Storage Management of VMware ESXi Using MegaCLI

A Dell Technical White Paper

July 2014

Avinash Bendigeri
Kiran Devarapalli
Krishnaprasad K

Dell Hypervisor Engineering
THIS WHITE PAPER IS FOR INFORMATIONAL PURPOSES ONLY, AND MAY CONTAIN TYPOGRAPHICAL ERRORS AND TECHNICAL INACCURACIES. THE CONTENT IS PROVIDED AS IS, WITHOUT EXPRESS OR IMPLIED WARRANTIES OF ANY KIND.

© 2014 Dell Inc. All rights reserved. Reproduction of this material in any manner whatsoever without the express written permission of Dell Inc. is strictly forbidden. For more information, contact Dell.

Dell, the DELL logo are trademarks of Dell Inc. VMware, ESXi are trademarks of VMware, Inc. in the United States and/or other jurisdictions. Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and names or their products. Dell Inc. disclaims any proprietary interest in trademarks and trade names other than its own.

July 2014
**Table of Contents**

- Introduction to MegaCLI ................................................................. 4
- Audience and Scope ........................................................................ 4
- Download MegaCLI ........................................................................ 4
- Installing MegaCLI on VMware ESXi .................................................... 4
- Monitoring VMware ESXi storage using MegaCLI ................................. 5
  - Display Storage Controller Information ........................................... 5
  - Commands to Monitor Physical Disks ............................................... 5
  - Commands to Monitor Virtual Disks ................................................. 5
  - Display the Virtual Drive Cache and Access Parameters ...................... 5
  - Display Information about Virtual Drives and Drives ......................... 5
- Storage Management using MegaCLI ....................................................... 5
  - Scanning the storage controller to initialize VD in ESXi ....................... 8
  - Marking a physical drive hot spare ................................................ 8
  - Formatting the virtual disk as a VMFS datastore .................................... 9
  - Simulate a disk failure .................................................................... 11
- Conclusion ......................................................................................... 13
- References ......................................................................................... 13
**Introduction to MegaCLI**

The MegaCLI Configuration Utility is a command line interface application to manage MegaRAID SAS RAID controllers. MegaCLI is used to configure, monitor, and maintain PERC SAS/SATA controllers and the storage-related devices connected to them.

**Audience and Scope**

This paper is intended towards IT administrators and users who have purchased, or are planning to purchase virtualization infrastructure. This document provides insight into the usage of MegaCLI, LSI storage management utility to monitor and manage the LSI storage controller that is being used. This document is divided into three major sections out of which the first section talks about downloading and installing MegaCLI package on VMware ESXi, followed by few command options for monitoring the controller, physical disks etc. The last section of this document explains about creating a virtual disk, formatting it as VMFS5 filesystem. It also talks about simulating a disk failure and making use of hot swap provision to rebuild the degraded virtual disk.

**Download MegaCLI**

MegaCLI VIB can be downloaded from LSI website under Management Software and Tools section. The MegaCLI tool is designed to work with the MegaCLI product line. The MegaCLI tool supports the following MegaRAID products.

- The 926x product line.
- The 928x product line.
- The 924x product line.
- LSI® MegaRAID SAS iMR
- LSI MegaRAID SAS 9280-8e
- LSI MegaRAID SAS 9260CV-8i
- LSI MegaRAID SAS 9266CV-8i

**DISCLAIMER:** MegaCLI is not officially supported by Dell on VMware ESXi. It can be used at your own risk. The below content is written to give a glimpse of how MegaCLI can be used in ESXi as well as for automation purposes.

**Installing MegaCLI on VMware ESXi**

To install MegaCLI on VMware ESXi, run the following command from tech support mode or from an ssh session to ESXi host.
~# esxcli software vib install -v=<path-to-VIB-package>/<VIB file>

Example:
esxcli software vib install -v=/vmfs/volumes/datastore1/MegaCLIMN/vmware-esx-MegaCli-8.07.07.vib

**Monitoring VMware ESXi storage using MegaCLI**

**Display Storage Controller Information**

The command `MegaCli –AdpAllInfo –a0/-aALL` provides information about the LSI storage controller present in the system where –a0 signified the first controller and –aALL signifies all controllers connected to the system. This command provides the controller model details, the firmware version details of the controller, configuration settings details etc.

**Commands to Monitor Physical Disks**

The command `MegaCli –PDInfo -PhysDrv[EN;S0,S1,...] -aN/-a0,1,2/-aALL` displays all the physical disks exposed from the underlying storage controller. The parameters `PDInfo` provides the details of the physical drives, `PhysDrv` accepts enclosure and slots to filter the physical drives from a pool of harddrives and -a0/-aALL signifies the controller exposed.

**Commands to Monitor Virtual Disks**

The command `MegaCli –LDInfo –Lall –a0` provides the details of all the virtual disks created on controller 0 such as RAID level, size of the virtual disk, cache policy used etc.

**Display the Virtual Drive Cache and Access Parameters**

The command `MegaCli –LDGetProp -Cache -L0 -a0` provides details about the cache policy used for the virtual disk created.

**Display Information about Virtual Drives and Drives**

The command `MegaCli –LDPDInfo –aALL` displays information about the virtual drives and physical drives for all the LSI based controllers available in the system such as the number of virtual drives, RAID level, and drive capacity.

**Storage Management using MegaCLI**

This section focuses on managing the RAID controller and the storage stack using MegaCLI. The following subsections detail about creating a RAID volume, marking a physical drive as a hot spare.

**Creation and Initialization of Virtual disk**

This section provides command line examples of creating a virtual disk on ESXi using MegaCLI. Follow the below steps to create a virtual disk on ESXi using MegaCLI.

1. Prior creating a virtual disk, we need to list the available harddrives to know the Enclosure Device ID and harddrive Device ID. The below command parameters helps to find out the
available harddrives exposed out of the controller used. The below command output is a sample output format taken from a VMware ESXi system.

```
/opt/lsi/MegaCLI#. /MegaCLI –PDList –aAll
```

Figure 1: Listing Physical drives exposed from controller

- `PDList` parameter displays all the physical drive’s IDs and `-aAll` parameter shows all the drives exposed out of the storage controller.

2. This step explains the command syntax for creating a virtual disk out of the harddrive available. From the earlier command 1, you can see the Enclosure Device ID and harddrives Device IDs. These parameters are required for creating a virtual disk. As an example we create a virtual disk with a RAID Level 5 as below.
Figure 2: Create Virtual Disk

cfgLdAdd parameter invokes the create VD operation to the controller. \(-r5\) means RAID level 5. 32:0 to 32:2 shows the harddrive device IDs which needs to be part of this virtual disk. \(-a0\) signifies that the controller device id that you use is 0. A return code of 0x00 means the command completed successfully.

3. The below screenshot explains to list the virtual disk details created above.

Figure 3: Virtual Disk Details

4. The next step is to initialize the virtual disk so that users can start utilizing the virtual disk for deploying virtual machines etc
Storage Management of VMware ESXi Using MegaCLI

Figure 4: Initialize virtual disk.

The initialization progress of the virtual disk can be monitored using –ShowProg parameter available in MegaCLI.

Figure 5: Background Initialization

Once the VD is fully initialized, the virtual disk will be made available to ESXi.

Scanning the storage controller to initialize VD in ESXi

Before we try to format the VD from ESXi, it is required to make sure that the virtual disk is detected and initialized in ESXi. The below command rescans the controller and make the VD available to ESXi.

```
/opt/lsi/MegaCLI # esxcli storage core adapter rescan --adapter vmhba1
```

Figure 6: Rescan adapter from ESXi

Where vmhba1 is the storage controller adapter name used in ESXi. The command ‘esxcli storage core adapter list’ command may be used to identify the adapter name of the storage controller used in your system. The command ‘esxcfg-scsidevs –l’ may be used to identify the VD details detected in ESXi.

Marking a physical drive hot spare

Using MegaCLI, we can mark an individual physical drive as a hot spare which can be utilized for handling virtual disk failures.

```
/opt/lsi/MegaCLI # ./MegaCli -PDHSP -Set -PhysDrv [32:3] -a0
```

Figure 7: Marking a physical Drive HotSpare

As shown in the screenshot #, -PhysDrv [32:3] is assigned as a hot spare for controller 0.
Formatting the virtual disk as a VMFS datastore

In the above section(s), we talked about setting up a VD, Initializing VD and marking a physical disk as a hot spare. This section talks about formatting the virtual disk as a VMFS datastore. Refer to [VMware KB](#) which details about the step by step procedure to create a VMFS filesystem on a virtual disk / physical disk exposed to ESXi. Before we create a VMFS filesystem, it's required to create a partition on the virtual disk. The below is an example which shows creating a VMFS5 filesystem on the virtual disk that we created in above steps.

1. Using fdisk or partedUtil commands available in ESXi, it’s possible to create partitions on a disk/VD. The below is an example which shows creating a partition on the VD using the default values. In this example, the whole VD size is utilized for the partition (number 1) created. Using fdisk/partedUtil commands, it’s possible to customize the partition sizes depending upon the requirement.

```
$ /opt/lsi/MegaCLI # fdisk /vmfs/devices/disks/naa.6d4ae5208fa52b001a29c31516f650ef -1

The fdisk command is deprecated: fdisk does not handle GPT partitions. Please use partedUtil

Disk /vmfs/devices/disks/naa.6d4ae5208fa52b001a29c31516f650ef: 598.8 Gb, 598879502336 bytes
255 heads, 63 sectors/track, 72009 cylinders
Units = cylinders of 16895 = 512 = 8225200 bytes

Command (n for help): n
Command action
   e  extended
   p  primary partition (1-4)

Partition number (1-4): 1
First cylinder (1-72009, default 1): Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-72009, default 72009): Using default value 72009
Command (n for help): u
The partition table has been altered.
Calling lobab() to re-read partition table
```

Here /vmfs/devices/disks/naa.6d4ae5208fa52b001a29c31516f650ef is the device path of the newly created VD. `esxcfg-scsidevs –l` command provides the device path corresponding to the virtual disk created.

---

**Figure 8**: Creating VMFS partition on virtual Disk

---

Storage Management of VMware ESXi Using MegaCLI
2. The second step is to change the partition ID to ‘fb’. Refer to VMware KB which talks about the various partition types that ESXi supports. By default a partition created in ESXi sets the ‘Id’ as ‘83’ (Linux EXT3). Before formatting the partition with VMFS filesystem, it is required to change its ID to ‘fb’ (VMware VMFS). The below screenshot explains to change the partition ID of the newly created partition.

```
/opt/ls/lsi/MegaCLI # fdisk /vmfs/devices/disks/naa.6d4ae52b0f0a52b001a29c31516f650ef

*** The fdisk command is deprecated: fdisk does not handle GPT partitions. Please use parted instead.
***

The number of cylinders for this disk is set to 72049.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): t
Selected partition 1
New mode (type L to list modes): Fb
Changed system type of partition 1 to Fb (VMFS)

Command (m for help): p

Disk /vmfs/devices/disks/naa.6d4ae52b0f0a52b001a29c31516f650ef: 598.8 GB, 598079582336 bytes
255 heads, 63 sectors/track, 72889 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

Device Boot Start     End     Blocks  Id  System
---  ------ ----- ------ ------ ------ ------
   1    72009  598079582336  83  Linux

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table

/opt/ls/lsi/MegaCLI #
```

Figure 9 : Changing the Device ID of partition to VMFS (fb)

3. The third step is to format the partition as VMFS 5 as below

```
/opt/ls/lsi/MegaCLI # vmkfstools -C vmfs5 -b lm -s Datastore_VD_MegaCLI /vmfs/devices/disks/naa.6d4ae52b0f0a52b001a29c31516f650ef

create fs deviceName="/vmfs/devices/disks/naa.6d4ae52b0f0a52b001a29c31516f650ef", fsShortName="vmfs5", fsName="Datastore_VD_MegaCLI" deviceFullPath=/dev/disks/naa.6d4ae52b0f0a52b001a29c31516f650ef devicefile=/dev/dsk/c0t0l0s2 VMDK file system creation is deprecated on a BIOS/MBR partition on device "naa.6d4ae52b0f0a52b001a29c31516f650ef" Checking if remote hosts are using this device as a valid file system. This may take a few seconds... Creating vmfs5 file system on "naa.6d4ae52b0f0a52b001a29c31516f650ef" with blocksize 1048576 and volume label "Datastore_VD_MegaCLI" Successfully created new volume: 5971662-d790e108-cbl2-d4ae52b00ef
```

Figure 10 : Formatting the partition as VMFS5

Where vmkfstools is the command provided by VMware to create the VMFS filesystem from tech support mode. With this step, the VMFS datastore is created and mounted under /vmfs/volumes with the name “Datastore_VD_MegaCLI” as defined as a parameter to vmkfstools.
From vCenter client, the newly added datastore is visible as shown below.

![Figure 11: Verifying the VMFS datastore created from vSphere Client](image)

**Simulate a disk failure**

Using MegaCLI, it’s possible to simulate a disk failure. Here is an example to simulate a physical drive failure by manually marking it offline and removed.

```
/opt/lsi/MegaCLI # ./MegaCli -PDOffline -PhysDrv [32:1] -a0
Adapter: 0: EnclId-32 SlotId-1 state changed to Offline.
Exit Code: 0x00
/opt/lsi/MegaCLI #
```

**Figure 12: Marking a Disk Offline**

```
/opt/lsi/MegaCLI # ./MegaCli -PDMarkMissing -PhysDrv [32:1] -a0
EnclId-32 SlotId-1 is marked Missing.
Exit Code: 0x00
/opt/lsi/MegaCLI #
```

**Figure 13: Marking a Disk Missing**

```
/opt/lsi/MegaCLI # ./MegaCli -PDPrpRmv -PhysDrv [32:1] -a0
Prepare for removal Success
Exit Code: 0x00
/opt/lsi/MegaCLI #
```

**Figure 14: Disk Removal**
This makes the virtual disk degraded. Since it’s a RAID5, the virtual machines running on the datastore (if any) will not be impacted as it can handle a single disk failure. As expected, the virtual disk failure imposes a degradation status for the Virtual disk as below.

```
/opt/lsi/MegaCLI # ./MegaCli -LDInfo -L0 -a0
```

![Figure 15: Virtual Disk Details](image)

Now, the virtual disk can be rebuilt using the hot spare that we have configured earlier.

```
/opt/lsi/MegaCLI # ./MegaCli -LDRecon -Start -R5 -Add -PhysDrv[32:3] -L0 -a0
```

Start Reconstruction of Virtual Drive Success.

```
Exit Code: 0x00
/opt/lsi/MegaCLI #
```

![Figure 16: Virtual Disk Rebuild using HotSpare Disk](image)
As shown above, the physical drive 32:3 is the drive which we had marked for hot spare. Using the –LDRecon parameter, MegaCLI starts rebuilding the virtual disk using the hot spare disk. The status of the Rebuilding process is monitored as below.

```
/opt/lsi/MegaCLI ./MegaCli -LDRecon -ShowProg -L0 -a0
Reconstruction on VD #0 (target id #0) Completed 3% in 8 Minutes.
Exit Code: 0x00
/opt/lsi/MegaCLI # 
```

**Figure 17 : Virtual Disk Rebuild Progress**

NOTE: The virtual machine(s) configured if any will not be affected during the rebuilding process

**Conclusion**

This paper explains usage of MegaCLI on ESXi. It also explains the MegaCLI parameters with a use case of creating VMFS5 filesystem, simulating a disk failure. This paper would be useful for end users who want to automate few storage related tasks on ESXi.

**References**

- [http://kb.vmware.com/kb/2004166](http://kb.vmware.com/kb/2004166)