

Backup avec BackupPC

Installation de BackupPC

BackupPC 3.3 :

BackupPC 3.3 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 a 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. J'ai donc réalisé ce script pour le faire a votre place :)

[backuppc4-setup.sh](#)

```
#!/bin/bash

if [ -z "$1" ];then
    echo -e "usage : ./backuppc4-setup.sh --install (or upgrade from
backuppc 3)\n\n: ./backuppc4-setup.sh --upgrade\n\nMUST BE RUN AS ROOT"
    exit 1
fi
if [[ $EUID -ne 0 ]]; then
    echo "This script must be run as root"
    exit 1
fi

case $1 in
    --install)
        echo -e "CAREFUL !\n\nIf you're installing from backuppc
3, backups and hosts will (normally) be conserved, \n\nBUT some
configuration need to be changed manually by you (Apache2, backup
settings,...).\n\nBackup your /etc/backuppc directory, and Good luck !"
        read -n 1 -s -r -p "Press any key to continue"
        echo "PURGE DE BACKUPPC ET AJOUT DES DÉPENDENCES"
```

```

    apt remove backuppc
    apt-get install -y par acl libacl1-dev 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

    if [ ! -d "/var/lib/backuppc" ]; then
        mkdir /var/lib/backuppc
        echo "CREATION DE L'UTILISATEUR BACKUPPC"
        adduser --system --home /var/lib/backuppc --
group --disabled-password --shell /bin/bash backuppc
        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" >
/var/lib/backuppc/.ssh/config
        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
    fi

    mkdir /opt/backuppc-build-env
    pushd /opt/backuppc-build-env

    echo "RECUPERATION DES DÉPÔTS 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

    echo "COMPILATION DE BACKUPPC-XS"
    cd backuppc-xs
    git tag | tail -1 | xargs git checkout
    perl Makefile.PL
    make
    make test
    make install

    echo "COMPILATION DE RSYNC-BPC"
    cd ../rsync-bpc
    git tag | tail -1 | xargs git checkout
    ./configure
    make
    make install

    echo "COMPILATION DE BACKUPPC"
    cd ../backuppc
    git tag | tail -1 | xargs git checkout
    ./makeDist --nosyntaxCheck --releasedate "`date -u "+%d

```

```

%b %Y" ` " --version git
    tar -zxf dist/BackupPC-git.tar.gz
    cd BackupPC-{bpcver}git
    if [ -f "/etc/backuppc/config.pl" ]; then
        ./configure.pl --batch --config-path
/etc/backuppc/config.pl
    else
        ./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
    fi

    echo "CONFIGURATION DE BACKUPPC"
    cp systemd/backuppc.service /etc/systemd/system/ &&
systemctl daemon-reload
    systemctl enable backuppc && systemctl start backuppc

    echo "CONFIGURATION D'APACHE2"
    cat << EOF > /etc/apache2/sites-available/001-
backuppc.conf
<VirtualHost *:80>
    DocumentRoot /usr/local/backuppc/cgi-bin

    Alias /hw /var/www/hw/

    <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>

EOF

    sed -i 's/www-data/backuppc/' /etc/apache2/envvars
a2dissite *
a2enmod cgid && a2ensite 001-backuppc.conf

    systemctl restart apache2

```

```
        if [ ! -f "/etc/backuppc/htpasswd" ]; then
            touch /etc/backuppc/htpasswd
            htpasswd -b /etc/backuppc/htpasswd 'admin'
'backuppc'
        fi
    popd
;;
--upgrade)
    systemctl stop backuppc
    if [ ! -d "/opt/backuppc-build-env" ]; then
        mkdir /opt/backuppc-build-env
    fi
    pushd /opt/backuppc-build-env

    echo "COMPILATION ET MISE A JOUR DE BACKUPPC-XS"
    if [ -d "backuppc-xs" ]; then
        cd backuppc-xs
        git pull
    else
        git clone https://github.com/backuppc/backuppc-
xs.git
        cd backuppc-xs

    fi
    git tag | tail -1 | xargs git checkout
    perl Makefile.PL
    make
    make test
    make install
    cd ..

    echo "COMPILATION ET MISE A JOUR DE RSYNC-BPC"
    if [ -d "rsync-bpc" ]; then
        cd rsync-bpc
        git pull
    else
        git clone https://github.com/backuppc/rsync-
bpc.git
        cd rsync-bpc

    fi
    git tag | tail -1 | xargs git checkout
    ./configure
    make
    make install
    cd ..

    echo "COMPILATION ET MISE A JOUR DE BACKUPPC"
    if [ -d "backuppc" ]; then
        cd backuppc
```

```
                git pull
            else
                git clone
                https://github.com/backuppc/backuppc.git
                cd backuppc

            fi
            git tag | tail -1 | xargs git checkout
            ./configure.pl --batch --config-path
/etc/backuppc/config.pl
            popd
            systemctl start backuppc

        ;;
    *)
        echo -e "usage : ./backuppc-setup4.sh --install (or
upgrade from backuppc 3)\nor: ./backuppc4-setup.sh --upgrade"
        exit 1
    ;;
esac

echo "FINI !"
```

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

Permanent link:
https://wiki.virtit.fr/doku.php/kb:linux:donnees:backup_avec_backuppc?rev=1560809059

Last update: **2019/06/17 22:04**

