BackupPC

Mr. Bruno Vernay

BackupPC

Mr. Bruno Vernay

Table of Contents

Chapter 1. Introduction

Le but de ce document n'est pas de remplacer la documentation de BackupPC dont la lecture reste indispensable, ainsi que l'abonnement à la mailing-list des utilisateurs pour qui veut exploiter cette application. Par ailleur, cette application n'est qu'un élément de la sauvegarde qui pour être fiable doit être régulierement testée en restaurant des fichiers.

BackupPC peut utiliser plusieurs moyens de transferts : Samba, rsync, rsyncd ... avec ou sans SSH. Je n'utilise et décri que la partie rsync avec SSH (et ssh-agent) qui est le plus sécurisé (pas de clées sans mot de passe), optimisé (saute les fichiers déjà existants) et fonctionne aussi sous MS-Windows (je n'ai pas encore configuré SSH sous MS-Windows.)

Pour ne pas perdre de temps, autant aller voir tout de suite les "limitations" dans la doc et annoncer ici que les ACL qui n'ont pas d'équivalent Unix ne sont pas sauvegardé (version 2.0.2) Autant citer directement l'auteur : [BackupPC_tarExtract doesn't understand the richer tar file format that star uses. In principal it could be supported, which would allow support for ACLs, UTF8 file names etc. ACL support is on the long-term feature list for BackupPC, mainly being paced by when smbclient, tar and rsync provide ACL support. It looks like star could be the natural upgrade to tar.] Une solution dégradée consisterait à utiliser des utilitaires (je n'ai pas les références) pour écrire/restaurer les ACL dans des fichiers textes.

Chapter 2. Installation du serveur

Prérequis

Configuration minimale des clients

Une des limitations à garder à l'esprit concerne la sauvegarde de machines avec beaucoup de fichiers et peu de mémoire. Cela vient de rsync : je cite la FAQ (au 01/01/2004) : [Yes, rsync uses a lot of memory. The majority of the memory is used to hold the list of files being transferred. This takes about 100 bytes per file, so if you are transferring 800,000 files then rsync will consume about 80M of memory. It will be higher if you use -H or --delete. To fix this requires a major rewrite of rsync. I do plan on doing that, but I don't know when I'll get to it].

On peut contourner le problème en sauvegardant répertoire par répertoire et pas directement "/".

Le problème est valable sur le serveur comme sur le client.

Limitations et systèmes de fichiers

Il est conseillé d'utiliser un système de fichier ReiserFS ou XFS (c'est à dire basés sur des b-tree, plutôt que Ext3.) pour des raisons de performance. En effet, les performances de Ext3 diminuent avec le nombre de fichiers.

En cas de messages d'erreurs signalant qu'il ne reste plus d'espace disque alors que la commande **df** indique qu'il reste de la place, il faut verifier le nombre d'`i-node` disponibles ! Pour cela utiliser la commande **df -i**.

Perl

Attention, il est conseillé d'utiliser une version 5.8.1 au minimum pour le support d'UTF-8 ! (Si vous avez encore 5.8.0 : voir le BUG Red Hat : https://bugzilla.redhat.com/bugzilla/show_bug.cgi?id=87682. Pour l'instant, le workaround consiste à se passer d'unicode : **export LANG=fr_FR** avant d'utiliser perl et **export LANG=fr_FR.UTF-8** après.)

Il faut installer certains modules supplémentaires et pour cela utiliser le CPAN pour les trouver sur le net.

Il suffit de lancer **perl -MCPAN -e shell**. La première fois, il demande quelques paramètres, il suffit d'avoir sous la main une liste de serveurs CPAN (il en existe plus de 200), que l'on peut trouver là : http://www.cpan.org/SITES.html#Europe. On a mis le cache dans `/var/lib/cpan/`.

Puis on peut installer les modules avec **perl -MCPAN -e 'install Chocolate::Belgian'**.

Il faut vérifier que Perl soit compilé avec le support pour les gros fichiers : **perl -V | egrep large** doit donner `uselargefiles=define`. Si on a `uselargefiles=undef`, il sera impossible d'avoir des fichier de plus de 2GB.

Utilisateur backuppc

Création d'un utilisateur `backuppc` simple pour l'instant.

BackupPC should run as a dedicated user with limited privileges. This user will need read/write permission on the main data directory and read/execute permission on the install directory.

The primary group for this user should also be chosen carefully. By default the install directories will have group write permission. The data directories and files will have group read permission but no other permission.

Installation

Paramètres

Attention, dans la documentation BackupPC le nom des répertoires est souvent avec des majuscules par défaut. Il est préférable de n'utiliser que des minuscules. (Noter qu'on utilise le répertoire /srv/ conformément aux recommandations FHS [http://samba.org/~cyeoh/pub/fhs-2.3.html] qui font partie du Linux Standard Base [http://www.linuxbase.org/].)

- Répertoire d'installation : `/usr/local/backuppc`
- données : `/var/backuppc/`
- cgi-bin : `/srv/www/cgi-bin/backuppc`
- Images pour le web : `/srv/www/html/backuppc` et URL pour les images : `/backuppc`

Mise en service (init.d)

Il est possible de faire tourner BackupPC en service. (Cette opération doit être faite après l'installation, après le "perl configure.pl".)

Pour avoir des informations à jours, suivre les instructions de `/usr/src/BackupPC-2.0.2/init.d/README`.

Copier le fichier `/usr/src/BackupPC-2.0.2/init.d/suse-backuppc` dans `/etc/ini.d/backuppc`. Le rendre executable (chmod 755) et lancer les commandes suivantes :

```
chkconfig backuppc 345       # démarre aux levels 3, 4 et 5.
chkconfig --list backuppc    # pour vérifier
```

Interface Web

Il est important de noter que Apache2 doit tourner avec un utilisateur particulier. Cette configuration concerne tout Apache et ne peut être changé par site. Il existe deux solutions : changer directement les fichiers de configurations ou faire une copie des fichiers de configurations différents pour garder la possibilité de lancer une autre instance "classique" d'Apache2. J'ai choisi la seconde solution.

BackupPC expose deux méthodes, on utilise la méthode mod_perl réputée plus performante (et non la "standard".) Activer l'interface web n'est pas si simple, car il faut qu'Apache tourne sous l'utilisateur `backuppc`, que le client soit authentifié et utilise mod_perl.

On va donc recopier `/etc/init.d/apache2` en `apache2-backuppc` et le modifier pour utiliser `/etc/apache2/httpd-backuppc.conf` ainsi que pour le distinguer du service Apache2. Listing des fichiers : En annexe.

De plus il faut changer les droits du cgi-bin : pour enlever le setuid (inutile dans notre configuration) : **chmod u-s __CGIDIR__/BackupPC_Admin**.

Pour l'authentification LDAP, le module `/usr/lib/apache2/mod_auth_ldap.so` est fourni avec Apache (au moins pour Suse et Mandrake.) Mais il n'est pas actif par défaut, surement parce qu'il est marqué comme expérimental. Il suffit de rajouter dans `/etc/sysconfig/apache2` les entrées "ldap auth_ldap" à la variable `APACHE_MODULES` (il ne faut pas mettre le "mod_" du fichier.) Lors du démarrage d'apache2 ces modules sont ajoutés au fichier `/etc/apache2/sysconfig.d/loadmodule.conf` qui est inclu dans `httpd.conf`.

Pour lancer le serveur il suffit d'utiliser **/etc/init.d/apache2-backuppc start**.

Pour mettre le nouveau service http en service **chkconfig apache2-backuppc 345**. (Il ne faut pas oublier de modifier la partie "`INIT_INFO`" du script pour le distinguer du service `apache2` et `httpd2`.)

On accède au site par l'url http://123.124.125.126/cgi-bin/backuppc/BackupPC_Admin.

SSH

Ssh ne doit produire aucune sortie, il doit être completement transparent pour rsync, d'où l'option `-q`. De plus, il faut se connecter une premiere fois pour ajouter le serveur à la liste des hotes connus, ou le faire manuellement. On peut tester le bon fonctionnement de ssh en faisant : **ssh -q -l root 123.124.125.126 whoami > test.txt** : `text.txt` doit contenir "root" et seulement "root".

Attention, la documentation de BackupPC montre l'utilisation de clée avec une phrase vide (option **-N ''**). Cela permet d'utiliser les clées sans interventions humaine, mais ce n'est pas recommandé pour la sécurité. Il est préferable d'utiliser des agents SSH., comme indiqué dans la doc d'administration distante.

On crée pour l'utilisateur backuppc une clée RSA avec la phrase "secrete" : "Backuppc, ça marche 24/24 !"

On utilise à la place de ssh un script `ssh-auto` qui charge **ssh-agent** avant de lancer **ssh**. On indique à BackupPC d'utiliser ce script à la place de **ssh** dans le parametre `SshPath` de `config.pl`. Attention ce script doit avoir des droits d'executions pour l'utilisateur backuppc.

`/home/backuppc/ssh-auto` :

```
#!/bin/sh
# BV 03/12/2003
# Appelle ssh apres avoir chargé l'agent ssh.
# Est appelé par BackupPC à la place de ssh.

. /home/backuppc/.ssh-agent-info
/usr/bin/ssh -q "$@"
```

Ressources

- La mailing liste de BackupPC sur SouceForge : http://backuppc.sf.net.
- Un exemple complet : http://www.jarunee.org/installation-serveur/t1.html.
- rsync : http://rsync.samba.org/.
- Concernant SSH : ma propre doc sur l'administration distante ne doit pas être loin.
- Les modules Perl avec CPAN : http://www.cpan.org/misc/cpan-faq.html#How_install_Perl_modules, http://theoryx5.uwinnipeg.ca/CPAN/perl/pod/perlfaq8/How_do_I_install_a_module_from_CPAN.html.
- Est-il besoin de le dire : http://httpd.apache.org/docs-2.0/.

Chapter 3. Configuration du serveur

Globale (config.pl)

Le fichier est assez long : 1300 lignes, mais bien fait. Une grosse partie peut être supprimé : ce qui concerne smb, tar, les users ..

- FullKeepCnt : 4; Nombre de sauvegardes completes à garder (1 par défaut.)
- BackupFilesExclude : ['/proc', '/temp', '/tmp', '/media', '/mnt']; (/dev peut être sauvegardé, il ne tiens pas beaucoup de place.)
- XFerMethod : rsync; méthode de transfert : SSH lance rsync sur le client.
- RsyncShareName : ['/'] ; Rsyncd permet visiblement de gérer les répertoires à sauvegarder sur le client.
- SshPath : `/home/backuppc/ssh-auto`; On utilise un script qui charge ssh-agent avant de lancer ssh.
- EMailNotifyMinDays : 0.5 ; pour commencer ....
- ClientTimeout : 14400; deux fois la valeur par défaut, permet d'éviter les messages "exiting after signal ALRM".
- EMailAdminUserName : Bruno.Vernay@padrol.fr ; pour commencer
- CgiAdminUserGroup ?;
- Langage : fr;
- CgiDateFormatMMDD : 0

Voir la section "Configuration file includes" dans la documentation pour configurer des groupes de machines (DMZ, clients MS -Windows, clients Linux ...) Le principe consiste à créer un fichier `/var/backuppc/conf/configMSWin.pl` qui regroupe les options communes aux hôtes MS-Windows. Ensuite, pour chaque hôte MS-Windows, on crée un fichier `/var/backuppc/pc/mon-hote-mswindows/config.pl` qui inclue le précédent et qui donne les options spécifiques :

```
do "/var/backuppc/conf/configMSWin.pl";
    #
    # Tell BackupPC which share to backup.  This should be the name
    # of the module from C:\rsyncd\rsyncd.conf on the client (the
    # name inside the square brackets).  In the sample rsynd.conf
    # file the cDrive module is the entire C drive.
    #
    $Conf{RsyncShareName}  = ['D_documents', 'D_Archives', 'G_Documents'];
```

Hôtes

Fichiers des hotes, très simple : une ligne par serveur à sauvegarder avec son IP ou son nom si on veut utiliser DHCP, ainsi que les personnes concernée par le serveur.

Interface Web

Voir l'installation. On peut aussi changer les couleurs ...mais je ne rentre pas dans ces détails.

Chapter 4. Configuration des clients

Rsync permettant de gagner beaucoup en traffic réseau, il est souvent préféré ainsi que SSH. Comme cela marche mieux avec Linux, certains utilisent un serveur Samba sur lequel les utilisateurs copient leurs fichiers, ce serveur étant sauvegardé par Rsync+SSH.

Par ailleur, la prochaine version permettra de reprendre une sauvegarde interrompue, si la méthode de transfert est rsync. (Actuellement 2.0.2, si une sauvegarde complete est interrompue, il faut la reprendre de zéro.)

Client Linux

Il suffit d'installer des versions récentes de SSH et rsync.

Il est possible de télécharger les sources de rsync et de le compiler si il n'existe pas de RPM assez récent. Il suffit de changer le préfix par défaut pour installer l'executable dans `/usr/bin` plutôt que `/usr/local/bin`. C'est ce qui a été fait sur une Mandrake 8.2 par exemple.

Client MS-Windows

Installation de rsyncd : Il faut preferer le package fourni sur le site de BackupPC. Il s'installe facilement en service. Attention, il utilise le port 873.

La configuration se fait dans un fichier où sont définis les répertoires à sauvegarder. Voir le fichier README, en fonction des besoins spécifiques.

Pour d'autres types d'installation, notamment SSH+Rsync, quelques ressources :

- http://lexa.mckenna.edu/sshwindows/ : Install d'OpenSSH simple et léger, tourne en service.
- http://pigtail.net/LRP/printsrv/cygwin-sshd.html : HowTo pour l'install de Cygwin + SSH.

Appendix A. /var/backuppc/conf/config.pl

```
#============================================================= -*-perl-*-
#
# Configuration file for BackupPC.
#
# DESCRIPTION
#
#   This is the main configuration file for BackupPC.
#
#   This file must be valid perl source, so make sure the punctuation,
#   quotes, and other syntax are valid.
#
#   This file is read by BackupPC at startup, when a HUP (-1) signal
#   is sent to BackupPC and also at each wakeup time whenever the
#   modification time of this file changes.
#
#   The configuration parameters are divided into four general groups.
#   The first group (general server configuration) provides general
#   configuration for BackupPC.  The next two groups describe what
#   to backup, when to do it, and how long to keep it.  The fourth
#   group are settings for the CGI http interface.
#
#   Configuration settings can also be specified on a per-PC basis.
#   Simply put the relevant settings in a config.pl file in the
#   PC's backup directory (ie: in __TOPDIR__/pc/hostName).
#   All configuration settings in the second, third and fourth
#   groups can be overridden by the per-PC config.pl file.
#
# AUTHOR
#   Craig Barratt  <cbarratt@users.sourceforge.net>
#
# COPYRIGHT
#   Copyright (C) 2001-2003  Craig Barratt
#
#   See http://backuppc.sourceforge.net.
#
#========================================================================

###########################################################################
# General server configuration
###########################################################################
#
# Host name on which the BackupPC server is running.
#
$Conf{ServerHost} = 'svbackup01.padrol.fr';

#
# TCP port number on which the BackupPC server listens for and accepts
# connections.  Normally this should be disabled (set to -1).  The TCP
# port is only needed if apache runs on a different machine from BackupPC.
# In that case, set this to any spare port number over 1024 (eg: 2359).
# If you enable the TCP port, make sure you set $Conf{ServerMesgSecret}
# too!
#
$Conf{ServerPort} = -1;

#
# Shared secret to make the TCP port secure.  Set this to a hard to guess
# string if you enable the TCP port (ie: $Conf{ServerPort} > 0).
#
# To avoid possible attacks via the TCP socket interface, every client
# message is protected by an MD5 digest. The MD5 digest includes four
# items:
#   - a seed that is sent to the client when the connection opens
#   - a sequence number that increments for each message
```

```
#    - a shared secret that is stored in $Conf{ServerMesgSecret}
#    - the message itself.
#
# The message is sent in plain text preceded by the MD5 digest.  A
# snooper can see the plain-text seed sent by BackupPC and plain-text
# message from the client, but cannot construct a valid MD5 digest since
# the secret $Conf{ServerMesgSecret} is unknown.  A replay attack is
# not possible since the seed changes on a per-connection and
# per-message basis.
#
$Conf{ServerMesgSecret} = '';

#
# PATH setting for BackupPC.  An explicit value is necessary
# for taint mode.  Value shouldn't matter too much since
# all execs use explicit paths.  However, taint mode in perl
# will complain if this directory is world writable.
#
$Conf{MyPath} = '/bin';

#
# Permission mask for directories and files created by BackupPC.
# Default value prevents any access from group other, and prevents
# group write.
#
$Conf{UmaskMode} = 027;

#
# Times at which we wake up, check all the PCs, and schedule necessary
# backups.  Times are measured in hours since midnight.  Can be
# fractional if necessary (eg: 4.25 means 4:15am).
#
# If the hosts you are backing up are always connected to the network
# you might have only one or two wakeups each night.  This will keep
# the backup activity after hours.  On the other hand, if you are backing
# up laptops that are only intermittently connected to the network you
# will want to have frequent wakeups (eg: hourly) to maximized the chance
# that each laptop is backed up.
#
# Examples:
#     $Conf{WakeupSchedule} = [22.5];         # once per day at 10:30 pm.
#     $Conf{WakeupSchedule} = [1..23];        # every hour except midnight
#     $Conf{WakeupSchedule} = [2,4,6,8,10,12,14,16,18,20,22];  # every 2 hours
#
# The default value is every hour except midnight.
#
$Conf{WakeupSchedule} = [1..23];

#
# Maximum number of simultaneous backups to run.  If there
# are no user backup requests then this is the maximum number
# of simultaneous backups.
#
$Conf{MaxBackups} = 4;

#
# Additional number of simultaneous backups that users can run.
# As many as $Conf{MaxBackups} + $Conf{MaxUserBackups} requests can
# run at the same time.
#
$Conf{MaxUserBackups} = 4;

#
# Maximum number of pending link commands. New backups will only be
# started if there are no more than $Conf{MaxPendingCmds} plus
# $Conf{MaxBackups} number of pending link commands, plus running jobs.
# This limit is to make sure BackupPC doesn't fall too far behind in
# running BackupPC_link commands.
```

```
#
$Conf{MaxPendingCmds} = 10;

#
# Maximum number of log files we keep around in log directory.
# These files are aged nightly.  A setting of 14 means the log
# directory will contain about 2 weeks of old log files, in
# particular at most the files LOG, LOG.0, LOG.1, ... LOG.13
# (except today's LOG, these files will have a .z extension if
# compression is on).
#
# If you decrease this number after BackupPC has been running for a
# while you will have to manually remove the older log files.
#
$Conf{MaxOldLogFiles} = 14;

#
# Full path to the df command.  Security caution: normal users
# should not allowed to write to this file or directory.
#
$Conf{DfPath} = '/bin/df';

#
# Command to run df.  The following variables are substituted at run-time:
#
#   $dfPath      path to df ($Conf{DfPath})
#   $topDir      top-level BackupPC data directory
#
$Conf{DfCmd} = '$dfPath $topDir';

#
# Maximum threshold for disk utilization on the __TOPDIR__ filesystem.
# If the output from $Conf{DfPath} reports a percentage larger than
# this number then no new regularly scheduled backups will be run.
# However, user requested backups (which are usually incremental and
# tend to be small) are still performed, independent of disk usage.
# Also, currently running backups will not be terminated when the disk
# usage exceeds this number.
#
$Conf{DfMaxUsagePct} = 95;

#
# How long BackupPC_trashClean sleeps in seconds between each check
# of the trash directory.  Once every 5 minutes should be reasonable.
#
$Conf{TrashCleanSleepSec} = 300;

#
# List of DHCP address ranges we search looking for PCs to backup.
# This is an array of hashes for each class C address range.
# This is only needed if hosts in the conf/hosts file have the
# dhcp flag set.
#
# Examples:
#    # to specify 192.10.10.20 to 192.10.10.250 as the DHCP address pool
#    $Conf{DHCPAddressRanges} = [
#        {
#            ipAddrBase => '192.10.10',
#            first => 20,
#            last  => 250,
#        },
#    ];
#    # to specify two pools (192.10.10.20-250 and 192.10.11.10-50)
#    $Conf{DHCPAddressRanges} = [
#        {
#            ipAddrBase => '192.10.10',
#            first => 20,
#            last  => 250,
```

```
#         },
#         {
#             ipAddrBase => '192.10.11',
#             first => 10,
#             last  => 50,
#         },
#     ];
#
$Conf{DHCPAddressRanges} = [];

#
# These configuration settings aren't used by BackupPC, but simply
# remember a few settings used by configure.pl during installation.
# These are used by configure.pl when upgrading to new versions of
# BackupPC.
#
$Conf{BackupPCUser} = 'backuppc';
$Conf{CgiDir}       = '/srv/www/cgi-bin/backuppc';
$Conf{InstallDir}   = '/usr/local/backuppc';

#
# Whether BackupPC and the CGI script BackupPC_Admin verify that they
# are really running as user $Conf{BackupPCUser}.  If this flag is set
# and the effective user id (euid) differs from $Conf{BackupPCUser}
# then both scripts exit with an error.  This catches cases where
# BackupPC might be accidently started as root or the wrong user,
# or if the CGI script is not installed correctly.
#
$Conf{BackupPCUserVerify} = 1;

#
# Maximum number of hardlinks supported by the $TopDir file system
# that BackupPC uses.  Most linux or unix file systems should support
# at least 32000 hardlinks per file, or 64K in other cases.  If a pool
# file already has this number of hardlinks, a new pool file is created
# so that new hardlinks can be accommodated.  This limit will only
# be hit if an identical file appears at least this number of times
# across all the backups.
#
$Conf{HardLinkMax} = 31999;

###########################################################################
# What to backup and when to do it
# (can be overridden in the per-PC config.pl)
###########################################################################
#
# Name of the host share that is backed up when using SMB.  This can be a
# string or an array of strings if there are multiple shares per host.
# Examples:
#
#   $Conf{SmbShareName} = 'c';          # backup 'c' share
#   $Conf{SmbShareName} = ['c', 'd'];   # backup 'c' and 'd' shares
#
# This setting only matters if $Conf{XferMethod} = 'smb'.
#
$Conf{SmbShareName} = 'C$';

#
# Smbclient share user name.  This is passed to smbclient's -U argument.
#
# This setting only matters if $Conf{XferMethod} = 'smb'.
#
$Conf{SmbShareUserName} = '';

#
# Smbclient share password.  This is passed to smbclient via its PASSWD
# environment variable.  There are several ways you can tell BackupPC
# the smb share password.  In each case you should be very careful about
```

```
# security.  If you put the password here, make sure that this file is
# not readable by regular users!  See the "Setting up config.pl" section
# in the documentation for more information.
#
# This setting only matters if $Conf{XferMethod} = 'smb'.
#
$Conf{SmbSharePasswd} = '';

#
# Which host directories to backup when using tar transport.  This can be a
# string or an array of strings if there are multiple directories to
# backup per host.  Examples:
#
#   $Conf{TarShareName} = '/';          # backup everything
#   $Conf{TarShareName} = '/home';      # only backup /home
#   $Conf{TarShareName} = ['/home', '/src']; # backup /home and /src
#
# The fact this parameter is called 'TarShareName' is for historical
# consistency with the Smb transport options.  You can use any valid
# directory on the client: there is no need for it to correspond to
# any Smb share or device mount point.
#
# Note also that you can also use $Conf{BackupFilesOnly} to specify
# a specific list of directories to backup.  It's more efficient to
# use this option instead of $Conf{TarShareName} since a new tar is
# run for each entry in $Conf{TarShareName}.
#
# On the other hand, if you add --one-file-system to $Conf{TarClientCmd}
# you can backup each file system separately, which makes restoring one
# bad file system easier.  In this case you would list all of the mount
# points here, since you can't get the same result with
# $Conf{BackupFilesOnly}:
#
#     $Conf{TarShareName} = ['/', '/var', '/data', '/boot'];
#
# This setting only matters if $Conf{XferMethod} = 'tar'.
#
$Conf{TarShareName} = '/';

#
# Minimum period in days between full backups. A full dump will only be
# done if at least this much time has elapsed since the last full dump,
# and at least $Conf{IncrPeriod} days has elapsed since the last
# successful dump.
#
# Typically this is set slightly less than an integer number of days. The
# time taken for the backup, plus the granularity of $Conf{WakeupSchedule}
# will make the actual backup interval a bit longer.
#
# There are two special values for $Conf{FullPeriod}:
#
#   -1   Don't do any regular backups on this machine.  Manually
#        requested backups (via the CGI interface) will still occur.
#
#   -2   Don't do any backups on this machine.  Manually requested
#        backups (via the CGI interface) will be ignored.
#
# These special settings are useful for a client that is no longer
# being backed up (eg: a retired machine), but you wish to keep the
# last backups available for browsing or restoring to other machines.
#
# Also, you might create a virtual client (by setting $Conf{ClientNameAlias})
# for restoring to a DVD or permanent media and you would set
# $Conf{FullPeriod} to -2 so that it is never backed up.
#
$Conf{FullPeriod} = 6.97;

#
```

```
# Minimum period in days between incremental backups (a user requested
# incremental backup will be done anytime on demand).
#
# Typically this is set slightly less than an integer number of days. The
# time taken for the backup, plus the granularity of $Conf{WakeupSchedule}
# will make the actual backup interval a bit longer.
#
$Conf{IncrPeriod} = 0.97;

#
# Number of full backups to keep.  Must be >= 1.
#
# In the steady state, each time a full backup completes successfully
# the oldest one is removed.  If this number is decreased, the
# extra old backups will be removed.
#
# If filling of incremental dumps is off the oldest backup always
# has to be a full (ie: filled) dump.  This might mean an extra full
# dump is kept until the second oldest (incremental) dump expires.
#
$Conf{FullKeepCnt} = 4;

#
# Very old full backups are removed after $Conf{FullAgeMax} days.  However,
# we keep at least $Conf{FullKeepCntMin} full backups no matter how old
# they are.
#
$Conf{FullKeepCntMin} = 1;
$Conf{FullAgeMax}     = 60;

#
# Number of incremental backups to keep.  Must be >= 1.
#
# In the steady state, each time an incr backup completes successfully
# the oldest one is removed.  If this number is decreased, the
# extra old backups will be removed.
#
$Conf{IncrKeepCnt} = 6;

#
# Very old incremental backups are removed after $Conf{IncrAgeMax} days.
# However, we keep at least $Conf{IncrKeepCntMin} incremental backups no
# matter how old they are.
#
$Conf{IncrKeepCntMin} = 1;
$Conf{IncrAgeMax}     = 30;

#
# Whether incremental backups are filled.  "Filling" means that the
# most recent full (or filled) dump is merged into the new incremental
# dump using hardlinks.  This makes an incremental dump look like a
# full dump.  Prior to v1.03 all incremental backups were filled.
# In v1.4.0 and later the default is off.
#
# BackupPC, and the cgi interface in particular, do the right thing on
# un-filled incremental backups.  It will correctly display the merged
# incremental backup with the most recent filled backup, giving the
# un-filled incremental backups a filled appearance.  That means it
# invisible to the user whether incremental dumps are filled or not.
#
# Filling backups takes a little extra disk space, and it does cost
# some extra disk activity for filling, and later removal.  Filling
# is no longer useful, since file mangling and compression doesn't
# make a filled backup very useful. It's likely the filling option
# will be removed from future versions: filling will be delegated to
# the display and extraction of backup data.
#
# If filling is off, BackupPC makes sure that the oldest backup is
```

```
# a full, otherwise the following incremental backups will be
# incomplete.  This might mean an extra full backup has to be
# kept until the following incremental backups expire.
#
# The default is off.  You can turn this on or off at any
# time without affecting existing backups.
#
$Conf{IncrFill} = 0;

#
# Number of restore logs to keep.  BackupPC remembers information about
# each restore request.  This number per client will be kept around before
# the oldest ones are pruned.
#
# Note: files/dirs delivered via Zip or Tar downloads don't count as
# restores.  Only the first restore option (where the files and dirs
# are written to the host) count as restores that are logged.
#
$Conf{RestoreInfoKeepCnt} = 10;

#
# List of directories or files to backup.  If this is defined, only these
# directories or files will be backed up.
#
# For Smb, only one of $Conf{BackupFilesExclude} and $Conf{BackupFilesOnly}
# can be specified per share. If both are set for a particular share, then
# $Conf{BackupFilesOnly} takes precedence and $Conf{BackupFilesExclude}
# is ignored.
#
# This can be set to a string, an array of strings, or, in the case
# of multiple shares, a hash of strings or arrays.  A hash is used
# to give a list of directories or files to backup for each share
# (the share name is the key).  If this is set to just a string or
# array, and $Conf{SmbShareName} contains multiple share names, then
# the setting is assumed to apply to only the first share name.
#
# Examples:
#    $Conf{BackupFilesOnly} = '/myFiles';
#    $Conf{BackupFilesOnly} = ['/myFiles'];     # same as first example
#    $Conf{BackupFilesOnly} = ['/myFiles', '/important'];
#    $Conf{BackupFilesOnly} = {
#       'c' => ['/myFiles', '/important'],      # these are for 'c' share
#       'd' => ['/moreFiles', '/archive'],      # these are for 'd' share
#    };
#
$Conf{BackupFilesOnly} = undef;

#
# List of directories or files to exclude from the backup.  For Smb,
# only one of $Conf{BackupFilesExclude} and $Conf{BackupFilesOnly}
# can be specified per share.  If both are set for a particular share,
# then $Conf{BackupFilesOnly} takes precedence and
# $Conf{BackupFilesExclude} is ignored.
#
# This can be set to a string, an array of strings, or, in the case
# of multiple shares, a hash of strings or arrays.  A hash is used
# to give a list of directories or files to exclude for each share
# (the share name is the key).  If this is set to just a string or
# array, and $Conf{SmbShareName} contains multiple share names, then
# the setting is assumed to apply to only the first share name.
#
# The exact behavior is determined by the underlying transport program,
# smbclient or tar.  For smbclient the exlclude file list is passed into
# the X option.  Simple shell wild-cards using "*" or "?" are allowed.
#
# For tar, if the exclude file contains a "/" it is assumed to be anchored
# at the start of the string.  Since all the tar paths start with "./",
# BackupPC prepends a "." if the exclude file starts with a "/".  Note
```

```
# that GNU tar version >= 1.13.7 is required for the exclude option to
# work correctly.  For linux or unix machines you should add
# "/proc" to $Conf{BackupFilesExclude} unless you have specified
# --one-file-system in $Conf{TarClientCmd} or --one-file-system in
# $Conf{RsyncArgs}.  Also, for tar, do not use a trailing "/" in
# the directory name: a trailing "/" causes the name to not match
# and the directory will not be excluded.
#
# Examples:
#    $Conf{BackupFilesExclude} = '/temp';
#    $Conf{BackupFilesExclude} = ['/temp'];     # same as first example
#    $Conf{BackupFilesExclude} = ['/temp', '/winnt/tmp'];
#    $Conf{BackupFilesExclude} = {
#       'c' => ['/temp', '/winnt/tmp'],         # these are for 'c' share
#       'd' => ['/junk', '/dont_back_this_up'], # these are for 'd' share
#    };
#
$Conf{BackupFilesExclude} = ['/proc', '/temp', '/tmp', '/media', '/mnt'];

#
# PCs that are always or often on the network can be backed up after
# hours, to reduce PC, network and server load during working hours. For
# each PC a count of consecutive good pings is maintained. Once a PC has
# at least $Conf{BlackoutGoodCnt} consecutive good pings it is subject
# to "blackout" and not backed up during hours and days specified by
# $Conf{BlackoutWeekDays}, $Conf{BlackoutHourBegin} and
# $Conf{BlackoutHourEnd}.
#
# To allow for periodic rebooting of a PC or other brief periods when a
# PC is not on the network, a number of consecutive bad pings is allowed
# before the good ping count is reset. This parameter is
# $Conf{BlackoutBadPingLimit}.
#
# Note that bad and good pings don't occur with the same interval. If a
# machine is always on the network, it will only be pinged roughly once
# every $Conf{IncrPeriod} (eg: once per day). So a setting for
# $Conf{BlackoutGoodCnt} of 7 means it will take around 7 days for a
# machine to be subject to blackout. On the other hand, if a ping is
# failed, it will be retried roughly every time BackupPC wakes up, eg,
# every one or two hours. So a setting for $Conf{BlackoutBadPingLimit} of
# 3 means that the PC will lose its blackout status after 3-6 hours of
# unavailability.
#
# To disable the blackout feature set $Conf{BlackoutGoodCnt} to a negative
# value.  A value of 0 will make all machines subject to blackout.  But
# if you don't want to do any backups during the day it would be easier
# to just set $Conf{WakeupSchedule} to a restricted schedule.
#
$Conf{BlackoutBadPingLimit} = 3;
$Conf{BlackoutGoodCnt}      = 7;

#
# The default settings specify the blackout period from 7:00am to
# 7:30pm local time on Mon-Fri.  For $Conf{BlackoutWeekDays},
# 0 is Sunday, 1 is Monday etc.
#
$Conf{BlackoutHourBegin}    = 7.0;
$Conf{BlackoutHourEnd}      = 19.5;
$Conf{BlackoutWeekDays}     = [1, 2, 3, 4, 5];

#
# A backup of a share that has zero files is considered fatal. This is
# used to catch miscellaneous Xfer errors that result in no files being
# backed up.  If you have shares that might be empty (and therefore an
# empty backup is valid) you should set this flag to 0.
#
$Conf{BackupZeroFilesIsFatal} = 1;
```

```
###########################################################################
# General per-PC configuration settings
# (can be overridden in the per-PC config.pl)
###########################################################################
#
# What transport method to use to backup each host.  If you have
# a mixed set of WinXX and linux/unix hosts you will need to override
# this in the per-PC config.pl.
#
# The valid values are:
#
#   - 'smb':    backup and restore via smbclient and the SMB protocol.
#               Best choice for WinXX.
#
#   - 'rsync':  backup and restore via rsync (via rsh or ssh).
#               Best choice for linux/unix.  Can also work on WinXX.
#
#   - 'rsyncd': backup and restre via rsync daemon on the client.
#               Best choice for linux/unix if you have rsyncd running on
#               the client.  Can also work on WinXX.
#
#   - 'tar':    backup and restore via tar, tar over ssh, rsh or nfs.
#               Good choice for linux/unix.
#
$Conf{XferMethod} = 'rsync';

#
# Full path for smbclient. Security caution: normal users should not
# allowed to write to this file or directory.
#
# smbclient is from the Samba distribution. smbclient is used to
# actually extract the incremental or full dump of the share filesystem
# from the PC.
#
# This setting only matters if $Conf{XferMethod} = 'smb'.
#
$Conf{SmbClientPath} = '/usr/bin/smbclient';

#
# Commands to run smbclient for a full dump, incremental dump or a restore.
# This setting only matters if $Conf{XferMethod} = 'smb'.
#
# The following variables are substituted at run-time:
#
#    $smbClientPath   same as $Conf{SmbClientPath}
#    $host            host to backup/restore
#    $hostIP          host IP address
#    $shareName       share name
#    $userName        user name
#    $fileList        list of files to backup (based on exclude/include)
#    $I_option        optional -I option to smbclient
#    $X_option        exclude option (if $fileList is an exclude list)
#    $timeStampFile   start time for incremental dump
#
$Conf{SmbClientFullCmd} = '$smbClientPath \\\\$host\\$shareName'
            . ' $I_option -U $userName -E -N -d 1'
            . ' -c tarmode\\ full -Tc$X_option - $fileList';

$Conf{SmbClientIncrCmd} = '$smbClientPath \\\\$host\\$shareName'
      . ' $I_option -U $userName -E -N -d 1'
      . ' -c tarmode\\ full -TcN$X_option $timeStampFile - $fileList';

$Conf{SmbClientRestoreCmd} = '$smbClientPath \\\\$host\\$shareName'
            . ' $I_option -U $userName -E -N -d 1'
            . ' -c tarmode\\ full -Tx -';

#
# Full command to run tar on the client.  GNU tar is required.  You will
```

```
# need to fill in the correct paths for ssh2 on the local host (server)
# and GNU tar on the client.  Security caution: normal users should not
# allowed to write to these executable files or directories.
#
# See the documentation for more information about setting up ssh2 keys.
#
# If you plan to use NFS then tar just runs locally and ssh2 is not needed.
# For example, assuming the client filesystem is mounted below /mnt/hostName,
# you could use something like:
#
#    $Conf{TarClientCmd} = '$tarPath -c -v -f - -C /mnt/$host/$shareName'
#                        . ' --totals';
#
# In the case of NFS or rsh you need to make sure BackupPC's privileges
# are sufficient to read all the files you want to backup.  Also, you
# will probably want to add "/proc" to $Conf{BackupFilesExclude}.
#
# The following variables are substituted at run-time:
#
#   $host        host name
#   $hostIP      host's IP address
#   $incrDate    newer-than date for incremental backups
#   $shareName   share name to backup (ie: top-level directory path)
#   $fileList    specific files to backup or exclude
#   $tarPath     same as $Conf{TarClientPath}
#   $sshPath     same as $Conf{SshPath}
#
# If a variable is followed by a "+" it is shell escaped.  This is
# necessary for the command part of ssh or rsh, since it ends up
# getting passed through the shell.
#
# This setting only matters if $Conf{XferMethod} = 'tar'.
#
$Conf{TarClientCmd} = '$sshPath -q -n -l root $host'
                    . ' $tarPath -c -v -f - -C $shareName+'
                    . ' --totals';

#
# Extra tar arguments for full backups.  Several variables are substituted at
# run-time.  See $Conf{TarClientCmd} for the list of variable substitutions.
#
# If you are running tar locally (ie: without rsh or ssh) then remove the
# "+" so that the argument is no longer shell escaped.
#
# This setting only matters if $Conf{XferMethod} = 'tar'.
#
$Conf{TarFullArgs} = '$fileList+';

#
# Extra tar arguments for incr backups.  Several variables are substituted at
# run-time.  See $Conf{TarClientCmd} for the list of variable substitutions.
#
# Note that GNU tar has several methods for specifying incremental backups,
# including:
#
#   --newer-mtime $incrDate+
#          This causes a file to be included if the modification time is
#          later than $incrDate (meaning its contents might have changed).
#          But changes in the ownership or modes will not qualify the
#          file to be included in an incremental.
#
#   --newer=$incrDate+
#          This causes the file to be included if any attribute of the
#          file is later than $incrDate, meaning either attributes or
#          the modification time.  This is the default method.  Do
#          not use --atime-preserve in $Conf{TarClientCmd} above,
#          otherwise resetting the atime (access time) counts as an
#          attribute change, meaning the file will always be included
```

```
#          in each new incremental dump.
#
# If you are running tar locally (ie: without rsh or ssh) then remove the
# "+" so that the argument is no longer shell escaped.
#
# This setting only matters if $Conf{XferMethod} = 'tar'.
#
$Conf{TarIncrArgs} = '--newer=$incrDate+ $fileList+';

#
# Full command to run tar for restore on the client.  GNU tar is required.
# This can be the same as $Conf{TarClientCmd}, with tar's -c replaced by -x
# and ssh's -n removed.
#
# See $Conf{TarClientCmd} for full details.
#
# This setting only matters if $Conf{XferMethod} = "tar".
#
$Conf{TarClientRestoreCmd} = '$sshPath -q -l root $host'
     . ' $tarPath -x -p --numeric-owner --same-owner'
     . ' -v -f - -C $shareName+';

#
# Full path for tar on the client. Security caution: normal users should not
# allowed to write to this file or directory.
#
# This setting only matters if $Conf{XferMethod} = 'tar'.
#
$Conf{TarClientPath} = '/bin/gtar';

#
# Path to rsync executable on the client
#
$Conf{RsyncClientPath} = '/usr/bin/rsync';

#
# Full command to run rsync on the client machine.  The following variables
# are substituted at run-time:
#
#        $host           host name being backed up
#        $hostIP         host's IP address
#        $shareName      share name to backup (ie: top-level directory path)
#        $rsyncPath      same as $Conf{RsyncClientPath}
#        $sshPath        same as $Conf{SshPath}
#        $argList        argument list, built from $Conf{RsyncArgs},
#                        $shareName, $Conf{BackupFilesExclude} and
#                        $Conf{BackupFilesOnly}
#
# This setting only matters if $Conf{XferMethod} = 'rsync'.
#
$Conf{RsyncClientCmd} = '$sshPath -l root $host $rsyncPath $argList+';

#
# Full command to run rsync for restore on the client.  The following
# variables are substituted at run-time:
#
#        $host           host name being backed up
#        $hostIP         host's IP address
#        $shareName      share name to backup (ie: top-level directory path)
#        $rsyncPath      same as $Conf{RsyncClientPath}
#        $sshPath        same as $Conf{SshPath}
#        $argList        argument list, built from $Conf{RsyncArgs},
#                        $shareName, $Conf{BackupFilesExclude} and
#                        $Conf{BackupFilesOnly}
#
# This setting only matters if $Conf{XferMethod} = 'rsync'.
#
$Conf{RsyncClientRestoreCmd} = '$sshPath -l root $host $rsyncPath $argList+';
```

```
#
# Share name to backup.  For $Conf{XferMethod} = "rsync" this should
# be a file system path, eg '/' or '/home'.
#
# For $Conf{XferMethod} = "rsyncd" this should be the name of the module
# to backup (ie: the name from /etc/rsynd.conf).
#
# This can also be a list of multiple file system paths or modules.
# For example, by adding --one-file-system to $Conf{RsyncArgs} you
# can backup each file system separately, which makes restoring one
# bad file system easier.  In this case you would list all of the mount
# points:
#
#     $Conf{RsyncShareName} = ['/', '/var', '/data', '/boot'];
#
$Conf{RsyncShareName} = ['/'];

#
# Rsync daemon port on the client, for $Conf{XferMethod} = "rsyncd".
#
$Conf{RsyncdClientPort} = 873;

#
# Rsync daemon user name on client, for $Conf{XferMethod} = "rsyncd".
# The user name and password are stored on the client in whatever file
# the "secrets file" parameter in rsyncd.conf points to
# (eg: /etc/rsyncd.secrets).
#
$Conf{RsyncdUserName} = '';

#
# Rsync daemon user name on client, for $Conf{XferMethod} = "rsyncd".
# The user name and password are stored on the client in whatever file
# the "secrets file" parameter in rsyncd.conf points to
# (eg: /etc/rsyncd.secrets).
#
$Conf{RsyncdPasswd} = '';

#
# Whether authentication is mandatory when connecting to the client's
# rsyncd.  By default this is on, ensuring that BackupPC will refuse to
# connect to an rsyncd on the client that is not password protected.
# Turn off at your own risk.
#
$Conf{RsyncdAuthRequired} = 1;

#
# Arguments to rsync for backup.  Do not edit the first set unless you
# have a thorough understanding of how File::RsyncP works.
#
# Examples of additional arguments that should work are --exclude/--include,
# eg:
#
#     $Conf{RsyncArgs} = [
#           # original arguments here
#           '-v',
#           '--exclude', '/proc',
#           '--exclude', '*.tmp',
#     ];
#
$Conf{RsyncArgs} = [
     #
     # Do not edit these!
     #
            '--numeric-ids',
            '--perms',
            '--owner',
```

```
            '--group',
            '--devices',
            '--links',
            '--times',
            '--block-size=2048',
            '--recursive',
     #
     # Add additional arguments here
     #
            '--exclude', '*.tmp',
            '--exclude', '*.*~',
];

#
# Arguments to rsync for restore.  Do not edit the first set unless you
# have a thorough understanding of how File::RsyncP works.
#
#
$Conf{RsyncRestoreArgs} = [
     #
     # Do not edit these!
     #
     "--numeric-ids",
     "--perms",
     "--owner",
     "--group",
     "--devices",
     "--links",
     "--times",
     "--block-size=2048",
     "--relative",
     "--ignore-times",
     "--recursive",
     #
     # Add additional arguments here
     #
];

#
# Amount of verbosity in Rsync Xfer log files.  0 means be quiet,
# 1 will give will give one line per file, 2 will also show skipped
# files on incrementals, higher values give more output.  10 will
# include byte dumps of all data read/written, which will make the
# log files huge.
#
$Conf{RsyncLogLevel} = 1;

#
# Full path for ssh. Security caution: normal users should not
# allowed to write to this file or directory.
#
#$Conf{SshPath} = '/usr/bin/ssh';
$Conf{SshPath} = '/home/backuppc/ssh-auto';

#
# Full path for nmblookup. Security caution: normal users should not
# allowed to write to this file or directory.
#
# nmblookup is from the Samba distribution. nmblookup is used to get the
# netbios name, necessary for DHCP hosts.
#
$Conf{NmbLookupPath} = '/usr/bin/nmblookup';

#
# NmbLookup command.  Given an IP address, does an nmblookup on that
# IP address.  The following variables are substituted at run-time:
#
#   $nmbLookupPath      path to nmblookup ($Conf{NmbLookupPath})
```

```
#   $host               IP address
#
# This command is only used for DHCP hosts: given an IP address, this
# command should try to find its NetBios name.
#
$Conf{NmbLookupCmd} = '$nmbLookupPath -A $host';

#
# NmbLookup command.  Given a netbios name, finds that host by doing
# a NetBios lookup.  Several variables are substituted at run-time:
#
#   $nmbLookupPath      path to nmblookup ($Conf{NmbLookupPath})
#   $host               NetBios name
#
# In some cases you might need to change the broadcast address, for
# example if nmblookup uses 192.168.255.255 by default and you find
# that doesn't work, try 192.168.1.255 (or your equivalent class C
# address) using the -B option:
#
#    $Conf{NmbLookupFindHostCmd} = '$nmbLookupPath -B 192.168.1.255 $host';
#
# If you use a WINS server and your machines don't respond to
# multicast NetBios requests you can use this (replace 1.2.3.4
# with the IP address of your WINS server):
#
#    $Conf{NmbLookupFindHostCmd} = '$nmbLookupPath -R -U 1.2.3.4 $host';
#
# This is preferred over multicast since it minimizes network traffic.
#
# Experiment manually for your site to see what form of nmblookup command
# works.
#
$Conf{NmbLookupFindHostCmd} = '$nmbLookupPath $host';

#
# For fixed IP address hosts, BackupPC_dump can also verify the netbios
# name to ensure it matches the host name.  An error is generated if
# they do not match.  Typically this flag is off.  But if you are going
# to transition a bunch of machines from fixed host addresses to DHCP,
# setting this flag is a great way to verify that the machines have
# their netbios name set correctly before turning on DCHP.
#
$Conf{FixedIPNetBiosNameCheck} = 0;

#
# Full path to the ping command.  Security caution: normal users
# should not be allowed to write to this file or directory.
#
# If you want to disable ping checking, set this to some program
# that exits with 0 status, eg:
#
#     $Conf{PingPath} = '/bin/echo';
#
$Conf{PingPath} = '/bin/ping';

#
# Ping command.  The following variables are substituted at run-time:
#
#   $pingPath      path to ping ($Conf{PingPath})
#   $host          host name
#
# Wade Brown reports that on solaris 2.6 and 2.7 ping -s returns the wrong
# exit status (0 even on failure).  Replace with "ping $host 1", which
# gets the correct exit status but we don't get the round-trip time.
#
$Conf{PingCmd} = '$pingPath -c 1 -w 3 $host';

#
```

```
# Compression level to use on files.  0 means no compression.  Compression
# levels can be from 1 (least cpu time, slightly worse compression) to
# 9 (most cpu time, slightly better compression).  The recommended value
# is 3.  Changing to 5, for example, will take maybe 20% more cpu time
# and will get another 2-3% additional compression. See the zlib
# documentation for more information about compression levels.
#
# Changing compression on or off after backups have already been done
# will require both compressed and uncompressed pool files to be stored.
# This will increase the pool storage requirements, at least until all
# the old backups expire and are deleted.
#
# It is ok to change the compression value (from one non-zero value to
# another non-zero value) after dumps are already done.  Since BackupPC
# matches pool files by comparing the uncompressed versions, it will still
# correctly match new incoming files against existing pool files.  The
# new compression level will take effect only for new files that are
# newly compressed and added to the pool.
#
# If compression was off and you are enabling compression for the first
# time you can use the BackupPC_compressPool utility to compress the
# pool.  This avoids having the pool grow to accommodate both compressed
# and uncompressed backups.  See the documentation for more information.
#
# Note: compression needs the Compress::Zlib perl library.  If the
# Compress::Zlib library can't be found then $Conf{CompressLevel} is
# forced to 0 (compression off).
#
$Conf{CompressLevel} = 3;

#
# Maximum round-trip ping time in milliseconds.  This threshold is set
# to avoid backing up PCs that are remotely connected through WAN or
# dialup connections.  The output from ping -s (assuming it is supported
# on your system) is used to check the round-trip packet time.  On your
# local LAN round-trip times should be much less than 20msec.  On most
# WAN or dialup connections the round-trip time will be typically more
# than 20msec.  Tune if necessary.
#
$Conf{PingMaxMsec} = 20;

#
# Timeout in seconds when listening for the transport program's
# (smbclient, tar etc) stdout. If no output is received during this
# time, then it is assumed that something has wedged during a backup,
# and the backup is terminated.
#
# Note that stdout buffering combined with huge files being backed up
# could cause longish delays in the output from smbclient that
# BackupPC_dump sees, so in rare cases you might want to increase
# this value.
#
# Despite the name, this parameter sets the timeout for all transport
# methods (tar, smb etc).
#
$Conf{ClientTimeout} = 14400;

#
# Maximum number of log files we keep around in each PC's directory
# (ie: pc/$host).  These files are aged monthly.  A setting of 12
# means there will be at most the files LOG, LOG.0, LOG.1, ... LOG.11
# in the pc/$host directory (ie: about a years worth).  (Except this
# month's LOG, these files will have a .z extension if compression
# is on).
#
# If you decrease this number after BackupPC has been running for a
# while you will have to manually remove the older log files.
#
```

```
$Conf{MaxOldPerPCLogFiles} = 12;

#
# Optional commands to run before and after dumps and restores.
# Stdout from these commands will be written to the Xfer (or Restore)
# log file.  One example of using these commands would be to
# shut down and restart a database server, or to dump a database
# to files for backup.  Example:
#
#    $Conf{DumpPreUserCmd} = '$sshPath -l root $host /usr/bin/dumpMysql';
#
# The following variable substitutions are made at run time for
# $Conf{DumpPreUserCmd} and $Conf{DumpPostUserCmd}:
#
#        $type         type of dump (incr or full)
#        $xferOK       1 if the dump succeeded, 0 if it didn't
#        $client       client name being backed up
#        $host         host name (could be different from client name if
#                                 $Conf{ClientNameAlias} is set)
#        $hostIP       IP address of host
#        $user         user name from the hosts file
#        $moreUsers    list of additional users from the hosts file
#        $share        the first share name
#        $shares       list of all the share names
#        $XferMethod   value of $Conf{XferMethod} (eg: tar, rsync, smb)
#        $sshPath      value of $Conf{SshPath},
#
# The following variable substitutions are made at run time for
# $Conf{RestorePreUserCmd} and $Conf{RestorePostUserCmd}:
#
#        $client       client name being backed up
#        $xferOK       1 if the restore succeeded, 0 if it didn't
#        $host         host name (could be different from client name if
#                                 $Conf{ClientNameAlias} is set)
#        $hostIP       IP address of host
#        $user         user name from the hosts file
#        $moreUsers    list of additional users from the hosts file
#        $share        the first share name
#        $XferMethod   value of $Conf{XferMethod} (eg: tar, rsync, smb)
#        $sshPath      value of $Conf{SshPath},
#        $type         set to "restore"
#        $bkupSrcHost  host name of the restore source
#        $bkupSrcShare share name of the restore source
#        $bkupSrcNum   backup number of the restore source
#        $pathHdrSrc   common starting path of restore source
#        $pathHdrDest  common starting path of destination
#        $fileList     list of files being restored
#
$Conf{DumpPreUserCmd}     = undef;
$Conf{DumpPostUserCmd}    = undef;
$Conf{RestorePreUserCmd}  = undef;
$Conf{RestorePostUserCmd} = undef;

#
# Override the client's host name.  This allows multiple clients
# to all refer to the same physical host.  This should only be
# set in the per-PC config file and is only used by BackupPC at
# the last moment prior to generating the command used to backup
# that machine (ie: the value of $Conf{ClientNameAlias} is invisible
# everywhere else in BackupPC).  The setting can be a host name or
# IP address, eg:
#
#         $Conf{ClientNameAlias} = 'realHostName';
#         $Conf{ClientNameAlias} = '192.1.1.15';
#
# will cause the relevant smb/tar/rsync backup/restore commands to be
# directed to realHostName, not the client name.
#
```

```
# Note: this setting doesn't work for hosts with DHCP set to 1.
#
$Conf{ClientNameAlias} = undef;

#
# Advanced option for asking BackupPC to load additional perl modules.
# Can be a list (array ref) of module names to load at startup.
#
$Conf{PerlModuleLoad}     = undef;

###########################################################################
# Email reminders, status and messages
# (can be overridden in the per-PC config.pl)
###########################################################################
#
# Full path to the sendmail command.  Security caution: normal users
# should not allowed to write to this file or directory.
#
$Conf{SendmailPath} = '/usr/sbin/sendmail';

#
# Minimum period between consecutive emails to a single user.
# This tries to keep annoying email to users to a reasonable
# level.  Email checks are done nightly, so this number is effectively
# rounded up (ie: 2.5 means a user will never receive email more
# than once every 3 days).
#
$Conf{EMailNotifyMinDays} = 0.5;

#
# Name to use as the "from" name for email.  Depending upon your mail
# handler this is either a plain name (eg: "admin") or a fully-qualified
# name (eg: "admin@mydomain.com").
#
$Conf{EMailFromUserName} = 'backuppc';

#
# Destination address to an administrative user who will receive a
# nightly email with warnings and errors.  If there are no warnings
# or errors then no email will be sent.  Depending upon your mail
# handler this is either a plain name (eg: "admin") or a fully-qualified
# name (eg: "admin@mydomain.com").
#
$Conf{EMailAdminUserName} = 'bruno.vernay@padrol.fr';

#
# Destination domain name for email sent to users.  By default
# this is empty, meaning email is sent to plain, unqualified
# addresses.  Otherwise, set it to the destintation domain, eg:
#
#    $Cong{EMailUserDestDomain} = '@mydomain.com';
#
# With this setting user email will be set to 'user@mydomain.com'.
#
$Conf{EMailUserDestDomain} = '';

#
# This subject and message is sent to a user if their PC has never been
# backed up.
#
# These values are language-dependent.  The default versions can be
# found in the language file (eg: lib/BackupPC/Lang/en.pm).  If you
# need to change the message, copy it here and edit it, eg:
#
#   $Conf{EMailNoBackupEverMesg} = <<'EOF';
#   To: $user$domain
#   cc:
#   Subject: $subj
```

```
#
#   Dear $userName,
#
#   This is a site-specific email message.
#   EOF
#
$Conf{EMailNoBackupEverSubj} = undef;
$Conf{EMailNoBackupEverMesg} = undef;

#
# How old the most recent backup has to be before notifying user.
# When there have been no backups in this number of days the user
# is sent an email.
#
$Conf{EMailNotifyOldBackupDays} = 7.0;

#
# This subject and message is sent to a user if their PC has not recently
# been backed up (ie: more than $Conf{EMailNotifyOldBackupDays} days ago).
#
# These values are language-dependent.  The default versions can be
# found in the language file (eg: lib/BackupPC/Lang/en.pm).  If you
# need to change the message, copy it here and edit it, eg:
#
#   $Conf{EMailNoBackupRecentMesg} = <<'EOF';
#   To: $user$domain
#   cc:
#   Subject: $subj
#
#   Dear $userName,
#
#   This is a site-specific email message.
#   EOF
#
$Conf{EMailNoBackupRecentSubj} = undef;
$Conf{EMailNoBackupRecentMesg} = undef;

#
# How old the most recent backup of Outlook files has to be before
# notifying user.
#
$Conf{EMailNotifyOldOutlookDays} = 5.0;

#
# This subject and message is sent to a user if their Outlook files have
# not recently been backed up (ie: more than $Conf{EMailNotifyOldOutlookDays}
# days ago).
#
# These values are language-dependent.  The default versions can be
# found in the language file (eg: lib/BackupPC/Lang/en.pm).  If you
# need to change the message, copy it here and edit it, eg:
#
#   $Conf{EMailOutlookBackupMesg} = <<'EOF';
#   To: $user$domain
#   cc:
#   Subject: $subj
#
#   Dear $userName,
#
#   This is a site-specific email message.
#   EOF
#
$Conf{EMailOutlookBackupSubj} = undef;
$Conf{EMailOutlookBackupMesg} = undef;

###########################################################################
# CGI user interface configuration settings
# (can be overridden in the per-PC config.pl)
```

```
###########################################################################
#
# Normal users can only access information specific to their host.
# They can start/stop/browse/restore backups.
#
# Administrative users have full access to all hosts, plus overall
# status and log information.
#
# The administrative users are the union of the unix/linux group
# $Conf{CgiAdminUserGroup} and the manual list of users, separated
# by spaces, in $Conf{CgiAdminUsers}. If you don't want a group or
# manual list of users set the corresponding configuration setting
# to undef or an empty string.
#
# If you want every user to have admin privileges (careful!), set
# $Conf{CgiAdminUsers} = '*'.
#
# Examples:
#    $Conf{CgiAdminUserGroup} = 'admin';
#    $Conf{CgiAdminUsers}     = 'craig celia';
#    --> administrative users are the union of group admin, plus
#      craig and celia.
#
#    $Conf{CgiAdminUserGroup} = '';
#    $Conf{CgiAdminUsers}     = 'craig celia';
#    --> administrative users are only craig and celia'.
#
$Conf{CgiAdminUserGroup} = '';
$Conf{CgiAdminUsers}     = '*';

#
# URL of the BackupPC_Admin CGI script.  Used for email messages.
#
$Conf{CgiURL} = 'http://svbackup01.padrol.fr/cgi-bin/backuppc/BackupPC_Admin';

#
# Language to use.  See lib/BackupPC/Lang for the list of supported
# languages, which include English (en), French (fr), Spanish (es),
# and German (de).
#
# Currently the Language setting applies to the CGI interface and email
# messages sent to users.  Log files and other text is still in English.
#
$Conf{Language} = 'fr';

#
# User names that are rendered by the CGI interface can be turned
# into links into their home page or other information about the
# user.  To set this up you need to create two sprintf() strings,
# that each contain a single '%s' that will be replaced by the user
# name.  The default is a mailto: link.
#
# $Conf{CgiUserHomePageCheck} should be an absolute file path that
# is used to check (via "-f") that the user has a valid home page.
# Set this to undef or an empty string to turn off this check.
#
# $Conf{CgiUserUrlCreate} should be a full URL that points to the
# user's home page.  Set this to undef or an empty string to turn
# off generation of URLs for user names.
#
# Example:
#    $Conf{CgiUserHomePageCheck} = '/var/www/html/users/%s.html';
#    $Conf{CgiUserUrlCreate}     = 'http://myhost/users/%s.html';
#    --> if /var/www/html/users/craig.html exists, then 'craig' will
#      be rendered as a link to http://myhost/users/craig.html.
#
$Conf{CgiUserHomePageCheck} = '';
$Conf{CgiUserUrlCreate}     = 'mailto:%s';
```

```
#
# Date display format for CGI interface.  True for US-style dates (MM/DD)
# and zero for international dates (DD/MM).
#
$Conf{CgiDateFormatMMDD} = 0;

#
# If set, the complete list of hosts appears in the left navigation
# bar for administrators.  Otherwise, just the hosts for which the
# user is listed in the host file (as either the user or in moreUsers)
# are displayed.
#
$Conf{CgiNavBarAdminAllHosts} = 0;

#
# Header font and size for CGI interface
#
$Conf{CgiHeaderFontType} = 'arial';
$Conf{CgiHeaderFontSize} = '3';

#
# Color scheme for CGI interface.  Default values give a very light blue
# for the background navigation color, green for the header background,
# and white for the body background.  (You call tell I should stick to
# programming and not graphical design.)
#
$Conf{CgiNavBarBgColor} = '#ddeeee';
$Conf{CgiHeaderBgColor} = '#99cc33';
$Conf{CgiBodyBgColor}   = '#ffffff';

#
# Hilight colors based on status that are used in the PC summary page.
#
$Conf{CgiStatusHilightColor} = {
    Reason_backup_failed           => '#ffcccc',
    Reason_backup_done             => '#ccffcc',
    Reason_no_ping                 => '#ffff99',
    Reason_backup_canceled_by_user => '#ff9900',
    Status_backup_in_progress      => '#66cc99',
};

#
# Additional CGI header text.
#
$Conf{CgiHeaders} = '<meta http-equiv="pragma" content="no-cache">';

#
# Directory where images are stored.  This directory should be below
# Apache's DocumentRoot.  This value isn't used by BackupPC but is
# used by configure.pl when you upgrade BackupPC.
#
# Example:
#     $Conf{CgiImageDir} = '/usr/local/apache/htdocs/BackupPC';
#
$Conf{CgiImageDir} = '/srv/www/html/backuppc';

#
# Additional mappings of file name extenions to Content-Type for
# individual file restore.  See $Ext2ContentType in BackupPC_Admin
# for the default setting.  You can add additional settings here,
# or override any default settings.  Example:
#
#     $Conf{CgiExt2ContentType} = {
#                 'pl'  => 'text/plain',
#          };
#
$Conf{CgiExt2ContentType} = { };
```

```
#
# URL (without the leading http://host) for BackupPC's image directory.
# The CGI script uses this value to serve up image files.
#
# Example:
#     $Conf{CgiImageDirURL} = '/BackupPC';
#
$Conf{CgiImageDirURL} = '/backuppc';
```

Appendix B. /var/backuppc/conf/hosts

```
#============================================================= -*-perl-*-
#
# Host file list for BackupPC.
#
# DESCRIPTION
#
#   This file lists all the hosts that should be backed up by
#   BackupPC.
#
#   Each line in the hosts file contains three fields, separated
#   by white space:
#
#     - The host name.  If this host is a static IP address this
#       must the machine's IP host name (ie: something that can
#       be looked up using nslookup or DNS).  If this is a DHCP
#       host then the host name must be the netbios name of the
#       machine.  It is possible to have a host name that contains
#       spaces, but that is discouraged.  Escape a space with "\", eg:
#
#                craigs\ pc
#
#     - DHCP flag.  Set to 0 if this is a static IP address host
#       or if the machine can be found using nmblookup.  Otherwise,
#       if the client can only be found by looking through the DHCP
#       pool then set this to 1.
#
#     - User name (unix login/email name) of the user who "owns"
#       or uses this machine.  This is the user who will be sent
#       email about this machine, and this user will have permission
#       to stop/start/browse/restore backups for this host.  This
#       user name must match the name the user authenticates with
#       via apache.
#
#     - Optional additional user names (comma separated, no white space) of
#       users who are also allowed to stop/start/browse/restore backups
#       for this client via the CGI interface.  These users are not sent
#       email.  These do not need to be valid email names; they simply
#       need to match the name the user authenticates with via apache.
#
# AUTHOR
#   Craig Barratt  <craig@arraycomm.com>
#
# COPYRIGHT
#   Copyright (C) 2001  Craig Barratt
#
#   See http://backuppc.sourceforge.net.
#
#========================================================================

#
# The first non-comment non-empty line gives the field names and should
# not be edited!!
#
host        dhcp    user    moreUsers     # <--- do not edit this line
#farside    0       craig   jill,jeff     # <--- example static IP host entry
#larson     1       bill                  # <--- example DHCP host entry
192.168.1.114 0      bruno   bruno.vernay  # SvDevLinux01
194.206.56.16 0      bruno
```

Appendix C. /etc/init.d/apache2-backuppc

```
#!/bin/sh
...
# /etc/init.d/apache2-backuppc
#
### BEGIN INIT INFO
# Provides:                     apache2-backuppc
# Required-Start:               $local_fs $remote_fs $network
# X-UnitedLinux-Should-Start:   $named $time postgresql sendmail mysql ypclient dhcp radiusd
# Required-Stop:                $local_fs $remote_fs $network
# X-UnitedLinux-Should-Stop:
# Default-Start:                3 5
# Default-Stop:                 0 1 2 6
# Short-Description:            Apache2 backuppc
# Description:                  Start the httpd daemon Apache 2 for BackupPC
### END INIT INFO
...
# ligne 49 httpd_conf=${APACHE_HTTPD_CONF:-$sysconfdir/httpd.conf}
httpd_conf=$sysconfdir/httpd-backuppc.conf
...
```

Appendix D. /etc/apache2/httpd-backuppc.conf

Le fichier est consitué d'une série d'includes. On modifie `uid.conf` en `uid-backuppc.conf`. `default-server.conf` en `backuppc-server.conf`.

Le plus important est `/etc/apache2/backuppc-server.conf` :

```
#
# Configuration of the backuppc server, which answers client requests that are
# not picked up by any virtual host.
#

DocumentRoot "/srv/www/html"

#
# Configure the DocumentRoot
#
<Directory "/srv/www/html">
        # Possible values for the Options directive are "None", "All",
        # or any combination of:
        #   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
        #
        # Note that "MultiViews" must be named *explicitly* --- "Options All"
        # doesn't give it to you.
        #
        # The Options directive is both complicated and important.  Please see
        # http://httpd.apache.org/docs-2.0/mod/core.html#options
        # for more information.
        Options None
        # AllowOverride controls what directives may be placed in .htaccess files.
        # It can be "All", "None", or any combination of the keywords:
        #   Options FileInfo AuthConfig Limit
        AllowOverride None
        # Controls who can get stuff from this server.
        Order allow,deny
        Allow from all
</Directory>

# Aliases: aliases can be added as needed (with no limit). The format is
# Alias fakename realname
#
# Note that if you include a trailing / on fakename then the server will
# require it to be present in the URL.  So "/icons" isn't aliased in this
# example, only "/icons/".  If the fakename is slash-terminated, then the
# realname must also be slash terminated, and if the fakename omits the
# trailing slash, the realname must also omit it.
#
# We include the /icons/ alias for FancyIndexed directory listings.  If you
# do not use FancyIndexing, you may comment this out.
#
Alias /icons/ "/usr/share/apache2/icons/"

<Directory "/usr/share/apache2/icons">
        Options Indexes MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
</Directory>


# ScriptAlias: This controls which directories contain server scripts.
# ScriptAliases are essentially the same as Aliases, except that
# documents in the realname directory are treated as applications and
# run by the server when requested rather than as documents sent to the client.
# The same rules about trailing "/" apply to ScriptAlias directives as to
# Alias.
#
```

```
ScriptAlias /cgi-bin/ "/srv/www/cgi-bin/"

# "/srv/www/cgi-bin" should be changed to whatever your ScriptAliased
# CGI directory exists, if you have that configured.
#
<Directory "/srv/www/cgi-bin">
        AllowOverride None
        Options +ExecCGI -Includes
        Order allow,deny
        Allow from all
</Directory>

<Directory /srv/www/cgi-bin/backuppc>

    AuthName "BackupPC Admin"
    AuthType Basic
    # Ce compte n'est utilisé que pour l'authentification
    AuthLDAPBindDN "uid=apane,ou=auth-only,o=padrol"
    AuthLDAPBindPassword "Why pwd shouldn't be put in conf file !"
    AuthLDAPURL "ldap://ldap.padrol.fr/ou=annuaire,o=padrol?uid?sub?(objectClass=*)"
    Require valid-user
</Directory>
```