Evaluation of SCSI Over TCP/IP and SCSI Over Fibre Channel Connections

Huseyin Simitci, Chris Malakapalli, Vamsi Gunturu
XIOtech Corporation
6455 Flying Cloud Dr., Eden Prairie, MN 55344
(Huseyin_Simitci,Chris_Malakapalli,Vamsi_Gunturu)@XIOtech.com

Abstract

This study explores the performance implications of an IP storage protocol (iSCSI) in a storage network. The results are compared with the characteristics of the Fibre Channel protocol. We set up a test-bed consisting of Linux PCs connected together with both Fibre Channel and Gigabit Ethernet adapters. We conducted experiments on user and kernel level TCP/IP communication and iSCSI data transfers over TCP/IP. By instrumenting the network and iSCSI device drivers on the initiator and target machines, we have collected performance data on various aspects of the protocols. Using a prototype SCSI target mode driver on the target machine, we were able to conduct similar experiments with the Fibre Channel protocol and interconnects. We present preliminary performance data for iSCSI and ways to improve the underlying TCP/IP bandwidth on Gigabit Ethernet.

1. Introduction

Availability of high bandwidth, low latency network interconnects, like Fibre Channel (FC) and Gigabit Ethernet (GbE), together with the complexities of managing dispersed islands of data storage, led to the development of Storage Area Networks (SANs).

SANs allow sharing of data storage over long distances and still permit centralized control and management. Currently, FC is the predominant interconnect for SAN implementations. FC allows block level SCSI commands and data to travel far more distances than the Parallel SCSI interconnects.

Lately, Internet Protocol (IP) is advocated as an alternative to transport SCSI traffic over long distances (Wide Area Networks). Proposals like iSCSI [1] try to standardize the encapsulation of SCSI data in TCP/IP (Transmission Control Protocol/Internet Protocol) packets. Once the data is in IP packets, it can be carried over a range of physical network connections. Today, GbE is the de facto standard for high bandwidth local area networks (LANs) and campus networks.

In this paper, we explore the usage of TCP/IP over GbE as the SCSI connection medium, and compare it with the FC networks. To achieve this, we have used a prototype iSCSI implementation, which is constructed as client and server modules inside the Linux operating system kernel. We explore the performance characteristics of the FC interconnect using a prototype SCSI target mode driver. The paper ends with the discussion of our conclusions.

2. Ethernet and TCP analysis

In this section, we present the experimental data on various factors that might affect the performance of the traffic flowing over Ethernet and TCP/IP. These are some of the typical issues one must consider for any type of TCP/IP/Ethernet workload—including iSCSI.

Figure 1 shows the test-bed used in our TCP/IP/Ethernet experiments. The two Linux PCs contain 400MHz PII processors, 64MB memory and NetGear GA620T copper 10/100/1000Mbps Ethernet NICs on 32bit, 33MHz PCI buses. They are connected through a NetGear GS504T copper 100/1000Mbps Ethernet switch. Both PCs run Red Hat Linux version 6.2 and Linux kernel version 2.3.50.

To generate TCP workloads, we used the publicly available TTCP benchmark program. We wrote a similar

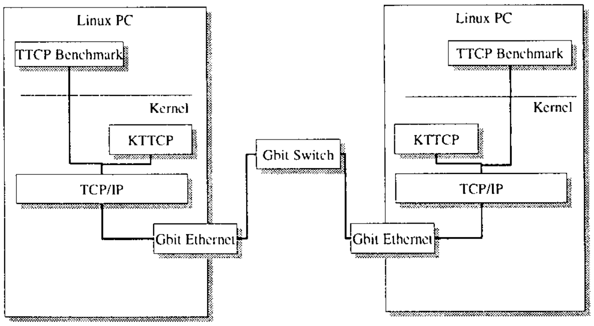

Figure 1. Experimental setup for user level and kernel level TCP benchmarks

0-7695-1357-3/01 $10.00 © 2001 IEEE

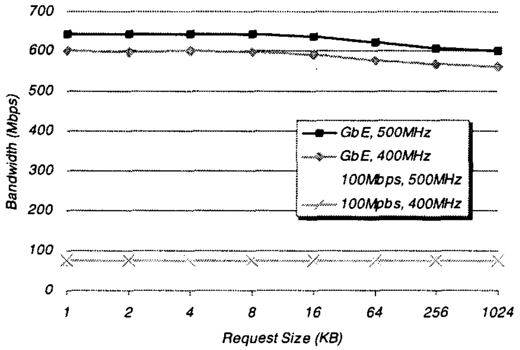

Figure 2. Gigabit Ethernet and 100Mbit Ethernet bandwidth at two different CPU speeds

benchmark program (KTTCP) that can generate similar TCP workloads inside the kernel instead of the user level.

Figure 2 shows the TCP bandwidth for various transfer sizes, and for two host CPU speeds. It shows that it is possible to obtain 650Mbps with GbE (with jumbo frame size) and 75Mbps with 100Mbit Ethernet (with standard frame size). The figure also shows that the CPU speed does not affect the 100Mbit Ethernet bandwidth even though the GbE bandwidth depends on the CPU speed.

GbE is more CPU intensive than 100Mbit Ethernet. On the sending node, over 60 percent of the CPU time is used for Gigabit transfers, while less than 10 percent of the CPU time is used for 100Mbit Ethernet transfers. The CPU usage on the receiving node is more significant. The utilization gets close to 100 percent for GbE and over 30 percent for 100Mbit Ethernet.

Some GbE implementations allow MTU (Maximum Transmission Unit) sizes bigger than the IEEE standard (802.3z) maximum value of 1500 bytes. Jumbo frames defined by the Alteon Networks can be up to 9000 bytes long.

As Figure 3 shows, jumbo frames provides 60 percent bandwidth increase over standard frames. In addition, the CPU utilization is smaller for jumbo frames for transfer sizes of 64KB and higher. We also found out that jumbo frame sizes require bigger socket buffer sizes to be effective.

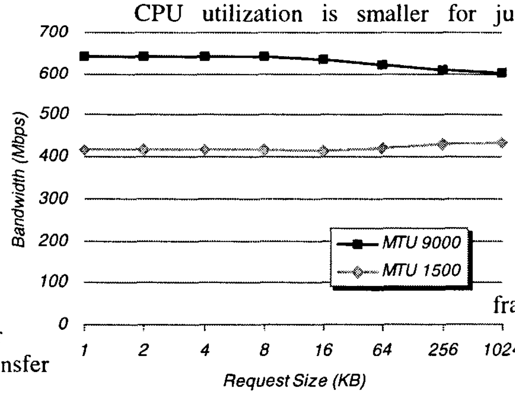

Figure 3. Gigabit Ethernet with standard and jumbo frame sizes (CPU 500MHz)

There is roughly six percent bandwidth difference between user level and kernel level transfers. Our tests showed that there is approximately 1.5 to 2.1 microseconds extra overhead per kilobyte of data for user level transfers. This is due to the buffer copies between the user buffers and the kernel buffers.

We have made similar TTCP experiments on the same systems with Windows NT Server 4 Service Pack 6. With Windows NT, the bandwidth is limited to 180Mbps compared to 435Mbps with Linux. Since the bandwidths are so different, a direct CPU utilization comparison cannot be done. Still, Windows NT uses roughly the same amount of CPU time as Linux, and provides 40 percent of bandwidth of Linux for this hardware and workload combination. Rindos et al. [2] report that using multiple connections, they have observed a bandwidth of 439Mbps with Windows NT.

3. iSCSI performance on Gigabit Ethernet

The same test-bed used for the TCP performance analysis is also used to evaluate the performance of the iSCSI prototype. As depicted in Figure 4, iSCSI runs as a Linux kernel module and communicates with other iSCSI modules on other computers using TCP connections. One of the computers acts as an iSCSI initiator and the other one acts as the iSCSI target.

To test iSCSI end-to-end performance, we have added back-end storage to the server PC. It is a Seagate Cheetah 36LP FC disk, connected through a QLogic FC controller. A kernel level SCSI benchmark generates SCSI read requests on the client side and they are sent to the iSCSI target on the server. The requests are served by the FC disk ultimately. The corresponding end-to-end bandwidth for iSCSI is 180Mbps for sequential accesses and 60Mbps

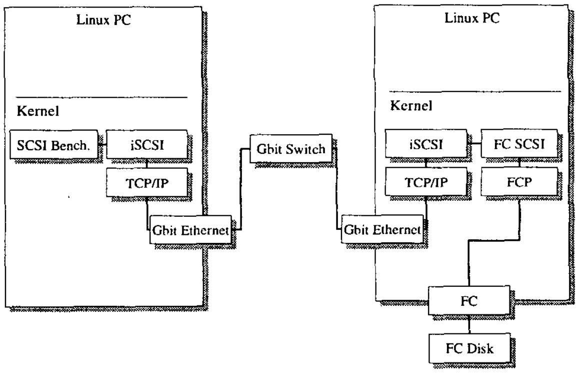

Figure 4. Experimental setup for iSCSI tests

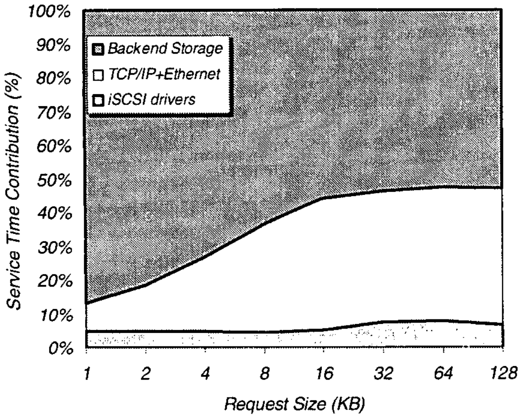

Figure 5. Service time contributions of various iSCSI path components

for random accesses, at a request size of 128KB.

Figure 5 shows the service time contributions of the backend storage, TCP network time, and iSCSI processing. One observation that can be made about Figure 5 is that the time it takes to access an iSCSI device is in the order of the time it takes to access a local device. The combined overhead of iSCSI and TCP constitutes 13 to 45 percent of the service. The service time contribution of the backend storage is higher for smaller request sizes because of the fixed mechanical positioning latencies. These are amortized better over bigger request sizes.

Figure 6 contains the buffer-to-buffer bandwidth comparison of iSCSI and raw TCP data transfers. In case of the kernel TCP, data is sent one-way from the transmitter to the receiver in units of the request sizes shown in the figure. For iSCSI, the client module sends SCSI read commands to the server module on the other computer and the server replies by data and status messages.

Depending on the request size, iSCSI adds an extra 13 percent to 60 percent overhead to the raw TCP service time. This is partly due to the extra command and status messages required for each data transfer in the SCSI protocol. The overhead of the extra iSCSI "handshake" messages is more apparent for small request sizes, because their data transfer time is shorter.

4. Fibre Channel performance

To be able to analyze the characteristics of the FC connections, we have developed a prototype FC target mode driver. The target mode driver makes the host computer running it look like a FC disk for the other host PC. Having a Linux PC emulating a target device enables us to conduct performance experiments with greater control. This is achieved by instrumenting the FC drivers on both sides.

The FC host bus adapters (FC HBAs) we used in this setup are QLogic FC adapters connected in a Fibre Channel Arbitrated Loop (FC-AL) fashion, which is the common setup for disk drives.

Figure 6 compares the bandwidth of a FC connection with the results of the bandwidth experiments with and without queued requests for TCP and iSCSI. The top line around 600 Mbps is TCP over GbE with an MTU of 9000 bytes. All other TCP and iSCSI experiments plotted in this figure use the standard MTU.

For TCP/IP/Ethernet, and consequently for iSCSI, queuing (pipelining) the requests dramatically increases the bandwidth. We have found out that the drivers and the

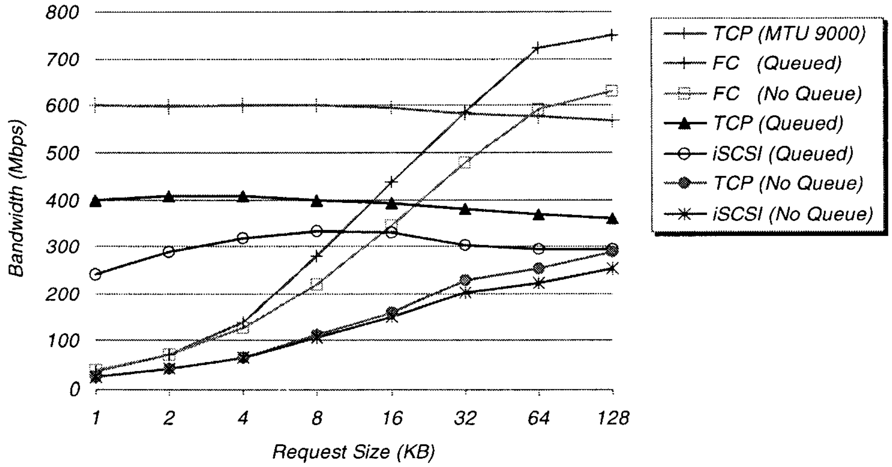

Figure 6. Fibre Channel target mode driver performance comparison with other methods (CPU 400MHz)

FC HBAs we used for FC experiments restricted the number of active requests to eight, which prevents the full utilization of the available FC bandwidth for small request sizes.

The other observation is the high cost of FC for small request sizes whether they are queued or not queued. The crossover between FC and iSCSI/TCP/Ethernet occurs between 16KB and 32KB request sizes.

The target mode driver might be used to route the SCSI requests to other storage devices residing on the same host. For example, Figure 7 shows the service time contributions of various components when the target mode driver uses a FC disk drive as the back-end storage. As the request size increases, the relative cost of the FC connection decreases together with the host and target SCSI drivers. Note that this behavior is in contrast to iSCSI service time contributions in Figure 5, and is due to fact that FC is more effective with bigger request sizes.

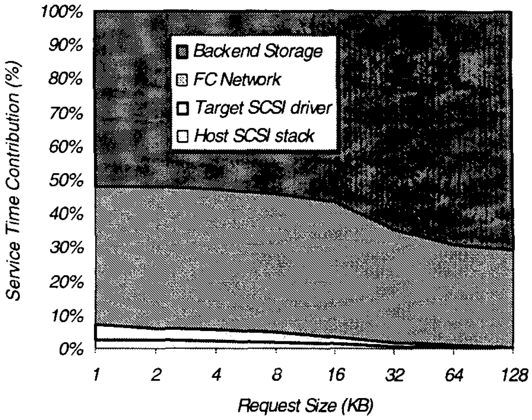

Figure 7. Relative service time contributions of various SCSI path components

5. Related work

Internet SCSI (iSCSI) [1] is an IETF project that targets the mapping of SCSI over TCP/IP. For successful transmission of high volume storage data over TCP, many limitations related to the TCP protocol processing need to be worked out.

Chase, Gallatin and Yocum [3] studied the optimizations of the TCP/IP stack to reduce the load on the host CPUs. Their optimizations—larger MTU sizes, interrupt suppression, copy avoidance by page remapping, integrated copy/checksum, and hardware checksum computation, result in near wire-speed TCP performance.

Farrell and Ong [4] studied the performance of the Gigabit Ethernet. They discovered that large MTU, socket buffer, and TCP window sizes are necessary to obtain sufficient bandwidth from GbE. Their experiments with different CPU speeds result in small bandwidth increase; and they conclude that the CPU speed is not the bottleneck. The study by Zhu, Lee, and Wang [5] concluded that the current bottleneck in GbE is between the CPU and the network interface card—the memory bus, the PCI bus, and the device drivers that move data on this path.

Rindos, Loeb, Hirasawa, Woolet, and Zaghloul [2] presented a comparison of the 100 Mbps Token Ring, 100 Mbps Ethernet, and Gigabit Ethernet. They achieved 201.2 Mbps TCP/IP throughput using Windows NT and dual 400 MHz Xeon processors. Using six clients (receivers) and six sessions per client, they have seen an aggregate throughput of 439 Mbps. Similar to our findings, they have observed the high CPU utilization of TCP/IP/GbE. They recommend increasing the maximum frame and/or protocol window sizes to tune the network bandwidth for all three network types they have experimented with.

In our study, we compared TCP/IP and FC for SCSI traffic. There are other research studies that compare these two interconnects for general network data transfer.

Ruwart [6] studied the performance of long Fibre Channel Arbitrated Loops. He points out to the cost of the several loop phases and finds out that a request size of at least 256 KB is required to fully utilize the available bandwidth of the FC-AL. In addition, he recommends increasing the number of frame buffers as the loop length is increased to compensate for the increasing delays.

Kim and Lilja [7] exploit Ethernet and Fibre Channel networks together to reduce the communication costs. They use Ethernet for short messages and Fibre Channel for long messages to achieve a balance between latency and bandwidth. In their experiments with 10 Mbps Ethernet and 256 Mbps FC, they have found Ethernet to have much better latency and bandwidth characteristics for message sizes smaller than 1500 bytes, after which FC starts to excel.

Fatoohi [8] compared several communication networks including Ethernet and FC. He found FC rate for small messages low compared to Ethernet and attributes this to the mature software base of the Ethernet.

6. Conclusions

In this paper, we have studied the performance of the SCSI traffic on TCP/IP and FC. We have presented experimental data on the performance tuning of TCP/IP traffic on Gigabit and 100Mbit Ethernet. Our test-bed consisted of Linux PCs connected together using GbE and FC adapters.

For our test platform, we observed TCP/IP bandwidths of 650Mbps and 75Mbps for Gigabit and 100Mbit Ethernet connections, respectively. For GbE, the sending side (transmitter) CPU utilization was 70%-90%. The receiver side CPU utilization is close to 100%. Special NICs that perform parts of the network protocol processing load are needed to overcome GbE's high CPU utilization problem.

The choices for the MTU (Maximum Transmission Unit) and the socket buffer size are very critical for GbE performance. Higher bandwidth rates are achieved by using big MTUs (jumbo frames) and buffer sizes of at least 256KB. In our test-bed with GbE connections, we obtained 400Mbps TCP throughput with standard MTU sizes, which, in turn, allowed us to achieve around 340Mps iSCSI throughput. Tests with and without a switch between the two computers showed that the GbE network switch does not introduce any observable latency.

Our single stream tests with the Windows NT and Linux operating systems showed that the performance of Gigabit Ethernet is at least twice faster with the Linux systems compared to the NT systems. Literature points out that Windows NT will perform well with multiple streams.

SCSI workload performance on FC strongly depends on the transmission data sizes. The startup cost of the FC-AL topology causes accesses with small data sizes to perform significantly below the physical bandwidth. With data sizes of 32 KB and more, FC outperforms GbE.

While comparing SCSI over TCP/IP and over FC, there are lots of other issues that needs to be considered in addition to the bandwidth performance in local area networks. These issues include reliability, scalability, availability, setup cost, and total cost of ownership, both in the context of local and wide area connections. These constitute a wide range of possible future work.

Acknowledgements

Special thanks to Ayman Ghanem for his help in conducting the experiments and the implementation of the iSCSI prototype.

References

[1] IPS Working Group, "iSCSI," available at http://www.ietf.org/internet-drafts, February 23, 2001.

[2] A. Rindos, M. Loeb, Y. Hirasawa, S. Woolet, and A. Zaghloul, "Performance evaluation of the latest high speed LAN adapters: 100 Mbps Token Ring; Gbps Ethernet," *Proceedings of Southeastcon '99*, pp. 98-101, 25-28 March 1999.

[3] J. Chase, A. Gallatin, and K. Yocum, "End-system optimizations for high-speed TCP," *IEEE Communications, Special Issue On High-Speed TCP*, vol. 39, pp. 68--74, June 2000.

[4] P. A. Farrell and H. Ong, "Communication performance over a Gigabit Ethernet network," *Proceedings of the IEEE International Performance, Computing, and Communications Conference, 2000. IPCCC '00*, pp. 181-189, 20-22 Feb. 2000.

[5] W. Zhu, D. Lee, and C.-L. Wang, "High performance communication subsystem for clustering standard high-volume servers using Gigabit Ethernet," *Proceedings of The Fourth International Conference/Exhibition on High Performance Computing in the Asia-Pacific Region*, vol. 1, pp. 184-189, 2000.

[6] T. M. Ruwart, "Performance characterization of large and long Fibre Channel Arbitrated Loops," *16th IEEE Symposium on Mass Storage Systems*, pp. 11-21, 15-18 March 1999.

[7] J. Kim and D. J. Lilja, "Exploiting multiple heterogeneous networks to reduce communication costs in parallel programs," *Proceedings of the Sixth Heterogeneous Computing Workshop, 1997. (HCW '97)*, pp. 83-95, 1 April 1997.

[8] R. Fatoohi, "Performance evaluation of communication networks for distributed computing," *Proceedings of the Fourth International Conference on Computer Communications and Networks*, pp. 456-459, 20-23 Sept. 1995.