

Corso "Sysadmin Linux" 2019

Lezione #2 - APT e Samba + Btrfs

Matteo F. Vescovi

Free Software Users Group Padova
<https://www.fsugpadova.org>

13 Novembre 2019
AViLUG - Schio (VI)



- 1 Il gestore di pacchetti APT (Advanced Packaging Tool)
- 2 Il server di condivisione file Samba (accoppiato al filesystem Btrfs)

- 1 Il gestore di pacchetti APT (Advanced Packaging Tool)
- 2 Il server di condivisione file Samba (accoppiato al filesystem Btrfs)

Le attività principali di *apt* riguardano la gestione dei pacchetti:

- *update*: aggiornamento del database dei repositories e dei pacchetti contenuti
- *upgrade*: aggiornamento dei pacchetti dei quali esiste una versione più aggiornata nei repositories
- *full-upgrade*: aggiornamento massivo del sistema con eventuale aggiornamento delle dipendenze richieste
- *install*: installazione di uno o più pacchetti dai repositories (ora anche locale)
- *remove* o *purge*: rimozione semplice o purga totale (con rimozione anche dei files di configurazione) di uno o più pacchetti dal sistema

Possono essere anche usate delle direttive extra al comando *apt* affinché faccia delle ulteriori operazioni sui pacchetti:

- *autoremove*: rimuove i pacchetti di cui non esistono più dipendenze inverse installate nel sistema (può essere accoppiato a *-purge*)
- *clean*: pulisce la directory di appoggio dove vengono scaricati i pacchetti prima di essere installati (ora viene fatto in automatico, se non diversamente specificato)
- *autoclean*: come sopra, ma rimuove solo i pacchetti che non possono essere più scaricati dai repositories
- *search*: compie delle ricerche all'interno della lista dei pacchetti disponibili
- *show*: mostra la descrizione del pacchetto selezionato

Oltre a ciò, *apt* può essere usato per gestire localmente i pacchetti scaricabili dai repositories.
Ad esempio:

- *download*: scarica il pacchetto selezionato senza però installarlo (→ si userà *dpkg*)
- *source*: scarica il pacchetto sorgente richiesto (deve essere specificato almeno un repository per i pacchetti sorgente)
- *policy*: descrive le possibili versioni in installazione per uno specifico pacchetto derivanti dai vari repositories
- *edit-sources* (sperimentale): modifica il file *sources.list* tramite l'editor di default

Ogni comando relativo ad APT va lanciato da **root** (sconsigliato!) oppure da un utente con permessi *sudo*:

```
$ sudo apt update
```

Per i tipi di comando che non richiedono l'installazione nel sistema di file, il comando *sudo* può essere omesso:

```
$ apt download remmina
```



- 1 Il gestore di pacchetti APT (Advanced Packaging Tool)
- 2 Il server di condivisione file Samba (accoppiato al filesystem Btrfs)

Sostanzialmente:

- permette di *condividere* una directory con altri utenti in rete locale con la finalità principale di mettere a disposizione degli utenti di client basati su piattaforma Windows i files contenuti in un server GNU/Linux
- risulta quindi *trasparente* all'utente finale che non percepisce la differenza rispetto alla memorizzazione dei dati su di un altro client o server Windows
- permette una definizione *granulare* dei *permessi* di accesso alle varie condivisioni in rete, anche tramite gruppi specifici o ACL



Cosa è il filesystem Btrfs

È pratica consolidata utilizzare una manciata di filesystem dimostratisi stabili ed affidabili negli anni. Tra questi i più importanti sono:

- ext4 (quarta versione del filesystem journaling per definizione)
- Btrfs (COW e journaling avanzato introdotto da Oracle nel 2009)
- JFS (journaling introdotto da IBM negli anni '90)
- ReiserFS (journaling introdotto nel 2001)
- XFS (journaling introdotto da SGI nel 1994)
- ZFS (journaling [e molto altro, e.g. COW] introdotto da Sun Microsystems nel 2005)

Nel caso specifico, noi descriveremo l'uso di *Btrfs*.



Lo snapshotting in Btrfs

- Una delle caratteristiche più interessanti del filesystem Btrfs è la possibilità di fare degli *snapshots* del disco, una sorta di istantanea dei suoi contenuti
- Questa funzionalità, accoppiata al sistema di condivisione Samba, permette di avere un controllo *temporale* su quanto è stato registrato sul disco con la possibilità di tornare a delle versioni precedenti di un file od una directory molto facilmente
- Solitamente, lo snapshotting viene messo in atto con dei cronjobs a cadenza multipla giornaliera, in modo da avere la possibilità di avere una maggior copertura delle variazioni apportate ai files
- Btrfs fa anche altri controlli (e.g. controllo coerenza online, SCRUB) senza necessità di smontare il disco preventivamente



Samba: il server di condivisione dei files (anche con Windows)

- Samba è il più famoso e utilizzato progetto F/LOSS per la condivisione di dati attraverso la rete basato sul protocollo SMB/CIFS
- Permette di condividere files e stampanti e si può tranquillamente sostituire alle strutture di Active Directory proprie delle varie versioni di Windows Server
- Samba, per sfruttare le potenzialità date dallo snapshotting del filesystem Btrfs, deve essere impostato per esportare le varie versioni memorizzate in maniera trasparente all'utente e recuperabili tramite i menu contestuali di Windows



Samba + Btrfs = shadow copies!

Vanno impostati dei parametri di condivisione, come il gruppo utenti abilitati all'utilizzo della condivisione e i permessi di lettura/scrittura; ad esempio:

```
[Documenti]
```

```
comment = Miei documenti
valid users = @users
writable = yes
force group = users
create mask = 0660
directory mask = 0770
vfs objects = btrfs shadow_copy2
path = /archivi/documenti
shadow:basedir = /archivi/documenti
shadow:snapdir = .snapshots
shadow:format = @GMT_%Y.%m.%d-%H.%M.%S
shadow:sort = desc
```



GRAZIE!