

MRTG を用いた地震観測網の監視

植平賢司^{*†}

The Seismic Network Monitoring System Using MRTG Software

Kenji UEHIRA^{*†}

Abstract

The objective of this paper is to describe the seismic network monitoring system using the Multi Router Traffic Grapher (MRTG) and the related software. MRTG has function monitors Simple Network Management Protocol (SNMP) network devices and gives illustration the traffic amount has through each interface. The mrtg-ping-probe is a ping probe for MRTG, which implementation to monitor the round trip time (RTT) and packet loss to seismic stations or servers. MRTG outputs function to generate graphs visualizing minimum and maximum RTT or packet loss. The seismic network monitoring system is developed using such software, and monitors the data traffic of the data acquisition servers and the IP line situations (i.e. RTT and packet loss) between the data center side and the observation station side end. In this paper, I describe concrete examples from actual operations, and show that this system is useful for maintenance of the seismographic network.

Key words : MRTG, SNMP, ping, RTT, packet loss, seismic data transmission, network monitoring

はじめに

九州大学地震火山観測研究センターでは、定常及び臨時を含め 40 点強の観測点のデータをテレメータ方式によりリアルタイムで取得している。テレメータは専用回線、衛星回線、無線などを使って行われてきたが、NTT のフレッツ回線に代表されるような高速で安価な IP 接続可能な回線が一般に普及した事により、2002 年より順次テレメータ方式を切り替えていき、2006 年にはそのほとんどが IP 通信に置き換わった。また、センター内でのデータ通信や、他機関とのデータ交換においてはこれまでも IP 通信を使って行われてきた。

観測データの欠測には様々な要因が考えられ、実際の欠測の時には複数の要因が重なってしまう事が少なくない。そのような欠測の時、どの部分に障害があるのか切り分けていき、悪い部分を一つ一つ潰していかななくてはならない。九州大学ではフリーのソフトウェアである MRTG (Multi Router Traffic Grapher) で IP ネットワークの監視を行

うことにより、障害が起こった時の原因の切り分け作業の一助としているので報告する。

MRTGによって、主に二つの情報を監視している。データ処理をしているマシンのネットワークカードのトラフィック量と、IP 接続されている観測点（フレッツ回線網及び専用回線）への往復遅延時間（Round Trip Time : 以後 RTT と略す）とパケット損失率の計測により得られる回線の障害状況の情報である。前者は SNMP (Simple Network Management Protocol) を用いて、後者は ping を用いて計測する。

MRTG とは

MRTGとはMulti Router Traffic Grapherの略で、ネットワークの負荷を監視するツールであり、GNU General Public License¹の下で無償配付されている。元々は、名前の通りルータのトラフィックを監視するためのツールとして開発されたが、SNMPというルータやコンピュータ、端末など、ネットワークに接続された通信機器をネットワーク経由で監視・制御するためのプロトコルを用いて通信しているため、SNMPが使えればルータに限らず一般のサーバやスイッチなどの情報も取得出来る。また、外部プログ

2008 年 8 月 15 日受付, 2008 年 10 月 30 日受理.

[†] uehira@sevo.kyusyu-u.ac.jp

^{*}九州大学大学院理学研究院附属地震火山観測研究センター.

^{*}Institute of Seismology and Volcanology, Faculty of Sciences, Kyushu University.

¹ <http://www.gnu.org/copyleft/gpl.html>

ラムにより SNMP 以外の手段による情報取得も可能になっている。MRTG は、現在のネットワークのトラフィックの状態を示すグラフィックイメージを含む HTML ページを生成する。MRTG は Perl と C 言語で記述されており、Unix と Windows NT で動作する。MRTG は、SNMP を使用してトラフィックカウンターを読み取る Perl スクリプトと、トラフィックデータを収集して監視しているネットワークのトラフィックをグラフにする高速な実行形式のプログラムで構成されている。トラフィックデータの監視だけでなく、あらゆる SNMP 変数を監視することが可能なので、システムの負荷やログインセッション、ディスクの空き状況などもグラフ化可能である。また、mrtg-ping-probe というツールを使う事により、ping コマンドによる RTT とパケット損失率の計測結果をグラフ化することが出来る。

MRTG は、過去 1 日間強、1 週間強、1 ヶ月間強、1 年間強のトラフィックをグラフ化する。それぞれ、トラフィックの 5 分間、30 分間、2 時間、1 日間の平均値をプロットするようになっているので、通常はデータを 5 分毎に定期的に取得するよう crontabなどで設定する。

ソフトウェアのインストールと設定

監視システムを作るためには表 1・表 2 に示したソフトウェアをインストールする事が必要である。

情報を提供するマシンに必要なソフトウェアは、net-snmp である (表 1)。これは、ネットワークのトラフィッ

ク量の情報を提供するための SNMP サーバであり、通常 snmpd という daemon を常駐させる。ping に対しては通常設定は必要ない。

情報を収集するマシンに必要なものを表 2 に示す。まず必要なのは MRTG 本体である。MRTG が依存するライブラリもあるのでこれらも予めインストールする必要がある。ping による RTT とパケット損失率を表示するためには、mrtg-ping-probe をインストールする。また、他のマシンから結果を参照する場合は apache などの http サーバを動かす必要がある (ただし、本論文では http サーバの設定については触れない)。

1. 情報提供マシンへのインストールと設定

情報提供マシンのセットアップの流れを図 1 に示す。このマシンには net-snmp が必要である。net-snmp は SNMP のサーバプログラムである。このソフトウェアは、ネットワークのトラフィック量を調べたいマシンそれぞれにインストールして動かす必要がある。コンパイルのためには、C コンパイラと perl が必要である。最近の Unix システムではすでにインストール済みであるものが多いので、通常はインストールの必要はないであろう。

設定は snmpd.conf に記述する。この設定ファイルの格納されるディレクトリはインストール時の設定によって異なる。ちなみに、FreeBSD の ports システムでインストールした場合、/usr/local/share/snmp/ である。この設定ファイルでは、SNMP サーバの認証関係などの設定を行う。

表 1. 情報提供マシンに必要なソフトウェア

ソフトウェア	説明	URL
net-snmp	SNMP サーバプログラム	http://net-snmp.sourceforge.net/

表 2. 情報収集マシンに必要なソフトウェア

ソフトウェア名	説明	URL
mrtg-ping-probe	ping による情報収集ソフトウェア。mrtg に依存。	http://pwo.de/projects/mrtg/
mrtg	MRTG 本体。gd, libpng, zlib に依存。	http://oss.oetiker.ch/mrtg/
gd	Thomas Boutell によって開発されたグラフを描くための基本的なライブラリ。	http://www.libgd.org/
libpng	PNG ファイルを生成するためのライブラリ。	http://www.libpng.org/
zlib	libpng が生成したグラフィックを圧縮するためのライブラリ。	http://www.zlib.net/
apache	http サーバ。apache 以外の http サーバでも可。	http://www.apache.org/

SNMP には大きく分けて3つのバージョンがある。すなわち、SNMPv1, SNMPv2, SNMPv3である。v1とv2では主にコミュニティ名によって認証を行いデータの送受信を行う。設定例を図2に示す。rocommunityの行が認証部分の設定であり、この場合sevo-snmpがコミュニティ名、192.168.8.0/24がアクセス可能なネットワークアドレスである。その他、SNMPサーバの設置場所や、管理者のメールアドレスなどが設定されている。

v3はv1やv2と比べてセキュリティが強化されたバージョンで、認証方式が大きく変わり設定の仕方も大きく異なっている。snmpconfという設定ファイル作成ツールも同時にインストールされているので、v3を使いたい場合や、v1やv2でもっと詳細な設定を行いたい場合はこれを

活用してみれば良いであろう。

設定が終了したら、SNMPサーバであるsnmpdデーモンをrootユーザで次のようにして立ち上げる。

```
# /usr/local/sbin/snmpd -p /var/run/snmpd.pid
-c /usr/local/share/snmp/snmpd.conf -a
```

また、リブート時に自動的に立ち上がるように設定を行う。この設定はOSによって異なるので、それぞれのOSの設定例を参考にされたい。

pingに対しては、通常何もしなくても返事を返すので、設定の必要はない。Firewallを使っているのであれば、ICMPパケットを遮断してないかどうかだけ確認すれば良いであろう。

2. 情報収集マシンへのインストール

情報収集マシンでのセットアップの流れを図3に示す。

MRTGとmrtg-ping-probeが情報を収集するマシンに必要であり、通常ある一つのマシンにインストールすれば良い。また、同じマシンでhttpサーバを動かし、他のマシンからも情報を閲覧出来るようにするのが一般的である。

MRTGのコンパイルのためには、Cコンパイラとperlが必要である。また、表2に示したいくつかのライブラリが必要なので、これらを予めインストールする事が必要となる。この他、SNMP通信のためのperlのモジュール等が

```
# syslocation: The [typically physical] location of the system.
syslocation "Server Room"

# syscontact: The contact information for the administrator
syscontact HogeHoge@sevo.kyushu-u.ac.jp

# rocommunity: a SNMPv1/SNMPv2c read-only access community name
rocommunity sevo-snmp "192.168.8.0/24"
```

図2. net-snmpのsnmpdの設定ファイルsnmpd.confの記述例

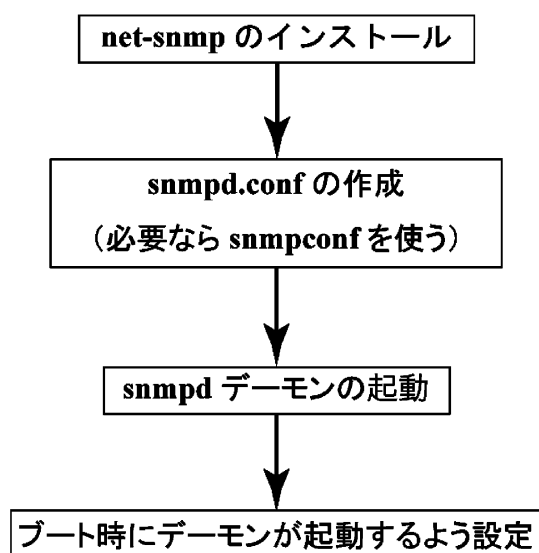


図1. 情報提供マシンでのセットアップの流れ

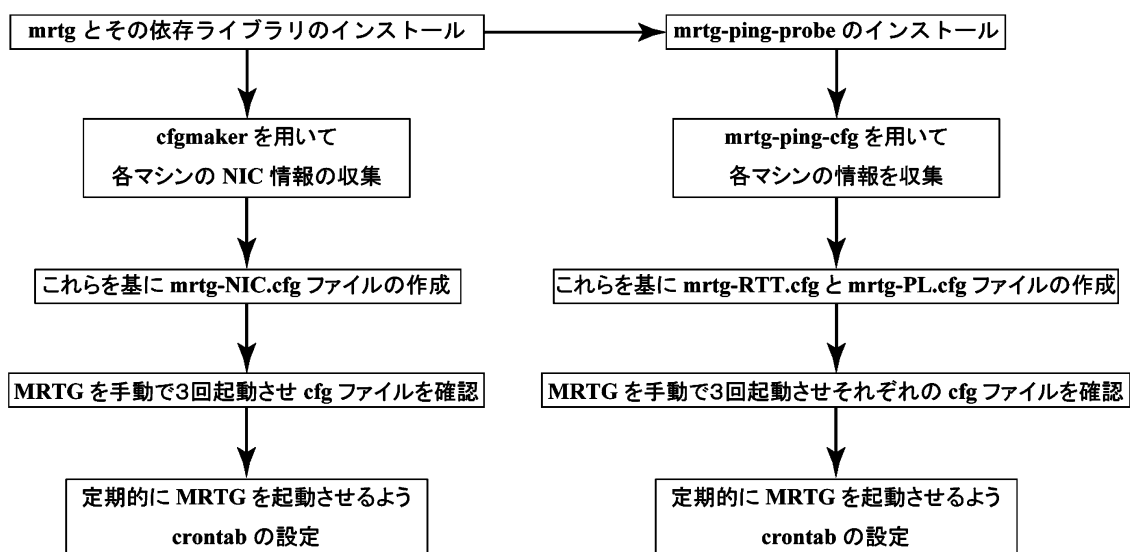


図3. 情報収集マシンでのセットアップの流れ

必要になる場合があるかもしれない。

mrtg-ping-probe の必要なライブラリやソフトは MRTG と同じであり、MRTG に依存しているので、MRTG のインストール後にこれのインストールを行う。

Linux や FreeBSD では rpm や ports システムなどがあり、依存関係のソフトウェアも自動認識してインストールしてくれるので、こういうシステムが使えるのならこち

らを使う方が楽であろう。

3. 情報収集マシンでの設定

1) ネットワークトラフィック量収集のための設定
ここでは、各マシンのネットワークカード（以後、NIC と略す）でどの程度のデータ量が流れているのかという情報を収集する方法を述べる。

各マシンの NIC の情報を集めるため、付属の cfgmaker

```
% cfgmaker sevo-snmp@purin
# Created by
# /usr/local/bin/cfgmaker sevo-snmp@purin

### Global Config Options
# for UNIX
# WorkDir: /home/http/mrtg
# or for NT
# WorkDir: c:\mrtgdata

### Global Defaults
# to get bits instead of bytes and graphs growing to the right
# Options[_]: growright, bits

EnableIPv6: no

#####
# System: purin.sevo.kyushu-u.ac.jp
# Description: FreeBSD purin.sevo.kyushu-u.ac.jp 6.3-RELEASE-p2 FreeBSD 6.3-RELEASE-p2
#0: Thu Apr 17 14:01:53 JST
2008 HogeHoge@purin.sevo.kyushu-u.ac.jp/amanda/obj/usr/src/sys/PURIN i386
# Contact: HogeHoge@sevo.kyushu-u.ac.jp
# Location: "HogeHoge's Room"
#####

### Interface 1 >> Descr: 'em0' | Name: '' | Ip: '192.168.8.39' | Eth: '00-07-e9-09-5f-20' ###

Target[purin_1]: 1:sevo-snmp@purin:
SetEnv[purin_1]: MRTG_INT_IP="192.168.8.39" MRTG_INT_DESCR="em0"
MaxBytes[purin_1]: 125000000
Title[purin_1]: Traffic Analysis for 1 -- purin.sevo.kyushu-u.ac.jp
PageTop[purin_1]: <h1>Traffic Analysis for 1 -- purin.sevo.kyushu-u.ac.jp</h1>
<div id="sysdetails">
<table>
<tr>
<td>System:</td>
<td>purin.sevo.kyushu-u.ac.jp in</td>
</tr>
<tr>
<td>Maintainer:</td>
<td></td>
</tr>
<tr>
<td>HogeHoge@sevo.kyushu-u.ac.jp</td>
<td></td>
</tr>
<tr>
<td>Description:</td>
<td>em0</td>
</tr>
<tr>
<td>ifType:</td>
<td>ethernetCsmacd (6)</td>
</tr>
<tr>
<td>ifName:</td>
<td></td>
</tr>
<tr>
<td>Max Speed:</td>
<td>125.0 MBytes/s</td>
</tr>
<tr>
<td>Ip:</td>
<td>192.168.8.39</td>
</tr>
</table>
</div>

### Interface 2 >> Descr: 'em1' | Name: '' | Ip: '192.168.0.39' | Eth: '00-30-48-71-02-bb' ###

Target[purin_2]: 2:sevo-snmp@purin:
SetEnv[purin_2]: MRTG_INT_IP="192.168.0.39" MRTG_INT_DESCR="em1"
MaxBytes[purin_2]: 125000000
Title[purin_2]: Traffic Analysis for 2 -- purin.sevo.kyushu-u.ac.jp
PageTop[purin_2]: <h1>Traffic Analysis for 2 -- purin.sevo.kyushu-u.ac.jp</h1>
<div id="sysdetails">
<table>
<tr>
<td>System:</td>
<td>purin.sevo.kyushu-u.ac.jp in</td>
```

```
"HogeHoge's Room"</td>
</tr>
</tr>
<td>Maintainer:</td>
<td></td>
</tr>
<td>HogeHoge@sevo.kyushu-u.ac.jp</td>
<td></td>
</tr>
<td>Description:</td>
<td>em1</td>
</tr>
<tr>
<td>ifType:</td>
<td>ethernetCsmacd (6)</td>
</tr>
<tr>
<td>ifName:</td>
<td></td>
</tr>
<tr>
<td>Max Speed:</td>
<td>125.0 MBytes/s</td>
</tr>
<tr>
<td>Ip:</td>
<td>192.168.0.39 ()</td>
</tr>
</table>
</div>

### Interface 3 >> Descr: 'lo0' | Name: '' | Ip: '127.0.0.1' | Eth: "" ###
### The following interface is commented out because:
### * it is a Software Loopback interface
### * has a speed of 0 which makes no sense
#
# Target[purin_3]: 3:sevo-snmp@purin:
# SetEnv[purin_3]: MRTG_INT_IP="127.0.0.1" MRTG_INT_DESCR="lo0"
# MaxBytes[purin_3]: 0
# Title[purin_3]: Traffic Analysis for 3 -- purin.sevo.kyushu-u.ac.jp
# PageTop[purin_3]: <h1>Traffic Analysis for 3 -- purin.sevo.kyushu-u.ac.jp</h1>
#
# <div id="sysdetails">
# <table>
# <tr>
# <td>System:</td>
# <td>purin.sevo.kyushu-u.ac.jp in</td>
# </tr>
# <tr>
# <td>Maintainer:</td>
# <td></td>
# </tr>
# <td>HogeHoge@sevo.kyushu-u.ac.jp</td>
# <td></td>
# </tr>
# <td>Description:</td>
# <td>lo0</td>
# </tr>
# <td>ifType:</td>
# <td>softwareLoopback (24)</td>
# </tr>
# <td>ifName:</td>
# <td></td>
# </tr>
# <td>Max Speed:</td>
# <td>0.0 Bytes/s</td>
# </tr>
# <td>Ip:</td>
# <td>127.0.0.1 (localhost)</td>
# </tr>
# </table>
# </div>
```

図 4. MRTG 付属のツール cfgmaker の実行例

というツールを用いる。引数にコミュニティ名とホスト名を指定して各マシンに対して実行する。図4に実行例を示す。情報提供マシンにおいて、設定と snmpd daemon の立ち上げがうまくいってれば、図4のようにこのホストのすべてのNICについての情報が取得出来る。#で始まる行はコメントである。1行目から19行目は全体の設定であり、それ以降が各マシン固有の情報である。このマシンには em0 と em1 というデバイス名で認識されるNICが2つ装備されており、さらに仮想NICのループバック lo0 が認識されている（この部分はコメントアウトされている）。この中で監視の必要のないNICはコメントアウトしておくか削除しておく。

このようにして取得した各マシンの情報をコンパイルして、MRTG への入力ファイルを作る。入力ファイルは、図4の19行目までの全体の設定を頭に置き、それ以降に各マシンの情報を並べていけば良い。図5に設定例の冒頭部分を示す。必ず設定しなければならないのは、WorkDir: である。このディレクトリに出力ファイルが置かれる。また、http サーバで他のマシンに閲覧させるのであれば、このディレクトリが http サーバからアクセス出来なければならない。また、必要であれば Options も設定する。図5では、グラフが右へ延びていく、すなわち、グラフの右側に最新の情報が入るようにし（デフォルトでは、左が最新の情報）、表示の単位は bit 単位で表すようにしている。そして、その下に各マシンの設定を記述していく。このファイルを例えば /home/mrtg/mrtg-NIC.cfg にセーブする。

次に設定ファイルに間違いがないかどうか次のようにして MRTG を手動で起動する。MRTG の実行は root ユーザーである必要は無いので、例えば mrtg という名前の MRTG 実行専用のユーザを作った方が良い。

```
%/usr/local/bin/mrtg /home/mrtg/mrtg-NIC.cfg
```

最初の二回は *.old ファイルがない、といった警告が出るので、3回実行させる。それでも警告が出る場合は、設定に間違いがあるか、相手先のホストの SNMP サーバが落ちているとか、ネットワークでアクセス出来ないとか何らかの不具合があると考えられるので対処する。警告が出なくなったら、crontab で定期的に行う。図6の2行目がその設定例である。5分毎に実行させる。

図7に実行例を示す。グラフが4枚並んでおり、それぞれ過去1日間強（5分間平均値）、1週間強（30分間平均値）、1ヶ月間強（2時間平均値）、1年間強（1日間平均値）のトラフィックを表示している。縦軸が bit per second 単位での通信量を示しており、表示区間内の最大値で自動的に規格化されている（オプションの指定によって変更可能）。緑色がこのNICに入ってくるデータ、青色が出て行くデータ量を示す。常時 1.3 Mbps 程度のデータを送出しており、4 Mbps 程度のデータを受信していることが分か

```
# Created by
# /usr/local/bin/cfgmaker sevo-snmp@purin

### Global Config Options

# for UNIX
WorkDir: /home/http/mrtg

# or for NT
# WorkDir: c:\mrtgdata

### Global Defaults

# to get bits instead of bytes and graphs growing to the right
Options[_]: growright, bits

EnableIPv6: no

#####
# System: purin.sevo.kyushu-u.ac.jp
# Description: FreeBSD purin.sevo.kyushu-u.ac.jp 6.3-RELEASE-p2 FreeBSD 6.3-RELEASE-p2
#0: Thu Apr 17 14:01:53 JST
2008 HogeHoge@purin.sevo.kyushu-u.ac.jp/amanda/obj/usr/src/sys/PURIN i386
# Contact: HogeHoge@sevo.kyushu-u.ac.jp
# Location: "HogeHoge's Room"
#####

### Interface 1 >> Descr: 'em0' | Name: '' | Ip: '192.168.8.39' | Eth: '00-07-e9-09-5f-20' ###

Target[purin_1]: 1:sevo-snmp@purin:
SetEnv[purin_1]: MRTG_INT_IP="192.168.8.39" MRTG_INT_DESCR="em0"
MaxBytes[purin_1]: 125000000
Title[purin_1]: Traffic Analysis for 1 -- purin.sevo.kyushu-u.ac.jp
PageTop[purin_1]: <h1>Traffic Analysis for 1 -- purin.sevo.kyushu-u.ac.jp</h1>
.....
各マシンの情報が続く。
```

図5. NICの情報を取得するためのMRTG設定ファイルの冒頭抜粋

```
# NIC traffic
0-59/5 * * * * /usr/local/bin/mrtg /home/mrtg/mrtg-NIC.cfg > /dev/null 2>&1

# ping RTT
0-59/5 * * * * /usr/local/bin/mrtg /home/mrtg/mrtg_RTT.cfg > /dev/null 2>&1

# ping packet loss
2-59/5 * * * * /usr/local/bin/mrtg /home/mrtg/mrtg_PL.cfg > /dev/null 2>&1
```

図6. corntabの設定例

る。もう少し細かい特徴を見てみる。一番上の1日間グラフを見ると、午前8時くらいからデータ量が増え、12時頃にピークになった後、12時から13時ではややデータ量が減る。13時から16時くらいまでピークを維持し、その後段々減っていき、夜中が一番低い状態になる。二番目のグラフを見ると、月曜日から土曜日までは同じようなデータ量であるが、日曜日のピーク時のデータ量は他の曜日に比べて低い。このNICでは九州大学で収録されている地震データの受信と送信を両方行っている。地震データはWINフォーマット（卜部，1994）なので、データ量は地動の震動レベルと相関がある。そのことから、これらの変化は人間活動が活発になる朝から昼にかけて地動のノイズレベルが上がり、お昼休みの12時から13時は少し下がり、お昼休み終了後から仕事が終わるまではノイズレベルが高く、そして、夜になるとノイズレベルが下がることを示していると考えられる。また、日曜日は仕事は休みになることから他の曜日よりもノイズレベルが低くなっていると考えられる。また、通常、悪天候の時や風の強い日は、好天の時に比べると地動ノイズが大きいのでデータ量が増える傾向である。

2) RTT 及びパケット損失率収集のための設定

ここでは、ping による RTT とパケット損失率の測定結果を表示する方法を述べる。

各ホストの設定を行うために、mrtg-ping-cfg という付属のツールを用いる。

```
% mrtg-ping-cfg SBE-OBSS266 'SBE staion'
```

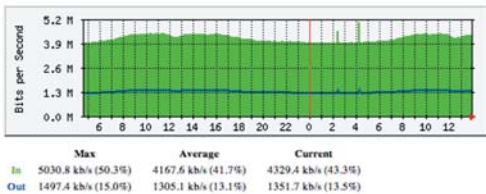
として、引数にホスト名とページのタイトルを指定して実行すると、そのホストの RTT についての設定ファイルが出力される。このコマンドの出力では、図 5 の冒頭部分にあたる Global Config Options については出力されていないので、

Traffic Analysis for 2 -- poppo.sevo.kyushu-u.ac.jp

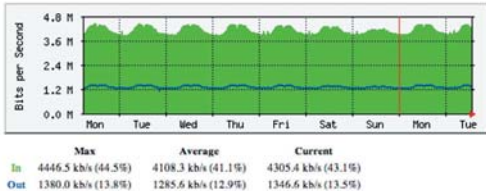
```
System: poppo.sevo.kyushu-u.ac.jp in "Clean Room"
Maintainer: snmp-master@sevo.kyushu-u.ac.jp
Description: stel
ifType: ethernetCsmacd (6)
ifName:
Max Speed: 1250.0 kBytes/s
Ip: 192.168.0.16 ()
```

The statistics were last updated Tuesday, 12 August 2008 at 13:55, at which time 'poppo.sevo.kyushu-u.ac.jp' had been up for 194 days, 19:10:37.

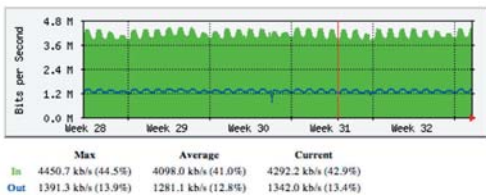
'Daily' Graph (5 Minute Average)



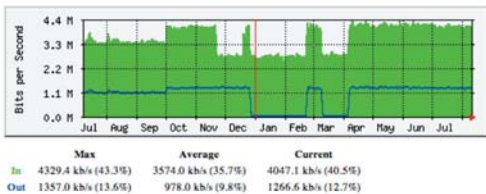
'Weekly' Graph (30 Minute Average)



'Monthly' Graph (2 Hour Average)



'Yearly' Graph (1 Day Average)



GREEN ## Incoming Traffic in Bits per Second
BLUE ## Outgoing Traffic in Bits per Second

MRTG MULTI ROUTER TRAFFIC GRAPHER
2.16.2
Tobias Oetiker <toet@postler.ch>
and Dava Rand <dr@bunji.com>

図 7. MRTG で取得した NIC のトラフィックの例

いので、各ホストの情報をコンパイルした後、WorkDir : などの情報を付け加える (図 8)。また、このファイルをコピーして編集する事により、パケット損失率の設定ファイルも作る (図 9)。mrtg-ping-probe から ping コマンドを呼び出し、デフォルトでは 56 bytes のパケットを 10 個送出し、RTT の最大値と最小値、それとパケット損失率をグラフにするものである。この値を変更したい時は、図 8 と図 9 の Target の欄の、mrtg-ping-probe の部分に“-k count -l length” をオプションとして与えれば良い。

RTT とパケット損失率の設定ファイルをそれぞれディレクトリ/home/mrtg/以下の mrtg-RTT.cfg と mrtg-PL.cfg にセーブする。そして、それぞれの設定ファイルを引数にして MRTG を手動で起動し、警告が出なくなるまで確認を行う。その後、図 6 の 6 行目と 9 行目のように、crontab で自動的に起動するように設定を行う。ところで、デフォルトの設定で ping は 1 秒毎に 10 パケット送出して返事を待つので最低でも 1 ホスト当たり 10 秒以上の時間が必要である。5 分毎に起動する場合、30 ホストの調査を行うだけで 5 分以上経過してしまう。ホスト数が多い場合は、いくつかの設定ファイルを分けてこれらを同時に起動させたり、プロセスの起動間隔を長くしたりした方がいいと思われる。九州大学の場合は 4 つのグループに分けて同時起動させている。

九州大学での運用事例

九州大学では、地震観測データのほとんどを、NTT 西

Global Config Options

```
# for UNIX
WorkDir: /home/http/mrtg-ping-RTT
```

```
# or for NT
# WorkDir: c:\mrtgdata
```

Global Defaults

```
# to get bits instead of bytes and graphs growing to the right
# Options[]: growright, bits
```

```
EnableIPv6: no
```

```
#####
```

```
# ping stats for SBE-OBSS266
```

```
# SBE staion
```

```
Title[SBE-OBSS266]: SBE staion
MaxBytes[SBE-OBSS266]: 5000
AbsMax[SBE-OBSS266]: 10000
Options[SBE-OBSS266]: gauge
Target[SBE-OBSS266]: /usr/local/bin/mrtg-ping-probe SBE-OBSS266
PageTop[SBE-OBSS266]: <H1>SBE staion</H1>
<P>Actually we are measuring the ping time between our web server and SBE-OBSS266.
```

```
YLegend[SBE-OBSS266]: Round Trip Time
ShortLegend[SBE-OBSS266]: ms
Legend1[SBE-OBSS266]: Maximum Round Trip Time in Milli Second
Legend2[SBE-OBSS266]: Minimum Round Trip Time in Milli Second
Legend3[SBE-OBSS266]: Maximal 5 Minute Maximum Round Trip Time
Legend4[SBE-OBSS266]: Maximal 5 Minute Minimum Round Trip Time
Legend[SBE-OBSS266]: &nbsp;Max:
LegendO[SBE-OBSS266]: &nbsp;Min:
WithPeak[SBE-OBSS266]: ymwd
```

```
#-----
```

```
.....
以下、各ホストの設定が続く。
```

図 8. RTT を取得するための MRTG 設定ファイルの冒頭部の抜粋

日本がサービスを提供しているフレッツグループ回線網を使うことによってテレメータしている (図 10)。使用しているグループは 3 つである。観測点へのアクセス回線は 64 kbps の ISDN 回線がほとんどである。また、他機関とのデータ交換は、近年においては JGN2plus² や SINET3³ を用いた JDXnet によって行っているが (例えば鷹野ほか、

2008 ; Takano *et al.*, 2008), 九州大学においては JGN2 plus と SINET3 のアクセスポイントは福岡市東区にある九州大学情報基盤研究開発センター (Research Institute for Information Technology : RI2T) 内にあるため、地震火山観測研究センター (SEVO) のある長崎県島原市までは、観測点データと同様にフレッツグループ網を使ってデータ通信を行っている (図 10)。SEVO と RI2T でのアクセス回線は 100 Mbps の光回線である。なお、図 10 には記載していないが京都大学防災研究所、東京大学地震研究所和歌山観測所と広島観測所、鹿児島大学南西島弧地震火山観測所がメンバーになっているフレッツグループに SEVO も接続しており、こちらを通してデータ交換を行っている。

NIC のトラフィックの監視は、SEVO 内のマシンと、RI2T 内のデータ交換マシンに対して行っている。RTT 及びパケット損失率の監視は、RI2T 内のデータ交換マシンと、IP 接続された観測点 (フレッツグループ及び専用線とも) に対して行っている。IP 接続された観測点には、観測点側でデータをバックアップするために OpenBlocks266 が設置してあるので (Uehira, 2008, 内田・植平, 2008), これに対して ping を行っている。

1. データ交換装置の各 NIC での通信量 (1)

上で述べたように、JDXnet のアクセスポイントは RI2T

```
### Global Config Options
# for UNIX
WorkDir: /home/http/mrtg-ping-packetloss
# or for NT
# WorkDir: c:\mrtgdata
### Global Defaults
# to get bits instead of bytes and graphs growing to the right
# Options[_]: growright, bits
EnableIPv6: no
#####
# ping stats for SBE-OBSS266
# SBE station
Title[SBE-OBSS266]: SBE station
MaxBytes[SBE-OBSS266]: 100
Options[SBE-OBSS266]: noinfo,gauge,growright,nopercent
Target[SBE-OBSS266]: /usr/local/bin/mrtg-ping-probe -p loss/loss SBE-OBSS266
PageTop[SBE-OBSS266]: <H1>SBE-OBSS266</H1>
<P>Actually we are measuring the ping packet loss between our web server and
SBE-OBSS266.
YLegend[SBE-OBSS266]: Packet loss (%)
ShortLegend[SBE-OBSS266]: %
Legend1[SBE-OBSS266]: packet loss
Legend2[SBE-OBSS266]: packet loss
Legend3[SBE-OBSS266]: Maximal 5 Minute packet loss
Legend4[SBE-OBSS266]: Maximal 5 Minute packet loss
LegendI[SBE-OBSS266]: &nbsp;loss:
LegendO[SBE-OBSS266]: &nbsp;loss:
WithPeak[SBE-OBSS266]: ymw
# -----
.....
以下、各観測点データが続く。
```

図 9. パケット損失率を取得するための MRTG 設定ファイルの冒頭部の抜粋

² <http://www.jgn.nict.go.jp/>

³ <http://www.sinet.ad.jp/>

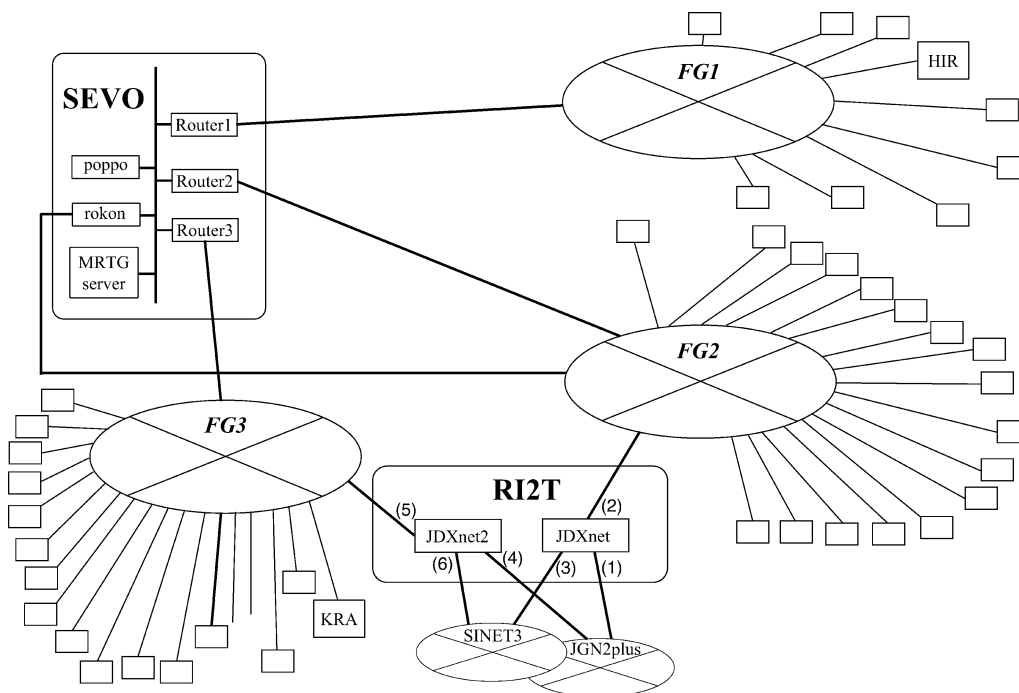


図 10. 九州大学地震火山観測研究センター (SEVO) の地震観測網の概略図

内にあるので、この中にJDXnetとJDXnet2という2台のFreeBSDマシンをデータ交換のために設置している。図10に示すように、2台ともSINET3とJGN2plusに接続しており、また、SEVOとはそれぞれフレツグループFG2とFG3を介して接続している。2008年7月23日から24日にかけての1日間強のこれら2台のマシンのNIC(1)から(6)のトラフィックの様子を図11に示す。この2台のマシンは両方ともSINET3とJGN2plusから全国全てのデータを同じ共有メモリで受け、受けた全てのチャンネルをそれぞれのマシンからSEVOに送信している。トラフィックの増減の様子は、図7の本文における説明がほぼ当てはまるような動きである。その中で一つ特徴的なのが、7月24日の0時30分頃にデータ量が通常より2Mbps程度増えていることである。これは、2008年7月24日の00:26に岩手県沿岸北部の深さ108kmにおいてMj 6.8の地震が発生し、岩手県の一部地域では震度6強を観測したが、この地震に伴いWINデータの圧縮率が下がり、データ量が増えたことによるものである。

2. データ交換装置の各NICでの通信量(2)

図12に、2008年8月3日から4日にかけて計測した、図10の(1)から(6)のNICの通信量を示す。図11で見たように、通常データ量は6Mbps前後であるが、JDXnet、JDXnet2とも、SINET3及びJGN2plusのネットワークで8月3日19時前後に最大100Mbps前後のデータ量になっている((1),(3),(4),(6))。この時間、大量の再送要求のログが残されていたので、何らかの理由で大量の再送要求が発生したことにより通常の数倍ものトラフィックになったと考えられる。この間、SEVOへの通信も若干の影響を受けているが((2),(5))、収録への影響は出なかった。

3. 機器保守の例

図13に、図10のRouter3の保守を行った時のRTTとパケット損失率の例を示す。JDXnet2以外はすべてフレツISDN回線である。左側のグラフがRTTを表し、右側のグラフがパケット損失率を表している。RTTについては、緑色グラフがRTTの最大値、青色が最小値を表し、パケット損失率については緑色も青色も同じパケット損失率を表す。

15時前後に、Router3のファームウェアのアップデートのためRouter3の通信を止めたが、その時間帯でRTTが0msec、パケット損失率が100%になっており、通信が出来なくなっていることが分かる。

4. 通信断による欠測とデータリカバリの例

図14に2008年7月16日から17日にかけてのKRA観測点におけるRTTとパケット損失率を示す。7月17日の0時30分頃から7時過ぎにかけてRTTが0msecで、パケット損失率が100%になっており、この間KRA観測点

への通信が出来なかったことが分かる。回線復旧後、通信が出来なかった時間帯においても観測データがOpenBlockS266のCFカードに残されていたことにより、観測点での電源トラブルではないことが断定出来るで、この時間帯においてはISDN回線の不具合、もしくはルータのハングアップにより通信が出来なかったことが推測される。

9時から10時にかけてRTTの最大値(緑色グラフ)の値が大きくなっている。この時間帯は、観測点に設置してあるOpenBlocks266にバックアップしていた欠測期間のデータをWRRPシステム(Uehira, 2008)によりSEVOに送信していた時間帯である。つまり、この時間帯はテレメータされている通常のデータの他にリカバリデータも同時に流れており、64kbpsの回線帯域をほぼ占有していた状態であるので、pingのRTTの最大値が通常よりも数倍程度大きくなったことが分かる。

5. 天候不良により回線が不安定な場合の例

図15に2008年7月20日から21日にかけてのHIR観測点におけるRTTとパケット損失率を示す。7月20日14時前後と、7月21日4時から7時の間にパケット損失率が大きい時間帯が断続的にあり、この間リアルタイムのデータも数秒から数分間程度、断続的に欠測した。この時間帯のアメダスや九州電力の雷情報をみると、この観測点付近において雷雨があったことが分かっている。雷雨によりISDN回線が不安定になり、それによりテレメータデータも欠測したと考えられる。

九州地方においては、夏季に局地的に雷雲が発生し雷雨を降らせるが、観測点のそばでこのような激しい雷雨が発生すると数秒から数分程度のパケット落ちが発生することが多い。これは、雷サージなどによりISDN回線が一時的に不安定になるからであろう。

6. フレツグループ網の障害の例

図16に2008年2月7日から8日にかけてのFG1とFG2に接続している観測点及びJDXnetのRTTを示す。2月7日の午前3時からRTTが全点で0msecになっているが、これはNTT島原局における工事の影響である。この工事により、FG1からFG3までのすべてのフレツグループの通信が午前3時頃切れた。約30分後に工事が終了し、FG1は自動復帰したが、FG2とFG3は自動復帰せず、朝8時過ぎにRouter2とRouter3を手動で再起動させることにより復旧させた。このため、FG1とFG2(とFG3)では通信の復旧時刻が異なっている。

その後、FG2のJDXnetに対するRTTの値が午前10時頃より段々大きくなっていき、通常は30msec程度なのが100msecを超えるような値になった。その後も通常より値が大きい状態が続き、23時頃ようやく通常状態に戻った。また、FG1とFG2(とFG3)のほぼすべての観測点について同じ時間帯にRTTの値が通常よりも大きく



図 11. 2008 年 7 月 23 日から 24 日にかけての JDXnet と JDXnet2 のそれぞれの NIC における通信量



図 12. 2008 年 8 月 3 日から 4 日にかけての JDXnet と JDXnet2 のそれぞれの NIC における通信量

なっていることより、SEVO のアクセス回線の不具合か、もしくはフレッツグループ網全体の障害が考えられた。原因の切り分けのため NTT 西日本のサポートに調査を依頼したが、フレッツグループ網の障害は報告されておらず、また、アクセス回線にも問題が無いようであった。結局、NTT 西日本の営業担当者に直接調査を依頼した所、福岡県内の通信網で障害があり（通信が断ではないので情報がいっていなかったらしい）、この影響で JDXnet までの RTT が大きくなっていったということが判明した。また、福

岡は九州域の通信網のハブであるので福岡以外の地域にも影響がでたと推測される。

この障害によるデータ収録への影響は出なかったが、RI 2T からの他機関からのデータ送受信で若干再送要求が増えた。

ま と め

インターネットを用いた地震データ伝送では、データの欠落が様々な要因で発生し、その原因を特定することは容

Flets Group 3

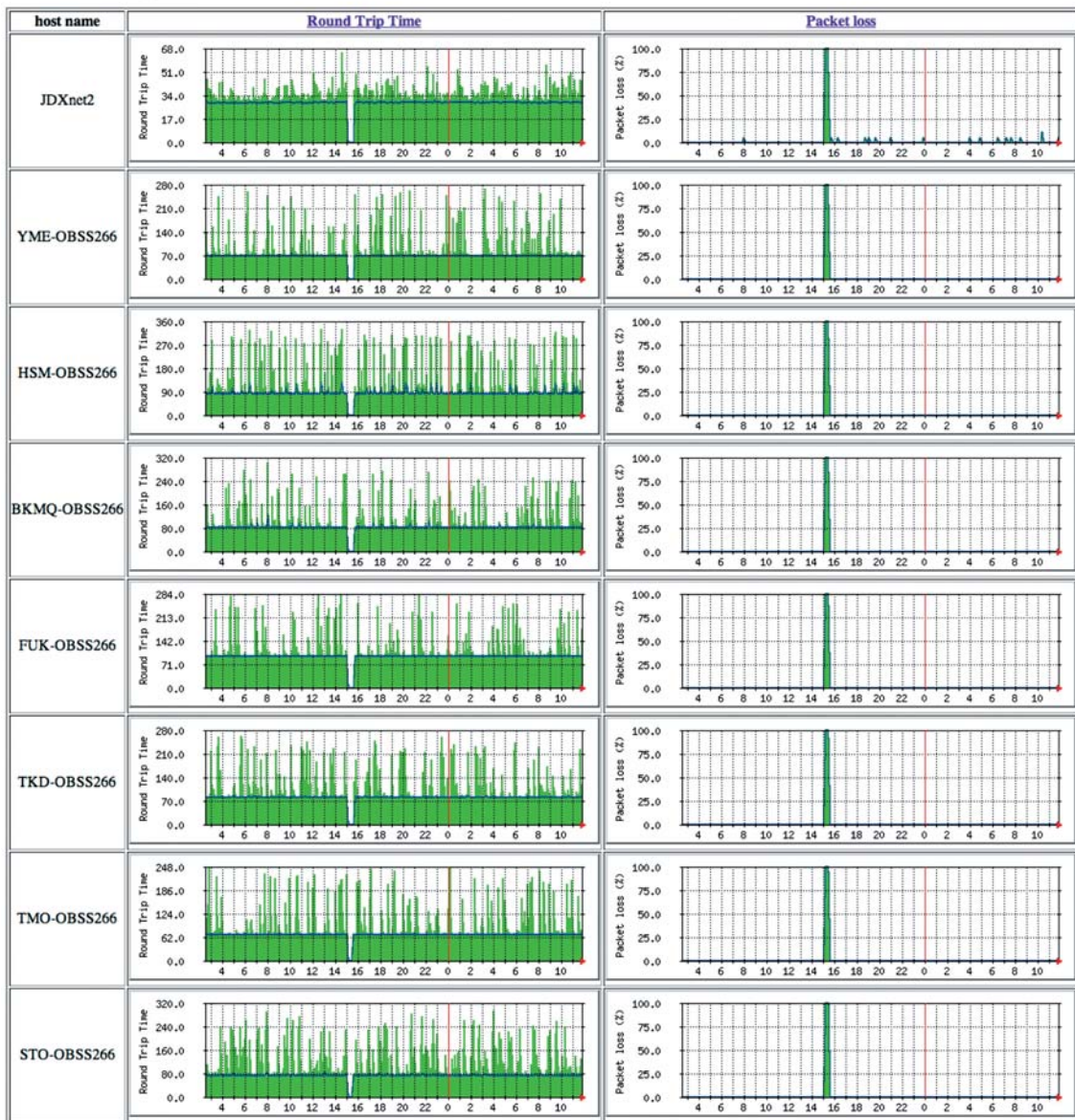


図 13. FG3 に接続されている JDXnet2 と観測点に設置された OpenBlockS266 までの RTT (左グラフ) とパケット損失率 (右グラフ)

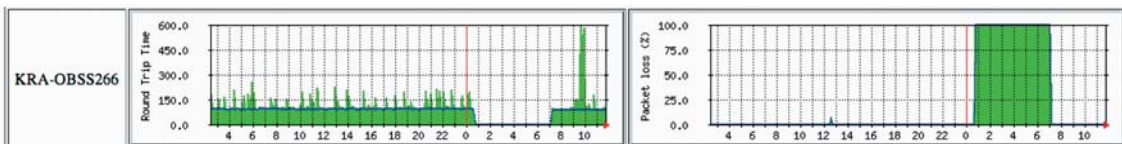


図 14. 2008 年 7 月 16 日から 17 日にかけての KRA 観測点の OpenBlockS266 までの RTT (左グラフ) とパケット損失率 (右グラフ)

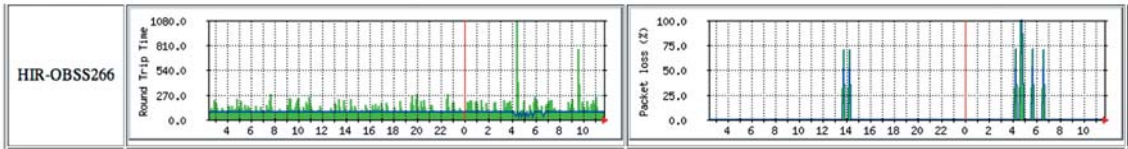
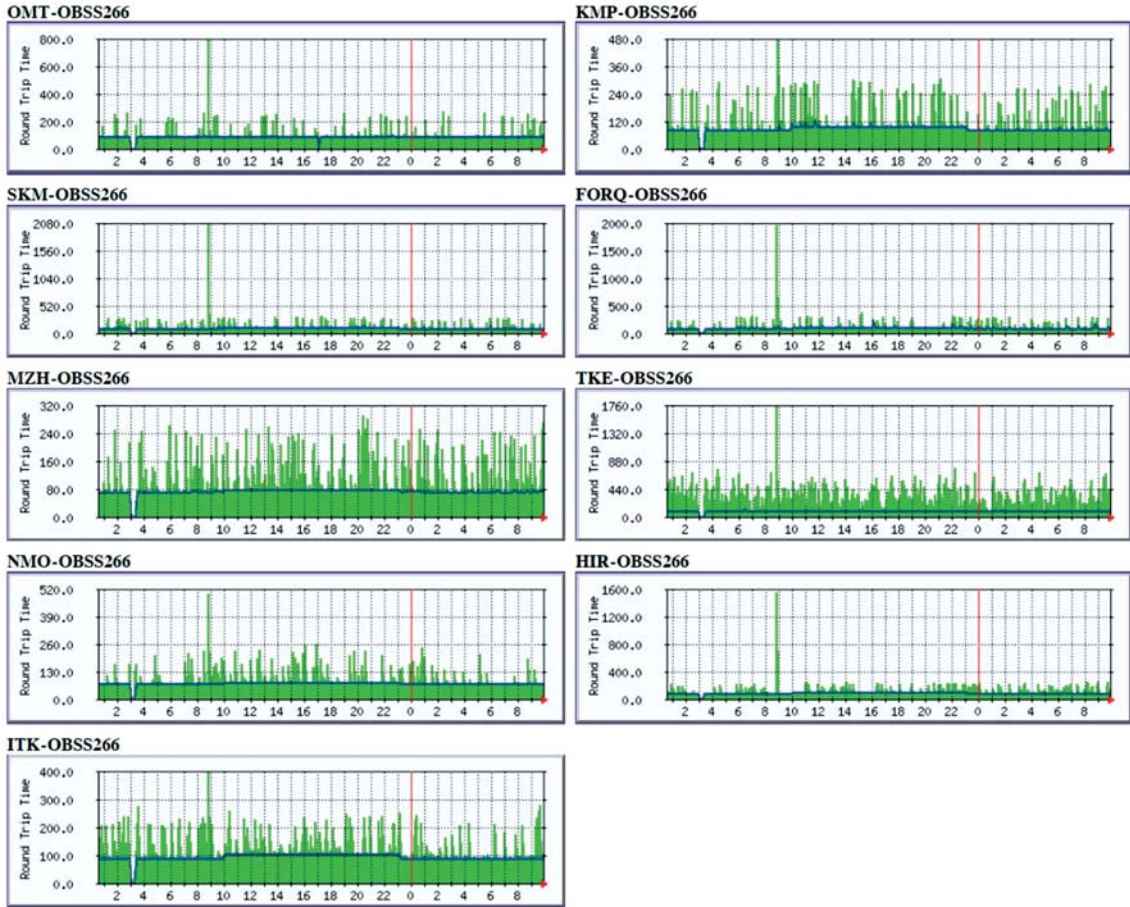


図 15. 2008 年 7 月 20 日から 21 日にかけての HIR 観測点の OpenBlockS266 までの RTT (左グラフ) とパケット損失率 (右グラフ)

Flets Group 1



Flets Group 2

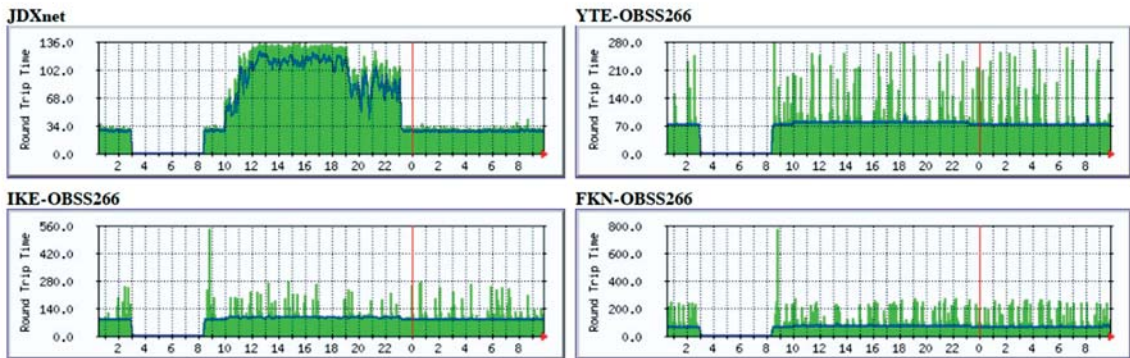


図 16. 2008 年 2 月 7 日から 8 日にかけての FG1 と FG2 に接続している観測点の OpenBlockS266 及び JDxnet までの RTT

易ではない。原因を特定する道具として、フリーのソフトウェアである MRTG を用いることにより、IP 接続された観測点までの回線状況を監視したり、データの収録マシンの NIC のトラフィックを監視したりする方法を述べ、またいくつかの障害や特別な状況の例を挙げて、その時に MRTG によってどのようなグラフが描かれるかを述べた。本論説で述べたように、MRTG でグラフ化することにより観測網の通信の様子が直感的に分かるため、その他の様々な情報と組み合わせることにより障害の切り分けが楽に行うことができ、地震観測網の維持管理の一助になると考えられる。

今回、データの収録マシンの NIC のトラフィックを監視するために各マシンに net-snmp をインストールする方法を述べたが、これらのマシンに接続しているスイッチやハブが SNMP をサポートしているのであれば、これらのポート毎のトラフィックも MRTG で同様にグラフ化することが出来る。場合によってはこちらの方法を使うのも良いであろう。

謝 辞：本稿をまとめるにあたり、森田裕一准教授と新谷昌人准教授には丁寧に査読していただき有益なコメント

をいただきました。記して感謝致します。

文 献

- Takano, K., N. Hirata, T. Urabe, M. Kasahara, M. Kosuga, S. Miura, T. Ito, Y. Kano, S. Ohmi and K. Uehira, 2008, The JDXnet : Japan Data eXchange network for earthquake observation data, Abstract of International Symposium : Fifty Years after IGY—Modern Information Technologies and Earth and Solar Sciences—.
- 鷹野 澄・鶴岡 弘・ト部 卓・中川茂樹・一柳昌義・高田真秀・山口照寛・高橋浩晃・笠原 稔・小菅正裕・渡邊和俊・内田直希・平原 聡・中山貴史・伊藤武男・中道治久・山中佳子・大見士朗・三浦 勉・加納靖之・須田直樹・植平賢司・内田和也・馬越孝道・八木原寛・久保篤規・坪井誠司・渡邊智毅, 2008, SINET3 広域 L2 網による次世代全国地震データ流通基盤システムの構築, 地球惑星科学関連学会 2008 年合同大会予稿集, S114-009.
- Uehira, K., 2008, Development of a distributed backup system and a recovery system for telemetric seismic data, *Earth Planets Space*, in press.
- ト部 卓, 1994, 多チャンネル地震波形データのための共通フォーマットの提案, 日本地震学会講演予稿集 1994 年度秋季大会, P 24.
- 内田和也・植平賢司, 2008, マイクロサーバ・OpenBlocks266 を使った地震観測点におけるバックアップ, 震研技報, 14, 38-46.