

# Backup avec BackupPC

## Installation de BackupPC

### BackupPC 3.4 :

BackupPC 3.4 est disponible sur les dépôts officiel de Debian:

```
# apt install backuppc
```

Il faut configurer Apache2, donc commencer par :

```
# rm /etc/apache2/sites-enabled/*
```

Il faut ensuite créer un VirtualHost d'Apache2 dans */etc/apache2/sites-available* comme ceci :

[backuppc.conf](#)

```
<VirtualHost *:80>
  DocumentRoot /usr/share/backuppc/cgi-bin
  Alias /backuppc /usr/share/backuppc/cgi-bin/
  <Directory /usr/share/backuppc/cgi-bin/>
    AllowOverride None
    Allow from all

    Options ExecCGI FollowSymlinks
    AddHandler cgi-script .cgi
    DirectoryIndex index.cgi

    AuthUserFile /etc/backuppc/htpasswd
    AuthType basic
    AuthName "BackupPC admin"
    require valid-user
  </Directory>
</VirtualHost>
```

Puis de l'activer :

```
# a2ensite backuppc.conf
# systemctl reload apache2
```

On va aussi définir un mot de passe à l'utilisateur backuppc en rentrant la commande :

```
# htpasswd /etc/backuppc/htpasswd backuppc
```

Il faut ensuite créer la clés SSH de l'utilisateur backuppc en lançant les commandes :

```
# sudo -i -u backuppc
# ssh-keygen -t rsa -b 4096
# exit
```

Attention à ne pas mettre de Passphrase a la clé.

## Facultatif

Pour changer l'emplacement de travail de BackupPC (en l'occurrence des backups), il faut d'abord copier le dossier de travail dans la nouvelle destination :

```
# rsync -aXv /var/lib/backuppc /path/to/dest/
```

puis de modifier les champs suivants dans la fichier `/etc/backuppc/config.pl` :

```
$Conf{TopDir}      = '/path/to/dest/';
$Conf{LogDir}     = '/path/to/dest/log';
```

Puis relancer le service

```
# systemctl restart backuppc
```

## BackupPC 4 :

Attention, cela nécessite de compiler le packet.

On commence par installer les dépendances nécessaires :

```
apt-get install -y apache2 apache2-utils libapache2-mod-perl2 glusterfs-
client par2 perl smbclient rsync tar sendmail gcc zlib1g zlib1g-dev
libapache2-mod-scgi rrdtool git make perl-doc libarchive-zip-perl libfile-
listing-perl libxml-rss-perl libcgi-session-perl
```

On créer le dossier qui contiendra les fichiers BackupPC :

```
# mkdir /var/lib/backuppc
```

Puis on créer l'utilisateur backuppc :

```
# adduser --system --home /var/lib/backuppc --group --disabled-password --
shell /bin/false backuppc
```

Ensuite on créer les clés ssh:

```
# mkdir -p /var/lib/backuppc/.ssh
```

```
# ssh-keygen -q -t rsa -b 4096 -N '' -C "BackupPC key" -f
```

```
/var/lib/backuppc/.ssh/id_rsa
```

```
# echo -e "BatchMode yes\nStrictHostKeyChecking no" >  
/var/lib/backuppc/.ssh/config
```

Et on redéfinit les droits :

```
chmod 700 /var/lib/backuppc/.ssh && chmod 600 /var/lib/backuppc/.ssh/id_rsa  
&& chmod 644 /var/lib/backuppc/.ssh/id_rsa.pub && chmod 644  
/var/lib/backuppc/.ssh/config && chown -R backuppc:backuppc  
/var/lib/backuppc/.ssh
```

On git clone ensuite les trois dépôts de BackupPC :

```
# git clone https://github.com/backuppc/backuppc.git && git clone  
https://github.com/backuppc/backuppc-xs.git && git clone  
https://github.com/backuppc/rsync-bpc.git
```

Ensuite pour chaque packets :

### backuppc-xs :

```
# cd backuppc-xs
```

```
# git tag | tail -1 | xargs git checkout
```

```
# perl Makefile.PL
```

```
# make
```

```
# make test
```

```
# make install
```

```
# cd ..
```

### rsync-bpc

```
# cd rsync-bpc
```

```
# git tag | tail -1 | xargs git checkout
```

```
# ./configure
```

```
# make
```

```
# make install
```

## backuppc

```
# cd backuppc
```

```
# git tag | tail -1 | xargs git checkout
```

```
# ./configure.pl --batch --cgi-dir /usr/local/backuppc/cgi-bin --data-dir /var/lib/backuppc --hostname backuppc --html-dir /usr/local/backuppc/cgi-bin --html-dir-url '' --install-dir /usr/local/backuppc --config-dir /etc/backuppc
```

```
# cp systemd/backuppc.service /etc/systemd/system/ && systemctl daemon-reload
```

```
# cd ..
```

Ensuite on supprime les configurations Apache2 par défaut

```
# rm /etc/apache2/sites-enabled/*
```

et on injecte cette configuration dans le dossier */etc/apache2/sites-available/*

### backuppc.conf

```
<VirtualHost *:80>
  DocumentRoot /usr/local/backuppc/cgi-bin
  <Directory /usr/local/backuppc/cgi-bin/>
    AllowOverride None
    Allow from all

    Options ExecCGI FollowSymLinks
    DirectoryIndex BackupPC_Admin

  <FilesMatch "BackupPC_Admin$">
    SetHandler cgi-script
  </FilesMatch>
  AuthUserFile /etc/backuppc/htpasswd
  AuthType basic
  AuthName "BackupPC admin"
  require valid-user
</Directory>
</VirtualHost>
```

Ensuite on va demander a Apache2 d'utiliser l'utilisateur backuppc plutot que le www-data :

```
# sed -i 's/www-data/backuppc/' /etc/apache2/envvars
```

On active les confs et les modules :

```
# a2enmod cgid && a2ensite backuppc.conf
```

Ensuite on redémarre apache2 :

```
# systemctl restart apache2
```

On crée ensuite le fichier de mot de passe :

```
# rm /etc/backuppc/BackupPC.users && touch /etc/backuppc/htpasswd
```

et ensuite il faut créer un utilisateur

```
# htpasswd /etc/backuppc/htpasswd $USERNAME
```

et il faudra ajouter ce compte en tant qu'admin de backuppc dans le fichier `/etc/backuppc/config.pl`

```
2032. #    --> administrative users are the union of groups admin and wheel,  
      plus  
2033. #    craig and celia.  
2034. #  
2035. #    $Conf{CgiAdminUserGroup} = '';  
2036. #    $Conf{CgiAdminUsers}     = 'craig celia';  
2037. #    --> administrative users are only craig and celia'.  
2038. #  
2039. $Conf{CgiAdminUserGroup} = '';  
2040. $Conf{CgiAdminUsers}     = '$USERNAME';
```

et pour finir, on redémarre le daemon backuppc :

```
# systemctl restart backuppc
```

From:

<https://wiki.virtit.fr/> - VirtIT

Permanent link:

[https://wiki.virtit.fr/doku.php/kb:linux:donnees:backup\\_avec\\_backuppc?rev=1547226687](https://wiki.virtit.fr/doku.php/kb:linux:donnees:backup_avec_backuppc?rev=1547226687)

Last update: **2019/01/11 17:11**

