# Panduan Singkat

# Implementasi Linux Fedora untuk Pengembangan Cluster

oleh : Gentur Widyaputra (gentur\_widyaputra@yahoo.com)

editing naskah : Sunu Wibirama, S.T. (sunu@mail.te.ugm.ac.id)

#### Tim HPC Universitas Gadjah Mada

http://hpc.ugm.ac.id http://grid.te.ugm.ac.id

Copyright © HPC UGM 2008

### 1. Pendahuluan

Tulisan ini adalah dokumentasi dari infrastruktur cluster yang dibangun oleh tim HPC Teknik Elektro UGM. Sebelum membahas mengenai teknis pembuatan infrastruktur cluster, terlebih dahulu akan dibahas mengenai perencanaan dan perancangan cluster yang akan dibuat. Cluster yang kami bangun adalah cluster yang ditujukan untuk HPC atau High Performance Computing. Cluster ini harus memiliki kemampuan untuk melakukan proses komputasi secara parallel. Mekanisme komputasi paralel yang kami gunakan adalah MPI (Message Passing Interface). Dengan MPI, sebuah proses komputasi dapat di distribusikan ke mesin - mesin komputasi yang lain.

#### 2. SSH

Untuk mengimplementasikan HPC cluster yang memanfaatkan MPI sebagai mekanisme pendistribusian proses, dibutuhkan sebuah Dedicated Private Network. Di dalam jaringan privat tersebut, perlu dijalankan service-service yang mengintegrasikan mesin-mesin yang ada pada jaringan tersebut. Service-service tersebut adalah NFS, NIS SSH dan MPI.

NFS digunakan sebagai global file system yang dapat diakses oleh semua mesin di dalam jaringan cluster. NIS digunakan sebagai sistem administrasi file-file (passwd,group,user,dll) terpusat termasuk administrasi user. SSH digunakan sebagai sistem komunikasi untuk menjalankan sebuah proses di mesin-mesin komputasi. MPI digunakan sebagai mekanisme pendistribusian proses komputasi.

Jaringan Cluster yang kami bangun memiliki sebuah head node dan dua buah compute node. **Head Node** adalah mesin yang digunakan sebagai inisialisasi proses atau pemberi proses ke mesin-mesin Compute Node. **Compute Node** adalah mesin-mesin yang digunakan untuk menjalankan suatu proses komputasi yang diberikan oleh Head Node. Topologi jaringan berbentuk star. Head Node memiliki dua buah NIC (Network Interface Card), satu NIC terhubung ke jaringan cluster dan NIC yang lain terhubung ke jaringan internet supaya dapat diakses oleh user atau pengguna cluster. Compute Node hanya memiliki sebuah NIC saja yang menghubungkan ke jaringan cluster. Masing-masing mesin di dalam jaringan cluster kami beri sistem penamaan sesuai dengan nama host-nya. Head Node kami berinama cluster0, dua Compute Node yang lain kami berinama cluster1 dan cluster2. Untuk memberikan nama pada mesin cukup dengan mengedit file /etc/sysconfig/network pada bagian HOSTNAME="nama host" (untuk distro linux yang berbeda, sangat mungkin berbeda pula letak file konfigurasinya).

Mesin Head Node perlu mengetahui nama-nama host mesin-mesin komputasi, untuk itu perlu di konfigurasi file /ets/hosts dengan memberikan nama host dan ip dari mesin - mesin komputasi (baca man hosts).

Sistem operasi yang digunakan di dalam cluster sebaiknya sama. Kami menggunakan Linux Fedora Core 6. Penggunaan sistem operasi yang sama dimaksudkan untuk mempermudah di dalam administrasi sistem.

#### 3. NFS server

Cluster membutuhkan sebuah file system yang dapat diakses secara global oleh semua mesin di dalam jaringan cluster. Untuk itu digunakan NFS (Network File System). NFS adalah sebuah aplikasi client server yang digunakan untuk mengeksport sebuah local file system ke jaringan sehingga file system tersebut dapat diakses oleh mesin2 yang lain.

Di dalam cluster yang kami buat, Head node kami jadikan NFS server sedangkan Compute Node sebagai NFS client. Directory yang dieksport adalah /home dan /usr/local. Kedua directory tersebut terletak pada partisi hard disk yang terpisah. Di bawah ini adalah partisi hard disk Head node kami:

/	ext3	9.5G	
/usr/local	ext3	29G	
/home	ext3	29G	
/var	ext3	1.9G	
/tmp	ext3	1.9G	
		2G	
	/ /usr/local /home /var /tmp	/ext3/usr/localext3/homeext3/varext3/tmpext3	/ ext3 9.5G /usr/local ext3 29G /home ext3 29G /var ext3 1.9G /tmp ext3 1.9G _2G

Untuk menjalankan sebuah NFS server diperlukan service portmap,nfsd,mountd. File konfigurasi yang perlu diubah adalah /etc/exports, /etc/hosts.allow dan /etc/hosts.deny Langkah kofigurasi /etc/exports :

```
# mengeksport /home
/home 192.168.0.0/24 (rw,no_root_squash,sync)
# mengeksport /usr/local
/usr/local 192.168.0.0/24(ro,no_root_squash,sync)
```

Konfigurasi /etc/hosts.allow :

#

"		
#	hosts.allow	This file describes the names of the hosts which are
#		allowed to use the local INET services, as decided
#		by the '/usr/sbin/tcpd' server.
#		
po	ortmap: 192.168	3.0.0/255.255.255.0

Konfigurasi /etc/hosts.deny :

Setelah semua file diatas dikonfigurasi, restart service portmap dan nfs :

```
# /etc/init.d/portmap restart
# /etc/init.d/nfs restart
```

Untuk melihat apakah file system telah berhasil dieksport dapat dilakukan dengan perintah showmount -e localhost atau dengan melihat apakah service-service nfs, portmap dan mountd berjalan dengan memberikan perintah rpcinfo -p.

#### 4. NFS client

NFS client dijalankan oleh mesin komputasi. Sebelum mengimpor file sistem dari NFS server directory /home dan /usr/local perlu diback up.

```
# mv /home /home.old
# mv /usr/local /usr/local.old
```

Setelah melakukan back up, buat directory baru /home dan /usr/local.

```
# mkdir /home
# mkdir /usr/local
```

Setelah NFS server telah berjalan perlu dilakukan konfigurasi pada NFS client pada mesin-mesin komputasi. Edit file /etc/fstab pada setiap mesin komputasi :

```
      192.168.0.1:/home
      /home
      nfs
      rw,sync,hard,intr
      0 0

      192.168.0.1:/usr/local
      /usr/local
      nfs
      ro,sync,hard,intr
      0 0
```

#### 5. NIS server

NIS atau Network Information System adalah sebuah servis di dalam jaringan yang memungkinkan untuk melakukan manajemen user dan konfigurasi-konfigurasi file secara terpusat. File-file yang dapat diatur secara terpusat diantaranya adalah /etc/passwd, /etc/shadow, /etc/ groups, /etc/hosts, /etc/services, /etc/networks, /etc/rpc, /etc/protocols,/etc/ aliases.

Komputer Head Node akan dijadikan sebagai NIS server sehingga mempermudah dalam melakukan pengaturan terhadap mesin-mesin Compute Node. Dengan adanya NIS server, penambahan node juga akan lebih mudah. Untuk membuat head node sebagai NIS server diperlukan paket paket aplikasi sebagai berikut :

- 1. ypbind
- 2. portmap
- 3. ypserv
- 4. yp-tools
- 5. nscd

Konfigurasi file /etc/sysconfig/network :

NETWORKING\_yes NETWORKING\_IPV6=yes HOSTNAME=cluster0 NISDOMAIN="NIGHTLOGIN"

Konfigurasi file /etc/yp.conf :

domain NIGHTLOGIN server 127.0.0.1

Konfigurasi /etc/nsswitch.conf :

passwd: files nis
shadow: files nis
group: files nis

Konfigurasi /etc/ypserv.conf :

```
dns: no
files: 30
slp: no
slp_timeout: 3600
xfr_check_port: yes
* : * : shadow.byname : port
* : * : passwd.adjunct.byname : port
```

Setelah file-file tersebut dikonfigurasi, lakukan langkah-langkah berikut:

```
# nisdomainname NIGHTLOGIN
# /etc/init.d/portmap restart
# /etc/init.d/yppasswdd start
# service ypserv start
# /usr/lib/yp/ypinit -m
# make -c /var/yp
# /etc/init.d/ypbind start
```

# 6. NIS Client

NIS client adalah mesin2 komputasi.

Untuk menjalankan NIS client diperlukan service-service :

- 1. ypbind
- 2. portmap
- 3. yp-tools
- 4. nscd

Konfigurasi file /etc/sysconfig/network :

NETWORKING=yes NETWORKING\_IPV6=yes HOSTNAME=cluster0 NISDOMAIN="NIGHTLOGIN"

Konfigurasi file /etc/sysconfig/yp.conf :

```
domain NIGHTLOGIN server 192.168.0.1
```

Konfigurasi file /etc/nsswitch.conf :

passwd: files nis
shadow: files nis
group: files nis

Setelah mengkonfigurasi file-file diatas jalankan perintah:

```
# nisdomainname NIGHTLOGIN
# /etc/init.d/portmap start
# /etc/init.d/ypbind start
# /etc/init.d/nscd start
# /etc/init.d/yppasswd start
```

Apabila Anda ingin menambah user di dalam jaringan cluster, cukup dengan menambah user di mesin Head Node kemudian lakukan perintah make di directory /var/yp

# 7. Konfigurasi SSH

SSH atau Secure Shell digunakan sebagai sarana komunikasi antar proses di dalam cluster. Head Node dan Compute nNode harus saling percaya satu sama lain, sehingga tidak perlu dilakukan proses authentikasi jika Head Node akan berkomunikasi dengan Compute Node. Head Node harus membuat public dan private key untuk diberikan ke mesin2 komputasi.

```
headnode$mkdir .ssh
headnode$cd .ssh
headnode$ssh-keygen -t rsa
(akan muncul isian2, tekan enter saja sampai selesai)
headnode$cp id_rsa.pub authorized_keys
(Kita tidak perlu melakukan pengkopian id_rsa.pub ke mesin2 komputasi lain
karena user dan directory di setiap mesin adalah sama)
```

Setelah menjalankan perintah diatas, head node dapat melakukan ssh ke mesin komputasi tanpa authentikasi. Konfigurasi SSH tersebut di atas *harus dilakukan pada setiap user*.

# 8. LAM-MPI

LAM-MPI adalah salah satu aplikasi yang mengimplementasikan MPI. Terdapat aplikasi lain yang mengimplementasikan MPI, diantaranya adalah OpenMPI dan MPICH. LAM-MPI dapat di download dari <u>www.lam-mpi.org</u>. Instalasi cukup dilakukan oleh Head Node saja.

Instalasi LAM-MPI:

```
#tar -xzvf lam-7.1.4.tar.gz
#./configure _prefix=/usr/local
(pastikan semua aplikasi yang akan dijalankan oleh semua mesin komputasi
diinstal di prefix=/usr/local)
#make
#make install
```

Setelah selesai menginstall LAM MPI, edit file /usr/local/etc/lam-bhost.def dengan memberikan nama-nama host setiap mesin yang ada di dalam cluster. Setelah itu jalankan perintah lamboot -v. LAM-MPI hanya dapat dijalankan oleh user biasa. Untuk mencoba aplikasi MPI silakan cari tutorial tentang pemrogramman mpi. Good luck !

----==