



# Guidelines for the Attachment of iSCSI Hosts - Errata

**Version 5.1.X**

November 3rd, 2009

# Table of Contents

Introduction .....	3
Who should use this guide .....	3
Last update .....	3
Change History .....	3
<b>System Prerequisite</b> .....	4
The Minimum <i>IBM System Storage SAN Volume Controller</i> Level for support of iSCSI attachment is Version 5.1.0 .....	4
Introduction .....	4
What is an iSCSI network? .....	4
iSCSI initiators and targets .....	4
Native and Heterogeneous iSCSI network .....	5
SAN Volume Controller iSCSI support .....	6
Ethernet Ports changes in SAN Volume Controller .....	8
iSCSI configuration with a single subnet .....	9
iSCSI configuration with multiple subnets and dual configuration interfaces .....	9
iSCSI configuration and host multipathing driver considerations .....	10
• Concurrent use of IPv4 and IPv6 on a single host .....	12
Host limits with iSCSI support: .....	12
Setting up your host server .....	14
Additional Considerations .....	14
Linux .....	15
Installing the Linux software iSCSI Initiator .....	15
Setting the initiator iSCSI qualified name .....	15
iSCSI target discovery .....	15
Discovering iSCSI targets using iSNS .....	15
Logging in to discovered targets .....	16
Logging out from targets .....	16
Setting up Authentication for Linux Hosts .....	16
Recommended iSCSI configuration for the SAN Volume Controller .....	18
Multipathing on Linux .....	18
• Windows .....	19
Obtaining and installing the initiator software .....	19
Setting the initiator name (IQN) .....	19
Target discovery .....	20
Connecting to discovered targets .....	20
Viewing and managing the discovered disks .....	21
The Windows Disk Initialization wizard .....	21
Optimizing the initiator for SAN Volume Controller .....	21
Setting up Authentication for Windows Hosts .....	21
AIX .....	27
Installing the AIX iSCSI software initiator .....	27
Configuring the AIX iSCSI software initiator .....	27
iSCSI target configuration and LUN discovery .....	27
Isattr -E -l hdisk10 -a rw_timeout .....	28
Setting up Authentication for AIX Hosts .....	28
ODM stanzas for SAN Volume Controller iSCSI devices .....	29
Solaris .....	30

Installing the Solaris iSCSI initiator.....	30
Inspecting the iSCSI configuration.....	31
Special note on Header Digest and Data Digest.....	33
Multipathing on Solaris.....	34
Setting up Authentication for Solaris Hosts .....	35
HP-UX .....	36
Installing the HP-UX iSCSI initiator.....	36
Configuring the HP-UX iSCSI initiator :- .....	36
Limitations of HPUX iscsi software initiator .....	39
HP-UX Native Multipathing:-.....	39
Additional consideration.....	40
Limiting queue depth in iSCSI network .....	41
Queue depth .....	41
Calculating a queue depth limit.....	41
Homogeneous queue depth calculation .....	41
Nonhomogeneous queue depth calculation .....	42
Limiting the queue depth.....	43

## Introduction

In previous releases of SAN Volume Controller, host servers communicated with the SAN Volume Controller system using a Fibre Channel SAN. SAN Volume Controller version 5.1 introduces a new host attachment method, using an iSCSI network, while continuing to support all the existing features of Fibre Channel host attachment.

**This guide provides updated information for iSCSI host attachment to *IBM System Storage SAN Volume Controller* and supersedes any iSCSI documentation contained in the following *IBM System Storage SAN Volume Controller Version 5.1.x* publications**

**Software Installation and Configuration Guide  
Host Attachment User's Guide  
Planning Guide**

## Who should use this guide

This errata should be used by anyone using *the IBM System Storage SAN Volume Controller Host Attachment Guide* that wishes to attach the following to their SAN Volume Controller

- iSCSI Hosts

## Last update

This document was last updated: November 3<sup>rd</sup> 2009

## Change History

The following revisions have been made to this document:

Date	Sections Modified
3rd November 2009	New Document

# System Prerequisite

The Minimum *IBM System Storage SAN Volume Controller* Level for support of iSCSI attachment is Version 5.1.0

## Introduction

In previous releases of SAN Volume Controller, host servers communicated with the SAN Volume Controller system using a Fibre Channel SAN. SAN Volume Controller 5.1 introduces a new host attachment method, using an iSCSI network, while continuing to support all the existing features of Fibre Channel host attachment.

## What is an iSCSI network?

In an iSCSI network, host servers use a conventional IP fabric to communicate with the storage subsystems and themselves. iSCSI network have analogous components to a Fibre Channel SAN:

### iSCSI

iSCSI Host Bus Adapter  
NIC and iSCSI software initiator  
IP Switch  
IP Router  
IQN

### Fibre Channel

Fibre Channel Host Bus Adapter  
Fibre Channel Host Bus Adapter  
FC Switch  
  
WWNN

## iSCSI initiators and targets

In an iSCSI storage configuration, the iSCSI host (or server) sends requests to an iSCSI storage system. The host contains one or more initiators that attach to an IP network, which initiate requests to, and receives responses from an iSCSI target. Each initiator and target are given a unique iSCSI qualified name (IQN) – this name is used to identify each initiator or target in the same way that WorldWide Node Names (WWNNs) are used to identify devices in a Fibre Channel fabric.

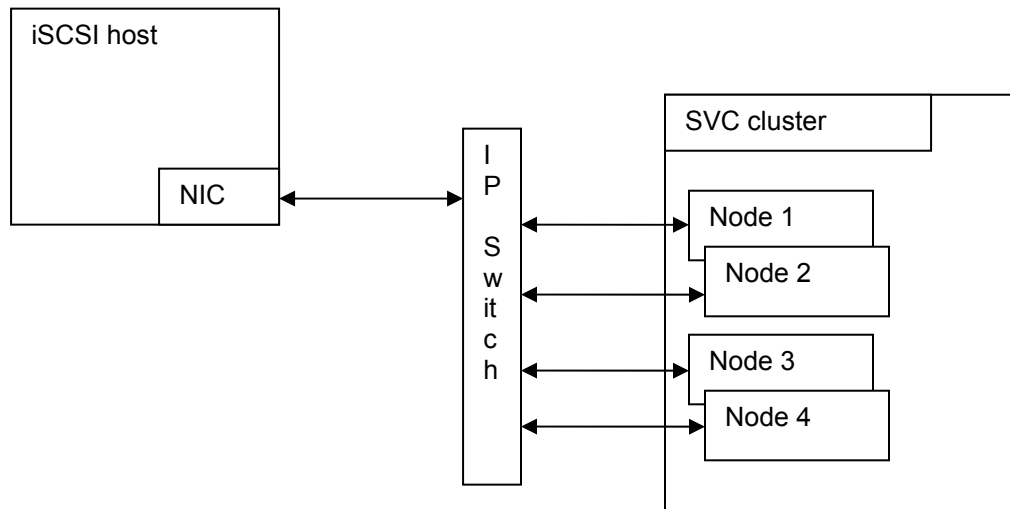
An iSCSI target is a storage device that receives SCSI commands over TCP/IP network. The device can be an end node, or it can be an intermediate device, such as a bridge between IP and Fibre Channel devices. In the current release, a SAN Volume Controller is an iSCSI target. Each iSCSI target is identified by a unique IQN, and each port on the storage array controller (or on a bridge) is identified by one or more IP addresses

To transport block (SCSI) commands over the IP network, an iSCSI driver must be installed on the iSCSI host and target. The driver allows iSCSI commands and responses to be sent via a NIC (Network Interface card) or an iSCSI HBA in the host or target hardware.

For maximum performance a Gigabit Ethernet adapter (transmitting 1000 megabits per second--Mbps) is recommended for connection between iSCSI host and iSCSI target. Like the standard 10/100 adapters, most Gigabit adapters use Category 5 or Category 6E cabling that is already in place. Each port on the adapter is identified by a unique IP address.

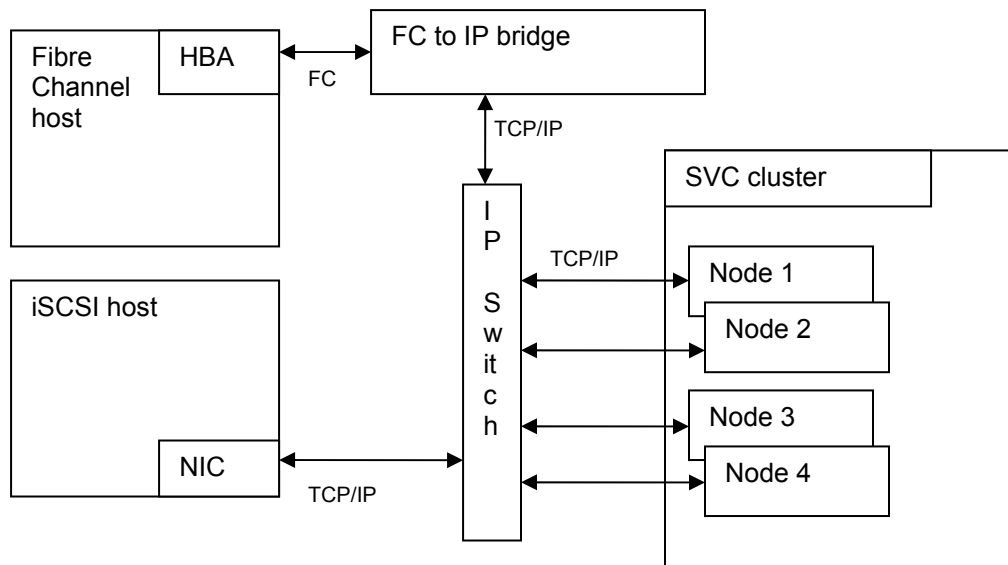
## Native and Heterogeneous iSCSI network

The relationship between an iSCSI initiator and the iSCSI target is shown in Figure 1. In this case, the iSCSI initiator (or client) is a host server and the iSCSI target is the storage array. This topology is considered a native iSCSI network, because it consists entirely of components that transmit the SCSI protocol over TCP/IP.



**Figure 1 - iSCSI support in an iSCSI local network**

In contrast, heterogeneous iSCSI networks, such as the one illustrated in Figure 2, consist of components that transmit SCSI both over TCP/IP and over Fibre Channel interconnects. To accomplish this, a bridge or gateway device is installed between the IP and the Fibre Channel components. The bridge serves to translate between the TCP/IP and Fibre Channel protocols, so that the iSCSI host sees the storage as an iSCSI target.



**Figure 2 - iSCSI support in a heterogeneous SAN**

## **SAN Volume Controller iSCSI support**

SAN Volume Controller iSCSI support is characterised as follows:

- SAN Volume Controller supports iSCSI Host attachment. SAN Volume Controller will act as iSCSI target and will represent virtualized storage disks to iSCSI initiator(s).
- iSCSI names . iSCSI uses either iqn names (223 bytes) ASCII names or eui (64 bit identifier) to represent a world wide unique naming scheme for iSCSI initiators
- IP take over facility in an I/O group should be ensured. In an I/O group if one of the SAN Volume Controller nodes acting as iSCSI target fails, then the other node should be able to take over the failed node IP addresses, so the iSCSI initiator will still be able to continue its operations. During takeover the iSCSI initiator will be logged out from the

failed node and a new session/login needs to be established with the working node with the IP address of failed node.

- New commands have been introduced from SAN Volume Controller 5.1 release to configure iSCSI. SAN Volume Controller commands such as `svctask mkhost` have been modified to also allow iSCSI names to define host objects.
- The NIC cards on an SAN Volume Controller node will carry both iSCSI traffic and be used for configuration UI traffic.
- SAN Volume Controller supports the standard iSCSI port (3260).
- iSCSI IP addresses can be IPv4 or IPv6.
- A single iSCSI target supports IPv4 and IPv6 concurrently.  
This SAN Volume Controller release only supports a single host initiator connection (either IPv4 or IPv6 but not both) between a SAN Volume Controller target port and the host initiator. Since each SAN Volume Controller node is a target and each SAN Volume Controller node has two physical Ethernet ports this means that the host initiator should choose to login to just one of these ports. Different hosts can login to different ports on the same node, thus achieving load balancing. Similarly although each SAN Volume Controller node port supports both IPV4 and IPV6 connections, each host should only use one connection type on each node port.
- SAN Volume Controller VDisks are mapped to a host server using the same 'VDisk-hostmapping' mechanism as used on Fibre Channel attachment. This means an SAN Volume Controller VDisk can be mapped either to an FC host and/or an iSCSI host. Mapping a VDisk through both iSCSI and FC to the same host is not supported in this SAN Volume Controller release.
- This SAN Volume Controller release supports a maximum mapping of VDisks to 256 FC hosts or 64 iSCSI hosts per SAN Volume Controller I/Ogroup.
- All IP addresses (Service & Configuration IP) associated with a Cluster Ethernet Port must be on same subnet. But IP addresses associated with Node Ethernet Ports (used for iSCSI traffic) can be configured to be a part of different subnets.
- SAN Volume Controller supports
  - iSNS
  - CHAP based authentication
- If a Host has multiple iSCSI clients (multiple IQNs) and different authentication secrets are to be used with each different client then the user must create a host object with a different name for use with each iSCSI client in SAN Volume Controller. Then use these appropriate IQNs and corresponding secrets in each of the corresponding host objects. The user must then use all these host objects to map the VDisk. As only a single session for each SAN Volume Controller target is supported per host IQN, the user should ensure that each host IQN logins into a different IP on the IO group for the VDisk of interest.
- Current SAN Volume Controller release supports IO from different initiators in the same host to the same IO group
- Current SAN Volume Controller release does support I/O from different initiators in different hosts to the same VDisks

- Current SAN Volume Controller release does support I/O from FC and iSCSI initiators in different hosts to the same VDisks.
- Current SAN Volume Controller release does not support I/O from FC and iSCSI initiators in the same hosts to the same VDisks
- The new CLI in this release of SAN Volume Controller introduces the concept of a clustered Ethernet port that consists of one Ethernet port from each node in the cluster that are connected to the same Ethernet switch. As SAN Volume Controller nodes have 2 Ethernet ports there will be 2 clustered Ethernet ports. Ethernet configuration commands now affect either a clustered Ethernet port (e.g. assigning gateway addresses, netmasks, configuration IP address, etc.) or a node Ethernet port (e.g. assigning an IP address to a port for iSCSI). The CLI design allows a different cluster/service IP address to be assigned to each clustered Ethernet port so that it is now possible to configure SAN Volume Controller clusters with redundant Ethernet networks.
- The SAN Volume Controller command line interface associates iSCSI authentication information with a SAN Volume Controller host object rather than with a SAN Volume Controller host port. Typically a physical host server will have a single iSCSI name and therefore can be represented by a SAN Volume Controller host object with a single iSCSI host port. A clustered host server may have multiple iSCSI names with different authentication information for each host - in this case it will be necessary to create multiple SAN Volume Controller host objects each with a single iSCSI host port and different authentication information
- Current SAN Volume Controller release supports the use of the iSNS protocol. This allows iSCSI initiators to locate appropriate storage resources using the Internet Storage Name Service (iSNS) protocol.

## Ethernet Ports changes in SAN Volume Controller

In SAN Volume Controller 4.3.1 and before, the SAN Volume Controller cluster presented a single IP interface:

From SAN Volume Controller 5.1 onwards, both Ethernet ports (1 and 2) are used, either for configuration access or for iSCSI attachment.

Please refer to SAN Volume Controller Planning Guide for more information on Ethernet Port related information.

The following examples show how SAN Volume Controller uses the redundant ports provided for the SAN Volume Controller management interface and for the iSCSI target ports on each IO group.

Note: The examples only illustrate the use of IPv4 addresses

- The equivalent configurations can be setup with just IPv6 addresses
- The equivalent configurations can be setup with IPv4 and IPv6 addresses
- It is valid to use IPv4 addresses on one port and IPv6 addresses on the other
- It is valid to have different subnet configurations for IPv4 and IPv6 addresses



## iSCSI configuration with a single subnet

Figure 3 shows a 2 node SAN Volume Controller cluster connected to a single subnet. Each node has two Ethernet ports, each of which is used for iSCSI data transfers. One node in the cluster also acts as the cluster configuration node. In this case port 1 on the configuration node provides the cluster management IP interface.

I

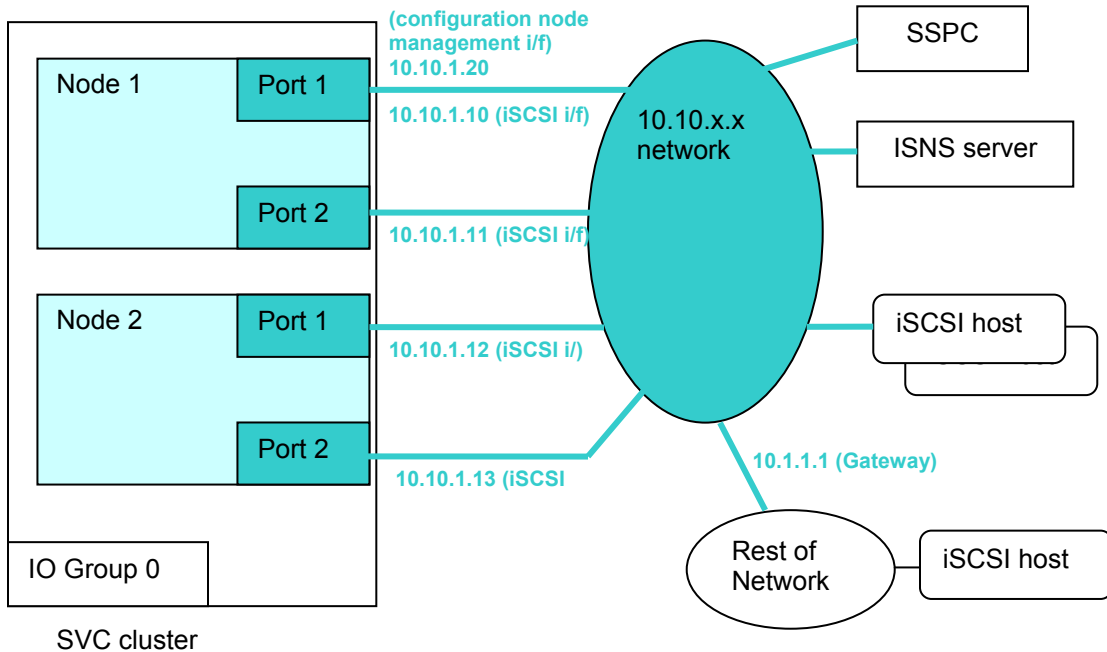
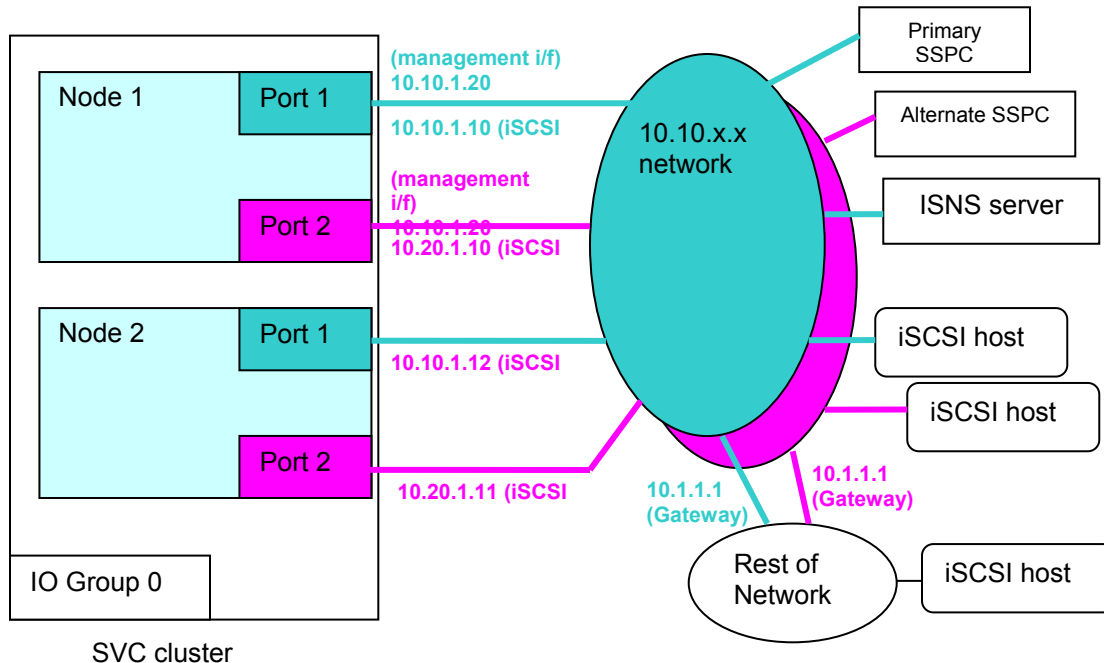


Figure 3 - iSCSI configuration using a single subnet

## iSCSI configuration with multiple subnets and dual configuration interfaces

Figure 3 shows a 2 node SAN Volume Controller cluster connected to multiple subnets. Each node has two Ethernet ports, port1 and port 2, and these are connected to different IP subnets. In addition, one node in the cluster also acts as the cluster configuration node and this provides alternate IP interfaces, again on different subnets for the cluster management interface.



**Figure 4 - iSCSI configuration using multiple subnets and providing alternate configuration interfaces**

## iSCSI configuration and host multipathing driver considerations

Figure 5 shows a 2 node SAN Volume Controller cluster connected to multiple subnets. Each node has two Ethernet ports, port1 and port 2, and these are connected to different IP subnets. In addition, one node in the cluster also acts as the cluster configuration node and this provides alternate IP interfaces, again on different subnets for the cluster management interface.

In Figure 5 Host 1 does not use multipathing. A vdisk in the SVC IO group will appear as 4 separate devices in Host 2. The host will select one device to do i/o to the vdisk – this corresponds to a particular IP address at an SVC node port, eg 10.10.1.10. If a connection between the host and this SVC port is broken (e.g. if the link at X is broken), then, if IO is in progress, an I/O error will be recorded on host 1 for that vdisk. No SVC state changes or IP failover take place.

In Figure 5, Host 2 does use Multipathing. A vdisk in the SVC IO group will therefore appear as a single device to the applications on host 2, even though the multipathing driver can see 4 separate devices for each vdisk. The Multipathing driver will select one or more of these devices when doing i/o. If the connection between the host and one SVC node port is lost, the Multipathing driver can select an alternative path to the SVC IO group and the i/o between the host and SVC will continue without error. However, host 2 only has one NIC and will therefore report i/o errors if the connection between that NIC and the network is lost (eg if the link at Y is broken).

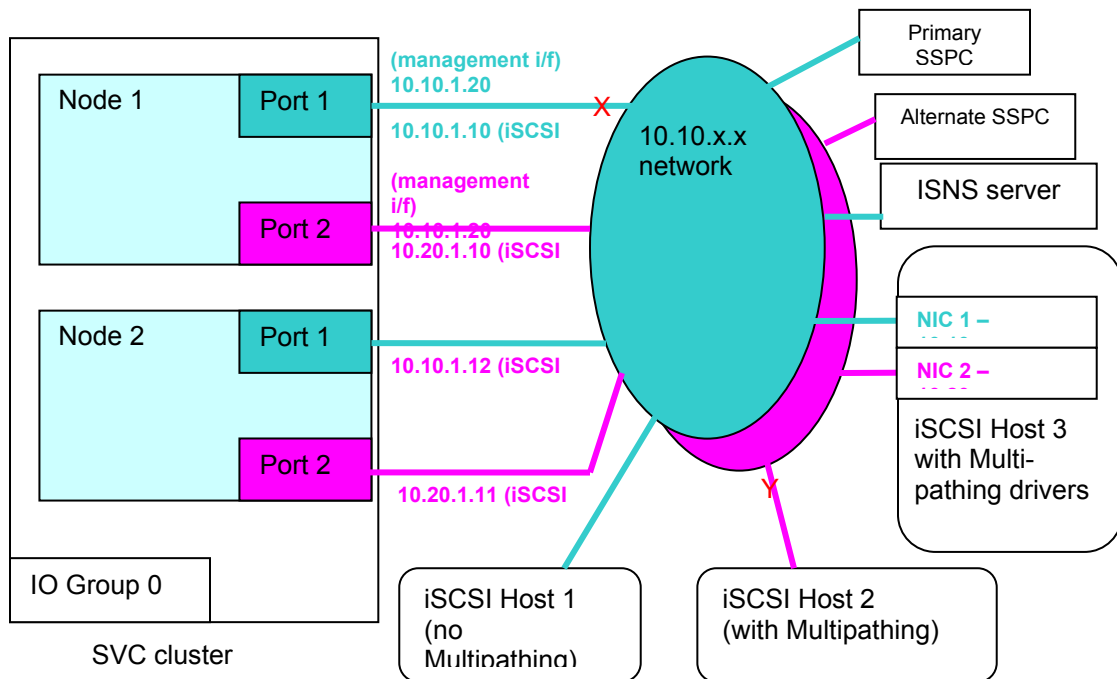
Host 1 does not use multipathing. A VDisk in the SAN Volume Controller IO group will appear as 4 separate devices in Host 2. The host will select one device to do i/o to the VDisk – this

corresponds to a particular IP address at an SAN Volume Controller node port, eg 10.10.1.10. If a connection between the host and this SAN Volume Controller port is broken (e.g. if the link at X is broken), then, if IO is in progress, an I/O error will be recorded on host 1 for that VDisk. No SAN Volume Controller state changes or IP failover take place.

In Figure 5, Host 2 does use Multipathing. A VDisk in the SAN Volume Controller IO group will therefore appear as a single device to the applications on host 2, even though the multipathing driver can see 4 separate devices for each VDisk. The Multipathing driver will select one or more of these devices when doing i/o. If the connection between the host and one SAN Volume Controller node port is lost, the Multipathing driver can select an alternative path to the SAN Volume Controller IO group and the i/o between the host and SAN Volume Controller will continue without error. However, host 2 only has one NIC and will therefore report i/o errors if the connection between that NIC and the network is lost (eg if the link at Y is broken).

In Figure 5, Host 3 uses Multipathing and redundant NICs. This means if a NIC fails then the Multipathing driver can still find paths from the host to a VDisk in the SAN Volume Controller IO group and application i/o's can continue without error. In addition, since the NICs are connected to different IP networks, the overall configuration can tolerate a single network failing without I/O errors occurring on Host 3.

From this it can be seen that iSCSI support has been designed in SAN Volume Controller such that multipathing drivers are not required to do cluster maintenance (when SAN Volume Controller nodes are removed or replaced in an IO group). Multipathing host drivers are however required for load balancing and for surviving NIC, link or network failures.



**Figure 5 - iSCSI configuration with host multipathing**

- **Concurrent use of IPv4 and IPv6 on a single host**

If you have both ipv4 and ipv6 addresses configured at the target ip port of the SAN Volume Controller nodes, and your host is running an ipv4 and ipv6 stack, then you should only discover SAN Volume Controller devices using the SAN Volume Controller IP addresses rather than the SAN Volume Controller IQN names in the host discovery command, and you should choose to use just an IPV4 or an IPV6 address, but not both for each SAN Volume Controller node port. The reason for this restriction is that the SAN Volume Controller code only supports a single connection between an SAN Volume Controller target port and a host initiator.

So if you have a 2 node cluster, with nodes A and B and

node A port 1 has ipv4 and ipv6 addresses  
node A port 2 has ipv4 and ipv6 addresses  
node B port 1 has ipv4 and ipv6 addresses  
node B port 2 has ipv4 and ipv6 addresses

You should only run a discovery command on your host that discovers target devices at node A port 1 ipv4 or node A port 1 ipv6 but not both. You can choose to mix ipv4 and ipv6 discovery on different node ports. (To ensure this happens you should use the node ip address in the host discovery command, as this means you can explicitly indicate a single ip address in the discovery command. If you use the IQN name for the SAN Volume Controller node port in the discovery command then you are implicitly saying you want to discover at ipv4 and ipv6 on the node port, if such addresses exist).

If you just have a single IPV4 or IPV6 address configured at each node port then you can use the normal IQN name in the host discovery methods.

### **Host limits with iSCSI support:**

iSCSI support in SAN Volume Controller allows a SAN Volume Controller Host to be created with iSCSI name . For host mapping purposes, an iSCSI name replaces a WWPN in the SAN Volume Controller host mapping tables. This means that the limits on WWPN mapping in previous versions of SAN Volume Controller now also have an effect on the maximum number of iSCSI mappings supported in this current release of SAN Volume Controller.

In this release of SAN Volume Controller there are 512 WWPN entries per SAN Volume Controller iogrp. An iSCSI host object can use upto a maximum of 4 WWPNs allocations in SAN Volume Controller depending on the number of I/O groups the host participates in. This introduces some changes in maximum number of hosts that can be configured in a SAN Volume Controller cluster.

Assuming a host is created using a single iSCSI IQN Name, The max number of hosts (X) that participate in multiple iogrps (Y) can be calculated by using  $(X) \leq 512 / (Y)$ .

Examples:

If we create iscsi hosts with single iscsi name in single iogrp, say io\_grp0. (These host objects would be created with commands of the form `svctask mkhost -iscsiname iscsi1 -iogrp 0` refer to The Command Line Interface User's Guide for details):

- A maximum of 256 iSCSI hosts can be created in iogrp 0. Or a mix of up to 256 FC and iSCSI based hosts can be created in iogrp 0.

- 256 FC or iSCSI based hosts can be created in each of iogrp 1,2,3
  - **Total hosts for the cluster 1024**
1. If we create a host with single iscsi name in two iogrps (svctask mkhost -iscsiname iscsi1 -iogrp 0:1 ):
    - Up to 256 iscsi hosts can be created in iogrp 0,1
    - 256 FC or iSCSI hosts can be created in iogrp 2
    - 256 FC or iSCSI hosts can be created in iogrp 3
    - **Total hosts for the cluster 768**
  2. If we create a host with single iscsi name in three iogrps (svctask mkhost -iscsiname iscsi1 -iogrp 0:1:2 )
    - 170 hosts max can be created in iogrp 0,1,2
    - 256 FC or iSCSI hosts can be created in iogrp 3
    - **Total hosts for the cluster 426**
  3. If we create a host with single iscsi name in four iogrps (svctask mkhost -iscsiname iscsi1 ): ---> 128 hosts can be created in iogrp, giving 128 total hosts for the cluster, versus 256 for Fibre Channel.
    - hosts max can be created in a iogrp 0,1,2,3
    - **Total hosts for the cluster 128**

**Note: At the time of writing, (October 2009), SAN Volume Controller 5.1 release is tested with a maximum of 64 iSCSI hosts per IO group, or 256 iSCSI hosts per cluster.**

## Setting up your host server

The following basic procedure must be performed when setting up your host server for use as an iSCSI initiator with SAN Volume Controller VDisks. The specifics of some of the steps will vary according to the particular host type and operating system involved.

1. Set up your SAN Volume Controller cluster for iSCSI.
  - a. Select a set of ipv4 or ipv6 address for the clustered Ethernet ports on the node that are in IO groups that will use iSCSI VDisks.
  - b. Configure the node Ethernet ports on each node in the cluster using the 'svctask cfgportip' command refer The Command Line Interface User's Guide for details.
  - c. Verify that you have configured the node and clustered Ethernet ports correctly by reviewing the output of the 'svcinfoloportip' and 'svcinfolclusterip' commands.
2. Set up your host server.
  - a. Ensure you have configured your IP interfaces on the server
  - b. Install the software for the iSCSI software initiator on the server
3. Create VDisks on the SAN Volume Controller cluster using the 'svctask mkvdisk' command.
4. Create a host object SAN Volume Controller server that describes the iSCSI server initiator to which the VDisks are to be mapped using the 'svctask mkhost' command.
5. Map the VDisk to the host object in SAN Volume Controller using the 'svctask mkvdiskhostmap' command refer The Command Line Interface User's Guide for details.
6. On the host server run the configuration methods for iSCSI so that the host server iSCSI initiator logs in to the SAN Volume Controller cluster and discovers the SAN Volume Controller VDisks – the host will then create host devices for these VDisks
7. The host devices that were created in (6) can then be used with your host applications.

The details for setting up each host server type are described below.

## Additional Considerations

1. At the time of writing (Oct 2009), AIX does not support iSCSI with IPV6.
2. iSCSI target name (IQN) comprises of Cluster\_name and node\_name refer to the commands svctask chcluster and svctask chnode in The Command Line Interface User's Guide for more information. Please note that changing the node name or cluster name will result in a change of iSCSI target name (IQN) change and will require all hosts to halt I/O and rediscover their targets.
3. Each connection between a host initiator and SAN Volume Controller clustered Ethernet port may use IPV4 or IPV6 (where supported). Concurrent use of both IPV4 and IPV6 for the some initiator to SAN Volume Controller clustered Ethernet port is not supported.
4. If MultiPath drivers are not selected at the host end then special care needs to be taken through configurations by:

- a) Performing iSCSI Login through just one target IP address from the concerned host. Otherwise multiple paths via multiple target IP's to the same SAN Volume Controller VDisk risk corruption, due to the host end usage.
- b) Ensure Application use the LUNID to identify the multiple device entries that correspond to same target. Thence, it's important that hosts ensure that they don't write to same disk thinking it's a different disk and risk data corruption.

## Linux

### Installing the Linux software iSCSI Initiator

The SAN Volume Controller supports the use of the iSCSI initiator software provided by your Linux distribution vendor. You should ensure that you have the latest release of the initiator software installed on your host before continuing. The initiator software is packaged as *iscsi-initiator-utils* on RHEL systems and *open-iscsi* on SLES.

Once installed the iSCSI initiator daemon can be configured by editing the files below */etc/iscsi*, and managed using the usual *chkconfig* and *service* commands. The service is named *iscsid* on RHEL systems and *open-iscsi* SLES,

Default iscsi initiator settings are recommended in Linux and these are described in */etc/iscsi/iscsid.conf*.

### Setting the initiator iSCSI qualified name

Specify the iSCSI initiator name by editing the file */etc/iscsi/initiatorname.iscsi*. The initiator name that you choose here must match the iSCSI name given to the SAN Volume Controller host object for this machine. An example is given below:

```
InitiatorName= iqn.1986-03.com.ibm:iscsihost.0
```

The iSCSI initiator service must be restarted following changes to the initiator name.

### iSCSI target discovery

Discovering iSCSI targets using Send Targets

Run the discovery command:

```
iscsiadm --mode discovery --type sendtargets --portal x.x.x.x
```

where *x.x.x.x* is the IP address of a node Ethernet port on the SAN Volume Controller cluster. This will return the IQN and target portals associated with the SAN Volume Controller node. For example:

```
x.x.x.x:3260,1 iqn.1986-03.com.ibm:2145.china5.hlnc111874
```

### Discovering iSCSI targets using iSNS

Before attempting iSNS discovery, you must complete the following tasks:

- Configure the cluster to use your iSNS server.
- Verify that the SAN Volume Controller nodes have registered their target portals with it.

Please follow below steps to do discovery using iSNS:

1. Modify the iSNS server settings in `/etc/iscsi/iscsid.conf` by entering the IP address and port number used by your iSNS server:  

```
isns.address = <iSNS server IP address>
isns.port = <iSNS server port>
```
2. Restart the iSCSI initiator service to pick up the iSNS configuration changes.
3. Run the iSNS discovery command:  

```
iscsiadm --mode discovery --type isns
```

which will output a list of all iSCSI target portals registered with the iSNS server.

## Logging in to discovered targets

The SAN Volume Controller supports only one iSCSI session between an initiator and a target so ensure that you do not attempt to connect to the same target (SAN Volume Controller node) more than once.

Login the iSCSI initiator to your selected SAN Volume Controller targets using the command:

```
iscsiadm --mode node --target <IQN> --portal x.x.x.x --login
```

For example:

```
iscsiadm --mode node --target iqn.1986-03.com.ibm:2145.china5.hlnc111874 --portal 9.71.43.131 --login
```

Once the login completes, the device nodes for the SAN Volume Controller VDisks mapped to the host will be created.

## Logging out from targets

To logout from a particular SAN Volume Controller target using the command:

```
iscsiadm --mode node --target <IQN> --portal x.x.x.x --logout
```

Or to logout from all established sessions:

```
iscsiadm --mode node --logoutall=all
```

Once you logout from an SAN Volume Controller target, the iSCSI VDisks it serves will no longer be available to the host

## Setting up Authentication for Linux Hosts

There are two CHAP authentication mechanisms supported by SAN Volume Controller:

- One way CHAP authentication (SAN Volume Controller authenticates the host iSCSI initiator)
- Two way CHAP authentication (both SAN Volume Controller and the initiator authenticate each other)

Both mechanisms are configured in the "CHAP Settings" section of `/etc/iscsi/iscsid.conf`, which should look like this example:

```
# *****
# CHAP Settings
# *****

# To enable CHAP authentication set node.session.auth.authmethod
# to CHAP. The default is None.
```



```

#node.session.auth.authmethod = CHAP

# To set a CHAP username and password for initiator
# authentication by the target(s), uncomment the following lines:
#node.session.auth.username = username
#node.session.auth.password = password
node.session.auth.username = iqn.suse.nmp.com
node.session.auth.password = xxxxxxxxxxxxxx
# To set a CHAP username and password for target(s)
# authentication by the initiator, uncomment the following lines:
#node.session.auth.username_in = username_in
#node.session.auth.password_in = password_in
node.session.auth.password_in = yyyyyyyyyyyyyy
# To enable CHAP authentication for a discovery session to the target
# set discovery.sendtargets.auth.authmethod to CHAP. The default is None.
#discovery.sendtargets.auth.authmethod = CHAP
discovery.sendtargets.auth.authmethod = CHAP
# To set a discovery session CHAP username and password for the initiator
# authentication by the target(s), uncomment the following lines:
#discovery.sendtargets.auth.username = username
#discovery.sendtargets.auth.password = password

# To set a discovery session CHAP username and password for target(s)
# authentication by the initiator, uncomment the following lines:
#discovery.sendtargets.auth.username_in = username_in
#discovery.sendtargets.auth.password_in = password_in

```

1. Setting up one way authentication:

- a. Please set the username to your Initiator name it will look like this
  - i. node.session.auth.authmethod = CHAP
  - ii. node.session.auth.username = <initiator IQN name>
  - iii. node.session.auth.password = < CHAP secret for host >
- b. Please set the username to your Initiator name it will look like this
  - i. discovery.sendtargets.auth.authmethod = CHAP
  - ii. discovery.sendtargets.auth.username = <initiator IQN name>
  - iii. discovery.sendtargets.auth.password = <CHAP secret for host>

c. Save this setting.

Note: In the above example xxxxxxxxxxxxxx is the CHAP secret for host and iqn.suse.nmp.com is the IQN name of the initiator. This IQN must be same as used while create a host object in SAN Volume Controller using svctask mkhost command If you want to configure two way authentication open up same file and follow the instructions.

2. Setting up two way authentication: It is not mandatory to setup two way authentication (Initiator also authenticates target), it is also possible to just have one way authentication. However if you decide to use two way authentication, then follow these instructions

- a. Edit the password\_in to CHAP secret which you have set up using svctask chcluster command (refer the Command Line Interface User's Guide) on cluster. It will look like this.
  - i. node.session.auth.password\_in = < CHAP secret for cluster >
  - ii. discovery.sendtargets.auth.password\_in = <CHAP secret for cluster>
- b. Save this setting.

Before attempting to configure two-way CHAP authentication, ensure that you already have one-way authentication configured and working for your host. Then you can configure two-way CHAP authentication. The CHAP secrets selected for one and two way authentication must be different. In the above example xxxxxxxxxxxx is the host chap secret set using svctask chhost, while yyyyyyyyyyyy is the chap secret set using chcluster command.

Note: 1) Both the Secret (for one way and two ways) can not be same.

## Recommended iSCSI configuration for the SAN Volume Controller

The disk timeout is set in /sys/block/<device>/device/timeout file. The default value of 60 seconds is recommended for SAN Volume Controller in most situations.

## Multipathing on Linux

After the iSCSI devices are discovered, please do the following steps to enable Multipathing.

SAN Volume Controller supports the Linux device-mapper multipathing driver for iSCSI host attachment.

You can enable multipathing for the devices by following these steps:

1. Stop the multipathing daemon (e.g. service multipathd stop).
2. Edit the multipath configuration file and add an entry for the IBM SAN Volume Controller to the *devices* section:  
vi /etc/multipath.conf

```
devices {
    device {
        vendor          "IBM"
        product         "2145"
        path_grouping_policy group_by_prio
        prio_callout    "/sbin/mpath_prio_alua /dev/%n"
        features        "1 queue_if_no_path"
    }
}
```

Prevent blacklisting of the IBM 2145 devices by either disabling blacklisting of devices entirely, or adding an appropriate entry in the *blacklist\_exceptions* section.

3. Save the file.
4. Restart the multipathing daemon (e.g. service multipathd start)

**Note:** If you are using SUSE Linux Enterprise Server 10 Service Pack 2 or later, the line:

```
prio_callout "/sbin/mpath_prio_alua /dev/%n"
```

should be replaced with:

```
prio "alua"
```

to disable the Using deprecated prio\_callout message.

## Other useful commands

1. multipath -ll :shows the current multipath topology from all available information (sysfs, the

- device mapper, path checkers ). When a new device is added to system which is running multi-pathing. You need to run multipath -v2 command so that the device can be monitored by the multi-pathing software
2. multipathd -k : will go into interactive mode. Various command can be used from listing the Current configuration to change the configuration. like **<show config>** will list the current configuration and **<reconfigure>** will get the latest configuration from config file (etc/multipath.conf) .

## References

Man pages : multipath, multipathd

## • Windows

This chapter will guide you through the steps required to attach a Windows host to the IBM SAN Volume Controller using iSCSI. Before you begin, ensure that you have followed the steps in Chapter 23 to configure your SAN Volume Controller cluster for iSCSI host attachment.

## Obtaining and installing the initiator software

If you are running Windows Server 2008 you should find that the Microsoft iSCSI software initiator is pre-installed. For Windows Server 2003 you will need to download the initiator software from the Microsoft download centre: <http://www.microsoft.com/downloads/en/default.aspx>.

When you run the installation program for the initiator, you will be presented with a list of components to install:

- Initiator Service
- Software Initiator
- Microsoft MPIO Multi-pathing Support for iSCSI

The first two options are required so make sure they are selected. MPIO support is useful if you have a redundant IP network configuration and want to protect your host I/O against network hardware failures.

NOTE: MPIO support is not required to support online cluster maintenance such as software upgrades due to IP fail-over between nodes in the cluster I/O groups.

Once the installation has completed you will be prompted to restart the computer, do this now before continuing with these instructions.

The iSCSI initiator can now be configured from the *iSCSI Initiator* control panel module.

## Setting the initiator name (IQN)

The iSCSI initiator name (IQN) can be viewed and altered from the **General** tab of the iSCSI Initiator control panel module on Windows Server 2003, or from the **Configuration** tab on Windows Server 2008.

The initiator name you choose here *must* match the iSCSI name for the SAN Volume Controller host object you have created for this machine. Refer to the SAN Volume Controller Software Installation and Configuration guide if you are uncertain how to configure the iSCSI name for a host.

## Target discovery

There are two supported methods for iSCSI target portal discovery: iSNS and Send Targets discovery. The two methods are both configured from the **Discovery** tab of the **iSCSI Initiator** control panel module.

- Discovery using Send Targets

This form of discovery requires you to enter the address of one or more of your SAN Volume Controller cluster node Ethernet ports. For each node that you wish to perform discovery on:

1. Click the **Add (Discover Portal on Windows Server 2008)** button associated with the list of target portals.
2. Enter the IP address or DNS name of the node Ethernet port to perform discovery on. Leave the IP port number at the default value of 3260. If CHAP authentication is configured for this host on the SAN Volume Controller cluster then refer to section **#ref Authentication for Windows hosts** at this point.
3. Click **OK** and the SAN Volume Controller node port address will be added to the list of target discovery portals.

- Discovery using iSNS

If you want to use iSNS discovery you must first have configured the cluster to use your iSNS server, and verify that the SAN Volume Controller nodes have registered their target portals with it.

Click the **Add** button associated with the list of iSNS servers and enter the IP address or DNS hostname of your iSNS server.

## Connecting to discovered targets

Open the **Targets** tab of the iSCSI Initiator control panel module. The list of discovered targets should have an entry for each SAN Volume Controller node you performed discovery on.

1. Select the node to connect to from the list and click on **Log On... (Connect on Windows Server 2008)**. The Connect to Target window allows you to select whether this connection should be restored automatically at boot, and whether to enable multipathing for this iSCSI target.
2. If you wish to connect to a specific portal on the target (e.g. for load balancing between the node Ethernet ports) or to configure CHAP authentication (see section **#ref**) you can access these settings by clicking on the **Advanced...** button.
3. Click **OK** and you should see the the selected targets status change from *Inactive* to *Connected*.
4. Select the node to disconnect from the list and click on **Log Off**. This will stop I/Os to iscsi disks. Use this option to log off from the iscsi target.

NOTE: SAN Volume Controller only supports a single iSCSI session between an initiator and a target so make sure you do not attempt to connect to the same target (SAN Volume Controller node) more than once.

## Viewing and managing the discovered disks

The SAN Volume Controller VDisks mapped to the iSCSI host should now be visible to Windows disk management services and can be initialised, formatted and mounted.

If you want to ensure that the iSCSI devices are available before dependant applications or services are started, you can *bind* these devices to the **Microsoft iSCSI Initiator Service** from the **Bound Volumes/Devices** tab on the iSCSI Initiator control panel module.

## The Windows Disk Initialization wizard

An iSCSI-based volume follows the same rules as any other Windows volume. You can create this volume as basic or dynamic (although dynamic is *not* recommended for iSCSI) or even as GPT (GUID partition table) volumes, which support volumes in excess of 2TB. Just as is the case with any Windows volume, you need to initialize the new drive, create a partition, and format the new volume.

## Optimizing the initiator for SAN Volume Controller

For reliable iSCSI operation during cluster maintenance the following changes to the system registry should be made:

Registry Key	Type	Value (decimal)
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E97B-E325-11CE-BFC1-08002BE10318}\<bus id>\Parameters\LinkDownTime	DWORD	120
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E97B-E325-11CE-BFC1-08002BE10318}\<bus id>\Parameters\MaxRequestHoldTime	DWORD	120
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Disk\TimeOutValue	DWORD	60

You will need to restart the computer once these changes have been made for them to take effect.

## Setting up Authentication for Windows Hosts

There are two CHAP authentication mechanisms available .

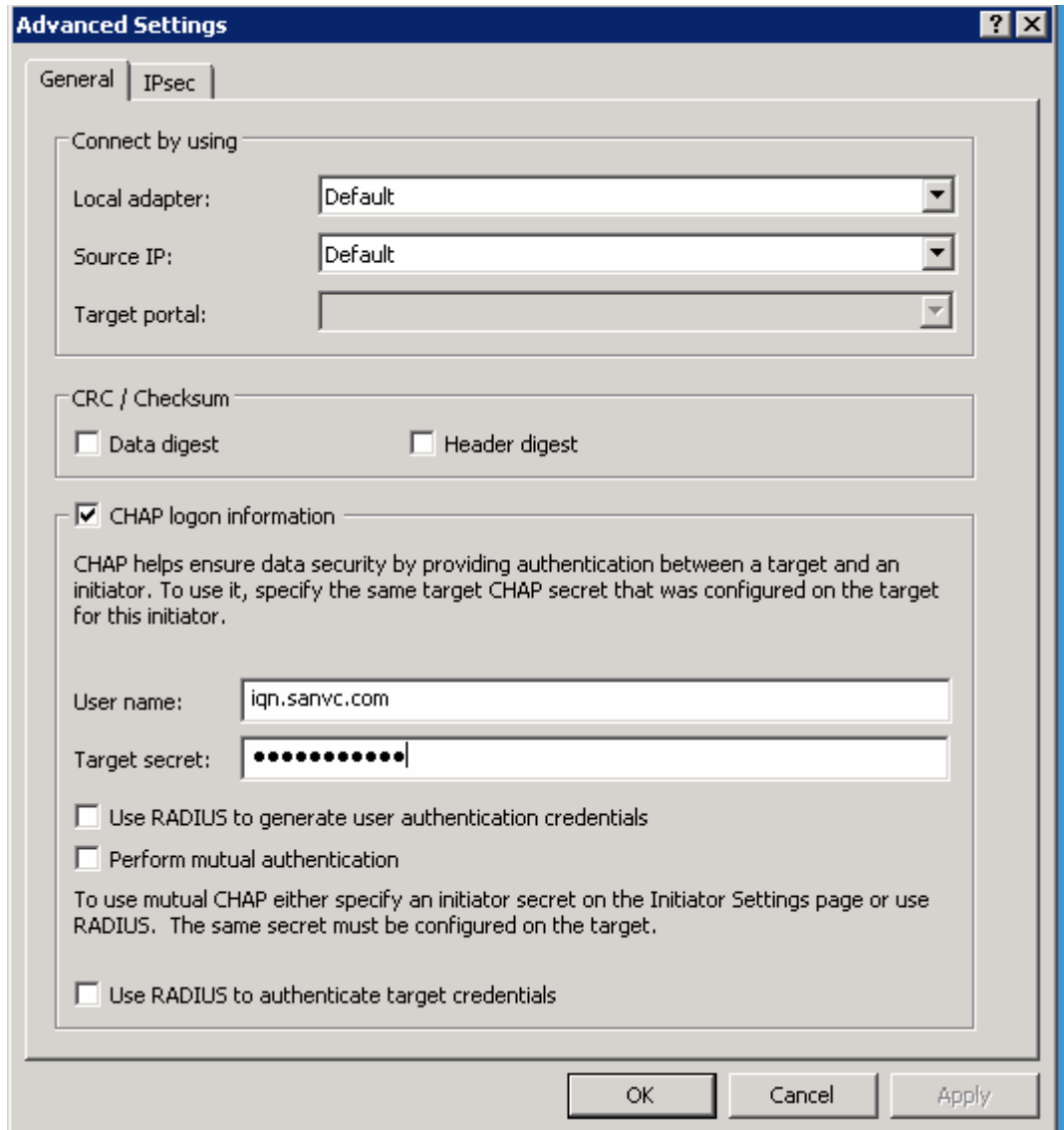
- **1-way authentication:** where SAN Volume Controller authenticates the host initiator
- **2-way authentication:** where both SAN Volume Controller and the host initiator authenticate each other.

## Setting up one way authentication

The following settings can only be configured while adding new discovery targets, or when logging on to a discovered target portal. You should logout from any existing iSCSI sessions and remove any existing discovery targets associated with the SAN Volume Controller before continuing.

### 1) Authentication for send targets discovery sessions:

Follow the advice in "Discovery using Send Targets" earlier in the chapter, and click on the Advanced... button of the "Add Target Portal" window.



**Figure 15- 1 way authentication (target authenticate Initiator, Discovery Session)**

- a. Tick the "CHAP logon information" checkbox. Enter the hosts iSCSI initiator IQN into the "User name" field. Enter the hosts CHAP secret into "Target secret" field.
- b. Click on OK button.

Note:

- 1) The CHAP secret must be at least 12 characters in length.  
This secret is the same that has been set with the `chost` command on the SAN Volume Controller cluster for this host.

## 2) Authentication for normal iSCSI sessions:

**While following the advice in section "Connecting to discovered targets" earlier in this chapter, click on the Advanced... button on the "Connect to target" window.**

- a. Repeat the steps you for configuring CHAP authentication for discovery sessions, entering the same CHAP username and password.
- b. Click the OK button.

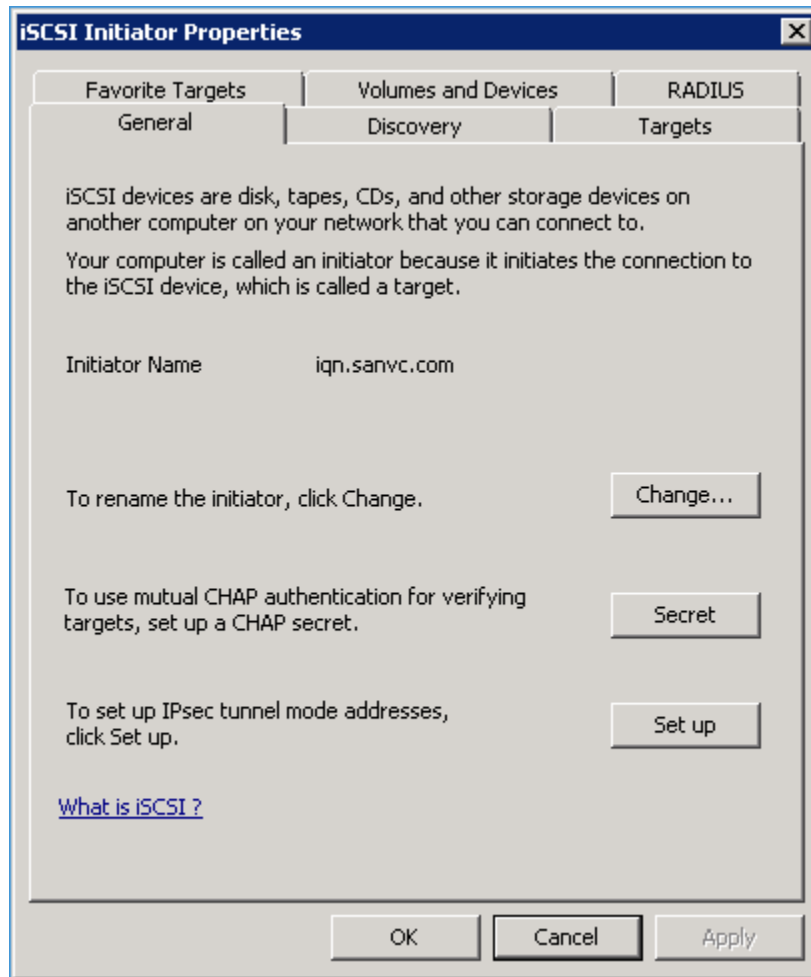
### Setting up two way authentication:

Two way iSCSI authentication depends allows the iSCSI initiator and target to authenticate each other. Before following the advice in this section you should have:

- Verified you have a working one way authentication by following the advice in the previous section.
- Configured a cluster iSCSI CHAP secret on the SAN Volume Controller (see the SAN Volume Controller Installation and Configuration Guide for details on configuring the cluster CHAP secret).

Now follow these steps to setup two way authentication:

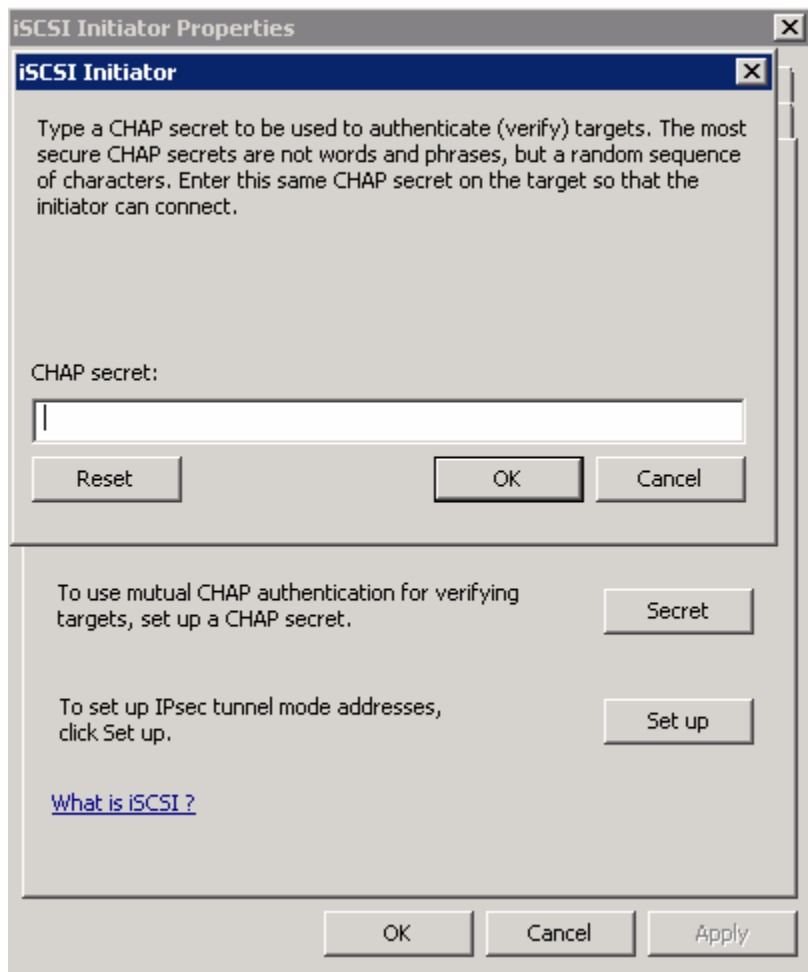
- a) Open the General tab of iSCSI initiator properties control panel (see Figure 17)



**Figure 17- 2 way authentication**

- b) Click on the Secret button.





**Figure 18- setting up CHAP secret (2 way authentication)**

- c) Enter the SAN Volume Controller cluster CHAP secret where shown in Figure 18 and click OK.
- d) Now you must repeat the instructions in the previous section for one way authentication, but this time ensure that the "Perform mutual authentication" checkbox is ticked on the "Advanced Settings" window for discovery and login sessions as shown in figure19.

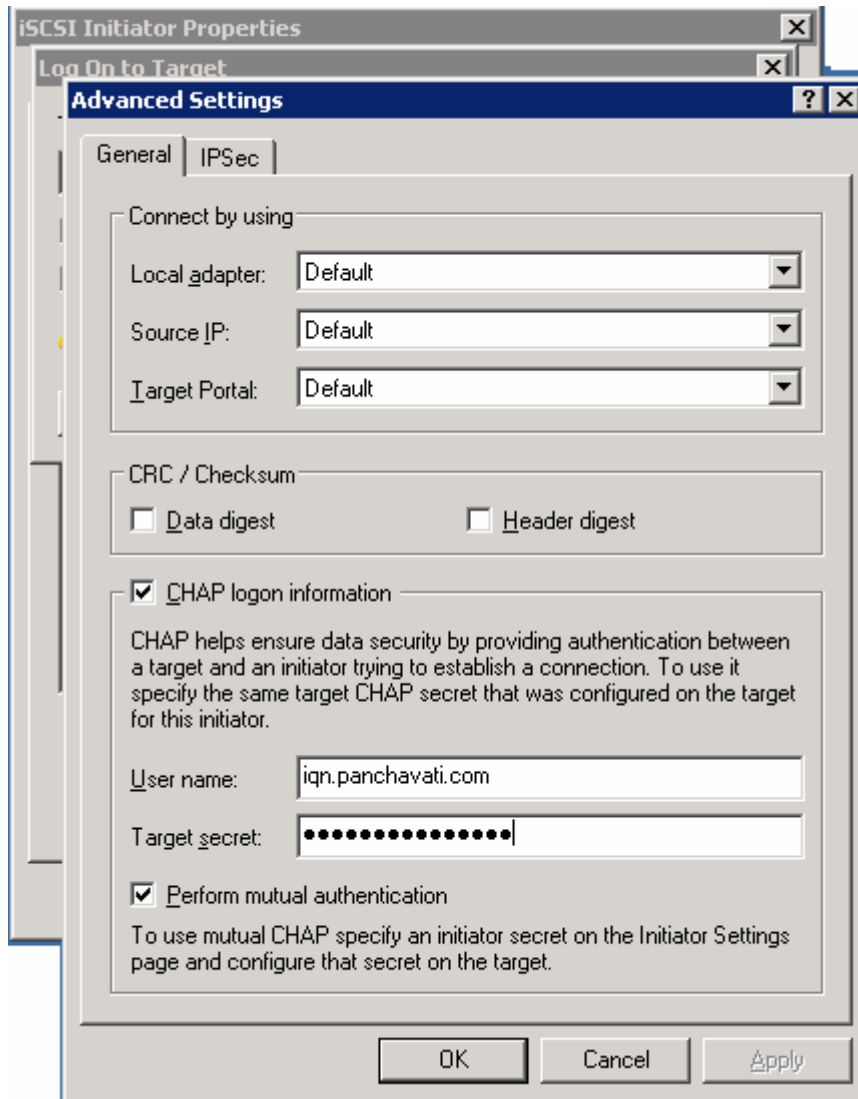


Figure 19 - Selecting two way (mutual) authentication

- Note: 1) These settings apply both to discovery and normal sessions.  
2) The CHAP secrets must be at least 12 characters in length.  
3) The CHAP secrets for the SAN Volume Controller cluster and the host initiator cannot be the same.

## AIX

### Installing the AIX iSCSI software initiator

- Install the AIX iSCSI Software Initiator on your AIX host if it is not already installed. To check if the fileset **devices.iscsi\_sw.rte** is installed, use the `lspp -l` command.
- Ensure that you have the ODM stanza updates for your platform installed. See section "ODM stanzas for SAN Volume Controller iSCSI devices" later in this chapter for details. ODM stanzas for SAN Volume Controller iSCSI devices

### Configuring the AIX iSCSI software initiator

1. The software initiator can be configured using SMIT as follows: Navigate to Devices -> *iSCSI* -> *iSCSI Protocol Device*.
2. Select Change / Show Characteristics of an iSCSI Protocol Device and select the device associated with the software iSCSI initiator (e.g. `iscsi0`). Check the *iSCSI* Initiator Name field. It must match the iSCSI name that has been set for the SAN Volume Controller host object associated with this host.

Note: A default initiator name is assigned when the software is installed. This initiator name can be changed by the user to match local network naming conventions.

3. The Maximum Targets Allowed field corresponds to the maximum number of iSCSI targets that can be configured. If you reduce this number, you also reduce the amount of network memory pre-allocated for the iSCSI protocol driver during configuration.

### iSCSI target configuration and LUN discovery

After the software initiator is configured, do the following:

1. Edit the `/etc/iscsi/targets` file and add the iSCSI target portal details for all SAN Volume Controller nodes the host will be required to logon to

Each uncommented line in the file represents an iSCSI target, and is formatted like the example given below:

```
192.168.1.7      3260      iqn.1986-03.com.ibm:2145.sahyadri.node1
```

2. After editing the `/etc/iscsi/targets` file, type the following command:

```
cfgmgr -v -l iscsi0
```

This command causes the driver to attempt to logon to each of the targets listed in the `/etc/iscsi/targets` file, and to define a new hdisk for each LUN on the targets that are found.

You can review the discovered hdisks with the command `lsdev -c disk`. The SAN

Volume Controller VDisks should appear in the output as:

```
hdisk2    Available                IBM 2145 iSCSI Disk Drive
```

Note: If the SAN Volume Controller VDisks are not discovered, review the configuration of the initiator, the target, and any iSCSI gateways to ensure correctness, and then rerun the `cfgmgr` command.

### Configuring iSCSI hdisk timeouts for SAN Volume Controller

The recommended value of the `rw_timeout` attribute for SAN Volume Controller iSCSI hdisks is 60 seconds.

You can set it using the `chdev` command as shown for each iSCSI device, for example:

```
chdev -l hdisk10 -a rw_timeout=60
```

The changes can be verified using the `lsattr` command, for example:

```
lsattr -E -l hdisk10 -a rw_timeout
```

## Setting up Authentication for AIX Hosts

Although the SAN Volume Controller support both one and two way CHAP authentication for iSCSI, the AIX software initiator currently only supports one way CHAP (where SAN Volume Controller target authenticates the initiator).

The CHAP settings are defined in the `/etc/iscsi/targets` file on the host. The AIX initiator or HBA always uses its iSCSI initiator IQN as the CHAP user name which is what SAN Volume Controller expects.

1. Open the `/etc/iscsi/targets` file for editing.  
For each line containing an SAN Volume Controller target definition, append the initiators CHAP secret in double quotes, e.g.:

```
192.168.1.7    3260    iqn.1986-03.com.ibm:2145.sahyadri.node1 "secret"
```

The CHAP secret set here value must match the value configured on the SAN Volume Controller cluster for the host object associated with this host. As SAN Volume Controller authenticates on a per-initiator basis, the CHAP secret will be the same for all SAN Volume Controller targets on a particular cluster.

An example `/etc/iscsi/targets` file is shown in the figure below:

```
#
# ChapSecret    = %x22 *( any character ) %x22
#              ; "
#              ; ChapSecret is a string enclosed in double quotes. The
#              ; quotes are required, but are not part of the secret.
#
# EXAMPLE 1: iSCSI Target without CHAP(MD5) authentication
```

```

# Assume the target is at address 192.168.3.2,
# the valid port is 5003
# the name of the target is iqn.com.ibm-4125-23WTT26
# The target line would look like:
# 192.168.3.2 5003 iqn.com.ibm-4125-23WWT26
#
# EXAMPLE 2: iSCSI Target with CHAP(MD5) authentication
# Assume the target is at address 10.2.1.105
# the valid port is 3260
# the name of the target is iqn.com.ibm-K167-42.fc1a
# the CHAP secret is "This is my password."
# The target line would look like:
# 10.2.1.105 3260 iqn.com.ibm-K167-42.fc1a "This is my password."
#
# EXAMPLE 3: iSCSI Target with CHAP(MD5) authentication and line continuation
# Assume the target is at address 10.2.1.106
# the valid port is 3260
# the name of the target is iqn.2003-01.com.ibm:00.fcd0ab21.shark128
# the CHAP secret is "123ismysecretpassword.fc1b"
# The target line would look like:
# 10.2.1.106 3260 iqn.2003-01.com.ibm:00.fcd0ab21.shark128

192.168.1.41 3260 iqn.1986-03.com.ibm:2145.pahar.dvt110702
192.168.1.43 3260 iqn.1986-03.com.ibm:2145.moscow.dvt110706 "svcchapsecret"

```

The two targets in this example are members of different SAN Volume Controller clusters, one of which is configured to authenticate the initiator, the other is not:

Target "iqn.1986-03.com.ibm:2145.pahar.dvt110702" is not configured to have authentication so the chapsecret field for it is kept blank.

Whereas for the target "iqn.1986-03.com.ibm:2145.moscow.dvt110706" we have added the chapsecret as svcchapsecret for authentication.

## ODM stanzas for SAN Volume Controller iSCSI devices

The official PTFs are:

APAR_aix_53V = IZ56763	53V_PTFs = U824933 U821212
APAR_aix_53X = IZ57424	53X_PTFs = U827012 U827039
APAR_aix_61H = IZ57456	61H_PTFs = U821333 U827773
APAR_aix_61F = IZ57672	61F_PTFs = U828452 U828391
APAR_aix_61B = IZ57872	61B_PTFs = U828504 U828552
APAR_aix_61D = IZ58242	61D_PTFs = U828341 U828364
APAR_aix_610 = IZ58288	610_PTFs = U828629 U828620
APAR_aix_61J = IZ58592	61J_PTFs = U827222 U825727

This package is valid for all supported levels of AIX. When installed, AIX will give SAN Volume Controller VDisks the following attributes:

```
lsdev -C -l hdisk1
```

```
hdisk1      Available          IBM 2145 iSCSI Disk Drive
```

```
# lsattr -E -l hdisk1
clr_q      no          Device CLEARS its Queue on error True
host_addr  9.71.43.106  Hostname or IP Address      False
location   Location Label              True
lun_id     0x0          Logical Unit Number ID      False
max_transfer 0x40000     Maximum TRANSFER Size      True
port_num   0xcabc       PORT Number                  False
pvid       none         Physical volume identifier   False
q_err      yes          Use QERR bit                 True
q_type     simple       Queuing TYPE                 True
queue_depth 8           Queue DEPTH                  True
reassign_to 120         REASSIGN time out value     True
reserve_policy no_reserve  Reserve Policy              True
rw_timeout 60          READ/WRITE time out value   True
start_timeout 60        START unit time out value   True
target_name iqn.1986-03.com.ibm:2145.china6.hlcn111890 Target NAME      False
unique_id  352136005076801910296880000000000000000204214503IBMiscsi Unique device identifier False
```

```
lscfg -v -l hdisk1
```

```
hdisk1          IBM 2145 iSCSI Disk Drive

Manufacturer.....IBM
Machine Type and Model.....2145
ROS Level and ID.....30303030
Serial Number.....
Device Specific.(Z0).....0000043268101002
Device Specific.(Z1).....
Device Specific.(Z2).....
Device Specific.(Z3).....
```

## Solaris

### Installing the Solaris iSCSI initiator

1. Use the following command to confirm if the initiator service is installed

```
pkginfo SUNWiscsiu SUNWiscsir
```

If it is not installed, follow the OS instructions for installing this package.

2. Use this ip address with the `iscsiadm add` command to indicate which discover address should be used with this host and svc cluster connection.

```
iscsiadm add discovery-address 192.168.1.97:3260
```

Repeat this command for all SAN Volume Controller clustered Ethernet ports that you are using.

3. Use the `iscsiadm add iSNS-server` command to locate the iSNS server, if you are using this facility, eg

```
iscsiadm add iSNS-server 10.0.0.1:3205
```

4. Enable the iSCSI target discovery method using the iscsiadm modify discovery command. For file based discovery this command is of the form:

```
iscsiadm modify discovery --sendtargets enable
```

For iSNS discovery, this command looks like:

```
iscsiadm modify discovery --iSNS enable
```

5. Create the iSCSI device links for the local system

```
devfsadm -i iscsi
```

6. Use the command iscsiadm remove to stop the I/Os from iscsi disks. Ex:  
iscsiadm remove discovery-address 192.168.1.22

## Inspecting the iSCSI configuration

1. To list the initiator parameters use the iscsiadm list initiator-node command, eg

```
iscsiadm list initiator-node
```

displays

```
Initiator node name: iqn.1986-03.com.ibm:01:0003bad935da.4906be64
```

```
Initiator node alias: -
```

```
Login Parameters (Default/Configured):
```

```
Header Digest: NONE/-
```

```
Data Digest: NONE/-
```

```
Authentication Type: NONE
```

```
RADIUS Server: NONE
```

```
RADIUS access: unknown
```

```
Configured Sessions: 1
```

2. To list the discovered targets and active sessions use the iscsiadm list target command, eg

```
iscsiadm list target
```

displays

```
Target: iqn.1986-03.com.ibm:2145.lodestoneGB1.node1
```

```
Alias: - TPGT: 1
```

```
ISID: 4000002a0000
```

```
Connections: 1
```

3. To list the target parameters use the iscsiadm list target command, eg

```
iscsiadm list target -v iqn.1986-03.com.ibm:2145.lodestoneGB1.node1 <your target name >
```

```
Target: iqn.1986-03.com.ibm:2145.lodestoneGB1.node1
```

Alias: - TPGT: 1  
ISID: 400002a0000  
Connections: 1  
CID: 0  
IP address (Local): 192.168.1.93:32828  
IP address (Peer): 192.168.1.97:3260  
Discovery Method: SendTargets  
Login Parameters (Negotiated):  
Data Sequence In Order: yes  
Data PDU In Order: yes  
Default Time To Retain: 20  
Default Time To Wait: 2  
Error Recovery Level: 0  
First Burst Length: 32768  
Immediate Data: no  
Initial Ready To Transfer (R2T): yes  
Max Burst Length: 32768  
Max Outstanding R2T: 1  
Max Receive Data Segment Length: 8192  
Max Connections: 1  
Header Digest: NONE  
Data Digest: NONE

4. To list the target parameters use the `iscsiadm list target-param` command, eg

```
iscsiadm list target-param -v iqn.1986-03.com.ibm:2145.lodestoneGB1.node1
```

displays

```
Target: iqn.1986-03.com.ibm:2145.lodestoneGB1.node1
```

```
Alias: -  
Bi-directional Authentication: disabled  
Authentication Type: NONE  
Login Parameters (Default/Configured):  
Data Sequence In Order: yes/-  
Data PDU In Order: yes/-  
Default Time To Retain: 20/-  
Default Time To Wait: 2/-  
Error Recovery Level: 0/-  
First Burst Length: 65536/-  
Immediate Data: yes/-  
Initial Ready To Transfer (R2T): yes/-  
Max Burst Length: 262144/-  
Max Outstanding R2T: 1/-  
Max Receive Data Segment Length: 8192/-  
Max Connections: 1/-  
Header Digest: NONE/-  
Data Digest: NONE/-  
Configured Sessions: 1
```

5. To remove a discovered target use the `iscsiadm remove discovery-address` command, eg

```
iscsiadm remove discovery-address 192.168.1.97:3260
```



6. To modify the parameter of the iSCSI initiator

For example, set the header digest to CRC32.

```
iscsiadm modify initiator-node -h CRC32
```

To set data digest to CRC32

```
iscsiadm modify initiator-node -d CRC32
```

Please verify that the parameter was modified

Using < iscsiadm list initiator-node >

### Special note on Header Digest and Data Digest

When you are discovering target do not set Header Digest and Data Digest to CRC32.as it is not supported in the discovery phase. It should be set to none and if the digest fields have some other value please use the following command to unset it.

A) iscsiadm modify initiator-node -h none -d none

B) After connection has been set up with target, the digest fields can be changed to CRC32 with command described in above section.

C) Display the updated parameter information for the iSCSI target device, for example:

```
iscsiadm list target-param -v iqn.1992-08.com.svcstorage:sn.84186266
```

```
Target: iqn.1992-08.com.svcstorage:sn.84186266
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
Login Parameters (Default/Configured):
  Data Sequence In Order: yes/-
  Data PDU In Order: yes/-
  Default Time To Retain: 20/-
  Default Time To Wait: 2/-
  Error Recovery Level: 0/-
  First Burst Length: 65536/-
  Immediate Data: yes/-
  Initial Ready To Transfer (R2T): yes/-
  Max Burst Length: 262144/-
  Max Outstanding R2T: 1/-
  Max Receive Data Segment Length: 65536/-
  Max Connections: 1/-
  Header Digest: CRC32/-
  Data Digest: CRC32/-
Configured Sessions: 1
```

D) Initiator reconnects with the target after you modify the parameter.

Verify that the iSCSI initiator has reconnected to the iSCSI target. For example:

```
iscsiadm list target -v iqn.1992-08.com.svcstorage:sn.84186266
Target: iqn.1992-08.com.svcstorage:sn.84186266
  TPGT: 2
```

```

ISID: 4000002a0000
Connections: 1
  CID: 0
    IP address (Local): nnn.nn.nn.nnn:64369
    IP address (Peer): nnn.nn.nn.nnn:3260
    Discovery Method: SendTargets
    Login Parameters (Negotiated):
      .
      .
      .
    Header Digest: CRC32
    Data Digest: CRC32

```

## Other considerations

1. You can see disks in Solaris using `<format >utility`.
2. All disks need to be labelled before running `i/o` on them, labelling can be done using `format utility`
3. Solaris use slice and partition terminology. `s2` slice represent the whole disks. you may use other slice (from `s0` to `s7`) if you have done partitioning . Without doing partitioning you can use `s2` slice.
4. Change the default IO timeout used by Solaris SCSI stack. The **`sd_io_time`** variable is the limiter on how long an I/O can be outstanding before an error condition is returned. The Solaris default is 60 seconds (`0x3c`), but this is often set to 31 seconds (`0x1f`). The default timeout can be changed using the following steps :
  - a. Open the file `/etc/system`  
Add this line at the very end:  
`set sd:sd_io_time = 0x3c`
  - b. Reboot the machine.

## Multipathing on Solaris

The `scsi_vhci` driver is a SCSI compliant pseudo nexus driver that supports Solaris operating system I/O multi-pathing services for SCSI-3 devices. By default multi-pathing is enable system-wide basis. You can disable it by using the mechanism provides by HBA driver.

In case of iSCSI , the related configuration file is `/kernel/drv/iscsi.conf` in which there is configuration variable called `mpxio-disable`.

## Steps to enable Multipathing

1. open `/kernel/drv/iscsi.conf`
2. set the `mpxio-disable="no"`
3. For each candidate SCSI target device, the `scsi_vhci` code must identify a failover module to support the device. If a failover module can't be identified, the device will not function under `scsi_vhci(7D)` multipathing control. a vendor/product specific override mechanism is available. This `scsi_vhci.conf` base mechanism can be used to direct a device to a specific failover module. To add a third-party (non-Sun) symmetric storage device to run under `scsi_vhci` (and thereby take advantage of `scsi_vhci` multipathing), you add the vendor ID and product ID for the device, as those strings are returned by the SCSI Inquiry command. for our case the vendor ID is `<IBM>` and product ID is

<2145> please add following lines in **/kernel/drv/scsi\_vhci.conf**

```
device-type-scsi-options-list ="IBM 2145", "symmetric-option";  
symmetric-option = 0x1000000;
```

and save the file

Note: you may just find and commented entry for SUN, you can just replace SUN with IBM and SENA with 2145 and uncommented those lines.

4. reboot the system using `reboot --- -r` . here `-r` option is a boot argument which is used for Reconfiguration boot. The system will probe all attached hardware devices and configure the logical namespace in `/dev`.
5. you will find device in this form when use `format` command.

```
c2t6005076801A9027E6000000000000006d0 <IBM-2145-0000 cyl 10238 alt 2 hd 32 sec 64>  
/scsi_vhci/ssd@g6005076801a9027e6000000000000006
```

## Steps to disable Multipathing

1. open **/kernel/drv/iscsi.conf**
2. set the `mpxio-disable="yes"`
3. reboot the system using `reboot -- -r` . here `-r` option is a boot argument which is used for Reconfiguration boot. The system will probe all attached hardware devices and configure the logical namespace in `/dev`.
4. After disabling multi-ipathing if you have two path for same LUN then it looks like  
When use `format` command.

```
c5t3d0 <IBM-2145-0000 cyl 5118 alt 2 hd 32 sec 64>  
/iscsi/disk@0000iqn.1986-03.com.ibm%3A2145.nilgir.dvt1105980001,0  
c5t4d0 <IBM-2145-0000 cyl 5118 alt 2 hd 32 sec 64>  
/iscsi/disk@0000iqn.1986-03.com.ibm%3A2145.nilgir.dvt1109120001,0
```

You can see that targets are different in both path but they refer to same LUN, This is determined by the 'd0' at the end of the device name

Note: The SCSI inquiry mechanism is available under `format` command. In case of iSCSI you can not enable/disable multipathing per device basis.

## Setting up Authentication for Solaris Hosts

- 1) One way chap authentication can be enabled/disabled done using command:  
`iscsiadm modify initiator-node --authentication <CHAP/none>`  
`iscsiadm modify initiator-node --CHAP-secret`
- 2) Two way chap authentication can be enabled/disabled done using command:  
`iscsiadm modify target-param --authentication <CHAP/none>`  
`iscsiadm modify target-param --bi-directional-authentication <enable/disable>`  
`iscsiadm modify target-param --CHAP-secret`

For more information on this refer to the Solaris Initiator manual.

Note:

- 1) One way chap authentication means target authenticates the initiator

- 2) Two way chap authentication means initiator also authenticates the target.
- 3) The chap secret is of maximum of 16 characters and minimum of 12 characters.

## References

man scsi\_vhci, man iscsiadm, man iscsi,  
<http://docs.sun.com/app/docs/doc/817-5093/fmvcd?a=view>.

## HP-UX

### Installing the HP-UX iSCSI initiator

1) Login as root.

2) Download iSCSI software initiator from  
<http://h20293.www2.hp.com/portal/swdepot/displayProductInfo.do?productNumber=T1452A>

After you have downloaded the iSCSI-00\_B.11.31.01\_HP-UX\_B.11.31\_IA+PA.depot file, move (mv) it to the /tmp directory on your system.

3) Verify the depot has downloaded correctly using swlist.

```
# swlist -d @ /tmp/iSCSI-00_B.11.31.01_HP-UX_B.11.31_IA+PA.depot
```

```
Initializing...  
Contacting target "myhost"...
```

```
Target: myhost:/tmp/iSCSI-00_B.11.31.01_HP-UX_B.11.31_IA+PA.depot
```

```
Bundle(s):
```

```
iSCSI-00 B.11.31.01 HP-UX iSCSI Software Initiator
```

4) On a stand-alone system, run swinstall to install the product:

```
# swinstall -x autoreboot=true -s /tmp/iSCSI-00_B.11.31.01_HP-UX_B.11.31_IA+PA.depot iSCSI-00
```

Note: The HP-UX iSCSI Software Initiator is a kernel product. The autoreboot=true option will cause a system reboot after installation is complete.

### Configuring the HP-UX iSCSI initiator :-

1. Add the path for iscsiutil and other iSCSI executables to the root path as:

```
# PATH=$PATH:/opt/iscsi/bin
```

Instead of updating PATH environment variable manually each time, you could put this in /.profile file.

To display the iSCSI initiator name that has been configured

```
#iscsiutil -l
Initiator Name      : iqn.2001-04.com.hp.stor:svcio
Initiator Alias     :
Authentication Method : None
CHAP Method         : CHAP_UNI
Initiator CHAP Name :
CHAP Secret         :
NAS Hostname        :
NAS Secret          :
Radius Server Hostname :
Header Digest       : None,CRC32C (default)
Data Digest         : None,CRC32C (default)
SLP Scope list for iSLPD :
```

2. To change iSCSI initiator name  
iscsiutil -i -N <initiator name in iqn or eui format>  
You could confirm the initiator name change by command iscsiutil -l.

**Note:** HP needs standard iSCSI naming convention to be followed, so please make sure that initiator name chosen is standard compliant, otherwise it is observed, that disks doesn't get visible on HP host from SAN Volume Controller.

3. Adding a discovery targets

```
iscsiutil -a -I <IP>
```

Eg : # iscsiutil -a -I 192.168.1.149

Target address "192.168.1.149:3260,1" has been successfully added.

4. Note - HP-UX iSCSI software initiator does not support IPv6 address, so do not configure IPv6 address as a target IP.
5. To display the discovery target(s) that have been configured, enter:

```
iscsiutil -p -D
```

Eg. : # iscsiutil -pD

This will display information as follows:

Discovery Target Information

-----

Target # 1

-----

```
IP Address          : 192.168.1.149
iSCSI TCP Port      : 3260
iSCSI Portal Group Tag : 1
```

User Configured:

```
-----  
Authentication Method : None  
CHAP Method           : CHAP_UNI  
Initiator CHAP Name   :  
CHAP Secret           :  
Header Digest         : None,CRC32C (default)  
Data Digest           : None,CRC32C (default)
```

6. To discover the operational target devices and create device special files, enter:

```
# /usr/sbin/ioscan -NH 64000
```

```
H/W Path   Class           Description  
=====
```

64000/0x0	usbmsvbus	USB Mass Storage
64000/0x0/0x0	escsi_ctrl	USB Mass Storage Virt Ctlr
64000/0x0/0x0.0x0	tgtpath	usb target served by usb_ms_scsi driver
64000/0x0/0x0.0x0.0x0	lunpath	LUN path for disk9
64000/0x2	iscsi	iSCSI Virtual Root
64000/0x2/0x0	escsi_ctrl	iSCSI Virtual Controller
64000/0x2/0x0.0x10	tgtpath	iscsi target served by isvctlr driver
64000/0x2/0x0.0x10.0x0	lunpath	LUN path for disk586
64000/0xfa00	esvroot	Escsi virtual root
64000/0xfa00/0x0	disk	HP DG146BABCF
64000/0xfa00/0x1	disk	HP DG146BABCF
64000/0xfa00/0x2	disk	HP DG146BABCF
64000/0xfa00/0x3	disk	HP DG146BABCF
64000/0xfa00/0x4	disk	TEAC DVD-ROM DW-224EV
64000/0xfa00/0xae	disk	IBM 2145

7. To display operational targets, enter:

```
# iscsiutil -pO
```

```
Operational Target Information
```

```
-----  
Target # 1
```

```
-----  
Target Name           : iqn.1986-03.com.ibm:2145.LodestHP99.node1  
Target Alias          :  
No. of Target Addresses : 1
```

```
Target Address # 1
```

```
-----  
IP Address             : 192.168.1.149  
iSCSI TCP Port        : 3260  
iSCSI Portal Group Tag : 1
```

```
User Configured:
```

```
-----  
Authentication Method : None  
CHAP Method           : CHAP_UNI
```

```

Initiator CHAP Name :
CHAP Secret :
Header Digest : None,CRC32C (default)
Data Digest : None,CRC32C (default)

```

- To see all the luns exported

```
# ioscan -kfnC disk
```

```

Class | H/W Path | Driver S/W State | H/W Type | Description
=====
====
disk 0 0/1/1/0.0.0.0 sdisk CLAIMED DEVICE HP DG146BABCF
      /dev/dsk/c0t0d0 /dev/rdisk/c0t0d0
disk 1 0/1/1/0.0.0.1 sdisk CLAIMED DEVICE HP DG146BABCF
      /dev/dsk/c0t1d0 /dev/rdisk/c0t1d0
disk 3 0/1/1/0.0.0.2 sdisk CLAIMED DEVICE HP DG146BABCF
      /dev/dsk/c0t2d0 /dev/rdisk/c0t2d0
disk 2 0/1/1/0.0.0.3 sdisk CLAIMED DEVICE HP DG146BABCF
disk 485 255/0/16.0.0.0 sdisk CLAIMED DEVICE IBM 2145
      /dev/dsk/c15t0d0 /dev/rdisk/c15t0d0
disk 8 255/1/0.0.0 sdisk CLAIMED DEVICE TEAC DVD-ROM DW-224EV
      /dev/dsk/c1t0d0 /dev/rdisk/c1t0d0

```

- Command to check disk information  
diskinfo -v <disk>

- To remove iSCSI disk out of service log-off from the HP iscsi initiator using command  
iscsiutil -d -l <ip>

## Limitations of HPUX iscsi software initiator

- At the time of writing (October 2009) the HP-UX iSCSI Software Initiator does not support IPv6 addresses. So do not configure IPv6 address as a target address. In discovery phase target sends all IPs to initiator and IPv6 address could lead to HPUX system crash.
- HP-UX iSCSI initiator tries to login on all the target IPs that have been found in discovery phase. In this case SAN Volume Controller iSCSI login toggles.

## HP-UX Native Multipathing:-

HP-UX native multi-pathing is statically linked with kernel and nothing special needs to be done to use multipathing. You just need to find out pseudo device (Persistent DSF or Persistent device specific file) that has been created by multipathing kernel module. If IO's are going through Persistent DSF then HP-UX native multipathing is being used otherwise it is not. For more information on multipathing refer to references section.

For Eg. Let's say only one LUN is exported to HPUX box and initiator logs into both targets then LUN would be accessible via two different paths. In this case HP-UX 11i v3, creates only one persistent DSF per LUN. For instance, the following persistent DSFs exist

```
# ioscan -kfnNC disk
```

Class	I	H/W Path	Driver S/W	State	H/W Type	Description
disk	4	64000/0xfa00/0x0 /dev/disk/disk4	esdisk	CLAIMED	DEVICE	HP DG146BABC
disk	5	64000/0xfa00/0x1 /dev/disk/disk5	esdisk	CLAIMED	DEVICE	HP DG146BABC
disk	6	64000/0xfa00/0x2 /dev/disk/disk6	esdisk	CLAIMED	DEVICE	HP DG146BABC
disk	7	64000/0xfa00/0x3 /dev/disk/disk7	esdisk	CLAIMED	DEVICE	HP DG146BABC
disk	9	64000/0xfa00/0x4 /dev/disk/disk9	esdisk	CLAIMED	DEVICE	TEAC DVD-ROM DW-224EV
<b>disk</b>	<b>586</b>	<b>64000/0xfa00/0xae</b> <b>/dev/disk/disk586</b>	<b>esdisk</b>	<b>CLAIMED</b>	<b>DEVICE</b>	<b>IBM 2145</b>

The ioscan command can query the mapping combinations between LUN and lunpaths using DSF.

*To which legacy DSF does /dev/rdisk/disk587 map?*

```
# ioscan -m dsf /dev/rdisk/disk586
Persistent DSF      Legacy DSF(s)
=====
/dev/rdisk/disk586 /dev/rdisk/c15t0d0
                  /dev/rdisk/c16t0d0
```

*To which persistent DSF does /dev/rdisk/c5t2d4 map?*

```
# ioscan -m dsf /dev/rdisk/c15t0d0
Persistent DSF      Legacy DSF(s)
=====
/dev/rdisk/disk586 /dev/rdisk/c15t0d0
```

## Additional consideration

For HP-UX FC host attachments, it is recommended to use 'hpux' host type while creating host object, if you want to use more than eight LUNs per SCSI target. However for iSCSI HP-UX host attachment, this is not the requirement, the host object can be created with 'default' host type. You can get more information on these host types in SAN Volume Controller Command line interface guide.

## References

- 1) For more information on HP-UX iSCSI software initiator configuration, Please refer to HP-UX iSCSI Software Initiator Support Guide <http://www.docs.hp.com/en/T1452-90012/index.html>
- 2) For Multipathing, Please refer to HP-UX 11i v3 Native Multi-Pathing for Mass Storage [http://docs.hp.com/en/native-multi-pathing/native\\_multipathing\\_wp\\_AR0709.pdf](http://docs.hp.com/en/native-multi-pathing/native_multipathing_wp_AR0709.pdf)
- 3) The Command Line Interface User's Guide. For information on commands supported by the SVC  
<http://www->



## Limiting queue depth in iSCSI network

If you are designing a configuration for iSCSI network, you must estimate the queue depth for each node in order to avoid application failures. The queue depth is the number of I/O operations that can be run in parallel on a device. If a SAN Volume Controller node reaches the maximum number of queued commands, it will start returning to host with error codes like Resource unavailable, many operating systems cannot recover if the situation persists for more than 15 seconds. This can result in one or more servers presenting errors to applications and application failures on the servers.

### Queue depth

The queue depth is the number of I/O operations that can be run in parallel on a device. Ensure that you configure the servers to limit the queue depth on all of the paths to the SAN Volume Controller disks in configurations that contain a large number of servers or virtual disks (VDisks). Note: You might have a number of servers in the configuration that are idle or do not initiate the calculated quantity of I/O operations. If so, you might not need to limit the queue depth.

### Calculating a queue depth limit

Several factors are considered in the formula for calculating the queue depth limit. The formula for queue depth calculation considers the following factors:

- The maximum number of queued commands is per node and there are two nodes in an I/O group. The system must continue to function when one of the nodes in an I/O group is not available. Thus, an I/O group is considered to have the same number of queued commands as a node. If a node fails, the number of paths to each disk is cut in half.
- If a virtual disk (VDisk) is mapped so that it can be seen by more than one server, then each of the servers can send commands to it.
- If a device driver times out of a command, it immediately reissues the command. The SAN Volume Controller will have both commands in its command queue.

### Homogeneous queue depth calculation

Ensure you are familiar with the homogeneous queue depth calculation. The homogeneous queues must meet the following statements:

1. The queued commands are shared among all paths rather than providing servers with additional resources.
2. The virtual disks (VDisks) are distributed evenly among the I/O groups in the cluster.

You can set the queue depth for each VDisk on the servers using the following calculation:

$$q = (n/(v * c)) * 0.75$$

where n = Allowed command pool per iogroup, this value is fixed for this release of SAN Volume Controller, the value is: 4000

v = number of vdisks per server for the particular iogroup under consideration

c = number of servers in a iogroup

So the equation becomes :  $q = (4000/(v * c)) * 0.75$

### Example

Consider the following example : A SAN Volume Controller cluster with 20 host servers (c = 20), each having 20 vdisks mapped (v = 20). For this configuration the value of queue depth would be

$q = (4000/(20 * 20)) * 0.75 = 7.5$  (or the lower bound value is 7).

If the value of derived queue depth exceeds the maximum allowed value on a particular host operating system, then the minimum of two should be used.

Following table summarizes some common host server configuration, considering 20 vdisks mapped to each host server.

<b>Number of iSCSI hosts per IO group</b>	< 20	20-40	64
<b>Recommended queue depth</b>	7 (or the max number allowed by the host OS whichever is smaller)	3 to 4 (or the max number allowed by the host OS whichever is smaller)	2

## Nonhomogeneous queue depth calculation

For nonhomogeneous queues, use the following calculation. Nonhomogeneous queues meet one of the following criteria:

1. One or more servers are allocated additional resources so that they can queue additional commands.
2. VDisks are not distributed evenly among the I/O groups in the cluster.

Set the queue depth for each VDisk on the servers using the following calculation. For each VDisk, consider each server to which that VDisk has a mapping in an iogroup. This gives a set of server/VDisk pairs. If the sum of the server and VDisk queue depth for all of the pairs is less than 4000, the server will not experience problems due to a full queue.

### Example

Consider the case where we have two iogroups (iogrp0, iogrp1), and 20 host servers, iogroup0 has 20 VDisks mapped to 20 host servers and iogroup1 has 40 VDisks mapped to the same 20 host servers. So going by the formula mentioned in Homogeneous queue depth calculation

For iogrp 0: v = 20 , c = 20

$q = (4000/20 * 20) * 0.75 = 7$  (considering the lower bound)

The total server-VDisk mapping = 20 \* 20 = 400, and queue depth for each vdisk is 7, So the sum of queue depth for all mapping is : 400 \* 7 = 2800, which is less than 4000, so it should not experience the problem of full queue.

For iogrp 1 : v = 40, c = 20

$q = (4000/40 * 20) * 0.75 = 3$  (considering the lower bound)

The total server-VDisk mapping =  $40 * 20 = 800$ , and queue depth for each vdisk is 3, So the sum of queue depth for all mapping is :  $800 * 3 = 2400$ , which is less than 4000, so it should not experience the problem of full queue.

## **Limiting the queue depth**

After you have calculated the queue depth limit, you must apply it. Each operating system has a way to limit the queue depth on a per virtual disk (VDisk) basis. Please follow instructions from respective host operating systems to change queue depth.