

Chi ha la fortuna di avere in casa piu pc e si trova spesso nell'esigenza di dover compilare sorgenti che richiedono molto tempo per la compilazione, è possibile sfruttare il programma distcc. Distcc non è altro che un programma che serve per distribuire il lavoro richiesto per la compilazione di sorgenti scritti in C, C++ aumentando quindi la velocità. In realta il discorso è applicabile anche in uno scenario in cui le macchine non sono unicamente all'interno di una lan(10-100mbit), ma anche a macchine collegate tra loro attraverso internet o comunque attraverso un protocollo TCP. Ovviamente le prestazioni dipendono anche dalla velocità di trasmissione dei dati poiché le macchine si devono scambiare i sorgenti e le compilazioni parziali; piu veloce è il canale di trasmissione e maggiore sarà l'efficienza del sistema di compilazione. La cosa interessante di questo programma è che sfruttando più macchine per la compilazione, a lavoro finito, si ottiene esattamente lo stesso risultato che compilando con una sola macchina, in un tempo però molto inferiore. La flessibilità del programma sta nel fatto che non richiede particolari distribuzioni o librerie per le varie macchine, ma è sufficiente che su ogni macchina sia installato un cross-compiler (e distcc ovviamente): è comunque fortemente consigliato avere anche la stessa versione del compilatore su ogni macchina per evitare problemi o noie in fase di compilazione.

I suggerimenti che darò ora per la configurazione del programma si riferiscono alla distribuzione slackware, ma credo sia facilmente adattabile alle altre distribuzioni.

Se avete fatto una installazione completa di slackware vi troverete già tutti i programmi necessari sulla vostra macchina. Se non li avete, potete scaricare i pacchetti da un qualsiasi mirror di slackware, oppure se volete compilare i programmi è sufficiente seguire la solita procedura `./configure && make && sudo make install`. Se volete seguire le direttive ufficiali del pacchetto di slackware il configure è il seguente:

```
# ./configure --prefix=/usr --datadir=/usr/share/distcc --sysconfdir=/etc --with-gnome
```

Una volta installato il programma passiamo alla configurazione. Per prima cosa dobbiamo dire al programma quali nodi sono disponibili per la compilazione distribuita. Si può fare in piu modi: usando le variabili d'ambiente \$DISTCC_HOSTS, oppure \$DISTCC_DIR/hosts, altrimenti editando i file `~/.distcc/hosts`, oppure `/etc/distcc/hosts`. Se usiamo il metodo delle variabili d'ambiente un esempio è quello si esportare la variabile in questo modo:

```
# export DISTCC_HOSTS='localhost macchinal macchina2'
```

dove macchinal e macchina2 indicano 2 pc che saranno usati per la compilazione.

nel caso invece vogliate editare uno dei 2 file già citati, è sufficiente inserire in una riga di testo, il nome delle macchine separate da uno spazio.

Vi ricordo che nel caso non disponiate di un dns interno alla rete, dovete editare il file `/etc/hosts` in modo da associare all'ip delle macchine il loro nome aggiungendo ad esempio:

```
192.168.0.3          macchinal
192.168.0.4          macchina2
```

Se volete potete usare anche direttamente l'indirizzo e non il nome della macchina, evitando di dover mettere mano al file `/etc/hosts`. A questo punto distcc saprà con quali macchine potrà compilare nel momento opportuno.

Se usate una configurazione come quella appena presentata, sappiate che le comunicazioni tra le varie macchine avverranno sulla porta 3632. Nel caso vogliate cambiare la porta, lo potete fare usando una notazione del tipo `macchinal:3633 macchina2:3634` per spostare la comunicazione della macchinal sulla porta 3633 e della macchina2 sulla 3634. Se invece volete usare una comunicazione ssh potete usare la notazione `@macchinal` oppure `utente@macchinal` per usare l'utente appropriato. Usando ssh per il trasferimento dati comporta un perdita di velocità rispetto la comunicazione in chiaro ma è piu sicura e ha il vantaggio di non avere demoni in ascolto sulle varie macchine.

Nel caso vogliate settare invece un preciso numero di job massimi su ogni macchina, lo potete fare mettendo in coda ai nomi delle macchine un `/n` dove n è un numero intero che indica appunto il numero di job massimo.

A questo punto passiamo alla creazione dell'utente distcc per compilare con l'utente distcc nel caso lanciate la compilazione da root:

```
# echo "distcc:x:240:2:distccd:/dev/null:/bin/false" >> /etc/passwd
```

Seguiamo ora la procedura di avvio del programma nel caso di utilizzo di comunicazioni in chiaro.

Possiamo lanciare distcc in piu modi. In modalita standalone attraverso il comando

```
# distccd --daemon
```

oppure attraverso init aggiungendo a `/etc/inittab` la riga:

```
dscc:2345:respawn:/usr/bin/distccd --verbose --no-detach --daemon
(in questo modo il demone resterà in ascolto in automatico, all'avvio del pc)
```

oppure attraverso inetd aggiungendo a `/etc/inetd.conf` la riga:

```
distcc stream tcp nowait.6000 root /usr/bin/distccd distccd --inetd
(in questo modo il demone resterà in ascolto in automatico, all'avvio del servizio inetd)
```

Per un configurazione un po piu avanzata potete passare alcuni parametri all'avvio del programma:

```
-jN
dove N è un numero e indica il numero di lavori che possono essere accettati (per volta)
```

```
-p porta
per modificare la porta in ascolto del demone
```

```
--user USER
per modificare l'utente con viene lanciato distcc nel caso venga lanciato da root
```

```
--allow IPADDR[/MASK]
per fare in modo che vengano accettate connessioni da un ip specifico o una sottorete specifica
```

Una volta che avete tutte le macchine con il demone attivo e dopo aver configurato la macchina da cui parte la compilazione in modo che veda le altre macchine, è sufficiente che alla compilazione al momento del make scriviate:

```
# make -jN CC=distcc
```

mettendo al posto di N un numero intero: è consigliato scegliere un numero pari al numero di cpu complessive +1.

Ovviamente questo procedura comporta solo la distribuzione del codice che verrebbe compilato con CC.

Per rendere il sistema completamente flessibile per la compilazione è necessario allora fare in modo che distcc si occupi anche di GCC C++ e G++.

Per fare ciò possiamo seguire il consiglio del man distcc, andando a creare dei link simbolici di CC GCC C++ e G++ a distcc nel momento in cui lo desideriamo.

Faremo quindi:

```
# mkdir /usr/lib/distcc/bin
# cd /usr/lib/distcc/bin
# ln -s ../../../../bin/distcc gcc
# ln -s ../../../../bin/distcc cc
# ln -s ../../../../bin/distcc g++
# ln -s ../../../../bin/distcc c++
```

A questo punto però è necessario dire al sistema di seguire questi link andando ad esportare il PATH in questo modo:

```
# export PATH=/usr/lib/distcc/bin:$PATH
```

Andiamo così ad aggiungere la directory che contiene i link a distcc prima di quelli contenenti i veri gcc cc g++ e c++.

Così facendo distcc si occuperà di gestire interamente la compilazione.

Per monitorare l'utilizzo di distcc sulle varie macchine potete usare il programma `distccmon-gnome` (modalita grafica) oppure `distcc-text` (modalita testo).

Nella compilazione di alcuni programmi può accadere che la compilazione non venga distribuita: questo accade perché alcuni sviluppatori hanno disabilitato questa modalita di compilazione perché può introdurre dei problemi nella creazione del programma.

Per ulteriori chiarimenti e per una documentazione piu dettagliata vi consiglio di consultare [questa pagina](#) e `man distcc` e `man distccd`.

Devo anche fare un ringraziamento a [slash] per i chiarimenti e le info.

Anche per questa volta è tutto! Spero di avervi aiutato a ridurre i tempi delle compilazione!

A presto!