



Raid e Lvm con Linux: più spazio, più velocità

Come scegliere la migliore configurazione per ottenere prestazioni e sicurezza, gestendo con precisione i dischi collegati.

Nel numero scorso di questa rubrica abbiamo visto come Linux gestisce partizioni, file e cartelle dei suoi utenti, proteggendoli nei limiti del possibile da guasti o attacchi informatici, grazie ai suoi file system. Questo mese, come promesso, continuiamo il discorso presentando la maniera Linux di vivere, senza compiacersi troppo la vita, con quantità sempre maggiori di file e dischi sempre più grandi. Già, perché se si vuole stare tranquilli non basta avere un file system capace di contenere tanti Terabyte di

dati, o singoli file di dimensioni gigantesche. Occorre anche avere e configurare, correttamente ma in maniera flessibile, una quantità di dischi sufficiente a ospitare quel file system. I due strumenti Linux più comuni per questo scopo, Raid con mdadm e Lvm, costituiscono l'argomento di queste pagine. L'altro articolo della rubrica discute invece la configurazione avanzata dei file system per Linux.

Prima di cominciare qualche consiglio generale da seguire sempre, pur senza spaventarsi troppo. Per quanto utili, potenti e stabili siano, Linux e i metodi di cui stiamo per parlare *non* devono essere mai, mai considerati come sostituti dei backup. Un furto, un baco in una qualsiasi applicazione o, ancora di più, un errore umano, possono cancellare file importanti in qualsiasi momento, qualunque combinazione di Raid o Lvm sia stata adottata. Backup prima di tutto, quindi.

L'altro consiglio è sperimentare quello che state per leggere su un disco *extra*, prima di applicare una procedura o l'altra a quelli che effettivamente contengono, o conterranno, i vostri dati. Per imparare a lavorare con Raid o Lvm basta un disco o chiave Usb, anche di capacità molto ridotta, e dividerli in diversi partizioni. A questo punto potrete, senza timore, non solo combinarle in array e volumi a vostro piacimento, ma anche simulare eventi traumatici come: guasti, rimozioni o sostituzioni a caldo di dischi, per verificare la correttezza della configurazione scelta.

L'ultimo suggerimento prima di entrare nel merito è cercare aiuto nel posto giusto. Configurare array e volumi sotto Linux non è particolarmente difficile, ma dipende molto più di altre cose dalla combinazione di vari elementi, dalla

versione di Linux usata a numero e specifiche hardware di tutti i dischi coinvolti, passando per il Bios della scheda madre. Raccomandiamo perciò di raccogliere tutte queste informazioni *prima* di cominciare, per chiedere consiglio su come procedere direttamente nel forum o mailing list della vostra distribuzione.

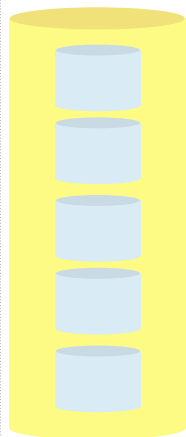
Raid, velocità e sicurezza a basso costo

Raid è un acronimo inglese (*Redundant Array of Inexpensive Disks*) che letteralmente significa gruppo ridondante di dischi rigidi economici. Dal punto di vista dell'utente finale, un Raid è un dispositivo di archiviazione dotato, a seconda del tipo, di alta velocità e/o protezione da singoli guasti. La velocità si ottiene collegando e usando due o più dischi come se fossero parti diverse di uno stesso disco, e accedendo in parallelo a entrambi. La protezione dai guasti si ottiene, come indicato dal nome, quando uno o più dischi del gruppo sono in effetti *ridondanti*, perché contengono gli stessi dati: di conseguenza, quando uno di quei dischi si rompe, non c'è alcuna perdita reale di dati.

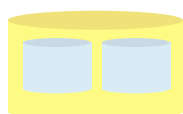
Un Raid può essere un prodotto commerciale hardware vero e proprio, che magari racchiude in un singolo contenitore dischi ottimizzati per quel compito, il relativo controller hardware e altri accessori. Spesso però si parla anche di Raid *software*: in questo secondo scenario vari dischi rigidi interni o esterni vengono connessi in modalità Raid con software apposito.

Le varie configurazioni Raid sono distinte da numeri, di cui qui per mancanza di spazio ricordiamo i più comuni: Raid 0 indica due o più dischi, più o

CONFIGURAZIONI RAID PIU' COMUNI



RAID 0
Veloce, non protetto



RAID 1
Duplicazione dei dati

I vari tipi di Raid possono confondere, ma in realtà sono basati su due concetti fondamentali, a cui si aggiunge la parità per il controllo degli errori: creazione di dischi virtuali più grandi ma veloci (Raid 0) oppure di dischi ridondanti, in cui ogni bit è copiato due volte (Raid 1).

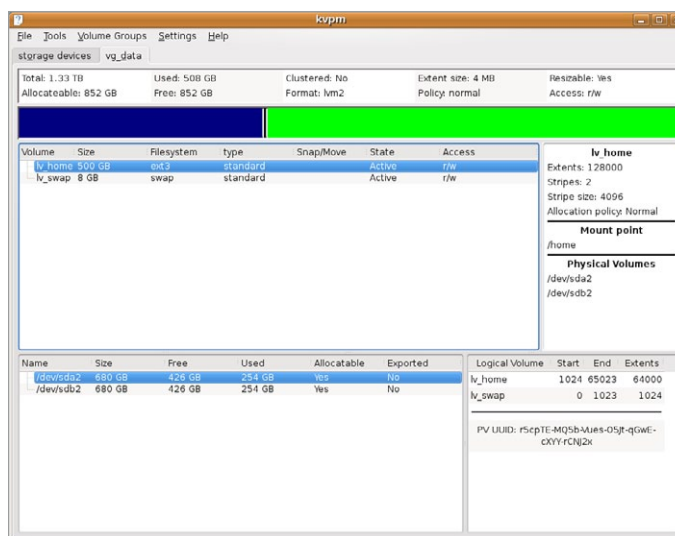
meno delle stesse dimensioni, utilizzati in parallelo: massima velocità, capacità pari alla somma di quelle dei singoli dischi, ma nessuna protezione. Se un disco si guasta si perde tutto o quasi. Raid 1 è, in un certo senso, l'opposto dell'altro, poiché indica dischi sempre più o meno uguali, ognuno dei quali contiene sempre una copia completa dei dati. Altre combinazioni combinano N dischi in vari modi, per avere sia velocità sia ridondanza: la più popolare di queste configurazioni è forse la Raid 5, che fornisce un disco virtuale di dimensioni pari a N -1 volte quella del disco più piccolo del gruppo.

La velocità effettiva del Raid dipende sia da quelle dei suoi componenti fisici, sia da alcuni parametri di configurazione, almeno per i Raid software. Uno che certo fa la differenza è la dimensione della relativa cache, soprattutto per le scritture di dati. Scegliendo la dimensione giusta, in funzione del numero medio di scritture e dell'intervallo minimo fra due scritture consecutive, si possono migliorare notevolmente le prestazioni dei Raid, soprattutto di tipo 5.

Qualunque siano la sua natura (hardware o software) e la sua configurazione, una volta pronto un dispositivo Raid si può usare a tutti gli effetti come un singolo disco rigido tradizionale, su cui creare quante partizioni si vuole. Linux supporta sia Raid hardware sia software, con quasi tutti i file system che saprebbe gestire su dischi standard, su storage Scsi, Ide, Sata e Usb, anche di diverso tipo. Più in generale, Linux può creare e gestire Raid su tutte le periferiche (fisiche o virtuali) che siano dispositivi a blocchi (*block device*) del tipo spiegato nel numero scorso. L'unica limitazione è che la partizione di boot (*/boot*) non può stare su un dispositivo Raid 1. D'altra parte, il motivo per cui un Raid software Linux può essere diviso in partizioni è che anch'esso è un dispositivo a blocchi, sia pur virtuale. Combinando insieme queste caratteristiche, è possibile costruire su Linux dei Raid di Raid.

Queste configurazioni vengono chiamati Raid 10, perché combinano i vantaggi dei tipi 1 e 0, ovvero massima velocità e ridondanza.

Un discorso a parte merita la partizione di *swap*. Conviene o no metterla su un dispositivo Raid? In pratica, dipende da cosa si vuole ottenere. Tecnicamente, per le ragioni che abbiamo appena spiegato, non c'è nessun problema. La



Linux offre interfacce grafiche, come Kvm, in cui è possibile configurare sia Raid che Volumi Logici.

swap è una partizione e qualsiasi Raid, essendo un dispositivo a blocchi, può ospitare le stesse partizioni di un disco vero. Detto questo, porre una *swap* su Raid è necessario solo per aumentare l'affidabilità e la disponibilità di un computer Linux, non la sua velocità. Fin dalla serie 2 del kernel, infatti, Linux è in grado da solo di suddividere una *swap* in più parti accessibili in parallelo (*stripes* o *strisce*), su più dispositivi. Basta definire le strisce nell'elenco delle partizioni, quello conservato nel file */etc/fstab*, dando a tutte la stessa priorità. Per la disponibilità il discorso cambia completamente. Una *swap* su un dispositivo Raid 1, quindi duplicata su più dischi, può sopravvivere anche al crash di uno di loro, se configurata opportunamente.

Mdadm, il coltello svizzero per Raid su Linux

Le prime versioni di Raid su Linux funzionavano grazie all'uso di diversi programmi e programmini più o meno slegati fra loro, ognuno con una sua sintassi e configurazione indipendente. Fortunatamente questa situazione è stata superata. Oggi, anche se gli si parla solo attraverso qualche interfaccia grafica, amministrare Raid sotto Linux significa sicuramente usare il programma *mdadm* (<http://neil.brown.name/blog/mdadm>). Questa applicazione offre una singola interfaccia e sintassi per qualsiasi attività di gestione Raid, senza quasi nessun bisogno di leggere file di configurazione. Questa caratteristica è molto più rilevante di quanto non

Raid dà nuova vita anche a dischi Ide

Per organizzazioni no profit, scuole e piccole aziende non è infrequente avere ancora (magari attraverso donazioni) Pc abbastanza obsoleti da contenere o poter usare solo i vecchi hard disk Ide. Grazie a Linux è possibile ridare a quei computer non solo una nuova vita, ma anche una velocità forse mai conosciuta, nemmeno quando erano nuovi. Il Raid su Linux funziona infatti senza problemi anche su dischi Ide, a patto di tener presenti poche semplici regole. Prima di tutto, per sua stessa natura, qualsiasi disco Ide oggi è un componente molto vecchio, statisticamente piuttosto vicino a guastarsi senza rimedio. Sì al Raid, quindi, ma solo in configurazioni ridondanti, o almeno con backup talmente frequenti da evitare ogni perdita di dati significativi. L'altro limite, più tecnico ma non negoziabile, è ricordarsi di connettere solo *un* disco a ogni bus Ide. In caso contrario, per prima cosa si vanificherebbe (parzialmente) l'uso stesso del Raid, che è veloce finché si può accedere a ogni singolo disco fisico in maniera diretta e indipendente. In secondo luogo, un guasto del disco potrebbe bloccare l'intero bus, e quindi rendere inutilizzabile tutto il Raid. Per lo stesso motivo non è possibile, a meno di configurazioni molto particolari e rischiose, effettuare la sostituzione a caldo di dischi Ide nei Raid.

ARCHITETTURA LVM



sembri, perché significa che mdadm è utilizzabile *subito*, anche quando lo si lancia da un Dvd Linux Live, per riparare sistemi danneggiati! C'è però un altro motivo per cui è opportuno avere un'idea generale di come lavora mdadm, anche se non si intende usarlo direttamente: questo programma è lanciato anche dagli script di avvio e spegnimento di Linux per configurare i dispositivi Raid presenti, verificarne lo stato e tentare di risolvere i problemi più comuni.

Mdadm ha vari modi di funzionamento, di cui qui menzioneremo solo i più comuni: creazione, montaggio, monitoraggio e diagnostica di Raid.

Quando crea un Raid, mdadm deve inizializzarlo, un'operazione che in alcuni manuali è chiamata anche *ricostruzione*. Questo è necessario per garantire che tutti i bit di parità dei dati nel Raid siano in uno stato noto. La ricostruzione in pratica funziona nello stesso modo che se fosse stato appena rimpiazzato un disco difettoso del Raid. Per default, mdadm esegue questa procedura nel modo più trasparente possibile, per non rallentare eccessivamente il computer e permettere l'uso del Raid anche *durante* la ricostruzione stessa. Volendo, anche se non consigliamo certo di farlo, si potrebbe addirittura riformattare il dispositivo a blocchi corrispondente al Raid prima che questa fase sia finita. Una volta terminata l'inizializzazione, mdadm consente di attivare l'array, rendendolo disponibile al sistema Linux a cui è collegato.

Le funzioni di monitoraggio aggiornano automaticamente uno stato sia per l'intero Raid sia per ogni suo componente, e sono disponibili in tutte le configurazioni tranne Raid 0, in cui non essendoci ridondanza non c'è nulla da controllare.

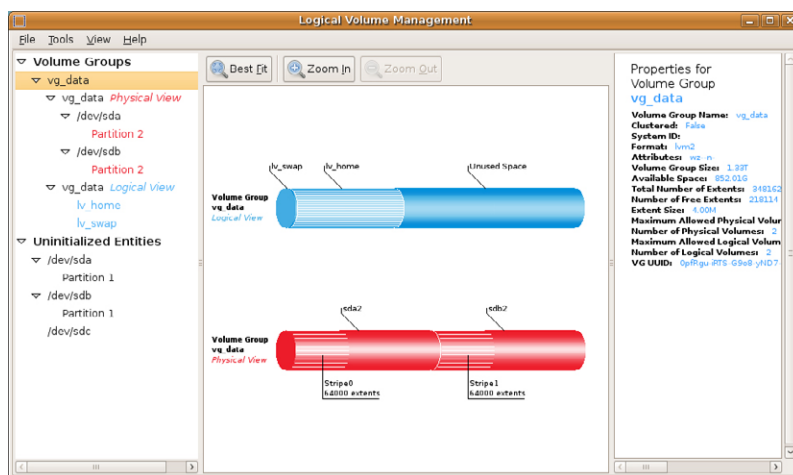
Lvm, quello che mancava ai Raid

Come Raid, Lvm nel mondo Linux è un acronimo, che in questo caso sta per Logical Volume Manager. A prima vista le funzioni di Lvm potrebbero sembrare una duplicazione di alcune di quelle Raid, tanto da far chiedere a un osservatore frettoloso se l'intera idea non sia tutto sommato inutile. In realtà, Lvm ha uno scopo complementare, per non parlare di un modo completamente diverso di lavorare. Un Raid aumenta la velocità di accesso ai dati, o aggiunge ridondanza, ovvero possibilità di perdere un disco *fisico* senza nessuna perdita di *dati*. L'uso più comune di Lvm, invece, è la creazione di unità di storage virtuali, o meglio *logiche*, grandi a piacere, facilitandone il più possibile la manutenzione.

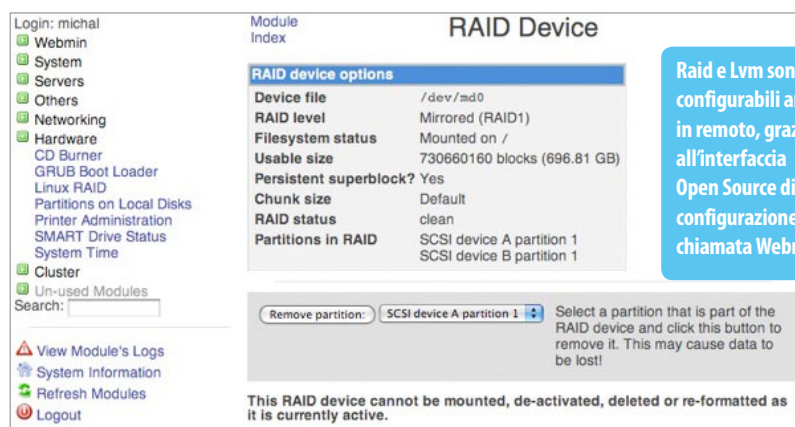
Dal punto di vista tecnico, Lvm è uno strato, puramente software, steso direttamente sopra dischi rigidi e partizioni. Per questo motivo un volume Lvm non

può ospitare la cartella /, ma questo in pratica non è mai un problema. Il compito di Lvm infatti è far apparire tutti gli oggetti che ricopre come un'unica, enorme unità di storage (volume, appunto), di capacità pari alla somma di quelle dei componenti. I volumi logici di Lvm sono, come i Raid, costruiti sopra generici dispositivi a blocchi, che possono essere singole partizioni o interi dischi fisici. Questi ultimi vengono prima configurati come Volumi Fisici (Pv, *Physical Volume*). Quei volumi vengono poi combinati in gruppi (Vg) da cui vengono ottenuti vari Volumi Logici (Lv), ognuno dei quali a sua volta ospiterà un file system. Per combinare unità fisiche di ogni dimensione e poter variare le dimensioni delle partizioni senza effettuare reboot, i Volumi Fisici vengono suddivisi in particelle, chiamate *physical extent*, di dimensione fissa. Queste particelle sono i mattoni con cui è costruito un volume, e corrispondono alla quantità di storage più piccola indirizzabile su di esso. I *physical extent* vengono poi associati uno a uno alle particelle della stessa dimensione, chiamate *logical extents*, che costituiscono il Volume Logico.

Se fosse tutto qui, si potrebbe avere ragione a considerare Lvm solo come una sorta di Raid software, ma la sua struttura permette ben altro. Il primo vantaggio di Lvm è che aumentare o diminuire le dimensioni del volume, aggiungendo o togliendo singoli dischi fisici, diventa facile e non richiede nessun riavvio del sistema o ripartizionamento. Certo, quanto abbiamo appena detto non è realmente applicabile a



Questa schermata dell'utility system-config-lvm di Fedora mostra in forma grafica un gruppo di volumi fisici (parte rossa) e i corrispondenti volumi logici costruiti su di esso.



RAID Device

RAID device options	
Device file	/dev/md0
RAID level	Mirrored (RAID1)
Filesystem status	Mounted on /
Usable size	730660160 blocks (696.81 GB)
Persistent superblock?	Yes
Chunk size	Default
RAID status	clean
Partitions in RAID	SCSI device A partition 1 SCSI device B partition 1

Remove partition: SCSI device A partition 1

Select a partition that is part of the RAID device and click this button to remove it. This may cause data to be lost!

This RAID device cannot be mounted, de-activated, deleted or re-formatted as it is currently active.

Mini glossario per l'ottimizzazione di Ext4 su Linux

I file system per Linux presentati nel numero scorso hanno tutti prestazioni molto buone, almeno per la maggior parte degli utenti. Anche Ext4, quello creato per default da quasi tutti gli installer Linux attuali, normalmente non ha alcun bisogno di aggiustamenti. D'altra parte, si sa, la velocità non basta mai, e anche quando non serve davvero avere un computer più rapido fa piacere, oltre a essere una buona motivazione per imparare l'informatica. Al momento, ottimizzare un file system durante o dopo l'installazione non richiede la riga di comando e non è una procedura particolarmente complicata... se si conosce in anticipo il significato di certi termini, che non sempre sono tradotti o spiegati in Italiano. Ecco quindi una breve spiegazione delle opzioni più interessanti per un file system Ext, per orientarsi durante l'installazione e farsi, anche prima, un'idea di quanto è possibile. Ricordiamo comunque che per tutte o quasi le operazioni che seguono è obbligatorio smontare il file system interessato prima di procedere, altrimenti la perdita di dati sarà quasi certa!

L'altra cosa da non dimenticare quando si sperimentano queste funzioni è tenere a portata di mano una distribuzione Live su DVD. In questo modo, in caso di problemi sarà possibile riavviare subito il computer e procedere alla riparazione dei file system o della relativa tabella di configurazione (il file /etc/fstab). Infine, per i dettagli di queste operazioni consultate le istruzioni specifiche per la vostra distribuzione!

noatime e nodiratime: queste due opzioni disabilitano la scrittura automatica dei tempi di accesso a tutti i file (o cartelle, per la seconda) del relativo file system. Anche se questa informazione è vitale in vari campi, da certi server Web ad altri per gestione documentale, la verità è che a tanti utenti non serve affatto. Fare a meno di salvarla su disco può quindi aumentare sensibilmente le prestazioni del computer. L'eccezione, non trascurabile, a questa regola è che alcuni programmi di uso comune su ogni desktop Linux, per esempio dei client di posta →

dischi tutti *interni* al computer, quindi accessibili solo togliendo la corrente. Se però si usano vari dischi esterni, trattarli come un unico volume logico può semplificare l'amministrazione del sistema. Allo stesso tempo, attivare Lvm durante l'installazione di Linux potrebbe far comodo anche a casa. Se in un secondo momento si decidesse di raddoppiare la capacità delle proprie *cartelle* (non partizioni!), Lvm permetterebbe di farlo senza problemi e ripartizionamenti, cosa non possibile con Raid. Per le stesse ragioni, un disco che fa parte di un volume Lvm si può svuotare del suo contenuto prima di rimuoverlo, senza arrestare il sistema. Lvm facilita anche la creazione e uso di *snapshot*, che sono "fotografie istantanee" di tutti i dati di un intero volume logico, nati principalmente per due scopi. Il più comune è la creazione di backup completi di un volume, relativi a uno ed un solo momento ben preciso, senza interrompere le attività dei programmi. Gli snapshot sono poi utilizzabili anche per creare copie

temporanee di dati, su cui collaudare programmi potenzialmente distruttivi. Gli snapshot di Lvm sono utilizzati anche per creare macchine Linux virtuali, da far girare all'interno di supervisor come Xen (www.xen.org).

Lvm e Raid insieme?

Si potrebbe pensare che la soluzione ideale, almeno per chi lavora regolarmente con video o grandi quantità di dati, sarebbe qualcosa che abbia sia le caratteristiche di un Lvm sia quelle di un Raid. Fortunatamente, su Linux è facile costruirlo da soli. Questo perché sia Lvm sia Raid sono dispositivi a blocchi virtuali, che vanno costruiti su altri dispositivi a blocchi. L'immediata conseguenza di ciò è che si può tranquillamente creare un Raid che dia la velocità e/o la ridondanza desiderate, e poi costruirci sopra un Volume Logico ingrandibile in futuro (con altri Raid!) senza rifare tutto da capo, per non parlare dei backup con snapshot.

RISORSE

La rete è ricca di documentazione, magari arida ma precisa e abbastanza aggiornata, sugli argomenti di questo mese. Architettura e caratteristiche del file system Ext4 si trovano su www.kernel.org/doc/Documentation/filesystems/ext4.txt. Per trovare più dettagli su come ottimizzare questo e altri file system di Linux consigliamo gli articoli www.softpanorama.org/Commercial_linuxes/Performance_tuning/disk_subsystem_tuning.shtml, <http://erikuguel.wordpress.com/2011/04/14/the-quest-for-the-fastest-linux-file-system/> e <http://blog.smartlogicsolutions.com/2009/06/04/mount-options-to-improve-ext4-file-system-performance/>.

Per saperne di più su Raid sotto Linux conviene partire dalla guida in Italiano www.linuxguide.it/linux/amministrazione_sistema/filesystem/234_RAID-con-MDADM.html, ma senza trascurare la documentazione ufficiale: https://raid.wiki.kernel.org/index.php/Linux_Raid (sullo stesso sito c'è anche una pagina dedicata a inserzione e rimozione a caldo di dischi in array Raid). Quanto a Lvm, le risorse più efficaci sono probabilmente il tutorial www.howtoforge.com/linux_lvm e il manuale www.centos.org/docs/5/pdf/Cluster_Logical_Volume_Manager.pdf.

→ elettronica, hanno bisogno di quelle informazioni per funzionare correttamente. In caso contrario non potrebbero, per esempio, capire se l'utente ha già aperto una certa email durante la sessione corrente. Fortunatamente, se si scopre che l'uso di noatime causa problemi, lo si può disabilitare senza conseguenze.

data=writeback: un file system Ext4 può funzionare in modalità writeback. Questo termine sta a significare che (se viene dato il comando adatto, normalmente tune2fs o qualche suo front-end grafico) i metadati di un file non vengono salvati in maniera sincrona, ovvero insieme al file stesso ogni volta che quest'ultimo cambia. La loro scrittura viene invece rimandata, entro certi limiti, a quando rallenterà il sistema il meno possibile. Lo svantaggio del writeback è che usarlo potrebbe causare la perdita dei cambiamenti più recenti di un file in caso di crash. Occorre perciò valutare con attenzione se usarlo o no. In ogni caso, writeback e noatime rimangono le opzioni più facili da usare per aumentare sensibilmente le prestazioni del filesystem.

Barrier=0: questa opzione, più misteriosa delle prime due, abilita o disabilita l'uso delle barriere di scrittura per la funzione di journaling. In parole povere, quando le barriere sono attive, i dati di journaling vengono sempre scritti in un ordine predeterminato, più lento ma più capace di minimizzare i danni in caso di blackout. Per la cronaca, Ext4 è il primo file system Linux in cui queste barriere sono attive per default, cosa che al suo apparire ha causato proteste da chi lo trovava più lento del suo predecessore. Disabilitare le barriere fa lavorare il codice di journaling con meno vincoli, quindi più in fretta, ma è davvero sicuro farlo solo quando un gruppo di continuità protegge da interruzioni di corrente.

errors=remount-ro: a differenza degli altri casi, qui non si tratta di aumentare le prestazioni ma la robustezza del sistema, o almeno di limitare i danni. Quando è presente, questa opzione dice al kernel che, qualora venissero rilevati errori in un file system, quest'ultimo dovrà essere immediatamente rimontato in modalità Read Only. Questo dovrebbe evitare che file o cartelle già corrotti per qualsiasi ragione da ulteriori riscritture che potrebbero renderne impossibile il recupero.



Sempre connessi a Ubuntu con Splashtop Streamer

Questa è una app per dispositivi Android e iOS con cui è possibile connettersi in remoto a computer su cui è installato il programma Splashtop2 (www.splashtop.com/splashtop2). Quest'anno Splashtop 2 e il suo Streamer sono utilizzabili anche per connettersi direttamente a desktop Ubuntu 12.04. Oltre a tutti gli utenti Ubuntu che possiedono terminali Android, Splashtop intende servire esplicitamente i giocatori di videogame e gli amministratori di sistema Linux.

Le prestazioni, secondo gli sviluppatori, dovrebbero mettere Streamer ben al di sopra degli altri prodotti per desktop remoto compatibili con Ubuntu, da Vnc con Vinagre (<http://projects.gnome.org/vinagre>) a Remmina (<http://remmina.sourceforge.net/>) o TeamViewer (www.teamviewer.com/it/). Tutte le comunicazioni dello Streamer sono cifrate con codici Aes a 256 bit. Oltre ad aprire e modificare normali documenti molto più rapidamente dei suoi concorrenti, lo Streamer dovrebbe permettere di ricevere video fino a 30 fotogrammi al secondo, con un decimo della latenza. Sono supportati anche giochi sofisticati e conversione in tempo reale dei flussi audio o video ricevuti in streaming dal proprio desktop. Splashtop Streamer per Ubuntu consente infine di personalizzare il frame rate e impostare manualmente la porta Tcp con cui connettersi al desktop.

Rapporto delle Nazioni Unite elogia il Software Libero e il Linux Professional Institute

Uno studio pubblicato dalle Nazioni Unite a novembre 2012 LPI, sull'Information Economy e l'industria del software nei paesi in via di sviluppo, raccomanda fra le altre cose di dare adeguata attenzione al Software Libero e Open Source, soprattutto negli appalti pubblici. Il motivo è che tale software viene riconosciuto in grado di promuovere lo sviluppo del mercato nazionale e l'innovazione locale. Questo, a sua volta, aiuta a mantenere risorse preziose all'interno dell'economia locale, creando opportunità di lavoro ed evitando allo stesso tempo la dipendenza da singoli fornitori. Come esempio di tutto questo, la relazione cita esplicitamente l'opera nei paesi in via di sviluppo del Linux Professional Institute (www.lpi.org), tramite il suo programma mondiale di certificazione.

Samsung tradisce Android con Tizen

Tizen (www.tizen.org) è un sistema operativo basato su Linux ma ottimizzato per piattaforme mobili. Tizen è nato 2011, dalla fusione di altre piattaforme dello stesso tipo chiamate LiMo e MeeGo, al termine di una collaborazione fra Nokia e Intel su quest'ultima. Oggi lo sviluppo di Tizen ha come sponsor principali Intel e Samsung. A settembre 2012, quando è uscita la prima versione alfa di Tizen 2.0, si erano anche diffuse voci che Samsung stesse lavorando su uno smartphone Tizen, in contrasto con tutti gli altri suoi modelli Android. A gennaio 2013 Samsung e NTT Docomo, il più grande operatore mobile giapponese, hanno confermato che stanno lavorando insieme su un terminale Tizen, che dovrebbe essere sul mercato a fine anno, o a inizio 2014. Oltre a Samsung e Docomo, anche altri operatori e costruttori, da Vodafone a Panasonic e da France Telecom a Nec, hanno confermato di voler valutare terminali Tizen e contribuire al loro sviluppo.