



## L'intelligenza dei dischi

**Stanchi di vivere nella paura della schermata delle partizioni di Fedora? Linux Pro vi aiuta ad affrontare i vostri demoni e a scoprire che usare LVM è più semplice di quello che pensate**



Forse alcuni utenti, anche dopo anni e anni di installazioni di distribuzioni basate su Fedora, hanno ancora qualche problema con la schermata di partizionamento del disco. Che cosa cavolo stanno a significare le sigle **vg\_lvm** e **lv\_root**? Perché Fedora non usa i buoni vecchi riferimenti a dispositivi fisici e partizioni, come fanno quasi tutte le altre distro? E forse fino ad ora hanno preferito non perdere tempo nell'imparare il significato di questi oscuri riferimenti, cancellando semplicemente la configurazione di default e creando uno schema personale di partizionamento. Ma ora questo stato di cose è finito perché Linux Pro è qui per darvi una mano: e dopo averci seguito, sarete anche voi in grado di apprezzare quanto potente e utile sia questo tipo di tecnologia, e magari vorrete spendere qualche giorno ad approfondirne ulteriormente la conoscenza sui manuali oppure, come è successo a noi, vorrete usarlo in ogni nuovo schema di partizioni che creerete (almeno fino a quando Btrfs non diventerà di uso comune). Inizieremo questo tutorial spiegando cos'è effettivamente LVM e perché dovrete volerlo usare. Poi diventeremo completamente padroni della schermata d'installazione di Fedora e vi mostreremo come creare a partire da zero uno schema di partizioni basato completamente su LVM. Useremo anche la linea di comando per mostrarvi gli strumenti con cui potrete trarre vantaggio dalle caratteristiche sue caratteristiche avanzate. Dovete ridimensionare un volume? Nessun problema. Volete aggiungere un nuovo disco e rimuoverne uno vecchio? Nulla di più semplice. Vi mostreremo persino come prendere uno snapshot del sistema ed eseguire un backup completo con questo meraviglioso strumento.

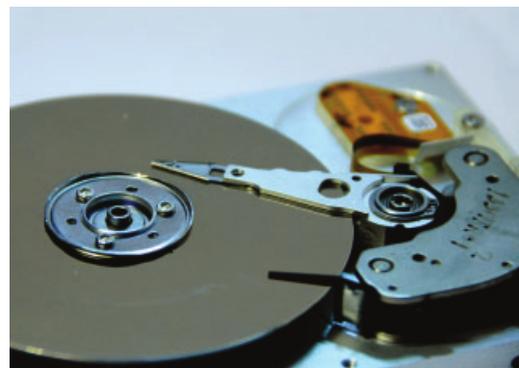
### Storage 101 in Linux

Tutti i file e le cartelle che appartengono a un sistema Linux, tutto il software, tutti i file utente e di configurazione sono ordinati all'interno di una gerarchia di cartelle strutturata ad albero. All'apice di questo albero c'è la cartella di root (rappresentata dal simbolo /) e tutto il suo contenuto. Naturalmente tutto questo è memorizzato su un disco fisico – con le informazioni su dove i file sono effettivamente immagazzinati contenute in qualcosa chiamato **filesystem**. Ora, anche se l'esperienza vi può far pensare al disco fisso come a un unico, monolitico archivio per la memorizzazione dei dati, la realtà dei fatti è raramente questa. La maggior parte delle distribuzioni partiziona il disco fisso, suddividendolo in settori che il sistema operativo tratta come dischi fisici separati. In questo contesto, a ciascuna delle cartelle importanti del sistema viene assegnata una partizione in cui risiedere, ciascuna con il relativo filesystem: molti sistemi creano di default partizioni dedicate per le cartelle **/**, **/home** e **/usr**. Osservate che, siccome le partizioni sono logicamente trattate come dischi diversi, queste possono indifferentemente risiedere effettivamente sia su dischi fisici separati sia sullo stesso disco fisico. È certamente possibile che l'intero albero delle cartelle sia contenuto dentro una sola partizione, ma ci sono molte buone ragioni che di fatto fanno scegliere questa soluzione solo in casi molto rari. Per gli utenti casalinghi, la prima di esse è che partizionando il disco e mettendo in una

partizione separata dal resto la propria cartella utente, si può cambiare distribuzione senza dover eseguire il suo backup e relativo ripristino, siccome solo le partizioni che contengono i file di sistema devono essere formattate. Per le imprese, un'altra ragione è che in questo modo il filesystem può essere reso accessibile in sola lettura, e questo conferisce un ulteriore grado di sicurezza e stabilità oltre alla semplificazione del backup. In aggiunta, siccome le cartelle importanti sono mantenute in sezioni diverse del disco, il corrompersi di una di esse non implica la perdita automatica di tutti i dati.

### Un paio di occhiali dalle lenti rosa?

Tutto questo può apparirci come un sistema ben congegnato, ma come al solito in realtà è ben lungi dall'essere esente da problemi. Il più frustrante di detti problemi è che si tratta di un sistema veramente poco flessibile; le partizioni sono difficili da ridimensionare, sia nel caso che le si voglia estendere per includere nuovi hard disk o che le si voglia comprimere per fare posto ad altre. Supponiamo ad esempio di disporre di un hard disk da 80 GB che desiderate partizionare. Chiaramente vorrete riservare il massimo spazio possibile alla partizione che contiene la vostra partizione **/home**, perché disponete ad esempio di una vasta collezione musicale. Dovete allora decidere quale sia l'ammontare minimo di spazio richiesto dalla partizione **/**, e assegnare lo spazio rimanente a **/home**. Alla fine del lavoro, per la partizione **/** vengono riservati 3 GB mentre la **/home** occupa i rimanenti 77 GB. Tutto quanto funziona bene fin quando non decidete di provare KDE o qualche altro enorme pacchetto software che rende lo spazio disponibile nella root semplicemente insufficiente. In un sistema comune, questo comporta una fastidiosa riduzione a 70 GB dello spazio a disposizione per la vostra musica, e la creazione di uno spazio di memoria da 7 GB largamente inutilizzato. Certo, potreste eseguire il backup di tutto il sistema, reinstallare la vostra distribuzione preferita e ripristinare una configurazione ottima – ma a patto di quale spreco di tempo! Non sarebbe più facile se fosse possibile aggiungere un altro hard disk e rendere disponibile il nuovo spazio di memoria alla partizione **/**, oppure comprimere **/home** di soli 5 GB, da aggiungere alla partizione root?



**Per Linux i dischi fissi sono più che un grande blocco di memoria: sono Master Boot Record, partizioni, file e inode. Un sacco di roba...**



**INTERMEDIO**

## Caratteristiche dei filesystem

Abbiamo accennato al fatto che LVM è in grado di variare facilmente le dimensioni delle partizioni senza doverle smontare. Per poter usufruire di questa possibilità è però necessario usare un filesystem che supporti le stesse caratteristiche, cosa che quelli più diffusi fanno con grado differente. Per semplificarvi la vita abbiamo compilato questa semplice tabella che vi mostra quali filesystem supportano questa caratteristica e in quale misura.

Nome	CONTRAZIONE DINAMICA	ESPANSIONE DINAMICA	CONTRAZIONE STATICA	ESPANSIONE STATICA
<b>ext3</b>	X	✓	✓	✓
<b>ext4</b>	X	✓	✓	✓
<b>btrfs</b>	✓	✓	X	X
<b>xfs</b>	X	✓	X	X
<b>jfs</b>	X	✓	X	X
<b>reiserfs</b>	X	X	✓	✓

### Lui fa tutto

LVM rende tutto questo possibile, nel senso che non dovrete più preoccuparvi di aver sottostimato gli spazi dedicati a una qualche partizione durante l'installazione del sistema. Usando questo strumento potrete allocare dinamicamente più spazio alle partizioni preesistenti, espandere una data partizione su più dischi fisici, e ridistribuire lo spazio in memoria già assegnato. In funzione delle caratteristiche offerte dal filesystem che avete scelto, tutto questo può essere fatto anche mentre le partizioni interessate sono in uso. La ragione di queste sue capacità risiede nel fatto che LVM si comporta come un ulteriore livello software interposto tra il filesystem e il disco fisico, mediando le loro interazioni. Questa mediazione è resa possibile grazie alla suddivisione del disco fisico in porzioni di dimensioni molto più piccole delle ordinarie partizioni, chiamate "estensioni fisiche", che è più facile spostare in giro per il disco e su cui è più facile eseguire anche altri tipi di operazione. Oltre a ciò, questa struttura rende irrilevante la locazione fisica reale delle estensioni fisiche - LVM suddivide tutti i dischi in estensioni della stessa dimensione e li riunisce creando una risorsa di memorizzazione di massa gigante - come un disco fisso virtuale. Anche se molto di quello che LVM fa usa concetti noti, come il filesystem e i dischi fisici, è necessario introdurre la sua terminologia prima di poter proseguire.

■ **Volumi fisici.** Sono i dischi fissi e/o le partizioni che LVM suddivide in estensioni fisiche.

■ **Gruppi di volumi.** Sono l'insieme dei volumi fisici: la grande raccolta di risorse di memoria costituita da tutte le estensioni fisiche.

■ **Volumi logici.** Sono le partizioni logiche create da LVM sul disco virtuale, i gruppi di volume. Queste partizioni possono essere formattate con un dato filesystem, possono essere usate per memorizzare dati e possono essere montate sull'albero delle cartelle principale, esattamente come le partizioni normali.

Ciascuna di queste strutture è parte della gerarchia di LVM: i volumi fisici occupano chiaramente la parte più in basso di detta gerarchia, i gruppi di volumi occupano la parte intermedia e i volumi logici occupano la parte più alta. Ognuna di queste parti è necessaria al funzionamento di LVM, e ognuna di esse deve essere creata in un preciso ordine:

■ i **dischi fisici** o le partizioni devono essere registrate in LVM come **volumi fisici** disponibili;

■ è necessario dire a LVM quali **volumi fisici** volete assegnare a quale **gruppi di volumi**;

■ infine è necessario dire a LVM quali **volumi logici** desiderate includere all'interno di un dato **gruppo di volumi** e la percentuale di spazio di memoria che volete assegnare a ciascuno di essi.

La sola limitazione che incontrerete nel creare un nuovo schema di partizioni con LVM è che ogni schema di questo tipo non può contenere una partizione di boot, quindi dovrete creare **/boot** separatamente. Una volta fatto questo, il gruppo dei volumi e i suoi volumi logici saranno disponibili per essere montati o altrimenti ispezionati nella cartella **/dev**, come succede per ogni altro disco fisico o partizione. Inoltre saranno facilmente identificabili in **/dev** grazie al loro nome, nella forma **/dev/NomeGruppoVolumi/NomeVolumeLogico**. Infine, per chiarire completamente la procedura, in uno specchio è descritta passo passo l'installazione di Fedora usando LVM. A questo punto sapete cosa è LVM, conoscete la sua terminologia e siete riusciti a creare uno schema di partizioni basato su questo strumento con lo scomodo wizard di Fedora. La prossima cosa su cui ci concentreremo è l'uso della linea di comando: partiremo mostrandovi come potete creare un nuovo sistema basato su LVM e come potrete modificarlo. Il primo passo da compiere è registrare i nostri dispositivi hardware e le nostre partizioni come volumi fisici disponibili, usando il comando **pvcreate**:

```
pvcreate /dev/sda1
```

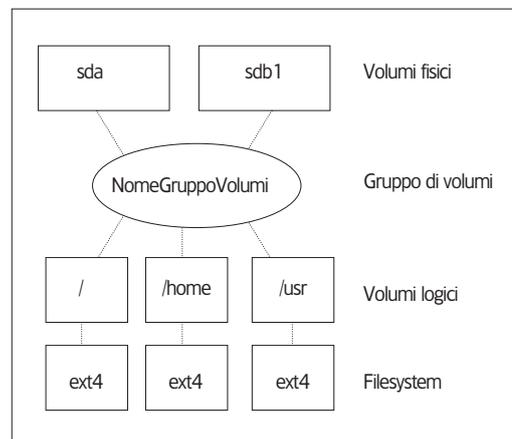
Con questo comando si registra la prima partizione sul primo disco fisico come disponibile. Se desiderate registrare l'intero secondo disco fisico come disponibile, eseguite

```
pvcreate /dev/sdb
```

Notate che se anche LVM funziona correttamente con dischi interi e con partizioni, altri sistemi operativi potrebbero vedere la memoria che avete riservato come libera, per cui fate attenzione a non cancellare informazioni accidentalmente. Procediamo creando un gruppo di volumi a partire dai volumi fisici che abbiamo registrato. Il comando da dare è semplice:

```
vgcreate NomeGruppoVolumi /dev/sda1 /dev/sdb
```

**NomeGruppoVolumi** deve essere ovviamente un nome



**Il gergo di LVM** oscuro un po' la comprensione di come lavora questo strumento, ma è in realtà estremamente logico e descrittivo, come vi mostra questo diagramma, che potete trovare in lingua inglese alla pagina Web <http://tldo.org/HOWTO/LVM-HOWTO/anatomy.html>

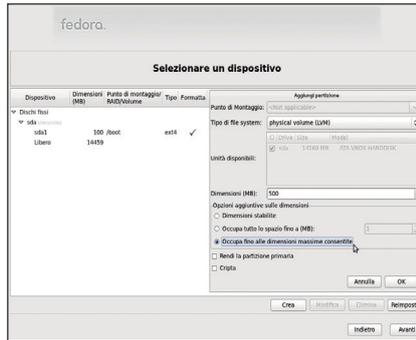


## Passo passo Alla conquista del wizard di partizione di Fedora



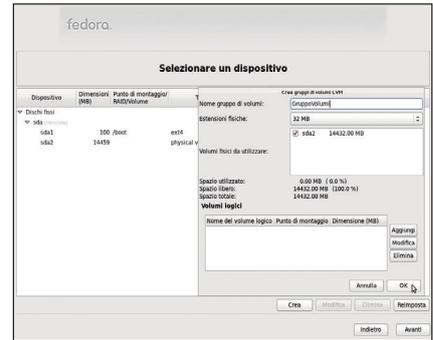
### 1 Creare una partizione

Partendo da un disco fisico vuoto, premete in sequenza i tasti **Crea** ➔ **Partizione standard**. Fissate il punto di montaggio a **/boot** e scegliete in quale disco fisico desiderate memorizzare questa partizione: deve avere dimensione fissa ed essere di soli 100 MB. Per terminare l'operazione premete **OK**.



### 2 Creare una volume fisico

Subito dopo registrate il rimanente spazio disponibile come volume fisico. Per farlo, premete la sequenza di tasti **Crea** ➔ **Volume fisico LVM**, poi impostate il parametro **Tipo di file system** a **physical volume** (**LVM**). Scegliete inoltre l'opzione **Occupi fino alle dimensioni massime consentite**. Per terminare l'operazione premete **OK**.



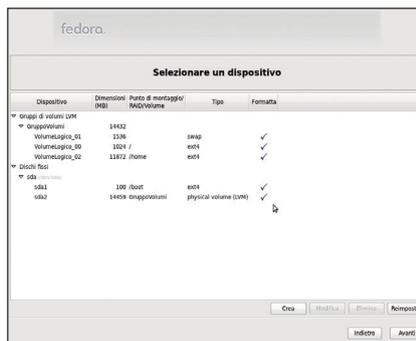
### 3 Creare un gruppo di volumi

Per creare il gruppo di volumi e i volumi logici premete in sequenza i tasti **Crea** ➔ **Gruppo di volumi LVM**. Dategli un nome riconoscibile e poi specificate i volumi fisici da usare. La dimensione delle estensioni fisiche può essere lasciata al valore di default. Per terminare l'operazione premete **OK**; potete però passare direttamente al prossimo passo senza farlo.



### 4 Creare volumi logici

Ora potete creare i volumi logici premendo il tasto **Aggiungi** nel riquadro **Crea gruppo di volumi LVM**. Specificate il punto di montaggio a **/**, **/home** o a qualsiasi altra cartella che desiderate lo contenga, scegliete il suo filesystem, chiamatelo con un nome semplice e descrittivo, e decidete la sua dimensione.



### 5 Controllare i cambiamenti

Prima di procedere al passo finale, controllate le modifiche che avete deciso di fare. Controllate che dischi fissi e partizioni siano registrati come volumi fisici, che abbiano tutti la dimensione corretta e che non abbiate inavvertitamente cancellato qualcosa che desiderate tenere.



### 6 Applicare i cambiamenti

Infine riceverete il solito avvertimento da parte del wizard, che vi avvisa di essere in procinto di creare un nuovo schema di memoria e che ogni vostro dato presente nelle partizioni cancellate o ridimensionate sarà perso. Se siete sicuri di aver lavorato bene, premete il tasto **Scrivere i cambiamenti su disco**.

descrittivo per il gruppo di volumi che si è così creato, specificando quelli che sono i volumi fisici che contribuiranno alla risorsa di memoria. Se non vi ricordate quali siano i volumi fisici che avete creato, potete eseguire il comando **pvscan** e ottenere come risposta una lista dei dischi e delle partizioni disponibili. **pvdisplay** può invece essere usato per avere una descrizione dettagliata degli attributi di un qualsiasi volume fisico, quali possono essere la dimensione, la locazione sul disco, il nome, ecc. È necessario creare i volumi logici a partire dal gruppo di volumi appena creato: per farlo digitiamo **lvcreate -L10G -nNomeVolumeLogico NomeGruppoVolumi**. L'opzione **-L** specifica la dimensione desiderata del nuovo volume logico, espressa rispettivamente in Kilobyte, Megabyte e Gigabyte se i suffissi sono K, M e G, l'opzione **-n** ne specifica il nome mentre l'opzione finale specifica da quale gruppo di volumi volete ricavarlo. Questi sono tutti i comandi che dovete conoscere direttamente legati a LVM e, come potete vedere, sono tutti molto semplici e diretti. Giunti a questo punto, è necessario creare un filesystem su ogni volume logico e definire come montare ciascuno di essi. Creare

il filesystem è un'operazione molto semplice se siete interessati a usare la famiglia ext, molto stabile e quindi diffusa. Dovrete installare il pacchetto **e2fsprogs**, che vi fornirà il comando **mke2fs**. A questo punto potete decidere quale versione di ext creare (la 4 è la più grande e più recente) e su quale volume logico crearla: 

```
mke2fs -t ext4 /dev/NomeGruppoVolumi/ NomeVolumeLogico
```

 Gli altri filesystem dispongono di strumenti analoghi, quindi nessuno di essi risulta molto difficile da creare. Google vi dirà volentieri di quali strumenti avete bisogno per creare un dato filesystem, e le pagine del manuale vi aiuteranno con i dettagli sulle opzioni da scegliere. Dopo che avrete creato il filesystem sul volume logico, dovrete solo decidere come montarlo. Per eseguire il montaggio automaticamente ogni volta che eseguirete il boot del computer, aggiungete semplicemente il volume logico nei dispositivi elencati nel file **/etc/fstab**. La struttura di questo file è costituita da un insieme di righe di testo, una per ciascun filesystem che si desidera montare: ciascuna di queste righe è composta

da una serie di comandi, tra loro separati da spazi o TAB, che indicano la partizione o il dispositivo da montare, dove montarlo, il suo tipo di filesystem, e numerose altre opzioni.

Per una partizione di tipo LVM, la forma tipica è la seguente:  
`/dev/NomeGruppoVolumi/NomeVolumeLogico /home ext4 defaults O 1`

Per la spiegazione in dettaglio del significato delle opzioni da **defaults** in avanti vi rimandiamo al manuale di **/etc/fstab**.

## Aggiungere memoria extra

Quello che abbiamo fatto fino a questo punto è creare uno schema che è funzionalmente equivalente a un sistema normalmente partizionato. Ora però che tutto funziona, è il momento di provare a usare qualcuna delle caratteristiche avanzate di LVM. Inizieremo con il vedere come aggiungere memoria extra, nella forma di un nuovo volume fisico, prima a un gruppo di volumi, poi a un volume logico.

Come accade per tutti i volumi fisici, la prima cosa da fare è inizializzarlo con il comando **pvcreate**. Una volta che avrete fatto, dovrete usare il comando **vgextend** e aggiungere il nuovo arrivato a un gruppo di volumi:

```
vgextend NomeGruppoVolumi /dev/sda2
```

La sintassi del comando è autoesplicativa:

**NomeGruppoVolumi** è il gruppo di volumi da estendere, mentre **/dev/sda2** è il volume fisico che lo estenderà (già inizializzato). Ora che è disponibile memoria extra, potete estendere a vostro piacimento anche i volumi logici all'interno del gruppo di volumi. Per procedere usiamo il comando **lvextend**, di sintassi simile al comando **lvcreate** (in particolare ha anch'esso l'opzione **-L**).

```
lvextend -L8G /dev/NomeGruppoVolumi/ NomeVolumeLogico
```

Facile, vero? Purtroppo questo comando non estende parimenti le dimensioni del filesystem, quindi dovrete provvedere voi a eseguire questa operazione, prima di potervi avvantaggiare del nuovo spazio disponibile. Il seguente comando provvede allo scopo nel caso che il filesystem impiegato sia ext4, espandendolo alla stessa dimensione del volume logico appena modificato:

```
resize2fs /dev/NomeGruppoVolumi/NomeVolumeLogico
```

La cosa realmente sorprendente di tutto questo è che potete farlo senza dover far ripartire il sistema. Quindi, per tornare al nostro esempio iniziale, quando il nostro sistema da 80 GB va a corto di spazio, ci basta semplicemente aggiungere un hard disk supplementare e estendere la partizione / usando lo spazio extra disponibile nel sistema. È chiaramente un modo di procedere molto meno problematico rispetto alla reinstallazione dell'intero sistema. A questo punto potreste chiedervi cosa sarebbe successo se quei famosi 80 GB fossero stati la memoria di massa di un portatile, senza alcuna possibilità di aggiungere un hard disk nuovo: è possibile, in casi come questo, comprimere la cartella /home in favore della cartella /? La risposta è sì, a patto di rispettare alcune avvertenze, la più importante e ovvia delle quali è che non potete ridurre lo spazio di memoria di un filesystem al di sotto della dimensione totale dei dati che contiene. Se forzerete un'operazione di questo tipo, perderete certamente dati e correrete il rischio di corrompere anche quelli che vi rimarranno: quindi assicuratevi che ci sia sufficiente spazio libero nel filesystem prima di iniziare. L'altra avvertenza è che, nonostante LVM riesca benissimo a ridurre la dimensione di un volume logico quando è già montato, la maggior parte dei filesystem non consente questa operazione. Se state modificando una partizione come / o /home dovrete

## Btrfs

**Btrfs** è un nuovo filesystem per Linux, progettato per aiutare il sistema operativo nella gestione di grandi e grandissimi sistemi di memoria quali quelli in uso nei Centri Elaborazione Dati e in altri ambiti industriali. Sua peculiarità è fornire un insieme di robusti strumenti di gestione e altre caratteristiche avanzate. Btrfs supporta anche il concetto dei sottovolumi, vale a dire cartelle di root interne separate (ed è infatti questa sua caratteristica che rende possibile avere un'immagine dello stato del sistema per il backup). Mentre questo concetto può sembrare simile a quello di volume logico, in realtà se ne differenzia molto perché in questo caso non si tratta di blocchi di memoria ma di cartelle

montabili all'interno dell'albero principale dei filesystem. A parte le differenze tecniche, i due concetti forniscono servizi molto simili. Btrfs riduce molto la necessità di usare LVM e, grazie al fatto che queste sue caratteristiche sono date congiuntamente a quelle di un filesystem standard senza la necessità di dover gestire partizioni o volumi logici, il lavoro per la sua amministrazione risulta grandemente ridotto. Inoltre il sistema è più sicuro – grazie a caratteristiche come il controllo online del filesystem, i checksum e il mirroring, caratteristiche che ogni altro filesystem che voi potreste usare congiuntamente a LVM non ha. Tutte queste caratteristiche rendono Btrfs molto allettante.

fare i reboot del sistema usando un disco live della vostra distribuzione, o comunque un altro sistema di ripristino.

A parte queste avvertenze, la procedura è semplice, e ve la descriviamo per il solito sistema con filesystem ext4.

Per prima cosa eseguite il comando **resize2fs** specificando la nuova dimensione del sistema, poi eseguite il comando **lvreduce** per ridurre la dimensione del volume logico.

```
resize2fs /dev/NomeGruppoVolumi/NomeVolumeLogico 2G  
lvreduce -L2G /dev/NomeGruppoVolumi/  
NomeVolumeLogico
```

Il parametro **2G** in tutti e due i comandi specifica rispettivamente la nuova dimensione del filesystem e del volume logico. È l'ultima cosa che c'è da dire: come potete vedere, anche una procedura come questa, che può incutere timore a causa dei possibili effetti disastrosi, si riesce a portare a termine tutto sommato senza troppa fatica.

## Spostando in giro i dati

In questa sezione vedremo come rimuovere un vecchio disco fisico e sostituirlo con uno nuovo. Per farlo senza rischiare di perdere nulla, è opportuno prima copiare i dati contenuti nel disco da sostituire in un'altra parte del gruppo di volumi. Questa operazione può, a prima vista, sembrarvi complessa: dopotutto, come fate a sapere quali parti del gruppo di volumi ha memorizzato LVM su un dato volume fisico? Fortunatamente LVM dispone di un altro utile comando che rende tutto questo incredibilmente semplice:

```
pvmove /dev/sda
```

La sua esecuzione causerà la ridistribuzione di tutte le estensioni fisiche memorizzate nel dato hard disk tra gli altri volumi fisici del gruppo di volumi – se chiaramente è rimasto sufficiente spazio disponibile. In caso contrario, prima di lanciare il comando e rimuovere il vecchio dispositivo fisico dovrete aggiungere il nuovo come abbiamo fatto in una delle sezioni precedenti: per verificare la sufficienza o meno dello spazio rimasto vi basterà confrontare i dati in uscita ai comandi **pvdisplay** e **vgdisplay**. A questo punto l'unica cosa da fare è rimuovere il vecchio dispositivo dal gruppo di volumi, usando il comando **vgreduce**:

```
vgreduce NomeGruppoVolumi /dev/sda
```

Questo comando causerà la rimozione del volume **/dev/sda/** dal gruppo di volumi **NomeGruppoVolumi**. Giunti a questo punto potete rimuovere il disco fisico – e bruciarlo, prenderlo a martellate o semplicemente assicurarvi che nessuno possa metter mano sui vostri dati in esso contenuti. Questo tutorial finisce qui. Speriamo di avervi aperto gli occhi sul potenziale di LVM e che consideriate di usarlo per i vostri sistemi futuri. [LXP](#)