
Using SYMCLI to Perform Device Masking

Abstract

This white paper describes device masking functionality that allows you to manage host access to Symmetrix devices when a host and a Symmetrix array communicate via Fibre Channel or iSCSI interfaces.

Published 1/25/2005

Copyright © 2005 EMC Corporation. All rights reserved.

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED “AS IS”. EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC², EMC, Symmetrix, Celerra, CLARiiON, CLARAlert, Documentum, HighRoad, Legato, Navisphere, PowerPath, ResourcePak, SnapView/IP, SRDF, TimeFinder, VisualSAN, and where information lives are registered trademarks and EMC Automated Networked Storage, EMC ControlCenter, EMC Developers Program, EMC OnCourse, EMC Proven, EMC Snap, Access Logix, AutoAdvice, Automated Resource Manager, AutoSwap, AVALONidm, C-Clip, Celerra Replicator, Centera, CentraStar, CLARevent, Connectrix, CopyCross, CopyPoint, DatabaseXtender, Direct Matrix, Direct Matrix Architecture, EDM, E-Lab, Enginuity, FarPoint, FLARE, GeoSpan, InfoMover, MirrorView, NetWin, OnAlert, OpenScale, Powerlink, PowerVolume, RepliCare, SafeLine, SAN Architect, SAN Copy, SAN Manager, SDMS, SnapSure, SnapView, StorageScope, SupportMate, SymmAPI, SymmEnabler, Symmetrix DMX, Universal Data Tone, and VisualSRM are trademarks of EMC Corporation. All other trademarks used herein are the property of their respective owners.

Part Number 300-000-650 REV F

Table of Contents

Introduction	5
Purpose and Scope	5
Related Documentation	5
Practical Uses	5
Device Masking Concepts.....	6
Adding HBA Access to Symmetrix Devices	7
Removing Devices	7
Viewing the VCMDB	8
Backing Up, Restoring, and Initializing the VCMDB	9
Initializing and Formatting the VCMDB Database	9
Preventing Unauthorized Modification of the VCMDB.....	9
HBA Identifiers (WWN, AWWN, and iSCSI Name).....	10
Swapping a New HBA for a Failed HBA.....	10
Adding Security Using Fibre Channel ID Lockdown	11
Turning on LUN Visibility to Discover Noncontiguous Devices	12
Offsetting LUN Addresses	12
Configuring Heterogeneous Hosts that Share an FA Port	13
Using Different Types of Device Masking Databases	13
Setting iSCSI Authentication with Enginuity Version 56xx.....	14
Setting iSCSI Authentication with Enginuity Versions Greater Than 5671..	15
Setting One-Way CHAP Authentication.....	15
Setting Two-Way CHAP Authentication.....	16
Using a RADIUS Server to Store Authentication Information.....	16
Backing Up, Restoring, Initializing Authentication Information	17
Initializing an Authentication Database	17
Displaying Authentication Information	18

Example 1: Adding Masked Devices for HBA Access	19
Example 2: Using Fibre Channel ID Lockdown	22
Example 3: New Options When Displaying the VCMDB.....	24

Introduction

The Symmetrix® Device Masking component of EMC® Solutions Enabler provides commands that allow you to manage a device masking environment in which a host and a Symmetrix array communicate via Fibre Channel or iSCSI interfaces. With Fibre Channel, each host connects to the Fibre Channel hub or switch through one or more Host Bus Adapter (HBA) ports. A Symmetrix array connects to the Fibre Channel hub or switch through one or more FA director ports, each of which provides access to a given set of Symmetrix devices that are mapped to it. Device masking commands allow you to:

- Add or remove devices from a Fibre Channel or iSCSI HBA entry in the Symmetrix device masking database to specify whether or not an HBA has access to a particular device.
- Display device masking objects and their relationships. Typical objects include hosts, HBAs, Symmetrix devices, and FA ports.
- Swap the definition of one HBA for a new HBA while retaining the associated device set defined for the original HBA.
- Discover all HBAs on a host and automatically assign an AWWN (ASCII World Wide Name or alias for the WWN) to each HBA's unique WWN (World Wide Name).
- Customize attributes of the Fibre-Channel-to-host interface for compatibility with your host platform (for example, Fibre Channel ID lockdown, device LUN visibility, adjustment for noncontiguous LUNs, and heterogeneous host configuration).
- Back up, restore, or initialize the Symmetrix-based device masking database.
- Convert one type of device masking database to another type.
- Set iSCSI authentication.

Purpose and Scope

This paper provides an introduction to the device masking functionality included in EMC Solutions Enabler up through version 6.0 and Enginuity version 5x71.

Related Documentation

The following EMC manuals and white papers provide information related to this paper:

- *EMC Solutions Enabler Symmetrix Device Masking CLI Product Guide*
- *Using the SYMCLI Configuration Manager* (P/N 300-000-475)

Practical Uses

Device masking allows you to control your host HBA access to a Symmetrix device by associating one or more devices with an HBA-to-FA connection that you define in the Symmetrix-based device masking database. Through centralized monitoring and access records, this database resolves any conflicts that might arise from multiple hosts having visibility to the same devices.

Device masking also allows you to configure heterogeneous hosts to share access to the same FA port, which is useful in an environment with different host types. However, you can also use Fibre Channel ID lockdown security to protect an HBA from WWN spoofing, where an unauthorized host can change its HBA's WWN to match one in the device masking database.

Device Masking Concepts

When several hosts connect to a single Symmetrix FA port, an access control conflict occurs because all hosts have the potential to discover and use the same storage devices. However, you can make an entry into the Symmetrix array's device masking database (VCMDB) to control host access to devices. This VCMDB entry specifies a host's HBA identity (using an HBA port WWN¹), its associated FA port, and a range of devices mapped to the FA port that should be visible only to the corresponding HBA. Once you make this VCMDB entry and activate the configuration, the Symmetrix makes visible to a host those devices that the VCMDB indicates are available to that host's initiator WWN through that FA port.

Figure 1 illustrates a network where two hosts have the potential to access the same Symmetrix devices because the two hosts share the same Symmetrix director port (FA 1). However, by creating logical connections that connect each host with the appropriate storage devices, you grant Host 1 access to devices 0001 and 0002, and Host 2 access to 0003. You use the `symmask add devs` command to make two entries in the Symmetrix array's VCMDB, one specifying HBA 1 access to devices 0001 and 0002, and another specifying HBA 2 access to 0003.

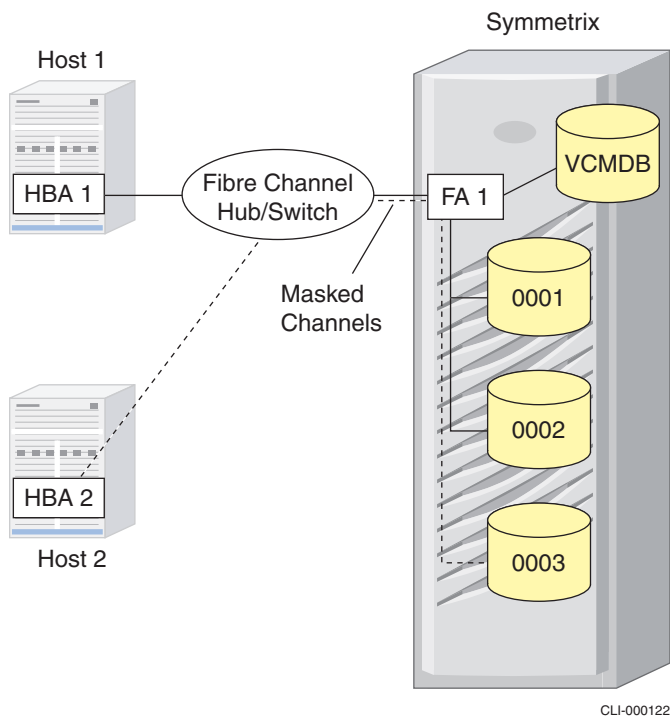


Figure 1. Fibre Channel Topology with Two Hosts Connected to the Same FA Port

At host login time, the WWN of each HBA is passed to a Symmetrix FA director port. The Symmetrix records the connection and stores the WWN in a login history table. The Symmetrix then compares the host WWN to the WWNs defined in the VCMDB. If it finds a match, the Symmetrix makes visible those devices that the VCMDB indicates are available to that WWN through that FA port.

You can create a configuration that provides continued availability if a hub or its connections fail. For example, a second HBA on Host 1 could connect to a different FA port through a different Fibre Channel hub. You could then define a logical connection for this second HBA to access the same devices as HBA 1.

¹ Fibre HBAs have a host WWN and a port WWN for each port on the HBA. Device masking always refers to a *port* WWN.

Adding HBA Access to Symmetrix Devices

To configure device masking in the VCMDB, log on to the control station as Administrator. The following steps outline how to add device access for an HBA-to-FA connection:

1. Discover local HBAs on a host that have a channel to a Symmetrix array and generate an AWWN for any HBA that does not have an AWWN assigned yet, updating the VCMDB with the new information.

```
symmask discover hba
```

2. List the HBAs on a host and the Symmetrix FA director port to which each HBA is connected. Note the WWN of the HBA that you will be configuring and which FA port connects to that HBA.

```
symmask list hba
```

3. List devices mapped to the FA director that you will be configuring (for example, director 16A).

```
symcfg list -FA 16A -addr
```

4. Make an entry for the HBA-to-FA connection in the VCMDB², specifying devices that the HBA can access. For example, add a range of devices (0030 through 0034) to the VCMDB on the Symmetrix array (`-sid 814`), specifying the HBA's WWN and the FA director/port that the HBA connects to.

```
symmask -sid 814 -wnn 20000000c920b484 add devs 0030:0034 -dir 16A -p 0
```

5. When you finish making HBA entries in the VCMDB, proceed with the following steps. Back up the revised VCMDB to a file (for example, a file called `MyDevMaskBackup`).

```
symmask -sid 814 backup -file MyDevMaskBackup
```

6. Refresh the WWN-related profile tables in the Symmetrix cache with the latest VCMDB data. The following command updates the cache on all FA directors with the contents of the VCMDB.

```
symmask refresh
```

7. Reboot the host whose HBA entry you added to the VCMDB.

8. When you reboot a host, you need to scan the Symmetrix devices and refresh the SYMAPI database.

```
symcfg discover
```

Removing Devices

You can remove some devices associated with an HBA entry, or you can remove the entire set of devices associated with an HBA entry. Removing some devices (but not all) requires syntax similar to that for adding devices and the same refresh, backup, and discover steps after completing the remove operation.

To remove devices 0031 and 0033 from the 0030-to-0034 range of devices that was added previously:

```
symmask -sid 814 -wnn 20000000c920b484 remove devs 0031,0033 -dir 16A -p 0
```

To remove the remaining devices in the 0030-to-0034 device range, you can specify individual devices or the range with an option (`-force`) that allows you to remove a noncontiguous range. For example:

```
symmask -sid 814 -wnn 20000000c920b484 remove devs 0030:0034 \
-dir 16A -p 0 -force
```

² If the VCMDB does not exist yet (that is, this is its initial creation), you need to first initialize the VCMDB device using the `symmask init -file` command. Note, however, that using this command on an *existing* VCMDB results in clearing the entire database. (Refer to the section "Initializing and Formatting the VCMDB Device".)

To remove the entire set of devices that an HBA can access, use `symmask delete` and specify the WWN of the HBA. The delete action removes the HBA entry completely, including any attributes set previously. For example:

```
symmask -sid 814 delete -wwn 20000000c920b484
```

Viewing the VCMDB

You can display the entire contents of the VCMDB or use options to restrict the display to your area of interest. For example, to view the entire VCMDB of Symmetrix 814:

```
symmaskdb -sid 814 list database
```

To display just those devices that an HBA can access, specify the HBA's WWN. For example:

```
symmaskdb list devs -wwn 10000000c9274156
```

```
Symmetrix ID          : 000184500313
```

```
Originator Port wwn : 10000000c9274156
```

```
User-generated Name : apil40/i@1f,4000,@2
```

Sym Dev Name	Dir:P	Physical Device Name	VBUS	TID	LUN	Attr	Cap (MB)
0009	14B:0	/dev/rdisk/clt0d1s2	0	0	1		1031
000A	14B:0	/dev/rdisk/clt0d2s2	0	0	2		1031
000B	14B:0	/dev/rdisk/clt0d3s2	0	0	3		1031
000C	14B:0	/dev/rdisk/clt0d4s2	0	0	4		1031
000D	14B:0	/dev/rdisk/clt0d5s2	0	0	5		1031
000E	14B:0	/dev/rdisk/clt0d6s2	0	0	6		1031
000F	14B:0	/dev/rdisk/clt0d7s2	0	0	7		1031
0050	14B:0	/dev/rdisk/clt0d16s2	0	0	10	(M)	8250
0051	14B:0	Not Visible	-	-	-	(m)	-
0052	14B:0	Not Visible	0	0	10		1031
0053	14B:0	Not Visible	-	-	-	(m)	-
0054	14B:0	Not Visible	-	-	-	(m)	-
0055	14B:0	Not Visible	-	-	-	(m)	-
0056	14B:0	Not Visible	-	-	-	(m)	-
0057	14B:0	Not Visible	-	-	-	(m)	-

To display your backup file of the VCMDB, use the `-file` option with a file name. For example:

```
symmaskdb -file MyDevMaskBackup list devs -wwn 20000000c920b484
```

You can use `symmaskdb list` commands to see which HBAs were assigned to a set of devices or to determine the capacity of all devices assigned to a specific host. Refer to Example 3 to see the outputs for these commands.

```
symmaskdb -sid 814 list assignment -dev 0030:0034
symmaskdb -sid 814 -host api213 list capacity
```

The host name that you specify (api213, for example) must coincide with the host part of the two-part AWWN or alias name. For information on the two-part format of an AWWN or alias name, refer to the section "HBA Identifiers (WWN, AWWN, and iSCSI Name)".

Backing Up, Restoring, and Initializing the VCMDB

Backing up the VCMDB on a regular basis ensures that you can restore a good version of the database in case incorrect changes or other abnormalities occur in the current VCMDB. Or you may wish to test temporary changes to the VCMDB and restore the backup (pre-test) version after testing is complete.

The following command creates a backup version of the VCMDB in a file called MyDevMaskBackup (each time you back up the VCMDB you must choose a new file name).

```
symmaskdb -sid 814 backup -file MyDevMaskBackup
```

The following command restores the VCMDB using the backup version in the file MyDevMaskBackup.

```
symmaskdb -sid 814 restore -file MyDevMaskBackup
```

Initializing and Formatting the VCMDB Database

In the *initial setup* of any device masking environment, you must initialize and format the database. In the process of formatting the VCMDB, the initialization of the database clears it of any current data. In most cases, you do not want to clear the data of an existing VCMDB. If you are unsure whether a VCMDB currently exists, issue the command that displays an existing VCMDB. For example, to view the VCMDB if it exists on Symmetrix 814:

```
symmaskdb -sid 814 list database
```

To initialize and clear the VCMDB database, you must specify a backup file name to safeguard against clearing data in the database that should not be lost. For example, the following command creates a file called MyInitBackup and attempts to write any current data to it prior to initializing and formatting the VCMDB on Symmetrix 814 (the `-vcmdb_type` option initializes it to a type 3 database):

```
symmaskdb -sid 814 init -file MyInitBackup -vcmdb_type 3
```

If you do not include the `-vcmdb_type` option, the default database type depends on the size of the VCMDB device: type 3 if the device is equal to or greater than 24 cylinders, but less than 48; a type 4 if equal to or greater than 48 cylinders, but less than 96; a type 5 (beginning with Enginuity version 5671) if equal to or greater than 96 cylinders. If the Symmetrix array is running an Enginuity level greater than version 5671, the default is a type 6 database.

Preventing Unauthorized Modification of the VCMDB

By default, the VCMDB grants access to all HBAs that log in to the FA director port where the database is configured. Without preventive measures, any host with access privileges can modify the VCMDB if it has the EMC Solutions Enabler Device Masking component or ESN Manager installed.

One way to prevent unauthorized host access is by using the Configuration Manager to set the `VCMDB_restricted_access` Symmetrix parameter. By setting this parameter value to `ENABLE`, you restrict access to the VCMDB to hosts having a VCMDB entry that includes the VCMDB device (the device on which the device masking database resides). You deny VCMDB access to all hosts except those whose HBAs have added the VCMDB device through the `symmask add devs` command. (You can display the VCMDB device on a Symmetrix array using the `sympd list -vcm` command.)

Prior to enabling this parameter, you should ensure that at least one host HBA has a valid VCMDB entry that *includes the VCMDB device*. (It is recommended that you have two HBA entries that include this device, in case of an HBA failure.) Without this VCMDB entry, no hosts can access the VCMDB database. To gain access to the VCMDB database again, you would need to reset this parameter to `DISABLE`.

HBA Identifiers (WWN, AWWN, and iSCSI Name)

An HBA can be specified in `symmask` commands by its unique WWN³, by an AWWN alias associated with the WWN, or by an iSCSI name (or its alias). These names appear in the Symmetrix login history table and in the VCMDB.

When you issue the `symmask discover hba` command to update the login history table, the Symmetrix API (SYMAPI) checks the VCMDB to determine whether an AWWN exists for each WWN record. If not, SYMAPI creates an AWWN that consists of two parts (the name of the host and the name of the HBA) and writes it to the login history table. However, you can rename the AWWN to a shorter name, if you prefer.

The following command allows you to examine AWWNs in the login history table of Symmetrix 814:

```
symmask -sid 814 list logins
```

To assign an AWWN to fit your naming requirements, you can use the `symmask rename` command at any time — even before SYMAPI generates an AWWN. For example, the following command assigns the two-part name Solaris3A/b4 as the AWWN for WWN 20000000c920b484:

```
symmask -sid 814 -wwn 20000000c920b484 rename Solaris3A/b4
```

If later you decide you want to replace *all* user-defined AWWNs (like Solaris3A/b4) with system-generated AWWNs, the following command overwrites *all* existing AWWN entries in the VCMDB.

```
symmask discover hba -rename
```

You can identify the HBA in `symmask` commands through an iSCSI name. The iSCSI is a SCSI-over-IP protocol that tries to take advantage of the IP network for SCSI traffic rather than requiring a new Fibre network. An iSCSI name is used like an IP address or a WWN and is displayed using the `symmask list logins` or the `symmask list hba` commands. A unique iSCSI name is determined by the hardware that logs into a Symmetrix array and, like the AWWN, a two-part alias name is generated from the `symmask discover hba` command.

Swapping a New HBA for a Failed HBA

If an HBA fails or needs replacement, you can use the following steps to swap in a new HBA without having to redefine the old HBA set of masked devices for the new HBA.

1. Issue the `symmask list logins` command to determine the WWN of the failed HBA.
2. Swap the HBA boards.
3. Issue the `symmask list hba` command to determine the new WWN.
4. Substitute the new WWN for all occurrences of the old WWN in the VCMDB. For example, to replace old WWN 20000000c920b484 with new WWN 20000000c920b393:

```
symmask -sid 814 -wwn 20000000c920b484 replace 20000000c920b393
```

5. Issue the `symmask discover` command to establish the new AWWN in the login history table or, if you prefer, issue the `symmask rename` command to assign an AWWN to the new HBA in both the login history table and the VCMDB.
6. Issue `symmask refresh` to update the VCM information in Symmetrix director cache.

³ Fibre HBAs have a host WWN and a port WWN for each port on the HBA. Most HBAs have only one port, but some have two. In rare cases, an HBA has four ports. HBA identification in device masking always refers to a *port* WWN.

Adding Security Using Fibre Channel ID Lockdown

Fibre Channel ID lockdown is a security feature that limits host access to devices when you add the Fibre Channel ID (FCID) value to an HBA entry in the VCMDB. This feature protects an HBA from WWN spoofing, where an unauthorized host can change its HBA's WWN to match one in the VCMDB. WWN spoofing can be a threat to your networked systems in a shared director port configuration (where HBAs from different hosts share the same FA port). However, once you lock down a Fibre Channel ID, no user with a spoofed WWN can log in. If a user with a spoofed WWN is already logged in, that user loses access through the protected HBA.

When you add the FCID value to an HBA entry in the VCMDB, the valid physical path for the HBA through the SAN is locked down. Only an HBA with a Fibre Channel ID that matches the one entered for it in the VCMDB is able to log in to the FA port. However, if an *incorrect* Fibre Channel ID is added to the VCMDB, the HBA loses access and host utilities may hang on the server with the locked-out WWN. For this reason, it is recommended that at least two HBAs (on different FA ports) be available on the administrator host. That way, if one HBA becomes locked out, the host has access through the other HBA and can correct an erroneous entry in the VCMDB.

The following steps outline a procedure for locking down the Fibre Channel ID:

1. Issue the `symmask list hba` command to determine the WWN and device path of the HBA that you want to protect.
2. Issue the `symmask list logins` command to determine the FCID value. Specify the HBA device path from step 1 (for example, `/dev/vx/rdmp/c4t0d0s2`):

```
symmask list logins -pdev /dev/vx/rdmp/c4t0d0s2
```

3. Set lockdown for the Fibre Channel ID 220413. For example:

```
symmask set lockdown on 220413 -dir 16A -p 0
```

4. Issue the `symmaskdb list database` command to verify that the Fibre Channel ID is locked down.
5. If you have completed your entire set of changes for this session, refresh the VCMDB in the Symmetrix cache.

```
symmask refresh
```

Locking down a Fibre Channel ID has no affect on `symmask delete` (the HBA entry is completely cleared) or on `symmask replace` when the cable is moved from one HBA to another HBA and not moved at the switch. However, if the cable is moved from one port on the switch to another port, the FCID value changes. In this case, do not unlock the Fibre Channel ID while swapping HBAs but, instead, leave at least one HBA device path open from the administrator host and reset the Fibre Channel ID in the VCMDB after the swap.

Resetting the Fibre Channel ID requires that you obtain the FCID value from the switch because you no longer have an HBA device path that `symmask list logins` can use to find the FCID value. For information on how to find the FCID value from Connectrix™ or Brocade switches, refer to *EMC Solutions Enabler Symmetrix Device Masking CLI Product Guide*.

Turning on LUN Visibility to Discover Noncontiguous Devices

Using the device LUN visibility feature allows the host driver to discover devices with noncontiguous LUN addresses. Certain host platforms (notably HP-UX and Linux) require that LUN 000 be present when it scans the interface for devices. Also, these host platforms and others cannot detect noncontiguous devices beyond the initial LUN sequence (that is, they cannot adjust for a break in the sequence of LUNs on the target). This can be a problem if you need to mask devices that are not visible to certain host platforms.

To allow your host to detect all these devices, turn on the device LUN visibility feature so that all devices attached to a specified FA director and port are made available to the HBA. For example, to turn on visibility for FA director 16A, port 0, when working with a host HBA whose WWN is 20000000c920b484:

```
symmask -sid 814 set visibility on -dir 16A -p 0 -wn 20000000c920b484
```

Allowing the host to scan the other devices past a break in the LUN sequence does not change *how* the host accesses them. For example, if an HP-UX host has LUNs 0000, 0002, 0003, 0004 assigned to it, that is exactly what the host will see. If you need the host to access this same sequence without the break between 0000 and 0002, you can adjust host visibility by offsetting LUN addresses as described in the following section.

Offsetting LUN Addresses

LUN offset is an enhanced visibility feature that allows any host type to adjust host visibility by offsetting (renumbering) LUN addresses. This is useful for host types that need to see LUN 0000 or transform a noncontiguous LUN sequence to a contiguous sequence. In a case where two hosts access the same Symmetrix director port and need to see a LUN 0000 but not the same device, you can use LUN offset so that one host sees the devices mapped from LUN “x” as starting from LUN 0000, and the other host sees devices from LUN “y” as starting from LUN 0000.

To account for noncontiguous device LUN addresses, specify a LUN base and offset as *hexadecimal* values to adjust for the break in the LUN sequence. The *base* hex value represents the first LUN in a renumbered LUN sequence. The *offset* hex value added to the base value determines where to begin renumbering. For example, if a host needs to detect LUN 0000 but you want your host to detect only LUNs 0005 through 0008, you can specify a LUN base address of 0000 and an offset of 0005. The following command rennumbers LUNs 0005 through 0008 as LUNs 0000 through 0003:

```
symmask -sid 814 set lunoffset on 0005 0000 -dir 16A -p 0 -wn 20000000c920b484
```

On the other hand, if your masked devices for an HBA-to-FA connection had LUN addresses 0000–0003 and 0007–0009, you would need to specify a LUN base address of 0004 and an offset of 0003 (to renumber LUNs 0007–0009 as LUNs 0004–0006). You can record only one gap per HBA-to-FA connection.

If you have multiple hosts that cannot discover devices with noncontiguous LUN addresses, you need to issue the `symmask set lunoffset` command for each host.

Configuring Heterogeneous Hosts that Share an FA Port

You can set a configuration flag that allows different host types to share a single director FA port, even though different hosts may require different port settings for their interface protocol. Setting this flag sets an entry in the VCMDB to hold connection protocol information on the specified host type that may differ from the current Fibre protocol setting on the director. You can use this feature in conjunction with the LUN offset/skip feature to allow different host types to share access to the same FA port.

The following command enables heterogeneous on FA director 16A, port 0, for a Solaris type host:

```
symmask -sid 814 -dir 16A -p 0 set heterogeneous on SOLARIS \
        -wwn 20000000c920b484
```

The following command disables heterogeneous for the Solaris type host:

```
symmask -sid 814 -dir 16A -p 0 set heterogeneous off SOLARIS \
        -wwn 20000000c920b484
```

The following command enables heterogeneous on FA director 16A, port 0, for an HP-UX type host:

```
symmask -sid 814 -dir 16A -p 0 set heterogeneous on HP_UX \
        -wwn 20000000c920b484
```

For a complete list of host configuration flag values, refer to the table “Host Platforms and Interface Configuration Flags” in the *EMC Solutions Enabler Symmetrix Device Masking CLI Product Guide*.

Using Different Types of Device Masking Databases

The type of device masking database controls the number of devices-per-record and the number of records that you can mask. Beginning with Solutions Enabler version 5.3 and Enginuity version 5670, a Type 4 device masking database was supported along with the Type 3 database of earlier software versions. Beginning with Solutions Enabler version 6.0 and Enginuity version 5671, a Type 5 device masking database is now supported also. With Enginuity levels greater than version 5671, a Type 6 database is the only valid type.

- Type 3 supports up to 8K devices per record and 32 fibre/32 iSCSI initiator records per port (the VCMDB device must be 24 cylinders or larger)
- Type 4 supports up to 8K devices per record and 64 fibre/128 iSCSI initiator records per port (the VCMDB device must be 48 cylinders or larger)
- Type 5 supports up to 16K devices per record and 64 fibre/128 iSCSI initiator records per port (the VCMDB device must be 96 cylinders or larger)
- Type 6 supports up to 64K devices per record and 256 fibre/512 iSCSI initiator records per port. This type of database resides in the Symmetrix File System (SFS) and is valid only with Enginuity levels greater than version 5671.

When initializing the VCMDB database where no database currently exists, SYMCLI *defaults* to creating a database according the size of the VCMDB device being initialized (Type 5 for a 96-cylinder device, Type 4 for a 48-cylinder device, Type 3 for a 24-cylinder device). This default behavior is also true for the VCMDB device that currently holds a database (for example, a Type 4 database on a 96-cylinder device will be initialized as a Type 5 database). A Type 4 or 5 database initializes with direct I/O writes blocked to protect against outside sources corrupting the database. You can block direct I/O writes to a Type 3 database using `set no_direct_io`.

You can also specify *explicitly* what database type to create in the initialization procedure. If you have a larger VCMDb device size (96 cylinders, for example), you might want to specify the smaller size to stay backward compatible with an earlier Solutions Enabler version of the software.

```
symmaskdb -sid 814 init -file MyInitBackup -vcmdb_type 4
```

You can convert a Type 3 database to Type 4 or Type 5, and a Type 4 to a Type 5, if the size of the VCMDb device on the specified Symmetrix array is large enough for the database type. For example, converting the VCM database on Symmetrix 814 to a Type 5 database.

```
symmaskdb -sid 814 convert -vcmdb_type 5 -file MyConvertBackup
```

If you convert from a lower type database to a higher type, other connected hosts running a Solutions Enabler version compatible only with the lower type will not be able to access the database. For example, if one host running Solutions Enabler version 6.0 converts an existing VCMDb to type 5, another connected host running Solutions Enabler version 5.4 will no longer have access to the database until this host is upgraded to version 6.0 or higher.

If you request that a VCMDb database be restored, by default it is restored in the format (Type 3, 4, or 5) that exists in the backup file from which the database is restored. However, you can specify that it be restored from a lower type to a higher type (from a Type 4 to a Type 5, for example).

```
symmaskdb -sid 814 restore -file MyDevMaskBackup -vcmdb_type 5
```

Setting iSCSI Authentication with Enginuity Version 56xx

The iSCSI authentication can be carried out with various methods and is usually negotiated during the HBA login phase. The CHAP method (Challenge Handshake Authentication Protocol) is supported here. With this method, you manage a credential name that is similar to a username, and a CHAP secret that is similar to a password.

Setting iSCSI authentication with the `symmask set authentication` command requires a DMX model Symmetrix using Enginuity version 5670 (or higher), a FigE board that manages the front-end connections in the Symmetrix, a GigE-configured port, host systems that provide driver support for iSCSI, and a VCMDb database.

Note: To set iSCSI authentication for a Symmetrix array using Enginuity levels greater than 5671, you need to use `symconnect` commands instead of the `symmask set authentication` command, which is valid only for Enginuity version 56xx (5670 and higher). For information about using `symconnect` commands, refer to the section “Setting iSCSI Authentication with Enginuity Levels Greater Than Version 5671.”

Before an iSCSI host can log in and see any devices, the iSCSI name of the host must have a valid VCMDb database entry specifying the director/port from which it is connecting.

Setting iSCSI authentication requires that you supply the credential name and the CHAP secret. The following command sets the authentication for the iSCSI initiator (iqn.2002-06.com.microsoft.host210) using the CHAP authentication type. The `-credential` option and `-secret` option specify the necessary information.

```
symmask -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210 \
  set authentication -type CHAP -credential MyCredentials -secret MySecret
```

In unidirectional authentication such as this, the Symmetrix authenticator challenges the host peer and expects to receive the username/password in response.

The `symmask show` command allows you to display the authentication, although not the CHAP secret (which is never displayed). For example:

```
symmask -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210 \
show authentication
```

You can also display authentication data using the `symmaskdb list database -v` command.

When you set authentication using the `symmask` command, authentication is automatically enabled. If you decide to disable authentication, the authentication values are retained for that time when you enable authentication again for the iSCSI. The following command disables authentication.

```
symmask -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210 \
disable authentication
```

Similarly, you can substitute `enable` to enable authentication again.

When you restore a database, authentication data is restored to the Symmetrix array from the backup file. Keep in mind that the same authentication data needs to be set also in the host's iSCSI driver software if authentication was changed there after the backup file was generated.

If you want to avoid restoring obsolete authentication data, you can use the `-skip_authentication` option. For example:

```
symmaskdb -sid 814 restore -file MyDevMaskBackup -skip_authentication
```

Setting iSCSI Authentication with Enginuity Versions Greater Than 5671

Beginning with Solutions Enabler version 6.0 running on Symmetrix arrays using Enginuity levels greater than version 5671, you set Symmetrix connection security for iSCSI ports using functionality provided by `symconnect` commands. You can set one-way or two-way CHAP authentication between a host HBA and a Symmetrix array. If a RADIUS server is configured to store the authentication data, you can set the RADIUS server information and enable the Symmetrix to look there for the authentication data.

Setting One-Way CHAP Authentication

With CHAP one-way authentication, the authenticator challenges the peer during the initial link negotiation process and expects to receive a valid username/password in response. When challenged, the host peer transmits a CHAP username/password to the authenticator Symmetrix array. The Symmetrix array looks for this username/password in its own CHAP authentication database or on a RADIUS server (if one is set and turned on). Once a positive authentication occurs, the Symmetrix sends an acceptance message to the host. However, if the Symmetrix fails to find any record of the username/password pair, it sends a rejection message, and the link is closed.

Setting iSCSI authentication for a Symmetrix authenticator requires that you supply the Symmetrix ID, the iSCSI name, the CHAP credential name (username), and the CHAP secret (password). The following command inserts an entry for the iSCSI initiator (iqn.2002-06.com.microsoft.host210) into the Symmetrix CHAP authentication database. The `-cred` and `-secret` parameters specify the authentication data that the host should send in response to a challenge by the Symmetrix array.

```
symconnect -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210 \
set chap -cred MyCredentials -secret MySecret
```

The CHAP protocol secret value (MySecret in this case) is a user-defined string up to 32 ASCII characters or 64 binary characters. Binary values should be prefixed with the string "0x". Microsoft users need to specify a secret between 12 and 16 characters and a credential name string between 8 and 256 characters.

The `set chap` command automatically enables CHAP authentication for the iSCSI initiator. You can use `disable chap` to disable CHAP authentication and `enable chap` to turn it back on.

```
symconnect -sid 6208 -iscsi ign.2002-06.com.microsoft.host210 disable chap
symconnect -sid 6208 -iscsi ign.2002-06.com.microsoft.host210 enable chap
```

Setting Two-Way CHAP Authentication

With CHAP two-way authentication, the host challenges and authenticates the Symmetrix too. Thus, both the host and Symmetrix array act as authenticators, and both act as peers. Configuring two-way authentication is just a case of configuring one-way authentication twice, but in opposite directions.

The previous section shows how to set up the Symmetrix array as an authenticator. This section describes authentication from the opposite direction. The following command sets up the Symmetrix array as a peer. That is, it defines what username/password the Symmetrix will send when challenged by the host authenticator.

```
symconnect -sid 6208 -dir 2D -p 0 \
  set chap -cred MyPeerCredentials -secret MyPeerSecret
```

The `-dir` and `-p` parameters specify the Symmetrix director (2D) and port (0) through which a host HBA is connected. In this way you store the username/password that the Symmetrix will transmit to the host when a host HBA connected to the director/port challenges the Symmetrix.

Naturally, the host authenticator must have a user authentication database that contains matching authentication data.

The `set chap` command automatically enables CHAP authentication for the director/port. You can use `disable chap` to disable CHAP authentication and `enable chap` to turn it back on.

```
symconnect -sid 6208 -dir 2D -p 0 disable chap
symconnect -sid 6208 -dir 2D -p 0 enable chap
```

Using a RADIUS Server to Store Authentication Information

If you use a RADIUS server to store the username/password authentication list, you need to use `symconnect` commands to identify the RADIUS server. The Symmetrix looks to the server then for matching authentication data when attempting to authenticate a host's username/password transmission.

You set the RADIUS server information at the Symmetrix director/port level, letting the Symmetrix know where to go for the authentication information when a host HBA tries to log in. The required information includes the server name, its IP address, its key (a password of up to 256 characters), and its rank (primary, backup1, or backup2). For example:

```
symconnect -sid 6208 -dir 2D -p 0 set radius -server AP22 \
  -key MyServerPassword -ip 108.15.139.220 -rank primary
```

The `-rank` parameter establishes the server order preference if two or three RADIUS servers are specified for the director/port. If a primary and two backup servers are enabled, the system tries them in the obvious order until successful (a server might be unreachable for some reason). If `primary` is disabled, the system looks to `backup1` and `backup2`.

If the RADIUS server port to be used is different than the server's default port, you can specify it using the `-port` option. You can also include options that specify a server retry interval in seconds (`-i`) and the number of times to retry (`-c`).

The `set radius` command automatically enables RADIUS authentication for the director/port. You can use `disable radius` to disable authentication and `enable radius` to turn it back on.

```
symconnect -sid 6208 -dir 2D -p 0 disable radius -rank primary
symconnect -sid 6208 -dir 2D -p 0 enable radius -rank primary
```

After all RADIUS server information is set and enabled at the Symmetrix director/port level, you can then enable the server authentication feature for an iSCSI HBA logging into a port. Use either `set radius` or `enable radius` to turn on the RADIUS server authentication for the iSCSI initiator. For example:

```
symconnect -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210 enable radius
```

Similarly, you can use `disable radius` to turn off RADIUS authentication for the iSCSI initiator.

Backing Up, Restoring, Initializing Authentication Information

Backing up authentication information on a regular basis ensures that you can restore a good version of the authentication database in case incorrect changes or other abnormalities occur in the current database. Or you may wish to test temporary changes to the authentication database and restore the backup (pre-test) version after testing is complete.

The following command creates a backup version of the Symmetrix user authentication database in a file called `MyChapBackup` (each time you back up the database, you must choose a new file name).

```
symconnect -sid 6208 -file MyChapBackup backup chap
```

The following command restores this authentication database using the backup version in the file `MyChapBackup`.

```
symconnect -sid 6208 -file MyChapBackup restore chap
```

Similarly, you can backup and restore the authentication database on a RADIUS server by altering these commands slightly: that is, by specifying `backup radius` and `restore radius`.

To display the contents of a backup authentication file, use the `symconnect list` command with the filename.

```
symconnect list -file MyChapBackup
```

Initializing an Authentication Database

In most cases, you do not want to clear the data of an existing authentication database. If you are unsure whether an authentication database currently exists for a Symmetrix array, issue a command that displays any existing Symmetrix CHAP or RADIUS authentication database for the array. For example, to view any authentication database if it exists on Symmetrix 6208:

```
symconnect -sid 6208 -list
```

To initialize and clear an authentication database, you must specify a backup file name to safeguard against clearing data in the database that should not be lost. For example, the following commands create backup files for Symmetrix CHAP and RADIUS authentication databases and attempt to write any current authentication data to these backup files prior to initializing:

```
symconnect -sid 6208 init chap -file MyChapInitBackup
symconnect -sid 6208 init radius -file MyRadiusInitBackup
```

Displaying Authentication Information

You can use the basic `symconnect list` command to display a Symmetrix array's authentication database (for example, any Symmetrix CHAP or RADIUS authentication database for Symmetrix 20).

```
symconnect -sid 20 list
```

To display just the Symmetrix CHAP authentication database, include the `chap` parameter. For example:

```
symconnect -sid 20 list chap
```

```
Symmetrix ID           : 000190300020
```

```
Director Identification : SE-2D
```

```
Director Port          : 0
```

```
Protocol                : CHAP
```

Identifier	Type	State	Credential
SE-2D:0	N/A	ENABLED	symm20
iqn.2002-06.com.microsoft.host210	iSCSI	DISABLED	api210
iqn.2002-06.com.microsoft.host211	iSCSI	ENABLED	api211

CHAP credentials have been defined in the Symmetrix authentication database for two iSCSI initiators, making it possible for the Symmetrix to authenticate either iSCSI HBA logging into its port. Credentials have also been defined for the Symmetrix director/port (SE-2D:0) in the event that two-way authentication is required and the Symmetrix needs to submit authentication data to the host.

To display just a RADIUS authentication database, include the `radius` parameter. For example:

```
symconnect -sid 20 list radius
```

```
Symmetrix ID           : 000190300020
```

```
Director Identification : SE-2D
```

```
Director Port          : 0
```

```
Protocol                : RADIUS_SERVER
```

Server Rank	State	Server Name	IP Address	Port
Primary	ENABLED	AP22	108.15.139.220	1812 <- default
1st Backup	ENABLED	AP23	108.15.139.221	1812 <- default
2nd Backup	ENABLED	AP24	108.15.139.222	1812 <- default

```
Protocol                : RADIUS
```

Identifier	Type	State
iqn.2002-06.com.microsoft.host210	iSCSI	DISABLED
iqn.2002-06.com.microsoft.host211	iSCSI	ENABLED

Example 1: Adding Masked Devices for HBA Access

The hardware setup consists of a single Symmetrix array (sid 313) connected to a Solaris controlling host.

The following command lists host HBAs that have a channel to a Symmetrix array and generates an AWWN (the User-generated Name column) for any HBA that does not have an AWWN assigned yet. This information is stored in the login history table. The Identifier column displays an HBA's unique WWN.

```
# symmask discover hba
```

```
Symmetrix ID       : 000184500313
Device Masking Status : Success
```

Identifier	Type	User-generated Name
10000000c9274156	Fibre	api140/i@1f,4000,@2
10000000c92741a1	Fibre	api140/i@1f,4000,@4

The following command lists the FA director/port (Dir:P) to which each HBA connects and the various physical device paths of the VCMDB device on the Symmetrix array. Notice that one of the director/ports (14D, port 0) is connected to both HBAs. Either HBA can access devices that are available to 14D:0.

```
# symmask list hba
```

Identifier	Type	Adapter	Physical Device Path	Dir:P
10000000c9274156	Fibre	i@1f,4000,@2	/dev/rdisk/c1t0d0s2	14B:0
			/dev/rdisk/c1t1d0s2	14D:0
10000000c92741a1	Fibre	i@1f,4000,@4	/dev/rdisk/c2t0d0s2	14B:0
			/dev/rdisk/c2t2d80s2	14C:0
			/dev/rdisk/c2t1d0s2	14D:0

The `symcfg list` command with the `-addr` option displays a list of devices that are available to a specified director/port (in this case, FA director 14D). This example uses devices 0019, 0060, 00A0, and 00D4. Notice that devices 0060 and 00A0 are meta devices (M).

```
# symcfg list -FA 14D -addr
```

```
Symmetrix ID: 000184500313 (Local)
```

Director		Device Name			Attr	Address		
Ident	Symbolic	Port	Sym	Physical		VBUS	TID	LUN
FA-14D	14D	0	0402	/dev/rdisk/c1t1d0s2	VCM	0	0	000
			0019	Not Visible		0	0	001
			001A	Not Visible		0	0	002
			001B	Not Visible		0	0	003
			001C	Not Visible		0	0	004
			001D	Not Visible		0	0	005
			001E	Not Visible		0	0	006
			001F	Not Visible		0	0	007
			0060	Not Visible	(M)	0	0	010
			00A0	Not Visible	(M)	0	0	011
			00D4	Not Visible		0	0	019

00D5	Not Visible	0	0	01A
00D6	Not Visible	0	0	01B
00D7	Not Visible	0	0	01C
00D8	Not Visible	0	0	01D
00D9	Not Visible	0	0	01E
00DA	Not Visible	0	0	01F
00DB	Not Visible	0	0	020
0018	Not Visible	0	0	030
03FD	Not Visible	0	0	060

The `symmask add` command adds devices 0019, 0060, 00A0, and 00D4 to the HBA whose WWN identifier is 10000000c92741a1 and whose director/port channel is director 14D, port 0.

```
# symmask add dev 19,60,A0,D4 -wwn 10000000c92741a1 -dir 14D -p 0
```

The following command checks the device masking database (VCMDB) to determine that the addition was correct. For the two meta devices, all of the meta members were added: devices 0060 through 0067, and devices 00A0 through 00A7.

```
# symmaskdb list database
```

```
Symmetrix ID          : 000184500313
```

```
Last updated at       : N/A
```

```
Director Identification : FA-14D
```

```
Director Port         : 0
```

Identifier	Type	User-generated		Devices
		Node Name	Port Name	
10000000c92741a1	Fibre	api140	i@1f,4000,@4	0019 0060:0067 00A0:00A7 00D4

EMC recommends backing up the VCMDB after *all* additions are made (not after each `symmask add` command). Since these are the only additions that this example makes, the following command backs up the VCMDB to a file called backup0723.

```
# symmaskdb backup -file backup0723
```

```
Backup Symmetrix SymMask database on Symmetrix 000184500313 to backup file  
backup0723 (y/[n])? y
```

```
Symmetrix SymMask database backed up to file backup0723 from Symmetrix  
000184500313
```

After completing your additions, refresh the WWN-related profile tables in Symmetrix director cache with the latest VCMDB data. The following command updates the cache on the FA director with the contents of the VCMDB.

```
# symmask refresh
```

```
Refresh Symmetrix FA directors with contents of SymMask database 000184500313  
(y/[n]) ? y
```

```
Symmetrix FA directors updated with contents of SymMask Database 000184500313
```

Reboot the host whose HBA entry you added to the VCMDB and then issue the `symcfg discover` command to scan the Symmetrix devices and refresh the SYMAPI database.

```
# symcfg discover
```

Example 2: Using Fibre Channel ID Lockdown

This example uses the same hardware setup as Example 1 and demonstrates the security feature that limits host access to devices by adding the Fibre Channel ID (FCID) value to an HBA entry in the VCMDB.

This command allows you to determine the physical device path for the HBA you want to lock down and the director/port on which it will be locked down. The example uses the first WWN on director/port 14B:0, which has a physical device path of /dev/rdsd/c1t0d0s2.

```
# symmask list hba
```

Identifier	Type	Adapter	Physical Device Path	Dir:P
10000000c9274156	Fibre	i@1f,4000,@2	/dev/rdsd/c1t0d0s2	14B:0
			/dev/rdsd/c1t1d0s2	14D:0
10000000c92741a1	Fibre	i@1f,4000,@4	/dev/rdsd/c2t0d0s2	14B:0
			/dev/rdsd/c2t2d80s2	14C:0
			/dev/rdsd/c2t1d0s2	14D:0

The `symmask list logins` command allows you to determine the FCID value (100200) of the Fibre Channel to be locked down (director/port 14B:0, and physical device path /dev/rdsd/c1t0d0s2).

```
# symmask list logins -dir 14B -p 0 -pdev /dev/rdsd/c1t0d0s2
```

```
Director Identification : FA-14B
Director Port           : 0
```

Identifier	Type	User-generated		FCID	Logged On	
		Node Name	Port Name		In	Fabric
10000000c920c484	Fibre	NULL	NULL	100001	Yes	Yes
10000000c9274156	Fibre	api140	i@1f,4000,@2	100200	Yes	Yes
10000000c92741a1	Fibre	api140	i@1f,4000,@4	100500	Yes	Yes
210000e08b046406	Fibre	NULL	NULL	100800	Yes	Yes
210000e08b046506	Fibre	NULL	NULL	100a00	Yes	Yes
210100e08b246506	Fibre	NULL	NULL	100b00	Yes	Yes

The following command locks downs the Fibre Channel specified by the FCID 100200.

```
# symmask set lockdown on 100200 -wwn 10000000c9274156 -dir 14B -p 0
```

The following command with the verbose (-v) option provides a detailed display of HBA-to-FA channel entry in the VCMDB and verifies that the correct FCID value was added to this entry.

```
# symmaskdb list database -dir 14B -p 0 -v

Symmetrix ID           : 000184500313

Last updated at        : N/A

Director Identification : FA-14B
Director Port          : 0

    Originator Port wwn : 10000000c9274156
    Type                : Fibre
    User-generated Name : apil40/i@1f,4000,@2
    Visibility          : No
    FCID Lockdown       : Yes
    FCID Value          : 100200
    Lun Offset          : No
    Heterogeneous Host  : No
    Devices              : 0008 000A:000F 00D5
```

The following command backs up the VCMDB to a file named backup0723.

```
# symmaskdb backup -file backup0723

Backup Symmetrix SymMask database on Symmetrix 000184500313 to backup file
backup0723 (y/[n])? y

Symmetrix SymMask database backed up to file backup0723 from Symmetrix
000184500313
```

If you have completed your entire set of changes for this session, refresh the VCMDB in the Symmetrix cache.

```
# symmask refresh

Refresh Symmetrix FA directors with contents of SymMask database 000184500313
(y/[n]) ? y

Symmetrix FA directors updated with contents of SymMask Database 000184500313
```

Example 3: New Options When Displaying the VCMDB

The following `symmaskdb list` commands are available with Solutions Enabler version 5.1 or higher.

The following `symmaskdb list capacity` command displays the capacity of all devices assigned through an HBA-to-FA channel to host `api140`. The ellipsis (...) indicates output that was omitted.

```
# symmaskdb list capacity -host api140
```

```
Symmetrix ID      : 000184500313
```

```
Host Name         : api140
```

```
Identifiers Found : 10000000c9274156
                   10000000c92741a1
```

Device	Cap(MB)	Attr	Dir:P
-----	-----	----	----
0009	1031		14B:0
000A	1031		14B:0
000B	1031		14B:0
000C	1031		14B:0
000D	1031		14B:0
000E	1031		14B:0
000F	1031		14B:0
0011	1031		14C:0
0012	1031		14C:0
0013	1031		14C:0
0014	1031		14C:0
0015	1031		14C:0
0016	1031		14C:0
0017	1031		14C:0
0050	8250	(M)	14B:0
0051	-	(m)	14B:0
0052	1031		14B:0
0053	-	(m)	14B:0
0054	-	(m)	14B:0
0055	-	(m)	14B:0
0056	-	(m)	14B:0
0057	-	(m)	14B:0
0090	8250	(M)	14B:0
0091	-	(m)	14B:0
0092	-	(m)	14B:0
0093	-	(m)	14B:0
0094	-	(m)	14B:0
0095	-	(m)	14B:0
0096	-	(m)	14B:0
0097	-	(m)	14B:0
.....			
0402	7		14C:0

```
-----
MB Total:  42282
GB Total:  41.3
```

The `symmaskdb list assignment` command displays which HBAs are assigned to a set of devices. The range of devices specified is 0001 through 000A.

```
# symmaskdb list assignment -dev 0001:000A
```

```
Symmetrix ID : 000184500313
```

Device	Identifier	Type	Dir:P
-----	-----	-----	-----
0001	10000000c9229db5	FIBRE	14A:0
0002	10000000c9229db5	FIBRE	14A:0
0003	10000000c9229db5	FIBRE	14A:0
0004	10000000c9229db5	FIBRE	14A:0
0005	10000000c9229db5	FIBRE	14A:0
0006	10000000c9229db5	FIBRE	14A:0
0007	10000000c9229db5	FIBRE	14A:0
0008	-	-	-
0009	10000000c9274156	FIBRE	14B:0
000A	10000000c9274156	FIBRE	14B:0