iSCSI Protocol Concepts and Implementation

Introduction

This white paper describes the concepts and implementation of the Internet draft proposal for the mapping of Small Computer Systems Interface (SCSI) commands, data, and status over TCP/IP networks. iSCSI is an SCSI transport protocol for mapping of block-oriented storage data over TCP/IP networks. This white paper covers standardization efforts along with the high-level mechanics of iSCSI, including naming, addressing and discovery, sessions and connections, and current issues such as performance and data integrity.

The iSCSI protocol enables universal access to storage devices and storage-area networks (SANs) over standard Ethernet-based TCP/IP networks. These networks may be dedicated networks or may be shared with traditional Ethernet applications. IP LAN/wide-area network (WAN) routers and switches can be used to extend the IP storage network to the wide area for applications such as synchronous and asynchronous remote disk copy or tape backup and restore. In the WAN environment, TCP will ensure data reliability, manage network congestion, and adapt retransmission strategies to WAN delays.

IP Ethernet network infrastructures provide major advantages for interconnection of servers to block-oriented storage devices. IP networks provide security, scalability, interoperability, network management, storage management, and is cost-effective.

IP network advantages:

- The availability of network protocols and middleware for the management, security, and quality of service (QoS).
- Skills developed in the design and management of IP local-area network (LAN) networks can be applied to native IP SANs. Trained and experienced IP networking staffs are available to install and operate these networks.
- Economies achieved from using a standard IP infrastructure, products, and service across the organization.
  – SANs are complex to design, lacking interoperability, and have high implementation costs.
- Gigabit Ethernet switches and routers have advanced capabilities including: ultra low error rates, flow control, link aggregation, and full duplex operation.
  – Transfer data at optimal data rates over LAN, WAN, and metropolitan-area networks (MANs).
- iSCSI is compatible with existing Ethernet and IP WAN infrastructures.
- iSCSI will coexist with other IP protocols on a network infrastructure.

SCSI

The Small Computer Systems Interface (SCSI) is a popular family of protocols for communicating with I/O devices, especially storage devices.

There are two types of devices in SCSI protocol; the SCSI *Initiators (clients)* start the communications and the *Targets (servers)* responds. The initiators are devices that request commands be executed. Targets are devices that carry out the commands. The endpoint, within the target, that executes the command is referred to as a "logical unit" (LU). A target is a collection of logical units, in general of the same type, and are directly addressable. The structure used to communicate a command from an application client to a device server is referred to as a Command Descriptor Block (CDB). An SCSI command or a linked set of commands is referred to as a "task." Only one command in a task can be outstanding at any given time. SCSI command execution results in an optional data phase and a status phase. In the data phase, data travels either from the initiator to the target, as in a WRITE command, or from the target to the initiator, as in a READ command. In the status phase, the target returns the final status of the operation. The status response terminates an SCSI command or task.

The basic function of the SCSI driver is to build SCSI Command Descriptor Blocks (CDB) from requests issued by the application, and forwards them to the iSCSI layer. The SCSI driver also receives CDBs from the iSCSI layer and forwarding the data to the application layer. Figure 1 is a basic SCSI Command Descriptor Block (CDB).

Table 1 SCSI Command Descriptor Block (CDB) Format

Bit	7	6	5	4	3	2	1	0
Byte								
0	Operation Code							
1	Command Specific Parameters							
n-1								
n	Control							

iSCSI

iSCSI protocol is an Internet draft standard being defined to allow SCSI commands to be carried over TCP/IP protocol. Internet drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and working groups. Cisco and IBM developed a draft standard for iSCSI in January of 2000. The iSCSI protocol Internet draft has been submitted to the IP Storage (IPS) Working Group of the IETF, in August of 2000, for standardization. The standard should be well defined by the fall of 2001, numerous vendors are currently developing products to the current draft standard. The iSCSI draft standards are available at the IETF Web site http://www.ietf.org/. The IP Storage Working Group (IPS) Web site is at http://www.ece.cmu.edu/~ips/.

iSCSI provides initiators and targets with unique names as well as a discovery method. The iSCSI protocol establishes communication sessions between initiators and targets, and provides methods for them to authenticate one another. An iSCSI session may contain one or more TCP connections and provides recovery in the event connections fail.

SCSI CDBs are passed from the SCSI generic layer to the iSCSI transport layer. The iSCSI transport layer encapsulates the SCSI CDB into an iSCSI Protocol Data Unit (PDU) and forwards it to the Transmission Control Protocol (TCP) layer. On a read, the iSCSI transport layer extracts the CDB from the iSCSI PDU, received from the TCP layer, and forwards the CDB to the SCSI generic layer. iSCSI provides the SCSI generic command layer with a reliable transport.

The following diagram illustrates the layering of the various SCSI command sets and data over different transport and physical layers.

	SCSI Applications (File Systems, Databases)		
SCSI Device Layer	SCSI Block Commands	SCSI Stream Commands	Other SCSI Commands
SCSI Generic Layer	SCSI Commands, Data, and Status		
SCSI Transport Layer	Parallel SCSI Transport	FCP SCSI over FC	iSCSI over TCP/IP
			TCP
			IP
Physical Layer	Parallel SCSI Interface	Fibre Channel	Layer 2 Ethernet

iSCSI Naming and Addressing

The iSCSI protocol enables a methodology for both naming and address of initiators and targets. iSCSI provides a means of uniquely identifying (naming) iSCSI initiators and targets with a URN like iSCSI Name. In addition to an iSCSI Name each iSCSI initiator and target has one or more addresses. Addresses can change as an initiator or target move, but the name stays the same. For human readability initiators and targets may also have a non-unique alias.

The iSCSI Names are used in iSCSI:

1. To identify an initiator and target that may be addressable via more than one IP address and TCP port.
2. As an identifier for configurations that present multiple initiators or targets or both behind a single IP address and TCP port.
3. As a method to recognize multiple paths to the same initiators or targets on different IP addresses and TCP ports.
4. As an identifier for initiators and targets to enable them to recognize each other regardless of IP address and TCP port mapping on intermediary firewalls.
5. As a symbolic address for source and destination targets for use in third-party commands.

The iSCSI Name defines a method to provide naming authorities with a unique top-level name space. The use of the naming authority means that iSCSI Names can be assigned by OS vendors, driver or NIC vendors, device vendors, gateway vendors, service provider, or even the customers. An iSCSI Name consists of three parts: a type designator, followed by a naming authority, with the remaining format designated by the naming authority. In the first iSCSI Name of the following examples “iscsi” is the type designator and the naming authority is “com.acme”.

Examples of typical iSCSI Names are as follows:

iscsi.com.acme.sn.8675309

iscsi.com.acme.sw.hostid.4567890

The target may also provide a default iSCSI Target Name called “iSCSI” that is not a globally unique name. An initiator can log into this default target iSCSI Target Name and use a text command called “SendTargets” to retrieve a list of iSCSI Target Names that exist at that address.

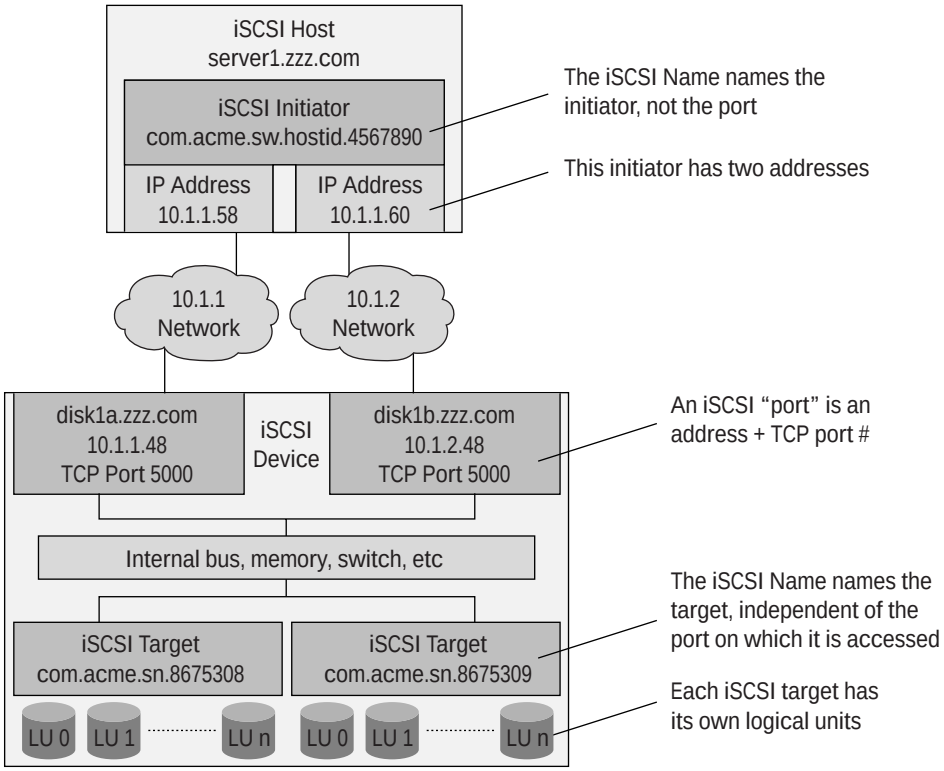
iSCSI targets can be identified by a flexible path address (URL), where the path is the combination of DNS name or IP address, a TCP port, and the target iSCSI Name. An iSCSI address specifies a single path to an iSCSI target. The iSCSI address contains the iSCSI Name and is presented in a URL-like format. The iSCSI address format is as follows.

<domain-name>[:<port >]/<iSCSI Name>

The iSCSI address or URL is not generally used within normal connections between iSCSI initiators and targets; it is primarily used during discovery.

An alias is simply a descriptive name that can be assigned to an initiator or target that is independent of the iSCSI Name, and does not have to be unique.

iSCSI Names and aliases are encoded in the UTF-8 text format, which allows them to include international characters, as well as ASCII.



iSCSI Discovery

Domain Name Service (DNS) may be used to resolve the <domain-name> of the URL to one or multiple IP addresses. When a domain-name resolves to multiple IP addresses, these addresses should be equivalent for functional purposes. This means that the addresses can be used interchangeably with consistent performance. The discovery process responds to two basic inquiries from an initiator:

1) Where is iSCSI Target Name "iscsi.com.acme.sn.8675309"?

2) I am iSCSI Initiator Name "iscsi.com.ame.sw.hostid.4567890" which target should I attempt to access?

An iSCSI initiator can discover an iSCSI target in the following different ways:

1. By configuring the target's address on the initiator.
2. By configuring a default target address on the initiator and the initiator connects to the target and requests a list of iSCSI Names, via a separate SendTargets command.
3. By issuing Service Location Protocol (SLP) multicast requests, to which the targets may respond.
4. By querying a storage name server for a list of targets that it can access.

iSCSI Login and Negotiations

Before iSCSI initiators can send SCSI commands to a target, it must first establish an iSCSI session. A session is composed of one or more TCP connections. The initiator establishes each TCP connection and begins the login phase of that connection. The login phase must be completed on each TCP connection before it can be used to transport SCSI commands.

iSCSI login is a mechanism used to establish a TCP connection, between initiators and targets. It authenticates the parties, negotiates the session's parameters, open security association protocol, and marks the connection as belonging to an iSCSI session. The initiator begins the login process by connecting to a well known TCP port. The target listens on the well-known TCP port for incoming connections.

A single TCP connection is established to transfer SCSI commands, data, and status information for a single "task." Communication between an initiator and target may occur over one or more TCP connections. One or more TCP connections linking an initiator and a target form a "session." A session is used to identify to a target all the connections with a given initiator. TCP connections may be added to or deleted from a session.

As part of the login process the initiator and target my wish to authenticate each other and set the security association protocol for the session.

Once the login process has completed the iSCSI session is said to be in the full feature phase. The initiator may then send SCSI commands and data to the various LUs on the target by encapsulating them in iSCSI messages that are sent over the established iSCSI session.

Command Numbering and Acknowledging

Commands in transit from the initiator to the target SCSI layer are numbered by iSCSI and the number is carried by the iSCSI PDU as the Command Sequence Number (CmdSN). Command numbering is on a session basis and during command delivery the allocated CmdSNs are unique session wide. The iSCSI target layer must deliver the commands to the SCSI target layer in the order specified by CmdSN. The CmdSN ceases to be significant once the target receives the command. The CmdSN can also be used as a mechanism for command flow control over a session. iSCSI PDUs that have a task association carry the CmdSN. A task is defined as a unit of work to be performed by the initiator and target either a command or group of linked commands. An initiator task tag identifies the task for the life of the task.

Response/Status Numbering and Acknowledging

Responses in transit from the target to the initiator are numbered by iSCSI and the number is carried by the iSCSI PDU as the Status Sequence Number (StatSN). Status numbering is on a per connection basis and is used to enable missing status detection and recovery in the presence of transient or permanent communication errors. An Expected Status Sequence Number (ExpStatSN) is maintained by the initiator to acknowledge status. A target may discard all the state information maintained for recovery after the status delivery is acknowledged through the ExpStatSN. A difference between StatSN and ExpStatSN may indicate a failed connection.

PDU Template, Header, and Op-codes

This section is subject to change do to standards activity.

All iSCSI PDUs begin with one or more header segments followed by zero or one data segment. The header segment group may be preceded by a header-digest (CRC). Data segments may be followed by a data-digest.

The first segment is the Basic Header Segment (BHS) a fixed-length 44-byte header segment. An Additional Header Segments (AHS) may follow the BHS. Each header segment is preceded by a 4-byte Next-Qualifier (What's Next WN). The WN field indicates what the next segment is. When there is only a BHS (with no data or digests) the net size of the iSCSI PDU is 48 bytes.

Overall structure of a PDU template is as follows

Byte	0	1	2	3
0	WN	WN specific fields		
4 + 44	BHS			
48	WN	WN specific fields		
52 + 92	AHS			
m	Header-Digest (optional)			
n + m	Data Segment (optional)			
	Data-Digest (optional)			

Basic Header Segment (BHS) for SCSI (Initiator) Command

Byte	0	1	2	3
0	Opcode	Opcode—specific fields		Reserved
4 + 8	Logical Unit Number (LUN)			
12	Initiator Task Tag			
16	Expected Data Transfer Length			
20	CmdSN			
24	ExpStatSN or EndDataSN			
28 + 44	SCSI Command Descriptor Block (CDB)			

Basic Header Segment (BHS) for SCSI (Target) Response

Byte	0	1	2	3
0	Opcode	Opcode—specific fields		Reserved (0)
4 + 8	Reserved (0)			
12	Initiator Task Tag			
16	Basic Residual Count			
20	StatSN			
24	ExpCmdSN			
28	MaxCmdSN			
32	EndDataSN or Reserved (0)			
36	R2TEndDataSN or Reserved (0)			

Byte	0	1	2	3
40	Bidi-Read Residual Count			
44	Digests if any			
48	Response Data or Sense Data (optional)			

iSCSI Commands and Responses

iSCSI sends SCSI commands, data and status over its TCP connections encapsulated in iSCSI Protocol Data Units (PDU). There are several types of iSCSI PDUs, they are identified by individual operation codes (Op-codes). Op-codes are divided into two categories: initiator op-codes and target op-codes also called responses. Some PDUs transport SCSI commands, data, and status and others are used for iSCSI control.

Valid initiator Op-codes (Commands) that support SCSI functions include:

0x01 SCSI Command (encapsulates an SCSI Command Descriptor Block)
0x02 SCSI Task Management Command
0x05 SCSI Data (for WRITE operations)

Valid initiator Op-codes (Commands) that support iSCSI functions include:

0x00 NOP-Out (from initiator to target)
0x03 Login Command
0x04 Text Command
0x06 Logout Command
0x10 SACK Request (optional)

Valid target Op-codes (Responses) that support SCSI functions include:

0x41 SCSI Response (contains SCSI status and possible sense information)
0x42 SCSI Task Management Response
0x45 SCSI Data (for READ operations)
0x50 Ready To Transfer (R2T)

Valid initiator Op-codes (Responses) that support iSCSI functions include:

0x40 NOP-In (from target to initiator)
0x43 Login Response
0x44 Text Response
0x46 Logout Response
0x51 Asynchronous Message
0x6F Reject

0x01 SCSI Command

This PDU encapsulates SCSI Command Descriptor Blocks (CDB). This PDU may contain all the data that is associated with the SCSI command. The header portion of this PDU contains the following control information.

- *Initiator Task Tag* is related to a (SAM-2) task and is valid for the life of the tag.
- *Expected Data Transfer Length* is the number of bytes of data to be transferred during the current SCSI operation.
- *Command Sequence Number (CmdSN)* is an assigned number that enables ordered delivery of PDUs across multiple TCP connections in a single session.
- *Expected Status Sequence Number (ExpStatSN)* is an assigned number that is used to detect missing status.

0x41 SCSI (Command) Response

This PDU is used to report the SCSI status of the SCSI Command. The header portion of this PDU contains the following control information.

- *Status/Response*, the Status field is used to report the (SAM-2) SCSI status of the command. The Response field is used to report a SAM type Service Response.
- *Sense or Response Data*, if the SCSI command fails this data will contain sense data for the failed command.

0x05 SCSI Data

This PDU is used to transfer data from the initiator to the target (write) or from the target to the initiator (read). The PDU specifies the length of the data payload, the Target Transfer Tag, provided by the receiver for this data transfer, and a buffer offset.

- *Final (F) Bit*, for output (write) data this bit is 1 for the last PDU of unsolicited data or the last PDU of a sequence answering an R2T. For input (read) data this bit is 1 for the last input PDU associated with the command.
- *Data Sequence Number (DataSN)*, for input (read) data PDUs the DataSN is the data PDU number within the data transfer for the command identified by the Initiator Task Tag. For output (write) data PDUs the DataSN is the data PDU number within the current output sequence identified by the Initiator Task Tag or by the Target Task Tag and LUN for R2T data.
- *Buffer Offset*, field contains the offset of this PDU data payload data against the complete data transfer.

0x02 SCSI Task Management

This PDU provides an initiator with a way to explicitly control the execution of one or more of the following tasks.

1. Abort task—aborts the task identified by the referenced Task Tag field.
2. Abort Task Set—abort all Tasks issued by this initiator on the Logical Unit.
3. Clear ACA—clears the Auto Contingent Allegiance.
4. Clear Task Set—aborts all Tasks, from all initiators, for the Logical Unit.
5. Logical Unit Reset—
6. Target Warm Reset—
7. Target Cold Reset—

For all of these functions, the SCSI Task Management Response is returned using the initiator Task Tag to identify the operation of which it is responding.

- *Referenced Task Tag*, the initiator Task Tag of the task to be aborted.

0x42 SCSI Task Management Response

This PDU provides a response back to the initiator upon completion of the SCSI Task Management Command. The responses may include the following values.

1. 0 Function Complete
2. 1 Task was not in task set
3. 255 Function Rejected

- *Referenced Task Tag*, initiator Task Tag of a task not found.

0x50 Ready to Transfer (R2T)

When an initiator issues an SCSI Command with a CDB that requires data to be sent from the initiator to the target (write), the target issues an R2T PDU to request the data blocks it requires. The target may send several R2T PDUs and thus have a number of data transfers pending.

- *Target Transfer Tag*, is a tag assigned to each R2T request sent to the initiator by the target. The transfer tag is used by the target to identify data PDUs it receives.

0x04 Text Command

The Text Command permits the initiator to inform a target of its capabilities or to request special operations.

- *Text*, The initiator sends the target a set of key=value or key=list pairs encoded in UTF-8 Unicode. Character strings are represented as plain text. Numeric and binary values are represented using either decimal numbers or hexadecimal notation.

0x44 Text Response

The Text Response message contains the target's response to the initiator's Text Command. The format of the Text Response matches that of the Text Command.

- *Text Response*, field contains responses in the same format as the Text Command.

0x03 Login Command

The Login Command PDU is used after a TCP connection has been established between an initiator and target. It is used to authenticate the parties, negotiate the session's parameters, open security associations, and mark connections as belonging to a session.

- *Version-max*, maximum iSCSI standard version number supported.
- *Version-min*, is the minimum iSCSI standard version number supported.
- *Connection Identifier (CID)*, is a unique ID for this connection within the session.
- *Initiator Session-Identifier*, is an initiator defined ID that must be the same for all connections within a session.
- *Login Parameters*, may be provided by the initiator to enable the target to determine if the initiator may use the targets resources and the initial text parameters for security exchange. Some typical keys used are as follows:
  - *MaxConnections—Lo*, is the maximum number of connections requested/acceptable that are negotiated by the initiator and target.
  - *UseR2T=<yes/no>*, is used to turn off the use of R2T, thus allowing the initiator to send data to a target without the target having to send an R2T to the initiator. The default is that R2T is required, unless both the initiator and target send this key pair specifying UseR2T:no.
  - *DataPDU Length*, is the maximum data payload in 512 byte units, negotiated by the initiator and target, for command or data PDUs.
  - *Vendor Specific Key Format*, is a key defined for use as vendor-specific purposes, these keys start with X-. To identify the vendor it is suggested the reverse DNS-name is used as a prefix to the key proper.

The Login process enables negotiation of digests for end-to-end data integrity, using cyclic checksums, enabling integrity checks beyond those provided by link layers.

The authentication exchange authenticates the initiator and target to each other. Authentication methods including Kerberos V5, Secure Remote Password (SRP), or proprietary methods can be negotiated.

0x43 Login Response

The Login Response PDU indicates the progress and/or end of the login phase. After security is established, the login response is authenticated.

- *Version-active/lowest*, is the version supported by both the initiator and target. If the target does not support a version in the range identified by the initiator, the target rejects the login and identifies the lowest version supported.
- *Status-Class*, the status returned in a Login Response indicates the status of the login request. Status classes are as follows:

1. Success—indicates the iSCSI target accepted the request.
2. Redirection—indicates further action must be taken by the initiator to complete the request.
3. Initiator Error—indicates the initiator likely caused the error.
4. Target Error—indicates the target is incapable of fulfilling the request.

0x06 Logout Command

The Logout Command PDU is used to perform a controlled closing of a connection. An initiator may use a Logout Command to remove a connection from a session or to close an entire session.

• *Connection ID (CID),* the CID is the connect identifier of the connection to be closed, including closing the TCP stream.

0x46 Logout Response

The Logout Response is issued by the target to indicate that the cleanup operation for a failed connection is complete.

0x00 NOP-Out

The NOP-Out with the p bit set acts as a "ping command" and used to verify that a connection is still active and operational.

• *P (Ping) Bit*, when set requests a NOP-In response.

• *Ping Data*, is reflected in the Ping response. The length is limited by the negotiated parameters.

0x40 NOP-In

The NOP-In is the response to a NOP-Out when the P bit is set. The target must respond with a NOP-In with the same Initiator Task Tag that was provided in the NOP-Out Command. And should duplicate as much of the ping data as allowed by configurable target parameters.

0x51 Asynchronous Message

An asynchronous Message may be sent from the target to the initiator without corresponding to a particular command. The target specifies the status for the event and sense data. Some Asynchronous Messages are related to iSCSI others are related to SCSI. Disabling SCSI Messages in a mode page will have no effect on iSCSI Asynchronous Messages.

• *iSCSI Event*, are codes returned for iSCSI Asynchronous Messages.

1. Target is being reset.
2. Target requests Logout.
3. Target indicates it will/has dropped the connection.

• *SCSI Event, are codes returned for SCSI Asynchronous Messages.*

1. An error condition was encountered after command completion.
2. A newly initialized device is available to this initiator.
3. Some other type of unit attention condition has occurred.
4. An asynchronous event has occurred.

0x6F Reject

If a target receives an iSCSI message with a format error (e.g., inconsistent fields, reserved fields not 0, non-existent LUN, etc.) or a digest error (e.g., invalid payload or header). The target issues a Reject and returns the header of the message in error as the data of the response.

• Reject Reason codes

1. 1—Format error
2. 2—Header Digest Error
3. 3—Payload Digest Error
4. 4—Data-SACK Reject
5. 5—Command Retry Reject
6. 15—Full Feature Phase Command before login

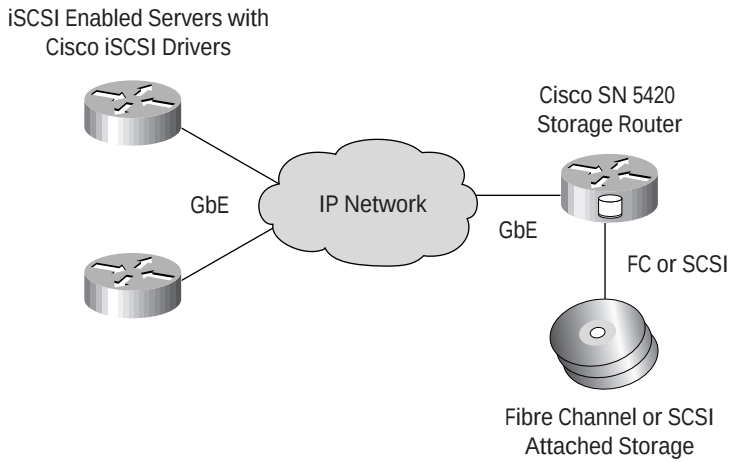
iSCSI Implementation

The Cisco SN 5420 Storage Router release 1.0 is the first product to implement the iSCSI protocol, compliant with the Internet Draft standard dated July 10, 2000. The SN 5420 makes storage systems accessible from anywhere, via IP networks, using the iSCSI protocol regardless of the operating system. Block level access, using iSCSI protocol, is transparent to the application and does not require additional software layers for IP network intelligence. Any application capable of accessing storage using SCSI protocols becomes an IP application that can pass through the SN 5420 Storage Router to Fibre Channel or SCSI attached storage devices.

The Storage Router is a combination of a host driver software and external hardware (SN 5420). The iSCSI drivers are platform specific and are loaded on the application servers. Drivers for Unix, Linux, NT, and Windows 2000 operating systems will be supported.

The iSCSI driver is installed and configured as one of the device drivers available to the operating system. The iSCSI driver is configured with the IP address(s) of one or more of the SN 5420's SCSI Router Service instances. When the iSCSI driver is started it queries the SN 5420s for iSCSI targets and Logical Unit Numbers (LUNs) available on the SN 5420. The host creates an entry in its device table for each of the target/LUN combinations that it is allowed access to based on an Access List on the SN 5420.

When the host/application wishes to communicate with one of the SCSI target/LUNs, the SCSI driver generates a CDB and forwards it to the iSCSI driver, which encapsulates it in an iSCSI PDU and forwards it to the TCP/IP layers. The TCP/IP layer encapsulates the iSCSI PDU in a TCP packet and then an IP datagram. The packet/datagram are forwarded to and encapsulated in a physical layer frame, Gigabit Ethernet, and transmitted over the network. The reverse process occurs at the SN 5420 and the SCSI CDB is forwarded to the addressed device (target/LUN) via a Fibre Channel or Parallel SCSI (future) interface.



SN 5420 iSCSI Address Mapping

The Cisco SN 5420 is manually configured to enable address mapping from logical targets or target and LUN addresses to physical storage device addresses discovered from fibre channel connections. The logical target address is referred to as the "iSCSI Target" address.

The first step is to configure the SCSI Router Service instances on the SN 5420. The SCSI router services form two associations; one is to the SN 5420 server interface (Gigabit Ethernet) to which the host data server(s) iSCSI configuration will point, and the other is the device interface (Fibre Channel or SCSI) to which the data storage device(s) are attached.

The ‘iSCSI Target’ logical (arbitrary) address is associated with an iSCSI Routing Service and mapped to the physical storage addresses. The physical storage addresses may be Fibre Channel Loop ID, World Wide Port Name (WWPN) and/or World Wide Node Name (WWNN) or SCSI target and LUN.

A full mapped path is then created by associating a named Access List (host data server IP addresses) with a named SCSI Router Service instance that directs the request to a named iSCSI Target.

iSCSI Host Drivers provide varied approaches to interfacing to the varied operating systems. The following description of address mapping for Windows NT provides the general concepts used by other host drivers.

The iSCSI driver, for Windows NT, emulates a locally attached SCSI interface to the NT operating system. The iSCSI driver provides NT with SCSI inquiry data responses, as would a standard SCSI driver. The iSCSI driver is manually installed and configured to point to the IP address(s) of the SN 5420 iSCSI Router Server instances that map to the logical iSCSI Target addresses that the host should see as local storage. The NT host then displays the devices in Disk Administrator as if they were locally attached. Now they can be partitioned, committed, and assigned drive letters and formatted. Upon a reboot, the iSCSI driver assigns the previously assigned NT drive letters by using configuration information retrieved from the NT registry and associates the drive letters to the “iSCSI Target” addresses after iSCSI login to the SN 5420 Router Service instance.

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

European Headquarters
Cisco Systems Europe
11, Rue Camille Desmoulins
92782 Issy-les-Moulineaux
Cedex 9
France
www-europe.cisco.com
Tel: 33 1 58 04 60 00
Fax: 33 1 58 04 61 00

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-7660
Fax: 408 527-0883

Asia Pacific Headquarters
Cisco Systems Australia, Pty., Ltd
Level 9, 80 Pacific Highway
P.O. Box 469
North Sydney
NSW 2060 Australia
www.cisco.com
Tel: +61 2 8448 7100
Fax: +61 2 9957 4350

Cisco Systems has more than 200 offices in the following countries and regions. Addresses, phone numbers, and fax numbers are listed on the

Cisco Web site at www.cisco.com/go/offices

Argentina • Australia • Austria • Belgium • Brazil • Bulgaria • Canada • Chile • China PRC • Colombia • Costa Rica • Croatia Czech Republic • Denmark • Dubai, UAE • Finland • France • Germany • Greece • Hong Kong SAR • Hungary • India • Indonesia • Ireland Israel • Italy • Japan • Korea • Luxembourg • Malaysia • Mexico • The Netherlands • New Zealand • Norway • Peru • Philippines • Poland Portugal • Puerto Rico • Romania • Russia • Saudi Arabia • Scotland • Singapore • Slovakia • Slovenia • South Africa • Spain • Sweden Switzerland • Taiwan • Thailand • Turkey • Ukraine • United Kingdom • United States • Venezuela • Vietnam • Zimbabwe

5/01 LW2392