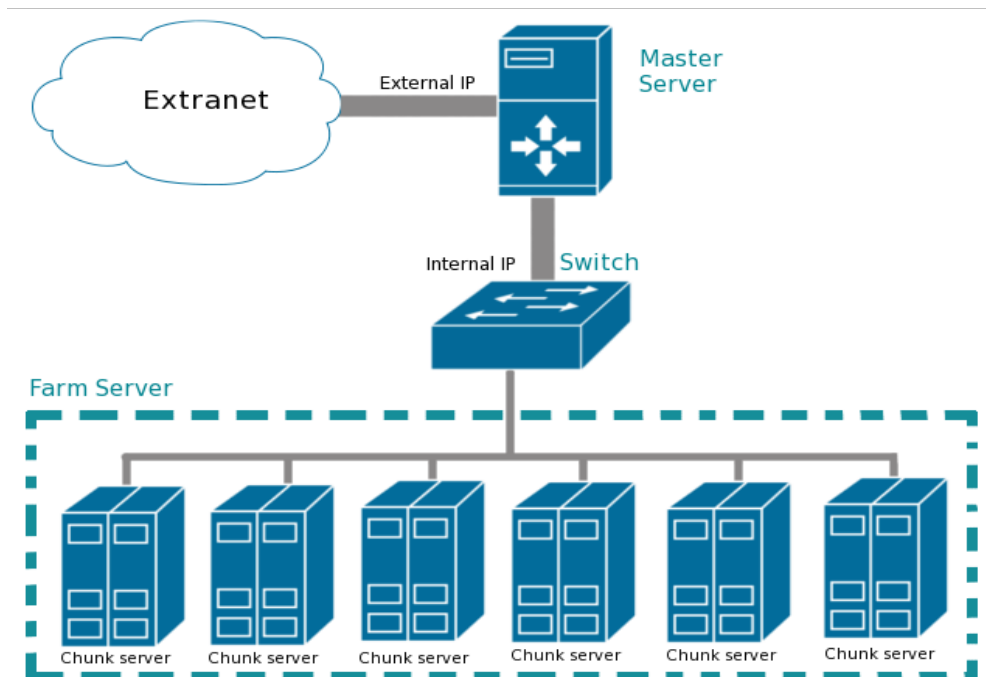


Berkenalan dengan Dunia Clustering

Membangun Automatic Storage Clustering
berbasis MooseFS



Oleh :

Bramandityo Prabowo

Pendayagunaan Open Source Software

Universitas Pendidikan Indonesia

2010

Daftar Isi

Daftar Isi	hal 1
Pendahuluan	hal 2
Persiapan	hal 3
Kebutuhan Perangkat Keras (Hardware)	hal 3
Kebutuhan Perangkat Lunak (Software)	hal 4
Skema Jaringan Komputer	hal 5
Implementasi	hal 6
Modifikasi SLAX	hal 6
Instalasi dan konfigurasi Sistem PXE Boot	hal 11
Instalasi dan konfigurasi mfsmaster pada master server	hal 13
Menjalankan semua service dan menggunakan Storage Clustering	hal 14
Monitor Storage Clustering	hal 15
Daftar Pustaka	hal 16
Tentang Penulis	hal 17

Pendahuluan

Storage clustering adalah sebuah media penyimpanan maya yang disusun oleh beberapa server (farm server) yang menjalankan fungsi penyimpanan bersama-sama. Secara logika farm server yang berkolaborasi tadi tampak membentuk sebuah media penyimpanan data yang besar.

Storage Clustering memiliki fitur yang sama dengan clustered computing, salah satu diantaranya adalah scalable dan reliable. Hal ini yang menjadikan teknologi ini banyak dipakai oleh perusahaan-perusahaan baik berskala besar atau menengah. Dengan storage clustering kita dapat dengan mudah menambahkan ukuran media penyimpanan saat sistem sedang berjalan.

Di dalam dunia storage clustering dikenal dua macam server, Master server dan Chunk server. Master server adalah server yang melakukan manajemen ke anggota farm server. Master server juga disebut integrator. Pengguna yang mengakses Master server tidak akan menyadari bahwa mereka sebenarnya sedang mengakses farm server. Jadi fungsi master adalah membuat sebuah lapisan abstrak yang menyediakan kemampuan akses ke farm server. Sedangkah chunk server adalah anggota farm server yang menyumbangkan volume hard disk nya untuk di satukan oleh master server menjadi sebuah media penyimpanan logical yang sangat besar.

Storage clustering ini tidak menggunakan hanya satu server untuk menjalankan fungsi penyimpanan datanya. Namun menggunakan lebih dari satu server. Dengan metode ini dapat dihindari hilangnya data saat salah satu server mengalami gangguan, karena server yang lain akan menggantikannya. Semakin banyak server yang digunakan sebagai farm server , semakin berat kerja sistem administrator. Untuk setiap server yang masuk kedalam farm server, sistem administrator harus menginstallkan sistem operasi baru. Cara ini sangat tidak efisien.

Pada modul kali ini akan dijelaskan bagaimana membangun sebuah storage clustering dari perangkat keras murah (bekas) namun dengan hasil maksimal. Selain itu dalam modul ini juga dijelaskan bagaimana cara membangun automatic storage clustering. Maksudnya adalah kita tidak usah menginstall sistem operasi apapun di komputer yang akan dijadikan chunk server. Sistem yang dibangun memungkinkan sebuah chunk server menjalankan sistem operasi lewat jaringan.

Vitur LiveCD yang dimiliki oleh GNU/Linux yang membantu membuat sistem ini terwujud. Jadi sistem operasi hanya akan ada di memory server, tanpa perlu terlebih dahulu diinstall ke server.

Persiapan

Untuk membangun Storage Clustering ini kita membutuhkan bahan-bahan yang cukup murah. Bahan-bahan yang di gunakan tidak mesti baru dan mahal. Saya sarankan anda untuk memanfaatkan sumber daya yang sudah ada karena alasan besarnya dana yang harus dikeluarkan untuk membeli komputer-komputer baru. Bahan-bahan yang di butuhkan untuk membangun storage clustering ini terbagi menjadi dua bagian, kebutuhan perangkat lunak dan kebutuhan perangkat keras. Adapun rincian kebutuhannya adalah sebagai berikut:

Kebutuhan Perangkat Keras (Hardware)

Bahan-bahan perangkat keras (Hardware) yang dibutuhkan untuk membuat sebuah storage clustering antara lain:

- Satu buah komputer yang akan dijadikan master server (Integrator), komputer ini akan menyediakan sistem operasi untuk chunk server, sistem operasi ini dapat di panggil lewat jaringan. Selain menyediakan sistem operasi jaringan, komputer ini nantinya akan di install Mfsmaster sebagai integrator (Penyatu Chunk server). Spesifikasi yang dianjurkan digunakan adalah:
 1. Prosesor : Intel Core2 Duo 2.5 Ghz
 2. Memory : 2 Gigabyte
 3. NIC : 2 buah FastEthernet
 4. Hard disk : 160 Gigabyte

- Switch 8 atau 16 port untuk menghubungkan chunk server dengan master server. Gunakanlah Switch yang menurut anda mampu menangani beban kerja yang berat. Switch ini merupakan bagian terpenting dari Storage Clustering. Menggunakan Switch yang tidak baik kinerjanya akan menurunkan kinerja sistem secara signifikan.

- Beberapa buah komputer yang akan dijadikan chunk server. Chunk server ini berfungsi sebagai hard disk virtual yang diakses lewat jaringan. Chunk server menyediakan ruang penyimpanan, maka spesifikasi yang dibutuhkan lebih mementingkan kapasitas hard disk. Berikut ini adalah spesifikasi komputer yang dianjurkan untuk dijadikan Chunk server:

1. Prosesor : Intel Pentium 4 2.6 Ghz
2. Memory : 512 Megabyte
3. NIC : 1 buah FastEthernet (Support boot from LAN)
4. Hard disk : 2 x 250 Gigabyte (*bisa ditingkatkan sesuai kebutuhan*).

Bahan-bahan yang sudah dijelaskan diatas dapat disesuaikan spesifikasinya dengan sumber daya yang anda milik dan kebutuhan sistem yang akan anda bangun. Semakin besar ukuran storage clustering yang ingin anda buat, harus semakin baik pula spesifikasi server master dan harus semakin baik jaringan yang anda miliki. Jaringan yang menghubungkan server master dengan chunk server harus sangat baik kinerjanya.

Kebutuhan Perangkat Lunak (Software)

Untuk membuat storage clustering ini, ada beberapa perangkat lunak yang harus disiapkan, berikut ini adalah bahan-bahan yang harus di siapkan dan penjelasannya:

1. **Slackware 13.1 GNU/Linux** . Digunakan sebagai sistem operasi pada master server (Integrator). Sebaiknya anda tidak menginstall Sistem operasi lain pada Master server (Singel Boot). Saat installasi sebaiknya pisahkan direktori “/usr” ke dalam partisi terpisah . Direktori “/usr” akan menjadi tempat penyimpanan metadata , maka usahakan mendapat partisi paling besar. Untuk mendapatkan Slackware silahkan kunjungi <http://kambing.ui.ac.id/iso/slackware/slackware-13.1-iso/>
2. **Fuse-2.8.1**. Fuse digunakan untuk membuat layanan filesystem berbasis userspace. Contoh penggunaan fuse adalah httpfs (http filesystem). Fuse ini adalah bagian terpenting dari sistem ini, diatas protokol ini lah MooseFS berjalan. Apa bila anda menginstall Slackware dengan mode “Full Installation” maka anda tidak perlu menginstallnya lagi.
3. **SLAX 6.12**. SLAX adalah variant Slackwere yang mampu berjalan dari mode liveCD. SLAX ini akan menjadi sistem operasi dari Chunk server. Jadi pada Hard disk Chunk server tidak di install kan sistem operasi, melainkan menggunakan sistem liveCD. Untuk mendapat kan SLAX, anda dapat mendownload nya di http://www.slax.org/get_slax.php?download=iso.

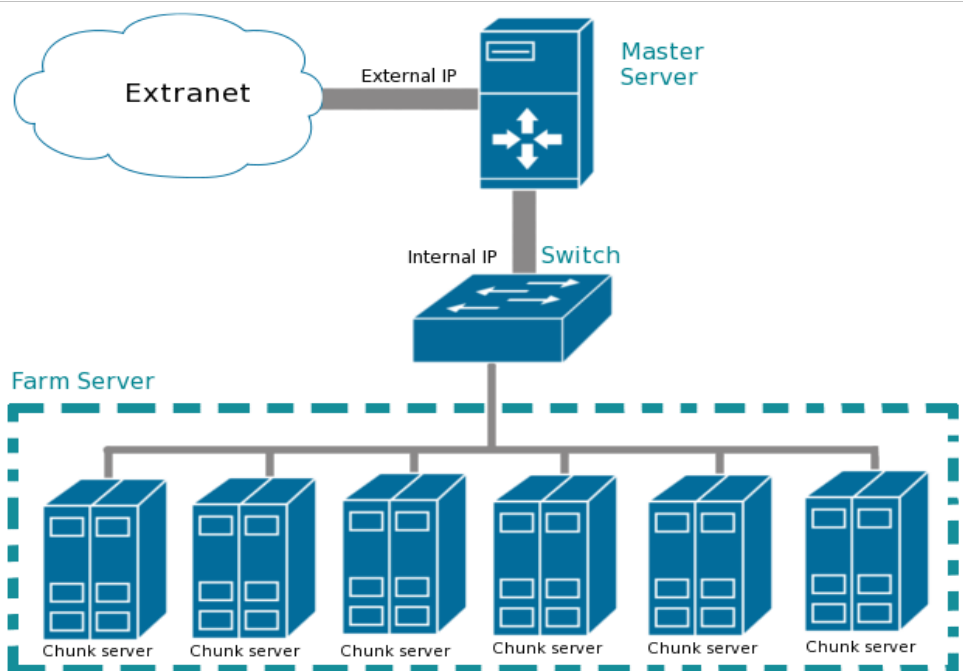
4. **PXE system.** PXE sistem ini berguna untuk menggantikan fungsi CDROM . Jadi nantinya SLAX yang seharusnya di booting lewat CDROM, di booting lewat Jaringan. Dengan metode ini proses boot akan lebih efisien dan tidak memakan banyak CD. Paket aplikasi ini sudah tersedia dalam liveCD SLAX.
5. **MooseFS** . Aplikasi ini menggunakan layanan FUSE untuk membuat file sistem baru diatas jaringan. Anda dapat mendownload nya di http://moosefs.org/tl_files/mfscode/mfs-1.6.17.tar.gz.

Dalam modul ini tidak akan dijelaskan bagaimana cara menginstall Slackware GNU/Linux. Saya berasumsi bahwa anda telah memiliki sebuah komputer server ber-sistem operasi Slackware GNU/Linux terinstall di dalam nya.

Skema Jaringan Komputer

Setelah menyiapkan kebutuhan Software dan Hardware seperti yang sudah dijelaskan sebelumnya, kita akan melangkah ke tahap pembangunan struktur jaringan yang akan digunakan. Dalam membangun Storage Clustering komputer master (integrator) akan menjadi sebuah gateway dari komputer chunk server. Jadi Pengguna Storage clustering tidak akan bisa mengakses komputer chunk server secara langsung. Bahkan seharusnya user tidak tahu bahwa mereka sedang menggunakan media penyimpanan yang terpisah.

Berikut ini adalah gambar skema jaringan komputer yang dianjurkan untuk diterapkan dalam membangun storage clustering:



Implementasi

Pada tahap ini akan dijelaskan bagaimana membangun sistem automatic storage clustering. Saya berasumsi bahwa anda menggunakan skema jaringan menyerupai skema jaringan yang saya sarankan. Dalam tahap implementasi ini anda akan menjalani lima tahapan yaitu:

- Modifikasi SLAX.
- Instalasi dan konfigurasi sistem PXE Boot & Jaringan komputer.
- Instalasi dan konfigurasi mfsmaster pada master server.
- Menjalankan semua service dan menggunakan Storage Clustering.
- Monitor Storage Clustering.

Sebelum melangkah lebih jauh, anda harus paham bahwa setiap perintah yang saya sajikan mungkin sangat berbeda dari apa yang anda hadapi. Contohnya, mungkin saya menggunakan blok IP yang berbeda, hal ini akan menyebabkan konfigurasi PXE boot akan jauh berbeda. Harap sesuaikan konfigurasi sistem dengan kebutuhan anda. Apabila anda masih mengalami masalah, silahkan hubungi saya untuk mendapatkan bantuan.

Modifikasi SLAX

Untuk keperluan modifikasi SLAX ini saya sangat anjurkan anda untuk menggunakan Sistem operasi Slackware GNU/Linux, karena SLAX berasal dari Slackware jadi masalah ketidakcocokan sistem akan sangat kecil kemungkinannya. Sebagai catatan, semua perintah yang diberikan di modul ini harus dijalankan oleh root !. Langkah langkah memodifikasi SLAX untuk bisa digunakan sebagai sistem operasi chunk server adalah sebagai berikut :

1. Pada Sistem operasi Slackware anda buat lah direktori tempat kita akan bekerja.

```
mkdir /home/SLAX
```

2. Download lah iso SLAX dari situs <http://slax.org> dengan perintah

```
wget -c http://slax.org/slax.iso
```

Apabila anda sudah memiliki iso slax maka langkah ini bisa dilewati.

3. Mount iso SLAX ke direktori /mnt/cdrom dengan perintah

```
mount -o loop -t iso9660 <tempat-iso-SLAX> /mnt/cdrom
```

4. Copy semua isi dari direktori /mnt/cdrom ke /home/SLAX dengan perintah

```
cp -arv /mnt/cdrom/* /home/SLAX/
```

5. Masuk kedirektori /home/SLAX dengan perintah

```
cd /home/SLAX
```

6. Copy kan semua peralatan yang akan digunakan untuk memodifikasi SLAX ke “/usr/bin” dengan perintah

```
cp -arv slax/tools/* /usr/bin
```

7. SLAX datang secara default dengan mode grafik , sedangkan kita tidak membutuhkan fitur tersebut. Kita harus membuang fitur yang tidak perlu ini. Ketikkan beberapa baris berikut ini untuk menghapus fitur tersebut.

```
rm slax/base/002*
```

```
rm slax/base/003*
```

```
rm slax/base/004*
```

```
rm slax/base/005*
```

```
rm slax/base/007*
```

Perintah diatas tadi akan mengapus fitur SLAX yang tidak perlu dan menyisakan file 001.core.lzm dan 006.devel.lzm .

8. Sekarang kita akan membuat sebuah paket SLAX yang nanti nya akan berjalan otomatis saat SLAX di muat ke memory chunk server. Paket itu bernama mfschunkserver.lzm. Berikut ini adalah cara membuatnya:

- Buat sebuah direktori di “/opt” sebagai tempat kerja anda, dengan perintah

```
mkdir /opt/clustering
```

- Masuk kedirektori kerja anda dengan perintah

```
cd /opt/clustering
```

- Masukkan source code [mfs-1.6.17.tar.gz](#) ke sana, dengan perintah

```
cp <tempat_mfs-1.6.17.tar.gz> /opt/clustering
```

- Ekstrak source code tersebut dengan perintah

```
tar -zxvf mfs-1.6.17.tar.gz
```


- Masuk kedirektori mfs-1.6.17 dengan perintah

```
cd mfs-1.6.17
```

- Buat direktori /usr/local/mfs, dengan perintah

```
mkdir /usr/local/mfs
```

- Lakukan konfigurasi Source code dengan baris perintah berikut

```
./configure --prefix=/usr/local/mfs \  
--sysconfdir=/usr/local/mfs/etc/ \  
--localstatedir=/usr/local/mfs/var/lib \  
--with-default-user=mfs --with-default-group=mfs \  
--disable-mfsmaster
```

- Buat group mfs dan user mfs dengan dua perintah berikut

```
groupadd mfs  
useradd -g mfs mfs
```

- Compile dan install source code tersebut dengan perintah

```
make && make install
```

- Buat direktori /opt/clustering/mfs/usr/local/ dengan beberapa baris berikut

```
mkdir /opt/clustering/mfs  
mkdir /opt/clustering/mfs/usr  
mkdir /opt/clustering/mfs/usr/local/
```

- Copy kan direktori /usr/local/mfs ke /opt/clustering dengan perintah

```
cp -arv /usr/local/mfs /opt/clustering/mfs/usr/local/
```

- Buat paket mfschunckserver.lzm dengan perintah

```
dir2lzm /opt/clustering/mfs /opt/clustering/mfschunckserver.lzm
```

- Copy kan modul baru tersebut ke /home/SLAX/slax/base dengan perintah

```
cp /opt/clustering/mfschunckserver.lzm /home/SLAX/slax/base
```

9. Sekarang kita akan melakukan sedikit perubahan pada module 001.core.lzm dan 006.devel.lzm. Langkah awal nya adalah buat direktori tempat kita akan mengekstrak isi modul tersebut dengan perintah

```
mkdir /opt/clustering/core
```

10. Setelah itu ekstrak isi modul 001.core.lzm dengan perintah

```
lzm2dir /home/SLAX/slax/base/001.core.lzm core
```

11. Ekstrak juga file 006.devel.lzm kedalam direktori yang sama dengan perintah

```
lzm2dir /home/SLAX/slax/base/006.devel.lzm core
```

12. Download paket ssh dan nano untuk slackware 12.1 dengan perintah

```
wget -c http://kambing.ui.ac.id/slackware/slackware-12.2/slackware/n/openssh-5.1p1-i486-1.tgz
```

```
wget -c http://kambing.ui.ac.id/slackware/slackware-12.2/slackware/ap/nano-2.0.9-i486-1.tgz
```

13. Install kedua paket tersebut kedalam direktori core dengan perintah

```
installpkg -root /opt/clustering/core *.tgz
```

14. Pindah root direktori ke direktori /opt/clustering/core, dengan perintah

```
chroot /opt/clustering/core
```

15. Sekarang kita berada di root directory /opt/clustering/core. Kita akan melakukan beberapa perubahan di dalam root direktori baru ini. Tambahkan user dan group baru dengan perintah

```
group mfs && useradd -g mfs mfs
```

16. Ganti lah password root, root yang kita gunakan sekarang berbeda dengan root yang kita gunakan untuk modifikasi SLAX. Root yang kan kita ganti passwordnya ini adalah root dari livecd Slax. Perintah untuk mengganti password adalah

```
passwd
```

aplikasi ini jenis aplikasi interaktif. Apabila saat anda mengetikan password pertama tidak muncul apa-apa, tidak usah khawatir, karena beginilah cara sistem passwd mengamankan password anda dari pengintip

17. Ubah permisision file /etc/rc.d/rc.ssh menjadi executable dengan perintah

```
chmod +x /etc/rc.d/rc.ssh
```

Dengan membuat file /etc/rc.d/rc.ssh menjadi executable, sistem akan otomatis menjalankan ssh saat chunk server di boot.

18. Edit file /etc/rc.d/rc.local, lalu tambahkan baris perintah dibawah ini

```
/usr/bin/prepare-cluster
```

```
/usr/local/mfs/sbin/mfschunkserver
```

19. Buat file /usr/bin/prepare-cluster dan ubah permission nya menjadi executable dengan perintah

```
touch /usr/bin/prepare-cluster  
chmod +x /usr/bin/prepare-cluster
```

20. Isi file /usr/bin/prepare-cluster dengan beberapa baris dibawah ini

```
#!/bin/bash  
#####  
#This script will help Garuda to prepare a computer become CHUNKSERVER  
#  
MASTER=$(route -n | grep UG | awk '{print $2}')  
  
#Add mfsmaster ipaddress to /etc/hosts  
echo "$MASTER mfsmaster  
127.0.0.1 localhost  
127.0.0.1 slax.org slax  
" > /etc/hosts  
  
#add mount point  
mount | grep /mnt | grep -v /mnt/live | awk '{print $3}' >  
/usr/local/mfs/etc/mfshdd.cfg  
  
#Change owner  
for a in $(cat /usr/local/mfs/etc/mfshdd.cfg); do  
    chown mfs:mfs $a  
done
```

Bari-baris diatas ini adalah bash script yang saya buat untuk menjadikan SLAX yang sudah termodifikasi ini dapat mengenali secara otomatis ada berapa harddisk yang digunakan , dimana di mount nya dan kemudian memasukan settingan itu kedalam file mfshdd.cfg. Saat mfschunkserver dijalankan, server ini akan membaca file mfshdd.cfg untuk menentukan hard disk mana saja yang digunakan sebagai media penyimpanan.

21. Setelah semua perubahan dilakukan, keluarlah dari root livecd slax dengan perintah

```
exit
```

22. Sekarang saat nya kita mem-build ulang paket core dari slax ini dengan perintah

```
dir2lzm /opt/clustering/core /opt/clustering/core.lzm
```

23. Kosongkan direktori /home/SLAX/slax dengan perintah

```
rm /home/SLAX/slax/*
```

24. Pindahkan core.lzm ke direktori /home/SLAX/slax dengan perintah

```
cp /opt/clustering/core.lzm /home/SLAX/slax
```

Sekarang SLAX yang di modifikasi oleh kita sudah siap untuk menjalankan tugas nya. Saat nya masuk ke tahap berikut nya yaitu instalasi dan konfigurasi PXE.

Instalasi dan konfigurasi Sistem PXE Boot

PXE boot adalah bagian yang sangat penting dari Sistem Storage Clustering yang sedang dibangun sekarang. Teknologi ini lah yang bertanggung jawab untuk menggantikan media cd (livecd GNU/Linux menjadi menggunakan media jaringan. Sebenarnya proses booting SLAX yang dijadikan sistem operasi chunk server sama saja ketika menggunakan CD atau pun menggunakan jaringan.

Untuk dapat menggunakan fitur PXE Boot ini, kita harus menggunakan komputer yang bios nya mendukung boot from lan. Kabar baik nya adalah motherboard yang ada sekarang sudah mendukung boot from LAN.

Proses boot menggunakan PXE ini sangat simpel, pertama saat kita memboot komputer kita yang sudah mendukung boot from Lan itu dia (komputer) akan menjalankan sesi PXE bootstrap yang diawali dengan mencari DHCP server. Setelah mendapatkan IP dari DHCP server, bootstrap berlanjut meminta koneksi ke server tftp (trivial File Transfer Protocol). Setelah mendapatkan respon dari tftp, bootstrap akan meminta image kernel ,initrd dan informasi dari mana root direktori akan dimount. Setelah menerima image kernel dan initrd maka proses booting pun berjalan seperti ketika kita menggunakan CD. PXE server ini akan di jalankan di Master Server

Sebelum kita mensetting PXE server sebaiknya kita mensetting Ip para interface yang akan digunakan oleh Master Server. Berikut adalah langkah-langkah mensetting ip nya.

- Tentukan interface mana yang akan digunakan sebagai external Interface dan mana yang akan dijadikan internal interface. Misalnya anda memiliki dua interface, eth0 dan eth1. Pilih eth0 sebagai external interface dan eth1 sebagai internal interface.

- Edit file `/etc/rc.d/rc.inet1.conf` , cari baris “# Config information for eth0:” lalu rubah isi nya menjadi seperti berikut

```
# Config information for eth0:
IPADDR[0]="192.168.40.1"
NETMASK[0]="255.255.255.0"
USE_DHCP[0]=" "
DHCP_HOSTNAME[0]=" "

# Config information for eth1:
IPADDR[1]="192.168.1.1"
NETMASK[1]=" "
USE_DHCP[1]=" "
DHCP_HOSTNAME[1]=" "
```

Nilai external ip dan internal IP bisa sangat berbeda dari apa yang dicontohkan Kita tidak usah lagi menginstall atau mendownload aplikasi tambahan untuk membangun pxe server karena didalam iso SLAX sudah terdapat semua fitur yang dibutuhkan. Namun karena sistem SLAX didesign dengan mode livecd maka kita harus menyesuaikan beberapa bagian agar dapat berjalan. Berikut ini adalah beberapa langkah yang dibutuhkan agar PXE server yang ada di dalam ISO SLAX dapat berjalan di Master server.

1. Masuk ke direktori `/home/SLAX/boot/pxelinux.cfg` dengan perintah

```
cd /home/SLAX/boot/pxelinux.cfg
```

2. Edit file “start” lalu ubah isinya menjadi seperti ini

```
#!/bin/bash
# This script will start a DHCP server and TFTP server in order to provide
# fully functional environment for PXE booting.
#
cd $(dirname $(readlink -f $0))
IP="192.168.1.1"
# if no IP is assigned to this computer, setup private address randomly
if [ "$IP" = " " ]; then
    killall dhcpd 2>/dev/null
    IP="10.$(($RANDOM/130+1)).${(($RANDOM/130+1)).1}"
    ifconfig $(ls -l /sys/class/net | head -n 1) $IP netmask 255.0.0.0
fi
# calculate C class range
RANGE=$(echo $IP | cut -d "." -f 1-3)

# make sure dnsmasq can be started
killall dnsmasq 2>/dev/null
mkdir -p /var/state/dnsmasq

# start the DHCP server and TFTP server
./dnsmasq -i eth1 --enable-tftp --tftp-root=/home/SLAX/boot \
--dhcp-boot=/pxelinux.cfg/pxelinux.0,"$IP", $IP \
--dhcp-range=$RANGE.2,$RANGE.250,infinite --log-dhcp
```

3. Pada sistem livecd, SLAX memuatkan modul sistem seperti 001.core.lzm dan sebagainya lewat CD-rom. Namun pada sistem PXE, modul dimuat lewat httpfs. Untuk membuat httpfs berjalan baik edit file “/home/SLAX/boot/pxelinux.cfg/web/start” . Cari baris

```
ROOT=$(verify_datadir $1 /mnt/live/mnt/* /mnt/*)
```

ganti menjadi

```
ROOT=$(verify_datadir $1 /home/SLAX/* /home/*)
```

Instalasi dan konfigurasi mfsmaster pada master server

Proses instalasi mfsmaster sangat mudah , berikut adalah langkah-langkah nya :

1. Copykan source code mfs-1.6.17.tar.gz ke direktori “/tmp” dengan perintah

```
cp <tempat-source-code> /tmp
```

2. Masuk ke direktori /tmp dan ekstrak source code tersebut dengan perintah

```
cd /tmp/ && tar -zxvf mfs-1.6.17.tar.gz
```

3. Masuk ke direktori mfs-1.6.17 dengan perintah

```
cd mfs-1.6.17
```

4. Konfigurasi source code untuk menjadi mfsmaster dengan perintah

```
./configure --prefix=/usr --sysconfdir=/etc \
--localstatedir=/var/lib --with-default-user=mfs \
--with-default-group=mfs --disable-mfschunkserver
```

5. Kompilasi dan install source code tersebut dengan perintah

```
make && make install
```

6. Setelah source code berhasil di install, proses instalasi tadi membuat template file konfigurasi pada direktori /etc. Konfigurasi yang terdapat pada file-file ini adalah konfigurasi standard yang dibutuhkan oleh mfsmaster. Karena masih template maka file-file ini masih ber-ekstensi *.dist, hilangkan ekstensi *.dist nya dengan perintah berikut

```
cd /etc
cp mfsmaster.cfg.dist mfsmaster.cfg
cp mfsmetallogger.cfg.dist mfsmetallogger.cfg
cp mfsexports.cfg.dist mfsexports.cfg
```

7. Masuk ke direktori `/var/lib/mfs`, disini lah metadata, dan changelog file disimpan. Pada instalasi standard, installer membuat file `/var/lib/mfs/metadata.mfs.empty`, kita harus me-rename nya menjadi `/var/lib/mfs/metadata.mfs` dengan perintah

```
cp /var/lib/mfs/metadata.mfs.empty /var/lib/mfs/metadata.mfs
```

8. Edit file `/etc/hosts` lalu tambahkan baris

```
192.168.1.1          mfsmaster
```

Menjalankan semua service dan menggunakan Storage Clustering

Setelah proses intallasi mfsmaster dan modifikasi SLAX selesai saat nya kita untuk menjalankan semua service dan memulai menyalakan automatic storage clustering. Berikut ini adalah langkah-langkah untuk menyalakan semua service nya:

- Jalankan master server dengan perintah

```
/usr/sbin/mfsmaster start
```
- Jalankan monkey httpfs server dengan perintah

```
/home/SLAX/boot/pxelinux.cfg/web/start
```
- Jalankan PXE server dengan perintah

```
/home/SLAX/boot/pxelinux.cfg/start
```
- Jalankan mfscgiserver untuk memonitor status storage clustering, dengan perintah

```
/usr/sbin/mfscgiserv
```
- Masukkan semua baris tersebut ke dalam file `/etc/rc.d/rc.local`

Sekarang coba anda nyalakan mesin-mesin yang menjadi anggota farm server dan tunggu beberapa saat. Setelah semua menyala mount lah storage clustering ini dengan perintah

```
mfsmount -H 192.168.1.1 /mnt/hd
```

Lihat lah drive yang baru di mount dengan perintah “df -h”, output nya seperti ini

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/root	18G	17G	393M	98%	/
tmpfs	944M	0	944M	0%	/dev/shm
192.168.1.1:9421	626G	272G	354G	44%	/mnt/hd

Monitor Storage Clustering

Setelah semua service berjalan dan anda sudah berhasil me-mount storage clustering, anda dapat memantau aktifitas Storage clustering yang baru saja di bangun. MooseFS sudah menyediakan perkakas yang digunakan untuk melakukan monitoring, jadi kita tidak perlu lagi menginstall alat monitoring tambahan.

Perintah `/usr/sbin/mfscgiserv` akan menjalankan perkakas monitoring dan mengeluarkan output nya lewat browser. Anda dapat mengakses nya dengan mengetikan `http://<ip_master_server>:9425` pada address bar browser anda.

Ini adalah contoh tampilan dari home page storage clustering monitor. Disini anda bisa melihat ukuran total hard disk , ukuran yang tersisa dan lain sebagainya.

Info												
version	total space	avail space	trash space	trash files	reserved space	reserved files	all fs objects	directories	files	chunks	all chunk copie	
1.6.17	660 GiB	354 GiB	0 B	0	0 B	0	39548	5916	33628	36941	3693	
All chunks state matrix (counts 'regular' hdd space and 'marked for removal' hdd space : switch to 'regular')												
goal	valid copies											
	0	1	2	3	4	5	6	7	8	9	10+	all
0	-	-	-	-	-	-	-	-	-	-	-	-
1	3	36938	-	-	-	-	-	-	-	-	-	3694
2	-	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-	-	-	-
6	-	-	-	-	-	-	-	-	-	-	-	-
7	-	-	-	-	-	-	-	-	-	-	-	-
8	-	-	-	-	-	-	-	-	-	-	-	-
9	-	-	-	-	-	-	-	-	-	-	-	-
10+	-	-	-	-	-	-	-	-	-	-	-	-
all 1+	3	36938	0	0	0	0	0	0	0	0	0	3694

Ini adalah sub menu server , pada tampilan ini kita bisa melihat ada berapa server yang terkoneksi membentuk sebuah storage clustering.

Chunk Servers									
#	host	ip	port	version	'regular' hdd space				
					chunks	used	total	% used	
1	(unresolved)	192.168.1.146	9422	1.6.17	15338	136 GiB	293 GiB	46.19	
2	(unresolved)	192.168.1.182	9422	1.6.17	18511	137 GiB	293 GiB	46.57	
3	(unresolved)	192.168.1.185	9422	1.6.17	1591	17 GiB	37 GiB	46.25	
4	(unresolved)	192.168.1.194	9422	1.6.17	1498	17 GiB	37 GiB	46.69	

Sub tab Export ini memuat tentang siapa saja yang melakukan mount terhadap storage clustering.

Exports													
#	ip range	path	minversion	alldirs	password	ro/rw	restricted ip	ignore gid	root uid:gid	all users uid:gid			
1	0.0.0.0 255.255.255.255	.(META)	0.0.0	-	no	rw	yes	-	-	-	-	-	-
2	0.0.0.0 255.255.255.255	/	0.0.0	yes	no	rw	yes	no	0	0	-	-	-

Sub tab ini memuat informasi tentang hard disk yang disumbangkan oleh setiap chunk server

Disks																
#	info				I/O stats last min (switch to hour,day)									space		
					transfer		max time (switch to avg)			# of ops						
	path	chunks	last error	status	read	write	read	write	fsync	read	write	fsync	used	total	used (%)	
1	192.168.1.146:9422:/mnt/sda1/	5635	no errors	ok	-	-	-	-	-	-	-	-	68 GiB	147 GiB	46.09	
2	192.168.1.146:9422:/mnt/sdb1/	9703	no errors	ok	-	-	-	-	-	-	-	-	68 GiB	147 GiB	46.30	
3	192.168.1.182:9422:/mnt/sda1/	9249	no errors	ok	-	-	-	-	-	-	-	-	68 GiB	147 GiB	46.57	
4	192.168.1.182:9422:/mnt/sdb1/	9262	no errors	ok	-	-	-	-	-	-	-	-	68 GiB	147 GiB	46.58	
5	192.168.1.185:9422:/mnt/hda1/	1591	no errors	ok	-	-	-	-	-	-	-	-	17 GiB	37 GiB	46.25	
6	192.168.1.194:9422:/mnt/hda1/	1498	no errors	ok	-	-	-	-	-	-	-	-	17 GiB	37 GiB	46.69	

Selamat mencoba

Daftar Pustaka

Michał Borychowski, Installing MooseFS Step by Step Tutorial, Gemius SA, 2010.

John Spiers, Storage Clustering, [online] Tersedia: <http://www.wwpi.com/>.

MooseFS, [online] Tersedia : <http://moosefs.org/reference-guide.html>.

PXE Boot, [online] Tersedia : <http://www.devshed.com/>

Tentang Penulis

Bramandityo Prabowo



Lahir di Lampung, 21 juli 1989. Saat ini tengah menyelesaikan program S1 di Universitas Pendidikan Indonesia Jurusan Ilmu Komputer.

Aktifitas saat ini adalah menjadi Asisten Laboratorium Jaringan Ilmu komputer UPI , Asisten dosen Sistem Operasi, Anggota lembaga Pendayagunaan Open Source Software Universitas Pendidikan Indonesia dan Staf pengajar pada Padepokan Open Source Universitas Pendidikan Indonesia.

Berpengalaman sebagai System administrator, Web developer dan Network engineer. Menggunakan produk Open Source sejak tahun 2005. Mampu mengoperasikan server maupun workstation bersistem operasi GNU/Linux, FreeBSD , NetBSD , OpenBSD dan OpenSolaris.

Informasi lebih lanjut mengenai penulis :

Email : admin@bramandityo.com

Blog : <http://bramandityo.com>

YM : freez_meinster