



「**研磨 Red Hat Certified Architect – RH436 篇**」

【RHEL5 上的 iSCSI 環境】

最近看了大前研一的「**研磨商業力**」，想要成為一個好的 Architect，真的是得好好研磨研磨，所以筆者東施效顰地把專欄名稱改為「**研磨 RHCA**」！上期文章介紹如何 iSCSI 概念及如何在 RHEL 4 上建置 iSCSI 環境，這期文章接著介紹在 RHEL 5 上如何實作 iSCSI 環境。



1 在 RHEL 5.1 上實作 iSCSI 環境

1.1 iSCSI on RHEL 5.1 環境規劃

老話一句，先規劃！先將下列表格填好。

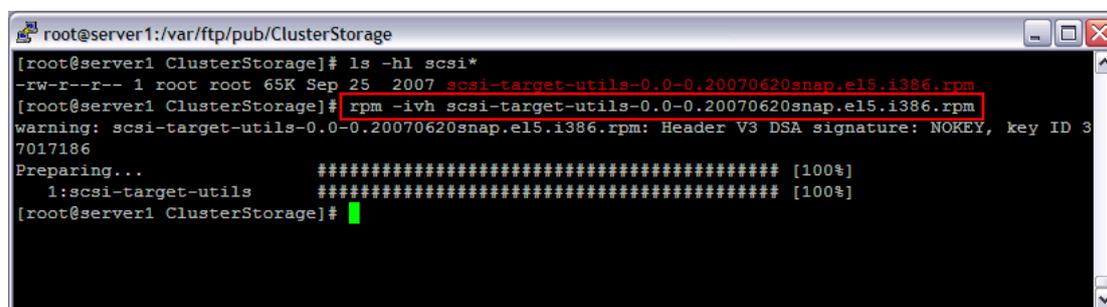
表 1：iSCSI 環境規劃表

功能	作業系統	主機名稱	IP Address
iSCSI-target server	RHEL 5.1	server1.example.com	192.168.0.254
iSCSI Initiator (node1)	RHEL 5.1	node1.example.com	192.168.0.101

1.2 實作 iSCSI Target Server on RHEL 5.1

1. 安裝 iSCSI-target 軟體

RHEL 4 光碟中並無內附 iSCSI target 軟體，所以必需額外下載安裝。總算到 RHEL 5.1，Red Hat 已將 iSCSI target 軟體整合進來了，就無需再去下載並自行編譯了！讀者可以原版光碟中 ClusterStorage 目錄中便可找到此套件 `scsi-target-utils-0.0-0.20070620snap.el5.i386.rpm`，可利用 `rpm`、`yum` 或是「system-config-packages」圖形套件管理工具安裝均可。



```
root@server1:/var/ftp/pub/ClusterStorage
[root@server1 ClusterStorage]# ls -hl scsi*
-rw-r--r-- 1 root root 65K Sep 25 2007 scsi-target-utils-0.0-0.20070620snap.el5.i386.rpm
[root@server1 ClusterStorage]# rpm -ivh scsi-target-utils-0.0-0.20070620snap.el5.i386.rpm
warning: scsi-target-utils-0.0-0.20070620snap.el5.i386.rpm: Header V3 DSA signature: NOKEY, key ID 37017186
Preparing... ##### [100%]
 1:scsi-target-utils ##### [100%]
[root@server1 ClusterStorage]#
```

圖 1：利用 rpm 指令安裝 `scsi-target-utils-0.0-0.20070620snap.el5.i386.rpm`



```
root@server1:/var/ftp/pub
[root@server1 pub]# yum -y install scsi-target-utils
Loading "installonlyn" plugin
Loading "kmod" plugin
Loading "protectbase" plugin
Loading "skip-broken" plugin
Loading "downloadonly" plugin
Loading "security" plugin
Loading "changelog" plugin
Loading "rhnpplugin" plugin
This system is not registered with RHN.
RHN support will be disabled.
Setting up Install Process
Setting up repositories
Reading repository metadata in from local files
0 packages excluded due to repository protections
Parsing package install arguments
Resolving Dependencies
--> Populating transaction set with selected packages. Please wait.
--> Downloading header for scsi-target-utils to pack into transaction set.
scsi-target-utils-0.0-0.2 100% |=====| 3.6 kB 00:00
--> Package scsi-target-utils.i386 0:0.0-0.20070620snap.el5 set to be updated
--> Running transaction check

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
scsi-target-utils i386 0.0-0.20070620snap.el5 ClusterStorage 65 k
=====

Transaction Summary
=====
Install 1 Package(s)
Update 0 Package(s)
Remove 0 Package(s)
=====

Total download size: 65 k
Downloading Packages:
(1/1): scsi-target-utils- 100% |=====| 65 kB 00:00
Running Transaction Test
warning: scsi-target-utils-0.0-0.20070620snap.el5: Header V3 DSA signature: NOKEY, key ID 37017186
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
Installing: scsi-target-utils ##### [1/1]

Installed: scsi-target-utils.i386 0:0.0-0.20070620snap.el5
Complete!
[root@server1 pub]#
```

圖 2：利用 yum 指令安裝 scsi-target-utils-0.0-0.20070620snap.el5.i386.rpm

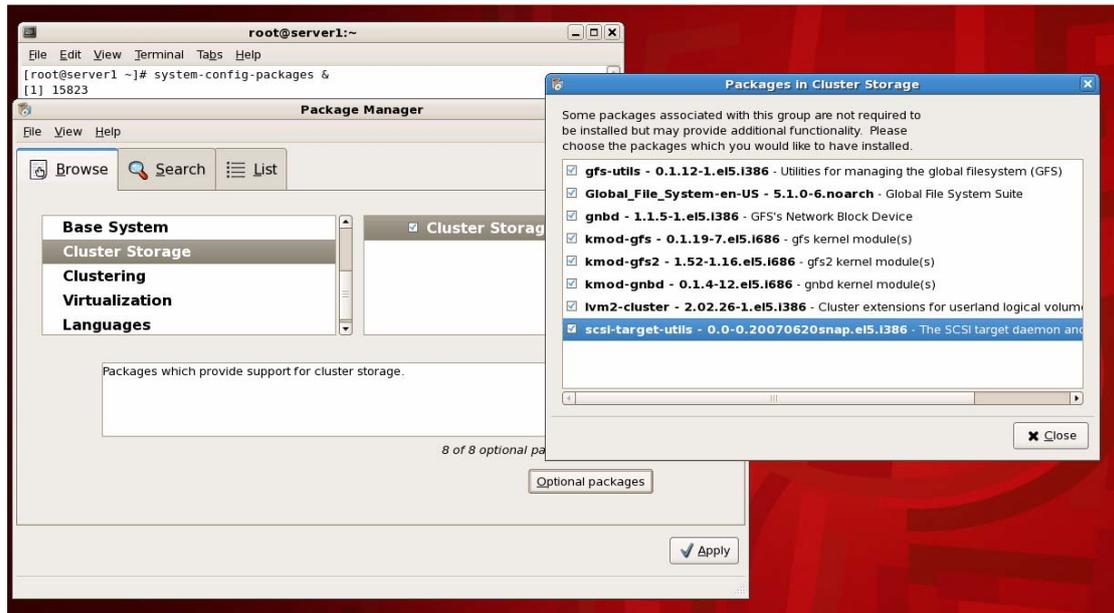


圖 3：利用 system-config-packages 安裝 scsi-target-utils 套件

2. 利用 fdisk 及 partprobe 指令新增 Partiton。

在 target Server 新增兩個硬碟分割區，無需格式化，讓給 iSCSI Initiator 存取，此將會成為 Initiator 中的兩顆硬碟。筆者新增兩個各為 1GB (/dev/sdb2) 及 2GB 的 partition (/dev/sdb3)。

3. 啟動 tgttd 服務

```
## service tgttd start
# chkconfig tgttd on
```

4. 建立 target device

接下來的步驟，和 RHEL 4 上的作法差異頗大，RHEL 5 是利用「tgtadm」指令來設定而不是編寫 vi /etc/ietd.conf。筆者覺得 tgtadm 指令相關參數真的是有點難記！幸好 scsi-target-utils 套件中包含如何利用 tgtadm 指令建立 target device 的步驟，建議讀者一定要把 /usr/share/doc/scsi-target-utils-0.0/README.iscsi 打開閱讀。



```
root@server1:/var/ftp/pub
[root@server1 pub]# rpm -ql scsi-target-utils.i386 0:0.0-0.20070620snap.e15
/etc/rc.d/init.d/tgtd
/usr/sbin/tgtadm
/usr/sbin/tgtd
/usr/share/doc/scsi-target-utils-0.0
/usr/share/doc/scsi-target-utils-0.0/README
/usr/share/doc/scsi-target-utils-0.0/README.iscsi
/usr/share/man/man8/tgtadm.8.gz
package 0:0.0-0.20070620snap.e15 is not installed
[root@server1 pub]#
```

圖 4：/usr/share/doc/scsi-target-utils-0.0/README.iscsi

筆者欲新增 iqn 名為 iqn.2008-08.com.example:rhcs-storage 的 target device。

『iSCSI Qualified Name』（簡稱：iqn），格式通常如下：

iqn.yyyy-mm.<reversed domain name>[:identifier]

格式	意義	範例
yyyy-mm	年份-月份	2008-08
reversed domain name	把網域名稱反過來寫，通常把公司網域名稱反過來寫。	com.example
identifier	識別字，通常註明這個儲存空間的用途，	rhcs-storage

新增 target device 指令如下：

```
# tgtadm --lld iscsi --op new --mode target --tid 1 -T iqn.2008-08.com.example:rhcs-storage
```

執行後，可利用下列指令查看 target device：

```
# tgtadm --lld iscsi --op show --mode target
```

```
root@server1:/var/ftp/pub
[root@server1 pub]# tgtadm --lld iscsi --op new --mode target --tid 1 -T iqn.2008-08.com.example:rhcs-storage
[root@server1 pub]# tgtadm --lld iscsi --op show --mode target
Target 1: iqn.2008-08.com.example:rhcs-storage
  System information:
    Driver: iscsi
    Status: running
  I T nexus information:
  LUN information:
    LUN: 0
    Type: controller
    SCSI ID: deadbeaf1:0
    SCSI SN: beaf10
    Size: 0
    Backing store: No backing store
  Account information:
  ACL information:
[root@server1 pub]#
```

圖 5：新增 target device



5. 將 target Server 新建的 partition 加入 target device。

```
# tgtadm --lld iscsi --op new --mode logicalunit --tid 1 --lun 1 -b /dev/sdb2
```

```
# tgtadm --lld iscsi --op new --mode logicalunit --tid 1 --lun 2 -b /dev/sdb3
```

將兩個 partitions 加入 target device 後，而利用下列指令查看 target device 的內容：

```
# tgtadm --lld iscsi --op show --mode target
```

```
root@server1:~  
[root@server1 ~]# tgtadm --lld iscsi --op new --mode logicalunit --tid 1 --lun 1 -b /dev/sdb2  
[root@server1 ~]# tgtadm --lld iscsi --op new --mode logicalunit --tid 1 --lun 2 -b /dev/sdb3  
[root@server1 ~]# tgtadm --lld iscsi --op show --mode target  
Target 1: ign.2008-08.com.example:rhcs-storage  
System information:  
  Driver: iscsi  
  Status: running  
I_T nexus information:  
LUN information:  
  LUN: 0  
    Type: controller  
    SCSI ID: deadbeaf1:0  
    SCSI SN: beaf10  
    Size: 0  
    Backing store: No backing store  
  LUN: 1  
    Type: disk  
    SCSI ID: deadbeaf1:1  
    SCSI SN: beaf11  
    Size: 980M  
    Backing store: /dev/sdb2  
  LUN: 2  
    Type: disk  
    SCSI ID: deadbeaf1:2  
    SCSI SN: beaf12  
    Size: 1G  
    Backing store: /dev/sdb3  
Account information:  
ACL information:  
[root@server1 ~]#
```

圖 6：新增 target device 內的 disk

5. 允許 initiator 主機 node1 (192.168.0.1) 可以存取此 Target device。

```
# tgtadm --lld iscsi --op bind --mode target --tid 1 -l 192.168.0.101
```



```
root@server1:~  
[root@server1 ~]# tgtadm --lld iscsi --op bind --mode target --tid 1 -I 192.168.0.101  
[root@server1 ~]# tgtadm --lld iscsi --op show --mode target  
Target 1: iqn.2008-08.com.example:rhcs-storage  
System information:  
  Driver: iscsi  
  Status: running  
I T nexus information:  
LUN information:  
  LUN: 0  
    Type: controller  
    SCSI ID: deadbeaf1:0  
    SCSI SN: beaf10  
    Size: 0  
    Backing store: No backing store  
  LUN: 1  
    Type: disk  
    SCSI ID: deadbeaf1:1  
    SCSI SN: beaf11  
    Size: 980M  
    Backing store: /dev/sdb2  
  LUN: 2  
    Type: disk  
    SCSI ID: deadbeaf1:2  
    SCSI SN: beaf12  
    Size: 1G  
    Backing store: /dev/sdb3  
Account information:  
  ACL information:  
    192.168.0.101  
[root@server1 ~]#
```

圖 7：target device ACL information

6. 修改/etc/rc.local

因為上述指令的設定結果，重新開機後便會消失，而且現版本的 target server，Red Hat 尚未修改成 System V 服務，所以如果需要重新開機上述設定仍有效，只能將上述指令寫至/etc/rc.local，讓系統重新開機時再自動執行一次。

```
[root@server1 ~]# cat /etc/rc.local  
  
#!/bin/sh  
  
#  
  
# This script will be executed *after* all the other init scripts.  
# You can put your own initialization stuff in here if you don't  
# want to do the full Sys V style init stuff.  
  
touch /var/lock/subsys/local  
  
#iSCSI target device setting - Alex YM Lin (2008.08.21)  
  
tgtadm --lld iscsi --op new --mode target --tid 1 -T iqn.2008-08.com.example:rhcs-storage  
tgtadm --lld iscsi --op new --mode logicalunit --tid 1 --lun 1 -b /dev/sdb2  
tgtadm --lld iscsi --op new --mode logicalunit --tid 1 --lun 2 -b /dev/sdb3
```



```
tgtadm --lld iscsi --op bind --mode target --tid 1 -l 192.168.0.101
```

1.3 實作 iSCSI Initiator on RHEL 5.1

在 RHEL 5.1 上實作 initiator 步驟就簡單多了，詳細過程如下：

1. 安裝 iscsi-initiator-utils 軟體

iscsi-initiator-utils 軟體位於原版光碟中 Server 目錄中，利用 rpm 或 yum 指令安裝即可。

```
[root@node1 ~]# rpm -ivh ftp://192.168.0.254/pub/Server/iscsi*
Retrieving ftp://192.168.0.254/pub/Server/iscsi-initiator-utils-6.2.0.865-0.8.el
5.i386.rpm
warning: /var/tmp/rpm-xfer.97NZFF: Header V3 DSA signature: NOKEY, key ID 370171
86
Preparing...                               ##### [100%]
 1:iscsi-initiator-utils                    ##### [100%]
```

圖 8：安裝 iscsi-initiator-utils 軟體

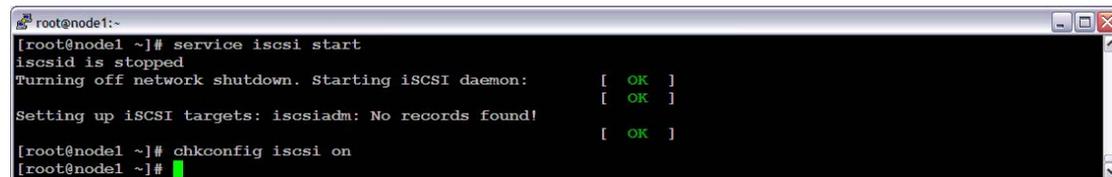
2. 在指 iSCSI initiator 主機上指定 initiator alias name

```
# echo "InitiatorAlias=node1" >> /etc/iscsi/initiatorname.iscsi
```

3. 啟動 iscsi daemon

```
# service iscsi start
```

```
# chkconfig iscsi on
```



```
root@node1:~
[root@node1 ~]# service iscsi start
iscsid is stopped
Turning off network shutdown. Starting iSCSI daemon:          [ OK ]
Setting up iSCSI targets: iscsiadm: No records found!       [ OK ]
[root@node1 ~]# chkconfig iscsi on
[root@node1 ~]#
```

圖 9：啟動 iscsi daemon

4. 利用 iscsiadm 指令偵測 target Server 的允許 node1 存取的 iSCSI Device。

```
# iscsiadm -m discovery -t sendtargets -p 192.168.0.254:3260
```



```
root@node1:~  
[root@node1 ~]# iscsiadm -m discovery -t sendtargets -p 192.168.0.254:3260  
192.168.0.254:3260,1 iqn.2008-08.com.example:rhcs-storage  
[root@node1 ~]#
```

圖 10：偵測 target Server 所設定的 iSCSI device

5. 登入 iSCSI target Server

```
# iscsiadm -m node -T <target_iqn_name> -p 192.168.0.254:3260 -u
```

登入成功之後，利用 `fdisk -l` 或是 `hwbrowser` 檢查可以發現多出兩個硬碟。

```
root@node1:~  
[root@node1 ~]# iscsiadm -m node -T iqn.2008-08.com.example:rhcs-storage -p 192.168.0.254:3260 -l  
Login session [iface: default, target: iqn.2008-08.com.example:rhcs-storage, portal: 192.168.0.254,3260]  
[root@node1 ~]# fdisk -l  
  
Disk /dev/hda: 19.3 GB, 19327352832 bytes  
255 heads, 63 sectors/track, 2349 cylinders  
Units = cylinders of 16065 * 512 = 8225280 bytes  


| Device    | Boot | Start | End  | Blocks   | Id | System    |
|-----------|------|-------|------|----------|----|-----------|
| /dev/hda1 | *    | 1     | 33   | 265041   | 83 | Linux     |
| /dev/hda2 |      | 34    | 1991 | 15727635 | 8e | Linux LVM |

  
Disk /dev/sda: 1028 MB, 1028160000 bytes  
32 heads, 62 sectors/track, 1012 cylinders  
Units = cylinders of 1984 * 512 = 1015808 bytes  
  
Disk /dev/sda doesn't contain a valid partition table  
  
Disk /dev/sdb: 2056 MB, 2056320000 bytes  
64 heads, 62 sectors/track, 1012 cylinders  
Units = cylinders of 3968 * 512 = 2031616 bytes  
  
Disk /dev/sdb doesn't contain a valid partition table  
[root@node1 ~]#
```

圖 11：登入 iSCSI target Server

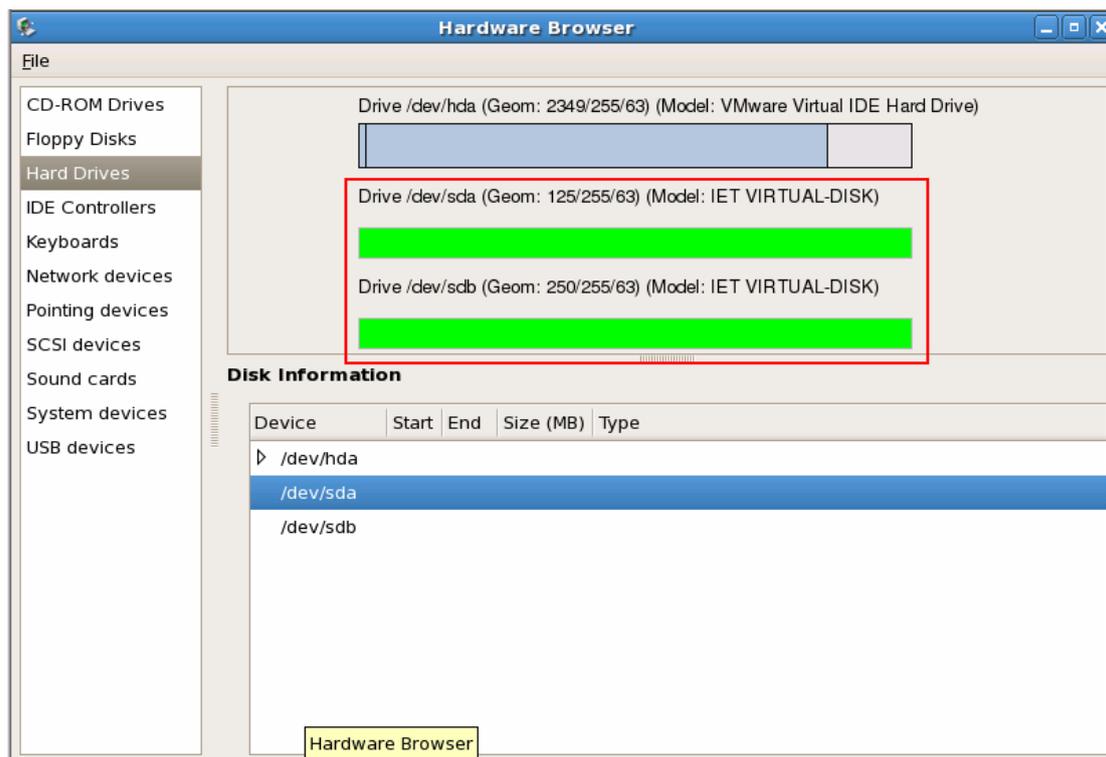


圖 12：利用 hwbrowser 檢查 iSCSI 硬碟

若是不再存取這個 target device，只需要利用下列指令 Log out，利用 fdisk -l 再檢查，新增的兩個硬碟就消失了！

```
# iscsiadm -m node -T iqn.2008-08.com.example:rhcs-storage -p 192.168.0.254:3260 -u
```



2 Open-E DSS Lite

看完上述在 RHEL 5 上建置 iSCSI target server 的步驟，有些讀者可能會覺得好像不是那麼人性化；六月號的 Run PC 針對市面的 iSCSI 儲存設備有專欄介紹，iThome 也有「IP SAN 產品介紹與實測」的專文，可參考下列網址：

<http://www.ithome.com.tw/001/0423/index2.html>。不過這些產品一般讀者也是所費不貲。所以筆者接下來介紹兩種「免費」又好用的 iSCSI target solution 「Open-E DSS Lite」及「openfiler」。

首先登場的是「Open-E DSS Lite」，open-e (<http://www.open-e.com/>) 開發了 iSCSI 嵌入式作業系統 flash rom OS，只要把 flash rom 插在一般的電腦上，就立刻變成一台 iSCSI Server；其實這 flash rom 就是 Linux 作業系統加上 iSCSI target 軟體加上 Web 的管理工具。

Open-E 目前由幾年前的十幾種版本，目前整合成三種正式產品：

- Open-E iSCSI-R3 (4TB, 8TB, 16TB...最大授權可至 64TB)
- Open-E NAS-R3 (4TB, 8TB, 16TB...最大授權可至 64TB)
- Open-E DSS (4TB, 8TB, 16TB...最大授權可至 64TB)

註：DSS 產品包含了 NAS 及與 iSCSI 兩種功能

如果單獨就產品功能來說，Open-E iSCSI 或許不是目前市面上功能最完整，但就產品穩定度與效能，應該是算不錯，支援 Online update 套件及 Restore CD (Restore CD 可用來將 Open-E iSCSI 寫回出廠預設值，也可以拿來更新至最新的版本)。

由下列網址 <http://www.open-e.com/service-and-support/product-compatibility/> 可得知 Open-E iSCSI 所支援硬體很廣泛，尤其是 2008 年更開始大力支援 IBM, DELL, HP 三家的硬體，這點對企業是很重要的，因為很多企業大都採購這三家的 PC 伺服器，Open-E iSCSI 支援這三家廠商常採用的 raid controller，對這些企業而言，便可利用 Open-E iSCSI 直接控制伺服器上的 raid controller 來建置 Hardware RAID。



Open-E NAS Network Attached Storage, iSCSI SAN Storage Area Network

Service & Support > Product Compatibility > HDD controller (hardware RAID)



- ▶ Product Registration
- ▶ Software Updates
- ▶ Downloads
- ▶ Demo-CD downloads
- ▶ **Product Compatibility**
 - Backup Agents
 - Certified Systems
 - Fibre Channel HBA (initiator mode)
 - Fibre Channel HBA (target mode)
 - **HDD controller (hardware RAID)**
 - Adaptec
 - AMCC - 3ware
 - Areca
 - Buslogic
 - Dell
 - HighPoint
 - HP Smart Array
 - **IBM**

IBM

	iSCSI	NAS-XSR	iSCSI-R.3	NAS-R.3	DSS
SATA RAID:					
ServeRAID 7t	-	-	✓	✓	✓
SAS:					
ServeRAID 8i	-	-	✓	✓	✓
ServeRAID 8k/8k-14	-	-	✓	✓	✓
ServeRAID 8k/8k-18	-	-	✓	✓	✓
ServeRAID 8s	-	-	✓	✓	✓

NOTICE: In SOHO product line hardware RAID controllers are not supported.

圖 13 : Open-E iSCSI 所支援 IBM hardware RAID 列表

Open-E 正式版產品約在 3 萬~5 萬之間，算是價格適中，如果讀者有 iSCSI 的需求，暫時又沒有預算採購 Open-E 正式版產品。可至 <http://www.open-e.com/products/open-e-dss-lite/> 註冊下載 Open-E DSS Lite (dss_lite_win_setup_20080201.exe，支援至 1TB 容量之免費版)，然後將其安裝至 USB 隨身碟 (300MB 左右即可)，執行 dss_lite_win_setup_20080201.exe，便會將 iSCSI Target O.S 安裝至此 USB 隨身碟。將 USB 隨身碟插至閒置的 PC 上，BIOS 設定由 USB HDD 開機，便可擁有一台企業級的 iSCSI 儲存設備 (圖 14)。

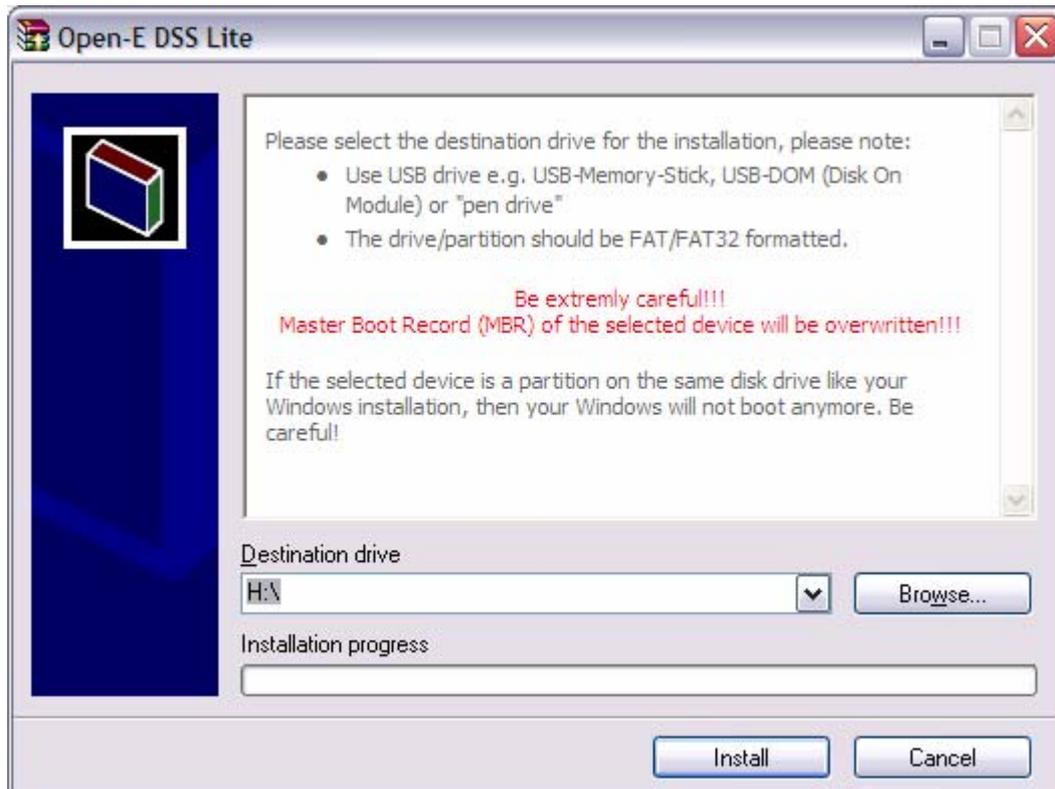


圖 14 : Open-E DSS Lite 安裝畫面

待 USB 姆指碟開機完成後，不用安裝額外軟體也不用下指令，打開瀏覽器鍵入：

[https:// iSCSI Target 主機的 IP](https://iSCSI Target 主機的 IP)

輸入密碼：「**admin**」，便可管理此台 iSCSI Targer Server。

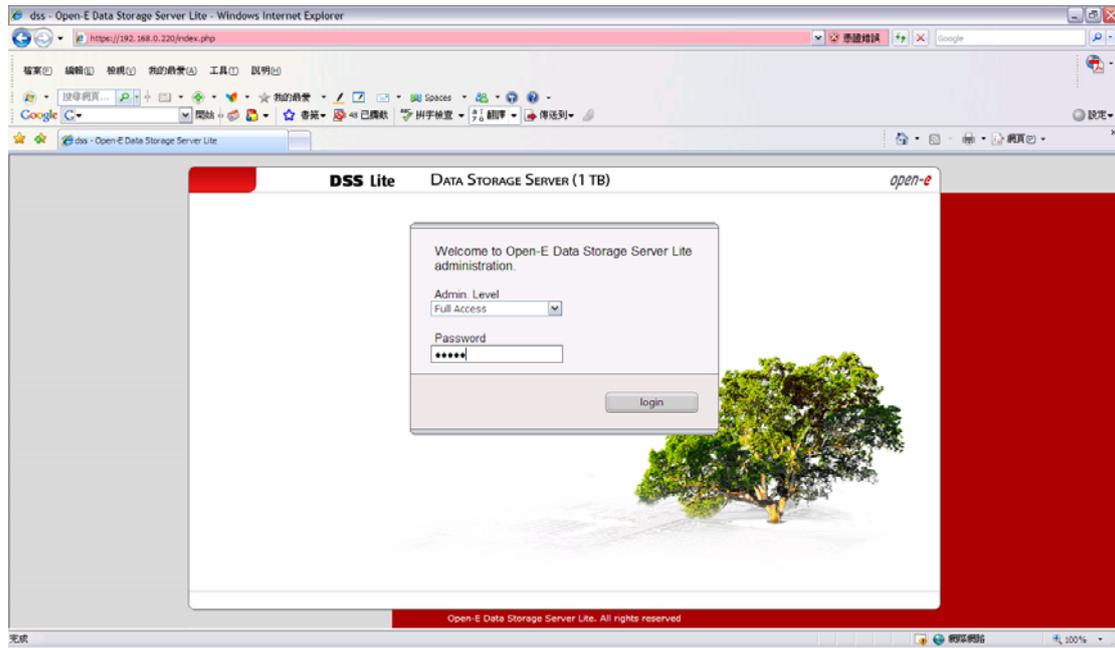


圖 15 : Open-E DSS Lite 登入畫面



3 Openfiler

Openfiler 是由 Xinit Systems 公司所開發的軟體，這是一套能讓 PC 伺服器變成 AS 或 iSCSI 設備的儲存管理作業系統。支援的檔案系統包括 Unix 的 Network File System 和微軟的 Common Internet File System，並且也支援 LDAP 和微軟的 Active Directory 等目錄服務，以及 Kerberos 5 的驗證功能。

Openfiler 安裝方式跟 Open-E DSS Lite 不同，首先到「<http://www.openfiler.com/>」下載 ISO 檔，燒成光碟，利用光碟安裝，其過程跟安裝 RHEL 並無太大差別，下列網址「<http://www.openfiler.com/learn/how-to/graphical-installation>」

有詳細的安裝過程及圖示。

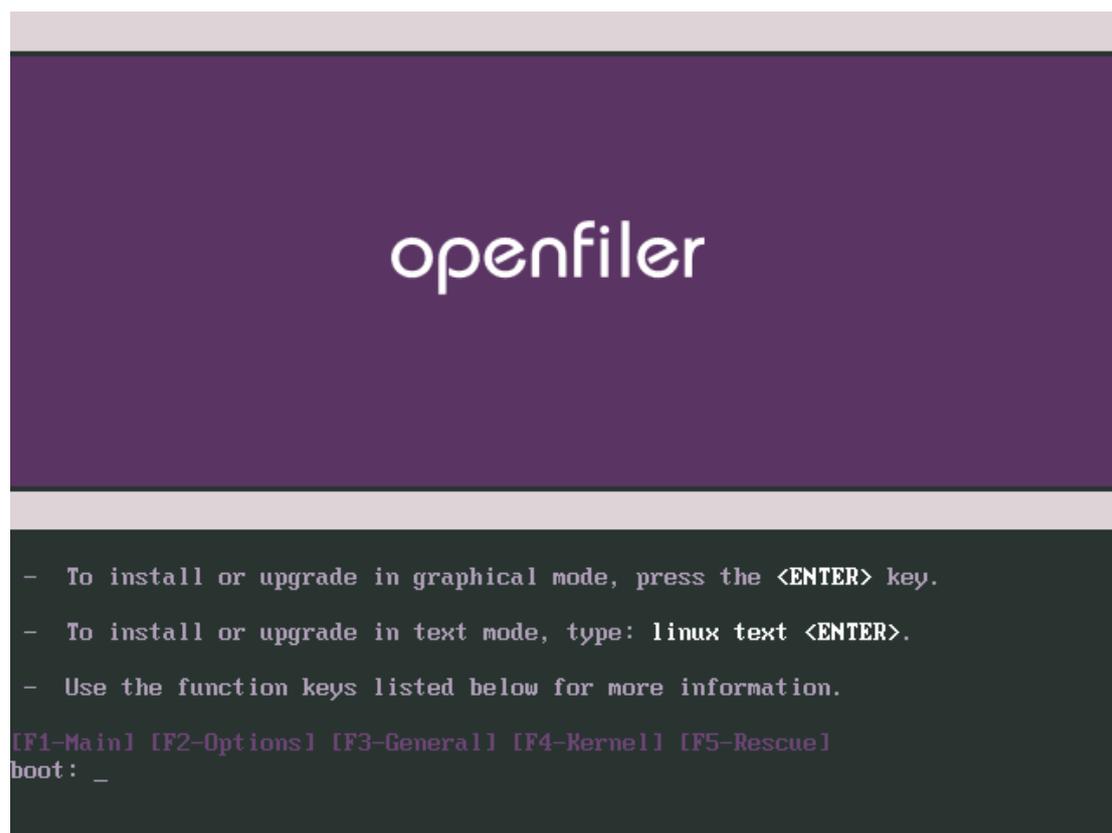


圖 16：openfiler 安裝畫面

安裝完畢後，亦是利用瀏覽器來進行管理，登入資訊如下：

Management Interface: <https://<ip of openfiler host>:446>



Administrator Username: *openfiler*

Administrator Password: *password*

至於如何管理 openfiler，可參考「<http://wwwold.openfiler.com/docs/manual/>」的手冊。

【後記】

筆者認為 Open-E DSS Lite 和 openfiler 建置 iSCSI 儲存媒體的方式，其實還蠻直覺；不過還是得了解 LVM 的觀念，兩者皆有運用 LVM 的觀念，若是看完文件還是不得其門而入，那只好期待筆者的下期文章的介紹了！^^

作者簡介

林彥明 (Alex YM Lin)：現任職於 IBM，負責 HPC 超級電腦、Linux 叢集系統建置、效能調校及技術支援等工作，近來參與 NCHC IBM Cluster 1350 (亞洲運算能力僅次日本的超級電腦) 及中山大學 p595 HPC 超級電腦專案。具有 RHCA (Red Hat 架構師)、RHCDS (Red Hat Certified Datacenter Specialist)、RHCX (Red Hat 認證主考官)、RHCE、NCLP (Novell Linux 認證專家)、LPIC、IBM AIX ... 等國際認證。