

Technical Memorandum

**Title: Linux Raid Configuration for Dell PowerVault (Fiber-Channel)  
660F and 224F with PowerEdge 2550 server running RedHat Linux  
7.0.**

Date: 24 September, 2001

Adi Zaimi

Stephen José Hanson

Psychology Department-RUMBA Labs

Rutgers University

Newark, NJ 07102

## Contents

<b>1</b>	<b>The Hardware</b>	<b>2</b>
<b>2</b>	<b>QLogic cards</b>	<b>3</b>
<b>3</b>	<b>Problems configuring the PowerVault</b>	<b>4</b>
<b>4</b>	<b>Windows NT adjustments to configure the PowerVault</b>	<b>4</b>
<b>5</b>	<b>Creating the volumes</b>	<b>5</b>
<b>6</b>	<b>Raid Level setup</b>	<b>5</b>
<b>7</b>	<b>The switch to Linux</b>	<b>6</b>
<b>8</b>	<b>Creating partitions and formating the volumes</b>	<b>6</b>
<b>9</b>	<b>Final testing and conclusions</b>	<b>7</b>

# 1 The Hardware

Starting with a PowerVault 660F and 224F units and 2 PowerEdge servers with factory installed RedHat 7.0, we initially configured 1 PowerEdge server under Linux to access the PowerVault units. We connected the two PowerVault together with the fiber connectors (lower connector of each side of the PowerVault 660F went into the upper connector of the same side in the PowerVault 224F), and we connected the fiber cord from one of the upper slots on the back of PowerVault 660F to the first PowerEdge server (to one of the two fiber QLogic cards). Figure 1 shows the configuration.

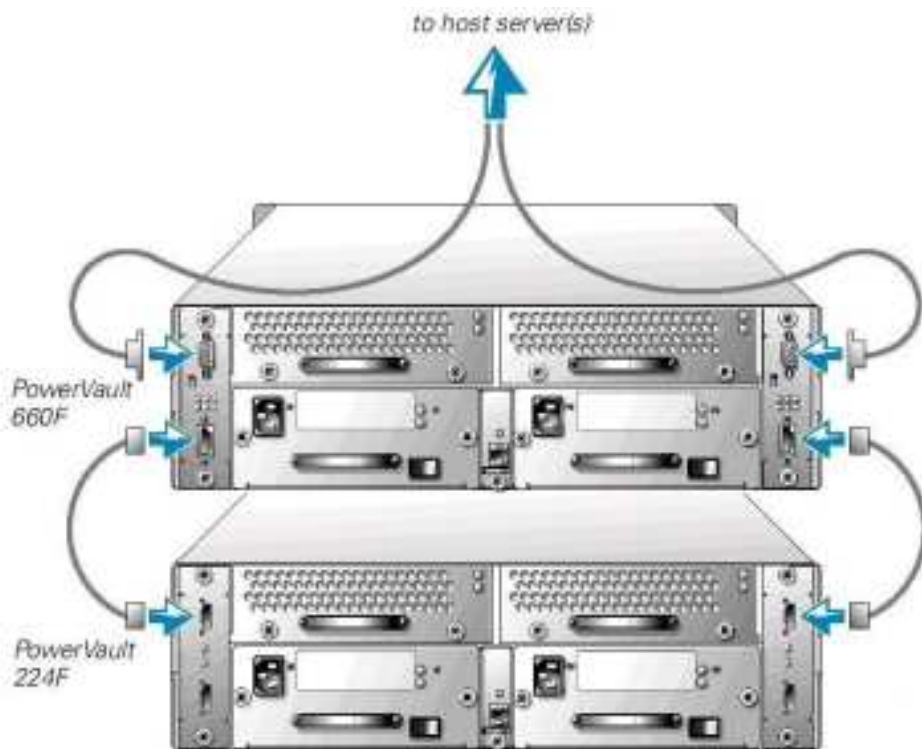


Figure 1. Connecting the PowerVault 660F and 224F

On the back of each PowerVault unit, there is a small switch; PowerVault 660F needs to be set to 0 and PowerVault 224F needs to be set to 1 for a Master/Slave configuration (Figure 2).



Figure 2. Position of the Master/Slave Configuration Switch

After powering up the first PowerEdge server connected to the PowerVault, we observed whether or not it detected the PowerVault. During the boot up the PowerEdge server did appear to detect the PowerVault 660F based on boot messages. Boot messages also indicated that the QLogic driver was loaded by the kernel, but subsequently it also indicated an error communicating to the PowerVault unit. Any attempts to access the PowerVault unit, for example: `fdisk /dev/sdb` (the scsi unit to which the PowerVault would correspond, `/dev/sda` is the local disk), would result in an error: “Unable to communicate with device....”

## 2 QLogic cards

The connection between the PowerVault and the PowerEdge server is made through a fiber channel and the QLogic 2200 adaptor (see <http://www.qlogic.com/products/qla2200.html>) which allows for communication bandwidth up to 200 MB/s (or 1Gbit) in full duplex. Each PowerEdge server came with 2 QLogic fiber cards, but apparently only one card is needed to communicate between a PowerEdge to the PowerVault. Two cards were probably installed as each PowerEdge was a Dual Processor. The other card of the first PowerEdge server (and both cards of the second PowerEdge) were useless therefore we took advantage of this fact to utilise one of these cards (from the second PowerEdge) to install it on a windows NT machine.

Dell Technical Support indicated that the PowerEdge servers would not be able to communicate to the PowerVault as we did not have the most recent QLogic driver for the PowerEdge Linux server. Once the new QLogic drivers were downloaded and installed (after downloading it from the QLogic website: [http://www.qlogic.com/bbs-html/csg\\_web/adapter\\_pages/fc\\_adapters.html](http://www.qlogic.com/bbs-html/csg_web/adapter_pages/fc_adapters.html)) the same error messages indicating lack of communication between the

Power Vaults and QLogic card/PowerEdge appeared. This implied that it was not the re-ency of the driver causing the communication link problem. Further investigation indicated that in order for the PowerEdge to talk to the PowerVault, we needed to configure the PowerVault Arrays first by unmasking the Logical Unit Numbers (LUNs) with some special microcode–PowerVault controller firmware.

### **3 Problems configuring the PowerVault**

Once we were aware that the a proper configuration procedure involved unmasking the LUNs of the PowerVault Arrays first, we investigated possible options to PowerVault configuration and the micro-controller. At present there is no direct option to perform such a configuration from a Linux server since Dell does not have such a software running under Linux. One other possibility involved accessing the PowerVault microcontroller to unmask the LUNs by using a laptop through a serial connection to the fiber slots of the PowerVault. This would provide a direct connection to the microcontroller on the PowerVault and would require using a special micrococde sequence. This microcode command sequence, however, was unavailable from Dell. The final possibility we considered was to temporarily configure the PowerVault through the Array Manager Software tool that Dell ships with the PowerVault but which runs under the Windows NT/2000 operating systems. Once configured it should be possible to access the disk as SCSI through standard Linux command line disk functions.

### **4 Windows NT adjustments to configure the PowerVault**

Concurrently we configured a generic Dell precision 410 workstation with microsoft windowsNT 4.0 SP5 in order to use the Array Manager Tools under NT. We removed one of the QLogic fiber cards from the second PowerEdge and put it into the PCI slot of the Dell workstation. The QLogic cards are 64bit PCI cards but they works just as well in the 32bit PCI slots which we had in our NT workstation (64bit PCI cards generally work in 32bit slots but their performance may be slower).

We installed the QLogic 8.0 driver in the windows machine, rebooted, and the QLogic card was detected. We installed Dell OpenManage Array Controller software, version 4.0

and connected the fiber cord from the PowerVault 660F to the Dell workstation with NT and rebooted the workstation. When it rebooted the QLogic bios detected the PowerVault 660F. When we started Array Manager we were able to see the PowerVault 660F and PowerVault 224F as unit 1 and unit 2.

## 5 Creating the volumes

It would be desirable for our application to have one single volume consisting of the two PV units, the 660F and the 224F. Each of the units has 14 disks for a total of 28 disks, making up a maximum total storage of 1000GB. But we found out that the Open Manage tool does not allow to create a group with more than 16 disks which means that we were forced to have more than one group. Given that we have 14 disks per PowerVault we found it fit to have each PowerVault (with 14 disks) as a separate group—this meant that each PowerVault will be a separate volume from the stand point of the disk organisation from the Linux prospective. So, we created two groups under the Disks entry by selecting the disks units for each PowerVault units, set them to Raid level 0+1, and initialised each group (each group separately, this takes 4-6 hours per group and it is equivalent to low level formatting, it probably could be skipped but we didn't).

## 6 Raid Level setup

Again for our application the Raid level 0+1 (striping after data mirroring) was most appropriate. Its advantages are that this level offers redundancy (two copies of the data) and it spreads the data over the disks for speed (striping). The disadvantage is that we lose half of the Arrays' space because of the redundancy, i.e. the maximum total Array space we could have was 1000GB and we now only had available a total of 500GB using Raid level 0+1. We could have used Raid level 5 which offers some redundancy and good performance (speed) and wastes only 25% of the overall maximum Array space. But the security of the data was an important issue for our application so we chose to have less space available but to better data protection from any eventual disk corruption in the PowerVault Arrays.

## 7 The switch to Linux

We disconnected the PowerVault from the Dell NT workstation, and reconnected it to the PowerEdge server running Redhat Linux 7.0 and restarted the server so that it potentially would detect the PowerVault. After reboot, in the boot messages we could see that `/dev/sdb` and `/dev/sdc` (which are the device names under which the PowerVault identifies itself—it shows up as a scsi drive) were recognised and this seemed to eliminate the error messages in the earlier transaction.

## 8 Creating partitions and formatting the volumes

After the linux server was able to see the two Arrays, we needed to create a partition and have a filesystem for each of the two volumes. These are operations that need to be performed on any new (blank) disk installed on a linux machine; refer to a Linux manual for further details or consult the manual pages of each command for instructions and internal options. To create the partitions we used the command `fdisk` which is a menu driven program for creation and manipulation of partition tables (type `man fdisk` to read the manual of the program). Any other program to manipulate the partition table for hard disk would work in a similar fashion. We typed:

```
fdisk /dev/sdb
```

and created a new primary partition (by selecting `n`—for new) and giving it the whole size of the volume (245GB). We did the same for `/dev/sdc`:

```
fdisk /dev/sdc
```

Once the partition tables were saved to disk, we needed to write a file system to each volume (to the new partition we just created for each volume) and for this we used the command `mke2fs` which is used to create a Linux second extended file system on a device, usually a disk partition (type `man mke2fs` for the manual pages). We typed:

```
mke2fs /dev/sdb1
```

```
mke2fs /dev/sdc1
```

Now the two volumes were ready to be mounted:

```
mount /dev/sdb1 /volume1
```

```
mount /dev/sdc1 /volume2
```

and the mounting operation was successful. Now the Raid Array was ready to read and write data.

## 9 Final testing and conclusions

Then we tested the reading and writing efficiency and speed for big files (larger than 100MB). The read/write operations were a success and the speed of the read and write process was astonishing: we were reading and writing files of size 1GB within 30 seconds (write=17s, read=32s) compared to the local scsi raid drive of the PowerEdge server (write=47s, read=31s) and compared to our SUN Enterprise server (write=63s, read=9s). At last, we made permanent entries in the `/etc/fstab` to have the Arrays automatically mounted at boot time:

```
/dev/sdb1 /volume1 ext2 defaults 1 2
```

```
/dev/sdc1 /volume2 ext2 defaults 1 2
```

So the PowerVault was successfully configured under Linux although the entire procedure was difficult especially given the uncertainty of whether Linux OS would be able to recognise the Raid Arrays as valid units (drives). Fortunately, Linux is a good choice for this kind of configuration and we had little further problem in getting the Arrays functional once the PowerVault Firmware was initialised and configured correctly from the Array Manager Tool which we intialized independently under the Windows NT environment.