        Encapsulating IP with the Small Computer System Interface

Status of this Memo

   This memo defines an Experimental Protocol for the Internet
   community. This memo does not specify an Internet standard of any
   kind. Discussion and suggestions for improvement are requested.
   Distribution of this memo is unlimited.

Table of Contents

1. Introduction

   As the capacity of local area networks increases to meet the demands
   of high volume application data, there is a class of network
   intensive problems which may be applied to small clusters of
   workstations with high bandwidth interconnection.

   A general observation of networks is that the bit rate of the data
   path typically decreases as the distance between hosts increases. It
   is common to see regional networks connected at a rate of 64Kbps and
   office networks connected at 100Mbps, but the inverse is far less
   common.

   The same is true of peripheral and memory interconnection. Memory
   close to a CPU's core may run at speeds equivalent to a gigabit
   network. More importantly, devices such as disks may connect a
   number of metres away from its host at speeds well in excess of
   current local area network capacity.

This document outlines a protocol for connecting hosts running the
TCP/IP protocol suite over a Small Computer System Interface (SCSI)
bus. Despite the limitation in the furthest distance between hosts,
SCSI permits close clusters of workstations to communicate between
each other at speeds approaching 360 megabits per second.

The proposed introduction of newer SCSI implementations such as
serial SCSI will bring much faster communication at greater
distances.

2. Background to the Small Computer System Interface (SCSI)

SCSI defines a physical and data link protocol for connecting
peripherals to hosts. Devices connect autonomously to a bus and send
synchronous or asynchronous messages to other devices.

Devices are identified by a numeric identifier (ID). For the
original SCSI protocol, devices are given a user-selectable SCSI ID
between 0 and 7. Wide SCSI specifies a range of SCSI IDs between 0
and 15. The most typical SCSI configuration comprises of a host
adapter and one or more SCSI- capable peripherals responding to
asynchronous messages from the host adapter.

The most critical aspect of the protocol with respect to its use as a
data link for the Internet protocols is that a SCSI device must act
as an "initiator" (generator of SCSI commands/requests) or a "target"
(a device which responds to SCSI commands from the initiator). This
model is correct for a master/slave relationship between host adapter
and devices. The only time an initiator receives a message addressed
to it is in response to a command issued by it in the past and a
target device always generates a response to every command it
receives.

Clearly this is unsuitable for the peer-to-peer model required for
multiple host adapters to asynchronously send SCSI commands to one
another without surplus bus traffic. Furthermore, some host adapters
may refuse to accept a message from another adapter as it expects to
only initiate SCSI commands. This restriction to the protocol
requires that SCSI adapters used for IP encapsulation support what is
known as "target mode", with software device driver support to pass
these messages up to higher layer modules for processing.

3. Link Encapsulation

   The ANSI SCSI standard defines classes of peripherals which may be
   interconnected with the SCSI protocol. One of these is the class of
   "communication devices" [1].

   The standard defines three message types capable of carrying a
   general-purpose payload across communication devices. Each of these
   are known as the "SEND MESSAGE" message type, but the size and and
   structure of the message header differs amongst them. The three
   forms of message header are six (6), ten (10) and twelve (12) bytes
   long.

   It was decided that the ten byte header offers the greatest
   flexibility for encapsulating version 4 IP datagrams for the
   following reasons:

      a. The transfer length field is 16 bits in size which is perfectly
         matched to the datagram length field in IP version 4.
         Implementations of IP will run efficiently as datagrams will
         never be fragmented over SCSI networks.

      b. The SCSI "stream select" field, which was designed to permit
         a device to specify the stream of data to which a block
         belongs, may be used to encode the payload type (in a similar
         manner to the Ethernet frame type field). For consistency, the
         lowest four bits of the "stream select" field should match the
         set of values assigned by the IEEE for Ethernet protocol types.

   Encapsulating an IP datagram within a SCSI message is
   straightforward:

        +-----------------+----------------------------------+
        | SCSI header | IP datagram |
        +-----------------+----------------------------------+
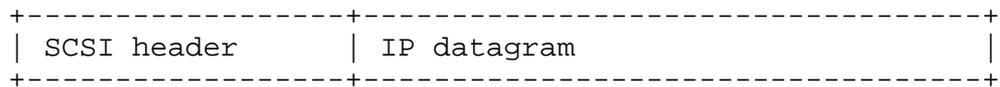
   The fields of the SCSI header should be completed as follows:

        Byte 0:   0x2A (SEND_MESSAGE(10) opcode)
        Byte 1:   Logical unit number encoded into top 3 bits | 0x00
        Byte 2:   0x00
        Byte 3:   0x00
        Byte 4:   0x00
        Byte 5:   Protocol type encoded into lowest 4 bits | 0x00
        Byte 6:   0x00
        Bytes 7/8: IP datagram length, big endian representation
        Byte 9:   0x00

4. An Address Resolution Protocol

   When IP decides that the next hop for a datagram will be onto a SCSI
   network supported by a SCSI IP network interface implementation, it
   is necessary to acquire a data link address to deliver the datagram.

   Network interfaces such as Ethernet have well-known methods for
   acquiring the media address for an Internet protocol address, the
   most common being the Address Resolution Protocol (ARP). In existing
   implementations, the forwarding host "yells" using a broadcast media
   address and expects the named host to respond.

   The SCSI protocol does not provide a broadcast data link address. An
   acceptable solution to the address resolution problem for a SCSI
   network is to simulate a broadcast by performing a series of round-
   robin transmissions to each target. Depending on the SCSI protocol
   being used, this would require upward of seven independent bus
   accesses. This is not grossly inefficient, however, if combined with
   an effective ARP caching policy. A further possible optimisation is
   negative ARP caching, whereby incomplete ARP bindings are not queried
   for some period in the future.

5. Scalability

   While the utility of a network architecture based around a bus
   network which can span less than ten metres and support only eight
   hosts may be questionable, the flexibility of IP and in particular,
   IP routing, improves the scalability of this architecture.

   Consider a network of eight hosts connected to a SCSI bus in which
   each host acts as a multi-homed host with a second host adapter
   connecting another seven hosts to it. When configured with IP packet
   routing capability, each of the 64 total hosts may communicate with
   one another at high speed in a packet switched manner.

   Depending on the I/O bus capabilities of certain workstations, it may
   also be possible to configure a multi-homed host with a greater
   number of SCSI host adapters, permitting centralised star
   configurations to be constructed.

   It should be apparent that for little expense, massively parallel
   virtual machines can be built based upon the IP protocol running over
   the high-bandwidth SCSI protocol.

6. Possible Applications

   Research has been made into the capability of "networks of
   workstations", and their performance compared to supercomputers. An
   observation that has been made thus far is that bottlenecks exist in
   the channels by which executable code is transported between hosts
   for execution. A very high-speed network architecture based around
   the Internet protocol would permit a seamless transition of existing
   application software to a high-bandwidth environment.

   Other applications that have been considered are server clusters for
   fault-tolerant NFS, World-Wide Web and database services.

7. Security Considerations

   Transmitting IP datagrams across a SCSI bus raises similar security
   issues to other local area networking architectures. The scale of
   security problems relating to protocols above the data link layer
   should be obvious to a reader current in Internet security.

8. References

   [1] ANSI X3T9 Technical Committee, "Small Computer System
        Interface - 2", X3T9.2, Project 375D, Revision 10L, September
        1993.

9. Author's Address

   Ben Elliston
   Compucat Research Pty Limited
   Box 7305 Canberra Mail Centre
   Canberra ACT 2610
   Australia

   Phone: +61 6 295 1331
   Fax: +61 6 295 1855
   Email: ben.elliston@compucat.com.au