

iSCSI 遠隔ストレージアクセスの性能向上を実現する 汎用的な最適化手法とその評価

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

1 はじめに

近年、ストレージ管理コスト低減などの目的で SAN(Storage Area Network) の導入が進んでおり、その中でも、より経済的でコストパフォーマンスの高い IP-SAN のプロトコルである iSCSI は大いに期待されている。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保存する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システム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 の適用が期待されている。しかし iSCSI をリモートバックアッププロトコルとして実用化するには様々な問題がある。最も大きな問題として、iSCSI は高遅延環境になるほど性能が劣化してしまうというデメリットが一般的に知られている。

そこで我々は、リモートバックアッププロトコルとして iSCSI を使用することを目的として、高遅延環境における iSCSI シーケンシャルライトアクセスの性能向上を実現するための手法の提案と実装を行い、その実行性能を解析および評価してきた [1]。

本稿では、我々の提案手法の汎用性を示すべく既存研究で用いた Open-iSCSI とは別の Linux-iSCSI を用いて、性能の最適化を行い評価した。

2 実験システム

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 構成とした。実験環境を表 1 に示す。

3 既存研究における Open iSCSI を用いた評価

iSCSI ストレージアクセスは図 1 のように多段のプロトコルで構成されるため、全層を経由して通信処理が行われることから全層が性能劣化原因となる可能性

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

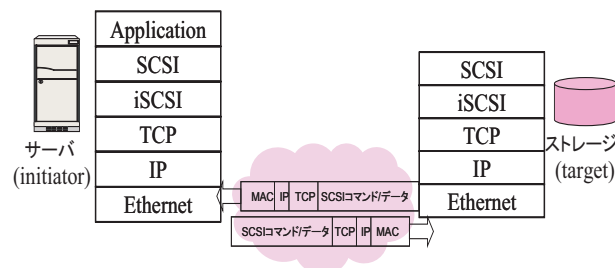


図 1: Configuration of iSCSI

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。そこで図 2 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、ネットワーク上を飛び交うパケット解析と Initiator におけるカーネル解析を行うことが可能となる。

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、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 パラメータ情報や log trace timestamp 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。このシステムツールを効果的に使用しすべての層の解析、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

[†] Reika Higa, Masato Oguchi

[‡] Kosuke Mtsubara, Takao Okamawari

[§] Saneyasu Yamaguchi

Ochanomizu University ([†])

SOFTBANK TELECOM Corp. ([‡])

Kogakuin University ([§])

詳しく解析していった。

その結果、特に TCP 層におけるソケットバッファ最適化がもっとも有効であり、高遅延環境における性能の劣化問題を解決した。

既存研究においては、iSCSI ソフトウェアに対して直接最適化を行うのではなく、より汎用的な最適化となるように、下位層である TCP 層に対する最適化を行っている。

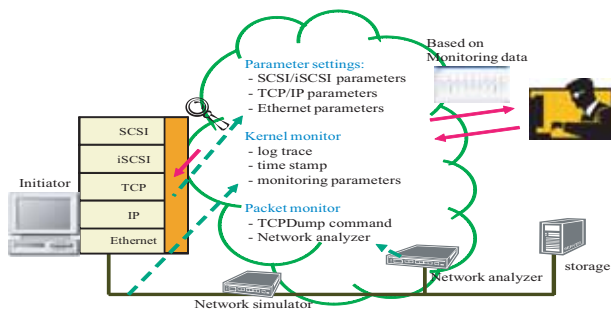


図 2: 解析システムツール概要

4 Linux iSCSI における最適化手法適用

本稿では、既存研究において効果的であった最適化である、iSCSI パラメータ最適化、TCP 層におけるソケットバッファ最適化について評価していく。その際アクセスブロックサイズを 4MB、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。

4.1 iSCSI パラメータ最適化

性能が最も高くなるように、iSCSI パラメータにおいて、MaxRecvDataSegmentLength, MaxXmitDataSegmentLength, FirstBurstLength, MaxBurstLength, InitialR2T の値を最適化した。比較として、デフォルト時を含め 3 種類のパラメータを図 3 に示す。

4.2 ソケットバッファ最適化

既存研究において、ソケット通信と iSCSI 通信においてはソケットバッファのキュー長の振舞に大きな違いがあることが確認されている。その理由は、iSCSI 通信時においては不必要なタイムアウト待ちが頻繁に生じているということであった。不必要なタイムアウト待ちを取り除くためにカーネル解析をして条件分岐点となるコードを特定し TCP レベルにおける対処をした。Linux iSCSI においても同じ箇所に対処をしソケットバッファの最適化を行った。

5 性能測定結果

最適化の結果を図 3 に示す。iSCSI パラメータ最適化後の性能はデフォルト時の性能と比較すると RTT20ms において約 3 倍の性能向上が達成されている。しかし、尚も高遅延環境での性能劣化という問題は解決されて

いない。ソケットバッファ最適化後の性能はデフォルト時の性能と比較すると RTT20ms において約 5 倍の性能向上が達成されていること、また、高遅延環境においてもほぼ性能が保たれていることがわかる。Linux iSCSI においても我々の提案した最適化手法によって、iSCSI リモートストレージアクセスの実現がされたと言える。

本稿においても、TCP 層におけるソケットバッファ最適化が一番効果的であったということが確認された。すなわち、我々の提案する最適化手法は、固有の iSCSI ソフトウェアに対するものではなく、汎用性をもつ手法であることができる。

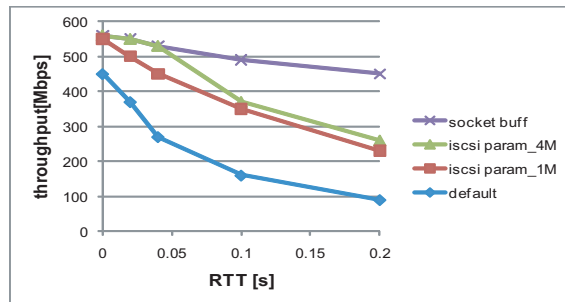


図 3: 各最適化後のスループット比較

6 まとめと今後の課題

既存研究では iSCSI リモートストレージアクセスの性能向上を実現するためのシステムツールの提案と実装を行った。本稿においては、我々の最適化提案手法が iSCSI ソフトウェア固有のものではなく、汎用性の高いものであることを示すべく、既存研究とは異なるソフトウェアを用いて評価を行った。

その結果、TCP 層におけるソケットバッファ最適化が非常に有効であることがわかった。このことから、iSCSI はローカル内での使用を想定したものであったため、低遅延環境下においてはソケットバッファの割当てが十分に間に合っていたが、高遅延環境においてバーストアクセスが行われることは想定外であり、その結果、高遅延環境においてはソケットバッファの割当てが不足し性能が劇的に劣化してしまったということが原因として考えられる。

今後は、パケットロス環境下などの、より現実的な環境における解析を行うことで、iSCSI リモートストレージアクセス性能向上を実現するための有効な手法を提案していきたい。

参考文献

- [1] Reika Higa et al. : "Analytical System Tools for iSCSI Remote Storage Access and Performance Improvement by Optimization with the Tools, " ANTS2009, New Delhi, India, December 2009