

Serveur de sauvegarde BackupPC

Debian GNU/Linux



Matthieu Vogelweith
17 août 2009

Résumé

L'objectif de ce document est de détailler l'installation d'un serveur sauvegarde BackupPc [1] sous Debian Lenny [2].

Ce document a été rédigé en LaTeX en utilisant l'excellent Vim sous Debian GNU/Linux. Il est disponible aux formats XHTML et PDF. Les sources LaTeX sont disponibles ici : [L^AT_EX](#)

Licence

Copyright ©2009 Matthieu VOGELWEITH <matthieu@vogelweith.com>.

Vous avez le droit de copier, distribuer et/ou modifier ce document selon les termes de la GNU Free Documentation License, Version 1.3 ou ultérieure publiée par la Free Software Foundation ; avec aucune section inaltérable, aucun texte de première page de couverture, et aucun texte de dernière page de couverture. Une copie de la licence est disponible dans la page [GNU Free Documentation License](#).

Historique

– 17-08-2009 : Version initiale

Table des matières

Table des matières	4
1 Préparation	5
1.1 Pré requis	5
1.2 Installation	5
1.3 Configuration d'Apache	5
2 Sauvegarde des serveurs Linux	6
2.1 Préparation	6
2.2 Configuration des base	6
2.3 Sauvegarde des bases de données et annuaires LDAP	6
3 Sauvegarde des postes Windows	8
4 Références	9

Chapitre 1

Préparation

1.1 Pré requis

Ce document suppose que le serveur est déjà installé avec une Debian Lenny [2] propre. L'installation et la configuration du système de base sont présentées en détail dans un document dédié à cet effet : [?].

1.2 Installation

```
# aptitude install backuppc
```

1.3 Configuration d'Apache

Chapitre 2

Sauvegarde des serveurs Linux

2.1 Préparation

```
# aptitude install sudo rsync
```

Dans /etc/sudoers :

```
backup    ALL=(root) NOPASSWD:/usr/bin/rsync
```

```
# chown backup.backup /var/backups
```

2.2 Configuration des base

2.3 Sauvegarde des bases de données et annuaires LDAP

Dans /etc/cron.d/backups :

```
# /etc/cron.d/backups: Backup crontab

# PostgreSQL backup (All)
00 20 * * *    root    [ -x /usr/bin/pg_dumpall ] && mkdir -p /var/backups/postgres && if
    [ -f /var/backups/postgres/pg_dumpall.0.sql.gz ]; then mv -f /var/backups/postgres/
    pg_dumpall.0.sql.gz /var/backups/postgres/pg_dumpall.1.sql.gz; fi && su - postgres -
    c /usr/bin/pg_dumpall | gzip > /var/backups/postgres/pg_dumpall.0.sql.gz

# PostgreSQL backup
30 20 * * *    root    [ -x /usr/bin/pg_dump ] && mkdir -p /var/backups/postgres && for
    db in `su - postgres -c "psql -lt | grep -v 'template[01]' | cut -d'|' -f 1"`; do if
    [ -f /var/backups/postgres/pg_dump_${db}.0.sql.gz ]; then mv -f /var/backups/postgres
    /pg_dump_${db}.0.sql.gz /var/backups/postgres/pg_dump_${db}.1.sql.gz; fi && su -
    postgres -c "pg_dump ${db}" | gzip > /var/backups/postgres/pg_dump_${db}.0.sql.gz; done

# MySQL backup (All)
00 21 * * *    root    [ -x /usr/bin/mysqldump ] && mkdir -p /var/backups/mysql && if [ -
    f /var/backups/mysql/mysqldump_all.0.sql.gz ]; then mv -f /var/backups/mysql/
    mysqldump_all.0.sql.gz /var/backups/mysql/mysqldump_all.1.sql.gz; fi && /usr/bin/
    mysqldump --all-databases | gzip > /var/backups/mysql/mysqldump_all.0.sql.gz
```

```
# MySQL backup
30 21 * * * root [ -x /usr/bin/mysqldump ] && mkdir -p /var/backups/mysql && for db
  in `usr/bin/mysql -Bse 'show databases'`; do if [ -f /var/backups/mysql/
mysqldump_${db}.0.sql.gz ]; then mv -f /var/backups/mysql/mysqldump_${db}.0.sql.gz /var/
backups/mysql/mysqldump_${db}.1.sql.gz; fi && /usr/bin/mysqldump --databases $db |
gzip > /var/backups/mysql/mysqldump_${db}.0.sql.gz; done

# LDAP backup
00 22 * * * root [ -x /usr/sbin/slapcat ] && mkdir -p /var/backups/ldap && if [ -f
/var/backups/ldap/slapcat.0.ldif.gz ]; then mv -f /var/backups/ldap/slapcat.0.ldif.
gz /var/backups/ldap/slapcat.1.ldif.gz; fi && /usr/sbin/slapcat | gzip > /var/
backups/ldap/slapcat.0.ldif.gz
```

```
# chmod +x /etc/cron.d/backups
```

Note : Si l'utilisateur root de MySQL ne peut se connecter sans mot de passe (ce qui est une bonne idée ...) il est possible de définir le mot de passe dans les commandes ci-dessus ou de créer un fichier de préférences pour l'utilisateur root. La deuxième solution est préférable puisqu'on pourra définir des permissions plus restrictives sur le fichier de préférences. Par exemple, pour définir le mot de passe à utiliser pour se connecter sur le serveur 127.0.0.1, ajouter les lignes suivantes dans /root/.my.cnf :

```
[client]
user = root
password = my_root_password
host = 127.0.0.1
```

Ne pas oublier ensuite de restreindre les droits d'accès avec la commande suivante :

```
# chmod 600 /root/.my.cnf
```

Chapitre 3

Sauvegarde des postes Windows

Chapitre 4

Références

[1] Site officiel de backuppc. <http://backuppc.sourceforge.net>.

[2] Site officiel du projet debian. www.debian.org.