

Backupeando la configuración de Linux

por [Dardo A.](#)

Siempre testeando, probando software nuevo, distribuciones nuevas, poniendo a prueba la resistencia de nuestro Linux configurado durante mucho tiempo, hasta que no dé más. Y luego empezar desde cero...otra vez. Formatear, Borrar, Instalar, reinstalar y volver a configurar todo (!) ¿Configurar todo otra vez dije? Linux es flexible dicen en todos lados, pero hay que saber por donde se dobla para aprovecharlo y salvar horas. El sistema operativo del Pinguino te puede ahorrar largas horas de aburrida reconfiguración con un poco de sentido común y sencillos backups.

Introducción

Este artículo trata de explicar un procedimiento práctico para realizar un backup de la configuración (no de TODO el sistema, sólo de su configuración), de un sistema Linux Workstation.

El uso eficiente de Linux suele ser difícil en primera instancia debido a la falta de conocimiento de la filosofía de trabajo en sistemas tipo *nix (todos los basados en Unix). Muchos de Uds. seguramente instalan y reinstalan Linux frecuentemente por diversos motivos, pero muy pocos resguardan los datos y configuraciones útiles de una instalación a otra. Esto hace que tengan que perder a veces tiempo y esfuerzo considerable en llevar una instalación nueva al mismo nivel de funcionalidad que tenía la anterior.

Al ser Linux un entorno con la filosofía *nix, es sumamente configurable. Pero al tener muchas opciones de configuración, generalmente se las va conociendo a medida que se las necesita. A menos que uno tenga una gran memoria difícilmente recuerde todos los parámetros y configuraciones que hizo en una instalación para aplicarlos a una nueva... y aún así está el factor tiempo: podes rehacerlo todo, pero te va a llevar casi seguro la misma cantidad de tiempo (tal vez un poco menos) que te costó la primera vez que configuraste el sistema.

Aparte en Linux no solo se configura el sistema, también están las aplicaciones: determinados tipos de letras, tamaños de ventana, los themes de aplicaciones y Window Managers, su comportamiento, los íconos personalizados que hacen tal o cual cosa, etc., etc.

Algunos acotarán que sólo usan una distribución y van de actualización en actualización, el problema es que no siempre las actualizaciones funcionan adecuadamente, es más, si se posee un entorno con configuraciones muy personalizadas es posible que la actualización no se realice exitosamente al 100% teniendo que hacer ajustes a mano luego. Aquí también corremos el riesgo de que mágicamente nuestras personalizaciones y configuraciones particulares vuelen luego de la actualización y tengamos que rehacerlas posteriormente.

Sería razonable encontrar una manera de salvar y/o resguardar los datos y configuraciones más importantes de una instalación Linux para reusarlos lo más rápidamente posible la próxima vez que reinstalemos el sistema o cambiemos de distribución.

Backupeando nuestra configuración

Primero analicemos dónde se realizan los cambios al configurar y personalizar nuestro Linux y qué se guarda en cada uno. Básicamente en dos lugares:

- En nuestro directorio de usuario (por ejemplo:/home/usuario) y en el directorio del superusuario (/root): donde se sitúan el 90% de las personalizaciones y configuraciones particulares que realizamos a los programas que usamos. Tomemos como ejemplo los íconos que personalizamos en GNOME o KDE, u otro Window Manager; también están nuestros themes, etc.
- En el directorio /etc : donde se sitúan el 100% de las configuraciones que realizamos al sistema, o propiamente dicho a su funcionamiento. Tómese como ejemplo las unidades y sus puntos de montaje (archivo fstab), el nivel de inicio con o sin interfase gráfica (inittab), el mensaje del sistema a cada usuario que se logea (motd), los servicios que queremos ejecutar (en /init.d, dentro del respectivo runlevel), comandos personalizados de inicio (rc.local), el nombre del sistema (hostname), módulos del kernel que se cargan para activar hardware como placas de sonido y de red (conf.modules), etc.

Para hacer un backup de la configuración de un sistema Linux es necesario guardar en un lugar seguro los directorios nombrados con todo su contenido (subdirectorios y carpetas, más detalles sobre esto luego).

Los directorios a copiar (o backupear mejor dicho) son entonces:

```
/etc  
/home  
/root
```

Es necesario copiarlos por completo, incluyendo los directorios y archivos ocultos (son los que tienen el punto delante del nombre), para hacer la copia es recomendable usar el Midnight Commander (mc) con la opción de copia que aparece por defecto al presionar F5 (Copiar), pues de esta manera conservamos permisos y propietarios de los directorios y archivos evitando futuros problemas de acceso a ellos.

Restaurando la configuración

El backup se realiza con el obvio propósito de usarlo para tener una instalación nueva, por lo general funcionando con un nivel alto de personalizaciones y configuraciones particulares minutos luego de finalizada la instalación o actualización, o en un tiempo mucho menor de lo que tomaría resetear desde cero toda la configuración que necesitamos o que usamos habitualmente.

El escenario ideal de restauración (y en el que destaca en rapidez este tipo de backups) es el de una instalación backupeada de cierta distro que se restaura hacia una instalación nueva de la misma distro (igual nombre y número de versión) realizada en la misma máquina de donde se tomó el

backup. Para este caso la precaución más importante por lejos es fijarse si la distribución usada es la misma (por ejemplo Linux Mandrake 7.2). Consideraremos éste como el Primer Caso.

El siguiente escenario de restauración añade dificultades como:

- Distribución Linux distinta a la backupeada (puede ser otra distro o una nueva versión de la que backupeamos).
- Máquina (hardware) distinto al original de donde se realizó el backup.

La combinación de estos dos factores (que consideraremos como el Segundo Caso), origina algunas precauciones y consideraciones adicionales, y no permite la directa restauración del backup hacia la instalación nueva.

El procedimiento para el Primer Caso

Tenemos que entrar en modo texto (en una consola), y usando el Midnight Commander (mc), sobrescribimos los directorios /etc, /home y /root con los directorios de igual nombre del backup, y luego hacemos un logout, para posteriormente reingresar al sistema. Y... listo! Seteos recuperados en minutos!

Si el backup fuera restaurado hacia una instalación nueva de la misma distro, pero instalada en una máquina distinta serán necesarios ajustes posteriores, y es importante evitar restaurar los siguientes archivos:

- /etc/lilo.conf, que setea LILO, el booteador por excelencia de muchas distros.
- /etc/fstab, pues éste setea las particiones del disco y sus respectivos puntos de montaje.
- /etc/modules.conf, que setea los módulos a cargar según tu hardware.
- /etc/isapnp.conf, que setea el funcionamiento de placas ISA Plug-and-Play.
- /etc/X11/XF86Config
- , que setea el entorno X Windows.
- /etc/cups, si existe, regula el funcionamiento del nuevo sistema de impresión de Linux (CUPS: Common Unix Printing System).
- /etc/printcap, si tenés alguna distro no tan nueva que no usa CUPS.

Faltarían señalar algunos archivos similares que configuran los seteos de placas SCSI, ciertos tipos no comunes de lectoras y dispositivos externos como lectoras ZIP. Sugerencia sobre estos son bienvenidos en los [Comentarios](#). Los dispositivos que tengas y se desconfiguren los tendrás que reseatear usando las utilidades que traiga la distro que usás.

En general sobre esto último, cada uno de nosotros suele saber (por experiencia) qué archivo configura cierto dispositivo, periférico, etc., particularmente difícil de setear y que no nos gustaría desconfigurar, bueno ese no lo borren ni sobrescriban!

El procedimiento para el Segundo Caso

El caso más difícil es en el que tenemos un backup de cierta distro, por ejemplo Red Hat 6.0, y restauramos hacia una instalación nueva como Mandrake 7.2. En éste caso se pueden dar diversos inconvenientes derivados mayormente de que las mejoras de las versiones nuevas de muchos programas y de los cambios y diferencias en los scripts de inicialización del sistema y configuración particular de una distro con

respecto a la otra. Aún así...se pueden restaurar del backup ciertos archivos y directorios siguiendo algunos consejos y precauciones.

También en éste caso existe la complicación de pasar de una máquina a otra (complicado más por el cambio de versión y/o de distro), por lo que hay que tomar precauciones similares al caso anterior.

Si son un poco jugados pueden intentar hacer un backup del /etc dejado por la instalación y empezar a reemplazar archivos para ver si funciona o no, sin tocar el /etc/init.d donde están los scripts de inicialización. Esto puede ser interesante, y/o funcionar, y/o ser bastante destructivo (pero en todo caso vas a salir sabiendo más de lo que sabías cuando comenzaste).

Con respecto a /home (el directorio de los usuarios), el 99% de las restauraciones (sobreescribiendo los directorios dejados por la instalación), de una instalación a otra independientemente de la distro o versión que se use suelen ser exitosas, excepto que los sistemas estén excesivamente alejados en versión o tipo: como ir de un Mandrake 5.5 a un Debian Woody.

El otro pero es que a veces de una versión a otra de una distro o de distros distintas ciertos programas (se dió con Sawfish y GNOME últimamente) cambian mucho su modo de configuración y los archivos anteriores no permiten funcionar apropiadamente a ese programa, entonces la solución es borrar la configuración de lo que no funcionó, reiniciar el programa para que se graben las configuraciones por defecto y personalizar otra vez a nuestro gusto.

Esta es una lista básica de archivos que sería posible restaurar limpiamente o casi (previa lectura comparativa con su igual originado por la instalación):

En /etc

- /etc/bashrc, configuración del shell Bash para todo el sistema
- /etc/profile, entorno del sistema y algunos programas de inicio
- /etc/HOSTNAME, nombre de la PC
- /etc/motd,
- /etc/anacrontab, configuración de anacron
- /etc/fstab, tabla con los puntos de montaje de las particiones del disco
- /etc/rc.d/rc.local, (de aquí solo tomen las líneas que Uds. agregaron al final de lo que ya tenía este archivo, si no agregaron nada no es necesario restaurar, es redundante decirlo, pero es mejor hacerlo)

Si tuvieron el placer de personalizar (en /etc/rc.d), algún runlevel y/o incluso crearon uno/s propio/s pueden verificar si los demonios que activa y/o desactiva tienen igual nombre en la nueva distro y ajustar las diferencias. Si no modificaste ningún runlevel y/o no entendiste este comentario entonces NO tienes que restaurar nada del backup hacia /etc/rc.d

Sobre el directorio /etc/X11 cabe hacer el mismo comentario que sobre el directorio anterior, al ser un directorio de configuración central del sistema X Windows y algunas de sus aplicaciones principales, si personalizamos alguno de sus directorios y/o archivos, seguro sabremos si es apropiado o no restaurarlos.

El archivo /etc/X11/XF86Config, si la versión de X Windows de la nueva distro es la misma que la original del backup, entonces podemos restaurarla

directamente. Si se da el caso (como ultimamente suele ocurrir) que estamos ante un salto de version muy pequeño (XFree86 4.0 a 4.01 o 4.02), el archivo de configuración casi seguro funcionará para nuestra nueva distro, pero seria mejor configurar X Windows nuevamente y comparar con nuestro viejo XF86Config backupado para ver si es que no hay alguna mejora que no tengamos en el XF86Config backupado o viceversa.

Bueno y seguro que quedan algunas cosas que no tuve el placer de restaurar y que funcionen, por acepto sugerencias y [comentarios](#), sobre todo en el tema de redes y demonios de redes especialmente.

En /home/usuario y /root

- /.gnome, configuraciones del usuario para GNOME
- /.kde, configuraciones del usuario para KDE
- /.netscape, importante pues aquí están tus bookmarks y otras configuraciones que haces si navegas con Linux.
- /.sawmill, aquí están los seteos particulares de este window manager, que al presentar tantas opciones puede tomar su tiempo configurarlo a nuestro gusto.
- /nsmail, nuestras configuraciones de correo electronico (!)
- .bash_profile, variables de entorno personales para el shell Bash
- .bashrc, más variables
- .cshrc, variables de entorno para este otro shell
- .tchrc, variables de entorno para otro shell

Nuevamente los insto a que [recomienden](#) y/o corrijan sobre otros directorios y/o archivos que sea posibles restaurar en este caso.

Aclaraciones y Precauciones importantes

- Es necesario aclarar, para muchos que recién empiezan, que el 100% de los archivos que conforman la configuración del sistema y de los usuarios son archivos de texto modificables con cualquier editor de texto disponible, no hay ningún registro al estilo Win9x, ni nada parecido.
- Con respecto a con qué copiamos los tres directorios del backup, podríamos usar la línea de comando, pero hay que tener en cuenta dos cosas: copiar TODOS los subdirectorios y archivos (ocultos incluidos) y conservar sus respectivos permisos y propietarios. Con esto, la línea de comando puede ser engorrosa. Otra vez, recomiendo usar el Midnight Commander (mc), copiar con F5, y copiar marcando la opción Copiar conservando permisos y dueños (puede estar en inglés) los directorios:

```
/etc  
/home, y  
/root
```

- Con respecto a /home y /root, estos contienen directorios y archivos ocultos (los que tienen un punto delante del nombre), algunos finalizan con rc (run control: control de ejecución) y son los que setean el comportamiento de ciertos programas. Algunos ejemplos de directorios y archivos ocultos:

```
/root/.kmail  
/root/.nsmail  
/root/.bashrc
```

Es vital copiar todos los archivos de /home y /root incluidos directorios y archivos ocultos pues en ellos se guardan por lo general los seteos de programas y entornos que usamos en Linux.

- Si alguien quiere hacer un backup en CD o en Zip (el disco) es necesario grabar los datos desde Linux, pues si los copiamos hacia una partición FAT32 (Win9x), cambiarán los permisos (se activarán los permisos de ejecución) y hará el backup inutilizable para una restauración directa; pues habría que setear TODOS los permisos de TODOS los archivos, lo que no es muy rápido ni simple de hacer.
- Algunos, muy pocos, programas no respetan el modelo de guardar datos en los 3 directorios a backupear (aunque no necesariamente en los 3 al mismo tiempo), y guardan configuraciones de usuarios y/o programas como configuraciones globales que luego tendrán todos los demás usuarios, un ejemplo (el único que encontré) es el instalador de themes del Panel de Control de GNOME, que al instalar un tema nuevo lo graba en /usr/share/themes, haciéndolo disponible a todos los usuarios, pero si uno quiere borrarlo no puede a menos que sea root. Algo como esto también se dá con el XMMS, al que para instalarle plugins hace falta grabarlos en /usr/lib/xmms/plugins y se debe ser root para eso.

Conclusión

Primero unas palabras para los que llegaron hasta aquí y les duele un poco la cabeza: Es más fácil hacerlo que explicarlo .

Como se ve, con cierta precaución y sentido común se puede ahorrar mucho tiempo y esfuerzo en las reinstalaciones desde cero, en los cambios de distribución, y en las actualizaciones de sistemas Linux.

En el caso de estar restaurando una distro sobre una instalación nueva de ella misma y con igual número de versión, y sobre la misma máquina desde donde se hizo el backup, obviamente el tiempo ahorrado es impresionante, pues se puede tener funcionando y configurada por completo esa instalación nueva en minutos. Con un poco de cuidado, estas líneas le pueden servir a aquellos que instalan, prueban (rompen) y vuelven a instalar su sistema para empezar desde cero. Espero que acorten sus tiempos entre experimento y experimento para llegar a dominar Linux con mayor rapidez.

Esto es también una apreciación acerca de la practicidad de la filosofía de sistemas tipo *nix, que permite un importante ahorro de tiempo y recursos teniendo un sistema nuevo funcionando a nivel óptimo en un tiempo mínimo, bajando el tiempo que pasa entre salir de funcionamiento y volver al mismo de estado en que estaba a pleno rendimiento.

Como conclusión se puede ver que es importante racionalizar (entender, comprender) la forma de trabajo que se tiene en sistemas tipo *nix y su diferencia en enfoque con los tipo Win*, apreciando que se puede ser más productivo si se conoce la primer filosofía de trabajo en vez de aplicar comparativamente las formas de proceder estándar de un sistema Win*, que consiste en borrar, reinstalar y reconfigurar masivamente prácticamente por completo todo el entorno, con un tiempo de reconfiguración bastante alto. Esto último es por demás innecesario en Linux y en general en todo sistema tipo *nix.

Dardo A.
alias «Yaco»
yacolinux@planetainux.com.ar

URL: <http://www.planetainux.com.ar/article.php?aid=42>