Manuale di installazione ed amministrazione del cluster Linux

Tiziano Fagni (tiziano.fagni@isti.cnr.it)
Giancarlo Bartoli (giancarlo.bartoli@isti.cnr.it)

24 marzo 2003

In questo rapporto tecnico descriveremo le caratteristiche principali del sistema di clustering Linux utilizzato dal gruppo di ricerca HPC (High Performance Computing) dell'istituto ISTI presso il CNR di Pisa.
Inizialmente faremo una panoramica dell'architettura del sistema descrivendo le caratteristiche principali dei nodi e i meccanismi utilizzati per permettere a questi ultimi di interagire con il resto del sistema. Il passo successivo sarà quello di descrivere la modalità di installazione di un nodo all'interno del cluster. Durante questa fase evidenzieremo le differenze principali di configurazione tra un semplice nodo "client" e il nodo "master". Saranno inoltre descritte le procedure di installazione dei principali software "non convenzionali" installati sul sistema. Queste applicazioni o librerie sono necessarie per realizzare alcune funzionalità particolari sul sistema di clustering. Infine, per ultimo descriveremo la sintassi e la semantica dei principali comandi non convenzionali a disposizione degli amministratori.

1 Architettura del cluster

Il cluster è composto da 9 nodi, ognuno avente le seguenti caratteristiche:

- scheda madre Intel SE7500WV2 con doppio processore Intel Xeon 2 Ghz;
- 1 Gigabyte di RAM;
- 1 disco EIDE (120 Gigabyte per il "master", 80 Gigabyte per gli altri nodi);
- 2 interfaccie di rete

In Figura 1 viene mostrata l'architettura di comunicazione del cluster. La macchina è stato costruita avendo in mente di rispettare alcuni requisiti. Il sistema è stato strutturato su due parti logiche distinte.

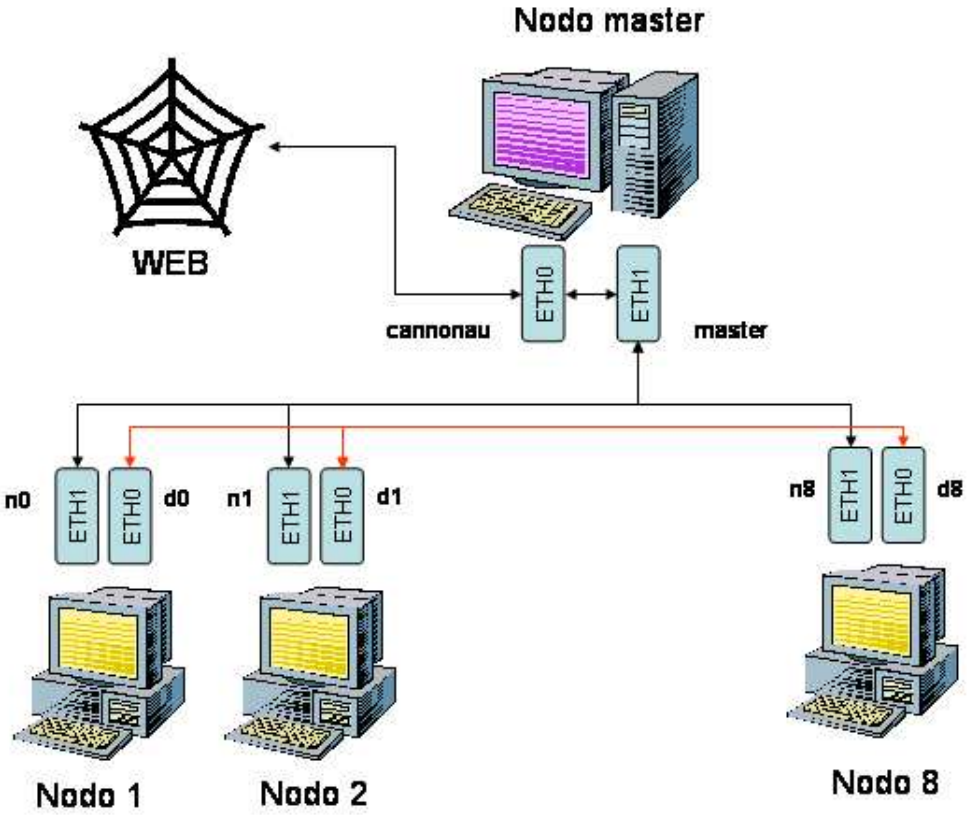

Figura 1: L'architettura logica del cluster

La prima parte, formata dal nodo master, rappresenta il front-end del cluster e permette agli utenti di collegarsi al sistema. Per l'accesso è stato scelto di utilizzare il servizio `ssh` in modo da fornire una shell sicura. Sul front-end sono inoltre presenti tutti gli strumenti di sviluppo necessari per la realizzazione delle applicazioni da testare sul cluster. Il nodo master ha anche il compito di gestire la *home* di tutti gli utenti e di renderla disponibile a tutti i nodi interni.

La seconda parte logica è formata dai nodi interni e mette a disposizione di ciascun utente le risorse di calcolo del cluster. Per sfruttare le risorse di ciascun nodo, ogni utente ha a disposizione sostanzialmente due possibili alternative. La prima consiste nel loggarsi sulla macchina che vogliamo utilizzare ed eseguire a mano le applicazioni che vogliamo testare. La seconda possibilità consiste nel sottomettere il nostro job al sistema di code (PBS) installato sul cluster. In quest'ultimo caso, sarà il sistema di code che deciderà per noi (in modo compatibile alla richiesta eseguita) dove e quando eseguire l'applicazione.

Ogni nodo del sistema di clustering presenta due interfaccie di rete.
Per quanto riguarda i nodi interni, la sottorete definita dall'interfaccia `eth0` è usata internamente dai programmi in esecuzione sul cluster e permette a questi ultimi lo scambio di dati nel modo più efficiente possibile grazie allo sfruttamento ottimale della banda disponibile. E' importante osservare che questa sottorete è accessibile esclusivamente dall'interno del cluster.
L'interfaccia `eth1`, invece, è utilizzata per connettere ogni nodo slave ad Internet. Questo è possibile impostando su questi ultimi, come gateway di default, l'indirizzo dell'interfaccia `eth1` del nodo master. Quest'ultimo nodo è collegato direttamente alla Rete tramite `eth0` e, grazie all'attivazione dei servizi di IP-masquerading e IP-forwarding, permette al traffico IP ricevuto dai nodi di essere inoltrato su Internet.
Il front-end del cluster, a differenza degli altri nodi, è collegato direttamente ad Internet tramite `eth0`. L'interfaccia `eth1` serve esclusivamente per far comunicare il front-end con i nodi interni e permette, come gia detto in precedenza, di condividere sia l'accesso a Internet sia eventuali altre risorse (ad esempio punti di mount NFS).

2 Installazione del cluster

2.1 Installazione di un singolo nodo slave

In questa sezione mostriamo i passi necessari per configurare un singolo nodo all'interno del cluster.

1. Installare una distribuzione (attualmente tutti i nodi contengono una RedHat 8.0) utilizzando le seguenti opzioni:

   (a) Partizionare il disco nel seguente modo:

- assegnare 256 Megabyte a "/boot";
- assegnare 2 Gigabyte a "/";
- assegnare 1 Gigabyte a "swap";
- assegnare il restante a "/extra".

(b) Scegliere di installare un insieme di pacchetti tipo il seguente:

- sistema di base;
- rsh;
- utility di sistema;
- utility di networking;
- NFS;
- emacs;
- in particolare, **NON** è necessario XWindow e tutti i programmi che fanno uso di questo ambiente. Inoltre, **NON** installare gli strumenti di sviluppo perchè la compilazione di programmi viene fatta solamente sul nodo "master".

(c) Impostare la password da amministratore uguale a quella del nodo master.

(d) Per quanto riguarda la configurazione di rete, utilizzare le seguenti impostazioni:

- disabilitare qualsiasi opzione di firewall;
- impostare l'interfaccia di rete "eth0" a: indirizzo IP 192.168.0.X (dove X è il numero di nodo in questione), netmask 255.255.255.0, nome dell'interfaccia a "dXX.hpclab.org" dove "XX" indica il numero della macchina (sullo chassis) su cui stiamo facendo l'installazione;
- impostare l'interfaccia di rete "eth1" a: indirizzo IP 192.168.1.X (dove X è il numero di nodo in questione), netmask 255.255.255.0, nome dell'interfaccia a "nXX.hpclab.org" dove "XX" indica il numero della macchina (sullo chassis) su cui stiamo facendo l'installazione;
- impostare il "default route" a 192.168.1.254;

2. Se durante la fase di installazione non sono state riconosciute le schede di rete, è necessario installare manualmente i relativi driver. Scaricare i driver direttamente dal sito della Intel (le schede di rete sono Intel PRO/1000) e seguire la procedura descritta all'interno dell'archivio. Le impostazioni fanno riferimento a quelle descritte precedentemente.

3. Configurare l'accesso in NFS al volume home del nodo master. Aggiungere la seguente riga al file */etc/fstab*:

```
192.168.1.254:/home  /home    nfs     auto,rw  0 0
```

4. Configurare il server *rsh* in modo che il nodo sia accessibile da remoto senza richiesta di password da ciascun utente autorizzato.

   - Attivare i servizi `rsh, rexec, rlogin` e aggiungere al file /etc/securetty le seguenti righe:

     ```
     rsh
     rexec
     rlogin
     ```

   - Aggiungere al file /etc/hosts.equiv tutti i nomi dei nodi del cluster più quello del master.
   - Creare il file /root/.rhosts contenente le seguenti righe:

     ```
     master root
     n01 root
     n02 root
     n03 root
     n04 root
     n05 root
     n06 root
     n07 root
     n08 root

     d01 root
     d02 root
     d03 root
     d04 root
     d05 root
     d06 root
     d07 root
     d08 root
     ```

5. Attivare i seguenti servizi:

   crond Utilizzato per eseguire comandi in intervalli di tempo prestabiliti.

   gpm Utilizzato per attivare il mouse in modalità console.

   nfs, portmap, rpc.statd Utilizzati per accedere al file system ditribuito `/home` e per fornire, in modo corretto, i servizi `rsh` e `rlogin`.

   ntpd Usato per sincronizzare l'orologio interno della macchina con quello degli altri nodi. In particolare, ogni nodo del cluster (compreso il master) utilizza il server NTP `time.ien.it`.

   syslogd, klogd Usati per loggare gli eventi verificatisi sul sistema. In particolare, per monitorare lo stato del cluster, è importante modificare

il file `/etc/syslog.conf` in modo che tenga traccia anche degli eventi occorsi al kernel. All'interno del suddetto file, inserire la seguente riga:

```
# Log all kernel messages to a specified file.
kern.*  /var/log/kernel.log
```

Ovviamente, per rendere effettive le modifiche, occorre riavviare il demone `syslogd`.

xinted Utilizzato per attivare i vari servizi Internet. In questo caso, attivare i sottoservizi di `rlogin`, `rsh` e `rsync`.

6. Sincronizzare tutti i file di configurazione del sistema con quelli degli altri nodi. Utilizzare, a questo scopo, lo script *ssync* presente sul master (per maggiori dettagli fare riferimento alla sezione 5.1).

2.2 Installazione del master

L'installazione del master è simile a quella di un singolo nodo. Vi sono, però, alcune differenze:

1. Il disco è stato partizionato in questo modo: 80Mb per `/boot`, 15Gb per `/`, 1Gb per swap e il restante per `/home`.

2. Per quanto riguarda il tipo di installazione, sono stati installati, oltre ai pacchetti presenti su un singolo nodo, anche XWindow, i vari demoni di networking (Apache, sendmail, etc..) e tutti gli strumenti (anche grafici) per la configurazione della macchina. Inoltre, sono stati installati anche gli strumenti per sviluppare software (gcc, perl, ecc.).

3. Le schede di rete sono state impostate in questo modo: **eth0** a IP 146.48.82.190, netmask 255.255.248.0, broadcast 146.48.83.255, default 146.48.80.1, nome *cannonau.isti.cnr.it* mentre **eth1** a IP 192.168.1.254, netmask 255.255.255.0, broadcast 192.168.1.255, nome *master.hpclab.org*.

4. Sono stati attivati il masquerading e il forwarding dei pacchetti IP. Aggiungere al file /etc/rc.local le seguenti righe:

```
modprobe ipt_MASQUERADE
iptables -F; iptables -t nat -F; iptables -t mangle -F
iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 146.48.82.190
echo 1 > /proc/sys/net/ipv4/ip_forward
```

3 Installazione di software aggiuntivi

I software da aggiungere al cluster devono essere compilati sul nodo master. Le eventuali librerie condivise e/o i demoni necessari al funzionamento del pacchetto, devono essere copiati su tutti i nodi in modo che le applicazioni funzionino correttamente.

3.1 STLport

La libreria è scaricabile all'indirizzo *www.stlport.com*. La versione attualmente installata è la 4.5.3 e le istruzioni che seguono fanno riferimento a questa versione.

1. Scompattare il file *tgz* su una directory temporanea e posizionarsi all'interno di `STLport-4.5.3/src`.

2. Al prompt dei comandi, digitare `make -f gcc-linux prepare all`.

3. Rendere la directory `STlport-4.5.3/lib` visibile a `ld`.

3.2 ACE

La libreria ACE può essere installata secondo due diverse modalità operative. La prima prevede che ACE utilizzi la libreria STL standard fornita con il compilatore GNU gcc. In questo caso, per l'installazione, è necessario seguire le istruzioni riportate in ACE-INSTALL.html, un file contenuto all'interno del pacchetto tgz della distribuzione ACE. La seconda modalità permette di installare ACE facendo in modo che questa utilizzi STLport come libreria STL. I passi da seguire per fare un'installazione di questo tipo sono i seguenti:

1. Scompattare l'archivio tgz.

2. Spostarsi in `ACE_wrappers/ace` e eseguire `ln -s config-linux.h config.h`

3. Aprire `config-g++-common.h` con un editor di testi e commentare la riga con `#define ACE_USES_OLD_IOSTREAMS`. Subito sotto inserire la riga `#define ACE_USES_STD_NAMESPACE_FOR_STDCPP_LIB 1`. A questo punto salvare il file ed uscire.

4. Andare in `ACE_wrappers/include/makeinclude` e eseguire

   ```
   ln -s platform_linux.GNU platform_macros.GNU
   ```

5. Aprire platform_macros.GNU e trasformare la riga 'CFLAGS+=...' in
   `CFLAGS+=-I<directory_STLport>/stlport ....`
   Successivamente modificare la riga 'LIBS ++=.....' in

`LIBS += -L<directory\_STLport>/lib -lstlport\_gcc .....`
In entrambi i casi *directory_STLport* rappresenta la directory dove si trovano i sorgenti della STLport.

6. Andare nella directory principale 'ACE_wrappers' e fare
`export ACE_ROOT=<path_directory_principale>`
dove *path_directory_principale* è la directory principale su cui si trovano i sorgenti di ACE.

7. In 'ACE_wrappers' digitare 'make'.

8. Rendere la directory `ACE_wrappers/ace` visibile a `ld`.

3.3 Il software PBS

Il software OpenPBS (Open Portable Batch System) serve a realizzare un sistema di code tramite il quale eseguire i vari job sul sistema di clustering. Il sistema è molto flessibile e permette all'utente di specificare in dettaglio come e quando eseguire uno specifico task.
Per installare il software (attulmente la versione 2.3), è necessario scaricare dal sito www.openpbs.org due pacchetti RPM: il pacchetto
`openpbs-2.3pl2-1.i386.rpm`
deve essere installato sul nodo master mentre
`openpbs-exechost-2.3pl2-1.i386.rpm`
deve essere installato su ciascun nodo del cluster.
Dopo aver installato i pacchetti tramite il tool `rpm`, dobbiamo apportare alcune modifiche alla configurazione di PBS:

1. Indichiamo a PBS quali sono i nodi del cluster su cui può schedulare i processi. Editiamo il file `/usr/spool/PBS/server_priv/nodes` sul nodo master e inseriamo le seguenti righe:

```
n01:ts np=4 ISTICluster
n02:ts np=4 ISTICluster
n03:ts np=4 ISTICluster
n04:ts np=4 ISTICluster
n05:ts np=4 ISTICluster
n06:ts np=4 ISTICluster
n07:ts np=4 ISTICluster
n08:ts np=4 ISTICluster
```

Osserviamo che il parametro `np` serve per specificare il numero di processori virtuali presenti su un singolo nodo.

2. Dobbiamo permettere l'utilizzo del sistema di code esclusivamente dall'interno del cluster. Per fare questo, eseguiamo `qmgr` dal nodo master e digitiamo i comandi
   ```
   set server acl_hosts=*.hpclab.org
   set server acl_host_enable=true
   ```

3. Su ciascun nodo (escluso il master) editiamo il file `/usr/spool/PBS/mom_priv/config` e inseriamo le seguenti righe:

   ```
   # Logs all events except 'debug events'
   $logevent 255

   # The main server of PBS system
   $clienthost master.hpclab.org
   ```

4. È necessario avviare il servizio `pbs` su ciascun nodo del cluster (compreso il master). Aggiungere la seguente riga al file `/etc/rc.local`:

   ```
   service start pbs
   ```

3.4 La libreria LAM / MPI

La libreria LAM è una implementazione efficiente, opensource e multipiattaforma di MPI. LA versione attualmente installata sul cluster è la 6.5.6 e può essere scaricata dal sito www.lam-mpi.org. Dopo aver scaricato il file RPM contenente la libreria, è necessario installarla sul sistema:

```
rpm -i <lam-file>
```

Osserviamo che l'installazione della libreria è necessaria sia sul master sia sui generici nodi del cluster. Il passo successivo è quello di indicare a LAM quali saranno le macchine utilizzate dal cluster per eseguire i processi. A questo scopo, editiamo il file `/etc/lam/lam-bhost.def` sul nodo master ed inseriamo le seguenti righe:

```
n01 cpu=4
n02 cpu=4
n03 cpu=4
n04 cpu=4
n05 cpu=4
n06 cpu=4
n07 cpu=4
n08 cpu=4
```

4 Modalità di accesso al cluster

Il cluster è accedibile dall'esterno attraverso il nodo front-end (cannonau.isti.cnr.it, 146.48.82.190). Per loggarsi nel sistema è necessario utilizzare il servizio *ssh*. Questo permette di mettere a disposizione di ogni utente autorizzato una shell criptata. Per esempio, per collegarsi al cluster come utente "utente" eseguire

```
ssh utente@cannonau.isti.cnr.it
```

e, quando richiesto, digitare la password corretta. Ricordiamo che per richiedere un account valido è necessario contattare gli amministratori del sistema (vedi sezione 6).

Una volta loggati nel sistema, è possibile accedere ai vari nodi del cluster attraverso il servizio *rsh*. Ogni nodo interno, come è stato descritto nella sezione 1, possiede due interfacce di rete che vengono utilizzate per creare due reti distinte.

La prima, pubblica e visibile anche dal nodo master, permette di far comunicare quest'ultimo con tutti gli altri nodi. In questo caso, ogni nodo ha un nome del tipo `nX` dove `X` è il numero di nodo considerato, antecedendo lo 0 nel caso `X` sia minore di 10.

La seconda rete è accessibile ai soli nodi interni e deve essere usata per far comunicare applicazioni distribuite in esecuzione sul cluster. I nomi utilizzati all'interno di questa rete seguono il formato `dX` dove il parametro `X` ha lo stesso significato del caso precedente.

Alla luce di quanto è stato detto, è possibile accedere ad un nodo interno, a partire dal master, eseguendo `rsh nX` dove X è il nodo da accedere. Viceversa, da ogni nodo interno è possibile accedere a ciscun altro nodo eseguendo sia `rsh dX` sia `rsh nX` dove `X` è la macchina destinazione.

La *home* di ogni utente è esportata in NFS dal nodo master su tutti i nodi. Questo permette di semplificare la gestione dei dati perchè i file locali sono acceduti in modo centralizzato indipendentemente dal nodo su cui un utente è loggato. Su ciascun nodo è inoltre disponibile una directory locale (in `/extra` e corrispondente al nome dell'utente) che può essere utilizzata per memorizzare dati critici che necessitano, per motivi di efficienza, di essere acceduti direttamente dall'hard-disk.

5 Comandi non convenzionali disponibili

Ecco una lista di comandi non convenzionali che l'utente del cluster può utilizzare per svolgere alcune operazioni.

5.1 Amministrazione del sistema

clusteradduser Il comando è a disposizione degli amministratori per aggiungere un nuovo utente sul cluster. Si tratta di uno script che wrappa il comando

standard `adduser` e sincronizza correttamente i file di configurazione interessati su tutti i nodi del cluster. L'unica limitazione del comando è che il nuovo utente ha la password disabilitata (nulla). È necessario quindi, in una fase successiva, aggiungere la password desiderata ed eseguire un aggiornamento dei file di configurazione sui nodi tramite il comando `ssync`.

ssync Talvolta è necessario fare qualche modifica alla configurazione del cluster perciò è importante sincronizzare i file di configurazione su tutti nodi. Il comando ssync, disponibile solo come superutente sul master, serve proprio a questo. Al prompt dei comandi del master, digitare *ssync*. Il comando utilizza due file per poter operarare correttamente. Il primo, `/etc/shosts`, permette di decidere su quali nodi fare l'aggiornamento mentre il secondo, `/etc/sfiles`, specifica quali sono i file di configurazione da aggiornare.

5.2 Amministrazione di PBS

qmgr Il comando permette di amministrare il sistema di code. Per maggiori informazioni consultare la man page.

6 Contatti

Per la richiesta di nuovi account e chiarimenti sulla configurazione ed utilizzo del cluster, potete contattare gli amministratori del sistema:

- Giancarlo Bartoli (giancarlo.bartoli@isti.cnr.it)
- Tiziano Fagni (tiziano.fagni@isti.cnr.it)
- Salvatore Orlando (orlando@dsi.unive.it)
- Paolo Palmerini (paolo.palmerini@isti.cnr.it)
- Raffaele Perego (raffaele.perego@isti.cnr.it)
- Fabrizio Silvestri (fabrizio.silvestri@isti.cnr.it)

7 APPENDICE

7.1 Configurazioni testate

In questa sezione mostriamo le configurazioni hardware-software testate e gli eventuali relativi problemi riscontrati. Osserviamo che, in ogni macchina, il BIOS della schedra madre è stata aggiornato all'ultima versione disponibile.

7.1.1 Due dischi in configurazione RAID-0

1. Mandrake 9.0
   L'installazione non riconosce il controller del disco e non c'è modo di utilizzare un floppy per caricare i relativi driver (sia di origine Intel sia di origine Promise).

2. RedHat 8.0
   Il kernel va in crash durante la fase di installazione subito dopo aver riconosciuto il controller del disco.

3. RedHat 7.3
   Provate varie configurazioni:

   (a) Eseguita procedura di installazione normale. In questo caso, il sistema Linux viene installato correttamente ma durante l'uso presenta "strani" errori di accesso al disk e conseguenti crash della macchina.

   (b) Il sistema è stato installato seguendo una procedura di installazione scaricata dal sito della Intel ma durante il funzionamento talvolta va in crash a causa di un bug del kernel (il modulo "ft" della Fasttrack, per la precisione). In ogni caso, dopo poco più di 3 giorni di test, il benchmark *iozone* sembra funzionare correttamente.

   (c) Eseguita procedura di installazione normale caricando però il modulo FastTrack (scaricato dal sito della Promise) in fase di inizializzazione dell'installazione. Sul sistema è stato eseguito, con esito positivo, il benchmark *iozone* per un periodo di tempo di circa 4 giorni.

7.1.2 Un disco in configurazione EIDE

Installata la RedHat 8 senza particolari problemi. Eseguendo il kernel monoprocessore abbiamo notato uno strano comportamento dello strato software "usb" che, in pratica, non riesce ad inizializzarsi. Questo comporta che la fase di caricamento di Linux rimanga bloccata sullo script di gestione dell'USB. L'unico rimedio valido a questo problema è stato quello di disabilitare questo supporto. Osserviamo, inoltre, che nel caso venga eseguito il kernel multiprocessore, il suddetto problema non si verifica.