



Grupo
Opentia
Interoperabilidad + Software Libre



[CÓMO] BACKUPS "HOTPLUG" E INCREMENTALES A DISCO DURO EXTERNO

REFERENCIA LIBRE-CONS-BACKUPS_HOTPLUG -- VERSIÓN 0.140

VIERNES, 22 DE ENERO DE 2010

AUTOR: LUCAS J. GONZÁLEZ (CONSULTORIA@OPENTIA.ES)

Consultoría de horizontes abiertos

www.opentia.com

Documento originalmente realizado para:

Proyecto ARQVIPO Digital

www.obvlco.org

por el equipo de Consultoría del
Grupo OPENTIA



consultoria@opentia.es

Proyecto:	ARQUIPO
Cliente:	ARQUIPO
Título:	[Cómo] Backups "HotPlug" e Incrementales a Disco Duro Externo
Versión:	1.140
Fecha:	2010-01-23
Fichero:	guiaburros_backups_halevt_rsnapshot-OPENTIA-20100123.odt
Difusión:	LIBRE DIFUSIÓN
Circulado para:	LECTURA, ANÁLISIS, ARCHIVO Y MEJORA
Derechos de autor:	© 2009, OPENTIA, S.L. (CIF B84267889)
Licencia	GNU Free Documentation License 3.0
Marcas	La palabra "OPENTIA" es una marca comunitaria con validez en toda la Unión Europea, propiedad del Grupo OPENTIA. El resto de marcas utilizadas en este documento son propiedad de sus respectivos dueños.
Patentes	OPENTIA declara no patentar software y respalda la actual ilegalidad de las patentes de software en el marco legislativo de los países europeos.

Índice de contenido

1 INTRODUCCIÓN.....	5
1.1 SOBRE ESTE DOCUMENTO.....	5
1.2 SOBRE LAS HERRAMIENTAS ESCOGIDAS.....	5
2 GUÍA DE INSTALACIÓN Y CONFIGURACIÓN.....	5
2.1 PASO 1: INSTALACIÓN DE LOS PAQUETES NECESARIOS.....	5
2.2 PASO 2: CONFIGURACIÓN DE RSNAPSHOT.....	5
2.3 PASO 3: CONFIGURAR LOS PERMISOS.....	8
2.4 PASO 4: IDENTIFICAR EL DISCO.....	9
2.5 PASO 5: CONFIGURACIÓN DE HALEVT.....	10
2.6 PASO 6: APLICAR LAS CONFIGURACIONES.....	12
3 MÁS INFORMACIÓN Y OPCIONES.....	12

1 Introducción

1.1 Sobre este documento

Podemos hacer que nuestra máquina reconozca el disco duro externo que le hemos conectado, realice una copia de seguridad incremental del sistema o de los directorios que queramos, la guarde en él y lo vuelva a desmontar, listo para ser guardado en lugar seguro.

Tenemos la posibilidad de hacer copias de seguridad incrementales mediante *rsync* gracias a un par de ingeniosos trucos con enlaces duros. Pero este método ha evolucionado mucho y son muchas también las soluciones completas y avanzadas que hay disponibles.

1.2 Sobre las herramientas escogidas

Para el sistema que nos ocupa se ha usado *rsnapshot* como solución para la realización de copias de seguridad incrementales, y *halevt* como listener de eventos del sistema y disparador de acciones. Ya habíamos implementado otras soluciones anteriormente utilizando *ivman*, el listener en el que está basado *halevt*, pero parece que *ivman* no se lleva bien con las nuevas versiones de *HAL*, además de que *halevt* nos ha sorprendido gratamente por su estabilidad y flexibilidad.

Para evitar problemas, primero configuraremos *rsnapshot*, y luego pasaremos a configurar *halevt* y la detección del disco.

2 Guía de instalación y configuración

2.1 Paso 1: Instalación de los paquetes necesarios

El sistema que vamos a utilizar es Debian. Para instalar todos los paquetes necesarios podemos ejecutar el siguiente comando como root:

```
:~# apt-get install sudo rsnapshot halevt
```

2.2 Paso 2: Configuración de *rsnapshot*

La configuración de *rsnapshot* tiene lugar en el archivo */etc/rsnapshot.conf*. Lo primero que tenemos que definir en él es la ruta a la raíz del directorio que contendrá las copias de seguridad. En el ejemplo que estamos mostrando la ruta especificada es */media/backups/COPIAS_SEGURIDAD/*:

```
snapshot_root /media/backups/COPIAS_SEGURIDAD/
```

Nótese que se trata de un subdirectorio ubicado dentro del disco externo. El disco externo se montará en **/media/backups**, pero no incluiremos nada en el archivo **/etc/fstab** para ello, sino que será *halevt* quien lo cree dinámicamente antes de montarlo y quien lo elimine al desmontarlo. De hecho, si **/etc/fstab** ya contuviese una entrada para el dispositivo, *halevt* se negaría rotundamente a montarlo.

```
Mount error for (null):  
DBus Error org.freedesktop.Hal.Device.Volume.PermissionDenied: Device /dev/sdb1  
is listed in /etc/fstab. Refusing to mount.  
wepssaint@eolo:~$
```

Figura 1: *halevt* no montará dispositivos definidos en el archivo **/etc/fstab**.

Como hemos apuntado, el directorio **/media/backups** no debe existir en el sistema de archivos antes de montar el disco externo, algo que nos vendrá muy bien para hacer un poco más sólida la solución aprovechando otra de las opciones de *rsnapshot*:

```
no_create_root 1
```

Con eso, *rsnapshot* no creará la raíz en caso de que ésta no exista, algo que nos vendrá muy bien ante eventuales problemas con el montaje del disco externo.

En el caso de los Linux, como es el nuestro con Debian, *rsnapshot* puede aprovechar la potencia de los comandos incluidos con el sistema, pero para ello debemos asegurarnos de que las líneas correspondientes estén descomentadas en el archivo de configuración:

```
cmd_cp          /bin/cp
cmd_rm          /bin/rm
cmd_rsync      /usr/bin/rsync
cmd_ssh        /usr/bin/ssh
cmd_logger     /usr/bin/logger
cmd_du         /usr/bin/du
```

Dependiendo de cómo se haya instalado *rsnapshot* en el sistema la ruta a ***rsnapshot-diff*** variará, pero el paquete *debian* incluido en los repositorios oficiales de la distro lo coloca bajo ***/usr/bin***:

```
cmd_rsnapshot_diff  /usr/bin/rsnapshot-diff
```

Lo siguiente será especificar los intervalos para los distintos tipos de copia de seguridad, con ello se consigue que *rsnapshot* elimine automáticamente las copias que sobrepasen una determinada cantidad. Así, con estas líneas conservará un máximo de cincuenta copias.

```
interval        hourly 50
```

Nos interesa además tener un registro con la salida del programa, que se escribirá en el archivo que especifiquemos.

```
logfile /var/log/rsnapshot
```

Dependiendo de la versión de ***du*** de la que se disponga en el sistema (que en nuestro caso es la de Debian), y también de las preferencias personales en cuanto a la visualización de su salida,

podremos pasar unos parámetros u otros a este comando. En este caso le pasamos las opciones -c, -s y -h:

```
du_args -csh
```

Y por último hay que especificar qué partes del sistema de archivos se van a respaldar y dónde se guardará la copia. Con la siguiente línea le decimos a rsnapshot que haga copia del directorio */home* y que lo guarde en el directorio */eolo/*. Los valores deben ir separados por tabulaciones, de lo contrario, si los separamos con espacios recibiremos un error al ejecutar *rsnapshot*. El último valor conviene que sea un nombre que describa la máquina de la que se está haciendo la copia (*rsnapshot* permite hacer copias de respaldo de equipos remotos sin ningún tipo de problema, no olvidemos que la herramienta subyacente es *rsync*).

```
backup /etc /eolo
```

Ya está configurado rsnapshot y listo para ejecutarse, tan sólo falta montar el disco. Antes podemos comprobar si hay algún error sintáctico en el archivo de configuración de rsnapshot mediante el comando

```
rsnapshot configtest
```

2.3 Paso 3: Configurar los Permisos

Llegados a este punto ya podemos montar el disco y comenzar a hacer copias de seguridad incrementales del directorio */etc*. Para ello (suponiendo que existiese el directorio de destino) bastaría con ejecutar

```
sudo rsnapshot hourly
```

Pero lo que queremos no es eso, ¿verdad? Queremos conseguir que el proceso de copia se inicie automáticamente al conectar el disco externo, sin que requiera ninguna intervención manual adicional. Para ello hay que preparar algunas cosas antes.

Rsnapshot requiere que sea ejecutado con permisos de *root*, pero *halevt* ejecuta todo con los permisos del usuario *halevt* del sistema. ¿Cómo podríamos hacer que *halevt* ejecutase *rsnapshot* con permisos de *root*? Muy sencillo, con el todopoderoso *sudo*. Las ventajas de *sudo* son muchas, pero hay que tener cuidado con cómo se usa. Para lo que estamos haciendo nos vendrá muy bien, pero tampoco queremos que *halevt* pueda hacer cualquier cosa en el sistema con privilegios de *root*. Editamos entonces el archivo */etc/sudoers*, mediante el comando *visudo* (que es mucho más seguro que editar el archivo directamente con un editor de textos), como *root* evidentemente, para añadir la siguiente línea de modo que el usuario *halevt* pueda ejecutar el comando *sudo rsnapshot* (y sólo *rsnapshot*) sin tener que proporcionar una contraseña:

```
halevt        eolo= NOPASSWD: /usr/bin/rsnapshot
```

2.4 Paso 4: Identificar el disco

Para que *halevt* pueda saber con qué disco iniciar el proceso de copia de seguridad hay que proporcionarle el UUID de la partición de destino de la copia. Podemos averiguar dicho UUID de al menos dos formas. Primero conectamos el disco al sistema. Pasados unos segundos, el comando

```
dmesg | tail -25
```

deberá mostrar algo parecido a la salida de la Figura 2.

```
[52338.796021] usb 2-2: new high speed USB device using ehci_hcd and address 6
[52338.933706] usb 2-2: New USB device found, idVendor=040d, idProduct=6208
[52338.933710] usb 2-2: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[52338.933712] usb 2-2: Product: USB 2.0 SATA Bridge
[52338.933714] usb 2-2: Manufacturer: VIA Technologies Inc.
[52338.933715] usb 2-2: SerialNumber: 123412341234
[52338.933779] usb 2-2: configuration #1 chosen from 1 choice
[52338.936087] scsi8 : SCSI emulation for USB Mass Storage devices
[52338.936197] usb-storage: device found at 6
[52338.936199] usb-storage: waiting for device to settle before scanning
[52343.936300] usb-storage: device scan complete
[52343.937032] scsi 8:0:0:0: Direct-Access          SAMSUNG  HD103SJ           1AJ1 PQ: 0 ANSI: 2
[52343.938749] sd 8:0:0:0: [sdb] 1953525165 512-byte hardware sectors: (1.00 TB/931 GiB)
[52343.939259] sd 8:0:0:0: [sdb] Write Protect is off
[52343.939262] sd 8:0:0:0: [sdb] Mode Sense: 00 00 00 00
[52343.939264] sd 8:0:0:0: [sdb] Assuming drive cache: write through
[52343.940503] sd 8:0:0:0: [sdb] Assuming drive cache: write through
[52343.940508]   sdb: sdb1
[52343.950008] sd 8:0:0:0: [sdb] Attached SCSI disk
```

Figura 2: Salida en **dmesg** al conectar el disco externo.

Identificamos entonces la partición que usaremos, que en nuestro ejemplo es **sdb1**, y ya podemos conocer su **UUID** mediante

```
:~$ ls -l /dev/disk/by-uuid | grep 'sdb1$'  
  
lrwxrwxrwx 1 root root 10 ene 23 03:43 ad177457-5316-  
4e6e-a2f2-f03a4f16586d -> ../../sdb1
```

o también mediante (en Debian al menos) el comando

```
:~$ sudo blkid /dev/sdb1  
  
/dev/sdb1: UUID="ad177457-5316-4e6e-a2f2-f03a4f16586d"  
SEC_TYPE="ext2" TYPE="ext3"
```

Ahora sí, ya podemos pasar a configurar **halevt** para que lleve a cabo las acciones que le vamos a encomendar: montar el disco al detectar que ha sido conectado al equipo, ejecutar **rsnapshot** y desmontar el disco al finalizar.

2.5 Paso 5: Configuración de halevt

La configuración de **halevt** se realiza en el archivo **/etc/halevt/halevt.xml**. Este archivo ya incluye una configuración funcional de la que sólo nos quedaremos con tres líneas, por lo que comenzaremos con un archivo en blanco.

El formato del archivo es XML. Las declaraciones irán todas dentro del elemento **halevt.Configuration**:

```
<?xml version="1.0" encoding="UTF-8"?>

<halevt:Configuration version="0.1"
xmlns:halevt="http://www.environnement.ens.fr/perso/dumas/
halevt.html">

</halevt:Configuration>
```

Lo primero que añadiremos a ***halevt:Configuration*** será el elemento ***halevt:Device*** correspondiente a la partición del disco externo donde se almacenarán las copias de seguridad. Este elemento contendrá a su vez un elemento ***halevt:Insertion*** con el que le indicaremos a ***halevt*** que debe montar la partición al detectar que se ha conectado ésta al sistema. A través del elemento ***halevt:Insertion*** indicaremos a ***halevt*** también el punto de montaje de la partición, que en este caso será el directorio ***backups*** (el directorio padre de backups es ***/media*** de manera predeterminada):

```
<halevt:Device match="hal.block.device &
hal.block.is_volume = true & hal.volume.uuid =
ad177457-5316-4e6e-a2f2-f03a4f16586d">

    <halevt:Insertion exec="halevt-mount -u $hal.udi$ -o
sync -m 002 -p backups"/>

</halevt:Device>
```

Pero no sólo queremos que lo monte, sino que además deberá lanzar el proceso de copia de seguridad una vez montado, para luego desmontar el dispositivo al terminar la copia (o al producirse algún error que impidiese hacerla).

```
<halevt:Device match="hal.volume.uuid = ad177457-5316-4e6e-a2f2-f03a4f16586d">
    <halevt:Property name="hal.volume.is_mounted">
        <halevt:Action value="true" exec="sudo rsnapshot hourly ; halevt-umount -u $hal.udi$"/>
    </halevt:Property>
</halevt:Device>
```

Y por fin las tres líneas rescatadas del archivo de configuración original, que hacen que halevt desmonte los dispositivos al desconectarlos.

```
<halevt:Device match="hal.*">
    <halevt:Removal exec="halevt-umount -u $hal.udi$;
halevt-umount -s"/>
</halevt:Device>
```

2.6 Paso 6: Aplicar las configuraciones

Ya sólo resta reiniciar el demonio de halevt para que adopte los cambios hechos

```
/etc/init.d/halevt restart
```

3 Más información y opciones

<http://rsnapshot.org/>

<http://www.nongnu.org/halevt/>

© 2010, Grupo OPENTIA
www.opentia.com