

# PC上で動作するスケーラブルなIPネットワーク実験システム

登大遊<sup>\*†</sup> 加藤和彦<sup>\*†</sup> 新城靖<sup>\*†</sup> 板野肯三<sup>\*†</sup> 佐藤聡<sup>\*</sup> 中井央<sup>\*</sup>

<sup>\*</sup> 筑波大学 <sup>†</sup> 科学技術振興機構

## 1. はじめに

インターネット上で動作する通信ソフトウェアを開発する場合、動作検証などのため、複数のルータによって構成されたIPネットワーク内で通信試験を行いたい場合がある。そのような実験システムを実際の多数のルータやスイッチで構成するのはコストがかかるので、計算機上で仮想的に構築したいという要求がある。

本研究では、ソフトウェアによって数千個のルータとスイッチを計算機上に作成して相互に接続し、複雑なIPネットワークトポロジを構築して稼働させるための実験システムを提案する。本実験システムは、中心となるIPネットワークと周辺アプリケーションノードにより構成される。IPネットワークは、仮想ルータとリンクから構成される。トポロジの設計はGUIによってマウス操作で絵を描くように行うことができる。トポロジを分割して複数台の計算機に分散させて稼働することもできる。構築したIPネットワーク内の任意のルータに、外部からアプリケーションノードを接続することを可能にする。また、任意のルータ間で通信遅延やパケットロスが発生させることにより、インターネット上でこれらの現象が生じている場合の動作をシミュレートすることもできる。本実験システムにより、高価なネットワーク回線や機器を多数用意することなく、スケーラブルなIPネットワークを構築できるので、インターネット上で動作するソフトウェアの開発などに利用できる。

## 2. 設計

本システムを以下のように設計する。

### 2.1. 仮想ルータ

仮想ルータは複数のインターフェイスを持つIPパケットのルーティング機能を持ったモジュールである。各インターフェイスにはIPアドレスが割り当てられ、仮想ルータが保持しているルーティングテーブルにより、入力されたIPパケットを適切なインターフェイスの先にあるネットワークにルーティングする。仮想ルータはソフトウェアで実現されるので、システムを稼働させるPCのハードウェア性能の上限に達するまで、複数個作成することができる。

### 2.2. 仮想ルータ間のリンク

仮想ルータのインターフェイス間をリンクにより

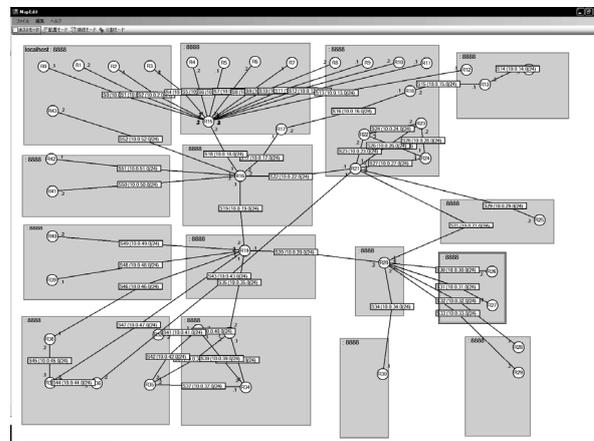


図1 GUIのトポロジ作成支援ツール

接続することで、インターネットのようなIPネットワークを作成する。通常、ルータのインターフェイス同士を接続する場合、2個のルータであればポイントツーポイント接続、3個以上のルータであればHUB(スイッチ)等のネットワークセグメントを用いた接続を行う。本システムでは簡略化のため、複数の仮想ルータのインターフェイスが相互リンクする場合は、そこに仮想HUBを1つ作成してセグメントを設ける。

### 2.3. ルーティングテーブル

本システム内の仮想ルータは、スタティックルーティングによって経路制御される。OSPFやBGPなどのダイナミックルーティングは使用しない。複数個の仮想ルータによってトポロジが構築されると、ダイクストラ法により、各ルータから自分以外の全ルータの持つインターフェイスの属するIPネットワークに対するルーティングテーブルが計算される。

### 2.4. アプリケーションノードからの接続

本システムによって構成されるIPネットワークには、アプリケーションノードとなる計算機(実計算機またはVM)をつなぎ込んで通信させることができる。アプリケーションノード上では、汎用のOSを使用できる。各OS上には、仮想ネットワークインターフェイス(NIC)を作成する。仮想NICと仮想ルータのインターフェイスとの間をVPNによって接続する。仮想NICにはその仮想ルータのインターフェイスの属するIPネットワークのIPアドレスと、デフォルトゲートウェイとして仮想ルータのインターフェイスの持つIPアドレスを指定する。これにより、本システムを用いて、既存のOSやその上で動作する通信ソフトウェアを修正することなく、実験を行うことが可能である。

### 2.5. 通信遅延およびパケットロスの発生

実験者が指定した仮想ルータの仮想インターフェ

## Scalable Testbed of IP Networks Running on PCs

Daiyuu Nobori<sup>\*†</sup>, Kazuhiko Kato<sup>\*†</sup>, Yasushi Shinjo<sup>\*†</sup>,  
Kozo Itano<sup>\*†</sup>, Akira Sato<sup>\*</sup> and Hisashi Nakai<sup>\*</sup>

<sup>\*</sup> University of Tsukuba

<sup>†</sup> Japan Science and Technology Agency

イスにおいて、通信遅延とパケットロスの発生を行わせることができる。通信遅延は 10 ミリ秒単位で、パケットロスは 0~100%の間で 1%単位で指定することができる。これにより、インターネットなどで頻繁に生じる遅延の発生や品質の悪い回線におけるパケットロスを再現し、そのような環境上で通信ソフトウェアのテストを行うことができる。

## 2.6. トラフィック量のモニタとトラフィックデータの複製

本システム内の指定した仮想ルータのインターフェイスを流れるトラフィック量(データ量およびパケット数)をモニタすることができる。また、指定した仮想ルータを流れるすべてのパケットを複製し、実験者の持つ PC 上に作成した仮想 NIC で取り出して、Tcpdump 等の一般的なパケット解析ソフトウェアや侵入検知システム (IDS) 等に入力することが可能である。

## 2.7. ネットワークトポロジの定義

本システムを稼働させるために、実験者は作成したい IP ネットワークトポロジを与える必要がある。トポロジの定義には、設定ファイルを外部のプログラムで作成する方法と、本システムの GUI のトポロジ作成支援シールによって、仮想ルータを配置してその間を線でつなぐ操作をグラフィカルに行う方法の 2 通りが利用できる。GUI のトポロジ作成支援ツールは Windows 上で動作する (図 1)。

## 2.8. スケーラブルな IP ネットワークの構築

仮想ルータはソフトウェアモジュールであるため、実験システムを稼働させる PC 上に、メモリの空き容量がある限り複数個作成することができる。しかし、多数の仮想ルータとそれらが連なった IP ネットワークトポロジが 1 台の PC 上に形成された場合、通信パケットのルーティング処理が CPU 時間を消費するため、動作が遅くなる。また、メモリ上限までしか仮想ルータを作成することができなくなる。

そこで、本システムの稼働のために利用可能な PC が複数台ある場合、実験者が定義したトポロジを複数のエリアに分割し、それぞれのエリアを各 PC に割り当てることにより、負荷分散を実現することができるようにする (図 1)。

## 3. 実装

### 3.1. IP ネットワーク

本システムの IP ネットワーク部分は、Windows、Linux、FreeBSD、Solaris および Mac OS X 上で、各 OS 間で機能に差異無く実装する。仮想ルータおよび仮想ルータのインタフェース間を結合するための仮想 HUB を実装するため、PacketiX VPN[1] のサーバーソフトウェアを拡張する形で実装する。

### 3.2. アプリケーションノード

IP ネットワークに参加するアプリケーションノードで使用する仮想 NIC は、Windows および Linux に対応する。仮想 NIC は PacketiX VPN のクライア

ントソフトウェアを拡張する形で実装する。

### 3.3. GUI のトポロジ作成支援シール

GUI のトポロジ作成支援シールは、.NET Framework 2.0 上で動作する Windows アプリケーションとして記述する。

## 4. 関連研究

StarBED[2] は多数の PC とそれらが接続されたレイヤ 3 スイッチで構成された、大規模なインターネットシミュレータである。制御は SpringOS によって自動化される。実験者が定義したトポロジに合わせてレイヤ 3 スイッチ内の論理ルータを作成したり、各ルータ間と PC との間で VLAN を構成したりすることにより、多数のルータによる IP ネットワークを構成することができる。本研究では、そのような IP ネットワークをソフトウェアで低いコストにより実現できる。

NIST Net[3] は通信遅延やパケットロスを発生させるための汎用的な PC で実行可能なソフトウェアである。PC の 2 つの NIC 間でパケットを転送する際、遅延やパケットロスを発生させることができる。本研究の仮想ルータで実現する機能として、同様のものを組み込む。

## 5. まとめ

本システムは現在実装途中である。現在、通信遅延およびパケットロスの発生のためのプログラミングを行っており、その他の部分はほぼ実装を終えた。

本システムのプログラミングが完了した後、詳細な動作検証を行うとともに、ドキュメントなどを整備する。本システムには一部著者が開発した市販ソフトウェアのコンポーネントが使用されているが、この部分の権利処理を行った後、インターネット上でフリーウェアとして公開・配布する予定である。

複数の仮想ルータが 1 台の PC の CPU 上でソフトウェアとして実行されている以上、実際の複数の物理的なルータによって構成されているインターネットなどの IP ネットワークトポロジと比較すると、負荷がかかったときにシステム全体で通信速度が低下したり遅延が生じたりする場合がある。このような、通信実験を行う際の支障を軽減させることが今後の課題である。

### 参考文献

- [1] 登 大遊, "SoftEther の内部構造", 情報処理学会誌「情報処理」, Vol.45, No.10, pp.1057-1062, 2004.
- [2] Toshiyuki Miyachi, Ken-ichi Chinen and Yoichi Shinoda, "StarBED and SpringOS: Large-scale General Purpose Network Testbed and Supporting Software," International Conference on Performance Evaluation Methodologies and Tools (Valuetools) 2006, Oct. 2006.
- [3] M. Carson and D. Santay, "NIST Net: A Linux-based Network Emulation Tool", ACM SIGCOMM Computer Communications Review, 33(3): pp.111-126, 2003.