2008

Study of TCP Issues over Wireless and Implementation of iSCSI over Wireless for Storage Area Networks

Rahul Sharma
San Jose State University

Study of TCP Issues over Wireless and Implementation of iSCSI over Wireless for Storage Area Networks

Presented to

The Faculty of the Department of Computer Science

San Jose State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

Rahul Sharma

Fall 2008

ABSTRACT

The Transmission Control Protocol (TCP) has proved to be proficient in classical wired networks, presenting an ability to acclimatize to modern, high-speed networks and present new scenarios for which it was not formerly designed. Wireless access to the Internet requires that information reliability be reserved while data is transmitted over the radio channel. Automatic repeat request (ARQ) schemes and TCP techniques are often used for error-control at the link layer and at the transport layer, respectively. TCP/IP is becoming a communication standard [1]. Initially it was designed to present reliable transmission over IP protocol operating principally in wired networks. Wireless networks are becoming more ubiquitous and we have witnessed an exceptional growth in heterogeneous networks. This report considers the problem of supporting TCP, the Internet data transport protocol, over a lossy wireless link whose features vary over time. Experimental results from a wireless test bed in a research laboratory are reported.

ACKNOWLEDGEMENTS

I would like to express the deepest appreciation to my advisor, Dr. Robert Chun, who has the attitude and the substance of a genius and whose supervision, support, and keenness is invaluable. Dr. Chun has repeatedly and persuasively conveyed a spirit of quest in regard to research and learning, and an enthusiasm in regard to teaching. Without his guidance and constant aid, this project would not have been possible.

I very much thank and appreciate the members of my committee Dr. Teng Moh and Gaurav Khanna for their time and effort put as thesis committee members. My committee has provided instructive insight, guidance in polishing the work presented in this report.

I would like to thank my Manager Gaurav Khanna whose enthusiasm for the "underlying structures" had lasting effect. He has been an asset and has mentored me for all the technical support and help in the Cisco Lab. Special thanks to Arkadiy Shapiro whose valuable inputs into the research helped me complete the project.

It has been an exigent, yet worthwhile expedition which I could not have concluded alone and am appreciative for your support.

Thank you

TABLE OF CONTENTS

LIST OF DIAGRAMS……..vii

LIST OF TABLES AND GRAPHS……viii

1. INTRODUCTION.....1

2. PROBLEM STATEMENT.....5

3. LITERATURE REVIEW…….....8

3.1 - Feedback Channel Losses for Wireless TCP.....8

3.2 - The Snoop protocol..…….....9

3.3 - TULIP.....10

3.4 - The split connection approach.....12

3.5 - Link-layer recovery…….....13

3.6 - Explicit Notification……15

3.6.1 - ICMP Messaging…….....16

3.6.2 - Explicit Loss Notification……...16

3.6.3 - Syndrome……....17

3.6.4 - Partial Acknowledgements……..…17

3.7 - Categorizing the Packet Loss.....18

3.8 - Other Proposed Schemes……..…...21

4. PROPOSED SOLUTION……..…23

4.1 - iSCSI Protocol…….....23

4.1.1 - iSCSI PDUs……27

4.1.2 - iSCSI Parameters…….....…31

4.1.3 - Data Integrity and Security…….……34

4.1.4 - Recovery by iSCSI…….36

4.2 - Strategy to Improve iSCSI Performance…….40

4.3 - Enhancing iSCSI Performance in Wireless Application……..….41
4.3.1 - Adaptive Control of iSCSI Protocol……..……..42

5. EXPERIMENT SETUP AND METHODOLOGY……...…43
5.1 Software and Hardware Tools, Development Kits Used……...43
5.2 Test Setup ……..46
5.2.1 Experiment 1: Wired Network ……49
5.2.2 Experiment 2: Wireless Network……..55
5.3 Results…….…...61
5.3.1 Wired test results……...…62
5.3.2 Wireless Test Results…….....63
6. CONCLUSION…….……74

7. ACRONYM.....77

8. REFERENCES.....79

LIST OF DIAGRAMS

Figure 1. Architecture of MCP...……..…….13

Figure 2. Link Layer recovery model……14

Figure 3. SAN vs.
iSCSI……24

Figure 4. File level And Block Level File storage…….25

Figure 5. iSCSI Model……...26

Figure 6. SCSI Command Sequence…….…….27

Figure 7. iSCSI protocol stack……...30

Figure 8. iSCSI Write/Read Operation……..31

Figure 9. Static initiator setup for iSCSI hosts..……....37

Figure 10. Proxy initiator setup for iSCSI hosts…….…...38

Figure 11. Sequence of phases from the establishment of a TCP session between host and MDS to the Full-Feature Phase…….……..39

Figure 12. iCache Architecture……..41

Figure 13. iSCSI over Wired Network……..……49

Figure 14. iSCSI Properties Applet……...50

Figure 15. iSCSI Initiator - Discovery Tab……50

Figure 16. iSCSI Initiator Properties – Persistent Target……..…51

Figure 17. Disk Fragment of JBOD……..53

Figure 18. Iometer showing Disk Targets…….53

Figure 19. Network target selection …….54

Figure 20. Access Specification selection…….54

Figure 21. Access Specification details…….55

Figure 22. iSCSI over Wireless test……...…56

Figure 23. The host connected to Access point SSID Rahul…….57

Figure 24. iSCSI with target discovery for wireless test……...58

Figure 25. JBOD storage disk selected……..59

Figure 26. Disk selection for the test…….…60

Figure 27. Target selection for the test……..…61

Figure 28. Iometer result for 512 data bytes …….……....62

Figure 29. Iometer result for 32k data bytes …….……....62

Figure 30. Iometer result for 512b data I/Os with access point 5ft away…….….64

Figure 31. Iometer results for 32k data I/Os with access point 5ft away……..64

Figure 32. Iometer results for 32k data I/Os with access point 20ft away…..…..65

LIST OF TABLES AND GRAPHS

Table 1. IP Addressing……..….49

Table 2. IP addresses used for the Wireless test setup……...…56

Table 3. Test results for 512b and 4k data bytes I/Os over wired setup……....63

Table 4. Test result with access point 5ft away from the host……..…….65

Table 5. Test result with access point 20ft away from the host……..……...65

Graph 1. Performance comparison for Number of I/Os per second for iSCSI over wired network and wireless with 5ft and 20ft host-access point distances…...…67

Graph 2. Performance comparison for Throughput in Mbps for iSCSI over wired network and wireless with 5ft and 20ft host-access point distances…….69

Graph 3. Performance comparison for Average I/O Response Time (ms) for iSCSI over wired network and wireless with 5ft and 20ft host-access point distance.…70

Graph 4. Performance comparison for Maximum I/O Response Time (ms) for iSCSI over wired network and wireless with 5ft and 20ft host-access point distances…….72

1. INTRODUCTION

Wireless links are not as strong as wired links. The radio quality may differ significantly over time, thus, the bandwidth is generally lower, and transmission errors occur more often [2]. More errors are generated while sending signals over a unidirectional radio based medium than in a guided medium such as coax or fiber. Signal strength is weakened with the distance between the portable station and the base station, and radio signals bounce off objects, leading to interference and multi-path effects. To screen upper protocol layers from transmission errors, interleaving, error correction and retransmissions can be used at lower layers. In several wireless networks, the Data link layer performs error recovery according to an Automatic repeat request (ARQ) protocol [3]. ARQ protocols can be categorized according to the intensity of reliability provided to the upper layers. An ARQ protocol is perfectly reliable, if it retransmits frames until they are acknowledged or, after a large number of retransmission attempts, disconnects the link and notifies upper layers. If there is a limit to the maximum number of retransmissions, then the ARQ protocol is defined as highly persistent or highly reliable. On the other hand, a low persistent or partially reliable ARQ protocol retransmits a frame 2-5 times before giving up and transmits the next frame as an alternative.

The Transmission Control Protocol (TCP) has proved proficient in classical wired networks, presenting an ability to acclimatize to modern, high-speed networks and present new scenarios for which it was not formerly designed [1]. Wireless access to the Internet requires that information reliability be reserved while data is transmitted over the radio channel. Automatic repeat request (ARQ) schemes and TCP techniques are often used for error-control at the link layer and at the transport layer, respectively. TCP/IP is

becoming a communication standard. Initially it was designed to present reliable transmission over IP protocol operating principally in wired networks. Wireless networks are becoming more ubiquitous and we have witnessed an exceptional growth in heterogeneous networks. This report considers the problem of supporting TCP, the Internet data transport protocol, over a lossy wireless link whose features vary over time. To prevent throughput deprivation, it is crucial to screen the losses and the time variations of the wireless link from Transmission Control Protocol. Numerous solutions to this problem have been proposed in earlier studies, but their performance was studied on a solely experimental basis. The new communication paradigm is a wireless network communicating with a fixed wired host at the end-point. Mixing wired and wireless networks under the same TCP/IP ceiling presents plentiful challenges. TCP is a link oriented transport protocol providing a reliable byte stream to the application layer. Application data provided to TCP is divided into Protocol data units (PDUs) called segments, before transmission [3]. Since TCP uses an ARQ mechanism based on positive acknowledgments, reliability is achieved. Every byte is numbered and the number of the first byte in a segment is used as a sequence number in the TCP header. A cumulative acknowledgment is transmitted by a receiver in reply to an incoming segment, which implies that many segments can be acknowledged at the same time. TCP manages a retransmission timer which starts when a segment is transmitted. If the timer expires before the acknowledgement of the segment, TCP retransmits the segment. The retransmission timeout value (RTO) is calculated dynamically based on measurements of the round trip time (RTT), which is the time taken from the transmission of a segment until the receipt of acknowledgment.

Intermittent connectivity and High bit error rates characterize wireless links. This can result in considerable deprivation in the performance of TCP over wireless networks because non-congestion related packet losses can be misunderstood by TCP as indications of network congestion which results in unnecessary congestion control. TCP interprets the entire data loss as congestion in the network and thus, TCP decreases its transmission rate to reduce the congestion. TCP was primarily designed for wired networks which have a very different character from wireless. TCP is the established reliable transport protocol in today's Internet and has been broadly listed in many application layer protocols. Thus it must be supported in the wireless system in order to make wireless networks integral parts of the Internet. However, TCP is well known to suffer severe performance degradation in wireless networks. Some of the challenges wireless networks present are high bit error rates, and disconnections due to mobility and route changes.

As a result, new issues need to be solved at the various layers of the protocol stack:

1) High bit rate should be provided in order to let users access any kind of service;
2) Wireless Internet should work indoor and outdoor, for both fixed and movable users.

The complex performance of TCP over wireless has been the center of attention in recent years, and researchers experimenting with TCP are always intense to show the efficiency of their enhancements by displaying results from numerous tests over varying network conditions, largely making use of accepted network simulators, and seldom using wireless testbeds. The key issue for TCP over wireless is of elevated packet loss

rate of unsystematic nature on the transmission channels, which was not present in TCP's design from day one. Many approaches such as End-to-end [24], Link Layer [12], and Split Connection [21] have been proposed and their relative virtues were extensively assessed in recent studies. A conclusion evolved from these studies is that even though end-to-end techniques are not as successful as local recovery schemes in managing wireless losses, they are promising as noteworthy gains can be achieved without broad support at the network layer in routers.

As packet switched Internet is reaching more and more private and business users, TCP/IP is becoming a communication standard. Originally it was intended to provide consistent transmission of IP protocol operating chiefly in wired networks. However today, wireless networks are becoming more ubiquitous and even more often we see an unprecedented growth in heterogeneous networks. The challenge in modeling the performance of TCP over wireless channels [2] is the complexity in setting up a true wireless testbed and performing experiments, and gathering precise statistics on TCP's true performance of a server placed somewhere in the Internet transporting data over HTTP or FTP. Another challenge is in the acquirement of right equipment and tools to set up the wireless testbed. Hence, researchers frequently choose to simulate and emulate as the mode of experimenting with TCP, using predefined wireless channel models [4]. Once limited, Wi-Fi hotspots are widespread now easily connecting users to web pages, email, or file downloads from a larger wired Internet. The innovative communication standard is a wireless network communicating to a fixed wired host at an end point. Mixing wired and wireless networks underneath the TCP/IP presents numerous challenges. TCP was formerly proposed for wired networks which are of a very different

nature from wireless networks. A few of the challenges presented by wireless networks are elevated bit error rates, route changes, and disconnections due to mobility [6]. If placed in this setting without any modifications, TCP will offer a consistent transport; on the other hand, performance will decline considerably. Numerous solutions have been proposed to adapt TCP to pure wireless and heterogeneous networks. This research discusses advantages and disadvantages of some of the most engaging proposals as they relate to wireless LAN's, cellular, satellite and ad-hoc wireless networks.

2. PROBLEM STATEMENT

The traffic congestion can frequently be caused by TCP operation as it is a reactive protocol that tries to explore the network for maximum accessible bandwidth, overrun that bandwidth and then scale back. TCP Reno / New Reno were the most general variant of TCP used today [6]. The process begins in a slow start phase where a packet is sent from a sender to a receiver. It then waits for an ACK and increases the congestion window by one for each ACK. For this reason, there is an exponential growth in the congestion window until it hits a threshold value. Subsequently it grows linearly, increasing by 1 for each RTT. This window increase is one of the cornerstones of TCP referred to as the "*sliding window*" algorithm [11]. As packet loss or reordering occurs, the sender either receives duplicate ACKs or timeouts waiting for an ACK. Duplicate ACKs cause the congestion window to decrease by half, and retransmission timeout causes it to drop to 1. TCP enters a congestion avoidance segment where window only increases linearly, or a slow start is used where it increases exponentially but it takes a short time before reaching its previous value. Additionally to these, there are numerous

options that can be used for improving TCP performance such as Selective Acknowledgements, bandwidth estimation as implemented in TCP Vegas and others. Yet with these additions, TCP remains to be a poor performer in wireless networks.

In wired networks with fairly low bit error rates, sturdy signals and route flaps typically caused by irregular physical layer issues, TCP makes an assumption that if a packet is lost or reordered then it is a result of network congestion. The postulation is one of the main reasons to state that current TCP is an appalling fit for wireless networks. In such networks there are more ways how a packet or ACK to that packet is lost causing duplicate ACK or timeout. Wireless networks usually exhibit a higher Bit Error Rate which can occur due to low signal level or interference. Also, in contrast to universal wired networks, there is link irregularity in wireless networks where the link capability from sender to receiver may not be the same as in the reverse direction. For this reason, when the trouble may in fact lie in the reverse channel, congestion can be supposed on the advance channel. This results in a transfer rate which is needlessly reduced, and it takes time before the transmission rate is increased up to its prior point. In cellular networks having higher delays caused by lower bit rates and longer distances, we observe that slow ramp-up is a destroyer of performance. In contrast to WLANs, cellular networks have portable users which need continual service during hand-offs from one device to another. In such a condition, a few packets will be lost in a single RTT. This is likely for either voice or data applications on the top layer. TCP would significantly reduce throughput. In ad-hoc wireless networks, regular mobility causes frequent route changes, which results in the same common effect on TCP [11].

Keeping in mind all the issues described above, this research project discusses certain solutions which have key applications in Wireless LANs/WANs. Split Mode solutions [20] are also described, many of which are tailored to issues in cellular and satellite networks. The report presents some techniques which maintain TCP as a continuous protocol but allow it to seek feedback from lower layers of the network and transitional nodes. The proposed solution will be demonstrated with an experimental implementation.

3. LITERATURE REVIEW

After much literature review in this area, it was discovered that researchers time and again made many assumptions about the situation in which they are testing TCP. Some of the key observations made were that researchers were infrequently making references to the packet error model that was being used on the feedback channel of experiments and tests were being conducted with importance on packet losses taking place only on the forward course of TCP connections.

3.1 Feedback Channel Losses for Wireless TCP

The paper "A Feedback Based Scheme for improving TCP Performance in Wireless Networks" by Chandran, K. et all, offered some preliminary results from TCP experiments over an emulation testbed that evidently stress the importance of packet losses on the feedback channel when testing the performance of TCP over wireless channels. The observations are meant mainly at those working with simulation tools and emulation test beds.

Numeral issues of practical apprehension have been highlighted, and it seems that researchers in this area are fully aware of the implications of not including a packet loss model for the reverse path of TCP experiments. The dilemma with TCP and the feedback channel is comparatively simple; TCP is a consistent transport protocol and uses the notion of acknowledging all data that is transmitted (or received) effectively. As a sender transmits a TCP data packet in the onward direction to a receiver, the receiver transmits an acknowledging packet, upon its successful appearance at the TCP layer; back to the sender i.e. provides feedback. The research displayed the differences in performance that

is created when experiments are performed with and without packet losses on the feedback channel of a TCP connection. A "Feedback Based Scheme for enhancing TCP performance in Ad-Hoc Wireless network" is also optimized for ad-hoc networks [22]. Authors message the identical problems in ad-hoc networks that make TCP to perform poorly as discussed previously: packet loss as a result of high BER, recurrent and random route failures. They put forward TCP-Feedback (TCP-F) where nodes send Route Failure Notification (RFN) messages to sender as route is interrupted, enabling TCP to go into persist state and freeze its timers. After route is re-established, the source receives a Route Re-establishment Notification (RRN) allowing it to restart transmission from where it stopped. Thus, allowing TCP sender not to face serial timeouts. Here authors deal with the case of numerous route failures probably occurring along diverse links. They comment that this will not have much poorer affect than single route failure since sources at all times receive RFN from adjoining failure point and get RRN from that breakdown point node only when route is re-established. In case of congestion duplicate ACKs come but no RFNs. Another difficulty is to inform all hosts that use a route that has failed. Authors were still working on finest solution for that and while this paper is a good proposition; it fails explore how TCP-F would work if fixed network is part of connection.

3.2 The Snoop protocol approach

Balakrishnan et .al. incorporate a transport layer aware agent (snoop agent) [4]. A *snoop* agent is introduced at the link layer (LL), which monitors the TCP connection, caches TCP segments that have not been acknowledged yet, suppresses duplicate

acknowledgments, and retransmits lost segments. The key benefit of this approach is that it forces down the duplicate acknowledgments for lost TCP segments that are retransmitted locally. Nevertheless, the snoop agent must be situated right ahead of the TCP receiver. Thus, as the mobile node transmits data to a distant receiver, TCP acknowledgments are returned too late for a proficient recovery of the missing segments. Authors present a resolution with support of the explicit notification to the TCP correspondent of link-related losses so as to evade the establishment of the congestion control measures. The loss is reported by setting the Explicit Loss Notification (ELN) bit in the header of the TCP acknowledgments. The disadvantages of such a method are that modifications in router algorithms are required and, as soon as an ELN is returned, a lost fragment can be retransmitted only after a round-trip time interval. Balakrishnan et al. [4] integrate a transport layer conscious agent (snoop agent) at the base station. The snoop agent grabs the TCP packets intended for the mobile host and executes limited retransmissions after losses are perceived by replica acknowledgments and timeouts. On the other hand, a timeout can take place at the sender source, and congestion control actions are raised, while the snoop agent tries to retransmit lost packets to the mobile host. In addition, both snoop and the split-connection techniques do not execute well in the existence of bursty losses on the wireless links.

3.3 TULIP

The paper “TULIP-A link-level protocol for improving TCP over wireless links” by C. Parsa, and J.J. Garcia-Luna-Aceves states that the Transport Unaware Link Improvement Protocol (TULIP) is analogous to the snoop protocol: both try to enhance

the performance of TCP above lossy wireless links with no need of modifying the transport layer protocols [15]. On the other hand, unlike snoop protocol, that requires an alternative between the sender and receiver hosts to assist TCP's performance, TULIP does not utilize this substitute approach. It is significant to note that the necessity for a transport-level proxy in snoop outcomes a requirement to maintain per-session status to aggressively observe the TCP packets and restrain any duplicate ACKs it comes across. This is the precise cause why the Snoop's link layer retransmission is strongly united with the high layer protocols. Conversely, TULIP is "service-aware" since it offers reliability for packets that necessitate such examination, but is not "protocol-aware" as it does not identify any information of the specific protocol to which it presents its consistent service.

Exclusively, TULIP provides both consistent service for packets transporting TCP data traffic and unpredictable service for added packet types, such as User Datagram Protocol (UDP) data traffic and TCP acknowledgments. TULIP does not present trustworthy service to TCP ACKs as successive ACKs succeed the information in the lost ACK. The recipient merely buffers packets and passes them to the subsequent layer in order, thus avoiding TCP from causing duplicate ACKs in the incident that a packet is absent from the anticipated sequential packet flow. This capability of TULIP to preserve local recovery of all missing packets at the wireless linkage to avert the needless and deferred retransmission of packet through the complete path and a successive decline in TCP's congestion window is analogous to SNOOP's capability to conceal losses from TCP.

3.4 The Split Connection approach

The *split-connection* approach proposed by M. Luglio et al. [20] states that a TCP link connecting a mobile host and a preset host should be split into two separate connections: One, among the mobile host and the base station over the wireless means and other involving the base station and the fixed host over the wired medium. It is revealed that this facilitates in improvement of TCP performance. However, the split-connection approach disobeys the semantics of end-to-end reliability. This is due to the fact that acknowledgments can emerge at the resource ahead of the packet basically arriving at the expected destination. Secondly, this approach necessitates loads of state maintenance at the base station. In mixed networks, wireless element is frequently less reliable and develops into a bottleneck. That is why the majority recommends some sort of intermediary node on the edge of wireless and wired networks where TCP link is split into two. This permits wired network to be secured from the problems of wireless network. Additionally, split mode approach may deal with the cellular network subject of steady hand-offs and mobility in linking. The aforementioned intermediate node would be in improved situation to handle such circumstances and provide appropriate Layer 4 response to the sender. Even as split mode approach can be used for other wireless networks, because of explanation stated above, the authors appear to focus on cellular and satellite networks as superior fit for this model.

M-TCP is one of the most eminent split path TCP performance improvement solutions proposed by Kevin Brown and Suresh Singh [18]. From the start, they confine the argument to wireless networks where users are forever on the move such as cellular, as contrasting to ones where immobile users move from time to time like WLANs.

Authors also believe that too much thought has been given to dilemma of high BER and not adequate to periodic disconnections and their affect on TCP performance. The proposed architecture is shown in Figure 1. It consists of Mobile Hosts (MH) which communicates with one Mobile Support Stations (MSS) per cell.

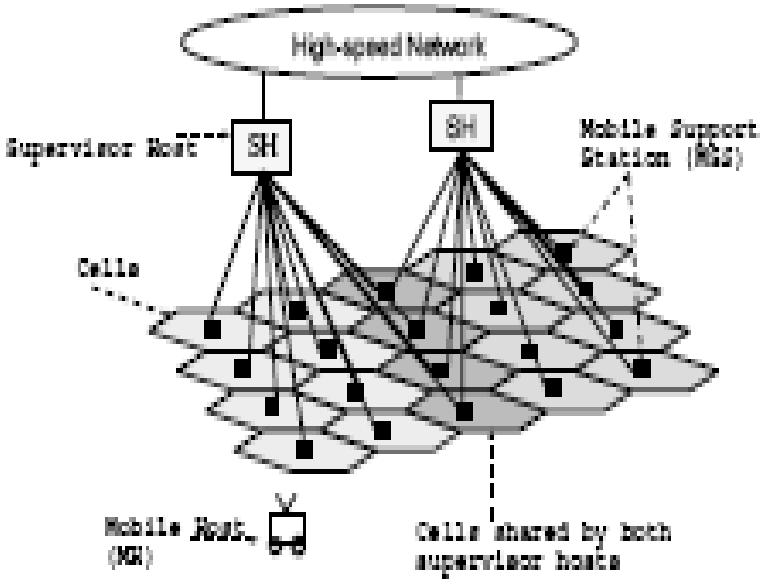


Figure 1. Architecture of MTCP [18]

3.5 Link-layer recovery

The paper, "Wireless TCP performance with link layer FEC/ARQ" by A. Chokalingam, Mzorzi, and V. Tralli, states that in case of multiple TCP associations which share the wireless link, scheduling protocols such as Round-Robin offer major performance enhancement over FIFO [12]. The key restraint of this approach is that the performance enhancement attainable depends generally on the exactness of the channel state interpreter. The trouble of source timeouts is present in this approach too. Besides end-to-end and split path, there is another class of resolution that is still prominent, although less widespread. These schemes do not amend TCP code as much but as an

alternative rely on Layer 2/3 feedback to make conclusions on packet loss character and suitable action to take. Identical to M-TCP, these papers consider that it is superior for error correction to be concerned of by link layer and congestion notification to be completed by network layer. While such resolutions can be functional to all types of wireless networks, from existing research it appears that wireless ad-hoc LANs have the most established application for this approach. Ad-hoc networks comprise of mobile hosts that communicate with no support nodes such as base stations or routers. In such an uncommon environment, an enhanced TCP is still needed to present reliable data transfer. Authors concentrate on a single TCP connection, where the receiver exists at the mobile terminal. Prior to reaching the fixed network access point, called Base Station (BS), TCP fragments navigate the wired network from which they identify an average delay and average loss probability (Figure 2). At the BS, TCP data segments depend on the Link Layer protocol to get to the receiver through the wireless link.

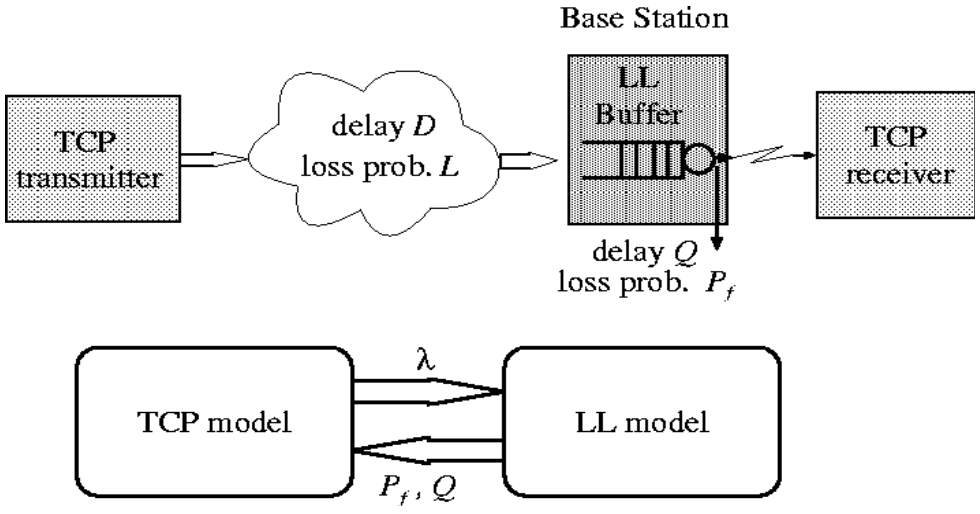


Figure 2: Link Layer recovery mode [12]

"ATCP: TCP for Mobile Ad Hoc Networks" proposed by Jain Liu and Suresh Singh [21] is centered on the idea of using Layer 3 for feedback to TCP in ad-hoc wireless networks. The authors study the problems of ad-hoc networks that may decrease TCP performance. The algorithm identifies that drops occur due to lossy link because, in case of congestion, it anticipates Explicit Congestion Notification (ECN) flag to be set in the duplicate ACKs [8]. In such situation, sender follows standard fast retransmit or fast recovery logic that is present in TCP Reno. Disconnections frequently occur in ad-hoc network: either short ones as a result of route re-computation or longer ones owing to network partitions. In such cases, routes are recomputed and consequently the congestion window is recomputed as well to present best possible performance on a new path. In case where connection navigates through fixed networks, ATCP will only function appropriately if ECN is sustained all the way. If that is not the case, connection will be split at the edge node, much like as in I-TCP [19]. During experimentation, it was established that ATCP gives much superior performance than standard TCP, mainly due to smaller number of congestion control invocations [17].

3.6 Explicit Notification

Another proposal by H. Balakrishnan, S. Seshan, and R.H. Katz on "Improving reliable transport and handoff performance in cellular wireless networks" [14] mentions that as TCP infers all losses as congestion in the network, different explicit notification proposals have been projected to allow the TCP sender to recognize the real basis of a data loss. The thought is that an explicit notification is sent out with the intention of informing the sender regarding data loss due to errors in the wireless network.

3.6.1 ICMP Messaging

In preference of trying to conceal problems due to a wireless link, Internet Control Message Protocol (ICMP) suggests that an explicit notification is sent out to the TCP sender. The thought is that the intermediary node that joins the fixed and the wireless network has information regarding the wireless transmission accomplishment, and should be liable for creating and transmitting explicit notifications. A latest ICMP message form, ICMP-DEFER, is employed for the explicit notification. If data goes missing over the wireless network, an ICMP-DEFER is transmitted to the TCP sender, which delays the termination of the retransmission timer. This evades disagreement between the link level retransmissions at the base station and end-to-end retransmissions. If a segment requires retransmission and ICMP-DEFER has been accepted, then the cwnd is not decreased at the sender. The method is additionally improved by adding an additional ICMP message, ICMP-RETRANSMISSION. The base station transmits an ICMP-RETRANSMISSION to the TCP sender when the utmost number of retransmission efforts is attained and the packet is discarded.

3.6.2 Explicit Loss Notification

Explicit Loss Notification (ELN) can be utilized if the mobile host is the sender. If information is lost on the wireless connection, the base station transmits an ELN to the mobile host. The base station inspects TCP headers and maintains track of the sequence numbers. A gap in the sequence space specifies that data must have been misplaced over the wireless link on route from the mobile station to the base station. The fixed host transmits a dupack in response. The base station sets an ELN flag in the dupack header

prior to dupack is advanced to the mobile station. The lost segment is retransmitted by the mobile station then but does not decrease its congestion window.

3.6.3 Syndrome

Syndrome is a amendment of the base station to facilitate discovery of packet loss over the wireless link. The base station counts the quantity of packets it transmits, and includes this counter as a TCP option. This counter is referred to as a ‘syndrome’, and is made use of collectively with the sequence number to establish if a loss happened in the wired or wireless part of the network. If a break occurs in the syndrome counter, this shows that packets were lost on the wireless part. While breaks in sequence numbers, but not in the Syndrome, signify that segments should have been lost in the fixed network. Explicit loss notification is subsequently used by the receiver to update the sender about the loss.

3.6.4 Partial Acknowledgments

Another way to differentiate congestion from data loss because of wireless link is to establish new types of acknowledgments. The scheme uses one added acknowledgment, a Partial Acknowledgment that is transmitted by the base station in reply to data from a sender in the fixed network. Considering that no segments are lost, the sender obtains two acknowledgments for each segment: a Partial Acknowledgment from the base station and a Total acknowledgment from the mobile host. If just the partial acknowledgment appears, then the sender infers that data must have been missing over the wireless network and congestion control action is not required. If no

acknowledgments turn up, then the probable cause is data loss because of congestion. If the mobile station be the receiver, then the base station sends out a last hop acknowledgment in response to the permanent host. The method works as well if the mobile station is the sender. Therefore, the base station transmits an initial hop acknowledgment to the mobile station. While in the partial acknowledgment scheme above, the sender obtains two acknowledgments for each effectively transmitted segment, one from the receiver and one from the base station.

The explicit notification schemes have a diverse viewpoint in contrast to most of the other proposals. The sender can differentiate congestion from data failure over the wireless network, as it obtains information about the transmission status. It does not solve the trouble with the elevated unpredictability of the wireless network, but the sender identifies it and is capable of making a more informed decision. One drawback of the explicit notification schemes presented above is that they presume that the base station is TCP conscious. The only exclusion is ICMP messaging. Explicit notification supposes a low down persistent link ARQ, such as in WLANs.

3.7 Categorizing the Packet Loss

Another scheme is presented by A. DeSimone, M.C. Chuah, and O.C. Yue, “Throughput performance of transport layer protocols over wireless LANs”. As performance deprivation of TCP in wireless links is the reason for TCP misreading random wireless errors as congestion, the most apparent approach to solve this is to integrate methods such that TCP responds differently to a wireless drop. When the packet loss is verified to be a result of an arbitrary wireless error and no congestion is essentially

present, it signifies that the network can hold the current transmission rate. In such cases the TCP should not react by merely decreasing the *cwnd* because it may be more favorable for the system to preserve the current transmission rate. This strategy, thus, engages two things; first, determining if the packet loss is as a result of congestion or arbitrary wireless error and secondly, by acting suitably based on what kind of error it is. Methods for error classification can be of three forms:

1) A computation at the sender which absolutely categorizes the loss,
2) An explicit "wireless loss" broadcast sent to the sender from some other component of the network and
3) An explicit "congestion" announcement sent to the sender by some additional element of the network.

The three types are studied with instances on how they were precisely implemented by various authors. Samaraweera presents a method to implicitly categorize the error called Non-congestion Packet Loss Detection (NCPLD) algorithm [2]. In the computations, the author employs the conception of the K*nee Point* of a network. The knee point is the point in the communication rate against throughput graph wherever the network executes at optimal power. Previous to the knee point, no congestion is there and consequently an increase in transmission rate creates a big increase in throughput and the round trip delay stays relatively constant. Subsequent to the knee point, packets necessitate to be queued at the routers and thus an increase in round trip setback is expected.

Bansal et al. present another method for the TCP sender to implicitly resolve the basis of the packet loss [5]. This system, conversely, can only be applied to a previous

version of TCP, TCP Tahoe. In *TCP Tahoe*, fast recovery algorithm is not present. Subsequent to fast retransmit, it sets the *cwnd* to 1 section then carries out a slow start guiding to an exponential enhance until *ssthresh*. The algorithm of Bansal et al. follows the *ssthresh* values throughout packet and the time between packet drops.

Another way of error categorization is to have the network with either the base station or the receiver informing the TCP sender about the error being caused by a wireless drop. The TCP sender gets an Explicit Wireless Loss Notification (EWLN) [4] for it to know that a wireless error occurred. The benefit of having the EWLN in the TCP header otherwise as an ICMP message is backward compatibility. Older version of TCP tends to reject the notification unambiguously. To one side from the subject on how the EWLN message will be sent, either as a TCP preference, an ICMP message or a dissimilar form, another key issue in EWLN methods is which fraction of the network sends the notification.

An alternative to the EWLN is the Explicit Congestion Notification (ECN) system. In place of receiving a message indicating a packet loss because of wireless error, the TCP sender anticipates to receive an ECN which indicates congestion someplace in the network. The differentiation between WECN and ECN is that in the entirely wired network, it is projected that the TCP sender must pursue the usual TCP algorithms on detection of other congestion signals (timeouts and duplicate ACKs) even with no presence of the ECN. In WECN, the need of the WECN means that a wireless error occurrence is detected and therefore it should not cite the standard congestion-control mechanisms. Owing to this, a huge drawback of WECN is that all routers must be WECN compliant for the proposal to work successfully.

3.8. Other Proposed Schemes

In above schemes the authors demonstrated logically the performance of TCP as a task of the size of wireless-wire line interface buffer. They disputed that the amount of interface buffer required to deal with channel time variations ought to scale logarithmically with the bandwidth-delay product to achieve high link utilization, while the size of buffer required to achieve high utilization using TCP in a wired network must balance linearly with the bandwidth-delay product. Some experiments disputed that link layer with FEC and ARQ will enhance the throughput of TCP performance as expected [12]. Nonetheless, the tradeoffs are complexity and bandwidth inefficiency but benefit is improvement in throughput and superior energy effectiveness.

As networks get larger and more varied, reliable data transfer develops into a larger concern and a more composite challenge. Unfortunately there is no solitary scheme that will fit all circumstances. Hence a thriving and feasible solution will have to integrate the finest from each approach and relate these ideas selectively based on explicit wireless network.

Recovering TCP performance in wireless network links is an essential concern that needs to be addressed as wireless applications are getting more accepted. The standard TCP fault recovery algorithms essentially reduce the throughput in wireless links as TCP presumes that all losses are attributable to congestion. In wireless channels, arbitrary errors because of fading and shadowing are largest. Different methods have been displayed to adapt TCP to wireless networks. After reviewing the different proposals, it was concluded that the most favorable end-to-end scheme includes the

subsequent characteristics: a suitable error classification method which will determine whether the packet loss is caused by congestion or wireless errors, a retransmit then retain *cwnd* reply to a wireless packet loss, selective acknowledgements [9] to recover the reply to multiple packet losses in a solo window and a simple technique to perceive long deep fade durations. The most favorable link-layer scheme should be TCP aware and executes local retransmission by means of selective acknowledgments. Usage of suitable buffer size at the edge of wired and wireless link as well FEC and ARQ at the link layer is also essential for an optimal link-layer scheme.

4. PROPOSED SOLUTION

The Fiber Channel protocol was developed to deal with the speed and distance constraints of traditional serial and parallel SCSI (Small Computer System Interface). On the other hand, Fiber Channel itself has restrictions when it comes to expanding the achievement of storage systems over larger distances or lower bandwidth links. The SCSI identifies optical interfaces and protocols for physical associations and transport of data.

4.1 iSCSI Protocol

The iSCSI protocol performs the transfer of data, commands, and status information over TCP/IP networks. iSCSI is not intended as an alternative for Fiber Channel in a SAN, but instead as a cost-effective resource for IP-enabled hosts to contact block-level storage over a TCP/IP transport. iSCSI is an ultimate option for applications and servers that do not create extensive I/O. The SCSI protocol is extensively used to access storage devices. The iSCSI protocol is a transport for SCSI over TCP/IP. Other SCSI transports contain SCSI Serial and Fiber Channel Protocol (FCP). iSCSI permits storage to be accessed over a storage area network (SAN), granting shared admission to storage. A key benefit of iSCSI over FCP is that iSCSI can run above usual off-the-shelf network mechanism, such as Ethernet. In addition, iSCSI can utilize present IP-based protocols such as IPSec for security and Service Location Protocol (SLP) for discovery. IP-based SANs using iSCSI can be administered using present and familiar IP-based tools such as Simple Network Management Protocol (SNMP).

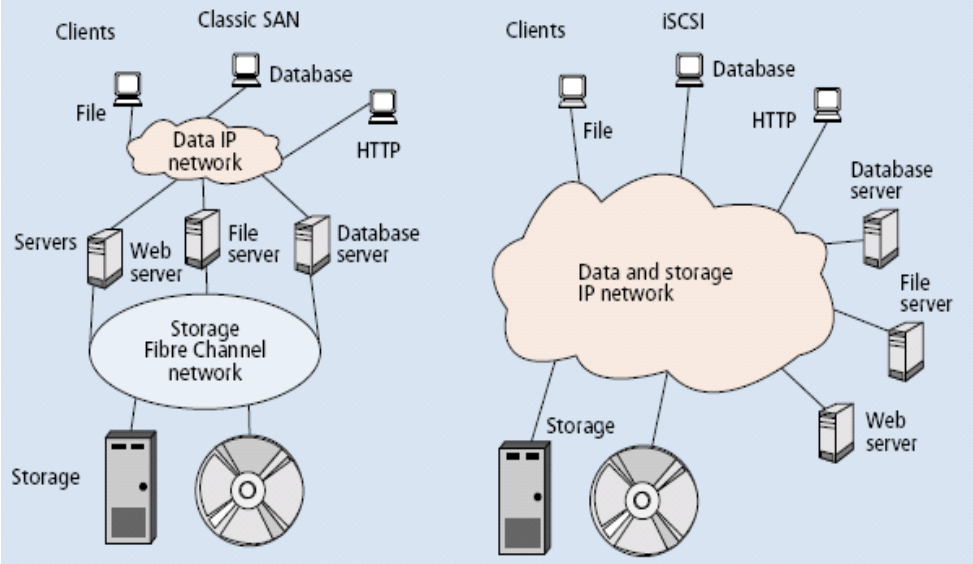


Figure 3. SAN vs iSCSI [30]

TCP was selected as the carrier for iSCSI. TCP has several properties that are employed by iSCSI:

- TCP offers consistent in-order deliverance of data.
- TCP offers regular retransmission of data that was not acknowledged.
- TCP is a responsive network resident because it provides the essential flow control and congestion control to evade overloading a congested network.
- TCP works above a large variety of physical medium and interconnects topologies.

Additionally to reduced costs and an integrated network framework, iSCSI permits the operation of storage networks above a commodity internet. Furthermore, there are economical software implementations of the iSCSI protocol so as to provide convincing proposal for iSCSI operation. The protocol effectively summarizes a SCSI subsystem

contained by a TCP/IP connection. This permits for superior flexibility inside storage area networks as they can now be applied by means of less expensive components.

In contrast to the desktop equivalent in wired setting, portable devices have formed new challenge for data accessibility because of limited storage capacity, low bandwidth, and unreliability. Few studies have attained elevated storage performance and network exploitation in the wireless network‘s restricted bandwidth. There are at present two methods of wireless network storage. The first one is to merely utilize a network file system such as NFS (Network File System). In this method, the server creates a subset of its limited namespace accessible to clients. Clients contact metadata and records on the server by means of RPC (Remote Procedure Call) [27]. This storage architecture is usually named as NAS (Network Attached Storage).

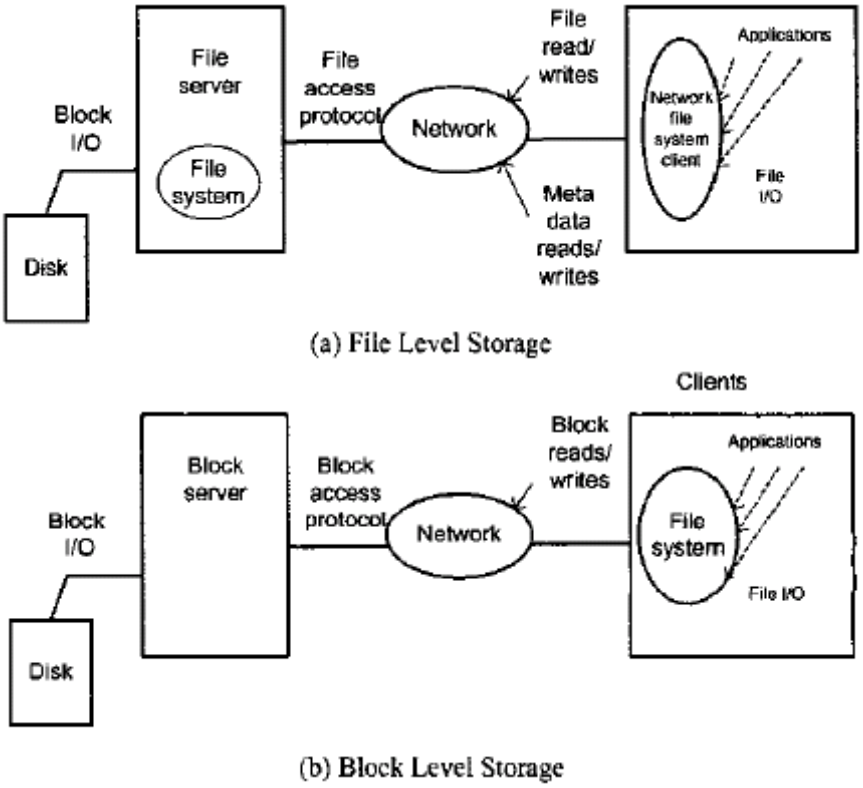


Figure 4. File level And Block Level File storage [27]

Besides the reduced costs, iSCSI also offers good scalability and simple way to apply remote backup and failure recovery. Many studies and projects have been conceded to investigate and employ iSCSI. Some resolutions have also been proposed to consider the performance concern. A research team proposed a resolution to utilize memory of iSCSI initiator to store iSCSI information. Owing to the restricted bandwidth of wireless network, all additional system resources are influential adequately as compared to the limited bandwidth to make it likely to optimize the storage performance in wireless environment for iSCSI software resolution.

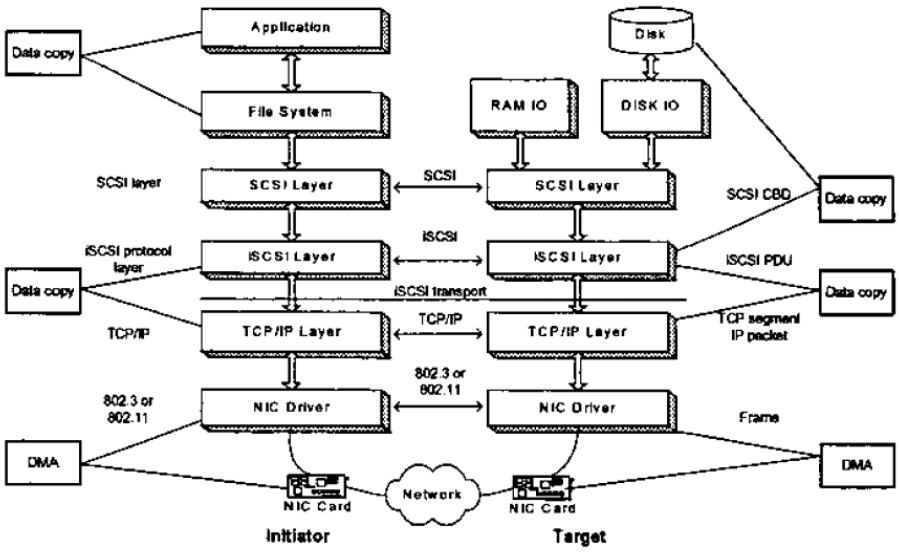


Figure 5. iSCSI Model [27]

iSCSI is assembled above TCP/IP layer. For iSCSI communication among an initiator and a target, it requires to institute a session. The data and command interactions occur within the context of the session. In initiator, the application issues file requests.

The file structure translates file demands to block requests starting application layer to SCSI layer. The SCSI command implementation comprises of three phases: Command, Data and Status Response as revealed in Fig. 6.

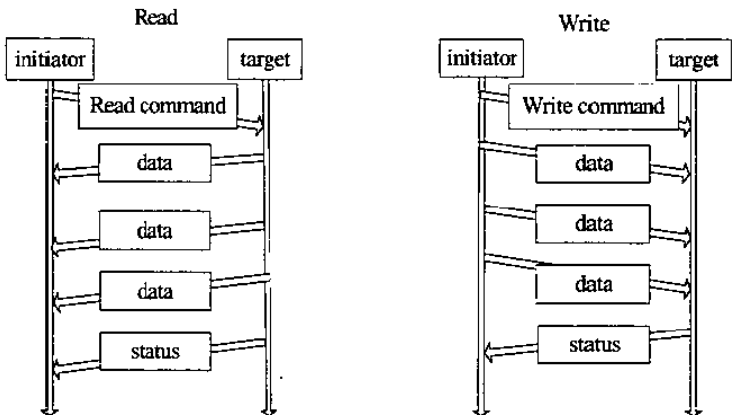


Figure 6. SCSI Command Sequence[27]

4.1.1 iSCSI Protocol Data Units

iSCSI characterizes its individual packets, referred as iSCSI Protocol Data Units (PDUs) [30]. iSCSI PDUs comprise of a header and probable data, wherever the data length is provided in the iSCSI PDU header. An iSCSI PDU is transmitted as contents of one or added TCP packets. The main frequently used iSCSI PDU types are:

- SCSI Command/Response
- Data In/Out
- Ready to Transfer (R2T)
- Login Request/Response

The SCSI Command PDU is employed to transport a SCSI command from the initiator to the target. The originator iSCSI driver surrounds SCSI commands into iSCSI

Protocol Data Units (PDUs) and transmits them to the network through TCP/IP layer. Subsequent to getting iSCSI PDUs from the TCP/IP layer, the objective iSCSI driver de-capsulates them, and subsequently SCSI commands are mapped to the storage mechanism. The target driver afterward sends reply data and status back by means of TCP/IP layer. iSCSI constraints, for instance PDU size, MaxBurstLength, FirstBurstLength, and the fundamental TCP flow manage algorithm, greatest frame size and MAC mechanism considerably influence iSCSI performance. There are information copy and one DMA process in the initiator through one I/O access and there are data copy and one DMA process for RAM I/O and in similar way two data copy and one DMA process for disk I/O on the target side. The SCSI Command PDU is utilized to transport a SCSI command from the originator to the target. If the SCSI command sends request to interpret data from the target, the target will then transmit the data to the originator in single or multiple Data In PDUs. If the SCSI command desires to write data to the target, the initiator sends the data to the target in one or more Data Out PDUs. The target may indicate to the initiator about what part of the data to send by transporting to the initiator an R2T PDU. On conclusion of the complete data transfer, the receiver sends a SCSI response PDU to the initiator signifying either successful conclusion of the command or some error state detected. For every SCSI Command PDU there is an equivalent solitary SCSI Response PDU, except perhaps multiple or none Data PDUs. SCSI Data and Response PDUs ought to be sent above the equivalent TCP connection on which their consequent SCSI Command PDU was issued.

Immediately after setting up a TCP connection among an iSCSI initiator and iSCSI target, a login process have to be executed. A Login Request PDU is sent by an

initiator to the target. The initiator and target possibly will validate each other and can discuss operational parameters. A defaulting authentication method, Challenge-Handshake Authentication Protocol (CHAP), should be sustained by all compliant iSCSI implementations. Several operational parameters that possibly be negotiated are greatest size of data PDUs, the highest number of connections designated in the session, the quantity of unwanted data that can be possibly sent by the initiator not using R2T, the intensity of error recovery sustained, and whether or not digests can be used for error recognition. Once sides are satisfied mutually with the validation and the set up of the operational settings, the target transfers a Login Response PDU with a signal indicating that the login process has concluded. Only then the connection could be used to pass SCSI commands and information.

iSCSI employs a URL-like method to name destinations. iSCSI names are intended to exist as global, analogous to World Wide Names applied by Fiber Channel. An iSCSI body may possibly have its IP address altered even as its name is retained. An iSCSI unit is thus recognized by its name and not by its address. This permits easier treatment of iSCSI names by alternatives, network address translation boxes, gateways, firewalls, and so on. While using storage procedures over a network, one has to take care of the capability of an initiator to notice the procedures it may use. One method requires an administrator to arrange the initiator statically, offering the initiator with a record of the names and addresses of the iSCSI means to which the initiator may connect. If further iSCSI devices are added later to the network, the statically constituted initiator would not be capable of accessing the latest devices with not being reconfigured. A substitute and more dynamic technique is to use Service Location Protocol (SLP) [33], which

previously subsist in the IP family of protocols. iSCSI targets can be registered by themselves using SLP, and initiators can question SLP representatives to attain information regarding registered targets. In this fashion, iSCSI targets are able to add to the network, and over time the topology can be changed, however initiators can effortlessly locate new targets with not a need for reconfiguration. An analogous method is given by the lately defined iSNS protocol. A supplementary breakthrough mechanism, SendTargets, is shown in the iSCSI protocol, particularly helpful for gateway devices. In this technique, an initiator is configured statically to connect to explicit iSCSI gateway devices. A discovery session is established by the initiator with the iSCSI gateway device, and subsequently sends the SendTargets appeal to the iSCSI gateway device. The iSCSI gateway tool then replies with an attached iSCSI target lists that are offered to the initiator. The initiator might then continue to connect to the specific iSCSI target devices.

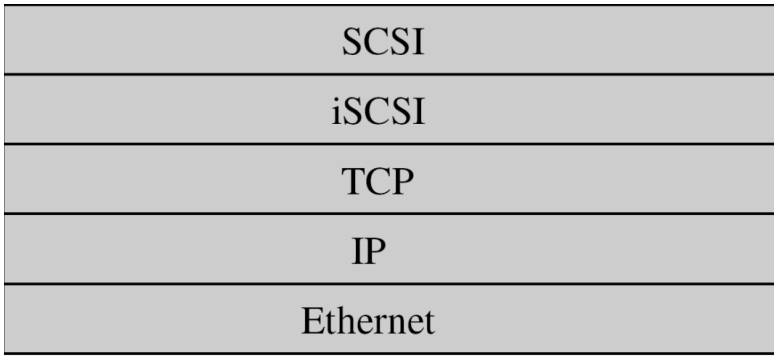


Figure 7. iSCSI protocol stack [35]

It is considered as a write and read demand. A SCSI write request sets off the iSCSI write command PDU transmission to the target. After the receipt of the command, the target assigns buffers for the transport and replies with one or more (R2T) PDUs.

Every R2T PDU is an authorization to the initiator to transmit a segment of the data linked with the command. The initiator reacts to a R2T PDU by transmitting a series of Data-Out PDUs consisting of the data requested. Lastly as all the data has been moved from the initiator to the target, for the command, an iSCSI Response PDU is sent by the target, indicating successful conclusion of the command. The command conclusion status information is passed to the iSCSI layer at the initiator. The Read transport has analogous process with the Write one apart from using R2T process. The transmission of iSCSI read command PDU is triggered to the target. The target, on receipt of the command, sends out Data-In PDUs constantly consisting of the data requested from iSCSI read command. Those are illustrated in figure 8.

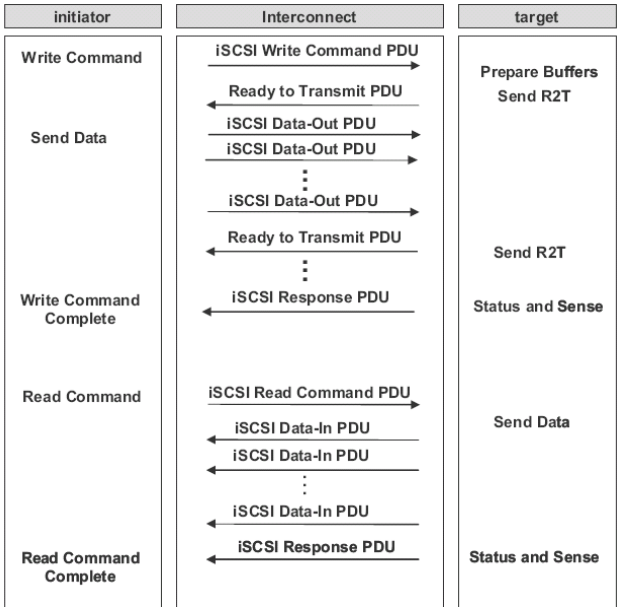


Figure 8. iSCSI Write/Read Operation [35]

4.1.2 iSCSI Parameters

The standards of iSCSI constraints can be found out through login phase and full feature phase [35]. Every iSCSI connection commences with a login phase. The target

and initiator can discuss iSCSI parameters to enhance performance through login phase. Subsequently, iSCSI goes into the complete attribute phase, throughout which iSCSI instructions and data are switched over the conventional iSCSI connections. iSCSI can modify the significance of several parameters throughout the full feature phase in accordance with the state of networks. There are two classes of iSCSI parameters. One is associated to iSCSI read procedure whereas the other is related to iSCSI write operation.

Parameters associated to iSCSI read procedure:

iSCSI read operation has two parameters.

Number of sectors per command: The majorities of SCSI disks identify a sector size, and need I/O procedures selected in multiples of a sector. The iSCSI initiator is frequently needed to state a bound on the number of sectors in a sole SCSI I/O procedure. This forms one of the significant parameters for iSCSI read procedure, as the value of the parameter bound the application size of an iSCSI read command. The target is capable of constantly sending out the data asked for from an iSCSI read rule using Data-In PDUs. It is not openly linked to iSCSI write process. However an iSCSI write command call for the data communication from initiator to target, R2T method has control over the transmission extent of the data linked with the command.

MaxRecvDataSegmentLength (MRDSL) of initiator: Every iSCSI PDU has one or more header segments and, preferably, a data fragment. The first segment is the Basic Header Segment (BHS) out of all the iSCSI PDUs. The BHS is a preset length 48 byte header. It can perhaps be tagged on by a Header-Digest, Additional Header Segment (AHS), a Data-Digest and a Data Segment. The greatest data segment length in an iSCSI PDU is

stated by the initiator. Therefore, a Data-In PDU's DataSegmentLength should not go beyond MaxRecvDataSegmentLength of the originator.

Parameters related to iSCSI write procedure

An iSCSI Write operation has two parameters:

MaxBurstLength: The highest SCSI data payload is conferred among the initiator and the target, in bytes, in a requested Data-Out iSCSI series. A series comprises of several successive Data-Out PDUs that are ending by a Data-Out PDU with the F bit positioned to one. The conclusion of a series of Data-Out PDUs needs the broadcast of a R2T PDU to the initiator through the target prior to the beginning of subsequent data-out sequence. Thus, the assessment of the MaxBurstLength constraint bound the whole quantity of data fragments of all PDUs in a requested data sequence by a R2T PDU.

MaxRecvDataSegmentLength (MRDSL) of target: The target announces the greatest data segment length within an iSCSI PDU. As a result, the Data-Out PDU's DataSegmentLength should not go above MaxRecvDataSegmentLength of the target.

Several parameters associated with the conduction of iSCSI PDU have an effect on the iSCSI based remote storage system's performance. The enhancement of the MaxRecvDataSegmentLength (MRDSL) data is capable of getting performance enhancement to iSCSI related remote storage system in steady wired networks. In read process, the requested data is sent by the target to the initiator after getting a read command. The data will be then divided into several iSCSI Data-In PDUs. The DataSegmentLength of an iSCSI Data-In Protocol Data Unit should not surpass MRDSL. At that point in time, the rise of the MRDSL value will reduce the number of iSCSI Data-

In PDUs for the convey of the request data, consequently reducing additional PDUs broadcast operating cost and additional processing overhead on either part. The result is performance enhancement of iSCSI implementing remote storage system. The rise of the Number of sectors for each command for read process and the rate of the MaxBurstLength for write process may also guide the reduction of further overhead in the identical approach as MRDSL. The performance is also improved in steady wired network. Conversely, this is not right each time in unsteady wireless networks having characteristics of an elevated bit error rate contained by the wireless channel, and a fine and erratic bandwidth of the wireless channels.

4.1.3 Data Integrity and Security

TCP have a checksum capability for detection of faults that arise during transmission. Whereas the possibility of the TCP checksum's failure to perceive a fault is fairly little, it is not fairly adequate for a few storage environments. The TCP checksum is also incapable of providing security for corruptions that crop up as a message is in the memory of a router where header data may be recalculated, and the information is no more confined by checksum. iSCSI consequently states its personal Cyclic Redundancy Check (CRC) checksum to guarantee end-to-end reliability of its packet headers and data. The targets and initiators might discuss whether to utilize this CRC checksum or not.

While storage devices were attached directly to host machines, the data present on the storage devices was supposedly secured while being remote to the external world. By means of iSCSI connected storage devices, this issue is no longer present. A grave safety difficulty might occur if responsive storage data is admitted above a universal data

network. One probable resolution is to make use of a physically separated network for the storing data, analogous to what is used with Fiber Channel [33]. This resolution needs another physical IP network that is economical than another physical Fiber Channel network. On the other hand, a sole physical IP network might be used mutually by encrypting the storage data. IPSec provides the encryption of data on an IP network. iSCSI basically makes use of the current IP security protocol to defend sensitive storage data from probable security assault such as sniff and spoof.

iSCSI was planned to permit proficient hardware and software applications to access I/O procedures involved above any IP network. iSCSI was as well intended for an extensive diversity of surroundings and applications which include local and remote representation, local and remote storage admittance, and local and remote backup/restore. iSCSI Data PDU headers include enough information which allows an iSCSI adapter to execute straight through data position. The information supplied in an iSCSI Data PDU header comprises of a transmit tag which identifies the SCSI rules and its equivalent buffer, a byte counterbalance in relation to the start of the consequent buffer, and a data length constraint representing the number of bytes getting transported in the recent data packet. The information is adequate to allow direct positioning of the incoming data into preregistered SCSI-provided buffers. An iSCSI adapter which carries out both TCP and iSCSI dispensation on the adapter has adequate information in the TCP and iSCSI headers to place arriving iSCSI data straight into the suitable SCSI buffers with not having to duplicate the data into added provisional buffers on the host machine [37].

4.1.4 Recovery by iSCSI

The iSCSI protocol identifies many stages of recovery to offer flexibility in the appearance of an extensive range of probable errors and failures. iSCSI error handling and recovery is anticipated to be an exceptional incidence, and might engage a major amount of overhead. It is projected that the majority of computing settings will not require all the stages of recovery stated in the iSCSI design.

The main revival class is Session Failure recovery. All iSCSI designs acquiescent to implementations should employ session failure recovery. Session recovery engages the finishing of every session's TCP connections, terminating all pending SCSI commands on that session, concluding all such called off SCSI commands with a suitable SCSI examination reaction at the initiator, and resuming a novel set of TCP connections for the specific session. Applications might execute session failure recovery for every iSCSI error discovered.

A less radical recovery implementation might be achieved is Digest Failure Recovery [30]. On an encounter of a CRC checksum error on iSCSI data, the data packet should be removed. As an alternative of carrying out session recovery, implementations might employ the Digest failure recovery method to request the linking peer to retransmit the lost data only. Likewise, if a sequence response timeout takes place, a related method might be used to inquire the linking peer to retransmit missing information, responses, or additional numbered packets that are anticipated. If a CRC checksum fault is noticed on an iSCSI packet header, the packet necessarily needs to be discarded as it was ruined. Consequently, synchronization among the target and the initiator might be misplaced. The iSCSI protocol permits for a novel TCP connection to be set up inside the session,

and describes methods for the initiator and target to coordinate among each other to maintain smooth interaction. A new TCP connection might be selected to capture over an older TCP connection that appears to have become faulty. This level of recovery is known as Connection recovery. Dealing out commands that were initiated on the faulty TCP connection might be continued on the fresh TCP connection.

Fiber Channel needs dedicated cabling but iSCSI can be used over long distances using current network setups. In an experimental study done by G. Khanna, R. Hurst, and M. brown, at a Cisco lab, iSCSI makes two hosts to confer and substitute SCSI commands making use of IP networks. iSCSI hosts are exposed and presented through the Cisco Fabric Manager at login. For the duration of login, the host is allocated an nWWN and a pWWN either dynamically or through manual configuration. When a host is allotted its relevant WWNs, it can be configured and monitored like it was attached in a straight line to the Fabric by means of a traditional Host Bus Adaptor.

There are two initiator modes available: *Static Initiator* and *Proxy Initiator [38]*

Static Initiator

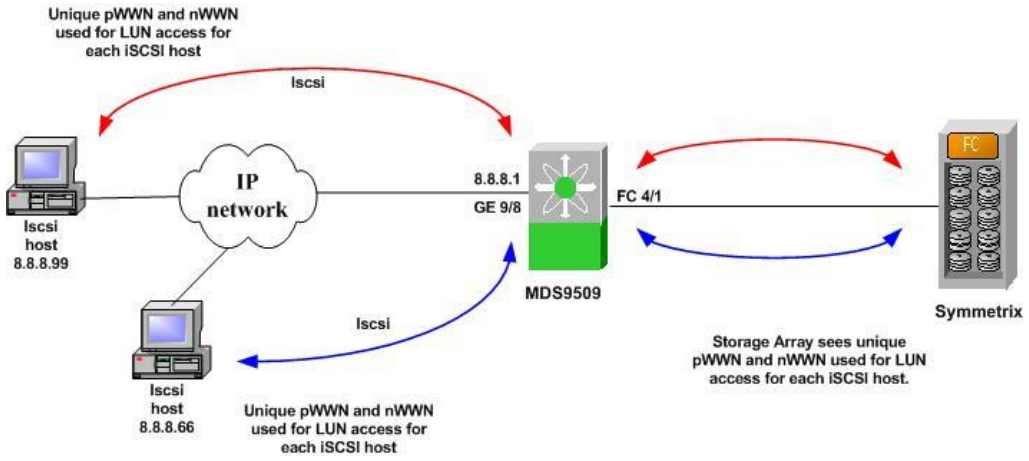


Figure 9. Static initiator setup for iSCSI host [38].

By means of iSCSI ‘static initiator’, every iSCSI host represents an exclusive pWWN and nWWNs. The WWNs might be assigned dynamically or configured manually.

Proxy Initiator

Another alternative is using ‘proxy initiator’. In this method, all iSCSI hosts that are registered into the port are signified on the Fiber Channel network as a single host, as shown in Figure 10.

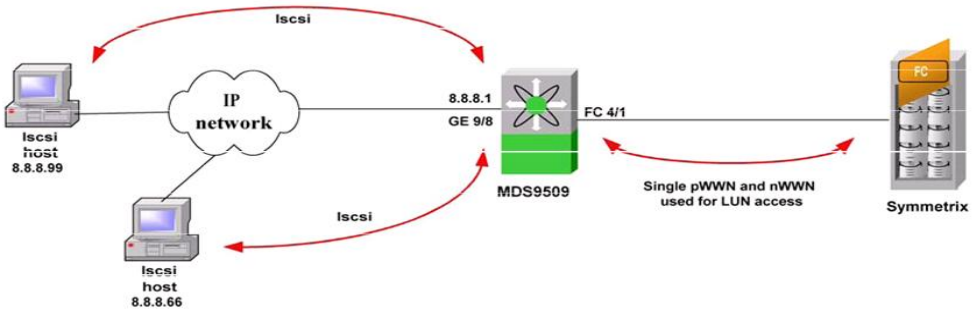


Figure 10. Proxy initiator setup for iSCSI hosts [38]

There are three major ways an iSCSI host system can determine targets:

- **Manual configuration**: The IP address, port, and iSCSI Name of the target are indicated by the Storage Area Network supervisor. Most iSCSI driver softwares do not support this technique.
- **iSCSI SendTargets command**: If IP address of the target is present; the SendTargets instruction permits the initiator to recover the iSCSI information. The target’s IP address might be manually configured or discovered through a management node. This technique should be supported by iSCSI initiators.

- **"Zero-configuration" discovery methods**: These techniques comprise of the Service Location Protocol (SLPv2) and/or the Internet Storage Name Service (iSNS). The iSCSI conditions do not need the host to hold any of these two protocols.

Once a source is revealed, the initiator and target undertake a four phase process to set up a Normal Operational Session that is complete connectivity:

1. Initial Login phase
2. Security Authentication Phase (optional)
3. The Login Operational Negotiation Phase
4. The Full-Feature Phase

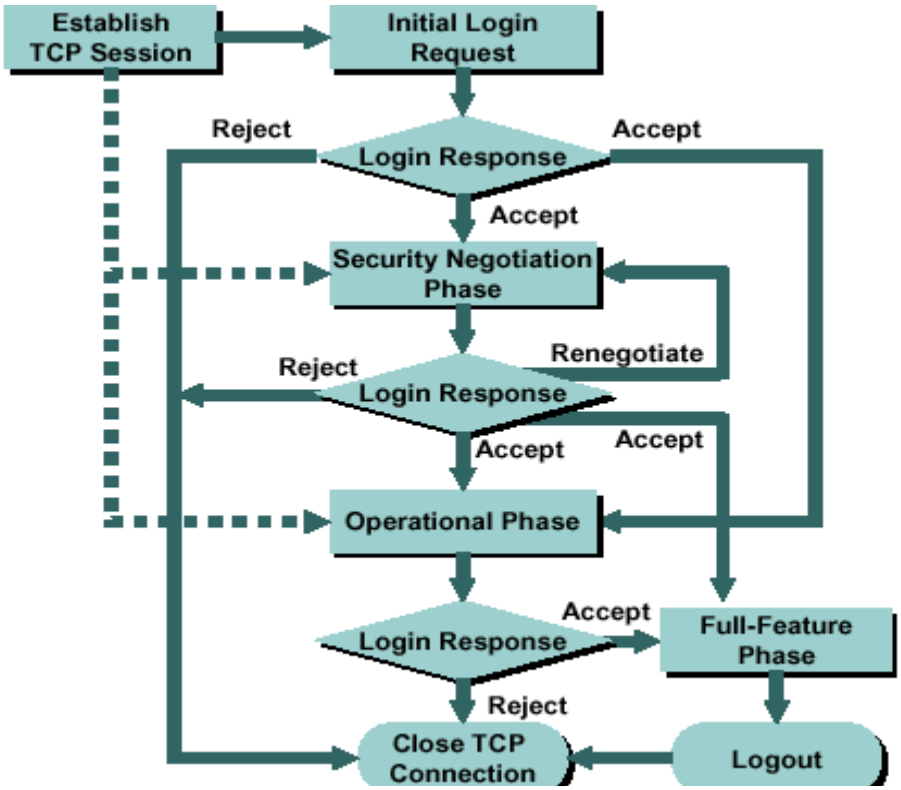


Figure 11. Phases sequence from the organization of a TCP session between host and MDS to the Full-Feature Phase [38]

The iSCSI protocol has become known as a transportation medium for moving SCSI block-level access protocol over TCP/IP. It facilitates the block-level access of a client to remotely stored data above present IP infrastructure. Though, the feat of the iSCSI supported remote storage system for portable devices crashed in wireless networks, because the protocol was initially developed for wired networks.

4.2 Strategy to improve iSCSI performance

To summarize SCSI protocol over IP necessitates considerable quantity of operating cost on traffic for SCSI commands transportation and handshaking above the Internet. Translating protocols at a switch consign unusual load to a previously overloaded switch and requires dedicated networking devices in a SAN. Another experiment showed, on a distinctive iSCSI execution, that about 58% of TCP/IP packets that were considered were less than 127 bytes in length, which implied that a huge amount of small size packets to transport SCSI commands. A cache method called *iCache* (iSCSI cache) was initiated intended to speedup existing iSCSI performance [25]. *iCache* transforms small demands into large ones in form of logs prior to writing information into remote storage through the network, and exploits the Log- structured file system to rapidly write information to log disks for caching data. iCache also enhances the consistency of the structure as both user data and meta data are cached on a log disk which is greatly consistent than RAM. Furthermore, *iCache* is absolutely apparent to the OS and it does not necessitate any modifications to the OS nor does it require admission to the kernel source code. Moreover, *iCache* also concentrates SCSI commands and

handshaking processes to decrease redundant traffic over the Internet. In this fashion, it operates as a storage filter to dispose off a portion of the data that would travel across the Internet, dropping the restricted access forced by the restricted Internet bandwidth and growing storage data rate.

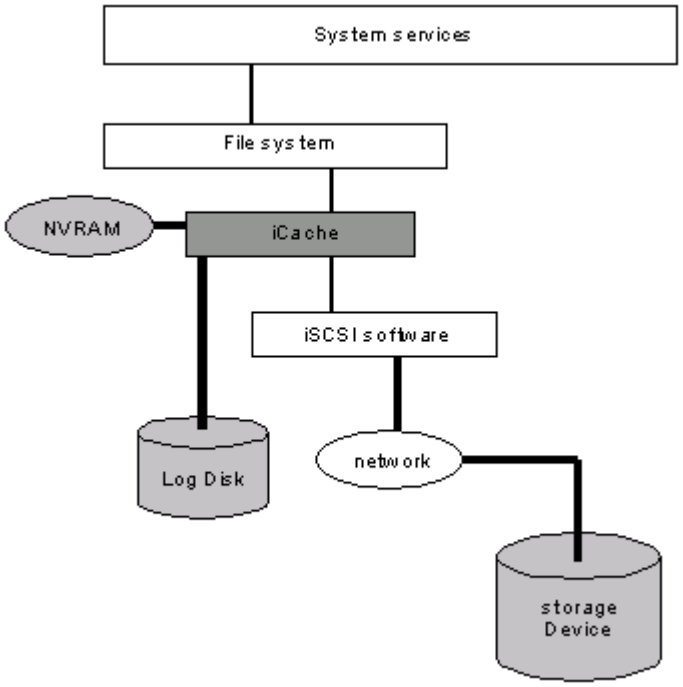


Figure 12. iCache Architecture[25]

By means of a two-level hierarchical cache comprising of a little amount of NVRAM and a log disk, *iCache* stabilizes the storage data traffic involving SCSI and IP.

4.3 Enhancing iSCSI performance in Wireless Application

iSCSI protocols are not much conventional in itinerant devices to offer a network-wide extensive storage subsystem. A local file system over the storage subsystem or

network file systems customized to mobile or dispersed setting must be established [26] so as to present high stage file system examination in mobile devices. Given that mobile devices such as laptop computers and smart phones are frequently used for individual use, sustaining local file systems consistently in movable devices is gaining importance. Conversely, the performance of wireless system is responsive to environmental features for instance distance and signal intervention. This formulates the file system examination unsteady in a mobile tool implementing iSCSI [31] as its fundamental storage protocol designed for a local file structure.

4.3.1 Adaptive Control of iSCSI protocol

An iSCSI initiator offers interface linking an iSCSI target and a SCSI subsystem which presents a storage room. To accomplish this, an iSCSI initiator employs SCSI tool generalization that is presented by a SCSI subsystem. Consequently, if the SCSI subsystem obtains a SCSI call from the upper layer, the SCSI request will be transferred to the iSCSI initiator. The iSCSI initiator then performs appropriate action to iSCSI target based on the expected request. Wireless network based SCSI procedures have unsteady routines with common short term disconnection, and the characteristics of these procedures are not measured in existing execution of SCSI subsystems.

5 EXPERIMENT SET UP AND METHODOLOGY

5.1 Software and Hardware Tools, Development Kits Used

1 Laptop with Windows XP Operating System (Host/Server)

1 MDS (Cisco MDS 9509)

1 Switch (Cisco 3750G)

1 Access Point (Cisco Aironet)

1 Storage device (JBOD)

Iometer software tool is being used for carrying out the tests and getting results.

Cisco MDS 9509 Multilayer Fabric Switch:

The Cisco MDS 9509 preserves Cisco Storage Area Network OS Software and is set to perform with smallest amount of configuration [38]. It maintains enterprise class attributes such as Virtual Storage Area Networks (VSANs), non-disruptive code upgrade, Port Channels, and security (authentication, authorization, and accounting). The important features consist of exceptional flexibility and scalability, exceptionally available platform for mission-critical procedures, comprehensive security configuration, and fundamental storage organization and complex diagnostics.

Cisco Catalyst 3750G-24PS-E:

The Cisco Catalyst 3750G-24PS switch is a division of the Cisco Catalyst 3750 Series, that increases LAN operational efficiency by integrating simplicity of utilization and the highest resiliency accessible for switches [38]. This product indicates the next conception in desktop switches, and has Cisco StackWise expert characteristics, a 32-

Gbps stack interlink that allows consumers to create an integrated, exceptionally resilient switching configuration on a switch at a time.

Using a 32-Gbps stack interlink, Cisco StackWise expertise is proposed to respond to network variations of all sort while keeping steady, high network performance. Cisco StackWise technology unites up to nine Catalyst 3750 switches in a single coherent unit through unique stack integrated wires. The stack performs as a single unit managed by a master switch selected from one of the element switches. Its extremely developed failover techniques generate the utmost levels of stackable resiliency for hardware and software consistency.

The physical features of Cisco MDS 9509 hold mix of 2, 4, and 8-Gbps Fiber Channel switching modules in the single chassis with 32-Gbps, high-speed stacking bus.

Cisco Aironet Access Point:

Cisco Aironet Access Points offer secure, suitable, and consistent wireless connectivity with exceptional performance and range [38]. The access points maintain vital 802.11a/b/g connectivity for indoor and outdoor setting. These access points work with Wireless LAN controller to focus to the protection, process, management, and manage concern in general enterprise wireless LANs. Cisco Aironet Access Points showcase complete potential which include:

- Wireless voice over IP
- Guest access
- Wireless intrusion detection and intrusion prevention
- Scalable Layer 3 roaming

Iometer:

Iometer is an I/O scheme measurement and organization way for isolated and clustered systems [39]. It is a workload originator meaning that it inspects and records the performance of its I/O processes. It can be configured to stimulate the disk or network I/O load of any program or benchmark. Iometer can be utilized for

- Performance of disk and network controllers.
- Bandwidth and latency capabilities of buses.
- Network throughput to attached drives.
- Shared bus performance.
- System-level hard drive performance.
- System-level network performance

Iometer comprises of two programs, *Iometer* and *Dynamo*.

(1) *Iometer* is the scheming series. With Iometer's GUI, we compose the workload, place operating restraints, and start and stop study. Iometer notifies Dynamo what to perform, collects the resulting data, and evaluates the results in output files. It is generally run on the server machine.

(2) *Dynamo* is the workload generator. It does not have a user interface. At Iometer's authority, Dynamo performs I/O processes and traces performance information, and then returns the information to Iometer. There might be added copies of Dynamo working at a time. Generally one copy runs on the server machine and one additional copy runs on each client machine. Each copy of Dynamo is called a *manager* and each thread within a copy of Dynamo is called a *worker*.

The Iometer user interface has the following major components:

Toolbar: Executes regular operations such as starting and stopping tests.

Status bar: Displays currently running test among a test series.

Topology panel: Shows available managers (Dynamos) and workers (threads).

Tabbed panels: Several different tabs controlling the parameters of the test:

Disk Targets tab: Specifies the disks used by each disk worker.

Network Targets tab: Specifies the network interfaces used by each network worker.

Access Specifications tab: Specifies the type of I/O operations each worker performs to its targets.

Results Display tab: Displays performance data during the test.

Test Setup tab: Specifies the tests to be performed in a test series.

5.2 Test Setup

iSCSI SAN gears are mostly similar to FC SAN components. These components are:

iSCSI Client/Host

The iSCSI client or host (also identified as the iSCSI initiator) is a structure, for instance a server (a Microsoft PC in this case), which connects to an IP network and commences requests and accepts responses from an iSCSI target. Each iSCSI host is recognized by an exclusive iSCSI Qualified Name (IQN), similar to a Fiber Channel World Wide Name (WWN). To convey block (SCSI) commands above the IP network, an iSCSI driver should be established on the iSCSI host. A Gigabit Ethernet adapter is used connect to the iSCSI target. Similar to the standard 10/100 adapters, the majority of Gigabit adapters use Category 5 or Category 6E cabling that is previously prepared. Each port on the adapter is identified by a unique IP address.

iSCSI Target

An iSCSI target is a mechanism that receives iSCSI commands (MDS in this experiment). The device could be an end node, such as a storage device (JBOD), or it can be an intermediary device, such as a bridge between IP and Fiber Channel devices. Each iSCSI target is recognized by a unique IQN, and each port on the storage array controller is identified by one or more IP addresses

A Microsoft iSCSI initiator is being used to manage the host and target.

Node Names: The MS iSCSI initiator service strictly pursues the rules identified for iSCSI node names [40]. The rules are functional for both the initiator's node name and some target node names revealed. A number of the important rules are:

- Node names are encoded in the UTF8 character set. The initiator service does not support UCS-4 characters.
- Node names are 223 bytes or less
- Node names may contain alphabetic characters (a - z), numbers (0 to 9) and three special characters: '.', '-', and ':'.
- Uppercase characters are always mapped to lowercase.

The MS iSCSI initiator service generates an initiator node name dynamically based on the computer name if it is not set. If the computer name has any invalid characters then those invalid characters are mapped to '-'.If the MS iSCSI software initiator determines a target with an invalid node name it will ignore that target and in some cases all targets discovered with it

Initiator Instance

The MS iSCSI software initiator method cumulates the software initiator and iSCSI HBAs into a common iSCSI node. During this, the service should keep trail of the individual iSCSI HBA and software initiators as a number of the APIs allow operations to take place on only one of those HBAs. For instance the LoginIScsiTarget API has a constraint specifying which HBA or software initiator to use. The service and APIs refer to each HBA or software initiator as an Initiator Instance Performance of iSCSI over Wired Network.

The iSCSI initiator tool is installed on the Microsoft Windows XP notebook as host.

In order to services on volumes created on Storage from iSCSI disks following steps were followed:

1) All of the targets of the machine were logged in. Made them persistent logins by using the iSCSICLI command PersistentLoginTarget. iSCSICLI is a command line tool for scripting and revealing all functionality obtainable by the Microsoft iSCSI initiator service.

2) Configured all volumes on top of the disks using Disk Administrator.

5.2.1. Experiment 1: **Wired Network**

The first part of the experiment is to view the performance of iSCSI protocol over a wired network using Fiber channel. The wired host is made to run the Iometer tool to start the I/O operations.

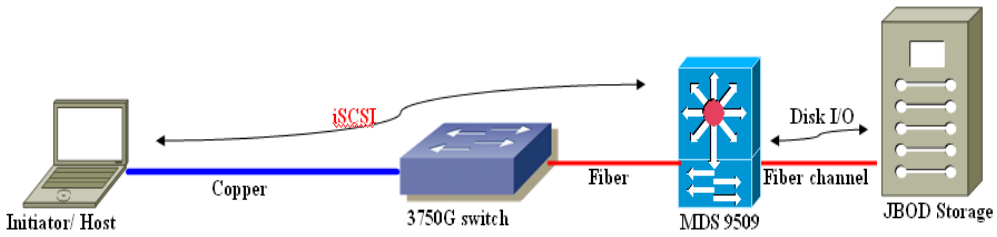


Figure 13. iSCSI over Wired Network

IP Addresses assigned are:

Device	IP
Host Laptop	15.1.1.3
3750 Switch	15.1.1.1
MDS	15.1.1.2

Table 1. IP Addressing

The graphical interface is shown in following figures with configuration fields entered.

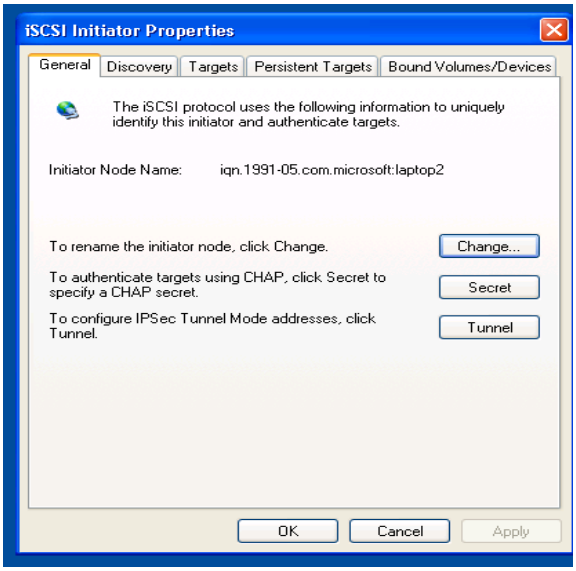


Figure 14. iSCSI Properties Applet

After running the iSCSI initiator on the Discovery Tab using the IP address or DNS name and Port number for the Target Portal is added as shown in Figure 15.

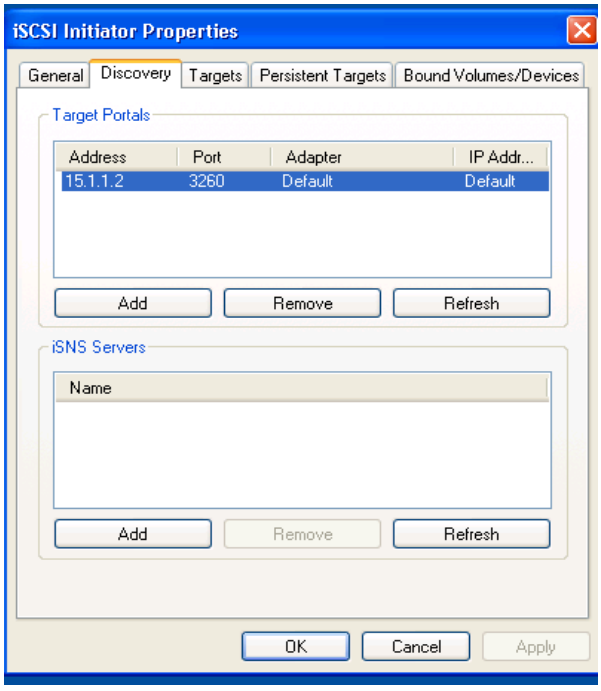


Figure 15. iSCSI Initiator - Discovery Tab

A target is selected out of available ones under the Targets tab to log on and a Persistent target is then displayed as shown in Figure 16.

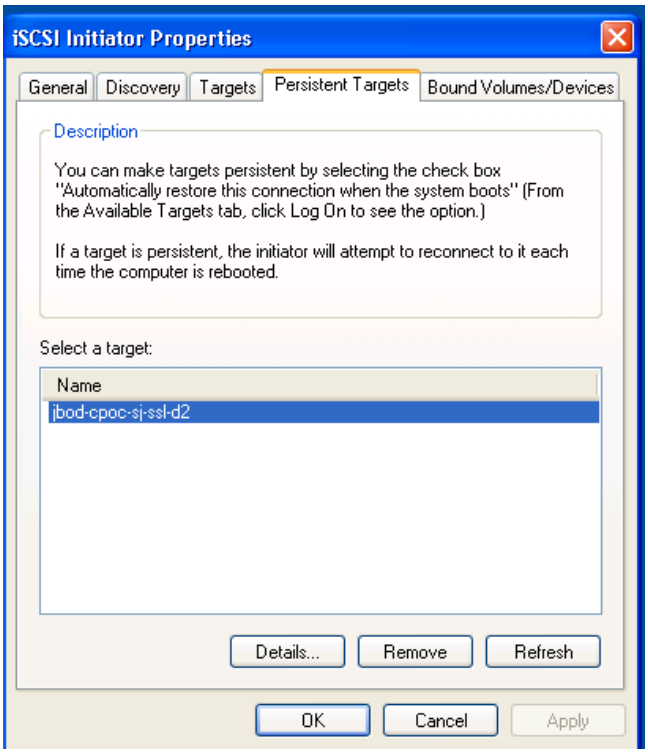


Figure 16. iSCSI Initiator Properties – Persistent Target

The MDS is configured to have iSCSI run through from the initiator to the target. The configuration is as follows:

```
iscsi enable
iscsi interface vsan-membership
vsan database
  vsan 3 interface iscsi9/1
ip default-gateway 30.3.155.1
ip routing
switchname MDS9509-SSL-D5-1
iscsi enable module 9
iscsi virtual-target name jbod-cpoc-sj-ssl-d2
  pWWN 21:00:00:04:cf:92:86:18
  advertise interface GigabitEthernet9/1
  initiator ip address 15.1.1.3 permit
iscsi initiator ip-address 15.1.1.3
```

```
  static pWWN 24:01:00:0d:ec:01:b6:42
  vsan 3
zone name rahul_zone1 vsan 3
    member pwwn 24:01:00:0d:ec:01:b6:42
    member pwwn 21:00:00:04:cf:92:86:18
    member pwwn 21:00:00:04:cf:92:8f:c4
zoneset name rahul vsan 3
    member rahul_zone1
zoneset activate name rahul vsan 3
interface fc1/16
  no shutdown
interface GigabitEthernet9/1
  ip address 15.1.1.2 255.255.255.0
  iscsi authentication none
  no shutdown
interface iscsi9/1
  no shutdown
  switchport initiator id ip-address
interface mgmt0
  switchport speed 100
  ip address 30.3.155.13 255.255.255.0
```

The Switch between the Fiber channel and the host is configured to let the iSCSI connectivity from host to target.

```
interface GigabitEthernet1/0/1
 switchport access vlan 15
 switchport mode access
interface GigabitEthernet1/0/3
 switchport access vlan 15
 switchport mode access
interface Vlan15
 ip address 15.1.1.1 255.255.255.0
```

The storage device (JBOD) is connected to the MDS and configured to carry out I/O read and write operations. A partition of the Disk is created and named for the wired test run.

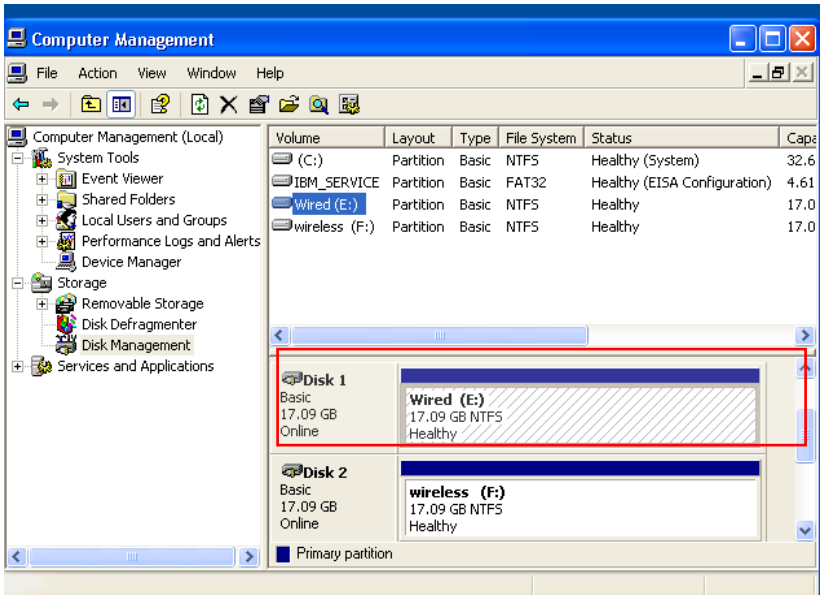


Figure 17. Disk Fragment of JBOD

At the end Iometer tool is installed on the host machine to carry out the actual test run. The number of outstanding I/O requests will affect performance. If you only have one outstanding I/O, it means that the next I/O sent will have to wait until the first I/O is completed. When testing for performance, the storage array will determine how many outstanding I/O requests will give the best performance.

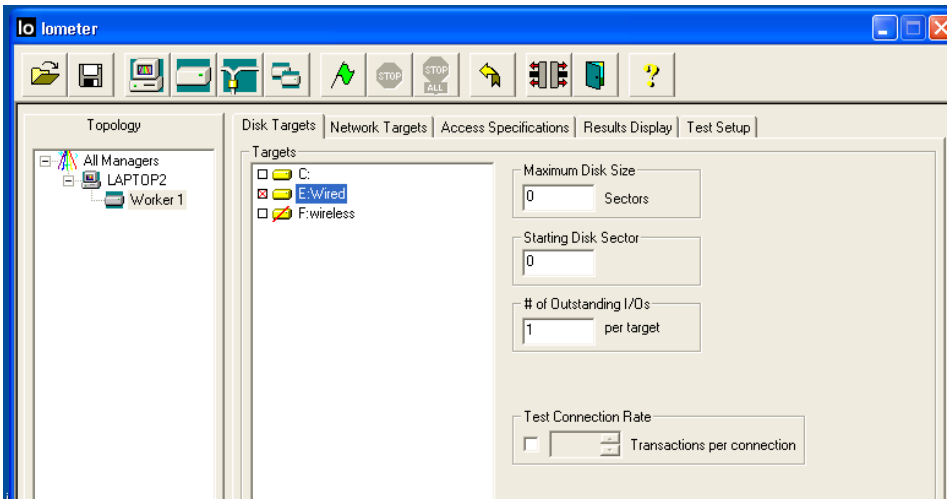


Figure 18. Iometer showing Disk Targets

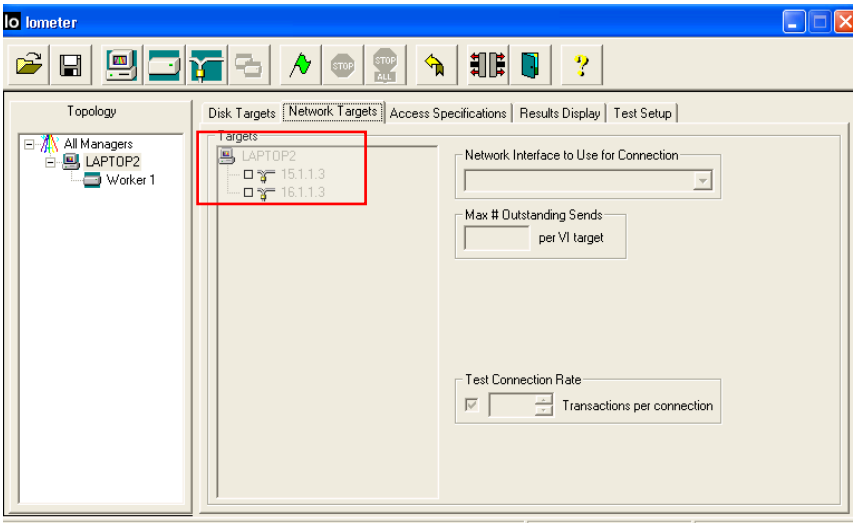


Figure 19. Network target selection

The next step is to configure the I/O profile, or the I/O workload that is to be used to simulate an application. Several parameters can be configured, including I/O size, reads vs. writes, I/O randomness, etc. Because this is a test of iSCSI and not the disk device itself, it is important to configure these parameters to maximize the performance of the disk array.

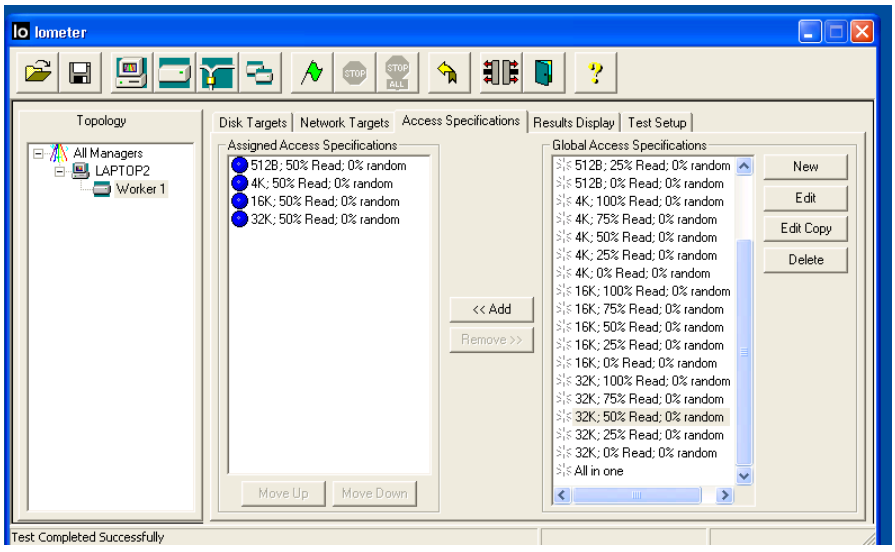


Figure 20. Access Specification selection

The access specification is required to see the performance of iSCSI over a variety of I/O operations (reads/writes) with different block sizes, and therefore it is important not to make the disk array itself a bottleneck.

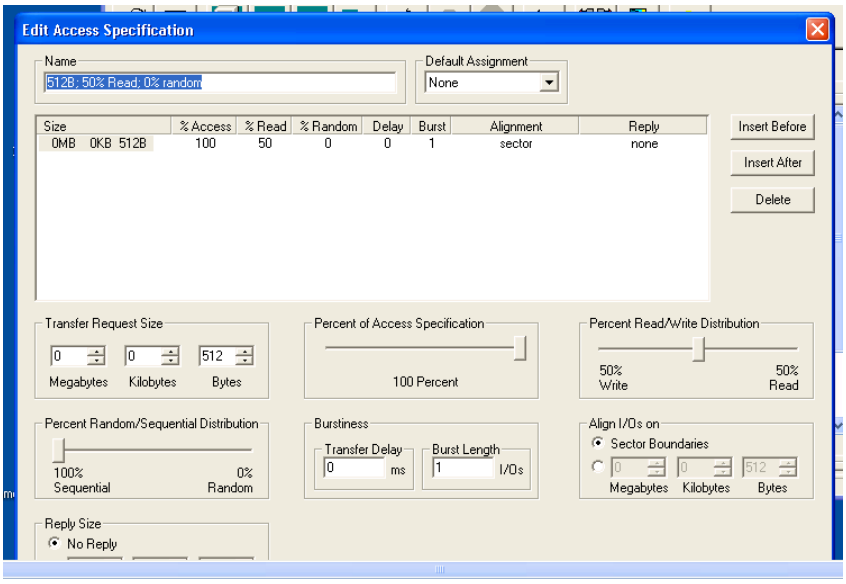


Figure 21. Access Specification details

Performance tests are conducted for different access specifications for the wired setup.

5.2.2. Experiment 2: **Wireless network**

The second experiment involves implementation of iSCSI protocol over wireless network using an access point. The access point is used to connect the host with iSCSI protocol on a wireless link and then wired connected to the Switch and then to the MDS through Fiber channel.

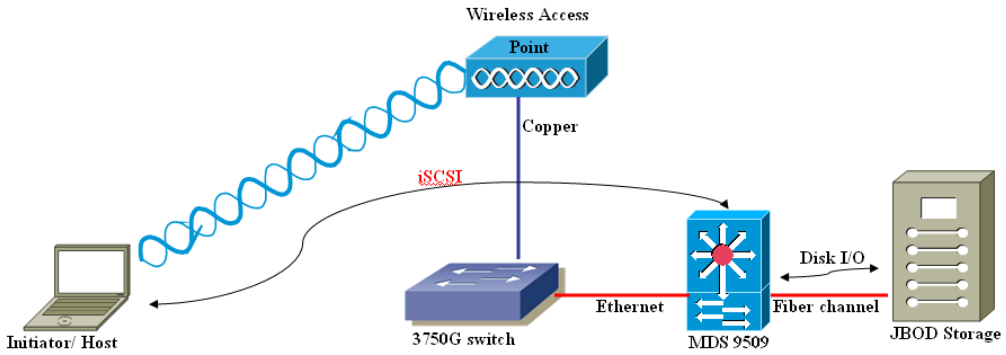


Figure 22. iSCSI over Wireless test

IP Addresses assigned are:

Device	IP
Access Point	16.1.1.1
Host Laptop	16.1.1.3
3750 Switch	16.1.1.100
MDS	16.1.1.2

Table 2. IP addresses used for the Wireless test setup

The Access point is connected to the setup for wireless connectivity to the host. Access point is then configured to connect to the host on wireless link. A wired link is made connecting the access point to the switch.

```
ip dhcp excluded-address 16.1.1.1
ip dhcp pool Rahultest
   network 16.1.1.0 255.255.255.0
   default-router 16.1.1.1 255.255.255.0
no aaa new-model
dot11 ssid Rahul
   authentication open
   guest-mode
   infrastructure-ssid
dot11 arp-cache
power inline negotiation prestandard source
interface FastEthernet0
 no ip address
 no ip route-cache
 duplex auto
 speed auto
```

```
 bridge-group 1
 no bridge-group 1 source-learning
 bridge-group 1 spanning-disabled
 hold-queue 160 in
interface BVI1
 ip address 16.1.1.1 255.255.255.0
 no ip route-cache
ip http server
no ip http secure-server
interface Dot11Radio1
 no ip address
 no ip route-cache
```

Figure 23 shows the successful connectivity between the iSCSI host and the wireless access point.

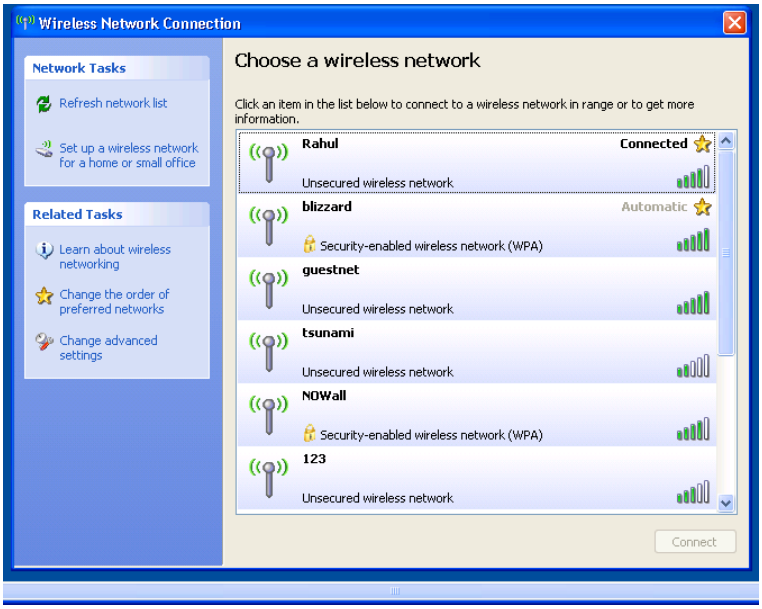


Figure 23. The host connected to Access point SSID Rahul

The iSCSI initiator is setup on the host for the wireless network.

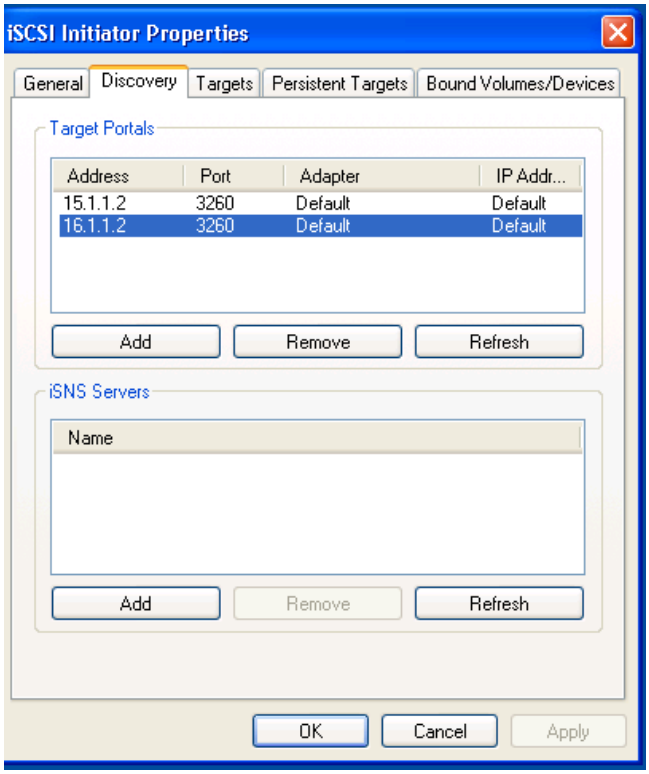


Figure 24. iSCSI with target discovery for wireless test

The target storage IP is fed in and target is searches and the assigned one is selected as in Figure 24.

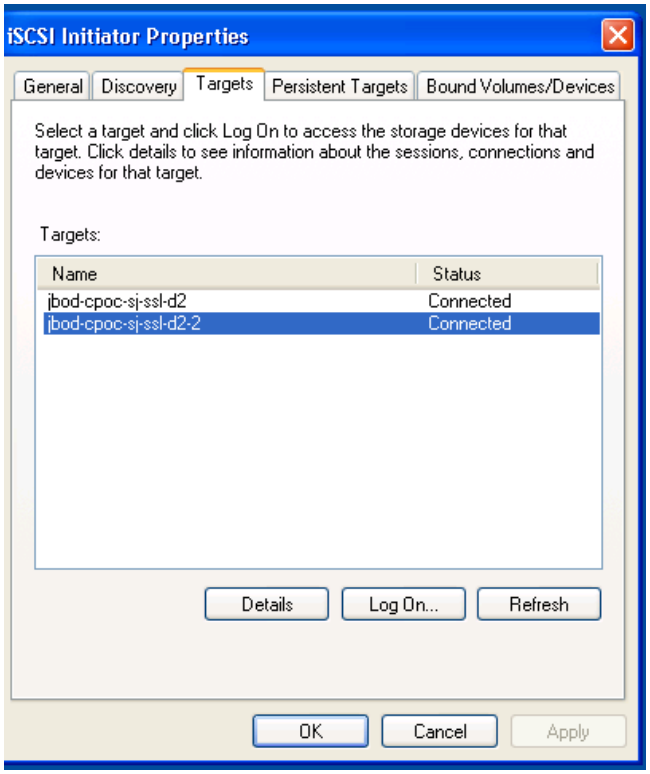


Figure 25. JBOD storage disk selected

The MDS is configured to have iSCSI run through from the initiator to the target for wireless setup. The configuration changes done are:

```
vsan 3 interface iscsi9/2
iscsi virtual-target name jbod-cpoc-sj-ssl-d2-2
  pWWN 21:00:00:04:cf:92:8f:c4
  advertise interface GigabitEthernet9/2
  initiator ip address 16.1.1.3 permit
iscsi initiator ip-address 16.1.1.3
  static pWWN 24:02:00:0d:ec:01:b6:42
  vsan 3
zone name rahul_zone2 vsan 3
    member pwwn 21:00:00:04:cf:92:8f:c4
    member pwwn 24:02:00:0d:ec:01:b6:42
interface GigabitEthernet9/2
  ip address 16.1.1.2 255.255.255.0
  iscsi authentication none
  no shutdown
interface iscsi9/2
  no shutdown
  switchport initiator id ip-address
```

The Switch between the Fiber channel and the host is configured to let the iSCSI connectivity from host to target.

```
interface GigabitEthernet1/0/2
 switchport access vlan 16
 switchport mode access
interface GigabitEthernet1/0/27
 switchport access vlan 16
 switchport mode access
interface Vlan16
 ip address 16.1.1.100 255.255.255.0
```

Next the storage device (JBOD) is connected to the MDS and configured to carry out I/O read and write operations. A partition of the Disk is created and named for the wired test run.

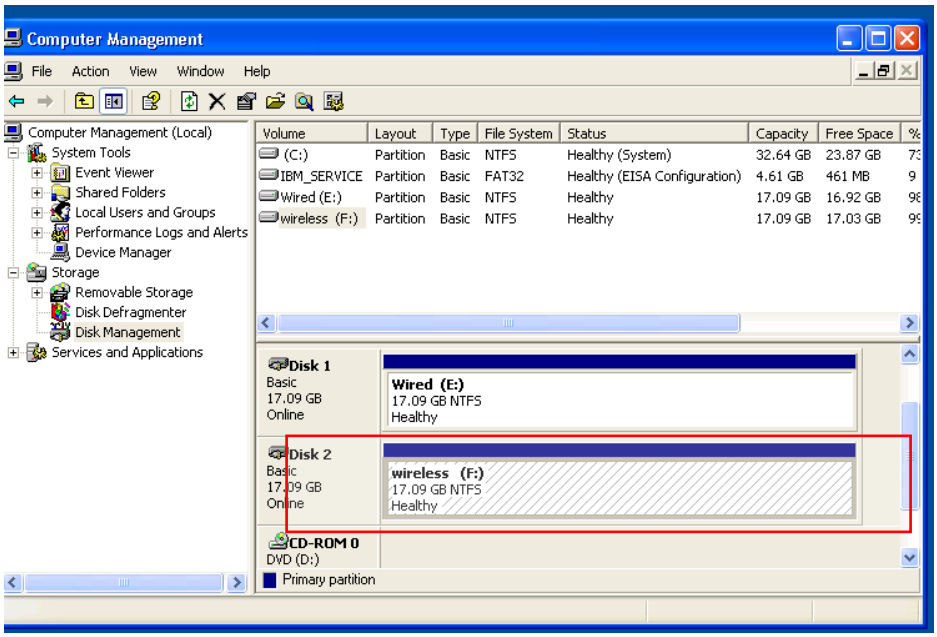


Figure 26. Disk selection for the test.

The Iometer is then setup to run test over the wireless network.

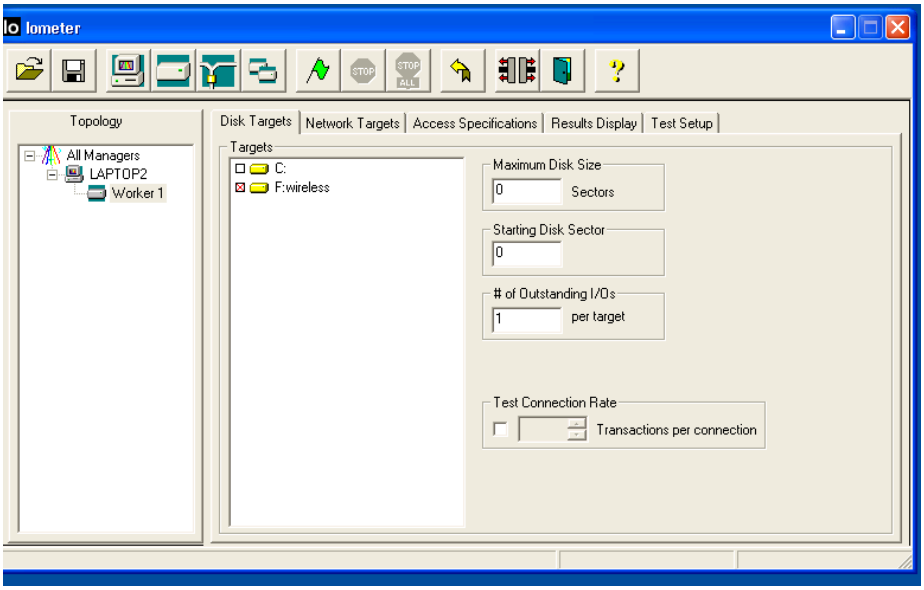


Figure 27. Target selection for the test

The access specifications are kept same for consistency in comparison of the results.

5.3 Results

The two tests are conducted in the Lab environment and the following results were obtained. The Iometer showed the running result of various performance features such as Total I/O per Second, Total Mbps, Average I/O Response Time, Maximum I/O response Time, Percent CPU Utilization and total Error Count.

5.3.1 **Wired test results:**

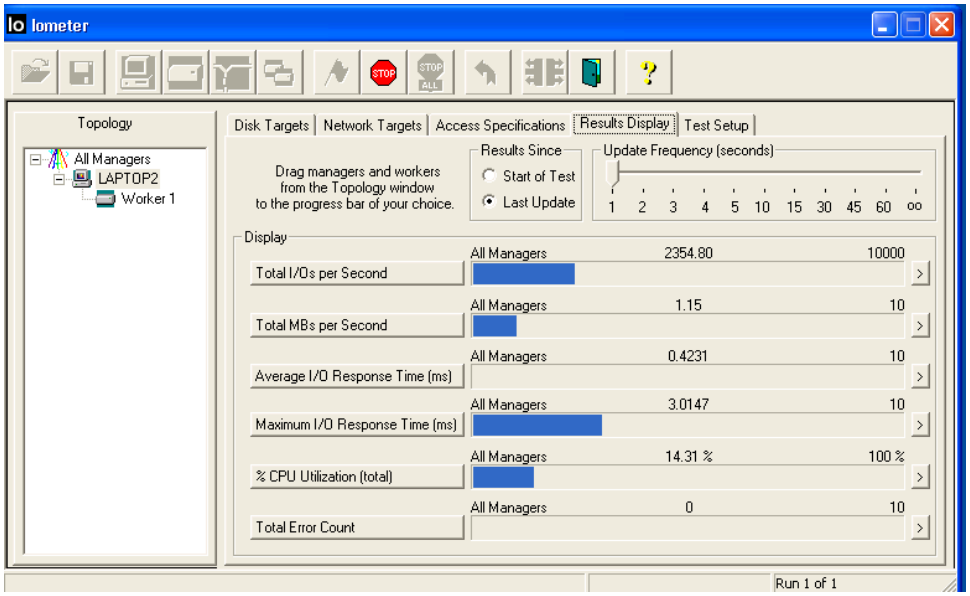


Figure 28. Iometer results for 512 data bytes

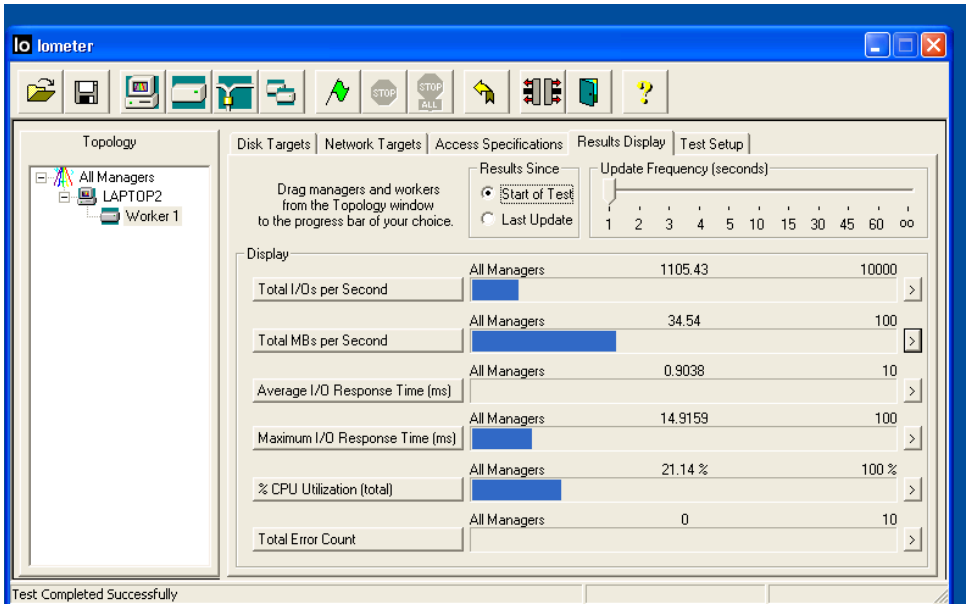


Figure 29. Iometer results for 32k data bytes

The figures 28 and 29 show the results after successful run of the iSCSI protocol over the wired setup for block sizes of 512b and 32kb data. The main performance features that need to be studied are Total I/Os per second, Total Mbps, Average Response Time (ms), and Maximum Response Time (ms). Result data is gathered and tables relating to different performance features are computed for 512, 4k, 16k and 32k bytes block sizes.

Number of I/Os per second					**Throughput in Mbps**			
Block Size	**Total I/Os**	**Read I/Os**	**Write I/Os**		**Block Size**	**Total Mbps**	**Read Mbps**	**Write Mbps**
512b	2118.5	1056.38	1062.12		**512b**	0.999834	0.496356	0.503478
4kb	2047.66	1016.54	1031.12		**4kb**	0.82754	0.41265	0.41489
16kb	1497.51	749.607	747.9		**16kb**	0.333985	0.157126	0.168593
32kb	1113	558.973	554.022		**32kb**	0.247811	0.117468	0.123132

Average Response Time					**Maximum Response Time**			
Block Size	**Avg. RT**	**Avg. RRT**	**Avg. WRT**		**Block Size**	**Max RT**	**Max RRT**	**Max WRT**
512b	0.457562	0.500986	0.277158		512b	8.576585	8.576585	5.101789
4kb	0.471303	0.497156	0.34613		4kb	6.844189	6.844189	4.432476
16kb	0.667038	0.648373	0.485288		16kb	6.740603	6.740603	4.237818
32kb	0.897677	0.910881	0.682567		32kb	6.503143	6.503143	4.936924

Table 3. Test results for I/Os over wired network

5.3.2 **Wireless Test Results**

The Iometer showed the running result of various features such as Total I/O per Second, Total Mbps, Average I/O Response Time, Maximum I/O response Time, Percent

CPU Utilization and total Error Count over the wireless network. Tests are conducted for 512, 4k, 16k and 32k bytes of data over distances of 5feet and 20feet. The environmental factors were also considered such as walls, other electrical devices and presence of other access points while increasing the host-access point distance.

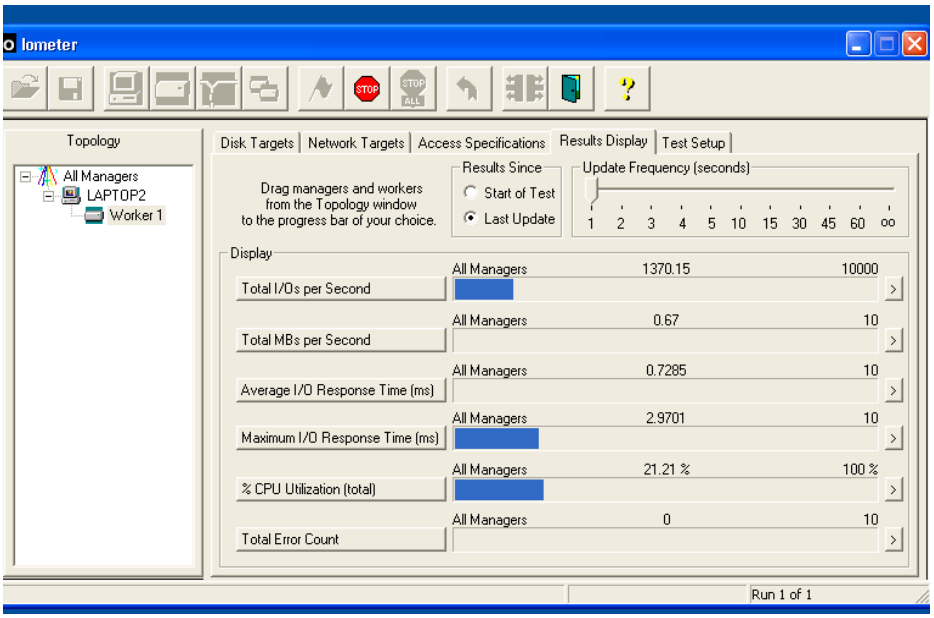


Figure 30. Iometer result for 512 bytes I/O with access point 5feet away.

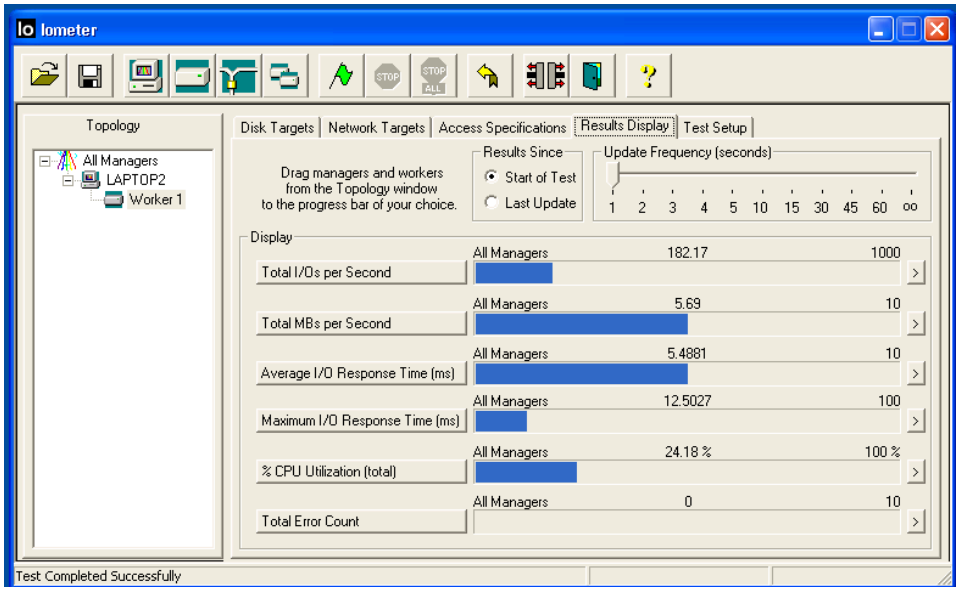


Figure 31. Iometer result for 32k data I/O with access point 5feet away

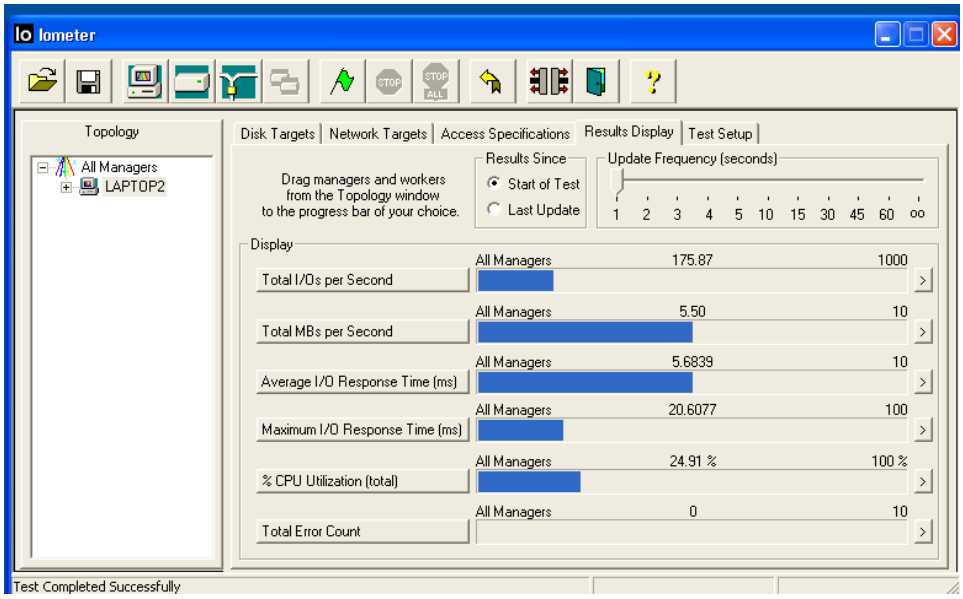


Figure 32. Iometer result for 32k data I/O with access point 20feet away

The figures 30, 31, and 32 show the results after successful run of the iSCSI protocol over the wireless setup for block sizes of 512b and 32kb data, but with host/initiator being at distances of 5ft. and 20ft. The main performance features that need to be studied are Total I/Os per second, Total Mbps, Average Response Time (ms), and Maximum Response Time (ms). Result data is gathered and tables relating to different performance features are computed for 512, 4k, 16k and 32k bytes block sizes.

Number of I/Os per second					**Throughput in Mbps**			
Block Size	**Total I/Os**	**Read I/Os**	**Write I/Os**		**Block Size**	**Total Mbps**	**Read Mbps**	**Write Mbps**
512b	1381.627	689.8424	691.7846		**512b**	0.569278	0.285619	0.283659
4kb	798.9058	399.7002	399.2057		**4kb**	0.509299	0.255947	0.253352
16kb	325.9514	163.8059	162.1455		**16kb**	0.312073	0.156133	0.15594
32kb	182.1688	91.39803	90.77079		**32kb**	0.067462	0.033684	0.033779

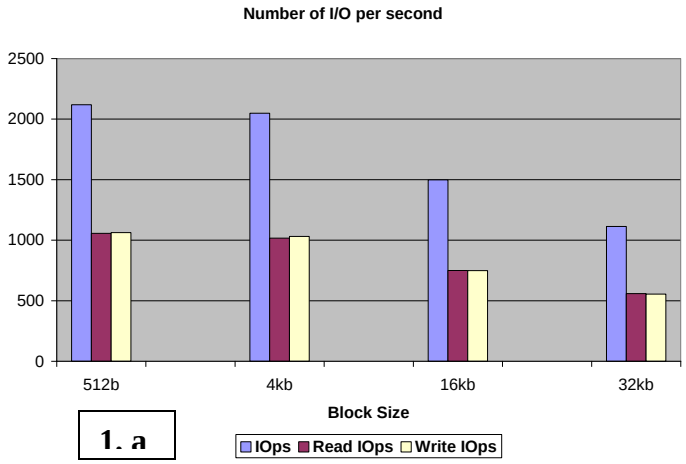
Average Response Time					**Maximum Response Time**			
Block Size	**Avg. RT**	**Avg. RRT**	**Avg. WRT**		**Block Size**	**Max RT**	**Max RRT**	**Max WRT**
512b	0.723025	0.835236	0.611129		**512b**	6.275461	6.275461	5.995195
4kb	1.250916	1.431274	1.070334		**4kb**	8.042927	8.042927	6.166938
16kb	2.93521	3.127707	2.742712		**16kb**	10.68786	10.68786	9.850245
32kb	5.236314	5.511906	4.960722		**32kb**	12.50269	12.50269	8.561676

Table 4. Test results with access point 5feet away from the host

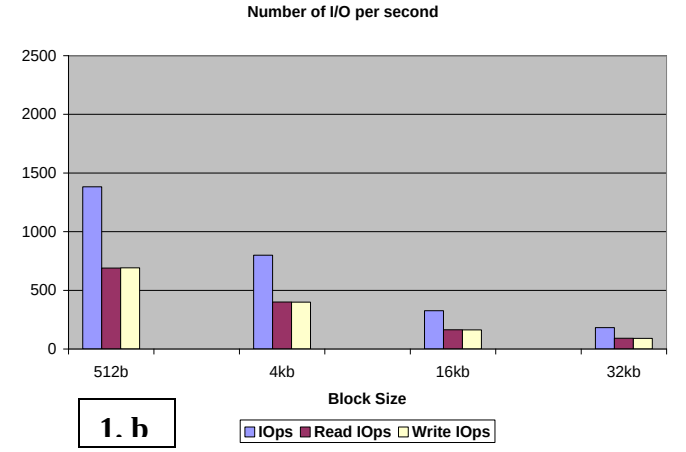
Number of I/Os per second					**Throughput in Mbps**			
Block Size	**Total I/Os**	**Read I/Os**	**Write I/Os**		**Block Size**	**Total Mbps**	**Read Mbps**	**Write Mbps**
512b	767.5042	388.2796	379.2245		**512b**	0.549587	0.2801	0.269486
4kb	196.2827	97.03134	99.25131		**4kb**	0.14595	0.071274	0.074675
16kb	175.8677	89.63209	86.2356		**16kb**	0.076673	0.037903	0.03877
32kb	93.40779	45.61551	47.79228		**32kb**	0.037476	0.018959	0.018517
Average Response Time					**Maximum Response Time**			
Block Size	**Avg. RT**	**Avg. RRT**	**Avg. WRT**		**Block Size**	**Max RT**	**Max RRT**	**Max WRT**
512b	1.196925	1.535198	0.858651		**512b**	5.454282	5.397371	5.245428
4kb	4.72783	5.555042	3.900617		**4kb**	7.759699	7.759699	4.60077
16kb	8.854154	10.29903	7.409276		**16kb**	9.79309	9.79309	9.934277
32kb	12.47591	14.09213	10.85969		**32kb**	20.06077	20.06077	18.05724

Table 5. Test results with access point 20feet away from the host

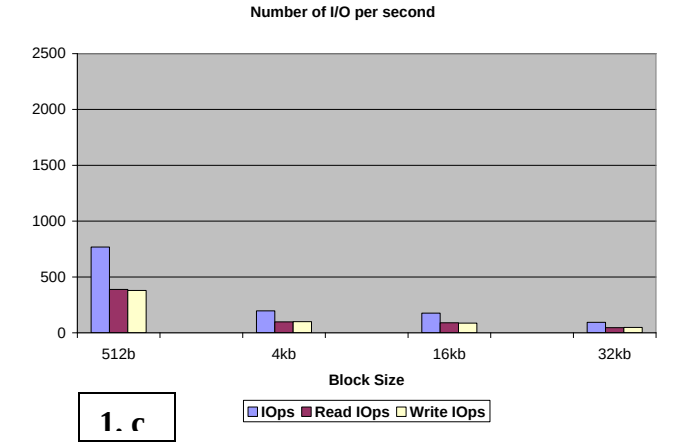
Graphs are then plotted for representing test results for various performance features of iSCSI gathered from Iometer.



1. a



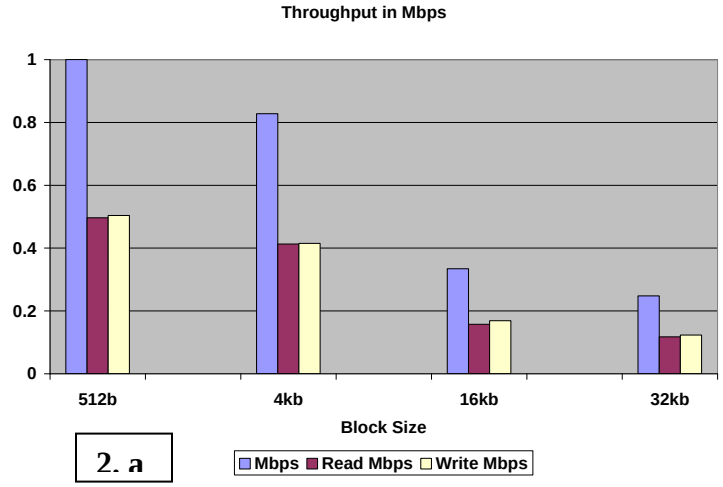
1. b



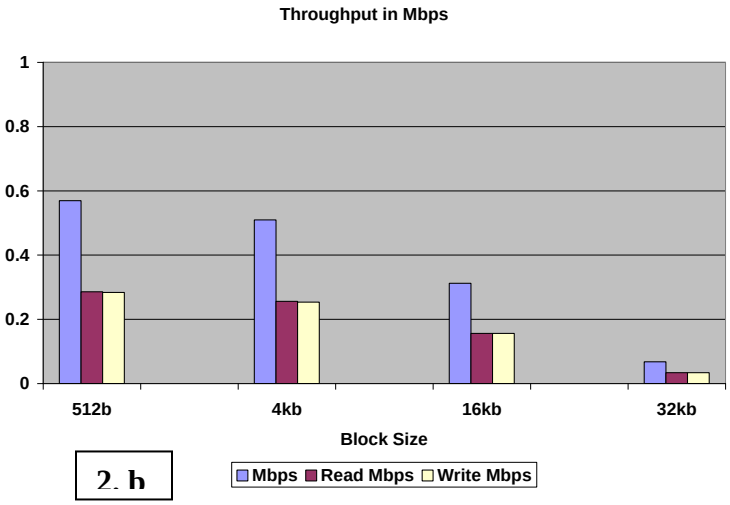
1. c

Graph 1. Performance comparison for Number of I/Os per second for iSCSI over wired (1.a) network and wireless with 5ft (1.b) and 20ft (1.c) host-access point distances

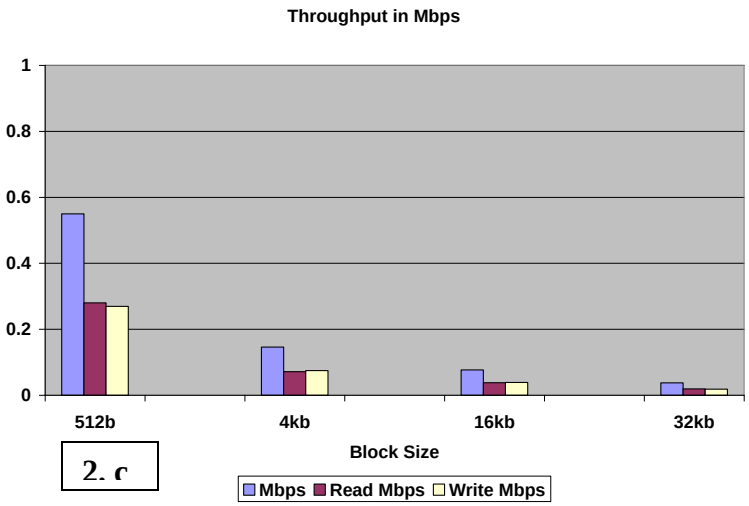
These graphs above confirm that iSCSI implementation on wired network and wireless network results in similar performance. The number of I/O operations per second tends to decrease with increase in the block size. Though there is a major decrease in the numbers of I/Os done for same block size data over wireless setup but iSCSI is able to transport small block size data over wireless is comparable. These results prove that the protocol finds it easier to transport small blocks of data with ease rather than big block size data over wireless medium.



2. a

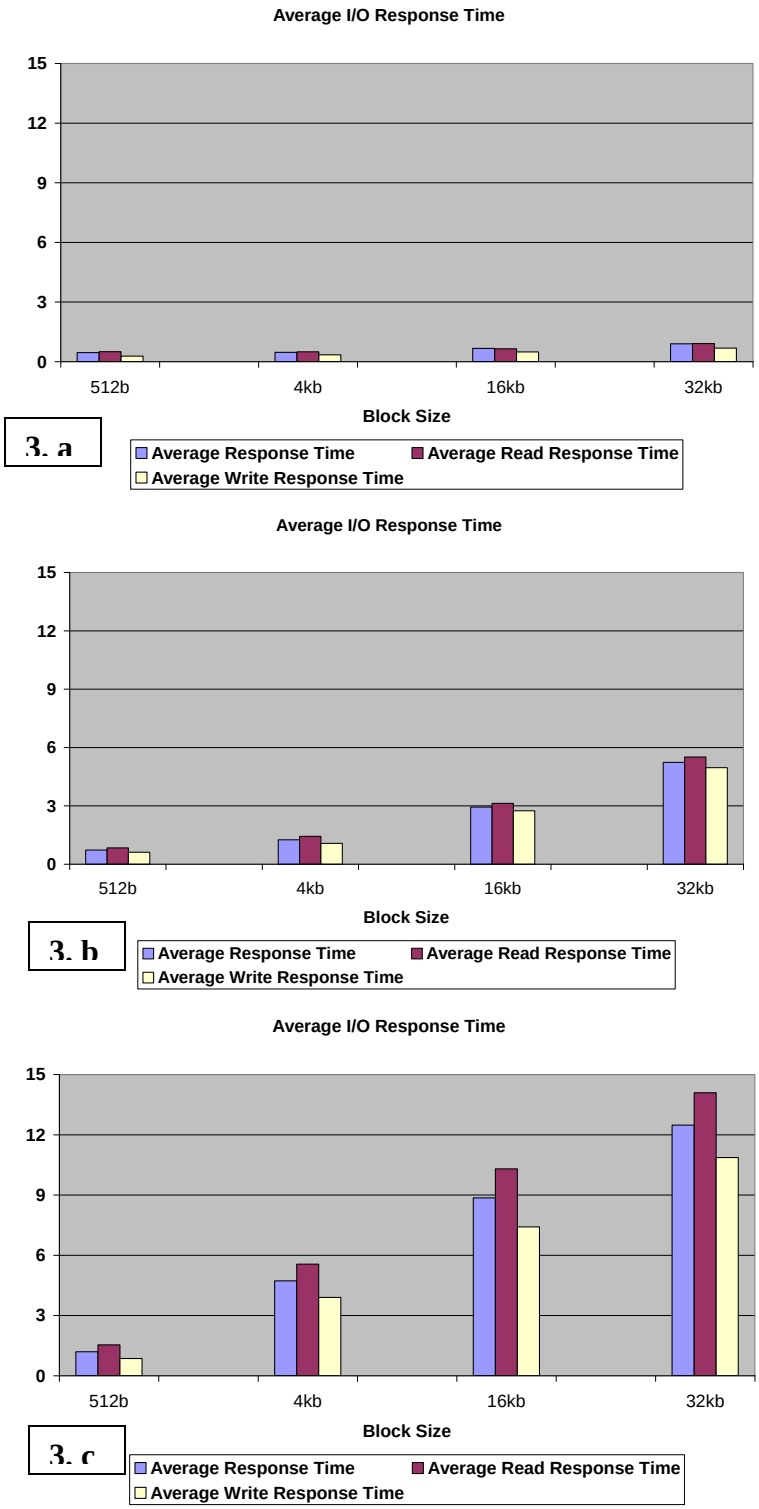


2. b



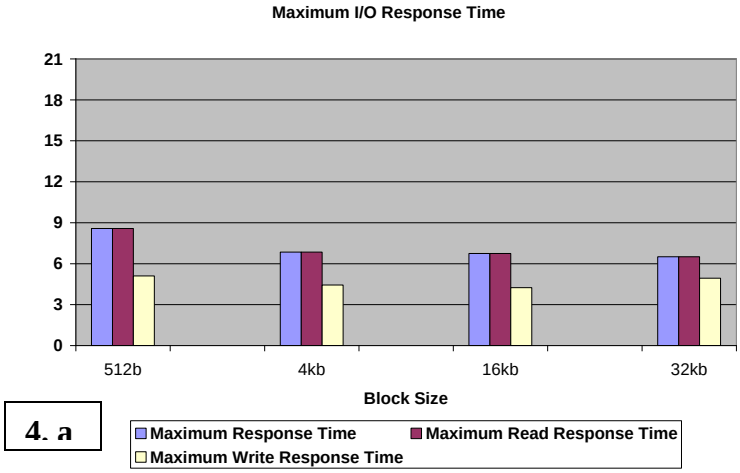
Graph 2. Performance comparison for Throughput in Mbps for iSCSI over wired network (2.a) and wireless with 5ft (2.b) and 20ft (2.c) host-access point distances

The results from Graph 2 represent the measure of throughput of iSCSI in Mbps. The curves show a trend of decrease in throughput with increase in block size over both wired and wireless network. Again the graphs display that the behavior of the protocol over wireless network is comparable to that over wired network even though the speed is reduced by greater margins for larger block sizes.

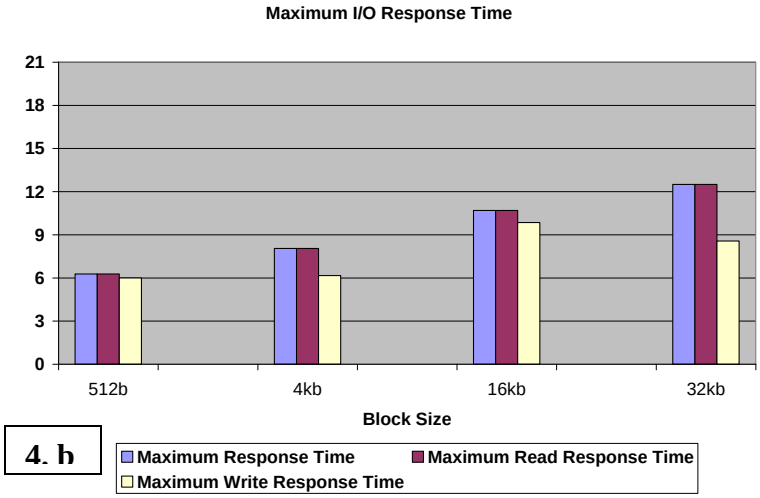


Graph 3. Performance comparison for Average I/O Response Time (ms) for iSCSI over wired network (3.a) and wireless with 5ft (3.b) and 20ft (3.c) host-access point distances

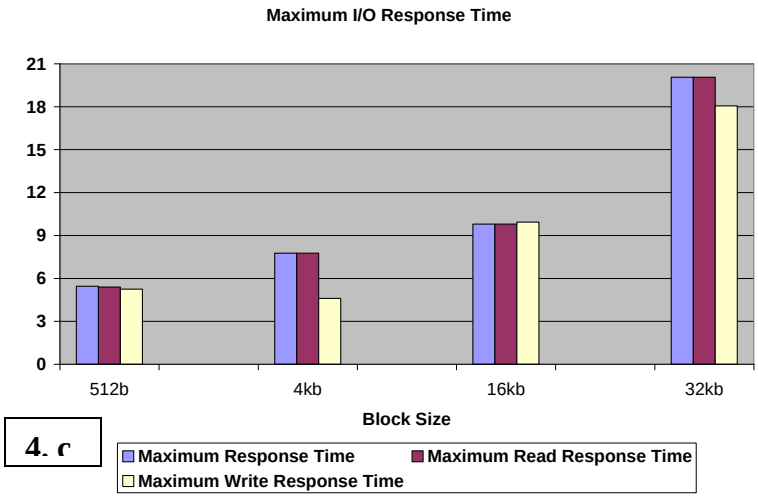
The graph 3 results show another similar performance of iSCSI in both wired and wireless networks. The curve for average I/O response time shows increase with increase in the data block size. This means that it takes more time to compute I/O operations for large chunk of data. While iSCSI shows marginal difference in the average response time for various block sizes over a wired connection, the difference in response time is more visible over wireless network with an increase in distance of host and the wireless access point. The implementation shows successful operation of iSCSI over wireless even though the I/O response time is greatly increased.



4. a



4. b



Graph 4. Performance comparison for Maximum I/O Response Time (ms) for iSCSI over wired network (4.a) and wireless with 5ft (4.b) and 20ft (4.c) host-access point distances

The final results of interest shown by graph 4 set is the study of maximum I/O response time for iSCSI over wired and wireless network. This performance feature shows opposite nature of curve in graphs plotted for wired and wireless implementation of iSCSI. There is a decreasing curve for wired run of iSCSI which means that the I/O response time decreases with increase in data block size. Whereas, the wireless network graphs for both scenarios show that the response time is increased drastically for larger chunk of data blocks.

Various tests were performed over different Access specifications of I/O for both Wired and Wireless test run for fixed distances. The tests showed that the performance of iSCSI was far better than what was expected over wireless network. The figures show that performance of iSCSI over wireless was comparable to the wired network even with increase in distance of host and the wireless point. There were certain specifications where iSCSI was unable to match the wired counterpart such as Maximum I/O response time. The total I/Os per second was almost half of what the protocol could achieve over

wired network. The tests were done to a maximum distance where the connectivity was persistent for the wireless with environmental factors in place.

The graphs show that the iSCSI was nearly successful over wireless as in wired setup environment for certain performance evaluation features. The I/Os per second plot showed the same trend of decreasing curve for increase in data block size even though the I/Os were reduced over wireless transfer. The average I/O response time graphs for wireless setup also showed an increase with increase in block size similar to wired transfer of iSCSI data.

But there were discrepancies for certain parameters such as Maximum I/O Response Time. The graphs indicate that the results were opposite to the trend in wired setting. While the maximum I/O response time showed a decrease with increase in data block size in wired environment, the wireless transfer showed an increase in maximum response time.

There are problems when large amount of data is transmitted over wireless network using iSCSI, for the given I/O specifications. The graphs showed the variation of various performance features as the distance of host and access point was increased. Another major drawback of iSCSI occurs when the host is mobile or when the distance of host /initiator is increased from the access point and the host is kept mobile instead of being fixed. But this situation hardly arises as most of the times host are stationary after some motion when they interact with the access point.

The iSCSI protocol worked on the wireless network with almost no errors and reduced packet drops comparable to a wired network run.

6. CONCLUSION

The experiments done above show that the iSCSI protocol is extremely useful for getting implemented for accessing storage over Storage Area Networks. The iSCSI protocol permits for a novel TCP connection to be set up inside the session, and describes methods for the initiator and target to coordinate among each other to maintain smooth interaction. Fiber Channel needs dedicated cabling but iSCSI can be used over long distances using current network setups. Even though with many drawbacks, the experiment was successful in displaying that iSCSI protocol can run over a wireless medium for SANs.

The iSCSI protocol is a transport for SCSI over TCP/IP. Other SCSI transports contain SCSI Serial and Fiber Channel Protocol (FCP). iSCSI permits storage to be accessed over a storage area network (SAN), granting shared admission to storage. A key benefit of iSCSI over FCP is that iSCSI can run above usual off-the-shelf network mechanism, such as Ethernet.

When running performance tests using Iometer from a single host, multiple factors affect the performance of the protocol. In most cases, the host is usually the bottleneck. If the host has a slow processor and insufficient memory, test performance are affected. The fact that iSCSI uses processing power from host CPU for the iSCSI stack could potentially affect performance as well. One can compensate for this by using multiple hosts to generate more iSCSI aggregate I/O or by using a single initiator with higher CPU and memory resources. The other major performance factor will be the block size of the I/O hitting the disks or tape. When determining the optimal block size for application performance, keep in mind that the block size range mentioned in this paper

must be used only as a guideline. If customers have specific applications that they wish to use Iometer to run the simulation tests on, modifications must be made in block size and other I/O parameters to make the I/O test closely resemble the application I/O. Finally, the number of disks and the number of outstanding I/Os will also affect the performance. Generally speaking, the more disks and the higher number of outstanding I/Os that are configured, the higher the aggregate I/O performance

Another observation revealed that while using storage procedures over a network, one has to take care of the capability of an initiator to notice the procedures it may use. One method requires an administrator to arrange the initiator statically, offering the initiator with a record of the names and addresses of the iSCSI means to which the initiator may connect. If further iSCSI devices are added later to the network, the statically constituted initiator would not be capable of accessing the latest devices without being reconfigured.

The performance tests of iSCSI over wired and wireless network showed that the implementation of the protocol can be successfully carried out over a wireless network for the I/O access of storage data. The iSCSI protocol which works on TCP/IP stack, is another way of solving a remote access of storage data over a reliable link. With more advancement in bandwidth it might provide great solutions of removing the costly Fiber channel links that run over long distance with a compromise on the speed and time. The protocol needs more research in this field and will be accepted as a trustworthy option for storage data access over SANs. More research needs to be done to improve the performance features that make iSCSI use over wireless thinkable by many people. Once the few lags are removed it will serve as an exceptional source of wireless transport of

storage data over TCP with removal of costly fibers and the people involved for the maintenance of the wired network.

7. ACRONYMS

ACK - Acknowledgement

AIMD – Additive Increase, Multiplicative Decrease

AQM – Active Queue Management

ARQ - Automatic Repeat reQuest

BER – Bit Error Rate

BHS - Basic Header Segment

CHAP - Challenge-Handshake Authentication Protocol

CRC - Cyclic Redundancy Check

ECN - Explicit Congestion Notification

ELN - Explicit Loss Notification

EWLN - Explicit Wireless Loss Notification

FEC – Forward Error Correction

FCIP – Fiber Channel over IP

FCoE – Fiber Channel over Ethernet

FIFO – First In First Out

HBA - Host Bus Adapter

ICMP - Internet Control Message Protocol

JBOD – Just A Bunch of Disks

MDS - Multilayer Director Switch

IQN - iSCSI Qualified Name

iSNS - Internet Storage Name Service

iSCSI – internet SCSI

LAN – Local Area Network

NCPLD - Non-congestion Packet Loss Detection

RFN - Route Failure Notification

RRN - Route Re-establishment Notification

RTT – Round Trip Time

SAN- Storage Area Network

SCSI – Small Computer System Interface

SLP - Service Location Protocol

TCP – Transmission Control Protocol

TULIP - Transport Unaware Link Improvement Protocol

UDP - User Datagram Protocol

WLAN – Wireless Local Area Network

WWN – World Wide Name

8. REFERENCES

[1] W.R. Stevens, TCP/IP Illustrated, Vol. 1. Reading, MA: Addison-Wesley, Nov. 1994.

[2] N.K.G. Samaraweera, "Non-congestion packet loss detection for TCP error recovery using wireless links," in IEE Proceedings - Communications, Vol. 146, No. 4, Aug. 1999.

[3] "Transmission Control Protocol (TCP)," DARPA, RFC 793, Sept. 1981; http://ietf.org/rfc.html

[4] H. Balakrishnan, V.N. Padmanabhan, S. Seshan, and R.H. Katz, "A comparison of mechanisms for improving TCP performance over wireless links," in IEEE/ACM Transactions on Networking, Vol. 5, No. 6, Dec. 1997.

[5] D. Bansal, A. Chandra, and R. Shorey, "An extension of the TCP flow control algorithm for wireless networks," in Proc. of 1999 IEEE International Conference on Personal Wireless Communication, 1999. *

[6] S. Goel, and D. Sanghi, "Improving TCP performance over wireless links, "In Proc. Of TENCON '98, 1998 IEEE Region 10 International Conference on Global Connectivity in Energy, Computer, Communication and Control, Vol.2, Dec. 1998.

[7] Fei P., Shiduan C., and Jian M., "An effective way to improve TCP performance in wireless/mobile networks," in Proc. of IEEE/AFCEA EUROCOMM 2000, Information Systems for Enhanced Public Safety and Security, 2000.

[8] K.K. Ramakrishnan, S. Floyd, and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP," Internet draft-ietf-tsvwg-ecn-00.txt, work in progress, November 2000.

[9] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow, "Selective acknowledgement Options," RFC-2018, 1996.

[10] S. Keshav and S. Morgan, "SMART retransmissions: Performance with overload and random losses," in Proc. IEEE INFOCOM '97, 1997.

[11] A. DeSimone, M.C. Chuah, and O.C. Yue, "Throughput performance of transport-layer protocols over wireless LANs," in Proc.IEEE Globecom '93, Nov. 1993.

[12] A. Chockalingam, M. Zorzi, and V. Tralli, "Wireless TCP performance with link layer FEC/ARQ," in Proc. 1999 IEEE International Conference on Communications, Vol. 2, 1999.

[13] H. Chaskar, T.V. Lakshman, and U. Madhow, "On the design of interfaces for TCP/IP over wireless," In Proc. MILCOM '96, Vol. 1, 1999.

[14] H. Balakrishnan, S. Seshan, and R.H. Katz, "Improving reliable transport and handoff performance in cellular wireless networks," ACM Wireless Networks, Vol. 1, Dec. 1995.

[15] C. Parsa, and J.J. Garcia-Luna-Aceves, "TULIP: A link-level protocol for improving TCP over wireless links," In Proc. WCNC '99, Vol.3, 1999.

[16] Xu, K. et all. TCP-Jersey for Wireless IP Communications. IEEE Journal On Selected Areas In Communications, Vol. 22, No. 4, May 2004. Proc. of the ACM Mobicom 2001, Rome, Italy, July 16-21 2001.

[17] Akyildiz, I.F. et all. TCP-Peach: A New Congestion Control Scheme for Satellite IP Networks. IEEE/ACM Transactions On Networking, Vol. 9, No. 3, June 2001.

[18] Brown, K. and Singh, S. M-TCP: TCP for Mobile Cellular Networks. ACM SIGCOMM Computer Communication Review, Vol. 27, Issue 5, October 1997.

[19] Bakre, A. and Badrinath, B.R. I-TCP: Indirect TCP for Mobile Hosts. Proceedings - International Conference on Distributed Computing Systems, Vancouver, Canada, 1995.

[20] Luglio, M. et all. On-board Satellite "Split TCP" Proxy. IEEE Journal on Selected Areas in Communications, Volume: 22, Issue 2, February 2004.

[21] Liu, J. and Singh, S. ATCP: TCP for Mobile Ad Hoc Networks. IEEE Journal on Selected Areas in Communications, Volume: 19, Issue 7, July 2007.

[22] Chandran, K. et all. A Feedback Based Scheme For Improving TCP Performance In Ad-Hoc Wireless Networks. IEEE Personal Communications, February 2001.

[23] Brakmo, L. et all. TCP Vegas: New Techniques for Congestion Detection and Avoidance. In proceedings of ACM SIGCOMM Conference, 1994.

[24] Saltzer, J.H. et all. "End-to-end arguments in system design. ACM Transactions on Computer Systems (TOCS), Vol. 2, Issue 4, Nov 1984.

[25] X.B. He, Q. Yang and M. Zhang, "A caching strategy to improve iSCS1 performance," *Proceedings of the 27th Annual IEEE conference on Local Computer Networks,* pp. 278-285, 2002.

[26] Y.P. Lu and D.H.C. Du, "Performance study of iSCSI-based storage subsystems," *IEEE Communications Magazine,* vol. 41, issue. 8, pp. 76-82,2003.

[27] Yan Gao, Yao-long Zhu, Hui Xiong, Renuga Kanagavelu, Jie Yan, Zhe-jie Liu, " An iSCSI Design Over Wireless Network" Data Storage Institute.

[28] J. Satran, et al., *"iSCSI draft standard,"* http://www.ietf.org/internet-drafts/draft-ietf-ips-iscsi-09.txt,2001

[29] J. Satran et al, Internet small computer systems interface (iSCSI), Technical Report RFC3720, Internet Engineering Task Force (IETF), April 2004.

[30] K.Z. Meth, J. Satran, "Features of the iSCSI protocol,"*IEEE Communications Magazine,* Aug. 2003, pp. 72-75.

[31] B. Sung, S. Park, W. Lee, C. Park, "Enhancing Robustness of an iSCSI-based File System in Wireless Networks", *Pohang University ofScience and Technology*.

[32] "Internet Protocol (IP)," RFC 791, DARPA, Sept. 1981; http://ietf.org/rfc.html

[33] M Rajagopal *et al.*, "Fiber Channel over TCP/IP (FCIP)," draft-ietf-ips-fcovertcpip-12.txt, Aug. 2002; ietf.org/html.charters/ips-charter.html or www.haifa.il.ibm.com/ satran/ips.

[34] J. Tseng *et al.*, "Internet Storage Name Service (iSNS)," draft-ietf-ips-isns-14.txt, Oct. 2002; ietf.org/html.charters/ips-charter.html or www.haifa.il.ibm.com/satran/ips

[35] J. W. Seo, H. S. Shin, and M. S. Park, "Optimizing iSCSI Parameters for improving the Performance of iSCSI based Mobile Appliance in Wireless Network", *Department of Computer Science and Engineering, Korea University*

[36] J. Chase, A. Gallatin, and K. Yocum. End system optimizations for high-speed TCP. *IEEE Communications Magazine,* 39(4):68–74, 2001.

[37] S Aiken, D. Grunwald, A. R. Pleszkun, "A Performance Analysis of the iSCSI Protocol", *Colorado Center for Information Storage.*

[38] http://www.cisco.com

http://www.cisco.com/en/US/products/hw/ps4159/ps4358/ps4359/index.html

http://www.cisco.com/en/US/prod/collateral/ps4159/ps6409/ps4358/prod_white_paper0900aecd801352e3.html

http://www.cisco.com/en/US/products/ps6069/index.html

[39] http://www.iometer.org/

[40] http://www.microsoft.com/