IPS
Internet Draft
draft-ietf-ips-iscsi-name-disc-03.txt
Draft Title: iSCSI Naming and Discovery

Mark Ba
Ci

Joe C

Jim Haf

Howard H
Pi

Jack Harw

John Huff

Yaron Kl
San

Marjorie Krue
Hewlett-Pack

Lawrence Lam
San Valley Syst

Todd Spe
Adap

Joshua Ts
Nis

Kaladhar Voruga

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026 except that the right to produce derivative works is not granted. Internet-Drafts are worki documents of the Internet Engineering. Task Force (IETF), its area

and its working groups. Note that other groups may also distribute working documents as Internet-Drafts. Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet- Drafts as reference material or to cite them other than as "work in progress." The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Comments

Comments should be sent to the ips mailing list (ips@ece.cmu.edu) to kaladhar@us.ibm.com

Abstract

This document describes iSCSI [7] naming and discovery details. Th document complements the iSCSI Protocol draft. Flexibility is the guiding principle behind this document. That is, an effort has bee made to satisfy the needs of both small isolated environments, as as large environments requiring secure/scalable solutions.

Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT" "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119.

Table of Contents

1. iSCSI Naming Philosophy

The notion of an iSCSI name is required at both the targets and at the initiators. iSCSI name is required at the target because it uniquely identifies a target as a storage resource for the initiat iSCSI initiator name is required at the initiator because it helps uniquely identify an initiator for the purpose of target resource allocation (i.e., which initiator has access to which target resource). iSCSI name is also used to provide a mechanism for worl wide unique identification of SCSI Initiator Ports (analogous to F WWPortnames). The SCSI port name is used by SCSI during SCSI reservations, SCSI initiator specific task queue management and SCSI mode page management. Furthermore, iSCSI initiator names can also potentially used by software layers such as security and management software to uniquely identify initiators to targets.

It is necessary for the iSCSI names to be unique within the operat domain of the end user. However, since user operation domains can potentially merge with other user operation domains, the iSCSI nam mechanism has been architected to ensure world wide uniqueness. In order to ensure both world wide name uniqueness iSCSI provides for the use of different types of naming authority mechanisms.

Furthermore, iSCSI names are associated with iSCSI nodes instead o with network adapter cards to ensure the free movement of network HBAs between hosts without carrying over the SCSI state informatio (reservations, mode page settings etc).

Since there can be multiple separate iSCSI sessions (via different iSCSI ports) between the same iSCSI initiator and target nodes, iSCSI has introduced the notion of an initiator session id (ISID) and a target session id (TSID) to help in uniquely identifying eac of the iSCSI sessions. The ISID and the TSID are not global identi but together uniquely identify a session only within the context o given named iSCSI initiator and iSCSI target.

In addition to the mandatory iSCSI concepts of iSCSI initiator nam iSCSI target name, ISID and TSID, iSCSI also optionally allows for the specification of initiator and target aliases. Initiator and target aliases are optional constructs which help the users to associate semantic meanings with a particular initiator or target.

2. iSCSI Names

The main addressable, discoverable entity in iSCSI is an iSCSI Node. An iSCSI node can be either an initiator, a target, or both.

The concepts of names and addresses have been carefully separat iSCSI:

- An iSCSI Name is a location-independent, permanent identifi for an iSCSI node. An iSCSI node has one iSCSI name, which stays constant for the life of the node. The terms "initia name" and "target name" also refer to an iSCSI name.

- An iSCSI Address specifies not only the iSCSI name of an iS node, but also a location of that node. The address consis a host name or IP address, a TCP port number (for the targ and the iSCSI Name of the node. An iSCSI node can have an number of addresses, which can change at any time, particul if they are assigned via DHCP.

A similar analogy exists for people. A person in the USA might

Robert Smith
SSN: 333-44-5555
Phone: +1 (763) 555.1212
Home Address: 555 Big Road, Minneapolis, MN 55444

Work Address: 222 Freeway Blvd, St. Paul, MN 55333

In this case, Robert's globally unique name is really his Socia Security Number his common name, "Robert Smith", is not guarant to be unique. Robert has three locations at which he may be rea two Physical addresses, and a phone number. In this example, Robert's SSN is like the iSCSI Name, his phone number and addre are analogous to the iSCSI Address, and "Robert Smith" would be human-friendly label for this person.

2.1. iSCSI Name Requirements

Each iSCSI node, whether an initiator or target, must have an iSCSI name.

iSCSI names may be assigned by a hardware manufacturer, softwar manufacturer, or the end user. A naming authority scheme is provided to ensure that each of these can confidently generate

unique names.

iSCSI names are designed to fulfill the following requirements:

1. iSCSI names are globally unique. No two initiators or targ should have the same name.

2. iSCSI names are permanent. An iSCSI initiator or target ha same name for its lifetime.

3. iSCSI names do not imply a location or address. An iSCSI initiator or target can move, or have multiple addresses. change of address does not cause a change of name.

4. iSCSI names must not rely on a central name broker; the nam authority must be distributed.

5. iSCSI names must support integration with existing unique n schemes.

6. iSCSI names must rely on existing naming authorities. iSCS must not create its own naming authority.

The encoding of an iSCSI name also has some requirements:

1. iSCSI names have one single encoding method when transmitte over various protocols.

2. iSCSI names must be relatively simple to compare. The algo for comparing two iSCSI names for equivalence must not rely any external server.

3. iSCSI names must be transcribable by humans. iSCSI names s be kept as simple as possible, and should not use more than few special characters. They must provide for the use of international character sets, and must not allow the use of different names that would be identical except for their ca

Whitespace characters must not be allowed.

4. iSCSI names must be transport-friendly. They must be transported using both binary and ASCII-based protocols, as as on paper.

An iSCSI Name really names a logical software entity, and is no tied to a port or other hardware that can be changed. For inst

an initiator name should name the iSCSI initiator node, and not particular NIC or HBA card. When multiple NICs are used, they should generally all present the same iSCSI initiator name to t targets, since they are really to the same entity. In most operating systems, the named entity is the operating system ima Most hosts will have a single OS running; some of the really bi ones could have multiples.

A target name should similarly not be tied to hardware interfac which can be changed. A target name should identify the logica target, and must be the same for the target regardless of the physical portion being addressed. This gives iSCSI initiators easy way to determine that two targets it has discovered are re two paths to the same target.

The iSCSI Name is designed to fulfill the functional requiremen for Uniform Resource Names (URN) [RFC1737]. Among these requirements are that the name must have a global scope, indepe of address or location, and that it be persistent and globally unique. It must be extensible, and scale with the use of namin authorities. The encoding of the name should be transcribable human, as well as be machine-readable. There are other require as well; please read RFC1737 (only 5 pages) for definitions of requirements.

2.2. iSCSI Name Encoding

An iSCSI name is a UTF-8 encoding of a string of Unicode characters,
with the following properties, described in [26]:

- it is in Normalization Form C [25]
- it contains only the following types of characters:
  - ASCII dash character ('-'=U+002d)

  - ASCII dot character ('.'=U+002e)
  - Any character allowed by the output of the iSCSI stringprep template [26]
- when encoded in UTF-8, it is no more than 255 bytes

The stringprep process is described in [24]; iSCSI's use of the

stringprep process is described in [26]. Stringprep is a method
designed by the Internationalized Domain Name (IDN) working
group to translate human-typed strings into a format that can be
compared as opaque strings, and does not include punctuation,
spacing, dicritical marks, or other characters that could get in the
way of transcribability. It also converts everything into its
equivalent of lower case.

Note that in most cases, the stringprep process does not need to be
implemented:

- If the names are just generated using lower-case (in any
  character set) plus digits, no normalization is required.

- If the names are generated from some other all-ASCII string,
  tolower() normalizes and isalnum() verifies.

- If the names are generated from more general, internationalized
  text, either the equivalent of tolower() and isalnum() appropriate
  to the character set may be used, or the full stringprep procedure
  can be used.

When included in Text or Login messages, an iSCSI Name MUST be
formatted in UTF-8 form.

Since iSCSI names encoded in UTF-8 are "normalized" (there is
one and only one representation for each possible name), they
may be safely compared byte-for-byte.

The iSCSI Name may be displayed by user interfaces, but its contents
are not parsed or interpreted by initiators and targets themselves.

2.3. iSCSI Name Structure

An iSCSI name consists of Two parts: a type designator, followe
a unique name string

The iSCSI Name does not define any new naming authorities. Ins
it supports two existing authorities: an iSCSI-Qualified Name,
domain names as an authority, similar to the Java class naming
hierarchy, and the EUI format used in Fibre Channel world-wide
names.

Since there are different types of naming authorities, there ar
different types of iSCSI Names to make use of them. Each name
prefixed with a short type designator string that indicates the
of naming authority being used.

Here are the type designator strings that may currently be used

```
iqn.       - iSCSI Qualified Name
eui.       - Remainder of the string is an EUI-64 addr
             in ASCII hexadecimal.
```

As these two naming authorities will suffice in nearly every ca
for both software and hardware-based entities, the creation of
additional type designators is discouraged. One of these two t
strings MUST be used when constructing an iSCSI name; any type
string not listed here is not allowed, as they cannot be guaran
to be unique.

2.3.1. Type "iqn." (iSCSI Qualified Name)

This iSCSI name type can be used by any organization which owns
Domain Name. This naming format is handy when an end user or
service provider wishes to assign the iSCSI Name for a target o
initiator. Customers which own domain names may not own an EUI

OUI, SCSI Vendor ID, or any of the other assigned identifiers t
could be used as a naming authority.

To generate names of this type, the person or organization
generating the name must own a DNS domain name. This name does
have to be active, and does not have to resolve to an address;

just needs to be reserved to prevent others from generating iSC
names using the same domain name. For example, "ACME Storage
Arrays, Inc.", might own the domain "acme.com".

Since a domain name can expire, be acquired by another entity,
used to generate iSCSI names by both owners, the domain name mu
be additionally qualified by a date during which the naming aut
owned the domain name. A date code is provided as part of the
format for this reason.

The iSCSI qualified name string consists of:

- The string "iqn.", used to distinguish these names from othe
  types, such as "eui".

- A date code, in yyyy-mm format. This date code uses the Gre
  calendar. All four digits in the year must be present. Bot
  digits of the month must be present, with January == "01" an
  December == "12". The dash must be present. The date
  reflected in this code MUST be a date during which the namin
  authority owned the domain name used in this format, and SHO
  the date on which the domain name was acquired by the naming
  authority.

- Another ".".

- A reversed domain name, owned by the person or organization
  creating the iSCSI name. For example, our storage vendor
  example would reverse its name to "com.acme".

- Another ".".

- Any string, within the character set and length boundaries,
  the owner of "acme.com" deems appropriate. This may contai
  product types, serial numbers, host identifiers, software k
  or anything else that makes sense to uniquely identify the
  initiator or target.

Everything after the backwards domain name, followed by another

".", can be assigned as needed by the owner of the domain name.
the responsibility of the Organizational (Company) naming author
ensure that the iSCSI names it assigns are world wide unique.

iSCSI has given the Organizational naming authority additional
flexibility by permitting it to hand out local naming authority

subordinate organizations. In this way it will be possible for Organizational naming authority to assign for example, the strin "storage", to one subgroup naming authority and "storage.tape" t another. In this case the subgroups may add a ":" following the assigned subgroup string to ensure ongoing uniqueness. For examp "storage:" and "storage.tape:". Also, additional sub-qualifiers assigned and separated by a "." as explained above.

Using this approach, the subgroup with the sub-naming authority of "storage" might, overtime, also create some Tape products. I case, both subgroups might use the same qualifying names. It wo expected in this case that a naming conflict might occur, howeve using the ":" appropriately the conflicts can be avoided. In th example com.acme.storage:tape.sys1.xyz and com.acme.storage.tape:sys1.xyz would not be in conflict even tho same sub-names are used.

The following are examples of iSCSI qualified names from an equipment vendor:

Type	Date	Organization Naming Auth	Subgroup Naming Authority and/or string Defined by Org. or Local Naming Authority

iqn.2001-04.com.acme.diskarrays-sn-a8675309
iqn.2001-04.com.acme.storage:tape.sys1.xyz
iqn.2001-04.com.acme.storage.tape:sys1.xyz

Where:

"iqn" specifies the use of the iSCSI qualified name as the authority.

"2001-04" is the year and month on which the naming autho acquired the domain name used in this iSCSI name.

"com.acme" defines the Organizational naming authority. Th owner of the DNS name "acme.com" has the sole right of use this name within an iSCSI name, as well as the responsibili keep the remainder of the iSCSI name unique. In this case, acme.com happens to manufacture disk arrays.

"diskarrays" was picked arbitrarily by acme.com to identi
the disk arrays they manufacture. Another product
that ACME makes might use a different name, and have it's
own namespace independent of the disk array group.

"sn" was picked by the disk array group of ACME to show t
what follows is a serial number. They could have just as
that all iSCSI Names are based on serial numbers, but the
thought that perhaps later products might be better ident
by something else. Adding "sn" was a future-proof measur

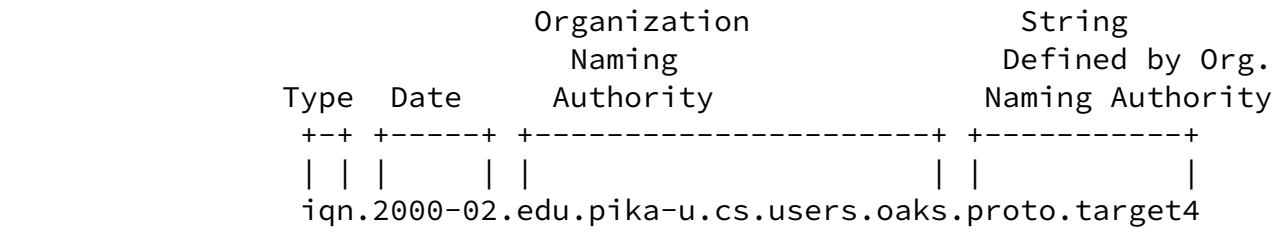
"a8675309" is the serial number of the disk array, unique
identifying it from all other arrays.

"storage:" is the string that represents another sub-naming
authority.

"storage.tape:" is still another sub-naming authority.

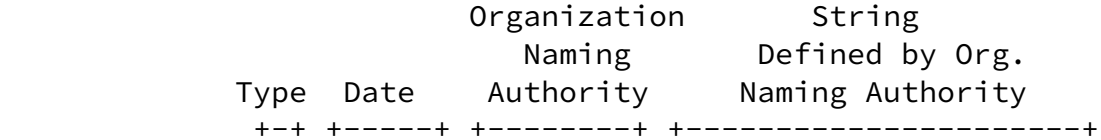
"sys1.xyz" is a naming sub-qualifier.

The following is an example of a name that might be constructed
an research organization:



In the above example, Professor Oaks of Pika University is buil
research prototypes of iSCSI targets. Pika-U's computer scienc
department allows each user to use his or her user name as a na
authority for this type of work. Professor Oaks chose to use
"proto.target4" for a particular target.

The following is an example of an iSCSI name string from a stor
service provider:



```
        | | |     | |       | |                       |
        iqn.1995-11.com.my-ssp.customers.4567.disks.107
```

In this case, a storage service provider (my-ssp.com) has decid re-name the targets from the manufacturer, to provide the flexibility to move the customer's data to a different storage subsystem should the need arise.

My-ssp has configured the iSCSI Name on this particular target one of its customers, and has determined that it made the most to track these targets by their Customer ID number and a disk number. This target was created for use by customer #4567, and the 107th target configured for this customer.

Note that when reversing these domain names, the first component(after the "iqn.") will always be a top-level domain n which includes "com", "edu", "gov", "org", "net", "mil", or one the two-letter country codes. The use of anything else as the component of these names is not allowed. In particular, compan generating these names must not eliminate their "com." from the string.

Again, these iSCSI names are NOT addresses. Even though they m use of DNS domain names, they are used only to specify the nami authority. An iSCSI name contains no implications of the iSCSI target or initiator's location. The use of the domain name is a method of re-using an already ubiquitous name space.

Note that the SCSI Vendor ID or IEEE OUI could have been specif as a naming authority. However, some large customers and servi providers may wish to use their own identification scheme, rath than that provided by the manufacturer. These customers would likely have a registered Vendor ID, but the domain name we used ubiquitous, and was deemed more appropriate.

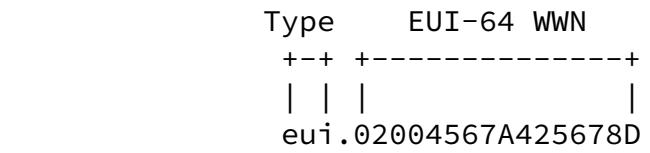
2.3.2. Type "eui." (IEEE EUI format)

The IEEE iSCSI name might be used when a manufacturer is alread basing unique identifiers on World-Wide Names as defined in the SPC-2 specification.

It may also be used by a gateway representing a Fibre Channel o SCSI device that is already adequately identified using a world name.

The format is "eui." followed by 16 hex digits.

Example iSCSI name :



2.4 iSCSI Alias

The iSCSI alias is a UTF-8 text string that may be used as an additional descriptive name for an initiator and target. This may not be used to identify a target or initiator during login, and does not have to follow the uniqueness or other requirement of the iSCSI name. The alias strings are communicated between

initiator and target at login, and can be displayed by a user interface on either end, helping the user tell at a glance whet the initiators and/or targets at the other end appear to be correct. The alias must NOT be used to identify, address, or authenticate initiators and targets.

The alias is a variable length string, between 0 and 255 charac and is terminated with at least one NULL (0x00) character. No other structure is imposed upon this string.

2.4.1 Purpose of an Alias

Initiators and targets are uniquely identified by an iSCSI Name These identifiers may be assigned by
a hardware or software manufacturer, a service provider, or eve the customer. Although these identifiers are nominally human-readable, they are likely be be assigned from a point of view different from that of the other side of the connection. For instance, a target name for a disk array may be built from the array's serial number, and some sort of internal target ID. Although this would still be human-readable and transcribable, it offers little assurance to someone at a user interface who

would like to see "at-a-glance" whether this target is really the correct one.

The use of an alias helps solve that problem. An alias is simply a descriptive name that can be assigned to an initiator or target, that is independent of the name, and does not have to be unique. Since it is not unique, the alias must be used in a purely informational way. It may not be used to specify a target at login, or used during authentication.

Both targets and initiators may have aliases.

2.4.2 Target Alias

To show the utility of an alias, here is an example using an alias for an iSCSI target.

Imagine sitting at a desktop station that is using some iSCSI devices over a network. The user requires another iSCSI disk, and calls the storage services person (internal or external), giving any authentication information that the storage device will require for the host. The services person allocates a new target for the host, and sends the Target Name for the new target, and probably an address, back to the user. The user th adds this Target Name to the configuration file on the host, an discovers the new device.

Without an alias, a user managing an iSCSI host would click on some sort of management "show targets" button to show the targets to which the host is currently connected.

```
+--Connected-To-These-Targets----------------------
|
|  Target Name
|
|  iqn.1995-04.com.acme.sn.5551212.target.450
|  iqn.1995-04.com.acme.sn.5551212.target.489
|  iqn.1995-04.com.acme.sn.8675309
|  iqn.2001-04.com.acme.storage:tape.sys1.xyz
|  iqn.2001-04.com.acme.storage.tape:sys1.xyz
|
+-------------------------------------------------
```

In the above example, the user sees a collection of iSCSI Names with no real description of what they are for. They will, of course, map to a system-dependent device file or drive letter,

but it's not easy looking at numbers quickly to see if everythi is there.

If a more intelligent target configures an alias for each targe
perhaps at the time the target was allocated to the host, a mor
descriptive name can be given. This alias may be sent back to
initiator as part of the login response, or found in the iSCSI M
then might be used in a display such as this. The new display m
look like:

```
+--Connected-To-These-Targets----------------------
|
|  Alias          Target Name
|
|  Oracle 1       iqn.1995-04.com.acme.sn.5551212.target.450
|  Local Disk     iqn.1995-04.com.acme.sn.5551212.target.489
|  Exchange 2     iqn.1995-04.com.acme.sn.8675309
|
+-------------------------------------------------
```

This would give the user a better idea of what's really there.

In general, flexible, configured aliases will probably be supported by larger storage subsystems and configurable gateway Simpler devices will likely not keep configuration data around for things such as an alias. The TargetAlias string could be either left unsupported (not given to the initiator during login) or could be returned as whatever the "next best thing" that the target has that might better describe it. Since it does not have to be unique, it could even return SCSI inquiry string data.

Note that if a simple initiator does not wish to keep or displa
alias information, it can be simply ignored if seen in the login
response.

2.4.3 Initiator Alias

An initiator alias can be used in the same manner as a target alias. An initiator may send the alias in a login request, when it sends its iSCSI Initiator Name. The alias is not used authentication, but may be kept with the session information fo display through a management GUI or command-line interface (for more complex subsystem or gateway), or through the iSCSI MIB.

Note that a simple target can just ignore the Initiator Alias

if it has no management interface on which to display it.

Usually just the hostname would be sufficient for an initiator
alias, but a custom alias could be configured for the sake of t
service provider if needed. Even better would be a description
what the machine was used for, such as "Exchange Server 1", or
"User Web Server".

Here's an example of a management interface showing a list of se
on an iSCSI target network entity.
For this display, the targets are using an internal target numb
which is a fictional field that has purely internal significanc

```
+--Connected-To-These-Initiators-------------------
|
|  Target   Initiator Name
|
|  450      iqn.1995-04.com.sw.cd.12345678-OEM-456
|  451      iqn.1995-04.com.os.hostid.A598B45C
|  309      iqn.1995-04.com.sw.cd.87654321-OEM-259
|
+-------------------------------------------------
```

And with the initiator alias displayed:

```
+--Connected-To-These-Initiators-------------------
|
|  Target   Alias                Initiator Name
|
|  450    Web Server 4         iqn.1995-04.com.sw.cd.12345678-O
|  451    scsigate.yours.com   iqn.1995-04.com.os.hostid.A598B4
|  309    Exchange Server      iqn.1995-04.com.sw.cd.87654321-O
|
+-------------------------------------------------
```

This gives the storage administrator a better idea of who is
connected to their targets. Of course, one could always do
a reverse DNS lookup of the incoming IP address to determine
a host name, but simpler devices really don't do well with that
particular feature due to blocking problems, and it won't
always work if there is a firewall or iSCSI gateway involved.

Again, these are purely informational and optional and requir
management application.

Aliases are extremely easy to implement. Targets just send a TargetAlias whenever they send a TargetName. Initiators just send an InitiatorAlias whenever they send an InitiatorName. If an alias is received that does not fit, or seems invalid in any way, it is ignored.

2.5. Initiator and Target Requirements for iSCSI Name support:

Each initiator and target implementation must support the use o iSCSI names.

The initiator MUST send an InitiatorName and a TargetName as te fields within the initial login request on all connections withi session.

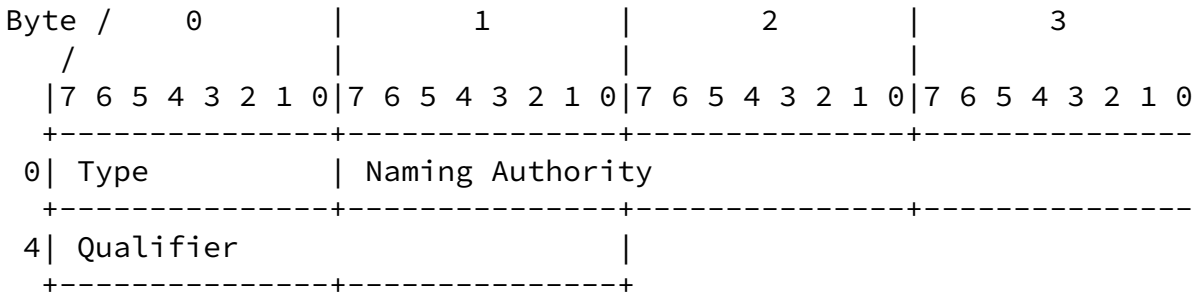
Initiators and targets shall support the receipt of iSCSI names up to the maximum length. If configuration of the initiator or target name is allowed, the implementation shall support the ma length.

In their user interfaces, both shall support, at a minimum, the display of the ASCII characters within the iSCSI Name's UTF-8 string.

If the other characters are unsupported, they may be displayed escape codes as specified in [RFC 2396].

3. ISID

The ISID used in the iSCSI protocol during login (see iSCSI [7]) and as part of the name and identification of SCSI Initiator Ports is specified as a structured field of the following format:



The Type field identifies the format of the Naming Authority field See 2.1.

The Naming Authority field identifies the vendor or organization that is generating this ISID. See 2.2.

The Qualifier field is a 16 bit value that provides a range of possible values for the ISID within the Type and Naming Authority namespace. See 2.3.

The purpose of this structured field is to allow a vendor to implement algorithms for generating, using and reusing ISIDs in their iSCSI components in a manner independent of other vendors' components that may also be present in an iSCSI Initiator. In thi way, compliance to the ISID RULE (see iSCSI [7]) can be managed by each vendor independently.

3.1 ISID Type

The Type field takes on valued defined in [iSCSI]. This is summarized in the following table:

Type	naming authority format
00h	IEEE OUI
01h	IANA Enterprise Number (EN)
02h	"Random"
03h-FFh	Reserved

The first two types provide a mechanism to uniquely (world wide) identify the naming authority (the name of the vendor whose components are generating this ISID). A vendor with one or more OUIs and/or one or more Enterprise Numbers must use at least one o these numbers when it generates ISIDs.

The "Random" type is for the case where the component that generates an ISID (SW or HW) is provided by an entity that has no OUI or EN. This includes, for example,
- a user-written program that builds sessions (and has access to the system level iSCSI Name)
- a university or other organization providing the component
- a testing tool

3.2 ISID Naming Authority

If the Type field is 00h, the Naming Authority field must be set t one of the OUI values assigned to the vendor whose component is generating this ISID. The OUI is set in the Naming Authority fiel in network byte order (BigEndian).

If the Type field is 01h, the Naming Authority field must be set t one of the IANA Enterprise Numbers assigned to the vendor whose component is generating this ISID. The Enterprise Number is set i

the Naming Authority field as a 24bit unsigned integer value in network byte order (BigEndian).

If the Type field is 02h, the Naming Authority field should be set to a random or pseudo-random 24bit unsigned integer value in network byte order (BigEndian). (See 2.4 on how this affects the principle of "conservative reuse").

3.3 ISID Qualifier

The Qualifier field is a 16bit unsigned integer value that may be set by the component generating this ISID to any value, within the constraints specified in the iSCSI protocol (see iSCSI [7] and 2.4).

3.4 Conservative reuse of ISIDs

The principle of "conservative reuse" of ISIDs (see iSCSI [7]) specifies that ISIDs should be reused as much as possible. This principle is there to both minimize the disruption of legacy applications and to better facilitate the SCSI features that rely on persistent names for SCSI ports.

To facilitate conservative reuse, the Qualifier field of a set of ISIDs should be generated using either a repeatable algorithm (e.g deterministic or pseudo-random but based on a fixed seed) or any algorithm to initialize a value or set of values but stored in a persistent location (e.g., registry or /etc file).

For the "Random" type, conservative reuse may not be an issue (e.g., in a user application that doesn't care about reservations, etc.). When it is an issue, the Naming Authority field should als be generated by a mechanism similar to that for the Qualifier fiel as specified above (e.g., defined in the SW at compilation time.)

3.5 Notes on ISIDs

(a) As noted, the structure of the ISID namespace provides each vendor with its own piece of the ISID namespace. In effect, this provides for a vendor-partitioning of that namespace within each initiator. An initiator will then fail to comply with the ISID RULE only if a vendor fails to implement the ISID generation use and reuse requirements correctly.

(b) This structure also allows for a consortium of companies to

develop common APIs or a common infrastructure for generation, use
and reuse of ISIDs. The consortium could, for example, select an
OUI from amongst the member companies to be used in the naming
authority field. Or, the consortium could request an IANA

Enterprise Number for the consortium itself and use this in the
naming authority field. Eventually, the OS implementers could
provide such APIs, in which case the OS vendor could use its own
OUI or EN in the naming authority. In short, the design allows fo
a migration path from vendor-fragmented implementations to
coordinated common implementations for ISID generation.

(c) ISIDs have no global uniqueness requirements or properties.
That is handled by the iSCSI Name of the initiator. This means
that a vendor can use the same algorithm to generate ISIDs (under
its naming authority) in every initiator.

4. iSCSI Discovery

The goal of iSCSI discovery is to allow an initiator to find th
targets to which it has access, and at least one address at whi
each target may be accessed. This should generally be done usin
little configuration as possible. This section defines the
discovery mechanism only; no attempt is made to specify central
management of iSCSI devices within this document. Moreover, iS
discovery mechanism only deals with target discovery and one st
needs to use the SCSI protocol for LUN discovery.

In order for an iSCSI initiator to establish an iSCSI session w
an iSCSI target, the initiator needs the IP address, TCP port
number and iSCSI target name information. The goal of iSCSI
discovery mechanism is to provide low overhead support for smal
iSCSI setups, and scalable discovery solutions for large enterp
setups. Thus, there are several methods that may be used to fin
targets ranging from configuring a list of targets and addresse
each initiator and doing no discovery at all, to configuring no
on each initiator, and allowing the initiator to discover targ
dynamically. The various discovery mechanisms differ in their
assumptions about what information is already available to the
initiators and what information needs to be still discovered.

iSCSI supports the following discovery mechanisms:

a. Static Configuration: This mechanism assumes that the IP add

TCP port and the iSCSI target name information are already avai
to the initiator. The initiators need to perform no discovery
in this approach. The initiator uses the IP address and the TCP
information to establish a TCP connection, and it uses the
iSCSI target name information to establish an iSCSI session. Th
discovery option is convenient for small iSCSI setups.

b. SendTargets: This mechanism assumes that the IP address and
port information are already available to the initiator. The
initiator then uses this information to establish a discovery s
to the Network Entity. The initiator then subsequently issues
SendTargets text command to query information about the iSCSI
targets available at the particular Network Entity (IP address)
SendTargets command details can be found in the iSCSI draft [7]
This discovery option is convenient for iSCSI gateways and rout

c. Zero-Configuration: This mechanism assumes that the initiator
does not have any information about the target. In this option, t
initiator can either multicast discovery messages directly to the
targets or it can send discovery messages to storage name servers
Currently, there are many general purpose discovery frameworks
available such as Salutation[2], Jini[2],UPnP[2], SLP[17] and iSN
However, with respect to iSCSI, SLP can clearly perform the neede
discovery functions [21], while iSNS [8] can be used to provide r
management functions including notification, access management,
configuration, and discovery management. iSCSI equipment that
need discovery functions beyond SendTargets should at least imple
SLP, and then consider iSNS when extended discovery management
capabilities are required such as in larger storage networks.
It should be noted that since iSNS will support SLP, iSNS can
be used to help manage the discovery information returned by SLP.

Appendix A: iSCSI Name Notes

Some iSCSI Name Examples for Targets

- Assign to a target based on controller serial number

iqn.2001-04.com.acme.diskarray.sn.8675309

- Assign to a target based on serial number

  iqn.2001-04.com.acme.diskarray.sn.8675309.oracle_database_1

Where oracle_database_1 might be a target label assigned by a u

This would be useful for a controller that can present differen

logical targets to different hosts.

Obviously, any naming authority may come up with its own scheme
hierarchy for these names, and be just as valid.

A target iSCSI Name should never be assigned based on interface
hardware, or other hardware that can be swapped and moved to ot
devices.

Some iSCSI Name Examples for Initiators

- Assign to the OS image by fully qualified host name

  iqn.2001-04.com.osvendor.dns.com.customer1.host-four

Note the use of two FQDNs - that of the naming
authority and also that of the host that is being
named. This can cause problems, due to limitations
imposed on the size of the iSCSI Name.

- Assign to the OS image by OS install serial number

  iqn.2001-04.com.osvendor.newos5.12345-OEM-0067890-23456

Note that this breaks if an install CD is used more than once.
Depending on the O/S vendor's philosophy, this might be a featu

- Assign to the Raid Array by a service provider

  iqn.2001-04.com.mydisk.users.mbakke05657

Appendix B: iSCSI Proxies and Firewalls Taxonomy

iSCSI has been designed to allow SCSI initiators and targets to
communicate over an arbitrary network. This, making some assum
about authentication and security, means that in theory, the wh
internet could be used as one giant storage network.

However, there are many access and scaling problems that would
up when this is attempted.

1. Most iSCSI targets may only meant to be accessed by one or a
initiators. Discovering everything would be unnecessary.

2. The initiator and target may be owned by separate entities,
with their own directory services, authentication, and other sc
An iSCSI-aware proxy may be required to map between these thing

3. Many environments use non-routable IP addresses, such as the
network.

For these and other reasons, various types of firewalls and pro
will be deployed for iSCSI, similar in nature to those already
handling protocols such as HTTP and FTP.

B.1. Port Redirector

A port redirector is a stateless device that is not aware of iS
It is used to do Network Address Translation (NAT), which can m
addresses between routable and non-routable domains, as well as
TCP ports. While devices providing these capabilities can ofte
filter based on IP addresses and TCP ports, they generally do n
provide meaningful security, and are used instead to resolve in
network routing issues.

Since it is entirely possible that these devices are used as ro
and/or aggregators between a firewall and an iSCSI initiator or
target, iSCSI connections must be operable through them.

Effects on iSCSI:

- iSCSI-level data integrity checks must not include informat
from the TCP or IP headers, as these may be changed in betw
the initiator and target.

- iSCSI messages that specify a particular initiator or targe
such as login requests and third party requests, should spe
the initiator or target in a location-independent manner.

is accomplished using the iSCSI Name.

B.2. SOCKS server

A SOCKS server can be used to map TCP connections from one netw
domain to another. It is aware of the state of each TCP connec

The SOCKS server provides authenticated firewall traversal for
applications that are not firewall-aware. Conceptually, SOCKS

"shim-layer" that exists between the application (i.e., iSCSI)
TCP.

To use SOCKS, the iSCSI initiator must be modified to use the
encapsulation routines in the SOCKS library. The initiator the
up a TCP connection to the SOCKS server, typically on the canon
SOCKS port 1080. A sub-negotiation then occurs, during which t
initiator is either authenticated or denied the connection requ
If authenticated, the SOCKS server then opens a TCP connection
iSCSI target using addressing information sent to it by the ini
in the SOCKS shim. The SOCKS server then forwards iSCSI comman
data, and responses between the iSCSI initiator and target.

Use of the SOCKS server requires special modifications to the i
initiator. No modifications are required to the iSCSI target.

As a SOCKS server can map most of the addresses and information
contained within the IP and TCP headers, including sequence num
its effects on iSCSI are identical to those in the port redirec

B.3. SCSI gateway

This gateway presents logical targets (iSCSI Names) to the
initiators, and maps them to real iSCSI targets as it chooses.
initiator sees this gateway as a real iSCSI target, and is unaw
any proxy or gateway behavior. The gateway may manufacture its
iSCSI Names, or use those provided by the real devices. This t
gateway is used to represent parallel SCSI, Fibre Channel, SSA,
other devices as iSCSI devices.

Effects on iSCSI:

- Since the initiator is unaware of any addresses beyond the ga
  the gateway's own address is for all practical purposes the r
  address of a target. Only the iSCSI Name needs to be passed.
  is already done in iSCSI, so there are no further requirement

support SCSI gateways.

B.4. iSCSI Proxy

An iSCSI proxy is a SCSI gateway that happens to be terminating the iSCSI protocol on both sides, rather than translate between iSCSI and some other transport. Since an iSCSI initiator's discovery or configuration of a set of targets makes use of address-independent iSCSI names, iSCSI does not have the same proxy addressing problems as HTTP, which includes address

information into its URLs. If a proxy is to provide services to an initiator on behalf of a target, the proxy allows the initiator to discover its address for the target, and the actua target device is discovered only by the proxy. Neither the initiator nor the iSCSI protocol needs to be aware of the existence of the proxy.

Effects on iSCSI:

- Same as a SCSI gateway. The only other effect is that iSCSI must separate data integrity checking on iSCSI headers and iSCSI data, to allow the data integrity check on the data to be propagated end-to-end through the proxy.

B.5. Stateful Inspection Firewall (stealth iSCSI firewall)

The Stealth model would exist as an iSCSI-aware firewall, that invisible to the initiator, but provides capabilities found in iSCSI proxy.

Effects on iSCSI:

- Since this is invisible, there are no additional requirements on the iSCSI protocol for this one.

This one is more difficult in some ways to implement, simply be it has to be part of a standard firewall product, rather than p an iSCSI-type product.

Also note that this type of firewall is only effective in the outbound direction (allowing an initiator behind the firewall to connect to an outside target), unless the iSCSI tar is located in a DMZ. It does not provide adequate security otherwise.

Appendix C

This document has described the creation and use of iSCSI Node Nam
There will be trusted environments where this is a sufficient form
identification. In these environments the iSCSI Target may have an
Access Control List (ACL), which will contain a list of authorized
entities that are permitted to access a restricted resource (in th
case a Target Storage Controller). The iSCSI Target will then use
that ACL to permit (or not) certain iSCSI Initiators to access the

storage at the iSCSI Target Node. This form of ACL is used to prev
trusted initiators from making a mistake and connecting to the wro
storage controller.

It is also possible that the ACL and the iSCSI Initiator Node Name
can be used in conjunction with the SCSI layer for the appropriate
SCSI association of LUNs with the Initiator. The SCSI layer's use
of the ACL will not be discussed further in this document.

There will be situations where the iSCSI Nodes exist in untrusted
environments. That is, some iSCSI Initiator Nodes may be authorize
to access an iSCSI Target Node, however, because of the untrusted
environment, nodes on the network cannot be trusted to give the
correct iSCSI Initiator Node Names.

In untrusted environments an additional type of identification is
required to assure the target that it really knows the identity
of the requesting entity.

The authentication and authorization in the iSCSI layer is
independent of anything that IPSec might handle, underneath
or around the TCP layer. This means that the initiator node
needs to pass some type of security related identification
information (e.g. userid) to a security authentication process
such as SRP, CHAP, Kerberos etc. (These authentication processes
will not be discussed in this document).

Upon the completion of the iSCSI security authentication, the
installation knows "who" sent the request for access. The
installation must then check to ensure that such a request,
from the identified entity, is permitted/authorized. This
form of Authorization is generally accomplished via an Access
Control List (ACL) as described above. Using this authorization
process, the iSCSI target will know that the entity is authorized
access the iSCSI Target Node.

It may be possible for an installation to set a rule that the secu identification information (e.g. UserID) be equal to the iSCSI Initiator Node Name. In that case, the ACL approach described abov should be all the authorization that is needed.

If, however, the iSCSI Initiator Node Name is not used as the secu identifier there is a need for more elaborate ACL functionality. T means that the target requires a mechanism to map the security identifier (e.g. UserID) information to the iSCSI Initiator Node N That is, the target must be sure that the entity requesting access

authorized to use the name, which was specified with the Login Key "InitiatorName=".
For example, if security identifier 'Frank' is authorized to acces the target via iSCSI InitiatorName=xxxx, but 'Frank' tries to acce target via iSCSI InitiatorName=yyyy, then this login should be rej

On the other hand, it is possible that 'Frank' is a roaming user (Storage Administrator) that "owns" several different systems, and could be authorized to access the target via multiple different iS initiators. In this case, the ACL needs to have the names of all t initiators through which 'Frank' can access the target.

There may be other more elaborate ACL approaches, which can also b deployed to provide the installation/user with even more security flexibility.

The above discussion is trying to inform the reader that, not only there a need for access control dealing with iSCSI Initiator Node but in certain iSCSI environments there might also be a need for o complementary security identifiers.

5. References

[1] Pascoe, R., "Building Networks on the Fly", in IEEE Spectrum,March, 2002.

[2] John, R., "UPnP, Jini and Salutation- A look at some popu coordination frameworks for future networked devices", http://www.cswl.com/whiteppr/tech/upnp.html", June 17, 19

[3] http://www.srvloc.org

[4] Freed, N., "Behavior of and Requirements for Internet Firewalls", RFC 2979, October 2000.

[5] ANSI/IEEE Std 802-1990, Name: IEEE Standards for Local an
Metropolitan Area Networks: Overview and Architecture

[6] Kessler, G. and Shepard, S., "A Primer On Internet and TC
Tools and Utilities", RFC 2151, June 1997.

[7] Satran, J., Sapuntzakis, C., Wakeley, M., Von Stamwitz, P
Haagens, R., Chadalapaka, M., Zeidner, E., Dalle Ore, L.,
Y., "iSCSI", draft-ietf-ips-iscsi-07.txt, July, 2001.

[8] Gibbons, K., Tseng, J. and Monia, C., "iSNS Internet Stor

Name Service", draft-tseng-ips-isns-04.txt, July 2001.

[9] RFC 1737, "Functional Requirements for Uniform Resource N

[10] RFC 1035, "Domain Names - Implementation and Specificatio
OUI - "IEEE OUI and Company_Id Assignments",
http://standards.ieee.org/regauth/oui/index.shtml

[11]EUI - "Guidelines for 64-bit Global Identifier (EUI-64)
Registration Authority
http://standards.ieee.org/regauth/oui/tutorials/EUI64.htm

[12] RFC 2396, "Uniform Resource Identifiers".

[13] RFC 2276, "Architectural Principles of URN Resolution".

[14] RFC 2483, "URI Resolution Services".

[15] RFC 2141, "URN Syntax".

[16] RFC 2611, "URN Namespace Definition Mechanisms".

[17] RFC 2608, SLP Version 2.

[18] RFC 2610, DHCP Options for the Service Location Protocol.

[19] P. Sarkar et al, "A Standard for Bootstrapping Clients us
the iSCSI Protocol", draft-ietf-ips-iscsi-boot-03.

[21] M. Bakke et al,"Finding iSCSI Targets and Name Servers us
SLP", draft-ietf-ips-iscsi-slp-01.txt, July, 2002.

[22] Sun Microsystems, "Java Language Specification", section
"Unique Package Names", 2000,

http://java.sun.com/docs/books/jls/second_edition/html/jTOC.doc.html.

[23] Flanagan, et. al, "Java in a Nutshell", O'Reilly, 1997.

[24] P. Hoffman, M. Blanchet, "Preparation of Internationalize Strings", draft-hoffman-stringprep-00.txt, September, 200

[25] Unicode Standard Annex #15, "Unicode Normalization Forms" http://www.unicode.org/unicode/reports/15

[26] M. Bakke, "String Profile for iSCSI Names",

draft-ietf-ips-iscsi-string-prep-00.txt, November 2001.

6. Author's Addresses

Address comments to:

Kaladhar Voruganti
650 Harry Road
IBM Almaden Research
San Jose, CA
USA
Email: kaladhar@us.ibm.com

Mark Bakke
Cisco Systems, Inc.
6450 Wedgwood Road
Maple Grove, MN 55311
Phone: +1 763 398-1054
Email: mbakke@cisco.com

Jim Hafner
IBM Research
Almaden Research Center
650 Harry Road
San Jose, CA 95120
Phone: +1 408-927-1892
Email: hafner@almaden.ibm.com

Joe Czap
IBM Corp.
600 Park Office Drive

RTP, NC 27709
Phone: +1 919 254-0828
Email: zapper@us.ibm.com

Josh Tseng
Nishan Systems
3850 North First Street
San Jose, CA 95134
Phone: 408 519-3749
Email: jtseng@nishansystems.com

Lawrence J. Lamers
SAN Valley Systems, Inc.

2105 South Bascom Avenue
Campbell, CA 95008
Phone: 408.234.0071
Email: ljlamers@ieee.org

Marjorie Krueger
Hewlett-Packard Corporation
8000 Foothills Blvd
Roseville, CA 95747-5668, USA
Phone: +1 916 785-2656
Email: marjorie_krueger@hp.com

Todd Sperry
Adaptec, Inc.
691 South Milpitas Boulevard
Milpitas, Ca. 95035
Phone: (408) 957-4980
Email: todd_sperry@adaptec.com