Enterprise Linux 實戰講座－RHEL High-Availability Solution（二）

Step by Step 打造 High Availability NFS Cluster

上期文章筆者已介紹 High Availability 解決方案中常見的技術及觀念，本期我們將利用兩台 x86 伺服器、Proware Rackmount 3000I3 磁碟陣列及 RedHat Cluster Suite 打造建置 High Availability NFS Cluster。

簡介

NFS 是 Unix 世界最通用分享檔案系統的方式，NFS Server 常扮演企業中重要的 File Server。但是實務應用上時常遇到一個問題，當 NFS Serverc Crash 會造成所有 NFS Client 相關 Session 無法正常運作及終止。問題嚴重時，甚至 NFS Client 及 Server 都需 reboot 才能解決問題。關於此問題，筆者嘗試用 RedHat Cluster Suite 架構 HA 的機制來解決此問題，當主要的 NFS Server 故障時，另一台備援的 NFS Server 能立刻接手繼續提供 NFS 服務，

測試環境

軟體

- **RedHat Enterprise Linux ES 版 Update 2**
- **RedHat Cluster Suite Update 2**

硬體

- x 86 伺服器兩台
- 兩張網路卡
- Adaptec SCSI Card 29320-R 兩張
- Proware Rackmount 3000I3 磁碟陣列

圖 1：Proware Rackmount 3000I3 磁碟陣列

實作步驟：

1. High Availability NFS Cluster 架構規劃

筆者測試架構的簡圖如圖 2。主要伺服器 node1 的 ip 爲 192.168.0.201，備援伺服器 node2 的 ip 爲 192.168.0.202，整個 HA Cluster 對外的 service ip 爲 192.168.0.200。

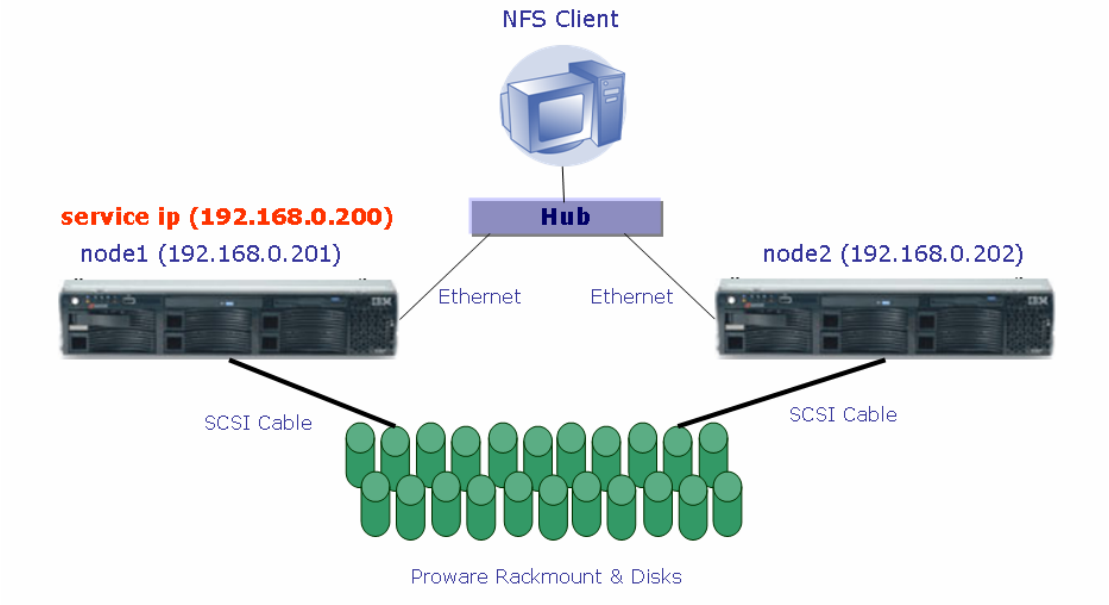

圖 2：High Availability NFS Cluster 架構圖

2.設定 Proware Rackmount 3000I3 磁碟陣列

這款磁碟陣列，很容易便可從面板上的「Sel」鍵設定成 RAID 5 的磁碟（可參

考其安裝手冊第五章）。假如 SCSI 排線連接正確，從 node1 或 node2 執行「hwbrowser」應可看到 Proware Rackmount 上的 Share Disk（圖 3）。

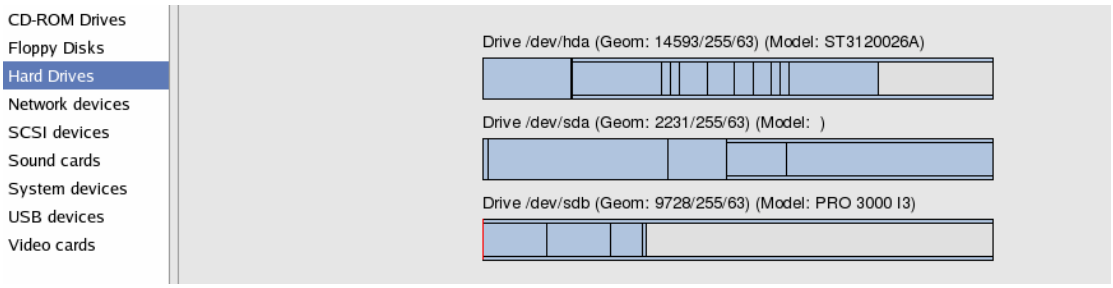

圖 3：hwbrowser 畫面

3.安裝 Red Hat 叢集管理員套件

用 root 登入 node1 安裝 clumanager 與 redhat-config-cluster 套件才能設定 Red Hat 叢集管理員，將光碟收入光碟機中，便會自動執行安裝程式（圖 4）。請選取「clumanager」及「redhat-config-cluster」套件進行安裝（圖 5）；在 node2 亦重複此步驟。

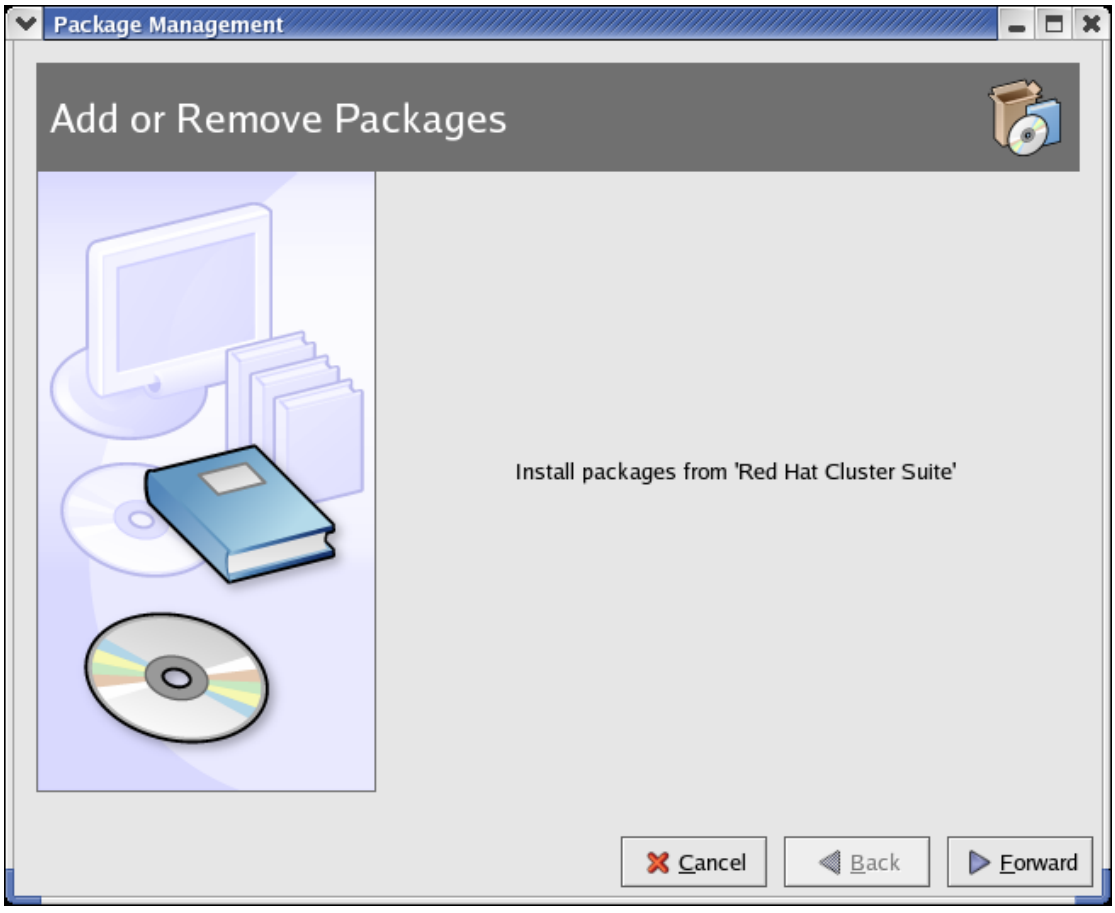

圖 4：Red Hat 叢集管理安裝畫面

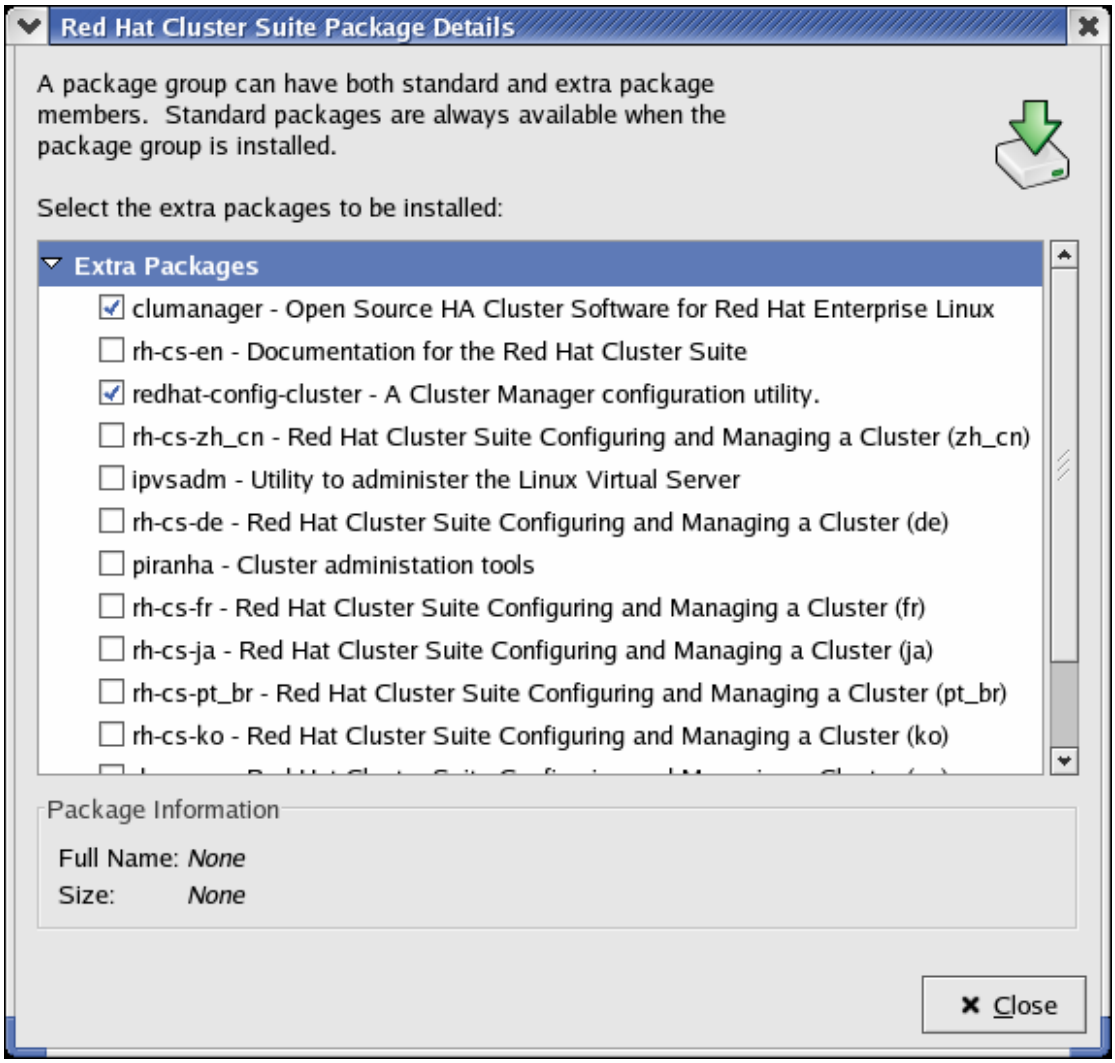

圖 5：選取「clumanager」及「redhat-config-cluster」套件

或利用 rpm 方式安裝：

```
[root@node2 RPMS]# rpm -ivh clumanager-1.2.12-1.i386.rpm
[root@node2 RPMS]# rpm -ivh redhat-config-cluster-1.0.2-1.1.noarch.rpm
[root@node1 root]# rpm -qa | grep clu
clumanager-1.2.12-1
redhat-config-cluster-1.0.2-1.1
```

4.編寫 /etc/hosts

```
[root@node1 root]#vi /etc/hosts
127.0.0.1                  localhost.localdomain localhost
192.168.0.201               node1.example.com node1
192.168.0.202               node2. example.com node2
並將此檔 scp 至 node2
[root@node1 root]#scp /etc/hosts node2:/etc/hosts
```

5.設定叢集共用分割區（Configuring Shared Cluster Partitions）
共用叢集共用分割區是用來存放叢集的狀態資訊，存放內容如下：

- 叢集鎖定狀態
- 服務狀態
- 設定資訊

每一個成員將會定期的寫入它的服務狀態到共用的儲存空間，共需要建立兩個叢集共用分割區：primary 及 shadow。假如 primary 的共用分割區毀損了，叢集成員還可以從 shadow 或備援共用分割區讀取資訊，並且在同時修復 primary 分割區，資料的一致性是透過檢查碼(checksums) 來維護，而且任何在兩個分割區間的不一致資料將會自動地修正。

假如一個成員在開機時無法寫入兩個共用的分割區，它將不被允許加入叢集。叢集共用分割區的需求如下：

- 兩個分割區至少需要 10MB 的空間
- 共用的分割區必須是 raw 裝置，它們不能含有檔案系統。
- 共用分割區只能由叢集的狀態與設定資訊所使用

```
[root@node1 root]# fdisk -l /dev/sdb
Disk /dev/sdb: 80.0 GB, 80018931712 bytes
255 heads, 63 sectors/track, 9728 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

Command (m for help): p ← 印出現有 partition table
Disk /dev/sdb: 80.0 GB, 80018931712 bytes
255 heads, 63 sectors/track, 9728 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System

Command (m for help): n  ← 新增第一個共用分割區
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
```

```
First cylinder (1-9728, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-9728, default 9728): +20M


Command (m for help): n← 新增第二個共用分割區
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (4-9728, default 4):
Using default value 4
Last cylinder or +size or +sizeM or +sizeK (4-9728, default 9728): +20M

Command (m for help): p

Disk /dev/sdb: 80.0 GB, 80018931712 bytes
255 heads, 63 sectors/track, 9728 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1               1           3       24066   83  Linux
/dev/sdb2               4           6       24097+  83  Linux
Command (m for help): w ← 寫入 partition table
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: 裝置或系統資源忙碌中.
The kernel still uses the old table.
The new table will be used at the next reboot.
Syncing disks.
[root@node1 root]# reboot ← 重新開機
```

6.建立 raw 裝置

在設定叢集共用分割區後，請在分割區上建立 raw 裝置，共用的分割區上不能

含有檔案系統。要建立一個 raw 裝置，必需編輯 /etc/sysconfig/rawdevices 檔案來繫結一個 raw 字元裝置到適當的區塊裝置以使得該 raw 裝置可以被開啓、讀取與寫入。

```
[root@node1 root]# cat    /etc/sysconfig/rawdevices
# raw device bindings
# format:    <rawdev> <major> <minor>
#                 <rawdev> <blockdev>
# example: /dev/raw/raw1 /dev/sda1
#                 /dev/raw/raw2 8 5
/dev/raw/raw1 /dev/sdb1
/dev/raw/raw2 /dev/sdb2
```

```
[root@node2 root]# cat    /etc/sysconfig/rawdevices
# raw device bindings
# format:    <rawdev> <major> <minor>
#                 <rawdev> <blockdev>
# example: /dev/raw/raw1 /dev/sda1
#                 /dev/raw/raw2 8 5
/dev/raw/raw1 /dev/sdb1
/dev/raw/raw2 /dev/sdb2
```

編輯完/etc/sysconfig/rawdevices 檔案後，可以重新開機 或者是執行下列指令來使其生效。

```
#service rawdevices restart
```

#使用 raw -aq 指令可查詢所有的 raw 裝置

```
[root@node1 root]# raw -aq
/dev/raw/raw1:    bound to major 8, minor 17
/dev/raw/raw2:    bound to major 8, minor 18
```

6.訂定叢集名稱

- 選擇『**主選單**』=>『**系統設定**』=>伺服器設定』=>『**叢集**』。
- 或在 shell 提示符號下輸入 redhat-config-cluster 指令。

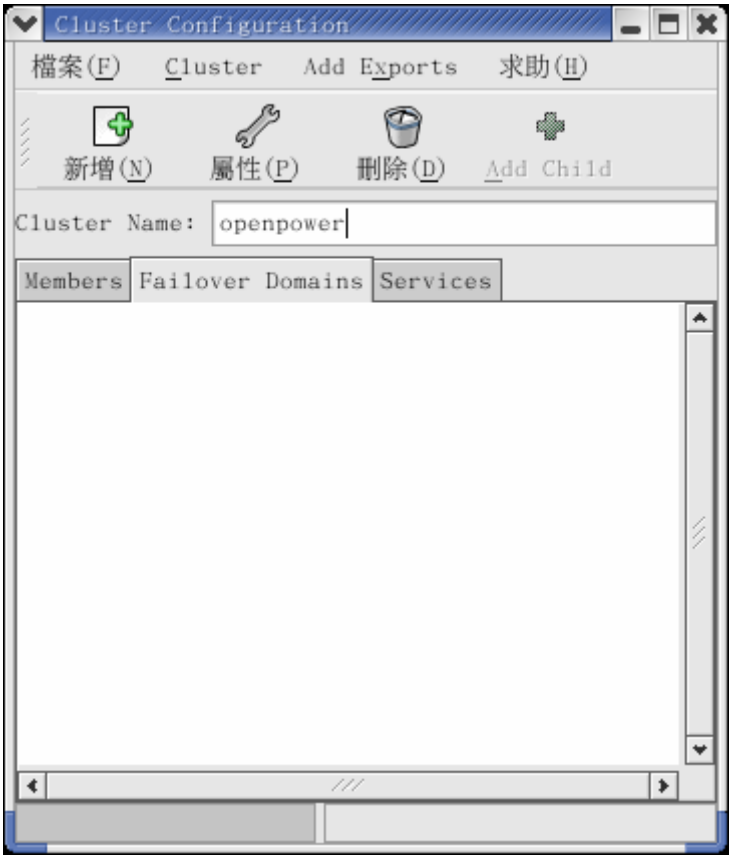

圖 6：訂定叢集名稱

7.設定 Share Raw Device

選擇 redhat-config-cluster 上的「Cluster」=>「Shared State」便可看到圖 7 的畫面，填入正確的 Raw Device。

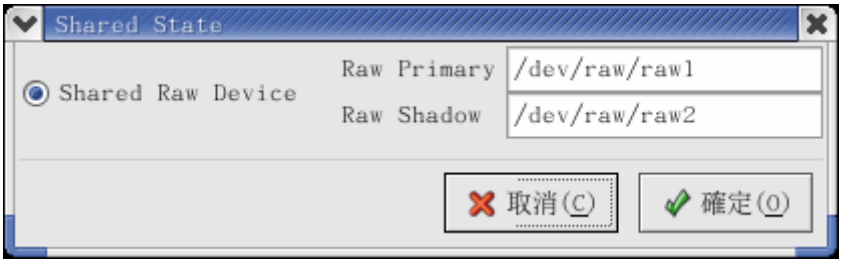

圖 7：Share Raw Device

8. 新增 Cluster Member

選取「Member」，再點選「新增」的按鈕（圖 8）。程式將會要求輸入 Member 名稱。請輸入 Cluster 中一部系統的主機名稱或位址，請注意每一個 Member 必須位於與執行 redhat-config-cluster 的機器在同一子網路中，而且必須在 DNS 或每一部叢集系統的 /etc/hosts 檔案中已經定義了。請新增兩個 Cluster Member「node1」及「node2」。（圖 9）

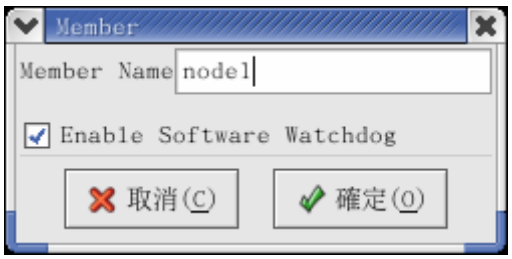

圖 8：新增 Cluster Member

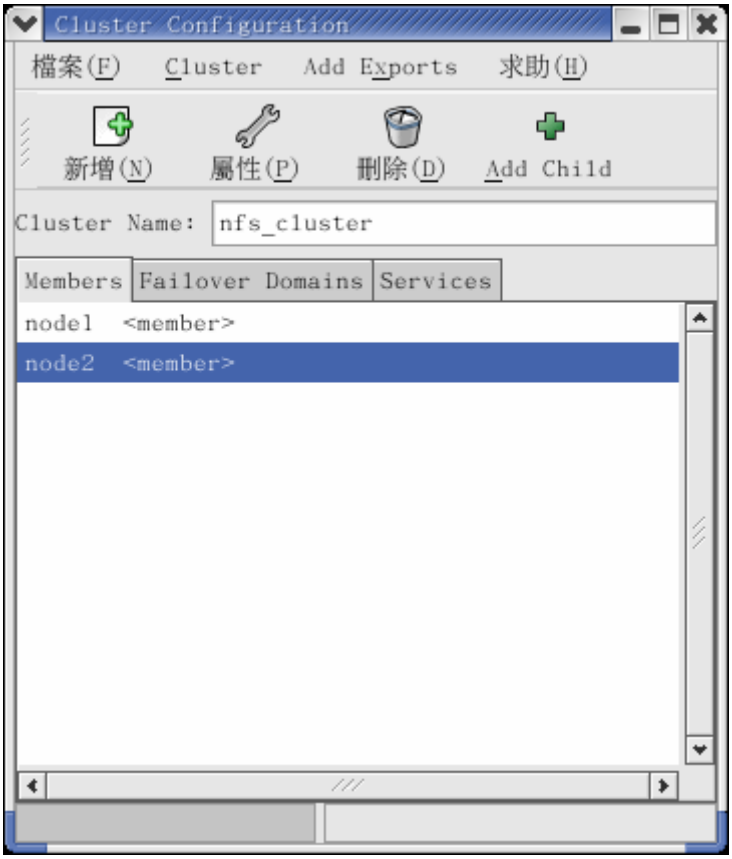

圖 9：nfs_cluster 的成員

9.設定 Failover Domain

「Failover Domain」是指提供某種服務，可互相備援的主機集合。一個 Failover Domain 含有下列的特徵：

■ Unrestricted — 指派給這個網域的一項服務可以在任何可用的成員（包括不列在此 Failover Domain 內的主機）上執行

■ Restricted — 指派給這個網域的一項服務只能可以在 Failover Domain 可用的成員上執行。

■ Unordered — 當一項服務被指派給一個 Unordered 的 Failover Domain，服務將執行於其上的成員將於未經優先順序排列的可用 Failover Domain 成員中來挑選。

■ Ordered —允許您在一個 Failover Domain 中的成員間指定一個優先順序，在清單頂部的成員便是最優先的，接下來便是清單中的第二個成員，依此類推。

選擇「**Failover Domain**」的標籤頁，再點選「**新增**」的按鈕。將會出現如圖 10 所示的「**Failover Domain**」對話視窗。

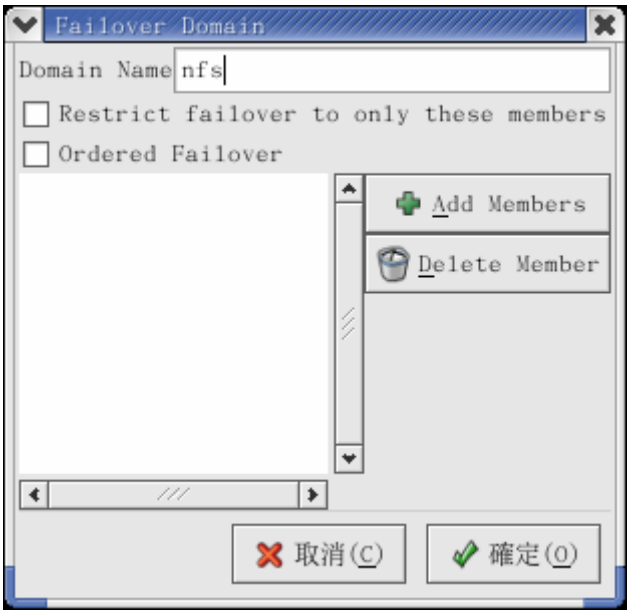

圖 10：「Failover Domain」對話視窗

勾選「Restrict failover to only these members」防止在這裡未列出的任何成員接手指派給這個網域的一項服務。
勾選「Ordered Failover」依據網域中成員清單的位置來決定接管服務的優先權，較優先的成員將位於頂端。

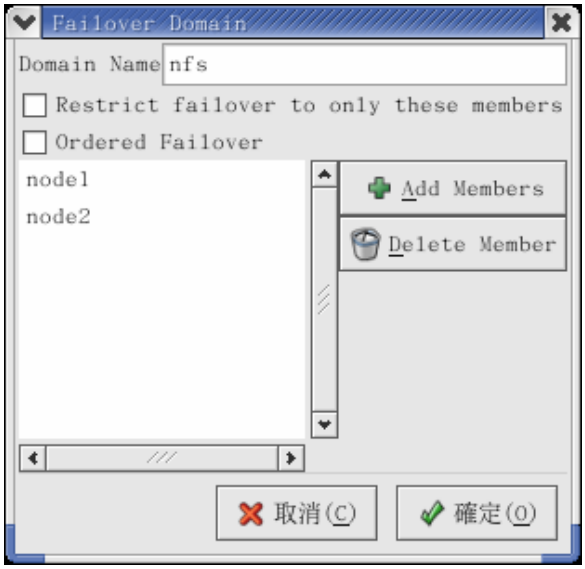

圖 11：設定「Failover Domain」的屬性

10.啓動 NFS 伺服器
要建立 High Availability NFS 服務，每一部叢集成員都必須啓動 NFS 服務

```
# service nfs start <-- 馬上啓動 NFS 服務
#chkconfig nfs on <-- 重開機後亦自動啓動 NFS 服務
```

還有一點需特別注意：檔案系統掛載以及叢集 NFS 服務所相關的匯出不應該收錄在/etc/fstab 或/etc/exports 檔案中。

11.利用「**NFS Druid**」來快速設定一個用戶端可存取的 NFS 共享

a.啓動 Cluster 服務：「**Cluster**」=>「**Start Cluster Service**」

b. 啓動 NFS 設定精靈：「**Cluster**」=>「**Configure** 」=>「**Add Exports**」=>「**NFS**」

你將會看到如圖 12 的畫面，然後按下「**Forward**」。

圖 12：NFS Druid 畫面

c.利用「**NFS Druid**」將/dev/sdb export 給 NFS Client，目錄名稱爲 /data，步驟如圖 13~圖 15。這個部份有幾點需特別注意：

■ Service Name—在叢集中用來辨識這個服務所使用的名稱

■ **Service IP**—叢集的 NFS 服務將被指定一個浮動的 IP 位址，以用來與叢集伺服器的 IP 位址做區分，這個 IP 位址稱爲「**Service IP**」。NFS Client 存取 NFS, 是透過 Service IP 而不是 node1 或 node2 的真實 IP。

這是爲了不讓 NFS Client 知道是叢集內那台伺服器提供服務。這個浮動的 IP 位址將會設定在主要伺服器（Master，在本例中是 node1）。藉由使用這個方法，NFS 用戶端只知道浮動 IP 位址，而不知道已經配置了叢集的 NFS 伺服器的事實。

如果主要伺服器故障（node1），則此 Service IP 會移轉至備援伺服器（node2），如些一來，當主要伺服器故障，備援伺服器接管 NFS 服務，NFS Client 完全不用做任何異動。

■ **避免使用 exportfs –r**

exportfs -r 指令將移除在 /etc/exports 檔案中沒有特別指定的任何 export 資訊，執行這個指令將會導致叢集的 NFS 服務變得無法被存取（直到服務被重新啓動）。由於這個原因，建議您避免於已設定高存取性之 NFS 服務的叢集中使用 exportfs -r 指令。如要回復不經意使用 exportfs -r 指令，必須先停止然後再重新啓動 NFS 叢集服務。

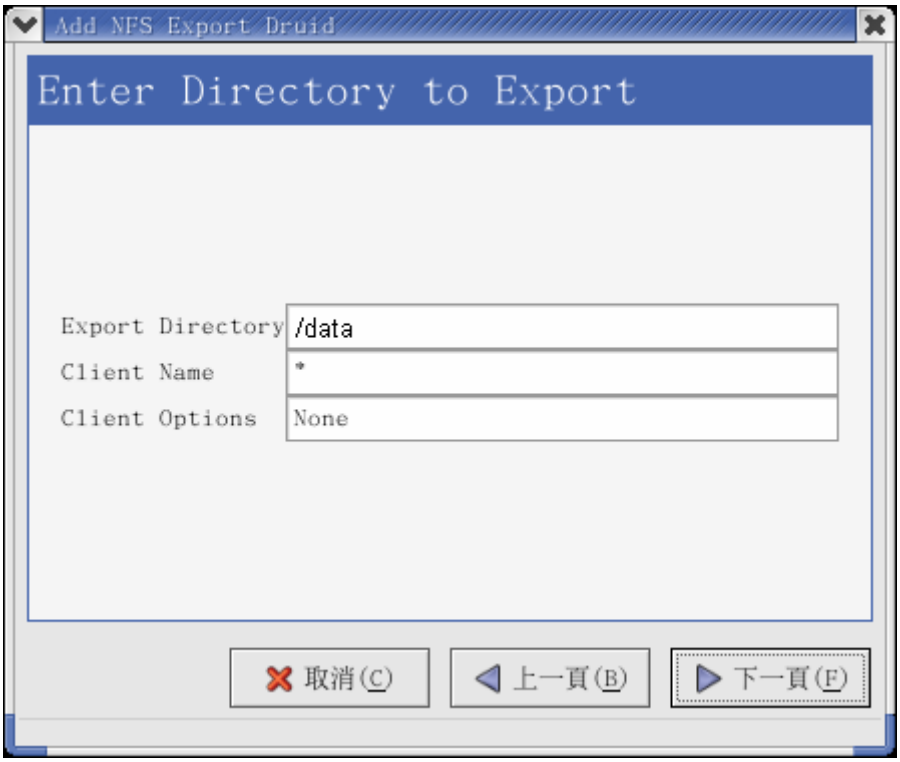

圖 13：export 與 Client 選項

圖 14：設定 Service Name 與 Service IP

圖 15：選擇 export 的 Device

在「**NFS Druid**」的最後，點選「**套用**」來建立這個服務。並從「**叢集設定工具**」中選擇「**檔案**」=>「**儲存**」來儲存設定。

12.將 node1 的/etc/cluster.xml 複製至 node2

叢集設定工具將叢集服務與系統程式、叢集成員與叢集服務的資訊儲存在 /etc/cluster.xml 設定檔。所以當 node1 已完成設定後，請將/etc/cluster.xml 複製至 node2 上，並啓動 node2 上的「clumanager」程式。

```
[root@node1 root]# scp /etc/cluster.xml node2:/etc/cluster.xml
root@node2's password:
cluster.xml                                   100% 1405
```

```
[root@node2 root]# service clumanager start
Starting Red Hat Cluster Manager...
Loading Watchdog Timer (softdog):                    [  OK  ]
Starting Quorum Daemon:
```

13.修改 node1 及 node2 上的/etc/syslog.conf 指定 Cluster Log 存放位置。

```
[root@node1 root]# vi /etc/syslog.conf
# Add for cluster
local4.*              /var/log/cluster

# service syslog restart
```

```
[root@node2 root]# vi /etc/syslog.conf
# Add for cluster
local4.*              /var/log/cluster

# service syslog restart
```

14.查看叢集狀態

選擇 『**叢集**』 => 『**設定**』 顯示**叢集**狀態（圖 16）。

圖 16：叢集狀態圖

15.測試 High Availability NFS Cluster，下面是筆者的測試過程，用此來證明當 node1 Crash 時，node2 確實可 Take Over NFS 服務，達到 High Availability 目的。

```
[root@ftp root]# showmount -e 192.168.0.200
Export list for 192.168.0.200:
/data *
```

```
[root@ftp root]#mkdir /mnt/nfs
[root@ftp root]#mount 192.168.0.200:/data /mnt/nfs
[root@ftp root]#mount
.....
.....
192.168.0.200:/data on /mnt/nfs type nfs (rw,addr=192.168.0.200)
```

[root@ftp root]# cd /mnt/nfs
此時將 node1 電源關掉或執行「poweroff」指令模擬 node1 Crash
[root@ftp nfs]# ls
此時大約會 hang 5~10 秒，node1 便接管 NFS 服務，ls 指令執行結果便會出現。

後記

本文筆者利用 RedHat Cluster Suite 加上 SCSI Share Disk 建置 High Availability NFS Cluster。由於 SAN 的當紅，而且愈來愈多廠商對於 Linux HA 解決方案感興趣，下期文章筆者將繼續介紹如何在 SAN 環境下實作 RedHat High Availability Cluster。