Hard Disk Upgrade Mini HOWTO

di Yves Bellefeuille (*yan@storm.ca*) e Konrad Hinsen (*hinsen@cnrs-orleans.fr*) Versione 2.1, 20 ottobre 1999

Come copiare un sistema Linux da un disco fisso ad un altro.

Indice

1 Introduzione

Questo documento spiega come trasferire o migrare, un intero sistema Linux, incluso LILO, da un disco fisso ad un altro.

Nella spiegazione che segue "`/dev/hda`" (primo disco fisso IDE) indica il "vecchio" disco e "`/dev/hdb`" (secondo disco fisso IDE) indica il "nuovo" disco.

Ci si riferisce alle partizioni specifiche nel "vecchio" disco come "`/dev/hda1`", "`/dev/hda2`", e così via. Ci si riferisce alle partizioni specifiche del "nuovo" disco come "`/dev/hdb1`", "`/dev/hdb2`", e così via.

Le spiegazioni in questo documento sono basate su **Red Hat 6.0**. Sono state verificate anche con **Debian 2.1** **Slackware 3.5** e **SuSE 6.2**; indichiamo alcune differenze da notare se state usando quelle distribuzioni.

Se i comandi non funzionano a dovere nel vostro sistema, per favore fateci sapere, dicendoci che distribuzione state usando.

2 Installare entrambi i dischi nel vostro sistema

I sistemi moderni possono accettare quattro dispositivi `"EIDE"` nel controller del disco fisso, così non ci dovrebbero essere problemi per installare entrambi i dischi nel vostro sistema allo stesso tempo, anche se avete già altri dispositivi `EIDE`. I dischi fissi e le unità CD-ROM sono tipicamente dispositivi `EIDE`. Le unità floppy e nastro sono di solito connesse al controller dell'unità floppy anziché al controller del disco rigido.

Gli adattatori SCSI sono ancora più flessibili e possono accettare sette dispositivi. Se siete abbastanza fortunati (e ricchi) da avere un adattatore SCSI, probabilmente già lo sapete, e probabilmente sapete quale dei vostri dispositivi sono SCSI! Per maggiori informazioni consultate lo *SCSI How-To*.

Anche i più vecchi sistemi possono accettare due dispositivi nel controller del disco rigido, così potete ancora installare entrambi i dischi fissi allo stesso tempo. Comunque, se avete già un altro dispositivo installato in aggiunta al disco fisso (per esempio, se avete sia un disco fisso che un'unità CD-ROM), dovrete rimuovere l'altro dispositivo per essere in grado di installare il vecchio e il nuovo disco fisso contemporaneamente.

Dovete configurare i dischi come `"master"` o `"slave"` installando i jumper in modo appropriato. Troverete spesso informazioni sulla configurazione sui dischi stessi, altrimenti consultate i manuali o i fabbricanti dei dischi.

Dovete anche informare il `BIOS` della presenza dei dischi e della loro "*geometria*". Solitamente per eseguire questa operazione si accede al setup del BIOS premendo un tasto durante l'avvio del sistema. Ecco cosa fare per alcuni `BIOS` comuni:

Acer notebooks: tasto**F2** durante il **Power-On Self-Test** (POST)

American Megatrends (AMI): Tasto **Canc** durante il **Power-On Self-Test**

Award: **Del** o **Ctrl-Alt-Esc**

Compaq: Tasto **F10** dopo che il quadrato appare nell'angolo in alto a destra dello schermo durante l'avvio.

Dell: **Ctrl-Alt-Invio**

DTK: Tasto **Esc** durante il **Power-On Self-Test**

IBM Aptiva 535: **F1** mentre il quadrato con le linee curve viene visualizzato nell'angolo in alto a destra durante l'accensione
[Funziona anche con altri modelli Aptiva?]

IBM PS/2: **Ctrl-Alt-Canc**, poi **Ctrl-Alt-Ins** quando il cursore è nell'angolo in alto a destra.

Mr. BIOS: **Ctrl-Alt-S** durante il **Power-On Self-Test**

Packard Bell: Per alcuni modelli, tasto **F1** o **F2** durante il **Power-On Self-Test**

Phoenix: **Ctrl-Alt-Esc**, or **Ctrl-Alt-S**, o **Ctrl-Alt-Enter**

(Siamo interessati a ricevere informazioni su altri BIOS per aggiungerli a questa lista.)

Riavviate il sistema e loggatevi come root. Se usate il comando `"su"` per diventare superutente, usate `"su -"`, con il trattino.

3 Smontare le partizioni non-Linux e le unità di rete

Ad alcune persone piace montare partizioni da altri sistemi operativi (*DOS*, *Windows*, *OS/2*, ecc.) così le possono usare sotto *Linux*. Queste partizioni devono essere create e copiate sotto il loro sistema operativo, e dovreste smontarle prima di copiare la vostra partizione Linux. Per esempio, se avete una partizione *DOS* montata in /dos, dovete smontarla con questo comando:

```
umount /dos
```

Notate che il comando è "`umount`", senza la prima lettera "n" nella parola "unmount".

Dovreste anche smontare le unità di rete.

4 Partizionare il nuovo disco

Usate questo comando per partizionare il nuovo disco:

```
fdisk /dev/hdb
```

I dispositivi `EIDE` sono identificati come hda, hdb, hdc e hdd nella directory */dev*. Le partizioni in questi dischi possono andare da 1 a 16 e sono anch'esse nella directory */dev*. Per esempio, `/dev/hda4` si riferisce alla quarta partizione del primo disco fisso (hda - primo disco fisso della catena `EIDE`).

I dispositivi `SCSI` sono identificati con sda, sdb, sdc, sdd, sde, sdf ed sdg nella directory */dev*. Similmente, le partizioni in questi dischi possono variare da 1 a 16 e sono anch'esse nella directory */dev*. Per esempio, `/dev/sda3` si riferisce alla partizione 3 nel disco `SCSI` a (primo disco `SCSI`).

Per le partizioni Linux con il file system ext2, usate l'ID di sistema 83. Per partizioni di swap, usate l'ID di sistema 82.

Per maggiori informazioni sul partizionamento, consultate l'*Installation How-To* e il *Partitioning Mini How-To*.

Se il vostro nuovo disco ha più di 1024 cilindri, consultate il *Large Disk Mini How-To*. In breve, dovreste installare tutti i file necessari per avviare Linux entro i primi 1024 cilindri. Un modo di farlo è di creare una piccola partizione solo per la directory `/boot` all'inizio del disco. (**Solo per Slackware**: Il kernel è in `/vmlinuz` anziché in `/boot/vmlinuz`, perciò dovreste mettere sia la directory `/` che la directory `/boot` in questa partizione)

Le partizioni per altri sistemi dovrebbero essere create con il loro `fdisk` o con loro comando equivalente anziché con l'`fdisk` di Linux.

5 Formattare il nuovo disco

Usate il seguente comando per formattare le partizioni Linux usando `ext2fs` sul nuovo disco:

```
mkfs.ext2 /dev/hdb1
```

Per controllare il disco per settori non validi (difetti fisici), aggiungete l'opzione `-c` giusto prima di "`/dev/hdb1`".

Se il nuovo disco avrà più di una partizione *Linux*, formattate le altre partizioni con "`mkfs.ext2 /dev/hdb2`", "`mkfs.ext2 /dev/hdb3`", e così via. Aggiungete l'opzione `-c` se desiderate.

(**Nota**: Con distribuzioni più vecchie, il comando "`mkfs -t ext2 -c /dev/hdb1`" non controlla i settori non validi né in *Red Hat*, né in *Debian* e neppure nella *Slackware*, contrariamente a quanto stabiliva la pagina di manuale. Questo ora è stato corretto.)

Per formattare una partizione di swap, usate questo comando:

```
mkswap /dev/hdb1
```

Ancora, potete aggiungere l'opzione `-c` prima di "`/dev/hdb1`" per controllare i settori non validi.

6 Montare il nuovo disco

Create una directory dove monterete il nuovo disco, per esempio `/new-disk`, e montatelo:

```
mkdir /new-disk
mount -t ext2 /dev/hdb1 /new-disk
```

Se il nuovo disco avrà più di una partizione Linux, montatele tutte sotto `/new-disk` con la stessa organizzazione che avranno più tardi.

Esempio: Il nuovo disco avrà quattro partizioni Linux, come segue:

```
/dev/hdb1:              /
/dev/hdb2:              /home
/dev/hdb3:              /var
/dev/hdb4:              /var/spool
```

Montate le quattro partizioni sotto `/new-disk` come segue:

```
/dev/hdb1:              /new-disk
/dev/hdb2:              /new-disk/home
/dev/hdb3:              /new-disk/var
/dev/hdb4:              /new-disk/var/spool
```

Dovete creare i mount-point per ogni "livello" prima di montare le partizioni a quel livello. Esempio:

```
mkdir /new-disk                  [Primo livello]
mount -t ext2 /dev/hdb1 /new-disk

mkdir /new-disk/home             [Secondo livello]
mount -t ext2 /dev/hdb2 /new-disk/home

mkdir /new-disk/var              [Ancora secondo livello]
mount -t ext2 /dev/hdb3 /new-disk/var

mkdir /new-disk/var/spool        [Terzo livello]
mount -t ext2 /dev/hdb4 /new-disk/var/spool
```

Se avete creato un mount point in `/new-disk/tmp`, avrete bisogno di corregere i permessi della directory per lasciare l'accesso a tutti gli utenti:

```
chmod 1777 /new-disk/tmp
```

7 Copiare i file del vecchio disco in quello nuovo

Dovete andare in modalità single-user prima di iniziare a copiare il disco, in modo da chiudere i demoni di sistema e preservare lo stato dei log, e per impedire agli utenti di loggarsi:

```
/sbin/telinit 1
```

Copiando il disco fisso, dovete copiare tutte le directory e i file, inclusi i link simbolici.

In ogni caso *non* dovete copiare la directory `/new-disk`, poiché questo copierebbe il nuovo disco stesso!

Inoltre, volete creare la directory `/proc` nel nuovo disco, ma non volete copiare il suo contenuto: `/proc` è un file system "*virtuale*" e non ha nessun file vero, ma piuttosto contiene informazioni sui processi che vengono eseguiti nel sistema.

Ecco quattro modi differenti per copiare il vecchio disco in quello nuovo. Questo può prendervi un po' di tempo, specialmente se avete un disco grande o poca memoria. Potete aspettarvi di essere in grado di copiare 10Mb al minuto, e possibilmente molto di più.

Potete seguire il progresso della copia usando il comando "`df`" da un altro terminale. Provate "`watch df`" o "`watch ls -l /new-disk`" per vedere un rapporto aggiornato ogni due secondi; premete Ctrl-C per terminare la visualizzazione. Siate consapevoli che eseguire il programma "`watch`" stesso rallenterà la copia.

1. `cp -ax / /new-disk`
   Questo è il metodo più semplice, ma funzionerà solo se il vostro sistema Linux originale è in un disco con una sola partizione.

   L'opzione `-a` conserva il file system il più possibile. L'opzione `-x` limita `cp` a un solo file system; questo è necessario per evitare di copiare le directory `/proc` e `/new-disk`.

   (**Solo per SuSE**: Solo con questo metodo, dovete anche creare la directori `/dev/pts` nel nuovo disco. Usate il comando "`mkdir /new-disk/dev/pts`.)

   (**Nota**: Usando l'opzione `-x`, le versioni recenti di `cp` creeranno le directory `/new-disk/new-disk` e `/new-disk/proc`, sebbene le directory siano vuote. Se queste directory verranno create, dovreste cancellare `/new-disk/new-disk` ma tenere `/new-disk/proc`.)

2. `cd / && echo cp -a '/bin/ls -1Ab | egrep -v"^new-disk$|^proc$"' /new-disk | sh`

   (sempre in una sola linea)

   Questo va alla directory root e poi copia tutti i file e le directory eccetto `/new-disk` e `/proc` in `/new-disk`. Notate che la prima opzione dopo `ls` è il numero 1, non la lettera L.

   Questo comando dovrebbe funzionare in ogni circostanza.

3. `(cd / && tar cpf - . -exclude new-disk -exclude proc) | (cd /new-disk && tar xpf -)`

   (sempre in una sola linea)

   Questo comando si sposta sulla directory root, comprime tutto eccetto `/new-disk` e `/proc`, passa a `/new-disk` e decomprime tutto. Notate che non ci devono essere slash prima o dopo i nomi delle directory nell'opzione `-exclude`.

   (**Nota**: L'opzione `-l` non funziona qui, poiché tar ricreerà ancora le directory `/new-disk` e `/proc` anche se non copia il loro contenuto. Inoltre, l'opzione `-l` di *tar* non ha lo stesso comportamento dell'opzione `-x` di vecchie versioni di *cp*.)

   Questo metodo è parecchio più lento degli altri.

4. `cp -a /bin /boot /dev /etc /home /lib /lost+found /mnt /root /sbin /tmp /usr /var /new-disk`

   (sempre in una sola linea)

   L'ultima directory, `/new-disk`, è la destinazione per il comando *cp*. Tutte le altre directory sono la sorgente. Perciò, stiamo copiando tutte le directory elencate in `/new-disk`.

   Con questo metodo elencate semplicemente da soli le directory che volete copiare. Qui listiamo tutte le mie directory tranne `/new-disk` e `/proc`. Se non potete usare gli altri metodi per qualsiasi motivo, potete sempre usare questo comando per specificare manualmente le directory che volete copiare.

   Solo con questo metodo, se ci sono dei file nella directory root stessa, avete bisogno di un altro comando per copiarli. In particolare, questo è richiesto con **Debian** e **Slackware**, poiché queste distribuzioni mettono file nella directory root:

   ```
   cp -dp /* /.* /new-disk
   ```

Dopo aver usato uno di questi quattro metodi, dovete anche creare la directory `/proc` nel nuovo disco, se non esiste già:

```
mkdir /new-disk/proc
```

A questo punto, potete verificare la struttura dei file nel nuovo disco, se desiderate:

```
umount /new-disk
fsck.ext2 -f /dev/hdb1
mount -t ext2 /dev/hdb1 /new-disk
```

Se il disco ha più di una partizione, dovete smontarle da quella più interna in sù, prima di eseguire `fsck.ext2`: nell'esempio menzionato sopra, dovreste prima smontare le partizioni al terzo livello, poi quelle del secondo e infine quella del primo.

Potete anche confrontare i due dischi, per assicurarvi che i file sono stati copiati a dovere:

```
find / -path /proc -prune -o -path /new-disk -prune -o -xtype f \\
                -exec cmp {} /new-disk{} \;
```

(scrivete tutto questo in una sola linea)

(**Solo per Slackware**: l'installazione basica di **Slackware** (solo la serie "A") non include il comando "*cmp*", perciò non potrete eseguire questo comando se avete installato solo i file di base. Il comando "*cmp*" è nella serie "AP1".)

Questo confronterà solo i file regolari, non file "*speciali*" a caratteri o a blocci (nella directory `/dev`), socket, ecc., poiché il comando "*cmp*" non funziona bene con questi. Sarebbe benvenuto qualsiasi suggerimento su come verificare questi file "*speciali*".

8 Modificare `/new-disk/etc/fstab` in modo appropriato

Se il vostro nuovo disco non ha le stesse partizioni o la stessa organizzazione del vecchio disco, modificate il file `/new-disk/etc/fstab` nel nuovo disco in modo appropriato.

Assicuratevi che le partizioni del disco (prima colonna) corrispondono all'organizzazione che avrete con il nuovo disco, una volta che il vecchio disco è stato rimosso, e che state solo montando una partizione in "/" come è mostrato nella seconda colonna.

Per maggiori informazioni sul formato del file `/etc/fstab`, consultate la "*Linux System Administrator's Guide*", sezione 4, sotto "*Mounting and unmounting*".

Per partizioni di swap, usate una linea simile a questa:

```
/dev/hda1       swap    swap    defaults        0       0
```

9 Preparare LILO per avviare il nuovo disco

(Grazie a Rick Masters per averci aiutato con questa sezione.)

Stiamo supponendo che LILO sia installato nel Master Boot Record del disco fisso (MBR); questa sembra essere la configurazione più comune. Volete installare LILO in quello che è al presente il secondo disco fisso ma che diventerà il primo disco fisso.

Editate il file `/new-disk/etc/lilo.conf` come segue:

```
disk=/dev/hdb bios=0x80         # Dice a LILO di trattare il secondo
                                # disco come se fosse il primo
                                # (BIOS ID 0x80).
boot=/dev/hdb                   # Installa LILO sul secondo disco.
map=/new-disk/boot/map          # Posizione del "map file".
install=/new-disk/boot/boot.b   # File da copiare nel settore di avvio
                                # del disco.
prompt                          # Dice a LILO di mostrare "LILO boot:"
timeout=50                      # Avvia il sistema di default dopo 5
                                # secondi. (Il valore è in
                                # decimi di secondo.)
image=/new-disk/boot/vmlinuz    # Posizione del kernel. Il vero nome può
                                # includere un numero di versione, per
                                # esempio "vmlinuz-2.0.35".
label=linux                     # Etichetta per il sistema Linux
root=/dev/hda1                  # Posizione della partizione di root nel
                                # nuovo disco fisso. Modificarlo come
                                # appropriato al vostro sistema. Notate
                                # che dovete usare il nome della posizione
                                # futura, una volta che il vecchio disco
                                # è stato rimosso.
read-only                       # Monta subito la partizione in sola
                                # lettura, per eseguire fsck.
```

(**Solo per Slackware**: Usate "`image=/new-disk/vmlinuz`".)

Se state usando un disco fisso SCSI, potete aver bisogno di aggiungere una linea con "`initrd`". Consultate il vostro file esistente `/etc/lilo.conf`.

Installate LILO sul nuovo disco:

```
/sbin/lilo -C /new-disk/etc/lilo.conf
```

L'opzione `-C` dice a LILO quale file di configurazione usare.

10 Fare un dischetto di avvio (facoltativo)

Se desiderate, potete fare un dischetto di avvio, nel caso che vi imbattiate in qualche problema cercando di avviare il nuovo sistema.

Inserite un dischetto vuoto, formattatelo, create un file system su di esso e montatelo:

```
fdformat /dev/fd0H1440
mkfs.ext2 /dev/fd0
mount -t ext2 /dev/fd0 /mnt
```

(**Solo per Debian**: Con **Debian** 2.x, usate "`/dev/fd0u1440`" invece di "`/dev/fd0H1440`". Con **Debian** 1.x, usate "`/dev/fd0h1440`", con una "`h`" minuscola.

(**Solo per Debian**: Con **Debian** 2.x, usate "*superformat*" invece di "*fdformat*". Potete ignorare l'errore "`mformat: command not found`". Con **Debian** 1.x, se non avete il comando "*fdformat*", potete ometterlo se il floppy è già formattato. In questo caso, dovreste controllare il dischetto per settori non validi aggiungendo l'opzione "`-c`" dopo il comando "*mkfs.ext2*".)

(**Solo per Slackware**: Usate "`/dev/fd0u1440`" invece di "`/dev/fd0H1440`". Con versioni più vecchie, provate "`/dev/fd0h1440`", con una "`h`" minuscola.)

(Solo per **SuSE**: Usate "`/dev/fd0u1440`" invece di "`/dev/fd0H1440`".)

Copiate tutti i file in `/boot` nel dischetto:

```
cp -dp /boot/* /mnt
```

(**Solo per Red Hat**: Se la directory `/boot` contiene sia i file di *vmlinux* che di *vmlinuz* (notate la differenza poiché l'ultimo è migliore), avete bisogno di copiare solo i file di *vmlinuz* nel dischetto di avvio. Solo gli stessi file di *vmlinux*, solo che sono compressi per salvare spazio.)

(**Solo per Slackware**: Copiate il file `/vmlinuz` nel dischetto di avvio; usate il comando "`cp /vmlinuz /mnt`".)

Create un nuovo file /mnt/lilo.conf come segue:

```
boot=/dev/fd0           # Installa LILO nel disco floppy.
map=/mnt/map            # Posizione del "map file".
install=/mnt/boot.b     # File da copiare nel settore di avvio del
                        # dischetto
prompt                  # Dice a LILO di mostrare "LILO boot:"
timeout=50              # Avvia il sistema di default dopo 5 secondi.
                        # Il valore è in decimi di secondo.)
image=/mnt/vmlinuz      # Posizione del kernel nel floppy. Il vero nome
                        # può includere un numero di versione,
                        # esempio "vmlinuz-2.0.35".
label=linux             # Etichetta per il sistema Linux
root=/dev/hda1          # Posizione della partizione di root nel nuovo
                        # disco fisso. Modificarlo come appropriato
                        # al vostro sistema. Notate che dovete usare
                        # il nome della posizione futura, una volta
                        # che il vecchio disco è stato rimosso.
read-only               # Monta subito la partizione in sola lettura,
                        # per eseguire fsck.
```

Installate LILO nel dischetto di avvio:

```
/sbin/lilo -C /mnt/lilo.conf
```

L'opzione `-C` dice a LILO che file di configurazione usare.

Smontate il dischetto:

```
umount /mnt
```

11 Rimuovere il vecchio disco

Spegnete il sistema e rimuovete il vecchio disco. Ricordate di modificare i jumper del disco e le informazioni del BIOS per riflettere i cambiamenti

12 Riavviare il sistema, modificare il file di configurazione di LILO

Riavviate il sistema. Se avete problemi, potete usare il dischetto di avvio che avete appena fatto. Per farlo, potreste aver bisogno di modificare la sequenza di avvio del BIOS a `"A:, C:"`. Dovreste modificare il file `/etc/lilo.conf` in caso vogliate più tardi eseguire LILO di nuovo. Ecco un esempio di come il file potrebbe essere:

```
boot=/dev/hda               # Installa LILO nel primo disco.
map=/boot/map               # Posizione del "map file".
install=/boot/boot.b        # File da copiare nel settore di avvio
                            # del disco.
prompt                      # Dice a LILO di mostrare "LILO boot:"
timeout=50                  # Avvia il sistema di default dopo 5
                            # secondi. (Il valore è in decimi di
                            # secondo.)
image=/boot/vmlinuz         # Posizione del kernel. Il vero nome può
                            # includere un numero di versione, per esempio
                            # "vmlinuz-2.0.35".
label=linux                 # Etichetta per il sistema Linux
root=/dev/hda1              # Posizione della partizione di root nel nuovo
                            # disco fisso. Modificarla come appropriato
                            # al vostro sistema. Notate che dovete
                            # usare il nome della posizione futura, una volta
                            # che il vecchio disco è stato rimosso.
read-only                   # Monta subito la partizione in sola lettura,
                            # per eseguire fsck.
```

(**Solo per Slackware**: Usate `"image=/vmlinuz"`.)

13 Riconoscimenti

Grazie a Scott Christensen, Frank Damgaard, Alexandre Fornieles, David Fullerton, Jerry Gaines, Chris Gillespie, Nicola Girardi, Per Gunnar Hansoe, Richard Hawes, Ralph Heimueller, Gerald Hermant, Andy Heynderickx, Paul Koning, Hannu Liljemark, Claes Maansson, Rick Masters, Jason Priebe, Josh Rabinowitz, Douglas Rhodes, Valentijn Sessink, Kragen Sitaker, Stephen Thomas e Gerald Willman.

Questo documento può essere tradotto in qualsiasi lingua. Se lo fate, per favore spedite una copia della traduzione a Konrad Hinsen <*hinsen@cnrs-orleans.fr*>.

Traduzione in italiano a cura di Nicola Girardi <*girardi@keycomm.it*>