iSCSI 遠隔ストレージアクセス性能向上を実現する最適化手法の異なる iSCSI 実装による評価

比嘉　玲華† 　松原　幸助†† 　岡廻　隆生†† 　山口　実靖††† 　小口　正人†

† お茶の水女子大学
†† ソフトバンクテレコム株式会社
††† 工学院大学

あらまし　TCP/IP の汎用性の高さからリモートバックアップの手法として iSCSI を適用したいという期待が産業界からも高まっている．しかし，iSCSI は高遅延環境において性能が劇的に劣化してしまうということが知られている．iSCSI は複雑なプロトコルスタックとなっており，性能劣化の問題を解決するためにはネットワーク上を飛び交うパケットを解析するとともに，各層を網羅的に解析することが必要になる．そこで我々は既存研究において，パケット解析と各層を解析し最適化を行うシステムツールを提案し実装を行った．既存研究における最適化が，固有の iSCSI 実装に対する最適化ではなく，汎用的な手法であるということを示すために，本研究では既存研究とは異なる iSCSI ソフトウェアにおける評価を行った．

キーワード　ストレージ，IP-SAN，iSCSI，Linux kernel

Evaluation of optimization technique for achieving improvement of iSCSI remote storage access on different iSCSI

Reika HIGA†, Kosuke MATSUBARA††, Takao OKAMAWARI††, Saneyasu YAMAGUCHI†††, and Masato OGUCHI†

† Ochanomizu University
†† SOFTBANK TELECOM Corp.
††† Kogakuin University

Abstract　As the volume of data that computer systems process increases, efficient management of storage becomes important. Because iSCSI is able to construct the wide area IP-SAN with low cost and execute easy data backup to remote place, for example data center, iSCSI is expected to be used as Storage Outsourcing service. However, iSCSI has a problem of drastic performance deterioration in the case of longer-latency. This is because iSCSI has a complex hierarchical structure, SCSI over TCP/IP over Ethernet. Because the protocol stack of iSCSI is complicated and the storage access is executed through all these layers, any layer can be a performance bottleneck. Therefore, system tools which can analyze through all the layers are needed. We have proposed and implemented such system tools on existing research. In this reserch, we have proven our approach of optimiation of iSCSI remote storage access.

Key words　storage, IP-SAN, iSCSI, Linux kernel

1. はじめに

コンピュータシステムにおける処理データ量の増大に伴い，効率的にストレージを管理したいという要望が高まっている．そこで SAN (Storage Area Network) が登場し，広く用いられるようになった．現在主流として用いられているのは，ファイバチャネルを用いた FC-SAN であるが，構築や管理のコストが非常に高価であるため，より経済的でコストパフォーマンスの高い SAN が望まれている．そこで，次世代 SAN として期待されているのが Ethernet と TCP/IP を用いて構築する IP-SAN である．iSCSI はその IP-SAN の代表的なプロトコルであり，SCSI コマンドを TCP/IP パケットでカプセル化する規格であ

る [1] [2]．IP-SAN は，IP 技術者が多い，接続距離に限界がない，比較的安価構築可能である，相互接続性が高いなどの利点が期待されている．逆に欠点としては，性能が FC-SAN よりも低い，サーバ計算機の CPU 使用率が高くなる，などが指摘されている．

また，テロや自然災害を考えた場合，データをできるだけ遠くに書き込むというリモートバックアップは非常に重要であり，リモートバックアップのプロトコルの研究は学術的興味のみならず，産業界からも強く望まれている．よりコストパフォーマンスの高いものが利用される昨今の時勢を考えた場合，リモートバックアップのプロトコルとして iSCSI の適用が望まれる．

しかし iSCSI は，複雑な階層構成のプロトコルスタックで処理しており，バースト的なデータ転送も多いことから，現状において，通常のソケット通信と比較して特に高遅延環境においては性能の劣化が著しいことが知られている．

そこで，本研究においては，リモートバックアップへの iSCSI の適用を考え，シーケンシャルライトアクセス性能の向上に焦点を絞り，高遅延環境下における iSCSI リモートストレージアクセス性能解析を行った．

iSCSI は複雑なプロトコルスタックからなるためにすべての層にまたがる解析を行う必要がある．そこで，SCSI 層，iSCSI 層，TCP 層，Ethernet 層における解析や最適化を行った．その結果，性能に一番影響を与えたのは TCP 層における最適化であった．我々が提案したシステムツールを用いたことにより，ソケットバッファの割り当てが不十分でパケットが断続的に送信されてしまっていることが原因であることが判明した．そこでカーネル解析を行い，条件分岐箇所をカーネルコード上で特定し，その場所に性能向上のための適切なソフトウェアモジュールを挿入した．その結果，高遅延環境下における iSCSI 性能劣化の問題を解決することができた．

本稿では，上記の提案手法が，固有の iSCSI ソフトウェアに対する最適化手法ではなく，iSCSI 全般の問題に対応可能であるということを示すために，Open iSCSI [3] と Linux iSCSI [4] の二つの iSCSI 実装で評価を行うことで有効性を示す．

本稿は以下のように構成されている．まず，第 2 章において研究背景として iSCSI と既存研究について紹介する．次に第 3 章では，実験システムと解析システムツールについて説明する．第 4 章において Linux iSCSI における最適化，第 5 章で，その考察について述べ，最後に，第 6 章において本稿のまとめと今後の課題を述べる．

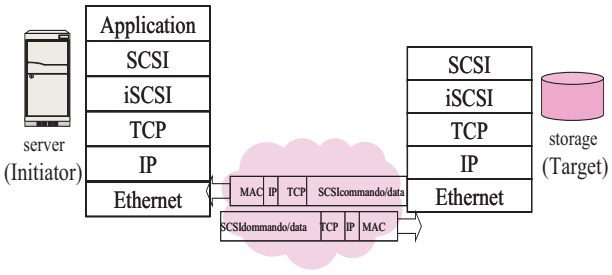

図 1　Configuration of iSCSI

2. 研究背景

2.1 iSCSI

IP-SAN の代表的なプロトコルに iSCSI がある．iSCSI は SCSI コマンドを TCP/IP パケットでカプセル化する規格で，iSCSI により SAN を IP 機器だけで構築することが可能になる．専用線を使用せず IP 機器だけで構築可能であるため，接続距離に制限を設けない，安価に SAN を構築可能である，管理が容易であるといったメリットを持っている．そのメリットを生かして，iSCSI の広域ネットワークへの適用の期待が高まっている．一方で，iSCSI は高遅延環境になるほど性能劣化してしまうことが一般的に知られている [5]．

iSCSI は図 1 のような複雑な階層構造をとることになり，下位のプロトコルの限界性能を超えることができない．iSCSI 通信中には全ての層を経由して通信が行われるため，全ての層が性能劣化の要因となり得てしまう．そこで，性能劣化の要因を知るためには，全ての層を階層横断的に解析する必要が生じる．

2.2 既存研究

我々は，これまでに Open iSCSI を用いたリモートストレージアクセスの最適化を行ってきた [6]．iSCSI リモートストレージアクセスにおける性能劣化要因を解析するために，階層横断的に解析が可能なシステムツールを提案，実装，評価を行った．システムツールについては，次章にて述べる．

既存研究では iSCSI の Initiator から Target へバーストアクセスを行いスループットを測定した．

Ethernet 層においては，NIC パラメータ最適化を行った．その結果，わずかな性能向上は達成されたが，大きな影響を及ぼすことはないことが確認された．

iSCSI 層においては，iSCSI パラメータ最適化を行った．その結果，RTT=20ms においてデフォルト時と比較して約 4 倍の性能向上が達成された．しかし，まだなお，高遅延環境下における急激な性能劣化という問題は解決されなかった．

そこで次に，急激な性能劣化の要因を解析するために，ネットワーク上を飛び交うパケットを解析した．SCSI レベルでブロックサイズを 4MB と指定してバーストアクセスを行ったのにもかかわらず，約 1MB ずつにパケットが分断されてパケット送信が行われていることが確認された．この結果より，パケット送信の断続が，高遅延環境における急激な性能劣化の要因であることがわかった．さらに解析を進めたところ，パケット送信の断続の原因はトランスポート層にあることが解明された．

TCP 層における解析としては，広告ウィンドウ，輻輳ウィンドウ，ソケットバッファ解析を行った．その結果，広告ウィンドウと輻輳ウィンドウ共に大きさは十分であることが確認された．iSCSI 通信においては，ソケットバッファのキューが十分に割当られていないことが確認されたので，ソケットバッファ最適化を行った．その結果，高遅延環境下における急激な性能劣化問題が解決されたことが確認された．

既存研究では，TCP 層におけるソケットバッファ最適化が性能向上に一番大きな影響を与えていた．ソケットバッファ最適化の手法としては，iSCSI ソフトウェアを直接修正するのでは

なく，その下の層の TCP 層に対して最適化を行うという，より汎用性の高いアプローチをとった．特定の iSCSI ソフトウェアに限った対処ではなく，幅広いケースに対応が可能であると考えている．

しかし，既存研究では Open iSCSI に対する最適化のみが行われており，汎用性を示せていないことが課題としてあげられる．

3. 実験システム

3.1 実験環境

本研究において,Initiator と Target 間は GigabitEthernet で接続し，TCP/IP コネクションを確立した．Target のストレージには SAS ディスクを用い RAID コントローラによる RAID0 構成で接続した. 使用した実装システムと実験環境を図 2 および表 1 に示す.

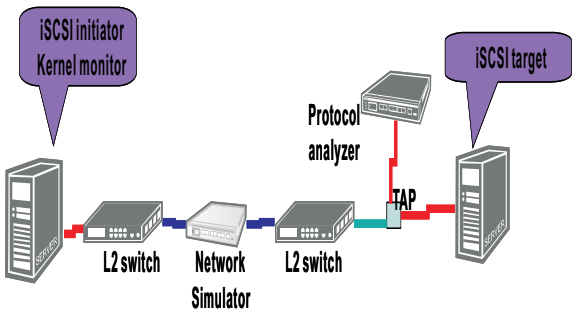

図 2　実装システム概要

表 1　実験環境

OS	Red Hat Enterprise Linux 2.618-8.e.15
CPU	Quad Core Intel Xeon 1.6GHz
Main Memory	2GB
NIC	Intel PRO/1000PT Server Adaptor on PCI Express
HDD	73GB SAS×2(RAID0)
RAID Controller	SAS5/iR
iSCSI	Initiator : open-iscsi-2.0-870, linux-iscsi-6.2.0
	Target : iSCSI Enterprise Target(IET)-0.4.15
Network Analyzer	ClearSight Network Recorder
Network Simulator	ANUE

3.2 解析システムツールの適用

iSCSI ストレージアクセスは多段のプロトコルで構成されるため，全層を経由して行われることから全層が性能劣化原因となる可能性があり，性能向上について考察するにはこれら全層を網羅的に解析する必要がある．そこで本稿において図 3 のようなシステムツールを構築した．このシステムツールを使用することで，以下が可能となる．

- 複数の層にまたがるパラメータ最適化
- ネットワーク上を往来するパケット解析
- Initiator 上におけるカーネル解析

システムツール構成要素の一つに我々のオリジナルのツールである「カーネルモニタ」がある．これは TCP カーネルの振舞をモニタするツールである．カーネル内部の TCP ソースにモニタ関数を挿入しカーネルを再コンパイルすることで，一般にはユーザ空間からは見ることができない TCP のパラメータ情報などを可視化できるというツールである．これによりモニタできるようになった値には，iSCSI 通信中のログのトレース情報，タイムスタンプ情報，輻輳ウィンドウやソケットバッファキュー長といった TCP パラメータ情報がある．その他のシステムツール構成要素としては，tcpdump コマンド，ネットワークアナライザ [7] がある．これらによって，ネットワーク上を飛び交うパケットを解析し，パケット情報を得ることが可能になる．本稿においては，このシステムツールを効果的に使用しすべての層を解析，最適化を行うことで，iSCSI 遠隔ストレージアクセスの性能低下の原因がどこにあるのかを詳しく解析していく．

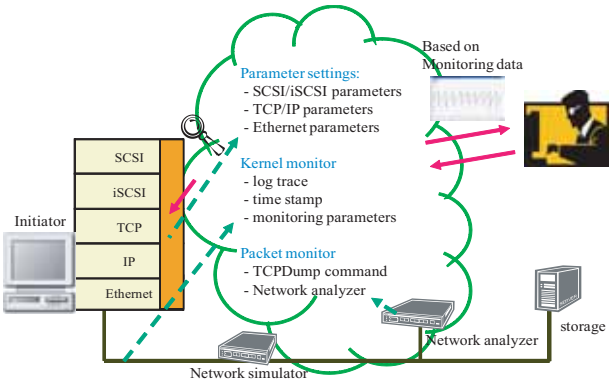

図 3　解析システムツール概要

4. Linux iSCSI における最適化

4.1 最適化手法

既存研究において効果的であった最適化としては，iSCSI パラメータ最適化，ソケットバッファ最適化がある．そこで，本稿においては，Linux iSCSI における最適化手法として，iSCSI パラメータ最適化，ソケットバッファ最適化の 2 つを行っていくことで，既存研究における最適化手法の汎用性を示す.

その際，sg_dd コマンドを使用できるようにカーネルを再構築し，アクセスブロックサイズを 4MB と指定した．sg_dd コマンドとは，dd コマンドと文法的に互換性はあるが，dd コマンドとは異なり，iSCSI アクセスにおいて SCSI レベルで指定したブロックサイズによるアクセスが可能となるコマンドである [8]．また，広告ウィンドウを通信の妨げにならない程度の値に設定した.

4.2 iSCSI パラメータ最適化

リモートバックアップを考慮し，ライトアクセス時に性能が最大になるように iSCSI パラメータを最適化した．iSCSI パラメータのうち，MaxRecvDataSegmentLength， MaxXmitDataSegmentLength，FirstBurstLength，MaxBurstLength，InitialR2T の値を最適化した．比較として，ソケット通信時とデフォルト時の iSCSI パラメータ時のスループットを含めたスループット測定の結果を図 4 に示す.

RTT=20ms においてデフォルト時と比較して約 3 倍の性能

向上が達成されていることが確認された．しかし，まだなお，高遅延環境下における急激な性能劣化という問題は解決されていない．

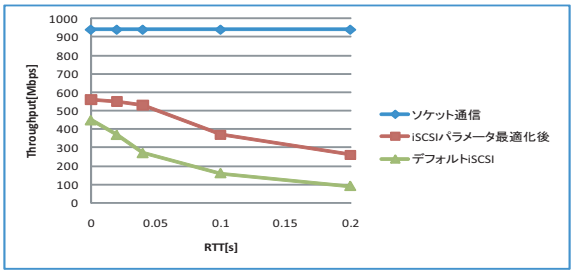

図 4　iSCSI パラメータ最適化後のスループット

4.3　ソケットバッファ最適化

以下の実験においては，前節の最適化後の iSCSI パラメータを使用して測定を行った．既存研究では，高遅延環境における急激な性能低下の原因は，送信パケットの断続であった．その送信パケット断続を引き起こす原因は，トランスポート層にあることが既存研究では確認されている．

4.3.1　ソケットバッファ最適化手法

本研究では，既存研究と同様な手法を用いてソケットバッファ最適化を行う．

以下に既存研究で用いた手法を示す．既存研究では，iSCSI 通信中ではソケット通信と比較して，不必要な Timeout 待ちが多く発生してしまっていることが原因でソケットバッファがうまく割当られていなかった．そこで，更なるカーネル解析を行ったところ，TCP 実装中の sk_stream_memory_free() 関数が Timeout 待ちに陥ってしまう分岐点となることを特定した．関数を図 5 に示す．この関数は，変数 sk_wmem_queued と変数 sk_send_buf の大きさを比較して値を返している．カーネルモニタを使用して，iSCSI 通信中の sk_send_buf の値を調べたところ，ソケット通信中ではこの値が 4MB まで成長するのに対して，iSCSI の場合は 1MB までしか成長しないことが確認された．このことから，iSCSI ソフトウェアの実行中に，sk_send_buf の値が設定されていると想定される．sk_wmem_queued が大きい場合にタイムアウト待ちに陥ることが解析の結果わかったので，sk_send_buf の値の最適化を TCP カーネル上で行った．

本研究においても，同様なアプローチを TCP カーネル上で行った．

linux+v2.6.18.5/include/net/sock.h#L446

```
static inline int sk_stream_memory_free(struct sock *sk)
{
 return sk->sk_wmem_queued < sk->sk_sndbuf;
}
```

図 5　sk_stream_memory_free() 関数

4.3.2　ソケットバッファ最適化後の評価

ソケットバッファ最適化後のスループットを図 6 の ”ソケットバッファ最適化後 ”に示す．RTT=20ms においても十分に高いスループットが出ており，高遅延環境下における急激な性能劣化問題解決が達成されていることがわかる．

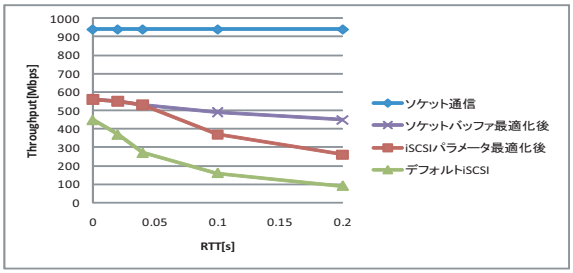

図 6　ソケットバッファ最適化後のスループット

5.　異なる iSCSI 実装評価に対する考察

Linux iSCSI においても，性能向上に最も影響を与えたのは，TCP 層におけるソケットバッファ最適化であることが確認された．ソケットバッファ最適化の際に，より汎用的な手法となるように，iSCSI ソフトウェアそのものを直接修正するのではなく，TCP 層に対する処理を行った．本稿で示したように，既存研究とは異なるソフトウェアに対してもこの手法が有効であることが確認された．

このことから，本手法の iSCSI 最適化における汎用性を示すことができたと考えられる．

6.　まとめと今後の課題

接続距離に制限がなく IP-SAN を比較的安く容易に構築可能というメリットを生かして，リモートバックアップのプロトコルに iSCSI を適用することが産業界からも高く期待されている．しかし，iSCSI には高遅延環境において性能が劇的に劣化してしまうという問題がある．そこで，既存研究において，複雑なプロトコルスタックを持つ iSCSI の解析を行うために，iSCSI が高遅延環境で性能低下する原因を解析し性能向上を実現するシステムツールの提案，実装をおこなった．

解析の結果，性能向上に一番影響を与えたのは TCP 層におけるソケットバッファの最適化であった．これにより，高遅延環境における性能劣化問題の解消が達成された．

性能向上の際に，iSCSI ソフトウェアを直接修正するのではなく，その下の TCP 層に対して最適化を行ったことは，より汎用性の高いアプローチであり，特定の iSCSI ソフトウェアに限った対処ではない．このことが，異なる iSCSI ソフトウェアにおける遠隔ストレージアクセス最適化を行うことで証明された．このことから，我々の行った最適化，システムツールは幅広いケースに対応が可能であると考えられる．

本来 iSCSI はローカル環境で使用されることを想定されており，高遅延環境においてバースト的な転送が行われることが想定されていなかった．高遅延環境下において iSCSI ストレージアクセスを行う場合においては，TCP 層においてソケットバッ

ファの最適化が必要であることが本研究により指摘された．また，他の iSCSI を使用したときに他の箇所において問題が生じた場合においても，本システムツールを使用した複数の層にまたがる解析，最適化は有効であり得る．

今後の課題としては，より汎用性を示すためにも，ターゲットとしてハードウェアを用いた実験を行っていきたい．また，本システムツールをより洗練されたものにしていきたい．そして，本研究は，1 対 1 の理想的な iSCSI 通信を構築して解析を行ってきたが，本研究で得られた最適化手法を用いて，実環境を想定した環境で解析を行っていきたい．

文　献

[1] iSCSI Specification, http://www.ietf.org/rfc/rfc3720.txt?number=3270

[2] SCSI Specification, http://www.danbbs.dk/~dino/SCSI/

[3] Open iSCSI Specification, http://www.open-iscsi.org/

[4] Linux iSCSI Specification, http://linux-iscsi.sourceforge.net/

[5] S. Yamaguchi, M. Oguchi, and M. Kitsuregawa: ”iSCSI Analysis System and Performance Improvement of Sequential Access in Long-Latency Environment,” IEICE Transaction on Information and Systems, Vol.J87-D-I, No.2, pp.216-231, February 2004.

[6] Reika Higa, Kosuke Matsubara, Takao Okamawari, Saneyasu Yamaguchi, and Masato Oguchi ”Analytical System Tools for iSCSI Remote Storage Access and Performance Improvement by Optimization with the Tools,”to be published in the 3rd IEEE International Symposium on Advanced Networks and Telecommunication Systems (ANTS2009), New Delhi, India, December 2009.

[7] http://www.toyo.co.jp/clearsight/product/analyzer.html

[8] http://sg.torque.net/sg/sg3_utils.html/