IPS
Internet Draft
Document: draft-ietf-ips-iscsi-boot-07.txt
Category: Standards
Prasenjit Sarkar
IBM
Duncan Missimer
Rhapsody Networks
Constantin Sapuntzakis
Stanford University
24 September 2002

Bootstrapping Clients using the iSCSI Protocol

Status of this Memo

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

Abstract

The Small Computer Systems Interface (SCSI) is a popular family of protocols for communicating with I/O devices, especially storage devices. iSCSI is a proposed transport protocol for SCSI that operates on top of TCP[12]. This memo describes a standard mechanism to enable clients to bootstrap themselves using the iSCSI protocol. The goal of this standard is to enable iSCSI boot clients to obtain the information to open an iSCSI session with the iSCSI boot server, assuming this information is not available.

1. Requirements

1. There must be no restriction of network topology between the iSCSI boot client and the boot server other than those in effect for establishing the iSCSI session. Consequently, it is possible for an

iSCSI boot client to boot from an iSCSI boot server behind gateways or firewalls as long as it is possible to establish an iSCSI session between the client and the server.

2. The following represents the minimum information required for an iSCSI boot client to contact an iSCSI boot server: (a) the client's IP address (IPv6 or IPv4); (b) the server's iSCSI Target Name; and (c) mandatory iSCSI initiator capability.

The above assumes that the default LUN for the boot process is 0 and the default port for the iSCSI boot server is the well-known iSCSI port. However, both may be overridden at the time of configuration.

Additional information may be required at each stage of the boot process.

3. It is possible for the iSCSI boot client to have none of the above information or capability on starting.

4. The client should be able to complete boot without user intervention (for boots that occur during an unattended power-up). However, there should be a mechanism for the user to input values so as to bypass stages of the boot protocol.

5. Additional protocol software (for example, DHCP) may be necessary if the minimum information required for an iSCSI session is not provided.

2. Related Work

The Reverse Address Resolution Protocol (RARP)[7](through the extensions defined in the Dynamic RARP (DRARP))[4] explicitly addresses the problem of network address discovery, and includes an automatic IP address assignment mechanism. The Trivial File Transfer Protocol (TFTP)[9] provides for transport of a boot image from a boot server. BOOTP[5,8,10] is a transport mechanism for a collection of configuration information. BOOTP is also extensible, and official extensions have been defined for several configuration parameters. DHCPv4[3,6] and DHCPv6[13] are standards for hosts to be dynamically configured in an IP network. The Service Location Protocol (SLP) provides for location of higher level services[1,15].

3. Software stage

Some iSCSI boot clients may lack the resources to boot up with the mandatory iSCSI initiator capability. Such boot clients may choose to obtain iSCSI initiator software from a boot server. Currently, there are many established protocols that allow such a service to enable

clients to load software images. For example, BOOTP and DHCP servers have the capability to provide software images on requests from boot clients. A particular implementation of this approach is the PXE protocol[17], which uses DHCP extensions and MTFTP to allow boot clients to load software images.

It is to be noted that this document does not recommend any of the above protocols, and the final decision of which boot protocol is to be used to load iSCSI initiator software is left to the discretion of the implementor.

4. DHCP stage

In order to use an iSCSI boot server, the following pieces of information are required for an ISCSI boot client.

- The IP address of the iSCSI boot client (IPv4 or IPv6)

- The IP transport endpoint for the iSCSI Target Port for the iSCSI boot server. If the transport is TCP, for example, this has to resolve to an IP address and a TCP port number. TCP is currently the only transport approved for iSCSI.

- The eight-byte LUN structure identifying the Logical Unit within the iSCSI boot server.

At boot time, all or none of this information may be stored in the iSCSI boot client. This section describes techniques for obtaining the required information via the DHCP stage. Otherwise, if the iSCSI boot client has all the information, the boot client may proceed directly to the Boot stage.

An iSCSI boot client which does not know its IP address at power-on may acquire its IP address via DHCP. An iSCSI boot client which is capable of using both DHCPv6 and DHCPv4 should first attempt to use DHCPv6 to obtain its IP address, falling back on DHCPv4 in the event of failure.

Unless otherwise specified here, DHCP fields such as the client ID and gateway information are used in an identical way as applications other than iSCSI do.

A DHCP server (v4 or v6) MAY instruct an iSCSI client how to reach its boot device. This is done using the variable length DHCP option named Root Path. The use of the option field is reserved for iSCSI boot use by prefacing the string with "iscsi:".

The option field consists of an UTF-8[8] string. The string MUST contain only alphanumeric characters, "." , ":" and "-"; no other characters are permissible. The string has the following composition:

"iscsi:"<servername>":"<protocol>":"<port>":"<LUN>":"<targetname>

The fields "servername", "port", "protocol" and "LUN" are OPTIONAL and should be left blank if there are no corresponding values. The "targetname" field is not optional and MUST be provided.

The "servername" is the name of iSCSI server and contains either a valid domain name, a literal IPv4 address, or a literal IPv6 address.

If the "servername" field contains a literal IPv4 address, the IPv4 address MUST be in standard dotted decimal notation as defined in Section 2.1 of RFC 1123[6].

If the "servername" field contains an IPv6 address, the address MUST be represented in the IPv6 address format x.x.x.x.x.x.x.x where the 'x's are the hexadecimal values of the eight 16-bit pieces of the address. Note that this format representation is specific to iSCSI boot.

If the "servername" is a domain name, the name MUST be a fully qualified domain name (FQDN) and should abide by the rules specified in Sections 3.1 and 3.5 of RFC 1034[7] and the reply from the host configuration server should contain the Domain Name Server Option[1]. It must also be pointed out that the use of DNS for address translation in enterprise environments must contain adequate levels of fault tolerance and security.

If the "servername" field contains 4 decimal components, the "servername" is assumed to be an IPv4 address. If there are more than 4 decimal components or if there is a hexadecimal component, the the "servername" is assumed to be an IPv6 address. If the least significant (rightmost) component is an approved domain extension, then the "servername" field is assumed to be a domain name. If the "servername" field is left blank, then no default value is assumed in its place.

The "protocol" field is the decimal representation of the IANA-approved string for the trasport protocol to be used for iSCSI. If the protocol field is left bank, the default value is assumed to be "6" for TCP. The transport protocol MUST have been approved for use in iSCSI; currently, the only approved protocol is TCP.

The "port" is the decimal representation of the port on which the iSCSI boot server is listening. If not specified, the port defaults

to the well-known iSCSI port.

The "LUN" field is a hexadecimal representation of the LU number. If the LUN field is blank, then LUN 0 is assumed. If the LUN field is not blank, the representation MUST be divided into four groups of four hexadecimal digits, separated by "-". Digits above 9 may be either lower or upper case. An example of such a representation would be 4752-3A4F-6b7e-2F99. For the sake of brevity, at most three leading zero ("0") digits MAY be omitted in any group of hexadecimal digits. Thus, the "LUN" representation 6734-9-156f-127 is equivalent to 6734-0009-156f-0127. Furthermore, trailing groups containing only the "0" digit MAY be omitted along with the preceding "-". So, the "LUN" representation 4186-9 is equivalent to 4186-0009-0000-0000. Other concise representations of the LUN field MUST NOT be used.

Note that SCSI targets are allowed to present different LU numberings for different SCSI initiators, so that to our knowledge nothing precludes a SCSI target from exporting several different LUs to several different SCSI initiators as their respective LUN 0s.

The "targetname" field is an iSCSI Name that is defined by the iSCSI standard[4] to uniquely identify an iSCSI target.

If the "servername" field is provided by DHCP, then that field is used in conjunction with other associated fields to contact the boot server in the Boot stage (Section 6). However, if the "servername" field is not provided, then the "targetname" field is then used in the Discovery Service stage in conjunction with other associated fields. (Section 5).

5. Discovery Service stage

This stage is required if the DHCP server (v4 or v6) is unaware of any iSCSI boot servers or if the DHCP server is unable to provide the minimum information required to connect to the iSCSI boot server other than the targetname.

The discovery service is based on the SLP protocol[1,24] and is an instantiation of the SLP Service or Directory Agent.

The iSCSI boot client may have obtained the targetname of the iSCSI boot server in the DHCP stage (Section 4). In that case, the iSCSI boot client queries the Discovery Service using query string 1 of the iSCSI Target Concrete Service Type Template as specified in Section 6.2 of the iSCSI SLP interaction document[24] to resolve the targetname to an IP address and port number. Once this is obtained, the iSCSI boot client proceeds to the Boot stage (Section 6).

It is possible that the port number obtained from the Discovery Service may conflict with the one obtained from the DHCP service. In such a case, the implementor has the option to try both port numbers in the Boot stage.

If the iSCSI boot client does not have any targetname information, the iSCSI boot client then may query the Discovery Service with query string 4 of the iSCSI Target Concrete Service Type Template as specified in Section 6.2 of the iSCSI SLP interaction document[24]. In response to this query, the discovery service provides the boot client with a list of iSCSI boot servers the boot client is allowed to access.

If the list of iSCSI boot servers is empty, subsequent actions are left to the discretion of the implementor. Otherwise, the iSCSI boot client may contact any iSCSI boot server in the list. Moreover, the order in which iSCSI boot servers are contacted is also left to the discretion of the implementor.

6. Boot stage

Once the iSCSI boot client has obtained the minimum information to open an iSCSI session with the iSCSI boot server, the actual booting process can start.

The actual sequence of iSCSI commands needed to complete the boot process is left to the implementor. This was done because of varying requirements from different vendors and equipment, making it difficult to specify a common subset of the iSCSI standard that would be acceptable to everybody.

The iSCSI session established for boot may be taken over by the booted software in the iSCSI boot client.

7. Security

The security discussion is centered around each stage of the iSCSI boot process.

The software stage can be secured by using public key encryption and digitial signatures. This is the approach taken by the popular PXE boot framework.

With regards to the DHCP stage, securing the host configuration protocol is beyond the scope of this document. Authentication of DHCP messages is described in [16].

The security issues in the Discovery Service stage are addressed by public key ciphering as stated in the the SLP version 2 document[1].

For the Boot stage, the iSCSI standard supports various methods of authentication and encryption for transport security[12]. The means to configure the security parameters of an iSCSI boot client is beyond the scope of this document.

The iSCSI boot service may be subjected to denial of service attacks. The use of IPSEC as mandated by the iSCSI standard[12] can be used to protect against such attacks. However, ARP is still vulnerable to such type of attacks.

Security in the Boot stage is also dependent on the verification of the boot image being loaded. One key difference between the iSCSI boot mechanism and BOOTP-based image loading is the fact that the identity of a boot image may not be known when the Boot stage starts. The identity of certain boot images and their locations are known only after examining the contents of a boot disk exposed by the iSCSI boot service. Furthermore, images themselves may recursively load other images based on both hardware configurations and user input.

Consequently, a practical way to verify loaded boot images is to make sure that each image loading software verify the image to be loaded using a mechanism of their choice.

Another point to be noted is that if a boot image inherits an iSCSI session from a previously loaded boot image, the boot image also inherts the security properties of the iSCSI session.

Acknowledgments

We wish to thank John Hufferd for taking the initiative to form the iSCSI boot team. We also wish to thank Doug Otis, Julian Satran, Bernard Aboba, David Robinson, Mark Bakke and Mallikarjun Chadalapaka for helpful suggestions and pointers regarding the draft document.

References

[1] Guttman, E., Perkins, C., Verizades, J., Day, M., "Service Location Protocol v2", RFC 2608, June 1999.

[2] Alexander, S., and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, Lachman Technology, Inc., Bucknell University, October 1993.

[3] R. Droms, "Dynamic Host Configuration Protocol", RFC 2131, Bucknell University, March 1997.

[4] Brownell, D, "Dynamic Reverse Address Resolution Protocol (DRARP)", Work in Progress.

[5] Croft, B., and J. Gilmore, "Bootstrap Protocol (BOOTP)", RFC 951, Stanford and SUN Microsystems, September 1985.

[6] Droms, D., "Interoperation between DHCP and BOOTP" RFC 1534, Bucknell University, October 1993.

[7] Finlayson, R., Mann, T., Mogul, J., and M. Theimer, "A Reverse Address Resolution Protocol", RFC 903, Stanford, June 1984.

[8] Reynolds, J., "BOOTP Vendor Information Extensions", RFC 1497, USC/Information Sciences Institute, August 1993.

[9] Sollins, K., "The TFTP Protocol (Revision 2)", RFC 783, NIC, June 1981.

[10] Wimer, W., "Clarifications and Extensions for the Bootstrap Protocol", RFC 1532, Carnegie Mellon University, October 1993.

[11] Bradner, S., "The Internet Standards Process -- Revision 3", RFC 2026, October 1996.

[12] Satran, J. et al., "iSCSI", Internet-Draft, September 2002.

[13] Bound, J., Canney, M., and Perkins, C., "Dynamic Host Configuration Protocol for IPv6", Internet-Draft, June 2002.

[14] Bakke, M. et al., "iSCSI Naming and Discovery", Internet-Draft, July 2002.

[15] Veizades, J., Guttman, E., Perkins, C., Kaplan, S., "Service Location Protocol", RFC 2165, June 1997.

[16] Droms, R., Arbaugh, W., "Authentication for DHCP Messages", RFC 3118, June 2001.

[17] Intel Corp., "Boot Integrity Services", http://www.intel.com/labs/manage/wfm/tools/bis

[18] Stewart, R., et al. "Stream Control Transmission Protocol", RFC 2960, October 2000.

[19] Droms, R., "Procedures and IANA Guidelines for Approval of New DHCP Options and Message Types", RFC 2939, September 2000.

[20] Yergeau, F., "UTF-8: A Transformation Format for ISO-10646", RFC 2279, January 1998.

[21] Hinden, R., Deering, S., "IP version 6 Addressing Architecture", RFC 2273, July 1998.

[22] Braden, R., "Requirements for Internet Hosts - Application and Support", RFC 1123, October 1989.

[23] Mockaopertis, P., "Domain Names - Concepts and Facilities", RFC 1034, November 1987.

[24] Bakke, M., et al. "Finding iSCSI Targets and Name Servers using SLP", Internet-Draft, March 2002.

Authors' Addresses

Prasenjit Sarkar
IBM Almaden Research Center
650 Harry Road
San Jose, CA 95120, USA
Phone: +1 408 927 1417
Email: psarkar@almaden.ibm.com

Duncan Missimer
Rhapsody Networks
3450 W Warren Avenue,
Fremont, CA 94538, USA
Phone: +1 510 743 3095
Email: dmissimer@rhapsodynetworks.com

Constantine Sapuntzakis
Stanford University
353 Serra Hall #406
Stanford, CA 94306, USA
Phone: +1 650 520 0205
Email: csapuntz@stanford.edu

Full Copyright Statement