Volume 2 No. 3, March 2012 ISSN 2223-4985

International Journal of Information and Communication Technology Research



http://www.esjournals.org

# iSCSI (Internet Small Computer System Interface) for your Startup

**Ibrahim Yakti, Walid A. Salameh**
Princess Sumaya University for Technology

## ABSTRACT

As number of startup companies has been increased in the last a few years, this revolution led to create a demand for affordable and reliable computing & storage solutions to satisfy their needs, including cloud computing, service oriented solutions, high rate data transfer, multiple physical locations that need to be connected together from anywhere in the world. The availability of high-bandwidth network infrastructures and of low-cost high bit-rate devices and network cards has enabled interesting new applications of distributed storage and storage virtualization techniques for Business Continuity, In this paper I will discuss the iSCSI protocol and how to adopt it in your SMEs (Small and Medium Enterprise) environments to solve the cost and complexity problems. I will present a real use-case analysis of iSCSI performance using existing infrastructure, or build a new affordable environment to implement storage over IP Network to get economical, high performance, reasonable and reliable infrastructure.

**Keywords**: *Block-Level Storage, Cloud Computing, Ethernet, Fibre-Channel, iSCSI, Network Storage, RAID.*

## 1. INTRODUCTION

As the demands grow for multiple physical locations around the world and reach the network quickly without going over-budgets, IETF RFC 3720 [1] introduces iSCSI protocol which provides robust and high-available system operation across multiple geographical enterprise networks.

This paper focuses on iSCSI and why you should set it up in your SMEs (Small and medium enterprises) environment. In Section 2 it will cover a brief about network storage and discusses the current solutions including Fibre-Channel and why it is not suitable for

SMEs from budget and management perspectives, then it will discuss IP networks. Section 3 Will discuss the available network storage protocols and compare iSCSI with Network File System (NFS) and explain the difference between file-level sharing with block-level.

Section 4, we will dig deeper and talk about iSCSI protocol and go through its specification that SME needs to adopt it in its environment.

In section 5 we will describe the benchmark environment, infrastructure setup, and measurement methodologies using native hardware and software without any special modification, then I will discuss the results. In section 6 we will list our findings and give the required summary that is needed to start adopting iSCSI, finally, we will discuss our contributions and possible future work that can be done.

## 2. INTRODUCTION TO NETWORK STORAGE

For a long time it was doubtless that any storage regardless it is a direct attached or a network storage is located in a near distance from its consumer, i.e.: in server room. The task to make this storage accessible 24x7 from any place in the world securely was very complicated, especially when it started to go out of the organization's boundaries.

In any company, the use of storage solutions will depend on a number of different factors:

- Available budget
- Data security requirements
- Network infrastructure
- Data availability requirements.

Many solutions are available like SAN (Storage Area Network) [2], most of the SANs uses Fibre-Channel (FC) as the standard connection in enterprise, FC has distance limitation of 10,000 meters. Other issues that faced the FC were cost of special hardware and trained resources, other protocols were invented like IFCP (Internet FC

International Journal of Information and Communication Technology Research



http://www.esjournals.org

Protocol) [3] and FcoIP (FC over IP) [4] to allow FC run over IP, but these solutions need special hardware to transport the FC over lower bandwidth networks like WAN, in addition to extra complexity and vulnerability to IP networks.

## 2.1 Fibre-Channel Problems

Obviously the most problem associated with FC is the cost, The need for special hardware with special host bus adapters (HBAs) for each FC host or FC switch, in practical this means to have the normal Local Area Network (LAN) and another separated FC network, this leads to extra cost for FC components and management of FC networks, the management of FC networks needs additional special trained network administrators other than administrators of the current organization LAN.

Alongside traditional FC networks, a growing proportion of SAN adopters find iSCSI connectivity compelling, particularly because it uses Ethernet, which is a standardized, widely deployed, and well-understood technology. Using iSCSI to transmit block-level storage commands over Ethernet networks adds benefits in terms of cost and simplicity.[5] Ethernet technology is more mature than FC, there are more resources to manage ethernet than there are for FC, so eventually the management of ethernet based will be much cheaper & reliable than FC.

## 2.2 IP Networks

The use of IP networks is very common in most entities over the globe for many reasons including:

- Most modern Operating Systems (OS) has a native support for IP.
- IP networks are very mature and has good performance throughput.
- Has lower limitation comparing to FC, distance is set by application timeout not by network.

Because of the aforementioned facts there are many reasons why organizations would prefer the iSCSI to be run over IP networks:

- IP networks offer decent remote backup and error recovery over distributed locations.
- Using the same IP-based technology in both ends (clients and servers) make the management easier and less expensive since there is only one type of network to manage.[6]
- All popular OS support internet protocol, thus we do not need to do any modification on the OS when we implement iSCSI.
- IP networks offer required levels of security, availability, management, and quality of service (QoS).
- iSCSI can be run over LAN or WAN networks.
- Simplicity and affordable low costs.

## 3. iSCSI vs. NFS (Network File System)

NFS [7] and iSCSI provide fundamentally different data sharing semantics. NFS is inherently suitable for data sharing, since it enable files to be shared among multiple client machines. In contrast, a block-level protocol such as iSCSI supports a single client for each volume on the block server. Consequently, iSCSI permits applications running on a single client machine to share remote data as in virtualized systems and data centers, but it is not directly suitable for sharing data across machines. It is possible, however, to employ iSCSI in shared multi-client environments by designing an appropriate distributed file system that runs on multiple clients and accesses data from block server [8].

Sometimes iSCSI and NFS appear to offer the same services. Both allow remote storage devices to be used as if they were local disks and both iSCSI and NFS work over TCP/IP networks. With NFS, the file system resides at the file server. i.e: The client issues read and write commands which are sent over the network. These commands are interpreted by the file system on the server and the appropriate actions are taken. iSCSI is different in that the files system resides on the client machine. The commands are interpreted and then translated to block commands which are forwarded to the network. In essence, the commands are already in the correct form once they are received by the target and therefore there is no need to translate the file I/O to block I/O.

“However, iSCSI takes advantage of aggressive caching techniques and when the cache is filled, the iSCSI performance substantially exceeds that of NFS.

A great deal of work has been done in order to identify the bottlenecks existing in an iSCSI-based SAN. While some believe the performance bottleneck lies on the network, others believe that it is a storage (disk) issue.” [9]

SCSI blocks work at lower levels (block-levels) by offering entire data stores Logical Units Numbers (LUNs) to iSCSI clients, while NFS is used at the file level.

International Journal of Information and Communication Technology Research



http://www.esjournals.org

## 4. iSCSI

The iSCSI protocol consists two main components: The target (server) processes the requests which comes from initiator and response back. The initiator (client) sends SCSI commands to target. They all perform communications using standard TCP/IP protocol. Thus, it provides lower overall cost and higher scalability than current Storage Area Networks (SAN) that typically rely on the FC technology.

I have already discussed the advantages of using IP networks in section 2 like easy management tasks due to the convergence of network technologies, since there is no need to run networks based on different technologies.

It is clear now that iSCSI uses the best cutting-edge technologies, it uses SCSI commands (SCSI Protocol) and IP netwoks (TCP/IP Protocol), iSCSI is simply SCSI over IP. iSCSI stands for IP SCSI which enables initiators (clients) to perform block data input/output (I/O) operations over IP.

You can consider iSCSI protocol to build cheap, affordable SAN environment over your existing ethernet network since the main goal of iSCSI is to extend the existing functionality without complicating the process or requiring new resources, training, or hardware.

The iSCSI protocol maps the SCSI client-server protocol into a TCP/IP interconnect. Initiators (clients) on a SCSI interface issue commands to a SCSI target (server) in order to request the transfer of data to/from I/O devices. The iSCSI protocol encapsulates these SCSI commands and the corresponding data into iSCSI Protocol Data Units (PDUs) and transmits them over a TCP connection.

## 4.1 iSCSI Architecture

Figure 1 shows the general iSCSI block diagram, an initiator and a target, which communicate to each other via the iSCSI protocol.

| **Initiator** | | **Target** |
|---|---|---|
| Application | …..................... | Logical Unit |
| SCSI | …..................... | SCSI |
| iSCSI Protocol | …..................... | iSCSI Protocol |
| TCP/IP | …..................... | TCP/IP |
| Ethernet | …..................... | Ethernet |

**Figure 1: iSCSI Block Diagram**

Figure 2 shows the initiator, the application, which needs to store and access data to/from the storage device, issues file requests. The file system converts file requests to block requests from application to block device layer and SCSI layer. The reverse order occurs on the target side, more explanation is available in the next section.

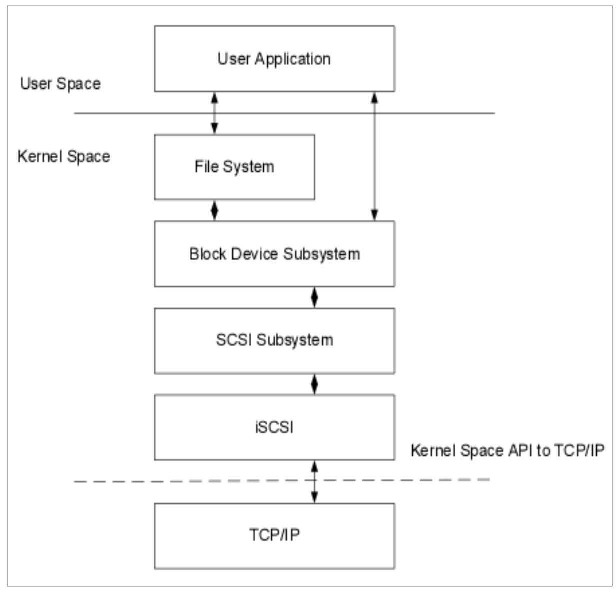


**Figure 2: Typical Initiator Protocol Stack [6]**

## 4.2 iSCSI Encapsulation

| | | | | | |
|---|---|---|---|---|---|
| Ethernet Header | IP Header | TCP Header | iSCSI Header | iSCSI Data | Ethernet Trailer |

**Figure 3: iSCSI Encapsulation**

Basically, iSCSI works by encapsulating SCSI commands and transporting them via TCP/IP. On opposing ends of the network are the pillars of iSCSI: the initiator and the target. The initiator (can be in the form of hardware or software) is installed on the host. The most basic responsibilities of the initiator are to establish a connection to an iSCSI target and start the transfer of information to/from it. The iSCSI target's primary function is to respond back to the requests started by the initiator.

## Terms and Definitions

- iQN: Initiator Node Name (Identity value for iSCSI clients, similar to MAC addresses).

Volume 2 No. 3, March 2012 ISSN 2223-4985

International Journal of Information and Communication Technology Research



http://www.esjournals.org

- Domain Sets: Logical segmentation of iSCSI entities (Targets and Initiators into separate groups).
- LUNs: Logical Unit Numbers (A logical array of storage units. One storage entity can be divided into multiple LUNs).
- Jumbo Frames: jumbo frames are Ethernet frames with more than 1500 bytes of payload. Conventionally, jumbo frames can carry up to 9000 bytes of payload, many Gigabit Ethernet switches and Gigabit Ethernet network interface cards support jumbo frames [10].

## 4.3 iSCSI Session

An iSCSI session has two phases. It starts with a 'login' phase during which the initiator and target negotiate the parameters for the rest of the session. Then, a 'full feature' phase is used for sending SCSI commands and data. Based on the parameters negotiated during the login phase, an iSCSI session can use multiple TCP connections multiplexed over one or more physical interfaces, enable data integrity checks over PDUs, and even incorporate different levels of error recovery. iSCSI sessions are typically long-lived. The login phase represents only a small part of the overall protocol processing load.

There are two types of sessions defined in iSCSI, a normal operational session and a discovery session used by the initiator to discover available targets.

A session is identified by a session ID (SSID), which is made up of an initiator (ISID) and target (TSID) components.

TCP connections may be added and removed within a session; however, all connections are between the same unique initiator and target iSCSI nodes. Each connection within a session has a unique connection ID (CID).

## 4.4 iSCSI Error Handling

"A fundamental portion of error recovery is maintaining enough state and data to recover an errant process. This is the case with iSCSI in that the initiator is expected to retain the necessary command and data information to be able to rebuild any outstanding PDU. Likewise, the target is expected to maintain any unacknowledged data-out along with status response information."[11]

Two mechanisms used by iSCSI for error handling are retry and reassignment. An initiator may attempt to "plug" any missing commands by resending the same command or data PDU to the target. The reassignment is used when the TCP connection between the initiator and the target is lost. In this case, the initiator sends a "Task Reassign" task management PDU via a new connection, instructing the target to continue an outstanding command on the new CID. It is not required for targets to support this feature, which is negotiated at login time. [11]

The iSCSI protocol supports three levels of error recovery:

1. Level 0: Any error results in the session being immediately dropped. The session must be started over by the application.
2. Level 1: Rather than taking down the entire session, this level of error recovery initiates a simple retransmission of the corrupted PDU in question. This process is effectively transparent to the SCSI layer.
3. Level 2: For the most robust error recovery, this level initiates full connection recovery. While an entire I/O transaction must occur over the same connection, a session may have multiple connections per session (MCS). If a link fails, the I/O can be moved over to surviving connections in a mildly transparent manner.

Level 1 and Level 2 error recovery are more appropriate for mission-critical applications where a dropped session is highly undesirable.

Since the iSCSI is still an emerging protocol, most targets support only Level 0 error which doesn't recover from errors; it is just detect errors and prevent data corruption. Practically users would certainly rather deal with unexpected session loss than have their data corrupted. [12]

## 4.5 Logout and Shutdown

The logout process provides for a graceful shutdown mechanism to close an iSCSI connection or session. The initiator is responsible for starting the logout procedure; however, the target may prompt this by sending an asynchronous iSCSI message indicating an internal error condition. In either case the initiator sends a logout request, after which no further request may be sent. The logout response from the target indicates that cleanup is complete and no further responses will be sent on this connection. Additionally, the logout response contains recovery information from the target. This includes the length of

International Journal of Information and Communication Technology Research



http://www.esjournals.org

time the target will hold, pending command information for recovery purposes (Time2Retain) and the length of time the initiator should wait before attempting to reestablish the connection (Time2Wait). Finally, connections are shut down by sending TCP FINs. [11]

## 5. BENCHMARKS

To evaluate iSCSI performance, the iSCSI target is a RAID disk over a Gigabit Ethernet LAN, The RAID 5 and RAID 6 were used to guarantee the high availability and data integrity features for the organization so we do not run into risk of data loss nor data corruption, thus we get the best solutions for performance and reliability. for best RAID setup please consult your system administrator as per your needs and budget.

As mentioned in section 4.3, The login phase represents only a small part of the overall protocol process. Because of this reason benchmarks will ignore the login phase and just do tests during the “full feature” phase of the protocol.

While some performance studies of the iSCSI protcol have been conducted in the past, none has evaluated iSCSI target systems. Peter Radkov et al. [13] have compared the performances of the iSCSI protocol and NFS (Network File System) [14]. Wee Teck Ng et al. [15] investigated the performance of the iSCSI protocol over wide area networks with high latency and congestion. Sarkar et al. [16] compared an initiator implementation using a Gigabit Ethernet adapter to implementations using specialized network adapters, known as TOE and HBA adapters. Ashish Palekar et al. [17] described the design of UNH iSCSI target software, I will be evaluating initiator on different OS (Windows 7 64bit, and OpenSUSE Linux 12.1 64bit), and the Synology DS411J iSCSI target over gigabit network.

### 5.1 Test-bed Environment

#### 5.1.1 Hardware Specification

- Asus Motherboard P8Z68-V LE / Intel® Z68 chipset.
- Intel® Core™ i7-2600K Processor (8M Cache, 3.40 GHz).
- Kingston HyperX blu memory 16GB (4x4GB) DDR3 1600MHz CL9 DIMM.
- Integrated Realtek® 8111E , 1 x Gigabit LAN Controller.
- Cat5e cables.
- Cat6e cables.
- Zyxel NBG460N Wireless N Gigabit Router.
- Synology DS411j Budget-friendly 4-bay NAS Server.
- 4 x HITACHI Deskstar 2TB 7200 RPM 64MB Cache SATA 6.0Gb/s 3.5"
- 4 disks Raid6 (4x2TB).
- 4 disks Raid5 (4x2TB).

#### 5.1.2 Software Environment

- OpenSUSE 12.1 Linux (kernel-3.1.9-2.1.x86_64).
  - hdparm: Utility to change hard drive performance parameters in Linux.
  - dd is a common Unix program whose primary purpose is the low-level copying and conversion of raw data.
  - iSCSI initiator.
- Microsoft Windows 7 64bit Edition
  - h2benchw: a benchmark tool that measures the interface transfer rate, sustained transfer rates and the average access time, the benchmark tool requires a blank hard drive without any physical partitions.

### 5.2 Benchmarks and results

#### 5.2.1 Sequential Access Versus Random Access

One of the significant performance factors is the effect of how the data is laid down on the hard disk. If writes are sequential, which minimize seek times on the hard drive, the data can be written much faster. If access is random, resulting in many seeks across the hard drive, these times can go down dramatically. Using an appropriate RAID configuration can reduce the effect of the seeks on random access. [18]

International Journal of Information and Communication Technology Research



http://www.esjournals.org

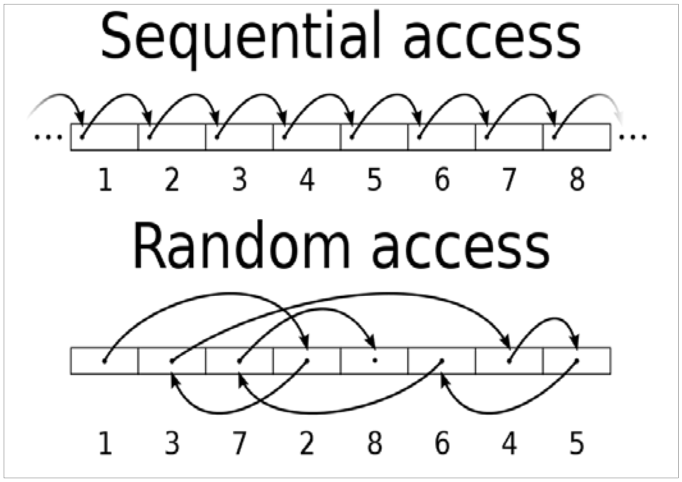


**Figure 4: Sequential Access vs. Random Access [19]**

Figure 4 shows the differences between the the sequential access and the random access, obviously the sequential access is more fast for both reading and writing operations.

### 5.2.2 Tests and outcomes

To guarantee the highest performance and stability it is recommended to implement a dedicated IP infrastructure for the iSCSI target. The easiest way to achieve this is via VLAN (Virtual LAN) to isolate the iSCSI traffic on an existing infrastructure.

To separate network performance from iSCSI performance, I have isolated the LAN. The test environment was completely standalone. Only TCP packets associated with iSCSI were placed on the wire. No other network traffic was enabled. Using this approach, the speed of the LAN and iSCSI components could be measured.

Local Disk Performance: After isolating network and guaranteeing that I am using 100% of its performance, I have benchmarked the performance of the disks by attaching them directly to machine using SATA interface, I have found that the average sequential read is 208MB/s throughout and 165.8MB/s for sequential writing, this test help me to determine if the disks are part of any bottlenecks when I benchmark the iSCSI Raid setup.

Figure 5 shows the benchmark results using the h2benchw benchmark tool on Windows 7.

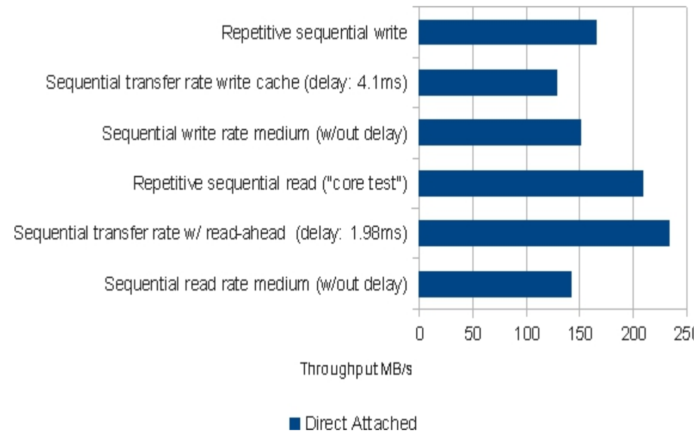


**Figure 5: Direct attached disk benchmark**

After that, I have setup a raid5 volume group using the synolgy admin interface, Figure 6 shows the benchmark results for Raid5 disks, and with the disk cache disabled using Windows 7 initiator and Cat5e cabling.

I have found that the average sequential read is 40.9MB/s throughout and 35MB/s for sequential writing.

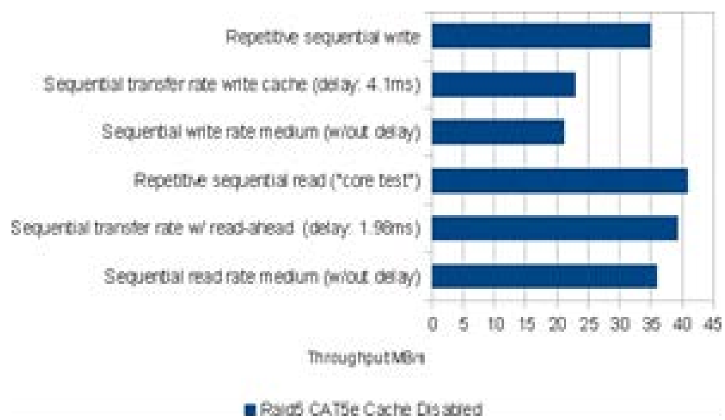


**Figure 6: iSCSI Raid5 Cat5e / Cache Disabled**

Figure 7 shows the benchmark results for Raid5 disks, and with the disk cache enabled using Windows 7 initiator and Cat5e cabling.

I have found that the average sequential read is 44.89MB/s throughout and 32.6MB/s for sequential writing.

International Journal of Information and Communication Technology Research



http://www.esjournals.org

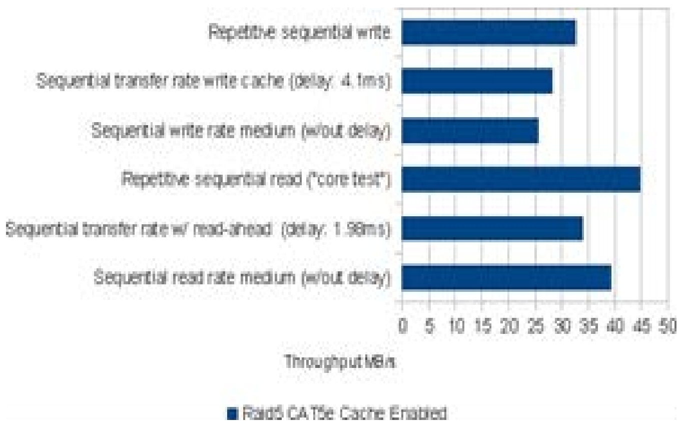


**Figure 7: iSCSI Raid5 Cat5e / Cache Enabled**

Figure 8 shows the benchmark results for Raid6 disks, and with the disk cache disabled using Windows 7 initiator and Cat5e cabling.

I have found that the average sequential read is 43MB/s throughout and 30.9MB/s for sequential writing.

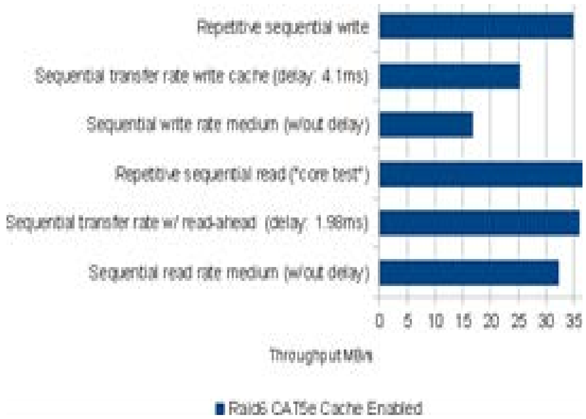


**Figure 8: iSCSI Raid6 Cat5e / Cache Disabled**

Figure 9 shows the benchmark results for Raid6 disks, and with the disk cache enabled using Windows 7 initiator and Cat5e cabling.

We have found that the average sequential read is 41.5MB/s throughout and 34.9MB/s for sequential writing.

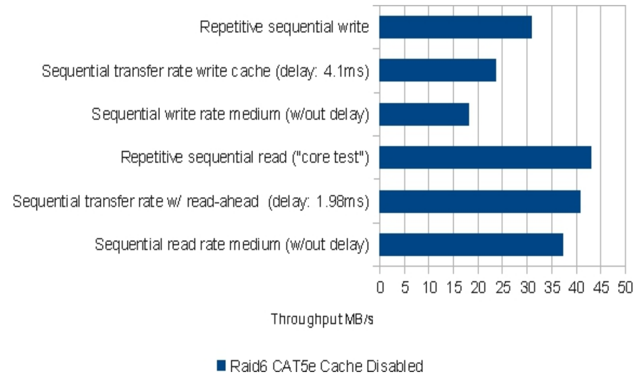


**Figure 9: iSCSI Raid6 Cat5e / Cache Enabled**

Figure 10 shows the benchmark results for Raid6 disks, and with the disk cache disabled using OpenSUSE 12.1 Linux initiator and Cat5e cabling.

We have found that the average read rate is 43.4MB/s throughout and 21.4MB/s for average writing.

Changing the initiator from Windows to Linux OS, i.e: by changing the initiator driver and implementation, as you can notice here the read speed has been increased while the writing speed has been decreased.

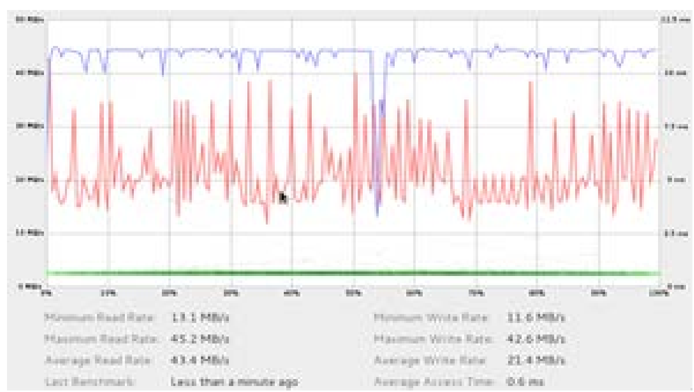


**Figure 10: iSCSI Raid6 Cat5e / Cache Disable / Linux**

Figure 11 shows the benchmark results for Raid6 disks, and with the disk cache enabled using OpenSUSE 12.1 Linux initiator and Cat5e cabling.

We have found that the average read rate is 46.1MB/s throughout and 22.1MB/s for average writing.

International Journal of Information and Communication Technology Research



http://www.esjournals.org

Changing the initiator from Windows to Linux OS, i.e: by changing the initiator driver and the implementation, as you can notice here the read speed has been increased while the writing speed has been decreased.


**Figure 11: iSCSI Raid6 Cat5e / Cache Enabled/ Linux**

In figure 12, we have enabled the Jumbo Frames on both initiator and target, the max value I was able to set on the initiator ethernet (Integrated Realtek® 8111E Gigabit LAN Controller) was mtu=7000 on the Linux machine, so I enabled the jumbo frames on the target and set it to 7000 (the max value can be set to 9000), however, we doubled checked that my router firmware supports the jumbo frames without any problem. I have found that the average read rate is 43.6MB/s throughout and 26.6MB/s for average writing.

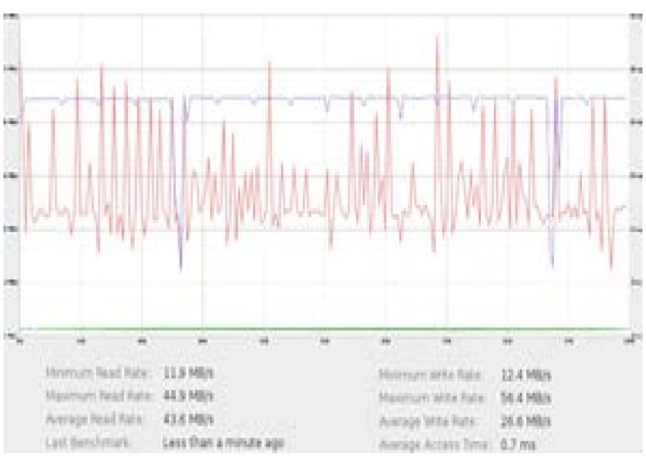


**Figure 12: iSCSI Raid6 Cat6e / Jumbo Frames/ Linux**

Apparently the linux initiator has better management for writing than the windows initiator.

## 6. SUMMARY, CONTRIBUTION AND FUTURE WORK

iSCSI is suitable for people who are looking for reliable, cheap, easy, and more throughput using their existing network infrastructure. iSCSI lets you create a low-cost Storage Area Network (SAN) using existing high-speed Ethernet hardware. This saves a lot costs, efforts, resources, and time comparing to creating an FC-SAN, if you stuck in iSCSI bottlenecks then check your network first, The simplicity and cost effectiveness of iSCSI brings affordable, simple storage networking solutions to all customer segments. Enterprises are increasingly demanding fast and reliable access to their data on a 24 X 7 basis.

Most of the used iSCSI initiators are the native OS initiators because using them are very easy, simple, and effective.

Networked block storage technologies are likely to play a major role in the development of next generation data centers. iSCSI allows you to create high performance applications distributed in different locations but virtually apear as one logical network. Data storage plays an essential role in today's fast-growing data-intensive network services. New standards and products emerge very rapidly for networked data storages.

Using iSCSI over fibre channel does not mean replacing the fibre-channel but providing a functionality to existing network infrastructure.

The performance (throughput) results were promising when comparing to common file sharing using the same network. iSCSI is optimal for applications that need to access the whole file as the block-level protocols are optimal for the applications that need the fastest access to chunk of data on a disk.

Network setup and basic components are the main factors for iSCSI performance, having a good cabling between the nodes as we tested Cat5e and Cat6e cables in our tests, having a good network interface card is important to get better results since some feature needs special hardware or drivers, for example I was not able to use jumbo frames greater than 7000. flow and congestion control is very crucial as well to get better performance.

Moreover, storage setup is important too, using appropriate RAID controller and configuration will enable you to access larger data blocks and it will allow you to

International Journal of Information and Communication Technology Research



http://www.esjournals.org

access your data sequentially, and eventually improve the performance.

"Many networks today rely on QoS alone to resolve congestion issues and prioritize different traffic types on a per port basis. Extreme Networks can augment the benefits of QoS with its CLEAR-Flow engine to process iSCSI traffic identified on the network. CLEAR-Flow is a feature that allows Extreme Networks switches to make forwarding decisions based on traffic type. Instead of simply looking at the source and destination of the traffic and forwarding it along the appropriate Layer 2 or Layer 3 path, CLEAR-Flow takes things a step further by allowing network administrators to specify certain types of traffic that require more attention." [20]

Last and most important point is knowing your application specifications and requirements will help you to have better setup, i.e: jumbo frames may be very bad for video streaming in some cases while it is excellent for large file transfer and they are good for applications that really need bigger data units.

## 6.1 Future work

iSCSI uses CRC for error detection, which needs a lot of calculations and time-consuming, I think the performance of iSCSI can be improved by working on error detection and correction, future work may consist of more benchmarks with error detection disabled to find out the impact of these calculations on the overall performance of the iSCSI.

## ACKNOWLEDGMENT

I would highly thank my friend "Rasem Brsiq" for inspiring us about iSCSI protocol at first place. However, all benchmarks were not able to be done without using khodarji.com equipments, environment, and hardware.

## REFERENCES

[1] Internet Small Computer Systems Interface (iSCSI). http://www.ietf.org/rfc/rfc3720.txt

[2] IP and ARP over Fibre Channel. http://tools.ietf.org/html/rfc2625

[3] iFCP - A Protocol for Internet Fibre Channel Storage Networking. http://tools.ietf.org/html/rfc4172

[4] Fibre Channel Over TCP/IP (FCIP). http://tools.ietf.org/html/rfc3821

[5] Intel Simple, Reliable Performance for iSCSI Connectivity. http://download.intel.com/support/network/sb/inteliscsiwp.pdf

[6] A Scalable and High Performance Software iSCSI Implementation. ftp://download.intel.com/technology/comms/perfnet/download/iscsi_fast.pdf

[7] NFS Version 3 Protocol Specification. http://www.faqs.org/rfcs/rfc1813.html

[8] Differences Between NFS and iSCSI http://www.usenix.org/events/fast04/tech/full_papers/radkov/radkov_html/node5.html

[9] iSCSI-based Storage Area Networks for Disaster Recovery Operations, By MATTHEW R. MURPHY, THE FLORIDA STATE UNIVERSITY -COLLEGE OF ENGINEERING http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.127.8245&rep=rep1&type=pdf

[10] Gigabit Ethernet Jumbo Frames, Phil Dykstra, WareOnEarth Communications, Inc. http://sd.wareonearth.com/~phil/jumbo.html

[11] http://cs.haifa.ac.il/courses/network_mem/imdpm_wp.pdf

[12] http://www.jdsu.com/ProductLiterature/Understanding-iSCSI-Digests-white-paper-30162803.pdf

[13] P. Radkov, L. Yin, , P. Goyal, and P. Sarkar, "A Performance Comparison of NFS and iSCSI for IP-Networked Storage," in the USENIX Conference on File and Storage Technologies, San Francisco, CA, March 2004, pp. 101–114.

[14] R. Sandberg, D. Coldberg, S. Kleiman, D. Walsh, and B. Lyon, "Design and Implementation of the Sun Network Filesystem," in the USENIX Summer Conference, Portland, OR, June 1985, pp. 119–130.

[15] W. T. Ng, B. Hillyer, E. Shriver, E. Gabber, and B. Ozden, "Obtaining High Performance for Storage

Outsourcing," in the USENIX Conference on File and Storage Technologies, Monterey, CA, January 2002, pp. 145–158.

[16] P. Sarkar, S. Uttamchandani, and K. Voruganti, "Storage over IP: When Does Hardware Support Help?" in the Conference on File and Storage Technologies (FAST 03). San Francisco, CA: USENIX, January 2003, pp. 231–244.

[17] A. Palekar, N. Ganapathy, A. Chadda, and R. D. Russell, "Design and implementation of a Linux SCSI target for storage area networks," in 5th Annual Linux Showcase & Conference. Atlanta, GA: USENIX, November 2001.

[18] http://www.novell.com/documentation/oes2/other_docs/iscsi_performance_case_study.pdf

[19] http://en.wikipedia.org/wiki/File:Random_vs_sequential_access.svg

[20] www.extremenetworks.com/doc.aspx?id=1140 High-Performance iSCSI SAN Architecture A Solution's View from Extreme Networks, Intel and NetApp