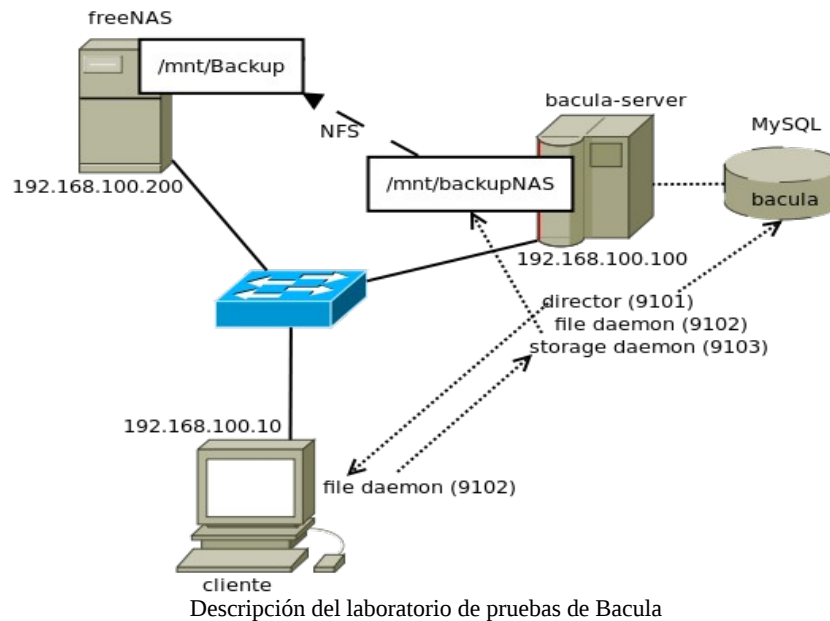


Backups con Bacula.

Recetario paso a paso para instalar y configurar Bacula en Ubuntu 10.04.3 LTS



Bacula es una colección de herramientas de respaldo, capaces de cubrir las necesidades de respaldo de equipos bajo redes IP. Se basa en una arquitectura cliente-servidor que resulta eficaz y fácil de manejar, dada la amplia gama de funciones y características que brinda; copiar y restaurar ficheros dañados o perdidos. Además, debido a su desarrollo y estructura modular, Bacula se adapta tanto al uso personal como profesional, para parques de ordenadores muy grandes. (Tomado de Wikipedia)

Contenidos

1. Instalar Bacula
2. Instalando freeNAS
3. Crear un volumen para escribir sobre él
4. Instalar y configurar un cliente
5. Comprobar el estado del cliente
6. Definir trabajos de copia en Bacula
7. Lanzar un trabajo de copia y restauración manual
8. Notas finales

PARTE 1. Instalar Bacula

En los repositorios de Ubuntu se incluye Bacula. Lo primero que vamos a hacer es instalar un servidor MySQL, y a crear una base de datos para Bacula, a la que llamaremos Bacula. Una vez creada la base de datos, daremos permisos totales sobre ella a un usuario exclusivo para el servicio. Podemos llamar a dicho usuario “baculaSQL”. Este usuario no tendrá ni siquiera una shell.

Partimos de la idea de que Ubuntu 10.04.3 LTS acaba de ser instalado.

```
$ sudo apt-get update
$ sudo apt-get install mysql-server
    # En este momento tendremos que asignar una contraseña al usuario root de MySQL.
    # Finalmente, el asistente terminará, y MySQL estará instalado.

    # Accedemos a MySQL como administradores de la base de datos.
$ mysql -u root -p
    # Tendremos que introducir la contraseña decidida anteriormente.
Mysql> create database bacula;
Mysql> exit

    # Ya hemos terminado con MySQL. Ahora vamos a instalar Bacula.
$ sudo apt-get install bacula
    # Durante el proceso de instalación nos preguntará sobre nuestras opciones de correo
    # preferidas. Diremos que el correo se entrega localmente. Como nombre de dominio
    # para el correo, podemos indicar algo como “bacula-server.local”.
    # También se nos pregunta sobre la base de datos. Se nos sugerirá la configuración del
    # paquete bacula-director-mysql, que pone en contacto el bacula-director con mysql.
    # Por defecto, el programa de configuración utilizará el usuario root@localhost para crear
    # el usuario bacula@localhost y darle permisos sobre la base de datos “bacula”.
    # También se configurará automáticamente la parte relativa a la base de datos en el archivo
    # bacula-dir.conf.

$ sudo vi /etc/bacula/bacula-dir.conf
```

Si revisamos el archivo de configuración, veremos que hay muchas opciones a configurar para adaptarlas a nuestro sistema. La primera consideración es la siguiente: los servicios serán accesibles desde la red. Por ello debemos cambiar las direcciones IP en el archivo “bacula-dir.conf”, así como en “bacula-sd.conf”, “bacula-fd.conf” y “bconsole.conf”.

- En el recurso Director:

```
DirAddress = 192.168.100.100
# Esto mismo debemos hacerlo en los otros archivos de configuración
```

- En el recurso Catalog de “bacula-dir.conf”, debemos comprobar la forma en que Bacula accede a la base de datos. Los datos deben ser:

```
dbname = bacula; DB Address = localhost; dbuser = “bacula”;
dbpassword = “cualquier-cosa”
```

En este instante, Bacula ya sabe acceder a la base de datos.

NOTA: Si comprobamos que hubo algún problema con la base de datos (por ejemplo, que está vacía) podemos recrearla editando el archivo /usr/share/bacula-director/make_mysql_tables. Modificamos el valor XXX_DBNAME_XXX por el nombre de la base de datos que creamos al principio: es decir “bacula”. Hecho esto, ejecutamos el siguiente comando:

```
$ sudo /usr/share/bacula-director/make_mysql_tables -u bacula -p
# Cuando nos pida la contraseña, introducimos la contraseña indicada en bacula-dir.conf.
```

Ahora ya se ha creado la base de datos para bacula. Podemos iniciar los demonios “bacula-director”, “bacula-sd” y “bacula-fd”.

```
$ sudo /etc/init.d/bacula-director restart
$ sudo /etc/init.d/bacula-sd restart
$ sudo /etc/init.d/bacula-fd restart
```

Nuestra primera experiencia con la consola de bacula:

Para interactuar con Bacula, podemos usar la consola. Para ello, ejecutamos el siguiente comando:

```
$ bconsole
```

```
*
```

*# el símbolo * que muestra Bacula, es el prompt. Espera que metamos un comando. Por ejemplo, # el siguiente:*

```
* status all
```

de este modo compruebo el estado de los componentes de bacula. Se puede observar las # dos copias programadas (por defecto) en /etc/bacula/bacula-dir.conf. También puedo # comprobar el estado de conectividad con los otros demonios.

PARTE 2. Instalando freeNAS

De momento dejaremos b acula a un lado. Necesitamos configurar la NAS para volcar las copias sobre ella. Por ello, lo siguiente es configurar freeNAS en otra m quina.

Utilizar  un m quina virtual con 3 discos. Uno de ellos (de 4GB) lo emplear  para instalar freeNAS. Los otros dos (de 5GB) los emplear  para montar un RAID 1 sobre ellos. Hay que tener en cuenta que este RAID es software. Siempre es preferible que sea hardware, sobre todo si la misi n es cr tica.

Arranco la m quina con el CD de freeNAS como primera opci n de arranque, y dejo que inicie con la primera opci n (que casi no da tiempo a ver).

En la primer pregunta que me hace freeNAS, me ofrece las siguientes opciones:
Install/Upgrade to hard drive/flash device, etc.
Shell
Reboot System
Shutdown System

freeNAS recomienda la instalaci n en una memoria flash USB, ya que ocupa muy poco y esto deja libre al disco solo para tareas de almacenamiento. En nuestro caso (supuestamente) contamos con un disco independiente para freeNAS. Elijo la primera opci n.

A continuaci n me pregunta en qu  disco debo instalar freeNAS. Elijo el disco de 4GB, etiquetado como "ada0 VBOX HARDDISK 1.0 – 4 GiB".

El proceso de instalaci n me vuelve a advertir, y a recomendar que instale en una memoria flash. Confirmo pulsando en "yes". El proceso de instalaci n a penas dura 1 minuto.

Extraigo el CD de la bandeja y reinicio. Durante el proceso de inicio, freeNAS intenta obtener una IP por dhcp. Si no disponemos dhcp fracasara la obtenci n de una IP, y seguir  el proceso de arranque hasta mostrar un men  donde podemos iniciar la configuraci n de freeNAS.

Lo primero es configurar la interfaz de red. Para ello, elijo "1) Configure Network Interfaces". Se iniciar  un di logo en el que puedo asignar una IP y m scara v lidas para poder acceder via web a freeNAS. Le asigno la IP 192.168.100.200 (esa es mi elecci n particular).

Lo segundo que har  ser  resetear el usuario y la contrase a para acceder la interfaz gr fica de administraci n de freeNAS. Elijo la opci n "Reset WebGUI login credentials". FreeNAS me indica que los nuevos usuario y contrase a son "admin/freenas".

Ahora solo me queda conectarme en remoto a freeNAS. Para ello, desde una m quina con explorador web (como por ejemplo bacula-server) . Abro el explorador y ataco a "<http://192.168.100.200>", e introduzco las credenciales "admin/freenas", entrando en la consola de administraci n gr fica.

Antes de nada, vamos a crear un nuevo usuario y un nuevo grupo. Se trata del usuario "bacula" y el grupo "tape". Nos va a hacer falta, ya que Bacula utiliza este usuario y este grupo para escribir en los vol menes de backup. Antes de nada tenemos que comprobar el UID y GID de "bacula" y de "tape" en el servidor Bacula. En mi caso son 116 y 26.

NOTA: si existe ya un grupo en freeNAS con dicho GID podemos cambiar el GID de tape en bacula-server. En mi caso le he asignado el GID 32 (en /etc/group).

Lo siguiente es crear un volumen RAID 1 en freeNAS con los dos discos de 5GB. Para ello elegimos la sección “Storage/Create volume”. Como nombre al volumen, yo le asigno “Backup”, marco los dos discos ada1 y ada2, como sistema de archivos he elijo ZFS y los agrupo en RAID 1 (mirror).

El nuevo volumen, está ubicado en /mnt/Backup. Edito las opciones del volumen y elijo la opción de compresión recomendada “lzjb”. También cambio los permisos del volumen, y como usuario propietario, coloco al creado anteriormente, “bacula”, con permisos de lectura y escritura. El grupo “tape”, tendrá permisos de lectura únicamente.

Ahora vamos a activar NFS en FreeNAS para poder montar en “bacula-server” el volumen “Backup”. Para ello, hago lo siguiente en la consola de administración de freeNAS:

1. Creamos una compartición. Abrimos la pestaña “sharing”, y elegimos “add new UNIX share”.
 1. Comment: Compartición para backup.
 2. Path: /mnt/Backup *# es la ubicación del volumen.*
 3. Authorized network or IP addresses: 192.168.100.100 *# la ip de bacula-server.*
 4. All directories
 5. maproot user: bacula
 6. maproot group: tape
2. Activamos el servicio NFS
 1. Nos vamos a la pestaña services.
 2. Activamos la pestaña “NFS”.

Ahora deberíamos poder montar el volumen en bacula-server. Es decir, deseamos poder acceder a la NAS desde bacula-server. Para ello, dejamos la máquina con freeNAS, y volvemos a “bacula-server”. En primer lugar tenemos que crear el punto de montaje. Yo elijo /mnt/backupNAS. Sigo los siguientes pasos:

1. Instalos los paquetes necesarios en bacula-server para poder montar unidades NFS
 1. `$ sudo apt-get install portmap`
 2. `$ sudo apt-get install nfs-common`
2. Creo el directorio que usaré como punto de montaje:
 1. `$ sudo mkdir /mnt/backupNAS`
 2. `$ sudo chown bacula.tape /mnt/backupNAS`
3. Monto el volumen:
 1. `$ sudo mount -t nfs 192.168.100.200:/mnt/Backup /mnt/backupNAS`

Ahora vamos configurar el demonio de almacenamiento “bacula-sd” para que cree los volúmenes de copia en la NAS. Para ello, editamos el archivo /etc/bacula/bacula-sd.conf. Buscamos dentro de dicho archivo el recurso “Device”. Dentro de dicho recurso encontraremos el parámetro “Archive Device”, al que le daremos el valor /mnt/backupNAS. Para que los cambios tengan efecto, reiniciamos el servicio:

```
$ /etc/init.d/bacula-sd restart
```

Bacula ya está listo para realizar copias. Nos faltan las siguientes cosas:

1. Crear un volumen sobre el que escribir.
2. Instalar y configurar un cliente.
3. Crear trabajos de copia.
4. Establecer una temporalización.

PARTE 3. Crear un volumen para escribir sobre él.

Si ya estamos conectados desde bacula-server por NFS a freeNAS, necesitamos crear un volumen sobre el que realizarán las copias. Hay que tener en cuenta, que bacula escribe sobre volúmenes de manera secuencial. Es decir, que al crear un volumen, aunque en nuestro caso sea un archivo, Bacula escribirá secuencialmente sobre dicho archivo como si fuese una cinta. La copia a cinta es más compleja, ya que necesitaremos cambiar la cinta periódicamente, y crear volúmenes continuamente. Sin embargo, en nuestro caso, que usamos una archivo como volumen, se simplifica todo, ya que al crear el volumen, éste estirará tanto como dé de sí el disco.

NOTA: En /etc/bacula-dir.conf solo hay un recurso "Storage" definido. Su configuración es la siguiente

```
Storage {
  Name = File
                                     # Do not use "localhost" here
  Address = 192.168.100.100           # N.B. Use a fully qualified name here
  SDPort = 9103
  Password = "IkceCtAChH3CLnV5Njbur2s5N0yj9p7v4"
  Device = FileStorage
  Media Type = File
}
```

Se trata de almacenamiento en disco. Si observamos bien el archivo de configuración, podemos ver que hay otros recursos "Storage", comentados, para copia en cinta, y DVD.

Además del recurso "Storage", también hay otro recurso necesario para las copias. Se trata del recurso "Pool". Es un recurso que permite gestionar varios volúmenes pertenecientes a una misma serie de copias. Podemos indicar, cuanto tiempo tardaremos en reutilizar los volúmenes, por ejemplo, en un esquema rotatorio de copias.

Al crear el volumen, bacula nos va a preguntar algunas cosas y otras la va a dar por hecho. Por ejemplo. Como solo tenemos definido un catálogo (MyCatalog), Bacula lo elige por nosotros. También elige por nosotros el punto de almacenamiento, (ya que solo hay un recurso "Device" en "/etc/bacula/bacula-sd.conf"). En cambio, nos preguntará el nombre del volumen, y Pool que usaremos. Yo elijo el nombre "vol001" y el Pool "File".

```
*label
Automatically selected Catalog: MyCatalog
Using Catalog "MyCatalog"
Automatically selected Storage: File
vol001
Defined Pools:
  1: Default
  2: File
  3: Scratch
Select the Pool (1-3): 2
```

Ahora debe de haber un archivo llamado vol001 en /mnt/Backup en la NAS. Podemos comprobarlo desde la shell de freeNAS. Desde bacula-server, no podremos, puesto que no pusimos permisos de lectura a terceros.

PARTE 4. Instalar y configurar un cliente

Suponiendo que tenemos un cliente Windows, descargamos el paquete de instalación (bacula-win32-5.0.3.exe) del cliente de bacula y lo instalamos.

Iniciamos la instalación. Durante la instalación, podemos elegir una configuración personalizada, aunque es preferible elegir la configuración típica. Posteriormente se generará un archivo, llamado C:\w7-fd.conf donde estarán los datos que nos interesan para que el Bacula-director y el cliente que estamos instalando se puedan comunicar.

A continuación, necesitamos indicar quién es el bacula-director. Los datos a indicar los debemos extraer del archivo /etc/bacula/bacula-dir.conf de bacula-server, en concreto, del recurso "Director", que muestro a continuación:

```
Director {                                # define myself
  Name = bacula-server-dir
  DIRport = 9101                          # where we listen for UA connections
  QueryFile = "/etc/bacula/scripts/query.sql"
  WorkingDirectory = "/var/lib/bacula"
  PidDirectory = "/var/run/bacula"
  Maximum Concurrent Jobs = 1
  Password = "yveAet993vzswug3KoZUKjVbdZy0nrSqIqQCFH80rIVS"      # Console
password
  Messages = Daemon
  DirAddress = 192.168.100.100
}
```

Por tanto, los datos a indicar son:

DIR Name:	bacula-server-dir
DIR:	9101
DIR Password:	yveAet993vzswug3KoZUKjVbdZy0nrSqIqQCFH80rIVS
DIR Address:	192.168.100.100

En un cliente normal, desde el que no requiremos monitorizar Bacula, basta con que instalemos el cliente. En nuestro caso instalaremos también la consola.

Solo nos queda una cosa para sincronizar cliente y director: hacer saber al bacula-director que hay un cliente nuevo en la ciudad... Para ello, editamos /etc/bacula/bacula-dir.conf, y añadimos el siguiente recurso "Client" (es buena idea hacerlo junto a los otros clientes para que el archivo de configuración tenga cierto orden). Los datos que indico son obtenidos del archivo [C:\w7-fd.conf](#). En concreto, necesito lo siguiente:

Nombre:	w7-fd
Puerto:	9102
Password:	fCk7TXoFJsKh+eqnUWxIdYeoPzTnxTjZ6D19UH/EssZa

El recurso a añadir es:

```
Client {
  Name = w7-fd
  Address = 192.168.100.10  # Dirección IP del cliente
  FDPort = 9102
  Catalog = MyCatalog
  Password = "fCk7TXoFJsKh+eqnUWxIdYeoPzTnxTjZ6D19UH/EssZa"
  File Retention = 30 days      # 30 días
```

```
Job Retention = 6 months      # seis meses
AutoPrune = yes              # Podar trabajos/archivos obsoletos
}
```

Para que el cambio de configuración tenga efecto, debemos reiniciar el bacula-director:

```
$ sudo /etc/init.d/bacula-director restart
```


PARTE 5. Comprobar el estado del cliente

Ahora podemos comprobar el estado del cliente. Es decir, ¿Está en línea?

En el cliente Windows, ejecutamos “Inicio\bacula\bconsole”. Debería aparecer algo como esto:

```
Connecting to Director 192.168.100.100:9101
1000 OK: bacula-server-dir Versión: 5.0.1(24 February 2010)
Enter a period to cancel a command.
*
```

Introduzco el comando “status client”:

```
*status client
Los recursos Clientes definidos son:
    1: bacula-server-fd
    2: w7-fd
Seleccione recurso Cliente (File Daemon) (1-2): 2
Conectando al Cliente w7-fd en 192.168.100.10:9102

w7-fd Version: 5.0.3 (04 August 2010)  VSS Linux Cross-compile Win32
Daemon started 12-Oct-11 13:00. Jobs: run=0 running=0.
  Heap: heap=0 smbytes=12,920 max_bytes=13,015 bufs=53 max_bufs=54
  Sizeof: boffset_t=8 size_t=4 debug=0 trace=1

Running Jobs:
Director connected at: 12-Oct-11 13:08
No Jobs running.
====

Terminated Jobs:
====
*
```

Como podemos ver, el cliente está en línea y además podemos usar “bconsole” en remoto.

Puedo probar también a ejecutar “Inicio\bacula\bat”. Es una consola de monitorización de Bacula.

PARTE 6. Definir trabajos de copia en Bacula

Vamos a definir un trabajo de copia para el cliente Windows. Para ello, debemos definir un recurso "Job" en "/etc/bacula/bacula-dir.conf" de bacula-server.

```
Job {
  Name = "w7-Backup"
  JobDefs = "DefaultJob"
  Type = Backup
  Client = w7-fd
  FileSet = "w7-FileSet"
  Schedule = "Temporizacion-clientes"
  Storage = File
}
```

Como se puede observar, hay parámetros que no se autodefinen, como JobDefs, FileSet, Schedule o Storage.

- JobDefs: es un conjunto de parámetros preddefinidos. Está definido en el recurso "JobDefs".
- FileSet: es la ruta a los archivos a copiar. Hay que definirlo expresamente para cada tipo de cliente.
- Schedule: indica la temporización. Podemos definir cuando se harán las copias y de qué tipo serán.
- Storage: Se trata del dispositivo de almacenamiento. En nuestro caso está en /mnt/backupNAS, como ya hemos visto anteriormente.

Vamos a crear los recursos FileSet y Schedule:

```
FileSet {
  Name = "w7-FileSet"
  Include {
    Options {
      signature = MD5
    }
    File = "C:/Backup"
  }
}
```

Es decir, se copiará lo que está en la carpeta "[C:\Backup](#)" del cliente Windows 7.

```
Schedule {
  Name = "Temporizacion-clientes"
  Run = Full sun at 19:00
  Run = Incremental mon-sat at 19:00
}
```

Reiniciamos el bacula-director. Podemos comprobar que la temporización ha sido entendida mediante:

```
$ sudo bconsole
* status Director
```

PARTE 7. Lanzar un trabajo de copia y restauración manualmente

Vamos a crear la carpeta "C:\Backup" en Windows 7. También vamos a crear una carpeta llamada "C:\Restauración". Coloca dentro de [C:\Backup](#) algunos archivos, por ejemplo las imágenes de muestra de Windows.

```
$ bconsole
* run # decimos a bacula que ejecute manualmente un trabajo.
Seleccionado automáticamente Catálogo: MyCatalog
Usando Catálogo "MyCatalog"
Un nombre de job debe de ser especificado.
Los recursos de trabajo definidos son:
    1: BackupClient1
    2: BackupCatalog
    3: w7-Backup
    4: RestoreFiles
Seleccione recurso Job (1-4):
```

Elegimos el trabajo 3. Nos mostrará lo siguiente:

```
Nombre del Job: w7-Backup
Nivel: Incremental
Cliente: w7-fd
FileSet: w7-FileSet
Pool: File (Desde Recurso Job)
Storage: File (Desde Recurso Job)
Cuando: 2011-10-12 13:46:57
Prioridad: 10
OK para ejecutar? (si/mod/no):
```

Indicamos "si".

Para comprobar lo que ha ocurrido, escribimos lo siguiente:

```
* messages
```

Obtendremos la siguiente información:

```
12-oct 12:50 bacula-server-dir JobId 0: Error Fatal: bsock.c:135 No se puede
conectar a Cliente: w7-fd en 192.168.100.10:9102. ERR=Conexión rehusada
12-oct 13:50 bacula-server-dir JobId 1: No encontrado registro anterior de Job
de respaldo completo.
12-oct 13:50 bacula-server-dir JobId 1: Ninguna copia de seguridad Completa
anterior encontrada en el catálogo. Haciendo copia de seguridad COMPLETA.
12-oct 13:50 bacula-server-dir JobId 1: Iniciar Respaldo JobId 1, Job=w7-
Backup.2011-10-12_13.50.26_04
12-oct 13:50 bacula-server-dir JobId 1: Usando Dispositivo "FileStorage"
*messages
12-oct 13:50 bacula-server-sd JobId 1: Wrote label to prelabeled Volume "vol001"
on device "FileStorage" (/mnt/backupNAS)
12-oct 13:50 w7-fd JobId 1: Generate VSS snapshots. Driver="VSS Vista",
Drive(s)="C"
12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "Task Scheduler
Writer", State: 0x1 (VSS_WS_STABLE)
12-oct 13:51 bacula-server-sd JobId 1: Job write elapsed time = 00:00:47,
Transfer rate = 124.2 K Bytes/second
12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "VSS Metadata Store
Writer", State: 0x1 (VSS_WS_STABLE)
12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "Performance Counters
Writer", State: 0x1 (VSS_WS_STABLE)
```

12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "System Writer", State: 0x1 (VSS_WS_STABLE)

12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "ASR Writer", State: 0x1 (VSS_WS_STABLE)

12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "Shadow Copy Optimization Writer", State: 0x1 (VSS_WS_STABLE)

12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "WMI Writer", State: 0x1 (VSS_WS_STABLE)

12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "COM+ REGDB Writer", State: 0x1 (VSS_WS_STABLE)

12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "MSSearch Service Writer", State: 0x1 (VSS_WS_STABLE)

12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "BITS Writer", State: 0x1 (VSS_WS_STABLE)

12-oct 13:51 w7-fd JobId 1: VSS Writer (BackupComplete): "Registry Writer", State: 0x1 (VSS_WS_STABLE)

12-oct 13:51 bacula-server-dir JobId 1: Bacula bacula-server-dir 5.0.1 (24Feb10): 12-oct-2011 13:51:16

```

Build OS:                i486-pc-linux-gnu ubuntu 10.04
JobId:                   1
Job:                     w7-Backup.2011-10-12_13.50.26_04
Backup Level:           Full (actualizar desde Incremental)
Client:                  "w7-fd" 5.0.3 (04Aug10) Linux,Cross-compile,Win32
FileSet:                 "w7-FileSet" 2011-10-12 13:50:26
Pool:                    "File" (From Recurso Job)
Catalog:                 "MyCatalog" (From Recurso de cliente)
Storage:                 "File" (From Recurso Job)
Scheduled time:         12-oct-2011 13:46:57
Start time:              12-oct-2011 13:50:28
End time:                 12-oct-2011 13:51:16
Elapsed time:            48 secs
Priority:                 10
FD Files Written:        9
SD Files Written:        9
FD Bytes Written:        5,838,995 (5.838 MB)
SD Bytes Written:        5,840,210 (5.840 MB)
Rate:                    121.6 KB/s
Software Compression:    None
VSS:                     si
Encryption:              no
Accurate:                 no
Volume name(s):          vol001
Volume Session Id:       1
Volume Session Time:     1318415553
Last Volume Bytes:       5,845,354 (5.845 MB)
Non-fatal FD errors:     0
SD Errors:                0
FD termination status:   OK
SD termination status:   OK
Termination:             Respaldo OK

```

12-oct 13:51 bacula-server-dir JobId 1: Empezar podando trabajos más antiguos de 6 months .

12-oct 13:51 bacula-server-dir JobId 1: No encontraron Jobs para podar.

12-oct 13:51 bacula-server-dir JobId 1: Comenzando a podar trabajos.

12-oct 13:51 bacula-server-dir JobId 1: No se encontraron archivos para podar.

12-oct 13:51 bacula-server-dir JobId 1: Fin auto podar(prune).

NOTA: Solo un detalle. En las líneas anteriores podemos leer lo siguiente:

12-oct 13:50 bacula-server-dir JobId 1: Ninguna copia de seguridad Completa anterior encontrada en el catálogo. Haciendo copia de seguridad COMPLETA.

Significa que si bacula detecta que no existe una copia COMPLETA de algo, e intento hacer una copia incremental o diferencial, cancelará la copia incremental/diferencial y realizará la completa. A partir de ella, podrá hacer la copias incrementales/diferenciales.

Ahora podemos comprobar si se ha transferido lo que esperábamos:

```
* status Director
bacula-server-dir Versión: 5.0.1 (24 February 2010) i486-pc-linux-gnu ubuntu
10.04
Demonio iniciado 12-oct-11 13:41, 1 Job ejecutando desde el inicio.
Heap: heap=368,640 smbytes=61,785 max_bytes=72,702 bufs=254 max_bufs=286
```

Scheduled Jobs:

Nivel Tipo Pri Scheduled Nombre Volumen

```
=====
===
Incremental Backup 10 12-oct-11 19:00 w7-Backup vol001
Incremental Backup 10 12-oct-11 23:05 BackupClient1 vol001
Full Backup 11 12-oct-11 23:10 BackupCatalog vol001
====
```

Jobs Ejecutando:

Consola conectada en 12-oct-11 13:46

Ningún Jobs ejecutando.

```
=====
```

Jobs No Terminados:

```
JobId Nivel Archivos Bytes Estado Finalizado Nombre
=====
1 Full 9 5.838 M OK 12-oct-11 13:51 w7-Backup
```

Para realizar una restauración, haremos lo siguiente. Ejecutamos restore en la consola de bacula:

```
* restore
```

Primero debe seleccionar una o más JobIds que contienen archivos para ser restaurado. Se le presentará varios métodos para especificar los JobIDs. Luego se le permitirá seleccionar los archivos de los JobIds que deben ser restaurados.

Para seleccionar el JobIds, usted dispone de las siguientes opciones:

- 1: Listar los últimos 20 Jobs ejecutados
- 2: Listado de Jobs donde un determinado archivo se ha guardado
- 3: Introduzca lista de JobIds separados por comas para seleccionar
- 4: Introduzca lista de comandos SQL
- 5: Seleccionar el respaldo más reciente para un cliente
- 6: Seleccione un respaldo de un cliente antes de un período de tiempo especificado
- 7: Introduzca una lista de archivos para restaurar
- 8: Introduzca una lista de archivos para restaurar antes de un período de tiempo especificado
- 9: Encuentre el JobIds del respaldo más reciente para un cliente
- 10: Encuentre el JobIds del respaldo de un cliente antes de un período de tiempo especificado
- 11: Introduzca una lista de directorios para restaurar por JobIds encontrado
- 12: Seleccione restauración completa para una fecha específica de Job
- 13: Cancelar

Seleccione un ítem: (1-13):

Elegimos la opción 5. Y bacula responde lo siguiente:

Cientes definidos:

- 1: bacula-server-fd
- 2: w7-fd

Seleccione el Cliente (1-2):

Elegimos al cliente que acabamos de respaldar, es decir 2. Bacula responde lo siguiente:

Seleccionado automáticamente FileSet: w7-FileSet

```
+-----+-----+-----+-----+-----+-----+-----+
| JobId | Level | JobFiles | JobBytes | StartTime          | VolumeName |
+-----+-----+-----+-----+-----+-----+-----+
|      1 | F     |          9 | 5,838,995 | 2011-10-12 13:50:28 | vol001     |
+-----+-----+-----+-----+-----+-----+-----+
```

Usted ha seleccionado el siguiente JobId: 1

Construyendo árbol de directorios para JobId(s) 1 ...
8 archivos insertados en el árbol.

Usted está entrando ahora en el modo de selección de archivo donde usted agrega(mark) y elimina(unmark) los archivos que va a restaurar. No hay archivos añadido inicialmente, a menos que usted utilice la palabra clave "all" en la línea de comandos. Introduzca "done" para salir de este modo.

```
cwd es: /
$
```

Ahora, bacula espera que elijamos los archivos que deseamos restaurar. En el prompt de selección de archivos (\$) podemos hacer lo siguiente:

```
$ ls
C:/
$ cd C:/
cwd es: C:/
$ dir
drwxrwxrwx  1 root    root          0  2011-10-05 23:25:30  C:/Backup/
$ cd Backup
cwd es: C:/Backup/
$ dir
-rwxrwxrwx  1 root    root    879394  2011-10-05 23:25:30
C:/Backup/Chrysanthemum.jpg
-rwxrwxrwx  1 root    root    845941  2011-10-05 23:25:30
C:/Backup/Desert.jpg
-rwxrwxrwx  1 root    root    595284  2011-10-05 23:25:30
C:/Backup/Hydrangeas.jpg
-rwxrwxrwx  1 root    root    775702  2011-10-05 23:25:30
C:/Backup/Jellyfish.jpg
-rwxrwxrwx  1 root    root    780831  2011-10-05 23:25:30
C:/Backup/Koala.jpg
-rwxrwxrwx  1 root    root    561276  2011-10-05 23:25:30
C:/Backup/Lighthouse.jpg
-rwxrwxrwx  1 root    root    777835  2011-10-05 23:25:30
C:/Backup/Penguins.jpg
-rwxrwxrwx  1 root    root    620888  2011-10-05 23:25:30
C:/Backup/Tulipans.jpg
```

Ahora podemos elegir los archivos que queremos restaurar mediante el comando "mark". Si escribimos "mark *" indicamos que los marque todos:

```
$ mark *
      8 archivos marcados.
$ ls
```

```
*Chrysanthemum.jpg
*Desert.jpg
*Hydrangeas.jpg
*Jellyfish.jpg
*Koala.jpg
*Lighthouse.jpg
*Penguins.jpg
*Tulipans.jpg
$ done
```

Y volvemos a la siguiente pregunta:

```
8 archivos seleccionados para ser restaurado.
Ejecutar Job de Restauración
Nombre del Job: RestoreFiles
Bootstrap: /var/lib/bacula/bacula-server-dir.restore.1.bsr
Donde: /nonexistent/path/to/file/archive/dir/bacula-restores
Reemplazar: always
FileSet: Full Set
Respaldo Cliente: w7-fd
Restaurar Cliente: w7-fd
Storage: File
Cuando: 2011-10-12 14:06:13
Catalogo: MyCatalog
Prioridad: 10
Opciones de Plugin: *Ninguno*
OK para ejecutar? (si/mod/no):
```

Ahora todo es correcto menos una cosa: el parámetro “Donde”. Para modificarlo, introducimos en la pregunta, el valor “mod”. Y báculo nos responde:

```
Parámetros para modificar:
  1: Level
  2: Storage
  3: Job
  4: FileSet
  5: Restaurar Client
  6: Cuando
  7: Prioridad
  8: Bootstrap
  9: Donde
 10: Reubicar Archivo
 11: Reemplazar
 12: JobId
 13: Opciones de Plug-in
Seleccione los parámetros para modificar (1-13):
```

Así que especificamos “9”.

Por favor, introduzca el prefijo de ruta para restaurar (/ para ninguno):

```
C:/Restauracion
Ejecutar Job de Restauración
Nombre del Job: RestoreFiles
Bootstrap: /var/lib/bacula/bacula-server-dir.restore.1.bsr
Donde: C:/Restauracion
Reemplazar: always
FileSet: Full Set
Respaldo Cliente: w7-fd
Restaurar Cliente: w7-fd
Storage: File
Cuando: 2011-10-12 14:06:13
Catalogo: MyCatalog
```

Prioridad: 10
Opciones de Plugin: *Ninguno*
OK para ejecutar? (si/mod/no):

Ahora ya estamos en condiciones de aceptar. Respondemos "si". Para comprobar lo que tiene que decir bacula sobre la restauración, le preguntamos con el comando "messages":

```
*messages
12-oct 14:10 bacula-server-dir JobId 2: Iniciar Job Restaurar RestoreFiles.2011-
10-12_14.10.22_06
12-oct 14:10 bacula-server-dir JobId 2: Usando Dispositivo "FileStorage"
12-oct 14:10 bacula-server-sd JobId 2: Ready to read from volume "vol001" on
device "FileStorage" (/mnt/backupNAS).
12-oct 14:10 bacula-server-sd JobId 2: Forward spacing Volume "vol001" to
file:block 0:204.
12-oct 14:10 bacula-server-sd JobId 2: End of Volume at file 0 on device
"FileStorage" (/mnt/backupNAS), Volume "vol001"
12-oct 14:10 bacula-server-sd JobId 2: End of all volumes.
12-oct 14:10 bacula-server-dir JobId 2: Bacula bacula-server-dir 5.0.1
(24Feb10): 12-oct-2011 14:10:24
Build OS: i486-pc-linux-gnu ubuntu 10.04
JobId: 2
Job: RestoreFiles.2011-10-12_14.10.22_06
Restaurar Cliente: w7-fd
Hora de inicio: 12-oct-2011 14:10:24
Hora de finalización: 12-oct-2011 14:10:24
Archivos Esperados: 8
Archivos Restaurados: 8
Bytes Restaurados: 5,838,751
Tasa: 0.0 KB/s
Errores de FD: 0
Estado de terminación de FD: OK
Estado de terminación de SD: OK
Terminación: Restauración OK

12-oct 14:10 bacula-server-dir JobId 2: Empezar podando trabajos más antiguos de
6 months .
12-oct 14:10 bacula-server-dir JobId 2: No encontraron Jobs para podar.
12-oct 14:10 bacula-server-dir JobId 2: Comenzando a podar trabajos.
12-oct 14:10 bacula-server-dir JobId 2: No se encontraron archivos para podar.
12-oct 14:10 bacula-server-dir JobId 2: Fin auto podar(prune).
```

Es decir, la restauración se ha hecho correctamente. De hecho, si vamos a [C:\Restauracion](#) en Windows 7, veremos que hay una nueva carpeta, llamada "C", donde están los archivos restaurados.

Algunas notas finales

1. Sobre el catálogo y bootstrap.

Bacula está configurada por defecto para que (con algunos ajuste de la configuración) se guarde también el catálogo de copias (es decir, la base de datos mysql). No solo el catálogo es importante. También es importante el archivo Bootstrap. Si ocurriese un desastre, y la máquina donde corre bacula muriese, queremos tener disponible tanto el catálogo con el bootstrap.

El bootstrap está en /var/lib/bacula/, así que debería haber un recurso “Job” para copiar este sitio.

La copia del catálogo, se hará a través de un dump de la base de datos. Para ello, usaremos un script con el comando siguiente:

```
mysqldump -u bacula -p"perro20" bacula > bacula-catalog.sql
```

Este script podemos ejecutarlo desde el recurso “Job” llamado “BackupCatalog” en el parámetro “RunBeforeJob”.

2. Sobre la NAS

Para que no haya que montar manualmente la NAS en bacula-server, podemos incluir una línea en /etc/fstab para que se monte automáticamente. Personalmente he tenido algún problema con fstab, lo que he resuelto sencillamente creando un script bash para montar la unidad, invocado desde un enlace en rc2.d

El script, al que yo llamo “backupNAS” sería el siguiente:

```
#!/bin/bash
mount 192.168.100.200:/mnt/Backup /mnt/backupNAS
```

Para crear forzar su ejecución en inicio:

```
$ sudo ln -s /etc/init.d/backupNAS /etc/rc2.d/S99backupNAS
$ sudo chmod 755 /etc/init.d/backupNAS
```

Referencia de freeNAS: http://doc.freenas.org/index.php/Main_Page
Referencia de Bacula: <http://www.bacula.org/5.0.x-manuals/en/main/main/index.html>