

Duplicazione di un sistema

Dopo aver chiuso la serie di articoli relativa alla gestione dei sistemi ritorno ad un articolo di taglio pratico, nato da un'esperienza sul campo, sulla duplicazione "a caldo" di un sistema.

Non so se sia molto frequente, ma penso che almeno una volta, nella vita lavorativa, di ogni sistemista sia capitato di dover duplicare l'hard disk di un PC o un server. Tale necessità può nascere, per esempio, dal bisogno di creare un backup di un sistema prima di aggiornarlo. Oppure per avere un clone di un sistema funzionante da tenere come scorta. Oppure ancora una copia su cui eseguire delle manutenzioni senza interrompere un servizio.

Per far fronte a queste esigenze sono disponibili molti programmi commerciali più o meno noti, ma anche l'open source offre delle soluzioni in tale senso, dallo spartano comando `dd` a distribuzioni 'live' dedicate a questo scopo. Nessuno di questi strumenti sembra, a quanto mi è dato di sapere, essere pensato per eseguire la duplicazione "a caldo" di un sistema mentre sta funzionando.

Quest'ultimo è proprio il caso che mi è capitato: La necessità era quella di aumentare lo spazio disco di un server FTP pubblico. Quindi si trattava di sostituire il disco esistente, possibilmente senza reinstallare completamente il sistema e preservando le varie configurazioni compresi gli utenti e le password. In pratica si aveva un sistema perfettamente funzionante ma con poco spazio su disco e ci si proponeva di sostituire il disco senza rischiare di alterare il sistema e renderlo non funzionante.

L'operazione è relativamente semplice quando è possibile fermare il server per il tempo necessario per duplicare l'hard disk e, magari, effettuarne le dovute copie di sicurezza, ma nel nostro caso, a complicare le cose, c'era la necessità di ridurre il fermo della macchina al minimo.

Normalmente per la duplicazione dei PC utilizzo una distribuzione live : [g4u](#) che permette di creare un'immagine di un PC su un server FTP. Purtroppo g4u richiede di eseguire il boot con il CD "live" e quindi introduce un fermo macchina per il tempo necessario alla copia; tempo che, per un HD da un centinaio di Gb, copiato attraverso una LAN da 100Mb, si aggira su un paio d'ore. Inoltre un ulteriore problema era dato dal fatto che il server FTP normalmente usato per salvare le immagini era proprio la macchina di cui si doveva effettuare la sostituzione del disco.

Sono, quindi, andato a sbirciare nel codice sorgente, degli script usati all'interno della distribuzione g4u per creare e trasferire le immagini, ed ho scoperto che si trattava di una serie di comandi standard uniti fra di loro in una serie di pipe in cascata. Colgo l'occasione per far notare che l'operazione è stata possibile proprio perché il sistema è Open Source. Andando a fondo della cosa e cercando ulteriori informazioni su internet ho trovato una serie di comandi che faceva al caso mio.

Prima di proseguire è d'obbligo un avviso:

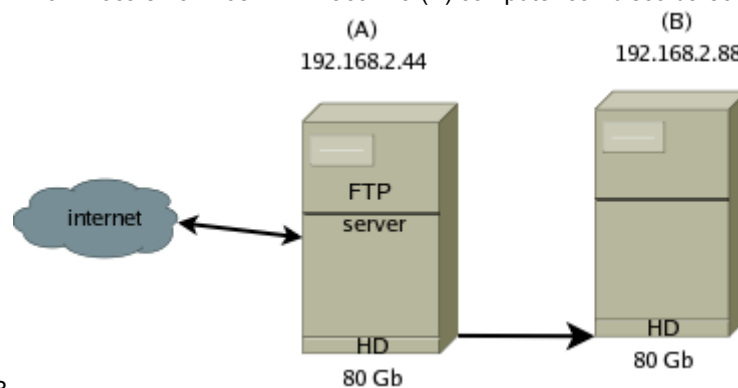
Prima di provare i comandi di seguito descritti, tenete presente che in caso di errore è elevato il rischio di sovrascrivere il contenuto di un disco di un server (o un PC) funzionante. E' quindi importante leggere completamente quanto scritto di seguito e aver chiari i concetti espressi prima di provare le procedure descritte.

Io stesso ho eseguito i test con macchine di prova prima di passare all'operazione reale.

Per duplicare a caldo un server, con questo sistema, ci sono dei prerequisiti:

- Il disco di destinazione deve essere di dimensione uguale o maggiore del disco di origine (come in qualsiasi duplicazione di disco).
- la macchina di origine deve avere il servizio `sshd` attivo e la macchina di destinazione deve poter utilizzare il comando `ssh`
- il disco di destinazione verrà sovrascritto completamente quindi non può essere lo stesso disco da cui si è effettuato il boot. La macchina di destinazione deve, quindi, avere un disco aggiuntivo su cui riversare il tutto oppure si può avviare la macchina con un cd live (g4u va bene) e poi utilizzare il suo HD come destinazione.

Nel nostro caso le operazioni sono state eseguite nelle seguenti condizioni: Macchina (A) il server FTP, in linea, con disco da 80 Gb da portare a 240 Gb. Indirizzo IP locale 192.168.2.44 Macchina (B) computer con disco da 80 Gb vuoto avviata con g4u da



CD all'indirizzo 192.168.2.88

Sulla macchina (B) è stato digitato il seguente comando:

```
ssh 192.168.2.44 'dd if=/dev/sda | dd of=/dev/sda'
```

in base alla seguente sintassi:

```
ssh source_server_ip 'dd if=/dev/sda1 | dd of=/dev/sda2
```

-- assumendo che il disco di origine sia sda1 e quello di destinazione sda2 (numeri aggiunti solo allo scopo di distinguere fra di loro le unità, non rappresentano in alcun modo la posizione fisica dei dischi)--

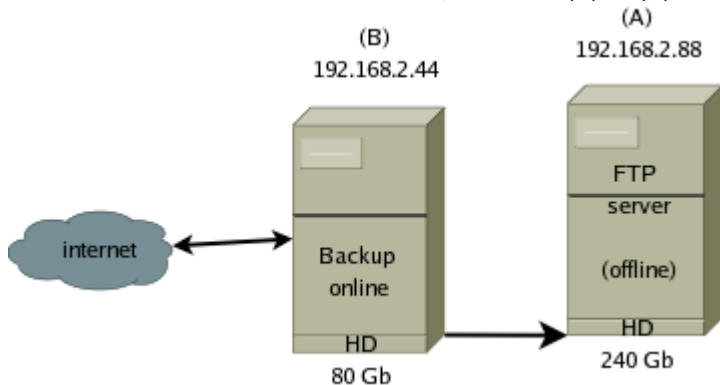
Vediamo il dettaglio: La prima parte del comando esegue attraverso **ssh** il comando **'dd if=/dev/sda'** (si notino gli apici) sul server di origine. Il comando **dd** esegue una copia byte a byte della sorgente (if) e in questo caso punta al device dell'hard disk del server. L' output del comando viene rediretto in pipe (|) verso la destinazione **dd of=/dev/sda** che in questo caso è l'hard disk del PC di destinazione. Il trucco consiste quindi nell'eseguire il primo dd in remoto via ssh e usare, attraverso il pipe, l'output per alimentare il secondo dd che viene eseguito in locale.

Dopo avere effettuato le operazioni di cui sopra ho riavviato il client (B) senza la connessione di rete per verificare il corretto riavvio del sistema e la configurazione. La rimozione del cavo di rete si è resa necessaria in quanto le due macchine, essendo, a questo punto, una la copia dell'altra, avevano lo stesso indirizzo IP.

Poi dopo aver verificato che non vi era nessun utente connesso, ho spostato il cavo di rete dal server originale (A) alla copia (B) e verificato che fosse accessibile normalmente.

Il sistema provvisorio a quel punto era in linea e funzionava al posto della macchina originale.

Dopo l'installazione fisica del nuovo hard disk è stata effettuata una nuova copia a caldo con le stesse modalità ma con le due macchine fisicamente scambiate di ruolo, ovvero da (B) ad (A).



In questo caso la copia è stata eseguita su un disco di

dimensioni maggiori che, quindi, risultava occupato solo parzialmente in quanto la copia trasferisce un immagine esatta del disco originale. Schematicamente la situazione era:

```
|-----disco originale 80Gb-----|-----spazio disponibile 160GB-----|
```

E' stato quindi riavviato il sistema con una versione live contenente varie utility in modo da effettuare il ridimensionamento del disco con qparted. Sicuramente ciò poteva essere effettuato a riga di comando ma questa utility grafica è enormemente più comoda.

Dopo un paio di riavvi a scopo di test i sistemi sono stati nuovamente scambiati di ruolo spostando il cavo di rete e rimettendo in linea il server ufficiale.

È importante notare che lo scambio delle macchine è stato possibile solo perchè il traffico era basso. Se le modifiche apportate dagli utenti nel tempo necessario alla copia sono elevate le due macchine rimangono disallineate e si rischia la perdita di informazioni.

In caso di sistemi a traffico elevato è opportuno pensare a soluzioni di clustering o ripartizione del carico in modo che sia possibile svincolare un nodo per il tempo necessario all'upgrade.

di [Rudi Giacomini Pilon](#)