

VPN 複数経路接続時における iSCSI ストレージアクセスの特性解析

千島 望[†] 山口 実靖^{††} 小口 正人[†]

近年、ストレージの管理コスト低減などの目的で SAN の導入が進んでおり、IP ネットワークを利用した IP-SAN として iSCSI が期待されている。しかし現状では SAN は主にサイト内のローカル環境のみで用いられている。そこで本稿では VPN を利用することにより、iSCSI を広域ネットワークに適用することを検討した。さらに、より信頼性の高い通信を実現するため VPN 広域ネットワーク内を複数経路で接続し、高遅延環境における iSCSI ストレージアクセスの特性を解析、評価した。

Analysis of iSCSI Storage Access in the case of Multi-routing VPN

Nozomi Chishima,[†] Saneyasu Yamaguchi^{††} and Masato Oguchi[†]

The introduction of SAN progresses for the purpose of the storage management cost reduction, and iSCSI is expected as IP-SAN that uses IP network. However, SAN is mostly used only in the server site currently.

Thus we have claimed iSCSI used in a local environment can be applied to the WAN using VPN. We have examined a realization method and performance for it. Furthermore, we have evaluated behavior of the congestion window, which is one of the TCP parameters, when multi-routing iSCSI access is performed using a multi-routing function of a VPN router.

1. はじめに

近年、インターネット技術の進展などにより、ユーザが蓄積し利用するデータ容量が爆発的に増加している。これに伴いストレージの増設、管理コストの増大が問題となっている。そこで SAN (Storage Area Network) が登場し、広く用いられるようになった。SAN とは、サーバとストレージを物理的に切り離し、各ストレージとサーバ間を相互接続してネットワーク化したもので、これにより各サーバにばらばらに分散していたデータの集中管理が実現された。

一般に SAN としてはファイバチャネルを用いる FC-SAN (Fibre Channel - SAN) が利用されている。しかし、FC-SAN はファイバチャネルを用いているため高価となり、また距離に制約がある。一方、SAN に IP ネットワークを利用した IP-SAN として iSCSI が期待されている [1][2]。iSCSI は、これまで DAS (Direct Attached Storage) で使われてきた SCSI コマンドを TCP/IP パケット内にカプセル化することにより、サーバ (Initiator) とストレージ (Target) 間でデータの転

送を行う。今後インターネットの発展により、ギガビットクラスの回線実現が期待され、iSCSI の有効性もさらに高まると考えられる。

現状において、SAN は主にサーバサイト内のみでしか使用されていない。しかし遠隔バックアップ等を目的として、離れたサイトのサーバとストレージを SAN で接続することが望まれている。そこで本研究では、VPN (Virtual Private Network) を利用することにより、ローカル環境で使用されている iSCSI を用いて広域ネットワーク上でリモートアクセスを行うことを検討した。さらに、より信頼性の高い通信を実現するため VPN 広域ネットワーク内に複数経路を構築した。

iSCSI は複雑な階層構成のプロトコルスタックで処理されており、バースト的なデータ転送も多いことから、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く、さらに下位基盤の TCP/IP 層が提供できる限界性能を超えることはできない [3]。また広域環境で iSCSI 複数経路アクセスを行う場合、経路によりネットワーク遅延やネットワーク性能が異なるため、iSCSI で適応的なパケット処理を行うことが望ましい。そこで本研究では、VPN 複数経路接続において異なる遅延時間をもつ経路を構築して実験を行い、iSCSI ストレージアクセスの特性を解析する。

[†] お茶の水女子大学
Ochanomizu University

^{††} 工学院大学
Kogakuin University

本稿の構成は以下の通りである．2章で研究背景を述べ，3章でVPNのマルチルーティング機能を利用した本実験システムの概要を述べる．4章で経路ごとのiSCSIアクセスについての性能評価結果を示し，5章で各経路の遅延時間による影響を比較し，最後に6章でまとめる．

2. 研究背景

2.1 iSCSI

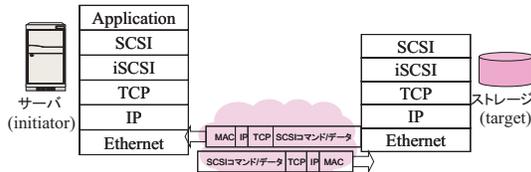


図 1 iSCSI

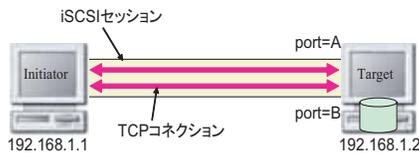


図 2 iSCSI 複数コネクション

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となる．一方で図1のように複雑な階層構成をとることになり，下位のプロトコルの限界性能を超えることはできない．また，iSCSIには長距離アクセスの実現が期待されているが，ギガビットの太い回線を用いることから遅延帯域積の問題も挙げられている．そこで下位基盤のTCP/IP層の適切な制御が求められている．

iSCSIは様々なチューニングを行うことができる．本実験で用いたニューハンプシャー大学が提供するUNH-iSCSIの実装では，1つのiSCSIセッション内に複数のTCPコネクションを確立するように設定することができる[4]．さらにこのコネクションをポート番号と対応付けることができ，1つのiSCSIセッションをポート番号が異なる複数のコネクションに分けることができる．つまり，図2に示すようにターゲットの1つのIPアドレス，1つのiSCSIドライブにポート番号の異なる複数のコネクションを接続可能である．

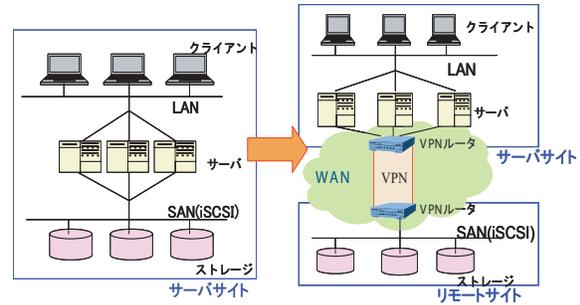


図 3 VPN 利用モデル

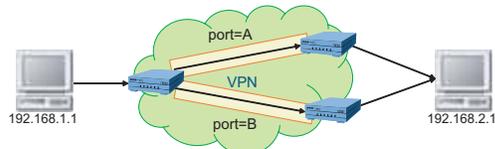


図 4 VPN マルチルーティング機能

2.2 VPN

VPNは，インターネットや通信事業者が持つ公衆ネットワークを使って，拠点間を仮想的に閉じたネットワークで接続する技術である．安価であるという公衆網のメリットを活かしつつ，機密性の低さを暗号化等の別の方法で補うことにより，「実質的な専用網」を実現できるということがVPNの利点である．一方，専用網と異なりネットワークの品質は保証されない場合が多い．

本研究ではiSCSIを用いて遠隔バックアップなどを行うために，図3に示すようにVPNルータで接続したリモート環境にネットワークストレージを設置し，広域ネットワーク内のVPN越しにアクセスを行うことを考えた．この場合，VPNルータを通ることによってネットワークの帯域幅が制限され，スループットが著しく低下することが起こり得る[5]．さらに広域ネットワーク内は不安定な通信路であることが想定される．そこで本稿では，VPN広域ネットワーク内を複数経路で接続することを考えた．これにより，データ転送の信頼性やネットワークの耐障害性なども向上すると考えられる．

本実験で用いたVPNルータFujitsu Si-R570はマルチルーティング機能を有している[6]．マルチルーティング機能を使用すると，ポート番号などの情報を利用して同じ先IPアドレスを持つネットワークへ複数の経路を用いて送信することが可能となる．それぞれの通信内容に通信経路を分離することが出来るため，片方の回線をバックアップ用に用いたり，音声

データは専用線を用いてそのほかの通信は公衆網を用いるなどと設定することができる。本稿ではこの機能を利用し、iSCSI 複数コネクションと対応付けることにより、コネクションごとに異なる経路を構築することを可能にした。

iSCSI は通常ギガビットクラス以上の太いネットワーク上で用いられるが、途中で VPN ルータの暗号化処理速度などによりスループットが決まる細い回線が挟まることにより、トラフィックとして大いに性質の異なるものになる [7]。従って iSCSI が最大限の性能が発揮できるように TCP パラメータなどを制御することが求められる。

2.3 TCP 輻輳ウィンドウ

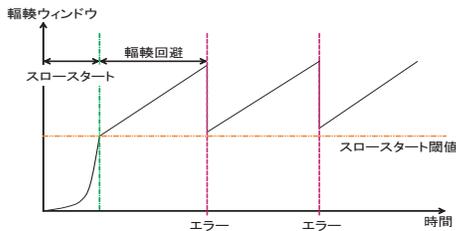


図 5 輻輳ウィンドウの変化

TCP では、通信能力の制御にウィンドウサイズという概念を用いている。ウィンドウサイズとは、ホストが確認応答パケット (Acknowledgement:ACK) なしに一度に送信できるデータの量である。また、データの送信側では輻輳ウィンドウ、受信側では広告ウィンドウという値が決定され、このどちらか小さい方がウィンドウサイズとして用いられる。広告ウィンドウは現在の受信バッファの空き容量を示しており、ACK のヘッダにその情報が含まれて送信側に送られる。一方、輻輳ウィンドウは送信側の制御パラメータで、ネットワークの混雑を回避するため送信側が自主的に制限する値である。

輻輳制御ではこの輻輳ウィンドウが利用されている。輻輳制御はネットワークの混雑解消の方法として TCP が行う機能である。通信開始時にはスロースタートと呼ばれるアルゴリズムに従って指数関数的に輻輳ウィンドウが大きくなる。これによりトラフィックが急激に増加するので、ネットワークが輻輳状態になる可能性がある。これを防ぐため、スロースタート閾値という値を用意し、輻輳ウィンドウがその大きさを超えると輻輳回避と呼ばれるフェーズに入り、一次関数的な増え方となる。そしてエラーが検出されると輻輳ウィンドウは急激に低下し、通常これらを繰り返すことで

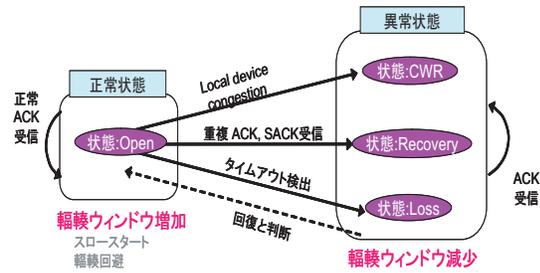


図 6 LinuxTCP の状態遷移

鋸型のグラフとなる。この様子を図 5 に示す。

また本実験で用いた LinuxOS における TCP の状態遷移を図 6 に示す。LinuxTCP においては、通信時の状態が正常であれば ACK の受信ごとに輻輳ウィンドウは増加するが、エラーが検出されると異常と判断され、輻輳ウィンドウは低下する。輻輳ウィンドウが低下する原因としては、送信側デバイスドライバのバッファが溢れることによる Local Congestion エラーを検出した場合 (CWR)、重複 ACK 又は SACK を受信した場合 (Recovery)、タイムアウトを検出した場合 (Loss) の 3 つが挙げられる。さらに Linux の TCP 実装では、通信中に一度設定された輻輳ウィンドウは、そのウィンドウ値を超えるデータ量が送られない限りは変化しないという特徴を持ち、この時スループットはほぼ一定の値で安定することが確認されている。

2.4 既存研究

我々は、これまでに iSCSI を用いたアプリケーション実行性能と TCP パラメータの相関関係の評価を行った [8]。その結果、広告ウィンドウの値を制限することで、輻輳ウィンドウの値も制限でき、それによって実行性能にも影響が出ることが確認された。また、VPN 利用時のネットワークや、iSCSI ストレージアクセスによる性能測定と TCP 輻輳ウィンドウの振舞いを観察した [5][7]。

iSCSI ストレージアクセスにおいて TCP 輻輳ウィンドウを制御する研究としては、輻輳ウィンドウ値を動的にコントロールする手法がある [9]。この手法は、まず Target の OS のカーネルに輻輳ウィンドウモニタ関数を挿入し、これによりモニタした輻輳ウィンドウの変化を観察して、Initiator にその値を通知する。通知を受けた Initiator は輻輳ウィンドウの値に基づきブロックサイズを再指定して、シーケンシャルリードアクセスを行うというものである。この手法を適用し輻輳ウィンドウを限界値で一定に保った場合には、高遅延環境において最大 28% のスループットの向上が確認されている。

さらに iSCSI 複数コネクションに関する研究として、広域 IP 網を介した長距離アクセス向けに iSCSI および関連プロトコルレイヤのプロトコルチューニングの検討が行われ、その有効性が確認されている [10] .

そこで本稿では、iSCSI 複数コネクションを VPN 複数経路に乗せてアクセスすることを検討した。これらを組み合わせることにより、性能の向上だけでなく信頼性の向上も期待できる。また広域環境で iSCSI 複数経路アクセスを行う場合、経路によりネットワーク遅延やネットワーク性能が異なるため、iSCSI で適応的なパケット処理を行うことが望ましい。そこで本研究では VPN 複数経路接続において異なる遅延時間をもつ経路を構築して iSCSI アクセスの特性を調べた。

3. 実験システム

3.1 TCP 輻輳ウィンドウモニタツール

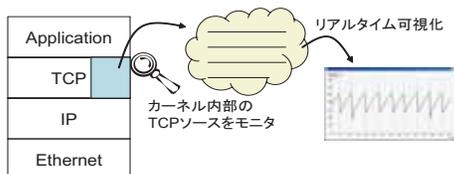


図 7 TCP 輻輳ウィンドウモニタツール

本実験では、TCP 輻輳ウィンドウをモニタするツールを構築した。図 7 に示すように、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した。ここでモニタできるようになったものには、輻輳ウィンドウの値の他、各種エラーイベントの発生 (Local device congestion, 重複 ACK, SACK 受信, タイムアウト検出) などがある。また X11 ウィンドウシステムライブラリ関数を用いて、モニタした値をリアルタイムに可視化することもできるようになっている。

3.2 VPN 複数経路アクセス制御システム

本実験では、VPN ルータを用いて複数経路を構築し、経路ごとに異なる遅延時間を設定し iSCSI ストレージアクセスを実行した時の、性能と輻輳ウィンドウを評価するために図 8 に示す実験環境を構築した。

iSCSI ストレージアクセスを行う Initiator とストレージを提供する Target の間に VPN ルータを 4 台を挟み、複数経路アクセスが実行できるように構築した。さらにそれぞれの経路に、遠距離アクセスを想定して人工的な遅延装置である FreeBSD Dummynet を挿入した [11] .

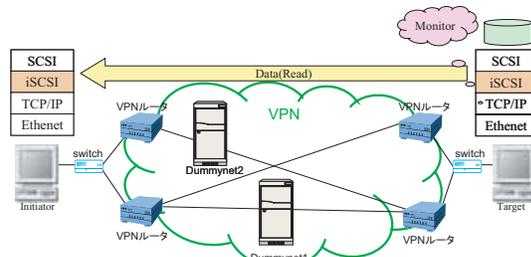


図 8 実験システムの概要

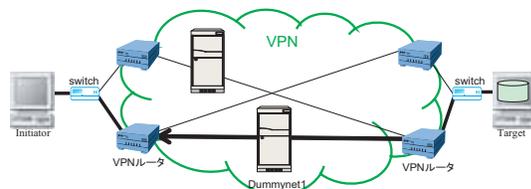


図 9 iSCSI 単数コネクション VPN 単数経路

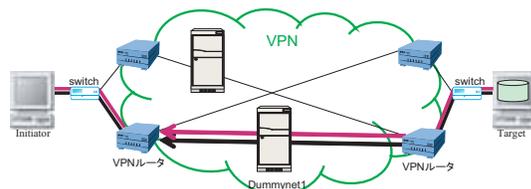


図 10 iSCSI 複数コネクション VPN 単数経路

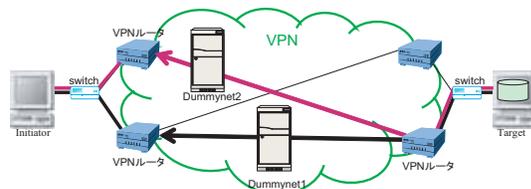


図 11 iSCSI 複数コネクション VPN 複数経路

この実験システムにおいて iSCSI の複数コネクション設定と VPN ルータのマルチルーティング機能を用いて通信制御を行った。以下では iSCSI リードアクセス、すなわち Target から Initiator へデータが転送される場合について説明しているが、逆も基本的に同じである。まず iSCSI 単数コネクション VPN 単数経路通信の場合は図 9 のような経路を通る。また iSCSI 複数コネクション VPN 単数経路通信の場合は図 10 のように同一経路上を 2 つのコネクションが張れるように iSCSI を設定した。さらに iSCSI 複数コネクション VPN 複数経路通信の場合は図 11 のようにコネクションごとに経路が異なるように VPN ルータの設定

を行った。このとき Target から送られるパケットは図 11 のように右下の VPN ルータに送るように設定する。そして、VPN ルータのマルチルーティング機能により左の 2 つの VPN ルータ宛にパケットは転送される。このとき、ポート番号の違いにより iSCSI コネクションごとに上下の VPN ルータに分かれるように設定した。

Initiator と Target には、OS は Linux2.4.18-3, CPU は Intel Xeon 2.4GHz, Main Memory は 512MB DDR SDRAM, NIC は Intel Pro/1000XT Server Adapter on PCI-X (64bit,100MHz), iSCSI は UNH IOL reference implementation ver.3 on iSCSI Draft 18 を用いた [4]。そして Dummynet1 には FreeBSD4.9-RELEASE, Dummynet2 には FreeBSD6.2-RELEASE を用いた。また VPN ルータには Fujitsu Si-R570 を用いた [6]。これは 3DES 暗号化速度最大 500Mbps を実現する。

この実験環境において、TCP 輻輳ウィンドウモニターツールを起動し、iSCSI シーケンシャルリードアクセス時の性能や TCP パラメータの振舞いを観察した。

本実験ではストレージアクセスのみの性能を評価するため、Initiator 側では raw デバイスを使用することにより、キャッシュの影響を排除した。また、iSCSI ストレージアクセスにおけるネットワーク性能に焦点を当てて評価を行うため、Target は UNH 実装が提供するメモリモードで動作させ、ディスクアクセスを伴わないようにした。

4. 経路ごとの実行結果の比較

4.1 スループット測定結果

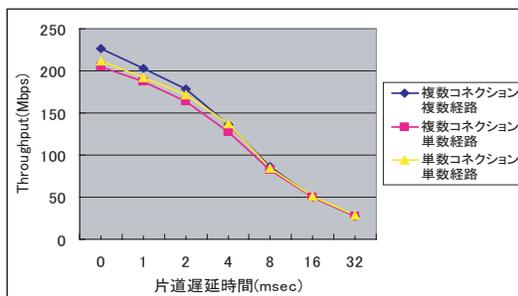


図 12 片道遅延時間とスループット比較

図 12 は iSCSI 単数コネクション VPN 単数経路, iSCSI 複数コネクション VPN 単数経路, iSCSI 複数コネクション VPN 複数経路各々のケースにおいて、片道遅延時間を変化させた時の iSCSI ストレージアク

セススループット比較のグラフである。この実験においてブロックサイズは 2MB に設定した。また、iSCSI 複数コネクション VPN 複数経路の時、経路ごとの遅延時間は同じ値を設定した。

どの場合も片道遅延時間を長くするとスループットは急激に減少した。遅延時間が短い時には複数コネクション複数経路の場合が一番性能が良く、続いて単数コネクション単数経路、複数コネクション単数経路となっている。

複数コネクション複数経路の場合にスループットが単数コネクション単数経路より向上した理由を考察する。Initiator と Target はギガビットイーサネットで接続しているが、VPN ルータの暗号化速度は最大 500Mbps である。また本実験の VPN 単数経路接続ネットワークのソケット間通信のスループットを測定したところ 330Mbps 程度の性能であった。一方、VPN ルータをはさみずには Initiator と Target 間を接続した場合、スループットは 400Mbps 程度であった。したがって VPN ルータの暗号化処理が通信のボトルネックとなっており、経路を複数にしたことにより、VPN ルータでの暗号化処理が分散され負荷が軽くなったと考えられる。

これに対し、2 つの回線を用いたにも関わらず、2 倍近い性能が得られていない理由を調べたところ、iSCSI はコネクションへのパケット転送をラウンドロビンによって振り分けているため、複数コネクション複数経路の場合でも 2 つの回線を 1 つずつ交互に使っているだけで、2 つの回線を同時には使っていないためであることがわかった。

また図 12 では、遅延時間を長くするとどの場合でも性能に変化がなくなっている様子がわかる。これは、遅延時間が短い時は複数経路にすることで、ルータでの処理が軽減され性能が向上したと考えられるが、高遅延環境にすると、経路の方がボトルネックとなり性能に差がなくなってきたためであると考えられる。

4.2 輻輳ウィンドウの比較

図 13, 14, 15 は TCP 輻輳ウィンドウをモニタした際の時間変化の様子である。この時のブロックサイズは 2MB, 片道遅延時間は 2msec に設定した。

図 13 は単数コネクション単数経路の場合に iSCSI シーケンシャルリードアクセス通信を行った時の輻輳ウィンドウをモニタした様子である。グラフ横軸に時間、縦軸に輻輳ウィンドウとエラーイベント番号を表示した。ここで ErrorNo.2 が Local device congestion(CWR エラー), 3 が重複 ACK, SACK を受信したこと, 4 がタイムアウトを検出したことを示す。図

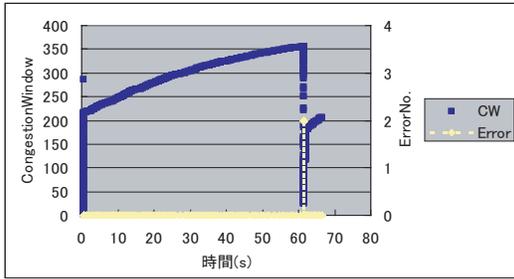


図 13 輻輳ウィンドウ (単数コネクション単数経路)

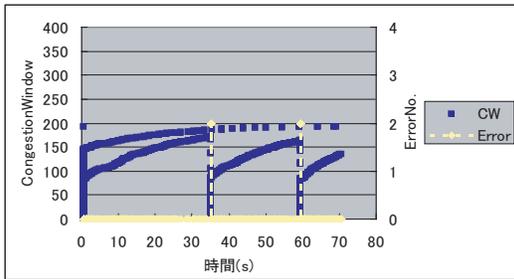


図 14 輻輳ウィンドウ (複数コネクション単数経路)

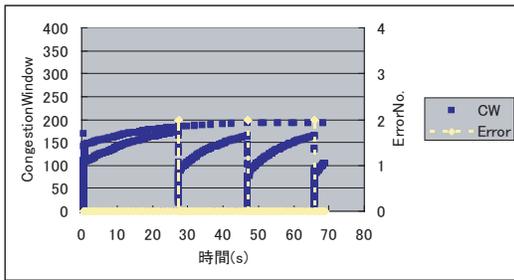


図 15 輻輳ウィンドウ (複数コネクション複数経路)

13 に示された ErrorNo.2 の縦の破線は Local device congestion(CWR エラー) が起こったことを表しており、これは送信側のデバイスドライバのバッファが溢れることによるエラーである。輻輳ウィンドウは約 350 パケットまで増加した後、CWR エラーが検出され急激に減少している。

図 14 は複数コネクション単数経路の場合の輻輳ウィンドウをモニタした様子である。ここで現れる ErrorNo.2 の縦の破線も CWR エラーが起きたことを示している。このとき、単数コネクションの場合のグラフと大きく変化している様子が分かる。ここで輻輳ウィンドウモニタツールは、2 つのコネクションのうち一方でも輻輳ウィンドウの値が変わったら表示され

るようになっている。このためそれぞれのコネクションとも独立で値が変わるため、ランダムにグラフ表示されることになる。すなわちコネクションごとの輻輳ウィンドウは、一方はなだらかに増加し一定になっており、もう一方は鋸型になっていると考えられる。

図 15 は複数コネクション複数経路の場合の輻輳ウィンドウをモニタした様子である。このときの ErrorNo.2 の縦の破線は CWR エラーを示しており、輻輳ウィンドウの変化は複数コネクション単数経路の場合とほぼ同じになった。

ここで複数コネクションにした場合、なだらかに増加するグラフと鋸型のグラフになった理由を考察する。2 つのコネクションのうちどちらか片方のコネクションが輻輳ウィンドウを使い切ったら ACK が返るまで次の iSCSI アクセス行われなくなる。iSCSI コネクションへのパケット振り分けはラウンドロビンで実行されているので、片方のアクセスが止まってしまうともう片方のアクセスも止まることになる。したがってもう一方のコネクションには輻輳ウィンドウ分を使い切る量のパケットが送られないので、Linux TCP 輻輳ウィンドウの特徴により片方の輻輳ウィンドウは一定となったと考えられる。

次に単数コネクションと複数コネクションの輻輳ウィンドウの違いを比較する。この 2 つのグラフを比べると、検出されるエラーの回数が複数コネクションの時の方が単数コネクションの時より多くなっている。また、複数コネクションのなだらかに増加するグラフと鋸型のグラフの輻輳ウィンドウを足し合わせると、最大で単数コネクションの時の輻輳ウィンドウの値である 350 パケットに近い値をとっていることが分かる。複数コネクションの場合、なだらかに増加するグラフと鋸型のグラフになるので、その両方の輻輳ウィンドウ分だけパケットは送信されることになる。したがって Target のデバイスドライバのバッファが溢れる頻度が高くなるものと考えられる。

5. 各経路の遅延時間による影響の評価

次に iSCSI 複数コネクション VPN 複数経路接続において異なる遅延時間をもつ経路を構築して実験を行った。

5.1 スループット測定結果

図 16 は経路ごとに遅延時間を変えていった時のスループット比較のグラフである。横軸は図 11 における Dummynet2 に設定した片道遅延時間、縦軸はスループットをとっている。また、グラフは上から図 11 における Dummynet1 に設定した片道遅延時間が 0、

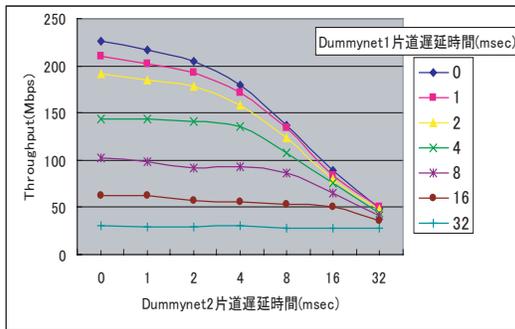


図 16 片道遅延時間とスループット

1, 2, 4, 8, 16, 32(msec) の場合である。

どのグラフも片道遅延時間を大きくするとスループットは減少しているが、Dummynet1, 2 の遅延時間が大きくなるにつれてその差は小さくなり、32msec の時にはほぼ一定の値をとっている。このように片方の遅延時間が大きくなると、もう一方の経路は十分早く通信できる状態でもスループットは低下してしまう。これは iSCSI の振り分けがラウンドロビンであることによる影響であり、パケットを交互に経路に送っているため、遅延時間の大きい経路のほうが通信全体のボトルネックとなってしまう。これにより経路の遅延時間が実行中に変化した場合も、例えば Dummynet1 の経路の遅延時間が実行中に大きくなった場合、それにつられて全体の通信性能は低減してしまう。

また、Dummynet1 の遅延時間 2msec、Dummynet2 の遅延時間 8msec(これを「遅延 2-8」と表す。以下同様)の時のスループットは 123Mbps であるが、遅延 8-2 の時は 91Mbps となっており、Dummynet1 の遅延時間の方が大きく影響している。同様に、遅延 4-16 と遅延 16-4 を比較しても遅延 16-4 の時の方がスループットが低くなっている。このことは全ての場合について同様であり、Dummynet1 の遅延時間の方がスループットに大きく影響しているということできる。これは VPN ルータの 2 経路への振り分けが全く対等ではなく優先順位などが付いていることによるものと考えられる。これについては次節で議論する。

5.2 輻輳ウィンドウの比較

次に輻輳ウィンドウの違いを比較する。図 17, 18, 19 はそれぞれ遅延 2-2, 遅延 2-16, 遅延 16-16 の場合である。

まず Dummynet1 の遅延時間は固定し、Dummynet2 の遅延時間を大きくしていった時の比較をする。図 17 と図 18 を比べると、図 18 の方が明らかに実行時間が長くなっている。また、輻輳ウィンドウの傾

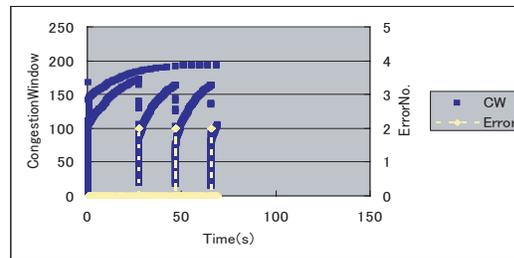


図 17 輻輳ウィンドウ (遅延 2-2)

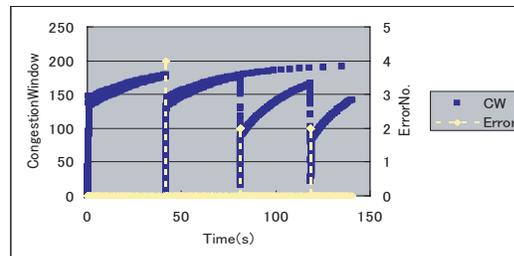


図 18 輻輳ウィンドウ (遅延 2-16)

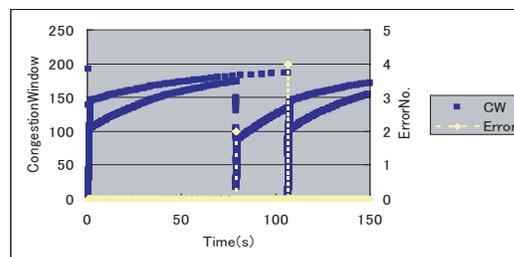


図 19 輻輳ウィンドウ (遅延 16-16)

きもなだらかになっている。さらにエラーに着目してみると、遅延 2-2(図 17) の時には ErrorNo.2 の Local device congestion による CWR エラーだけなのに対し、遅延 2-16(図 18) の時には ErrorNo.4 のタイムアウトを検出したことによるエラーが起きている。ここで ErrorNo.4 が検出されると輻輳ウィンドウは片方のグラフのみが減少するのではなく、両方の輻輳ウィンドウが減少しているのがわかる。

次に、Dummynet1 の遅延を大きくしていくと図 19 のようになる。ますます輻輳ウィンドウの傾きはなだらかになり、実行時間は長くなった。また、ErrorNo.4 のタイムアウトも度々起こるようになり、スループットも減少した。これらより、片方の遅延時間を長くするとそれに伴い両方のコネクションの輻輳ウィンドウが影響を受けており、もう片方には十分な通信帯域が残っているにも関わらず、それが有効に活用できてい

ない。

また、2つのコネクションのどちらがどのグラフであるかを確かめたところ、常に上側にあるグラフが Dummynet1 の方の経路を通るもので、下側にあるグラフが Dummynet2 の方の経路を通るものであることが分かった。これは VPN ルータの設定によるものである。VPN ルータでマルチルーティング機能を利用する場合、1つの VPN ルータから2つのあて先にパケットを振り分けるように設定をする。この設定の記述において、先に記述されたものが優先順位が高くなり、後に設定されたもののほうが優先順位が低くなる。この場合、Dummynet1 の方の経路を先に設定していたので優先順位が高く、そのためこちら側のコネクションは常に先に輻輳ウィンドウを確保することができる。

6. ま と め

本稿では、iSCSI ストレージアクセスにおいてコネクションを単数、複数に変化させ、VPN 接続も単数経路、複数経路に変化させた時のスループットの違いと輻輳ウィンドウの振舞いを観察し、比較した。また遅延時間の異なる経路を用いて iSCSI 複数コネクション VPN 複数経路を行ったときのスループットと輻輳ウィンドウの振舞いを比較した。

その結果、iSCSI 単数コネクション VPN 単数経路と iSCSI 複数コネクション VPN 単数経路を比較すると、特に高遅延環境においてはスループットはほとんど変わらないが、輻輳ウィンドウの振舞いが大きく変化している様子がわかった。これは iSCSI コネクションが複数になったため輻輳ウィンドウがコネクションごとに異なる振舞いをしているためであると考えられる。さらに、片方のコネクションにより iSCSI アクセスが止められた場合、もう一方のコネクションでは輻輳ウィンドウを使いきれずおよそ一定の値となっている。また、iSCSI 複数コネクション VPN 単数経路と iSCSI 複数コネクション VPN 複数経路を比較すると、スループットは複数コネクション複数経路の方が多少向上するが、輻輳ウィンドウはほとんど変化が見られない。これは VPN ルータでの暗号化処理が複数経路となったため負担が軽減されたものと考えられる。

また、高遅延環境においてはスループットはどの場合もほとんど変化がなくなることがわかった。これは遅延を入れたことで、経路によるボトルネックが問題となったためであると考えられる。

遅延時間の異なる経路を用いた実験では、片方の遅延時間が大きくなるとそれに引きずられてスループッ

トも低下してしまっていた。これも iSCSI がパケットの振り分けをラウンドロビンで行っているからであると考えられる。輻輳ウィンドウは遅延時間を大きくしていくと、Local device congestion だけではなくタイムアウトによってもエラーが起きていることが分かった。タイムアウトのエラーが起きた場合には、その影響が片方の経路だけにとどまらずもう一方の経路の輻輳ウィンドウも引き下げられている。

今後は経路によって遅延時間が異なる場合、iSCSI のパケットの振り分けを適切に制御し性能が向上するような通信を実現させていきたい。

参 考 文 献

- 1) iSCSI Specification ,
<http://www.ietf.org/rfc/rfc3720.txt?number=3270>
- 2) SCSI Specification ,
<http://www.danbbs.dk/~dino/SCSI/>
- 3) 山口実靖, 小口正人, 喜連川優: "高遅延広帯域ネットワーク環境下における iSCSI プロトコルを用いたシーケンシャルストレージアクセスの性能評価ならびにその性能向上手法に関する考察", 電子情報通信学会論文誌 Vol.J87-D-I, No.2, pp.216-231, 2004年2月
- 4) InterOperability Lab, Univ.of New Hampshire,
<http://www.iol.unh.edu/consortiums/iscsi/>
- 5) 千島 望, 豊田 真智子, 山口 実靖, 小口 正人: "VPN 接続環境における TCP パラメータと通信性能の相関関係評価", FIT2006, L-042, 2006年9月
- 6) 富士通 IP アクセスルータ GeoStream Si-R シリーズ GeoStream Si-R570 ,
<http://fenics.fujitsu.com/products/sir/sir570/index.ht ml>
- 7) 千島 望, 豊田 真智子, 山口 実靖, 小口 正人: "iSCSI アクセス時の VPN 環境における TCP 輻輳ウィンドウ制御手法の検討", DBWS2006, pp.709-712, 2006年7月
- 8) 千島 望, 豊田 真智子, 山口 実靖, 小口 正人: "iSCSI における TCP パラメータとアプリケーション実行性能の相関関係評価" 第 68 回情報処理学会全国大会, pp.131-132, 2006年3月
- 9) 豊田 真智子, 山口 実靖, 小口 正人: "iSCSI ストレージアクセスにおける TCP 輻輳ウィンドウコントロール手法の提案と性能評価", 電子情報通信学会論文誌 Vol.J90-D, No.2, pp.359-372, 2007年2月
- 10) 藤原 啓成, 若宮 直紀, 志賀 賢太: "広域 IP 網を介した iSCSI 通信におけるプロトコルチューニングの一検討", 第 68 回情報処理学会全国大会, pp.155-156, 2006年3月
- 11) L.Rizzo: "dummynet",
http://info.iet.unipi.it/~luigi/ip_dummynet/