

SCTP-CMT による複数経路伝送の実験

2008MI227 清水祐介

指導教員：後藤 邦夫

1 はじめに

近年、インターネット上で音声や動画配信が一般的になり、動画配信サービスが増え利用者も増加している。それに伴い一般的な回線速度は早くなったが、十分な帯域が確保できないことが問題となっている。特に単一経路によるデータ転送では、インターネット利用が多い時間帯に遅延やパケット損失などの影響がでてしまう可能性がある。そこで OSI 参照モデルのトランスポート層プロトコルである SCTP(Stream Control Transmission Protocol)[4] を使用することを提案する。また、SCTP に CMT(Concurrent Multipath Transfer)[3] という拡張機能を使用することで、複数経路でのデータ転送を可能にし、TCP よりも十分な帯域幅を得ることができる。

前述の SCTP-CMT を実装を実現している OS は 2012 年 1 月現在 FreeBSD[2] のみである。よって本実験での OS は FreeBSD Ver9.0 を用いる。

これまでの研究は、UDP と TCP を用いた研究が主であり、CMT-SCTP を用いた複数経路の実験は少ない。CMT-SCTP を用いることで、複数経路でのデータ転送が可能となり、本来 TCP でのデータ転送よりも多くのデータ転送を可能にする。よって本研究では Goto's IP Network Emulator(以下 GINE)[1] で仮想ネットワークを作成し、SCTP プロトコルでの複数経路実験を行う。そして、CMT を使用し SCTP を用いた場合 TCP よりもスループットが向上することを示す。

2 GINE の概要

本節では GINE の概要について説明する。GINE[1] とは南山大学の後藤邦夫教授が考案したネットワークエミュレータの一つである。GINE を使用することで、多数のルータやリンクから構築された様々なネットワークをエミュレートすることが可能である。各リンクに遅延やパケットロスなどの通信障害を設定することが可能で、バンド幅も自由に設定することができる。

3 SCTP について

SCTP は他のトランスポート層プロトコルより優れている機能がある。

3.1 SCTP におけるマルチホーミング

マルチホーミングとはサーバとクライアントとの通信路を多重化することで、いずれかの通信路が通信不能になった場合でも、ほかの通信路を介して通信を続行できる機能である。TCP コネクションは、クライアントとサーバのそれぞれの 1 つのインターフェース間で作成される。確立されたコネクションは、各インターフェースにバインドされる。それに対して SCTP は、2 つのネットワーク・インターフェースが含まれる。2 つのパスが独立

したネットワークを経由して提供されている。SCTP は、この 2 つのパスを 1 つのアソシエーションとしてまとめることができる。

3.2 SCTP におけるイニシエーション

SCTP の接続の開始は、4way ハンドシェイクによって行われる。TCP は通常 3way ハンドシェイクであるが、SCTP は COOKIE を使用することで DoS(Denial of Service) 攻撃を防ぐことができる。一方でトランスポート層の UDP プロトコルは明確なハンドシェイクを省いたコネクションレスであり、送達確認を行わない極めて信頼性の低いプロトコルである。以上のことから、本実験では信頼性の高い SCTP を用いる。

3.3 SCTP-CMT

SCTP には、CMT という拡張機能が存在する。通常の SCTP では、複数経路を保持しても、一つのパスがプライマリパスとなり同時に複数経路で転送することができない。しかし、CMT を使用することで、マルチホーミングの機能を生かしデータを複数伝送することができる。CMT には、必要以上に再送してしまうという SCTP の仕様によるデメリットも存在する。また、CMT には実装するために 5 つのポリシーがあり、FreeBSD では RTX-SSTHRSH という再送ポリシーに基づいて実装されている。SSTHRSH は再送パケットを、複数ある宛先の中で再送を行う際に最も大きな ssthresh (slow start threshold) の宛先へ送信するポリシーである。

4 実験

この節では、本実験における実験環境について説明する。

4.1 FreeBSD, GINE ネットワーク構成について

PC を 3 台用意する。FreeBSD 9.0 をインストールした PC を 2 台用意しそれぞれにクライアント、サーバの役割を持たせる。クライアント、サーバの間に GINE ネットワークを接続する(図 1 参照)。

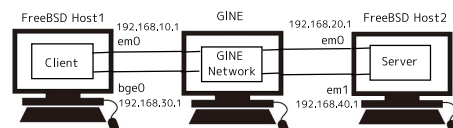


図 1 FreeBSD, GINE ネットワーク構成

4.2 GINE 内ネットワーク構成

次に GINE 内のネットワーク構成とアドレス設定を図 2 に示す。このネットワークは一台の PC で GINE ライブラリを用いて構築する。

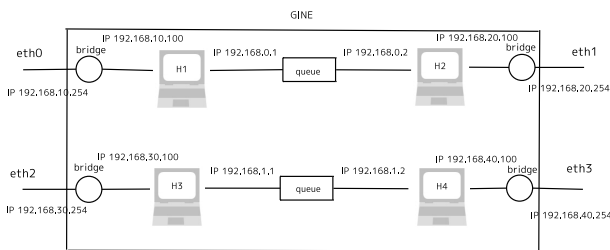


図 2 GINE ネットワーク構成

4.3 socket 通信プログラム

FreeBSD に対応して複数経路にデータを送信するプログラムを作成した。ソケットを作成するために使用したソケット関数は `sock = socket(AF_INET, SOCK_STREAM, IPPROTO_SCTP)` である。

4.4 帯域、遅延の設定について

ping で普段使用している yahoo ページの遅延を測定した。測定結果によりネットワーク遅延は 11ms であった。また普段使用している研究室 PC の帯域幅を iperf で測定する。測定結果により帯域幅は 113Mbps であった。以上より、帯域幅 113Mbps の経路を 2 経路用意し、その経路の往復遅延を 11ms とする。

4.5 スループット測定の実験手順

1. ネットワークエミュレータ GINE を起動する。
2. BSDhost1, BSDhost2 それぞれに GINE をネットワーク接続する。
3. FreeBSD-host2 サーバを起動し、ポート番号を設定する。
4. FreeBSD-host1 クライアントを起動する。IP アドレス、ポート番号、送るデータ量をコマンドで設定する。
5. TCP, SCTP, SCTP-CMT それぞれのプロトコルでデータを流す。SCTP-CMT の場合 CMT 機能をコマンド入力 ON にする。
6. 遅延なしの場合、片道経路のみ遅延を発生する場合、両方の経路で遅延を発生する場合の送れたデータ量を比較する。

5 実験結果・考察

この節では本実験における実験結果、考察を説明する。

5.1 実験結果

実験の結果、表 1 のようになった。遅延なしの場合と遅延ありの場合を比較すると遅延ありの場合の方が送信データ量は少なく、遅延ありと片道遅延を比較すると片道遅延の方がスループットが大きくなった。

5.2 考察

すべての実験環境の場合で SCTP より TCP が多くのデータ量を送信した。SCTP パケット構造は、パケット全

表 1 実験結果

実験結果	遅延なし	遅延あり	片道遅延
TCP	93.539Mbps	93.077Mbps	93.486Mbps
SCTP	91.867Mbps	91.691Mbps	91.780Mbps
SCTP-CMT	144.969Mbps	133.831Mbps	134.781Mbps

体についての情報と送信元および宛先のポートなどを伝える common header と、任意の数 chunk(パケットやウィンドウの代わりとなるもの) からなる。また、SCTP は 1 つのパケットの中に複数の chunk が含まれることがある。TCP の header 長は 20byte であるのに対し、SCTP の header 長は common header の 12byte と、data chunk を足した 28byte である。そのため、SCTP の方がペイロードが短くなり、TCP が多くのデータ量を送信したと考えられる。また、すべての実験環境で SCTP-CMT は 100Mbps 以上のデータ転送に成功した。単一経路では 100Mbps しかデータ転送できないため、複数経路を通して 100Mbps 以上のデータを送信していることがわかる。遅延を発生させた場合、片道遅延の場合も遅延なしの場合よりはデータ送信量は劣るが 100Mbps 以上のデータ送信が出来、実験は成功といえる。SCTP-CMT を使用することにより、従来の TCP, SCTP よりも多くのデータ送信が可能となることがわかった。

6 おわりに

本研究では単一経路でのデータ転送で通常 100Mbps しか送信できないデータを、複数経路で 100Mbps 以上のデータを送信した。当初ユーザ空間ライブラリを用いての実験を目的としていたが、実験環境が整わなかったため、FreeBSD を用いて SCTP-CMT の複数経路伝送の実験を行った。単一経路でのコネクションでは TCP の方がデータ転送速度は早かったが複数経路伝送でデータ転送した場合、一つの経路で障害が起きた場合は SCTP の方がデータ転送速度が早かった。DoS 攻撃にも強く安全かつデータを早く転送できる SCTP が今後トランスポート層の中でも重要なプロトコルになると考える。

参考文献

- [1] Goto, K.: Network Emulator with Virtual Host and Packet Diversion, *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT)*, Vol. 3, No. 3 (2012).
- [2] Iyengar, J. R.: CMT Implementation in BSD (accessed Sep. 2012). <http://www.eecis.udel.edu/~iyengar/dissertation/ch5-implementation.pdf>.
- [3] Iyengar, J. R., Shah, K. C. and Amer, P. D.: Concurrent Multipath Transfer Using SCTP Multihoming (accessed Sep. 2012). <http://citeseerx.ist.psu.edu>.
- [4] Stewart, R.: Stream Control Transmission Protocol (Sep. 2007). RFC4960.