

Linux 下 MPICH2 集群系统安装手册

本手册介绍安装 linux 系统后，如何进行并行环境的配置、mpich2 的安装和文件共享的设置。

系统环境: rhel-server-5.3-x86_64。

软件环境: 系统自带的软件完全安装、mpich2-1.0.8.tar.gz (第三方软件)。

本文档约定:

使用机器上的第一个千兆网卡口 eth0,

后面的内容为注释, 蓝色字体,

本文档已三个节点为例,

机器名依次为 c1、c2、c3,

机器分配的 IP 地址依次为:

192.168.0.11

192.168.0.12

192.168.0.13

1、准备工作:

1.1、设置 IP (以 c1 为例):

```
#vi /etc/sysconfig/network-scripts/ifconfig-eth0
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.0.11
NETMASK=255.255.255.0
GATEWAY=192.168.0.1
```

1.2、更改主机名 (以 c1 为例):

```
#vi /etc/sysconfig/network
NETWORKING=yes
HOSTNAME=c1
```

1.3、重启 network 服务:

```
#service network restart
```

其余节点 (c2、c3) 操作步骤同上

2、创建 SSH 信任连接 (c1 的 root 目录下):

2.1、更改/etc/hosts 文件

```
#vi /etc/hosts          ##编辑 hosts 文件, 更改如下:
127.0.0.1      localhost.localdomain localhost
192.168.0.11  c1
192.168.0.12  c2
192.168.0.13  c3
```

2.2、生成 SSH 秘钥对.

```
#ssh-keygen -t rsa      ## 一路回车即可(在此口令为空) 产生.ssh  
文件, 默认路径是/root/.ssh
```

```
#ls -a                  ## 查看是否有.ssh文件夹
```

2.3、进入.ssh目录

```
#cd .ssh
```

2.4、生成authorized_keys文件•

```
#cp id_rsa.pub authorized_keys
```

2.5、退出到root目录

```
#cd ..
```

2.6、建立本身的信任连接

```
#ssh c1                  ## 按提示输入 yes (三个字母要打全)
```

2.7、设置c2 (c2的root目录下)

```
#scp 192.168.0.11:/root/.ssh/* /root/.ssh/
```

```
#scp 192.168.0.11:/etc/hosts /etc/hosts
```

```
## 此步会提示是否信任对方密钥 (c1的密钥)
```

```
## 并要求输入对方密码 (c1的密码)
```

2.8、建立c2本身的信任连接

```
#ssh c2                  ## 按提示输入 yes (三个字母要打全)
```

2.9、确认机器的信任连接已建立

对每个节点执行:

```
#ssh c1
```

```
#ssh c2
```

```
#ssh c3
```

```
## 在提示处输入 yes 回车, 最后确定无需输入密码并且没有任何提示信息即可登陆
```

```
## ("Last login: 时间日期"提示信息除外)
```

3、安装MPICH2(在节点root目录下), 并配置并行环境:

3.1、安装MPICH2

```
#tar -zxvf mpich2-1.0.8.tar.gz
```

3.1.1、创建安装目录

```
#mkdir /usr/MPICH-insts11
```

3.1.2、进入mpich2解压目录

```
#cd mpich2-1.0.8
```

3.1.3、设置安装目录

```
#!/configure --prefix=/usr/mpich-install -rsh=ssh
```

```
##每个节点所设置的安装目录必须相同
```

3.1.4、编译

```
#make
```

3.1.5、安装

```
#make install
```

3.1.6、退出到root目录

```
#cd ..
```

3.2、通过编辑.bashrc文件修改环境变量

```
#vi .bashrc
```

修改后的.bashrc文件如下:

```
# .bashrc
# User specific aliases and functions
alias rm='rm -i'
alias cp='cp -i'
alias mv='mv -i'
PATH="$PATH:/usr/MPICH-install/bin"          ## 新增加的
#Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi
```

保存后执行:

```
# source .bashrc
```

3.3、测试环境变量设置

```
#which mpd
#which mpicc
#which mpiexec
#which mpirun
```

3.4、修改/etc/mpd.conf文件，内容为secretword=myword

```
#vi /etc/mpd.conf
设置文件读取权限和修改时间
#touch /etc/mpd.conf
#chmod 600 /etc/mpd.conf
```

3.5、创建主机名称集合文件/root/mpd.hosts

```
#vi mpd.hosts
## 文件内容如下:
c1
c2
c3
```

安装时两台机器上的安装目录要相同，不太懂防火墙的最好把防火墙关掉，命令为:

```
# service iptables stop
```

4、测试并行环境能否正常运行:

4.1、本地测试

```
#mpd &          ## 启动
#mpdtrace       ## 观看启动机器
#mpdallexit     ## 退出
```

4.2、通过mpd.hosts运行集群系统

```
#mpdboot -n number -f mpd.hosts    ## number 为要起动的机器个数
#mpdtrace
```

```
#mpdallexit
```

5、问题解决（上一步有问题则进行此步）

5.1、通过 mpdcheck 获得一写帮助信息

```
#mpdcheck -pc
```

5.2、查错

```
#mpdcheck -l
```

5.3、通过 mpd.hosts 文件查错

```
#mpdcheck -f mpd.hosts
```

```
#mpdcheck -f mpd.hosts -ssh
```

5.4、如果上述无错误，可略过此步

```
## 对任意两台机器进行查错
```

```
c1: #mpdcheck -s          ## 输出主机名 host 和端口 port
```

```
c2: #mpdcheck -c host port
```

```
## 注意：以上四步都是在没有运行 mpd 的情况下进行的
```

5.5、mpd 查错

```
c1: #mpd -e &           ## 返回使用的端口
```

```
c2: #mpd -h m1 -p echoed_port_m1 &
```

6、文件共享的设置:

此步是使 c2、c3 能挂载 c1 的目录/home，能让 c1、c2、c3 进行并行计算时数据都写在 c1 的/home 目录中，c2、c3 自己原来的目录/home 更名为/home1，即 c2、c3 上的/home1 用来存储自己的数据，算题时数据都往 c1:/home 上写。

6.1、编辑 NFS 服务器配置文件 /etc/exports，以允许其它机器通过 NFS 访问目录/home（步骤 6.1、6.2、6.3 在 c1 上操作）

```
[root@c1 ~]# vi /etc/exports
```

修改后的/etc/exports 文件内容:

```
/home *(rw,no-root-squash, sync)    ##把 c1 的目录/home 共享出来
```

其中:

* 表示任何 IP 地址都可以访问该目录。为了安全的考虑，也可以指定具体的 IP 地址。

no-root-squash 表示登入到 NFS 主机的用户如果是 ROOT 用户，他就拥有 ROOT 的权限，

rw 表示远程机器对该目录只有可读写的权限

其它选项请参考 NFS 手册。

6.2、重新启动 NFS 服务以使新配置生效

```
# service portmap restart
```

```
# service nfs restart
```

以下是测试片段:

```
[root@c1 ~]# service portmap restart
```

```
Stopping portmap: [ OK ]
```

```
Starting portmap: [ OK ]
```

```
[root@c1 ~]# service nfs restart
Shutting down NFS mountd: [ OK ]
Shutting down NFS daemon: [ OK ]
Shutting down NFS quotas: [ OK ]
Shutting down NFS services: [ OK ]
Starting NFS services: [ OK ]
Starting NFS quotas: [ OK ]
Starting NFS daemon: [ OK ]
Starting NFS mountd: [ OK ]
```

6.3、检测目录/home 是否已被正确共享

```
#showmount -e
```

以下是测试片段:

```
[root@c1 ~]# showmount -e
Export list for c1:
/home *
```

6.4、编辑/etc/fstab 文件（步骤 6.4、6.5 在 c2、c3 上操作）

以 c2 为例，把 c2 的/home 目录更名为/home1，然后挂载 c1 的/home

```
[root@c2 ~]# cp /etc/fstab /etc/fstab.bak ##备份原来的分区表
[root@c2 ~]# vi /etc/fstab
```

以下是编辑后的内容:

```
# This file is edited by fstab-sync - see 'man fstab-sync' for details
LABEL=/1 / ext3 defaults 1 1
None /dev/pts devpts gid=5,mode=620 0 0
None /dev/shm tmpfs defaults 0 0
LABEL=/home /home1 ext3 defaults 1 2 ##本
行是改了
None /proc proc defaults 0 0
None /sys sysfs defaults 0 0
LABEL=SWAP-sda9 swap swap defaults 0 0
c1:/home /home nfs defaults 1 2 ##本
```

行是新加的

6.5、挂载共享目录

```
[root@c2 ~]# mkdir /home1
[root@c2 ~]# umount /home
[root@c2 ~]# mount /home1
[root@c2 ~]# mount /home
[root@c2 ~]# reboot ##重新启动，使配置生效
```

6.6、更改用户的默认目录

修改/etc/default/useradd 文件，把 HOME=/home 改为你要的路径

特别提示：开机时先开 c1，再开 c2、c3，以便 c2、c3 能够顺利挂载 c1:/home。