

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

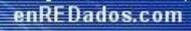
Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)

	Powered by 	Hosting Patrocinado por  
---	--	--

[Home](#) | [Quienes Somos](#) | [Empleo](#) | [Tutoriales](#) | [Contacte](#)

<p>Tutorial desarrollado por: Alberto Carrasco Montenegro</p> <p>Puedes encontrarme en Autentia Somos expertos en Java/J2EE Contacta en info@autentia.com</p>	
---	--

Descargar este documento en formato PDF [backupLinux.pdf](#)

[Firma en nuestro libro de Visitas](#)

Imperio Virtual

Networks & Design - web hosting multimedia redes dominios
www.imperiovirtual.com

Copias Seguridad 100%

Crear un clon de tu disco duro Protege tu sistema, doc, fotos, etc
www.interbel.es/productos/acronis

Anuncios Goooooogle

Anunciarse en este sitio

En [Autentia](#) nos gusta cuidar siempre los detalles fundamentales asociados al mantenimiento de cualquier sistema informático. Uno de estos aspectos consiste en realizar una buena gestión de los procesos de backup, asociados a la información que maneja cualquier sistema informático. Es fundamental programar y realizar un seguimiento de estos procesos a fin de evitar la pérdida de información relevante asociada a un sistema. Hoy os queremos mostrar cómo realizar backup de información contenida en un sistema Linux utilizando recursos compartidos en red.

1. Aspectos básicos sobre backup

El término *backup* se refiere a un concepto de especial importancia en el mundo informático: las copias de seguridad. Suele realizarse una copia de seguridad de aquella información que interesa protegerse frente a posibles pérdidas o borrados accidentales. Puede realizarse una copia de seguridad de la imagen de un sistema, bases de datos, ficheros de aplicación, documentos específicos, etc.

En su versión más sencilla una copia de seguridad, como su propio nombre indica, no es más que un duplicado de cierta información, sin ningún tipo de procesamiento sobre la misma (compresión, etc). Dicho duplicado debe almacenarse en un medio distinto al que se encuentra la información original, ya que si hubiera algún problema sobre el medio donde se encuentra almacenada la información original (rotura, robo, fallo software, etc) el *backup* realizado no serviría para nada.

Existen diversos tipos de *backup* que típicamente suelen aplicarse sobre la información relevante. Los principales son:

- **Completo:** Consiste en una copia completa de todos los datos principales. Requiere mayor espacio de almacenamiento y ventana de *backup*.
- **Incremental:** Consiste en copiar únicamente aquellos datos que hayan sido modificados respecto a otro *backup* incremental anterior, o bien, un *backup* total si éste no existe. Un *backup* incremental no sustituye a los *backup* incrementales anteriores. Para restaurar un *backup* incremental es necesario restaurar el *backup* completo y todos los *backup* incrementales por orden cronológico que estén implicados. Si se pierde uno de los *backup* incrementales, no es posible restaurar una copia exacta de los datos originales.
- **Diferencial:** Consiste en copiar únicamente aquellos datos que hayan sido modificados respecto a un *backup* completo anterior. Requiere menor espacio de almacenamiento y ventana de *backup*. Para restaurar un *backup* diferencial es necesario restaurar previamente el *backup* completo en el que se basa. Por tanto, requiere mayor tiempo de restauración. Un *backup* diferencial puede sustituir a otro *backup* diferencial más antiguo sobre el mismo *backup* completo.

Normalmente se utilizan herramientas que permiten flexibilizar y potenciar el proceso de generación y mantenimiento de copias de seguridad. Dichas herramientas suelen permitir diversas opciones relacionadas con las tareas de *backup*, tales como:

- Elección de tipo de *backup* a realizar.
- Programación de tareas de *backup*.
- Reestablecer información a partir de copias de seguridad.
- Compresión de la información, optimización del espacio ocupado por las copias de seguridad.

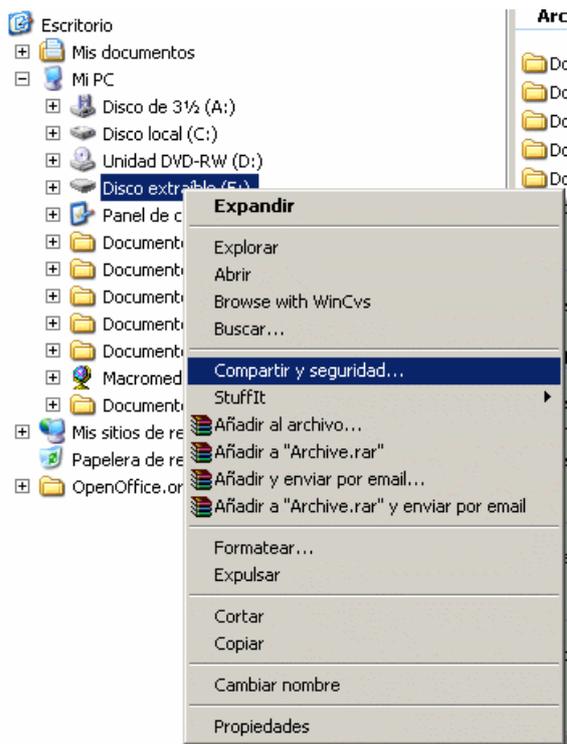
A continuación se mostrará como programar de forma sencilla tareas de *backup* en un sistema Linux, utilizando recursos compartidos en red como medio de almacenamiento de las copias de seguridad.

2. Configuración del sistema

Para ilustrar una forma sencilla de generar copias de seguridad desde un sistema Linux, se parte de un sistema Linux (*Debian*) que está conectado en red a una máquina con *Windows Server 2003*.

El medio de almacenamiento para las copias de seguridad será un disco duro extraíble que se conectará mediante usb a la máquina con *Windows Server 2003*. La forma de compartir en red un recurso en sistemas Windows es muy sencilla: una vez que el sistema

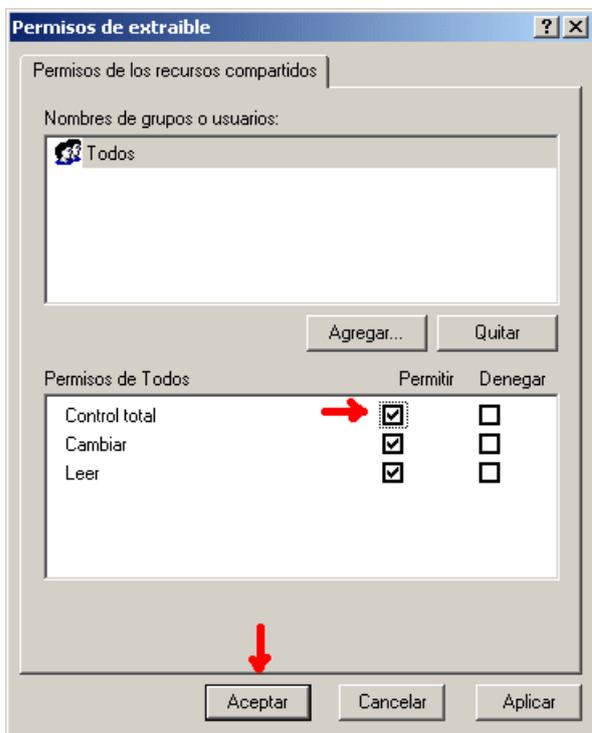
reconoce el nuevo medio conectado, basta con pulsar el botón derecho del ratón sobre el medio a compartir, y seleccionar la opción *Compartir y seguridad*.



En el diálogo emergente seleccionar la opción *Compartir esta carpeta*. Establecer el nombre deseado para el recurso compartido en el campo *Recurso compartido*.



Por último, habilitar los permisos para poder escribir sobre este medio. Pulsar el botón *Permisos* e indicar los permisos pertinentes. Una vez seleccionados los permisos (*Control total* en este caso, para poder leer y escribir) pulsar el botón *Aceptar* para finalizar el proceso.



El disco duro extraíble conectado a la máquina con *Windows Server 2003* ya está compartido en red, y puede ser accedido por la máquina Linux para almacenar las copias de seguridad.

De esta forma, la estructura de conexiones del sistema con el que se trabajará sera la siguiente:



3. Montar la unidad extraíble desde Linux

En primer lugar, se procederá a instalar sobre la máquina Linux un cliente de Samba, a fin de poder visualizar los recursos de la máquina Windows accesibles. Para ello se teclea el siguiente comando:

```
apt-get install smbclient
```

Una vez instalado el cliente de Samba, se podrán visualizar los recursos de la máquina Windows accesibles por la máquina Linux. Para ello, basta con teclear el siguiente comando:

```
smbclient -L [IP_MAQUINA_WINDOWS]
```

En este caso, mostrará el disco extraíble que se compartió en red anteriormente.

Sharename	Type	Comment
extraíble	Disk	

Para montar via red la unidad extraible de la máquina Windows sobre la máquina Linux, es necesario instalar la herramienta *smbfs*, invocando el siguiente comando en la máquina Linux:

```
apt-get install smbfs
```

Una vez instalada la herramienta *smbfs*, ya se puede montar el disco extraible de la máquina Windows en la máquina Linux utilizando la red. Antes de invocar el comando correspondiente, debe crearse un directorio en el directorio */mnt* donde se montarán los archivos del disco extraible. Para ello, teclear el siguiente comando en la máquina Linux:

```
mkdir /mnt/iomega
```

Una vez creado este directorio, ya se puede invocar el comando *smbmount* para montar el disco extraible conectado a la máquina Windows sobre el directorio */mnt/iomega* de la máquina Linux. Ejecutar este comando como se indica a continuación:

```
smbmount //[IP_MAQUINA_WINDOWS]/extraible /mnt/iomega
```

A partir de este momento se puede acceder al disco extraible conectado a la máquina Windows desde el directorio */mnt/iomega* de la máquina Linux. En este caso, el acceso será necesario para almacenar las copias de seguridad que se procederán a programar.

4. Implementar script para realizar la copia de seguridad

El siguiente paso será realizar un *script* para que sea invocado por una tarea cron de forma periódica.

La funcionalidad del *script* es muy sencilla: simplemente debe comprimir los archivos a salvar (por ejemplo, con *tar*) y copiarlo al medio de almacenamiento elegido (disco extraible del equipo con Windows montado en el directorio */mnt/iomega* de la máquina Windows). Por mayor claridad, en el nombre del fichero se especificará la fecha de realización de la copia de seguridad.

En este caso, el *script* para Linux podría contener la siguiente línea:

```
tar -cvzf /mnt/iomega/backup/backup-`date +%Y-%m-%d`.tgz /home/alberto
```

Este comando comprimirá en un fichero de nombre *backup-YYYY-MM-DD.tgz* el contenido de la carpeta */home/alberto* de la máquina Linux. El directorio destino será el directorio de nombre *backup* de la unidad extraible conectada a la máquina Windows, que se encuentra montada en el directorio */mnt/iomega* de la máquina Linux.

Este *script* se almacenará en el fichero */home/alberto/backup.sh*. Este *script* es el que se invocará desde el proceso de cron que se programará en el apartado siguiente.

5. Automatizar la tarea utilizando Cron

Para automatizar la ejecución de este *script* periódicamente, se utilizará el cron del sistema Linux. Se programará la ejecución de dicho *script* todos los jueves a las 0:00. Para establecer esta programación a la lista de tareas del cron se ejecuta el siguiente comando:

```
crontab -e
```

En este caso, la máquina Linux con Debian arranca el editor *vi* para añadir la nueva tarea programada. Se recuerda que la sintaxis de la línea a añadir en este archivo para programar una nueva tarea es:

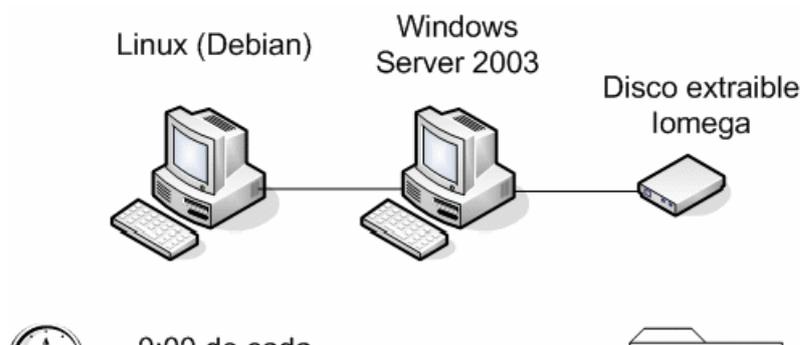
```
[hora] [minuto] [dia_del_mes] [mes] [dia_de_la_semana] [comando]
```

De esta forma, para programar la tarea de *backup* todos los jueves a las 0:00, la línea a añadir en el archivo abierto por *vi* con las tareas de cron asociadas al usuario, deberá ser la siguiente:

```
0 0 * * 4 sh /home/alberto/backup.sh
```

El símbolo asterisco (***) significa que abarcará todos los días, horas, etc según el campo que se especifique.

En este punto ya se tiene programada una tarea de *backup* desde la máquina Linux, utilizando como medio de almacenamiento el disco extraible conectado a la máquina Windows (compartido en red). El proceso de *backup* se resume en el siguiente gráfico:



Para recuperar la información, bastará con descomprimir el archivo de *backup* deseado utilizando el comando *tar* en la máquina Linux. Para ello, tomando el fichero de *backup* generado en el ejemplo del gráfico anterior, podría ejecutarse la siguiente sentencia desde el directorio donde se encuentre:

```
tar -xvzf backup-2006-04-14.tgz
```

6. Conclusiones

No debería ser demasiado complicado para cualquier iniciado en Linux, programar una tarea de *backup* utilizando herramientas básicas del sistema (scripts, cron, etc). Tal vez no es la forma más elegante (existen herramientas específicas), pero el objetivo de este documento en gran parte era ilustrar como pueden resolverse problemas interesantes utilizando las herramientas más básicas ofrecidas por los sistemas Linux.

Es una realidad en informática, el hecho de que muchas veces es importante saber resolver problemas utilizando unos pocos conocimientos sencillos asimilados anteriormente.

7. Fuentes

- Wikipedia. Enciclopedia libre.
<http://es.wikipedia.org/wiki/Portada>
- Artículo: "Acceder a un directorio compartido por red de Linux a Windows"
<http://bulma.net/body.phtml?nIdNoticia=1950>
- Artículo: "Tareas automatizadas con Cron"
<http://linuca.org/body.phtml?nIdNoticia=256>



[Puedes opinar sobre este tutorial aquí](#)

Recuerda

que el personal de [Autentia](#) te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#))

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?

¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

info@autentia.com

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos

Autentia = Soporte a Desarrollo & Formación

Gestión de contenidos

[Autentia S.L.](#) Somos expertos en:
J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ..
y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	<input type="text"/>
	<input type="button" value="Enviar"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto	Descripción
Backup en Windows	Os mostramos como realizar tareas de backup en Windows XP utilizando la sencilla herramienta que trae el propio sistema operativo
Introducción a flex	Os mostramos como utilizar la herramienta flex a partir de la resolución de un problema de conversión de formatos entre ficheros
Introducción a OCR	En este tutorial os mostramos los fundamentos de la tecnología OCR y cómo utilizar dos herramientas relacionadas: las librerías de Asprise y GOOCR.
Instalación de Virtual PC	Os mostramos como instalar Virtual PC, así como instalar un sistema operativo sobre el mismo
mod_jk en Linux / Apache2-JBoss	El conector mod_jk se encarga de enviar las peticiones dinámicas de Apache2 a un servidor de aplicaciones JBoss
Recordar primeros pasos por Linux	En este tutorial recordamos, de un modo visual, los primeros comandos en Linux sobre una emulación Qemu.
Validar en JSF con Commons Validator	En este nuevo tutorial sobre el framework JSF os mostramos como utilizar y extender la validación del Commons Validator
Compartir impresoras y ficheros con Linux	Cesar Crespo Martín y Alejandro Perez García nos enseñan como realizar la compartición de impresoras y ficheros con Linux, CUPS y SAMBA con clientes Windows.
Compiar en C y Linux con make (makefile)	Os recordamos como construir ficheros makefile en entorno Linux para compilar vuestros programas C
Wireless Centrino en Linux Debian	En este tutorial, Alejandro Perez nos enseña como configurar la red Wireless WIFI de tu centrino en Linux (Debian)

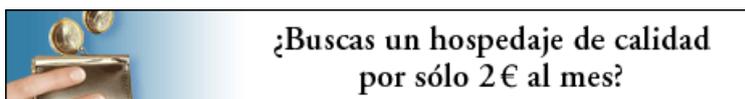
Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)



www.AdictosAlTrabajo.com Optimizado 800X600