利用 CentOS 6.3 和 DRBL 架設 PC-clustr

吳泰賢 Bruce Wu

2013/3/19

1 介紹

兩年前，剛開始摸Linux系統和pc-cluster相關，在實驗室架起簡單的pc-cluster並且寫了一篇簡易文章「 PC-Cluster架設圖解教學 作業系統 -linux-openSuSE 11.2」變回到研究上，但是系統這種東西對小弟我有深深的魔力，前陣子偶然間發現其實可以用無碟系統(node上並無硬碟)來架設pc-cluster，不但節省經費又節省時間(想像萬一很多台node，雖然我畢業的實驗室的node數目很少==)，忍不住就利用動手玩了起來，因此便花了些時間紀錄這陣子的心得，也希望有錯誤的地方請各位高手不吝指教。

剛開始接觸到無碟系統是在廠商的介紹會上，原本以為是一項大工程，回家google才發現中華民國的國網中心真的是非常的強大，早就已經發展出DRBL，利用DRBL，我們可以輕鬆得完成無疊系統的cluster架設，在此對國網中心的各位前輩們深深一鞠躬。

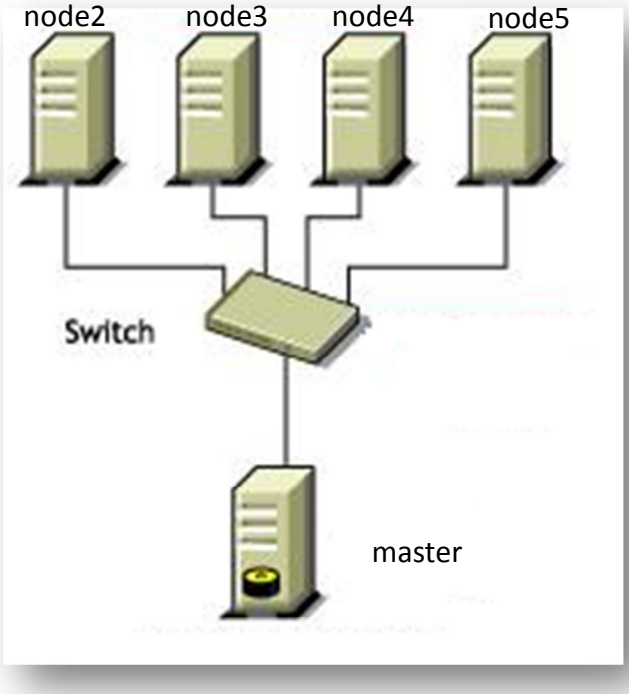
兩年前，我使用的系統地openSUSE，原因是因為其安裝套件的GUI介面非常好用(當時的我對指令仍然沒這麼熟悉，Linux又常常東缺一個套件，西區一個套件)，但是最近openSUSE的GNOME介面更新了，說實在話我用的沒有很習慣(也開始體會到指令操作的好處，不用因為介面升級而有很大的影響)，再加上我的Linux知識幾乎全部來自鳥哥(再次深深一鞠躬)，在鳥哥的教學文中，也是使用CentOS來示範，因此就投入了CentOS的懷抱，進一步發現DRBL和CentOS也配合得很好，因此架設上並沒有碰到太多的麻煩。

講了這麼多，開始介紹一下我們需要的硬體設備(最陽春)

(1)兩台電腦(或更多)，一台為master(需要兩張網卡)，其他台為nodes(各需一張網卡但不需要硬碟)。

(2)switch。

左圖為示意圖，一台master和4台nodes，透過一台switch連起來。現在個人電腦雖然不貴，但是也不是每個人都有辦法隨隨便便就弄到兩台以上來練習架設啦!在此提供大家一個方法，就是利用虛擬機器來練習啦!虛擬機器現在可以說是當紅炸子雞，小弟我也利用過免費的VMware player測試，是絕對沒有問題的。



2 master 設定

先說個好消息,這章結束大概已經完成 90%的架設了吧!有沒有士氣大增呢?其實相較於兩年前我架設的手法,利用 DRBL 架設 cluster 可以說是更簡單,首先我先簡述一下 cluster 的原理。

所謂的 cluster,其實只是數台電腦,利用網路線串起來,然後設定好彼此之間溝通的通訊協定,使得 master 可以將一些工作分配到 node 上,所以到底需要在 master 上架設那些 server 呢?其功能又是甚麼呢?簡述如下

(1)DHCP server

發給每個 nodeIP,node 數量少亦可省略。

(2)NFS server

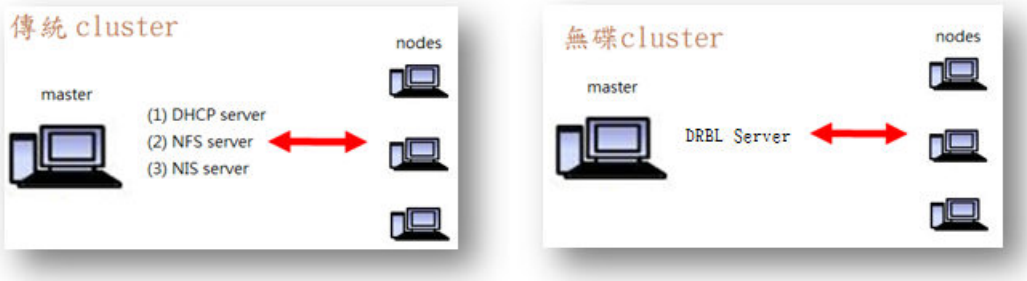
將硬碟空間分享給 node。

(3)NIS server

將 master 上的帳號分享給 node。

其實嚴格說起來,ssh 應該也是 master 上的一個 server,但是現在 ssh 幾乎已經成為主流,每種版本的 Linux 也幾乎預設就會安裝(openSUSE 好像卻沒有,匪夷所思?!),因此這邊我就不再多加強調。

如果有看過鳥哥網站的人,其實會發現雖然才 3 個 server,但是其實細節也是非常多,不過 DRBL 強大的地方,就是只要他把這三個 server 架設都包了,因此只要裝好 DRBL,基本上 master 就完成設定了,不過還是建議回頭看看鳥哥的教學,增加一些觀念,有好沒壞呀!



上圖我們可以看到,DRBL server 其實就涵蓋了原本傳統 cluster 的功能,其實 DRBL 還包含更多的 server(如 tftp 等…),不過有些太過細節的部份我們就不強調。

2.1CentOS 6.3 安裝

基本上安裝都是 GUI 介面，我就不贅述，我是選擇 software development workstation，內建有 GNOME。下面我將列出一些預設安裝後需要改變的設定

(1)分割硬碟

除了基本的配置外，請額外分割一個硬碟空間統為/tftpboot，之後 DRBL 會將 node 開機所需的系統組態檔放在這邊。若你忘記分割，系統也會自動產生，建議額外分割的道理就和我們盡量將/boot、/home 等和根目錄/放於不同的分割空間一樣，萬一某個分割空間掛了，其他檔案系統不會受到影響。

(2)設定網路

由於 master 有兩張網路卡，沒意外應該是預設是 eth0 和 eth1，此處我的系統是 eth0 對外(設成固定 IP)，eth1 對內(設為虛擬固定 IP：192.168.1.1)，此處我主要針對 eth1 設定(因為 eth0 對外的設定每個人都不一樣，有問題可以查看鳥哥的教學文)，以 root 身分利用 vi 去編輯設定檔

vi etc/sysconfig/network-scripts/ifcfg-eth1

```
DEVICE="eth1"
IPADDR="192.168.1.1"
NETMASK="255.255.255.0"
HWADDR="00:0C:29:15:7C:A3"
NM_CONTROLLED="no"
ONBOOT="yes"
TYPE="Ethernet"
```

稍微注意一下，此處 192.168.1.1 是我指定 master 對內網域的 IP，ONBOOT=”yes”表示開機自動啟動，HWADDR 為網卡位置，不要照抄喔!會出錯的。

設定完以後，記得要重新啟動 network 服務，可輸入

/etc/init.d/network restart

```
[root@master ~]# /etc/init.d/network restart
Shutting down interface eth0:                              [  OK  ]
Shutting down interface eth1:                              [  OK  ]
Shutting down loopback interface:                          [  OK  ]
Bringing up loopback interface:                            [  OK  ]
Bringing up interface eth0:
Determining IP information for eth0... done.
                                                           [  OK  ]
Bringing up interface eth1:                                [  OK  ]
[root@master ~]#
```

2.2 DRBL 安裝

DRBL 的安裝，其實在 DRBL 官方網站(http://drbl.nchc.org.tw/one4all/desktop/)已經介紹的非常詳細，以下我只是以我的經驗，更簡單迅速的介紹。

(1)把 selinux 關閉。

利用 vi 開啟/etc/sysconfig/selinux，將 SELINUX 設成 disabled，然後重開機。

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of these two values:
#     targeted - Targeted processes are protected,
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

(2)安裝 DRBL 金鑰，執行

rm -f GPG-KEY-DRBL; wget http://drbl.nchc.org.tw/GPG-KEY-DRBL; rpm --import GPG-KEY-DRBL

```
[root@master Downloads]# rm -f GPG-KEY-DRBL; wget http://drbl.nchc.org.tw/GPG-KE
Y-DRBL; rpm --import GPG-KEY-DRBL
--2013-03-20 01:27:10--  http://drbl.nchc.org.tw/GPG-KEY-DRBL
Resolving drbl.nchc.org.tw... 211.73.64.9, 2001:e10:3c00:8::e274
Connecting to drbl.nchc.org.tw|211.73.64.9|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1361 (1.3K) [text/plain]
Saving to: “GPG-KEY-DRBL”

100%[======================================>] 1,361       --.-K/s   in 0s

2013-03-20 01:27:11 (58.8 MB/s) - “GPG-KEY-DRBL” saved [1361/1361]

[root@master Downloads]#
```

(3)到官方網站下載 DRBL 的安裝包(我下載的檔案為 drbl-2.3.12-drbl1.noarch.rpm)，以 root 身分執行 rpm -Uvh drbl-2.3.12-drbl1.noarch.rpm，會看到如下圖的錯誤這

```
[root@master Downloads]# rpm -Uvh drbl-2.3.12-drbl1.noarch.rpm
warning: drbl-2.3.12-drbl1.noarch.rpm: Header V4 DSA/SHA1 Signature, key ID d7e8
df3a: NOKEY
error: Failed dependencies:
        perl(Digest::SHA1) is needed by drbl-2.3.12-drbl1.noarch
[root@master Downloads]#
```

這只是沒有安裝 perl 而已，執行
yum install -y perl-Digest-SHA1
完成後再次執行 rpm -Uvh drbl-2.3.12-drbl1.noarch.rpm 即可正常安裝 DRBL 的 rpm 檔。

```
[root@master Downloads]# rpm -Uvh drbl-2.3.12-drbl1.noarch.rpm
Preparing...                ########################################### [100%]
   1:drbl                   ########################################### [100%]
[root@master Downloads]#
```

(4)輸入 drblsrv -i 自動完成其他套件下載。會碰到一些問題，基本上沒有特殊需求選擇預設選項即可。

(5)在/etc/drbl/下，新增一個檔案叫 client-ip-hostname，編輯 IP 和電腦名稱的對應關係，下圖為範例

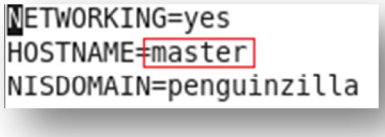
```
# DRBL clients IP-Address HOSTNAME Mapping
# The format is:
# --------------------
# IP-Address Hostname
# --------------------
# List them line by line, edit this before you run "drblpush -i".
# *******NOTE********
# 1. The hostname must be unique! Do NOT duplicate them.
# 2. For IP address format, do NOT use something like 192.168.001.010,
#    use 192.168.1.10, i.e. do NOT put extra "0" for IP address digits.
# 3. If some client you do not assign here, drblpush will automatically
#    create one for you. It is based on the prefix you assing when
#    running "drblpush -i"
# 4. Host names may contain only alphanumeric characters, minus signs ("-"),
#    and periods (".") They must begin with an alphabetic character and end
#    with an alphanumeric character. "man hosts" for more details.
# 5. This hostname is not FQDN (Fully Qualified Domain Name), it's just Unix hos
tname
#

192.168.1.1 master
192.168.1.2 node2
```

我希望拿到 192.168.1.2 的 node，電腦名稱為 node2，以此類推。此檔案在下一步 DRBL 中的設定會用到。

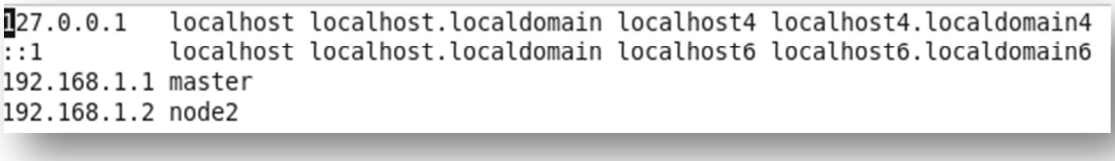
(6)輸入 drblpush -i 進行設定(建議先至官方網站 http://drbl.nchc.org.tw/one4all/desktop/看更詳細的教學)。
基本上到這邊，已經完成 DRBL 的基本設定，建議你將 master 重開機，重開機後，若你將 node 接上，並將 node 的 BIOS 設定網路開機(PXE)，node 即可正常啟動，意味著已經完成了 cluster 的架設啦。但是 DRBL 會自動編輯相當多的系統設定，有些和我習慣不盡相同，因此接下來的步驟，只是我試著將一些系統設定檔改成我原本的習慣。

(7)電腦名稱 hostname
DRBL 會修改我的 hostname，這是我所不願意見到的，先以 root 執行
hostname master
這邊 master 為我希望的 hostname，你可以自己修改你想要的，另外再用 vi 編輯/etc/sysconfig/network 裡面的 HOSTNAME



這兩個步驟可更改你電腦的 hostname。

(8)IP 對應的代號
以 root 執行
vi /etc/hosts
可以編輯 IP 位置對應的代號



DRBL 會自動編輯這個檔案，把每台電腦的 hostname 當作代號編輯進去，但是就如同前面，DRBL 會修改我 master 的 hostname，所以我也是習慣再自己回來編輯這個檔案，確保 IP 和電腦代號的對應關係。

3 node 設定

基本上，沒有特殊需求，node 只要開機在 BIOS 中使用網路開機(PXE)即可。但是若有特殊需求呢？以我的例子來說，我曾經想在 node 上安裝軟體到/opt，但是 node 預設是沒有修改 opt 的權限。這章就是對這些比較奇怪的需求特別說明一下，在此也感謝 DRBL 作者的 Ceasar Sun，在 DRBL 的留言板回覆我這些奇怪的需求，也藉此讓我更了解 DRBL 的運作方式。

首先 node 的系統組態檔基本上都是放在/tftpboot 下，所以儘管我們想要修改的是 node 的設定，也是在 master 下完成。下面我將列表說明幾個 node 下較重要的目錄對應 master 目錄的位置。

master	node
/tftpboot/node_root/	/
/tftpboot/nodes/[node 的 ip]/etc	/etc
/tftpboot/nodes/[node 的 ip]/var	/var
/usr	/usr
/opt	/opt
/home	/home

從上表也可以發現，為何預設 node 上的/usr 和/opt 是沒有寫入權限的，因為它們是直接對應到 master 的/usr 和/opt，若是在 node 上可以任意修改的話，很容易危害到 master 導致整組 cluster 系統毀滅。然而其他的像是/etc 等目錄，則是對應到/tftpboot 下(和 master 本身的檔案是分開的)，儘管胡搞瞎搞也不會危害到 master，因此允許 node 自己修改這些檔案。也因為/tftpboot 下的東西是 DRBL 自動產生的，因此當我們 master 有修改系統檔案在/etc、/var 下等等，而且希望 node 也跟著修改，就須重新執行 DRBL(因為要重新將 master 的檔案複製到/tftpboot)。也因為 DRBL 這樣子的設計，所以盡量將軟體安裝在/opt 或/usr(大部分軟體預設也確實是如此)下，這樣子 node 只需重新開機就可以使用新的軟體，而不用重新設定 DRBL。

不過假使真的要動用到/opt 和/usr 權限呢 ?下面我將分享國網中心 Ceasar Sun 給我的回覆。

(1)讓 node 有/opt 的權限

A. 使用 root 身分修改/etc/exports，將/opt 的設定從 ro 改成 rw

```
# Generated by DRBL at 16:46:58 2013/03/20
/tftpboot/node_root 192.168.1.2(ro,async,no_root_squash,no_subtree_check)
/usr 192.168.1.2(ro,async,no_root_squash,no_subtree_check)
/home 192.168.1.2(rw,sync,no_root_squash,no_subtree_check)
/var/spool/mail 192.168.1.2(rw,sync,root_squash,no_subtree_check)
/opt 192.168.1.2(rw,async,no_root_squash,no_subtree_check)

/tftpboot/nodes/192.168.1.2/ 192.168.1.2(rw,sync,no_root_squash,no_subtree_check
)
~
```

修改完畢後重新啟動 nfs server，可執行/etc/init.d/nfs restart。

B. 修改/tftpboot/nodes/[node 的 ip]/etc/fstab，一樣將/opt 的設定從 ro 改為 rw

```
192.168.1.1:/tftpboot/nodes/192.168.1.2/var      /var    nfs     rw,hard,intr,nf
svers=3,tcp,,defaults        0 0
192.168.1.1:/tftpboot/nodes/192.168.1.2/root     /root   nfs     rw,hard,intr,nf
svers=3,tcp,,defaults        0 0
192.168.1.1:/usr             /usr        nfs    ro,soft,nfsvers=3,tcp,,defaults
      0 0
192.168.1.1:/home            /home       nfs    rw,hard,intr,nfsvers=3,tcp,,defa
ults        0 0
192.168.1.1:/var/spool/mail /var/spool/mail       nfs     rw,hard,intr,nfsvers=3
,tcp,,defaults        0 0
none                    /proc           proc    defaults        0 0
tmpfs                   /tmp            tmpfs   defaults        0 0
#/dev/fd0                /mnt/floppy     auto    noauto,owner,kudzu 0 0
#/dev/sr0                /mnt/cdrom      iso9660 iocharset=cp950,noauto,owner,kud
zu,ro 0 0
192.168.1.1:/opt             /opt        nfs    rw,soft,nfsvers=3,tcp,,defaults
      0 0
192.168.1.1:/tftpboot/node_root/var/lib/rpm /var/lib/rpm       nfs     ro,soft,n
fsvers=3,tcp,,defaults        0 0
tmpfs                   /dev/shm                tmpfs   defaults        0 0
devpts                  /dev/pts                devpts  gid=5,mode=620  0 0
sysfs                   /sys                    sysfs   defaults        0 0
proc                    /proc                   proc    defaults        0 0
                                                                18,1          80%
```

修改完畢後 node 重開或是在 node 上以 root 執行 mount -a 即擁有/opt 寫入權限。

(2)讓 node 有/usr 的權限

A.使用 root 身分修改/etc/exports，將/usr 的設定從 ro 改成 rw

```
# Generated by DRBL at 16:46:58 2013/03/20
/tftpboot/node_root 192.168.1.2(ro,async,no_root_squash,no_subtree_check)
/usr 192.168.1.2(rw,async,no_root_squash,no_subtree_check)
/home 192.168.1.2(rw,sync,no_root_squash,no_subtree_check)
/var/spool/mail 192.168.1.2(rw,sync,root_squash,no_subtree_check)
/opt 192.168.1.2(rw,async,no_root_squash,no_subtree_check)

/tftpboot/nodes/192.168.1.2/ 192.168.1.2(rw,sync,no_root_squash,no_subtree_check
)
```

修改完畢後重新啟動 nfs server，可執行/etc/init.d/nfs restart。

B.以 root 身分修改/tftpboot/node_root/sbin/init，大約在 279 行，修改前為 do_nfs_mount $nfsserver:/usr /usr $RO_NFS_EXTRA_OPT $NFS_OPT_TO_ADD，將 RO 改為 RW

```
# NFS-based /usr is necessary both for normal mode and DRBL SSI mode
echo -n "Mounting NFS dir /usr... "
do_nfs_mount $nfsserver:/usr /usr $RW $NFS_EXTRA_OPT $NFS_OPT_TO_ADD
NFS_OPT_TO_ADD_FLAG=$((NFS_OPT_TO_ADD_FLAG + $?))
echo "done!"
```

node 重新啟動，即擁有/usr 寫入權限。

文章到這邊，就告一段落，除了最基本的架設，還加上一些奇怪需求的解決方法，希望此教學文章能對一些正準備架設 cluster 的人們有立即性的幫助，文章中若有任何錯，還請各位讀者不吝指教。由於小弟我要準備去保家衛國了，未來放假有空時會再將文章加入一些更新內容，如 TORQUE 排程軟體、MPICH 2 的安裝等。