth0:1
```
2. Passer le paramètre archive_mode à on dans le fichier de configuration /etc/postgresql/8.4/main/postgresql.conf
3. Créer les deux liens symboliques suivants :

```
root@sqlV:~# ln -sf /etc/postgresql/8.4/main/postgresql.conf /opt/data/postgresql/8.4/main/
root@sqlV:~# ln -sf /etc/postgresql/8.4/main/pg_hba.conf /opt/data/postgresql/8.4/main/
```
4. Redémarrer le serveur PostgreSQL:

```
/etc/init.d/postgresql-8.4 restart
```

A partir de là il est possible d'utiliser le serveur Master exactement comme un serveur normal, dès à présent. Il peut recevoir les connexions SQL.

C - Initialisation du Slave

1. Arrêter la base de données si elle est active :

```
root@sqlW:~# /etc/init.d/postgresql-8.4 stop
```
2. Préparer les répertoires de réplication :
(la commande cmd_standby demandera le mot de passe de l'utilisateur DBUSER postgres de sqlV.domain.com)

```
postgres@sqlW:~$ rm -rf /opt/data/postgresql/8.4/mainOLD/
postgres@sqlW:~$ mv /opt/data/postgresql/8.4/main /opt/data/postgresql/8.4/mainOLD/
postgres@sqlW:~$ cmd_standby -C /etc/postgresql/8.4/standby.ini -I
```
3. Commencer la restauration par une sauvegarde de base :
(la commande cmd_standby demandera, plusieurs fois, le mot de passe de l'utilisateur DBUSER postgres de sqlV.domain.com)

```
postgres@sqlW:~$ cmd_standby -C /etc/postgresql/8.4/standby.ini -B
```



4. Recréer les deux liens symboliques suivants :

```
root@sqlW:~# ln -sf /etc/postgresql/8.4/main/postgresql.conf
/opt/data/postgresql/8.4/main/
root@sqlW:~# ln -sf /etc/postgresql/8.4/main/pg_hba.conf /opt/data/postgresql/8.4/main/
```

5. Lancer la synchronisation du Slave avec le Master :

```
postgres@sqlW:~$ cmd_standby -C /etc/postgresql/8.4/standby.ini -S
```

A partir de cette étape, et après un laps de temps (qui dépend de plusieurs paramètres) un message indique que la restauration automatique est en cours et tout mouvement sur le Master sera répliqué sur le Slave.

VII - Maintenance de la réplication - Switch Master <-> Slave

Alors que toutes les instructions précédentes sont à réaliser pour l'installation initiale d'un serveur base de données PostgreSQL, les instructions figurant dans cette partie VII sont, quant à elle, à déclencher à chaque fois qu'un switch Master / Slave est souhaité (c'est-à-dire passer le Slave en Master et le Master en Slave)

Afin de fournir une procédure générique aux 2 situations suivantes:

- ☐ soit sql1.so.apps.valeo.com est le Master et sql1.lc.apps.valeo.com est le Slave
- ☐ soit sql1.lc.apps.valeo.com est le Master et sql1.so.apps.valeo.com est le Slave

=> nous dénommerons par la suite sql1.X.apps.valeo.com le serveur agissant en tant que Master et sql1.Y.domain.com pour l'autre serveur, agissant, de facto, en Slave.

A - Actions à effectuer sur le Master sql1.X.apps.valeo.com, afin de le désactiver en tant que Master

2 cas peuvent ici survenir :

a - (1er cas) le Master sql1.X.apps.valeo.com est fonctionnel et accessible

1. Demander à publier toutes les dernières modifications :
(Répondre à "Yes" à la question "I wish to force my slave into production: No/Yes")

```
postgres@sql1.X:~$ cmd_archiver -C /etc/postgresql/8.4/archiver.ini -f
```
2. Mettre le paramètre archive_mode à off dans le fichier de configuration
/etc/postgresql/8.4/main/postgresql.conf



3. Arrêt de la base de données

```
root@sql1.X:~# /etc/init.d/postgresql-8.4 stop
```

4. Suppression de l'IP Alias du Master

```
root@sqlX:~# ifdown eth0:1
```

```
root@sqlX:~# perl -pi -e 's/auto eth0:1/\#auto eth0:1/g' /etc/network/interfaces
```

5.

6. **Puis passer à l'étape suivante** "B - Actions à effectuer sur le Slave

sql1.Y.apps.valeo.com, afin de le rendre autonome"

b - (2ème cas) le Master sql1.X.apps.valeo.com n'est pas fonctionnel ou n'est pas accessible

1. Débrancher sql1.X.apps.valeo.com du réseau pour qu'il soit totalement inaccessible de l'extérieur

2. Désactiver l'IP Alias si sql1.X.apps.valeo.com fonctionne:

```
root@sqlX:~# ifdown eth0:1
```

```
root@sqlX:~# perl -pi -e 's/auto eth0:1/\#auto eth0:1/g' /etc/network/interfaces
```

3. **Passer dans tous les cas à l'étape suivante** "B - Actions à effectuer sur le Slave sqlY.domain.com, afin de le rendre autonome"

B - Actions à effectuer sur le Slave sql1.Y.domain.com, afin de le rendre autonome

1. Passer l'IPAlias master sur ce serveur (cf. plus haut)

```
root@sqlX:~# perl -pi -e 's/\#auto eth0:1/auto eth0:1/g' /etc/network/interfaces
```

```
root@sql1.Y:~# ifup eth0:1
```

2. Supprimer les liens symboliques

```
root@sql1.Y:~# rm -f /opt/data/postgresql/8.4/main/postgresql.conf
```

```
root@sql1.Y:~# rm -f /opt/data/postgresql/8.4/main/pg_hba.conf
```

3. Relancer le serveur en restaurant les journaux

```
postgres@sql1.Y:~$ cmd_standby -C /etc/postgresql/8.4/standby.ini -F999
```

4. Le Slave devient ainsi un serveur autonome prêt à accepter les connexions SSL.

C - Remise en état de fonctionnement du Master sql1.X.apps.valeo.com, s'il était précédemment down

1. Effectuer toutes les réparations nécessaires, comme changement d'une barette mémoire etc...

2. Mettre le paramètre archive_mode à off dans le fichier de configuration /etc/postgresql/8.4/main/postgresql.conf

3. Redémarrer le serveur PostgreSQL :

```
/etc/init.d/postgresql-8.4 restart
```



4. Désactiver l'IP Alias sql1.X.apps.valeo.com :

```
root@sqlX:~# ifdown eth0:1
root@sqlX:~# perl -pi -e 's/auto eth0:1/\#auto eth0:1/g' /etc/network/interfaces
```

5. Rebrancher sql1.X.apps.valeo.com au réseau

D - Réactivation de la réplication

Les étapes précédentes A, B et C ont permis de désactiver le Master et de rendre le Slave en serveur autonome afin qu'il accepte rapidement les connexions SSL. Cependant à cette étape la réplication n'est pas active.

Il reste donc à l'activer en suivant les procédures de l'étape "VI - Initialisation de la réplication" en définissant :

- ☐ sqlY (ancien slave, devenu serveur autonome) en Master
- ☐ sqlX (ancien master qui a été arrêté) en Slave

Si, à la suite de ces opérations, on se trouve dans un état où le Master n'est pas celui souhaité, refaire alors procédure VII pour les switcher.

VIII - Retour à Zero

Si, pour une raison X ou Y, on souhaite purger complètement PostgreSQL, afin de reprendre la procédure d'installation de la réplication à partir du début, il est nécessaire de suivre la procédure suivante :

1. Supprimer les paquets postgresql

```
apt-get autoremove --purge postgresql postgresql-client postgresql-contrib postgresql-common postgresql-client-common
```

(Attention au copier/coller sur 2 lignes)

2. Vérifier que tous les paquets ont bien été supprimés en lançant la commande suivante, elle ne doit renvoyer aucun paquet.

```
dpkg -l | grep postgres
```

3. Si c'est le cas, les supprimer alors:

```
apt-get autoremove --purge <package>
```

4. Vérification de la suppression du répertoire de data :

```
ls -la /opt/data/postgresql/
```

5. Si le répertoire existe, il faut le supprimer

```
rm -rf /opt/data/postgresql/
```

6. Supprimer les fichiers standby.ini et archiver.ini

```
rm /etc/postgresql/8.4/archiver.ini /etc/postgresql/8.4/standby.ini
```

7. Supprimer les répertoires à la main (lié à xqueries) :

```
rm -rf /usr/lib/postgresql/ /usr/share/postgresql/
```



8. Supprimer l'utilisateur postgres
`userdel postgresq`
9. Supprimer l'ancien répertoire de database
`rm -rf /var/lib/postgresql.old`

IX - Personnalisation du document

Le présent document est une version générique.

Au besoin de faire une version spécifique adaptée à l'architecture d'un réseau réel, il sera nécessaire de remplacer dans ce document toutes les occurrences de :

- /var/lib/postgresql par /opt/data/postgresql
- sql1.domain.com par sql1.so.apps.valeo.com
- sql2.domain.com par sql1.lc.apps.valeo.com
- sqlmaster.domain.com par db-server-postgres.apps.valeo.com
- sqlfriend.domain.com par db-slave-postgres.apps.valeo.com
- sqlX.domain.com par sql1.X.apps.valeo.com
- sqlY.domain.com par sql1.Y.apps.valeo.com
- Modification des ips dans les /etc/hosts et le hba_pg.conf
- Ajout de IV – C – Changement du répertoire de donnée postgresql
- Renommage du IV –C en IV –D Utilisateur Système postgres
- V – A : Pour rajouter la copie de clé ssh dans les 2 sens (nécessaire en cas de bascule).
- VIII : Retour à Zéro pour rajouter certaines commandes de nettoyage.
- V2.5_Agarik -> v2.6_Agarik : Modification des fichiers /etc/hosts pour mettre db-slave au bonne endroit.
- Ajout du chapitre IV – E : Configuration de l'alias IP pour la préconfiguration de l'alias (mais sans lancement automatique).
- Modification du chapitre VI – B -1 : Remplacement du ifconfig par un ifup pour monter l'alias et l'activer au démarrage
- Modification VII – A – b -4 : Remplacement du ifconfig par un ifdown pour démonter l'alias et le désactiver au démarrage
- Modification VII – A – a -4 : Remplacement du ifconfig par un ifdown pour démonter l'alias et le désactiver au démarrage
- Modification VII –B – 1 : Remplacement du ifconfig par un ifdown pour monter l'alias et l'activer au démarrage



Annexe I - Fichier “archiver.ini”



```
[DEFAULT]
; online or offline
state: online

; The base database directory
pgdata: /opt/data/postgresql/8.4/main

; where to remotely copy archives
r_archivedir: /opt/data/postgresql/8.4/archive

; where to locally copy archives
l_archivedir: /opt/data/postgresql/8.4/archive

; where is rsync
rsync_bin: /usr/bin/rsync

; option 2 or 3, if running RHEL5 or similar it is likely 2
; if you are running something that ships remotely modern software
; it will be 3

rsync_version = 3

; IP of slave
slaves: db-slave-postgres.apps.valeo.com

; the user that will be using scp
user: postgres

; if scp can't connect in 10 seconds error
timeout: 10

; command to process in ok
notify_ok: echo OK

; command to process in warning
notify_warning: echo WARNING

; command to process in critical
notify_critical: echo CRITICAL

; if you want to debug on/off only
debug: off

; if you want ssh debug (warning noisy)
ssh_debug: off
```



Annexe II - Fichier “standby.ini”



```
[DEFAULT]
; what major version are we using?
pgversion: 8.4

; Used for 8.2 (8.1?), should be set to something > than checkpoint_segments on master
numarchives: 10

; Commands needed for execution
; absolute path to ssh
ssh: /usr/bin/ssh

; absolute path to rsync
rsync: /usr/bin/rsync

; the path to the postgres bin
pg_standby: /usr/lib/postgresql/8.4/bin/pg_standby
pg_ctl: /usr/lib/postgresql/8.4/bin/pg_ctl

; path to psql on the master
r_psql: /usr/lib/postgresql/8.4/bin/psql

; Generalized information

; the port postgresql runs on (master)
port: 5432

; ip or name of master server
master_public_ip: db-server-postgres.apps.valeo.com

; the ip address we should use when processing remote shell
master_local_ip: 127.0.0.1

; the user performed initdb
user: postgres

; on or off
debug: off

; the timeout for ssh before we throw an alarm
ssh_timeout: 30

; should be the same as r_archivedir for archiver
archivedir: /opt/data/postgresql/8.4/archive

; where you executed initdb -D to
pgdata: /opt/data/postgresql/8.4/main

; Confs

; This is the postgresql.conf to be used when not in standby
postgresql_conf: /etc/postgresql/8.4/main/postgresql.conf

; This is the pg_hba.conf to be used when not in standby
pg_hba_conf: /etc/postgresql/8.4/main/pg_hba.conf

; Alarms

notify_critical: ls
```



```
notify_warning:
notify_ok:

; On failover action

; Whatever is placed here will be executed on -FS must return 0

action_failover: /opt/data/postgresql/8.4/failover.sh
```

