Windowsにおける大規模分散システムテストベッドの開発

新岡寛幸†　佐藤晴彦†　栗原正仁†
北海道大学　情報科学研究科†

1．はじめに

インターネットの普及，高速化により，インターネットを利用したサービスやシステムが数多く提案されてきている．これらのシステムは，インターネットを介して多数の計算機が通信を行い，協調して一連の処理を行っている．以下，本稿ではこのようなシステムを分散システムと呼ぶ．分散システムの動作検証は，多数の計算機が通信を正常に行っていることを検証しなければならないため，容易ではない．また，インターネット上での動作検証は，既存のシステムに影響を与える可能性があるため，安易に行うことはできない．

そのため，分散システムの動作検証を他のシステムに影響を与えることなく，容易に行う手法が提案されてきている．本稿では，IP 通信をエミュレートすることで，Windows 上で動作する分散システムの動作検証を可能にするミドルウェアを提案する．提案ミドルウェアは，既存のプログラムに修正を加えることなく動作検証が可能である．そのため，提案ミドルウェアを使用し，動作検証を行ったプログラムは，修正を加えることなく実環境上で動作させることが可能となっている．既存の手法では，検証用プログラムを利用する際に，プログラムの修正を必要とせずに，実行可能なものは存在するが，実装が OS 依存のため，Windows 環境で動作するものは存在しない．

提案ミドルウェアは，検証対象とする各プロセスに仮想 IP アドレスを設定することで，各プロセス同士は仮想 IP アドレスを用いて通信を行うことができる．提案ミドルウェアを利用することで，一台の計算機上で多数の IP アドレスを必要とする分散システムの動作検証が可能となる．また，複数台の計算機を利用することでより大規模な分散システムの動作検証を可能にする．

Development of Large Scale Distributed System Test Bed for Windows
†Hiroyuki Niioka, Haruhiko Sato and Masahito Kurihara
†Graduate School of Information Science and Technology Hokkaido University

2．関連研究

既存の分散システムの検証手法について説明する．

実際に多数の計算機を使用してネットワークを構築し，そのネットワーク上で分散システムを動作させるという手法が存在する．この手法は，実際のインターネットに近い環境で動作検証を行うことができる．しかし，多数の計算機を用意することは困難である．また，PlanetLab[2] や Netbed[3]，StarBED[4] 等のテストベッドは利用申請等の利用条件が存在する場合もあり，容易に利用することができないといった欠点が存在する．

仮想マシンモニタと呼ばれるプログラム [7][8] を利用する動作検証は，一台の計算機で実際のインターネットに近い環境で実験を行うことができる．しかし，仮想マシンを実現するためのオーバーヘッドが大きいため，多数の仮想マシンを実現することはできない．また，各仮想マシンに割り当てられる仮想 IP アドレスはユーザーが容易に変更することができないといった問題点が存在する．

ネットワーク環境をシミュレート，または，エミュレートするプログラム [1][5][6] は，一台の計算機上で自由にネットワーク環境を構築し，そのネットワーク上で動作検証を行うことができる．しかし，シミュレーションプログラム独自の API やコードを必要とする場合があり，利用するには，プログラムの修正が必要になる．プログラムを修正せずに動作させることを可能にするものは，実装が OS 依存であり，Windows 環境で動作するものは存在しない．

3．提案ミドルウェア

本ミドルウェアは，IP アドレスとポート番号を仮想化することで，仮想 IP アドレスと仮想ポート番号を用いた通信を可能にする．それにより，複数台の計算機上で各プロセスに異なる仮想 IP アドレスを割り当て，通信を行うことを可能にしている．

本ミドルウェアは，Windows OS の Layered

Service Provider (LSP) という機構を利用することで，通信に関する API のフックを行っている．これにより，プロセスが通信に関する API をコールし，実際に実行される前後に，任意の処理を行うことが可能になる．

具体的には，IP アドレスやポート番号を引数として持つ API がコールされた時に，API の実行前では，IP アドレスやポート番号の書き換えを行い，実行後には引数を元に戻すという処理を行う．この処理は，プロセスから渡される引数の値は仮想 IP アドレス，仮想ポート番号であるため，API 実行前に実 IP アドレス，実ポート番号に書き換え，実行後に元に戻す，という処理を行っている．この仮想化処理を適切に行うことで，プロセスが受け取る IP アドレス，ポート番号は全て仮想的なものになるため，プロセスから渡される引数の値は全て仮想的なものになる．

本ミドルウェアは，デバッグ用途として，通信に関する API がコールされた際に，API に渡された引数を出力する機能を提供する．これにより，利用者は通信が正常に行われているかを確認することができる．

本ミドルウェアの評価実験として，単純な通信プログラムを用いて実験を行った．仮想化処理が必要となる API をコールする通信を，500,000 回ループさせ，動作時間を計測した．実環境上とミドルウェア上での動作時間を比較した結果，ミドルウェアにより約 1.34 倍のオーバーヘッドが発生していることを確認した．本実験は一台の計算機上で行った．

4. まとめ

Windows OS 上で動作し，プログラムの修正を必要とせずに，分散システムの動作検証を可能にするミドルウェアを提案した．このミドルウェアは一台の計算機上でも多数のプロセスの動作検証を可能にするが，複数台の計算機を利用することでより大規模な動作検証を可能にする．通信に関する API に渡される引数を出力する機能を有し，プロセス内部の振る舞いを確認することができる．評価実験により，オーバーヘッドは実際の使用に耐えうる範囲に収まっていることを確認した．

5. 今後の課題・展望

今後の課題は，ノンブロッキングモードに設定されているソケットでの通信がタイムアウトする場合があるため，ノンブロッキングモードでの通信に対応することで，より完全なエミュレーションを実現することである．

また，デバッグ機能として，エラーが発生したプロセスに対してデバッガーを起動するような機能や，動作検証の支援機能として，各種ログを出力し分散システムの振る舞いを検証できるような機能を提供したいと考えている．

複数の計算機で提案ミドルウェアを利用した際のオーバーヘッドを測定することで，提案ミドルウェアの評価を行う予定である．

提案ミドルウェアは Windows 環境で動作するものであるが，他の OS で動作する同様の機能を持ったプログラムと協調動作させることで，複数の OS を利用した分散システムの動作検証を可能にすることを目標としている．

参考文献

[1] 西川賀樹，大山恵弘，米澤明憲：プロセスレベルの仮想化を用いた大規模分散システムテストベッド，情報処理学会論文誌コンピューティングシステム (ACS)，Vol. 1, No. 2 (2008), pp. 144-156.

[2] B. Chun, D. Culler, T. Roscoe, A. Bavier, L. Peterson, M. Wawrzoniak, and M. Bowman: "PlanetLab: An Overlay Testbed for Broad-CoverageServices". ACM SIGCOMM Computer Communication Review, Vol. 33, No. 3 (2003), pp. 3-12.

[3] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar: "An integrated experimental environment for distributed systems and networks". OSDI'02: Proceedings of the 5th symposium on Operating systems designe and implementation, (2002), pp. 255-270.

[4] T. Miyachi, K. Chinen, and Y. Shinoda: "StarBED and SpringOS: large-scale general pur-pose network testbed and supporting software". Proceedings of the 1st international conference on Performance evaluation methodologies and tools, Pisa, Italy, (2006).

[5] ns-2. http://www.isi.edu/nsnam/ns/

[6] L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia, and M. Gerla: "GloMoSim: A Scalable Network Simulation Environment". UCLA ComputerScience Department Technical Report, Vol. 990027, (1999).

[7] Xen. http://www.xen.org/

[8] VMware. http://www.vmware.com/