

# Design and Implementation of One-way IP Performance Measurement Tool

Jaehoon Jeong<sup>1</sup>, Seungyun Lee<sup>1</sup>, Yongjin Kim<sup>1</sup>,  
and Yanghee Choi<sup>2</sup>

<sup>1</sup> Protocol Engineering Center, ETRI, 161 Gajong-Dong, Yusong-Gu,  
Daejeon 305-350, Korea  
{paul, syl, kimyj}@etri.re.kr  
<http://pec.etri.re.kr/>

<sup>2</sup> School of Computer Science and Engineering, Seoul Nat'l Univ.,  
San 56-1 Shinlim-Dong, Kwanak-Gu, Seoul 151-742, Korea  
yhchoi@snu.ac.kr  
<http://mmlab.snu.ac.kr/>

**Abstract.** In this paper, we propose an architecture of measurement system which can measure IETF's IP Performance Metrics (IPPM) such as one-way delay, one-way packet loss and packet delay variation in the Internet. As the synchronization among measurement systems is very important in one-way delay measurement, we used the Global Positioning System (GPS) to synchronize the measurement systems and provided the precision up to one micro-second. To improve the accuracy of one-way delay measurement, the proposed system employs timestamps at the Ethernet frame level. We carried out measurements on the real Internet with the implemented system. It is seen that delay differs largely between path directions. Through these measurements, we present the need of one-way delay measurement.

## 1 Introduction

We need one-way measurement of the Internet in order for us to grasp the exact state of the Internet, which is asymmetric [1]. The IETF's IP Performance Metrics (IPPM) Working Group suggested one-way metrics and architecture for one-way measurement [2]. Metrics are one-way delay, one-way packet loss, instantaneous packet delay variation, etc [3-5]. One-way measurement is a kind of active measurement, which injects measurement packets in a path to measure and observes how the packets are served. With measurement data obtained through active measurement, the network management can be performed effectively. For example, we can conjecture that some problems happened in the network if we have observed the network for a long time and have found that one-way delay increased much more than at ordinary times. Through the analysis of measurement result that we have

---

This work has been partially supported by Brain Korea 21, National Research Laboratory program, and Korea Telecom 2001.

obtained through long observation of network, we can find what is the problem of the network (e.g., bottleneck path) and then can solve it by relocating resources, increasing link capacity, changing network configuration (e.g., routing configuration) and upgrading routers. In the result, we can improve the performance of the entire network.

The system for the active measurement that provides us with the necessary information for effective network management should provide operator with user interface with which operator can control the system easily and efficiently. It should also be able to measure the network stably for a long time and have functions of trouble-shooting, which are to find the troubles that can happen in measurement system or network during measurement and to solve them automatically without operator's intervention.

In this paper, we suggest an architecture of measurement system (AMT: Active Measurement Tool) that can perform one-way measurement efficiently and stably; AMT has been designed and implemented so that it may measure one-way metrics stably for a long time and be expanded easily in the point of the number of measurement systems. We present the analysis of result that we have measured in the Internet with AMT. This paper is organized as follows; Section 2 presents related work. In section 3, we explain the architecture of the suggested measurement system (AMT), the components of the system, the procedure of measurement, and the visualization of measurement result. We also evaluate the result of one-way measurement in test network. Finally, in section 4, we conclude this paper and present future work.

## **2 Related Work**

Many measurement systems were implemented for active measurement. We introduce two representatives among the systems; (a) Skitter and (b) Surveyor.

### **2.1 Skitter**

Skitter is a measurement system that Cooperative Association for Internet Data Analysis (CAIDA) Group has implemented [6]. Skitter was made for analysis of Internet's topology and performance. It injects measurement packets in Internet and observes how the packets are served. Main functions are as follows; (a) Measurement of Forward IP Path, (b) Measurement of RTT, (c) Trace of Routing Change, and (d) Visualization of Network Topology. Skitter provides users with easy and convenient user interface but has a demerit that it can not measure one-way metrics.

### **2.2 Surveyor**

Surveyor is a measurement system that Advanced Network & Services Group has implemented that can measure one-way metrics [7, 8]. The one-way metrics are based

on IETF's IPPM. Surveyor consists of two systems; (a) Measurement System and (b) Central Control System. Two systems use One-Way Delay and Packet loss protocol (OWDP) [9, 10]. Because Measurement Systems are synchronized with one another by GPS, they can perform one-way delay measurement accurately. Central Control System controls Measurement Systems and gathers measurement data from the Measurement Systems. To improve the accuracy of one-way measurement, Surveyor stamps the time information in Ethernet device driver. It is one of the most popular systems for one-way measurement.

### **3 Active Measurement Tool (AMT)**

AMT is an infrastructure that can measure various one-way metrics suggested by IETF's IPPM Working Group. AMT is a PC-based system that uses FreeBSD and MySQL as operating system and database management system respectively [11, 12].

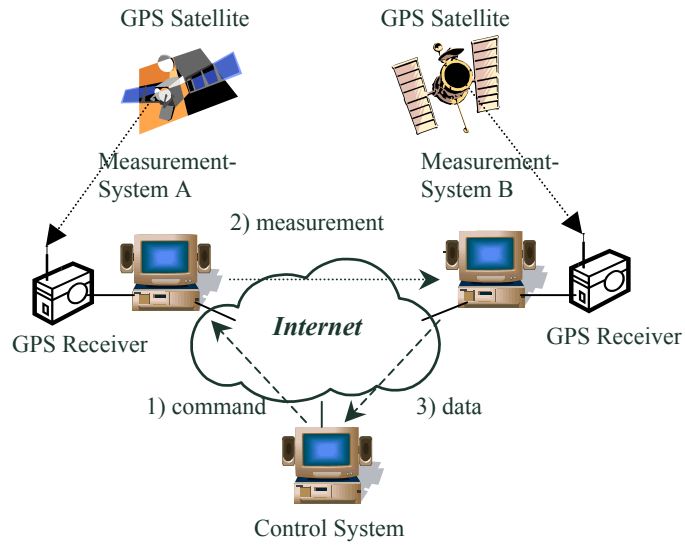
#### **3.1 Consideration for Implementation of One-way Measurement**

##### **3.1.1 How to synchronize Measurement Systems**

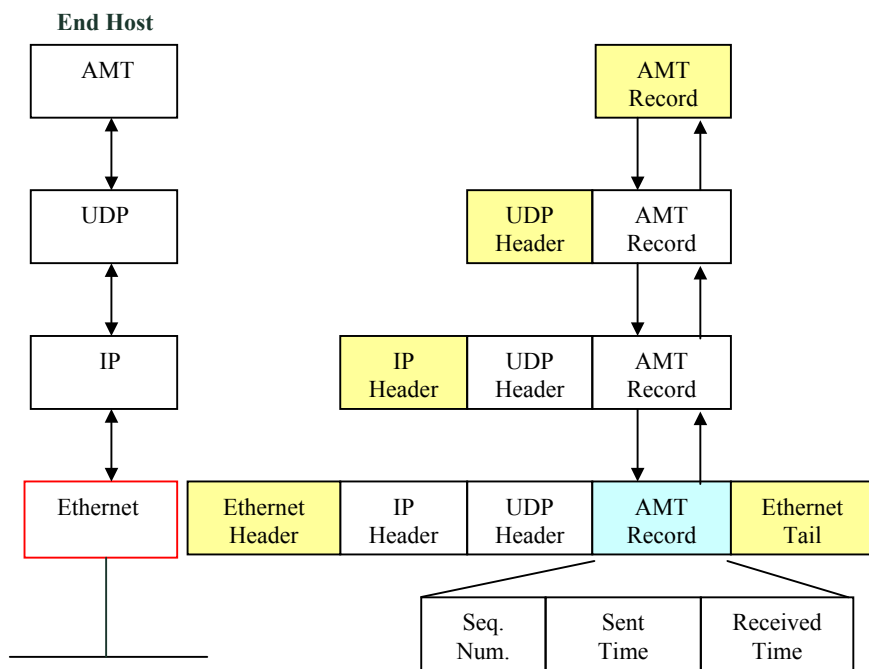
There is no need to synchronize measurement systems in order to measure RTT which means two-way delay. However, when it comes to measurement of one-way delay, we should synchronize measurement systems for the exact measurement. Fig. 1 shows how to synchronize systems by using GPS satellites. Through GPS satellites, the exact time information can be provided for measurement systems, which can maintain their system time correctly with it. Hardwares that are used to receive time information from GPS are as follows; (a) Oncore Remote Antenna and (b) Oncore GPS UT Receiver, which are the products of Motorola [13]. Network Time Protocol (NTP) Daemon (i.e., ntpd [14, 15]) modifies the kernel time with time information received from GPS. The time information encoded in Pulse Per Second (PPS) format can be provided for ntpd through either serial port or parallel port [14]. Device driver of the port transforms the PPS into binary format and provides ntpd with the time information formatted as binary. The ntpd updates the kernel time periodically with the time information. In this mechanism, all measurement systems are synchronized with GPS.

##### **3.1.2 Timestamp**

To improve the accuracy of measuring one-way delay, the measurement system has to timestamp on the field for time information in the payload which is one of fields in the Ethernet frame just before transmitting the Ethernet frame to the network interface card as well as just after receiving the Ethernet frame from the network interface card like Fig. 2. In this way, we are capable of reducing the delay which can occur through the protocol stack at end hosts [8, 16].



**Fig. 1.** Synchronization among Measurement Systems and Procedure of Measurement



**Fig. 2.** Timestamping at AMT

### 3.2 Architecture of AMT System (AMT)

AMT consists of two kinds of systems ; (a) Control System (CS) and (b) Measurement System (MS). While MS performs one-way measurement, CS controls and manages the MS's. Fig. 3 describes the architecture of AMT. Fig. 1 describes the procedure of measurement which is performed by CS and MS's.

#### 3.2.1 Control System (CS)

CS, main system of AMT, receives commands sent from Control Shell (CSH), with which operator controls and manages AMT. CSH is console-based user interface. CS has three processes like Fig. 3; (a) Control Server (CSV), (b) Storage Server (SSV) and (c) DB Server (DBS).

*Control Server (CSV):* CSV receives commands from operator, parses the commands, and then processes the commands. CSV consists of three threads; (a) Main Thread (MAT), (b) Measurement Thread (MET) and (c) Polling Thread (POT). MAT receives command from CSH and processes it. MET initiates a measurement and POT checks the health of measurement systems and network.

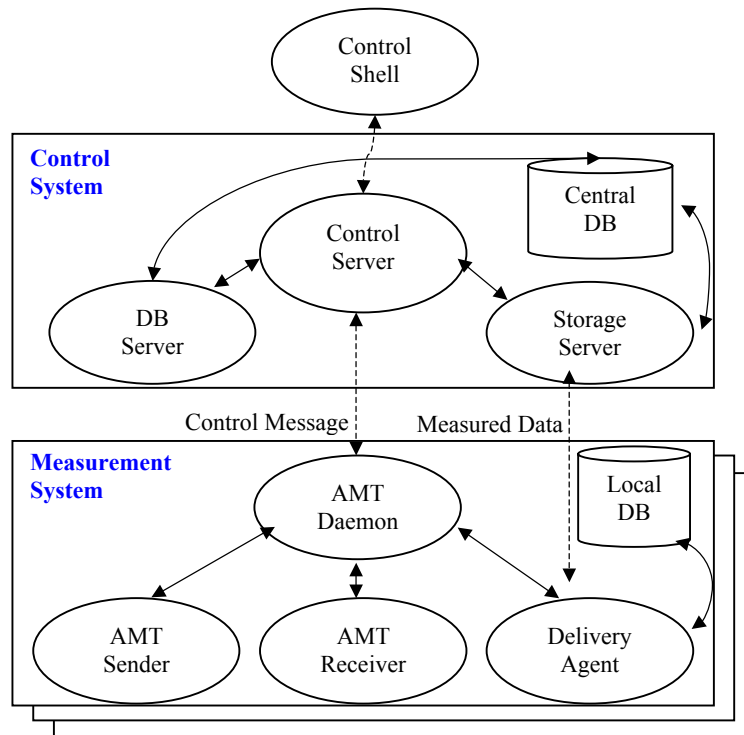


Fig. 3. Architecture of AMT System

*Storage Server (SSV)*: SSV collects measurement data from local database (Local DB) of each MS after the measurement and stores the data in the central database (Central DB). It is forked by CSV when preparing the collection. The collection is performed with the aid of Delivery Agent (DA) of each MS.

*DB Server (DBS)*: DBS analyzes the gathered raw data and stores them into Central DB.

### 3.2.2 Measurement System (MS)

MS has four processes like Fig. 3; (a) AMT Daemon (AMTD), (b) AMT Sender (AMTS), (c) AMT Receiver (AMTR) and (d) Delivery Agent (DA). AMT Daemon (AMTD)

*AMT Daemon (AMTD)*: After AMTD, main process of MS, first registers itself in CS, it receives all the control messages from CSV, processes them and sends the result to CSV. For example, when CSV sends the measurement preparation message to the registered AMTD of each MS, AMTD receives the message to prepare measurement. It forks AMT Sender (AMTS) and AMT Receiver (AMTR) which will perform actual measurement. All the control messages from CSV to AMTS or AMTR of each MS are sent to AMTS or AMTR via AMTD of the MS. The reason that we designed AMT system for all the control message messages between CSV and AMTS or AMTR to go via AMTD is that we tried to make AMTS and AMTR be light-weighted processes that can run stably for a long time.

*AMT Sender (AMTS)*: AMTS is forked by AMTD when CS starts measurement. After AMTS receives a measurement start message, it generates measurement packets. The packets are generated in Poisson process by a pseudo-random number generator. AMTS sends every packet to all the AMTRs which are joining in the measurement.

*AMT Receiver (AMTR)*: AMTR is forked by AMTD when CS starts measurement. After AMTR receives a measurement start message, it opens Local DB file to be ready to receive measurement packets. Whenever it receives a measurement packet, it stores the record of the packet in Local DB. The record consists of 5 fields; (a) Sequence Number, (b) Sender IP Address, (c) Sent Time, (d) Receiver IP Address, and (e) Received Time. 'Sequence Number' is 4-byte sequence number field. 'Sender IP Address' is 4-byte IP address field of AMT sender that sent the packet. 'Receiver IP Address' is also 4-byte IP address field of AMT receiver that received the packet. 'Sent Time' is 8-byte timestamp field in which the timestamp is written by Ethernet device driver just before packet's being sent into network interface card. The type of this field is struct timeval { u\_long tv\_sec; u\_long tv\_usec }. 'Received Time' is also 8-byte timestamp field where the timestamp is written by Ethernet device driver just after packet's being received from network interface card.

*Delivery Agent (DA)*: DA is forked by AMTD when CS gathers measurement data from each MS. After DA receives a gather start message, it opens Local DB and delivers the measurement data stored in it to SSV of CS.

### **3.3 Procedure of Measurement**

The procedure of measuring one-way delay is described as shown in Fig. 4.

#### **Step 1. Initialization of AMTD for measurement**

CSV sends all the AMTDs that take part in measurement a ‘measure-ready’ message indicating that they have to prepare a measurement. The control packet including the message provides them with a system parameter (i.e., lambda value for Poisson process) and a list of IP addresses of all the participating AMTDs together with the message.

#### **Step 2. Fork of measurement processes**

When AMTD of MS receives the ‘measure-ready’ message, it makes control channels that will be used to communicate with AMTS and AMTR that are implemented in UNIX domain stream socket. It forks AMTS and AMTR and then forwards the ‘measure-ready’ message to them through the control channels.

#### **Step 3. Establishment of control channel**

Just after AMTS and AMTR have been forked by AMTD, they establish control channel that is used to communicate with AMTD. AMTS and AMTR obtain the system parameter such as the list of IP addresses of participants from control packet including the ‘measure-ready’ message. When AMTS and AMTR are ready to measure, they report the readiness to AMTD through the control channel.

#### **Step 4. Confirmation about readiness from AMTD**

When AMTD receives the report from both AMTS and AMTR, AMTD sends CSV a ‘measure-ready-ack’ message indicating that MS is ready to measure.

#### **Step 5. Start of measurement**

When CSV has received the report from all participating AMTDs, CSV sends them a ‘measure-start’ message indicating that they have to start measurement.

#### **Step 6. Start of actual measurement**

When AMTD receives the ‘measure-start’ message, it forwards the message to its child processes; AMTS and AMTR.

#### **Step 7. Injection of measurement packets**

AMTS generates measurement packets in Poisson process. The packets are sent to all participating AMTRs except AMTR in the same host through UDP socket.

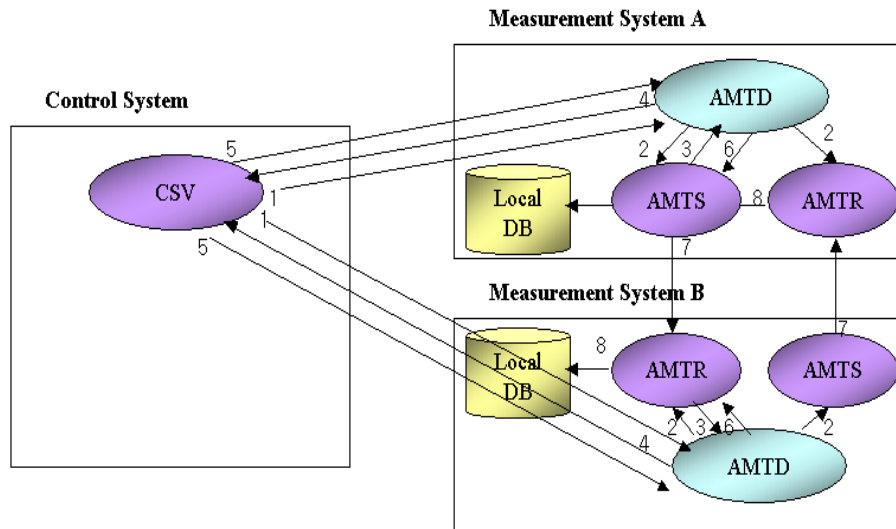


Fig. 4. Procedure of Measurement

#### Step 8. Storing of measurement records

When AMTR receives a measurement packet, it stores into Local DB a record that consists of the following fields; (a) Sequence Number, (b) Sender IP Address, (c) Sent Time, (d) Receiver IP Address, and (e) Received Time. The record is stored in binary format, not ASCII format in order to reduce the size of record.

#### 3.4 Visualization of Measurement Result

AMT Visualizer (AMTV) can provide operator with the result of measurement through web like Fig. 5 and Fig. 6. AMTV receives the following inputs; (a) Sender IP, (b) Receiver IP, and (c) Date. The combination of three input fields means that we want to get the result from the measurement packets which MS with 'Sender IP' address generated and sent to MS with 'Receiver IP' address on 'Date'.

We present an example with Fig. 5 and Fig. 6. The meaning of input in Fig. 5 is that we want to get the result from measurement packets that MS with IP address 147.46.14.69 sent to MS with IP address 203.232.127.20 on November 28, 2000. The output of the result is the graphs of one-way delay, one-way loss, and delay jitter during the day. Fig. 6 shows the one-way delay on November 20, 2000 as the result of the query of Fig. 5. Fig. 7 describes the procedure of visualization. When operator sends a query requesting measurement result between two end hosts on a specific day with AMTV, the query is transferred to CGI Module called as Measurement Analysis Agent (MAA) via Web Server (httpd). MAA processes the query with Central DB and returns the result to AMTV via httpd.



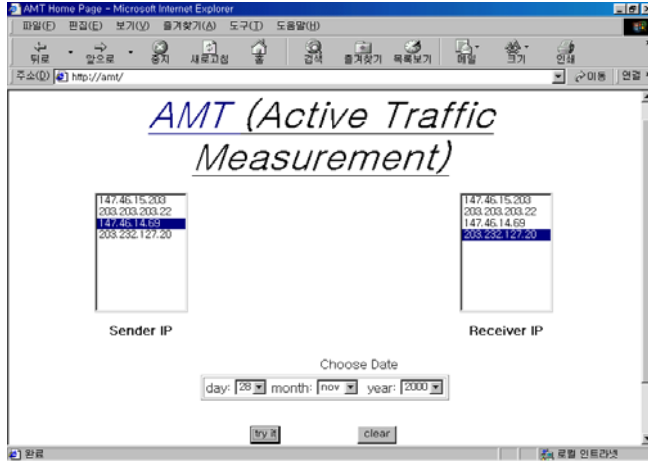


Fig. 5. AMT Visualizer (AMTV)

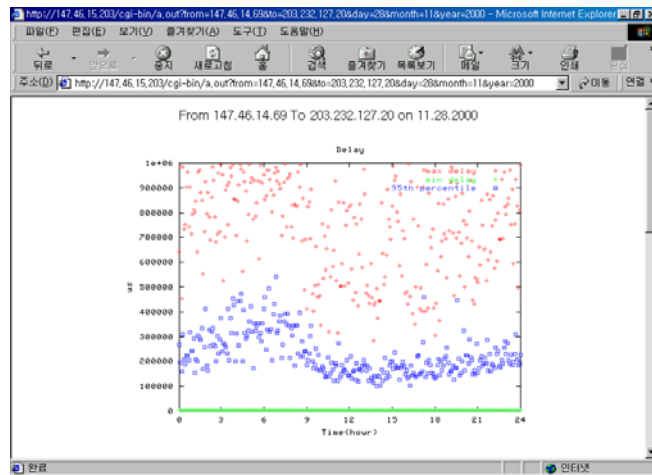


Fig. 6. Result of Query at AMTV

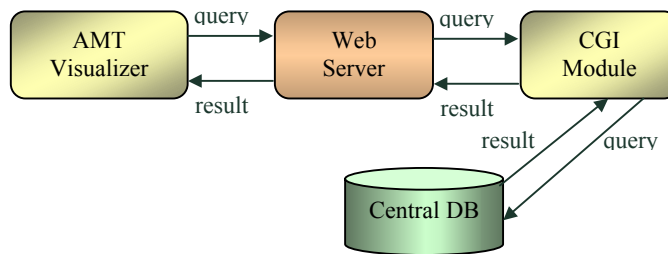
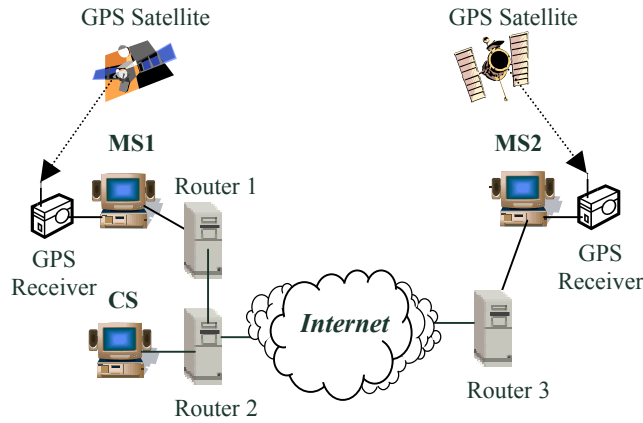


Fig. 7. Procedure of Visualization



**Fig. 8.** Test Network

### 3.5 Performance Evaluation

We measured one-way delay in Internet including Korea Commercial Network (KORNET) and evaluated the result of measurement.

#### 3.5.1 Test Environment

Fig. 8 shows the topology of test network. Measurement System1 (MS1) of which IP address is 147.46.14.69 is located in a subnet of Seoul National University and Measurement System2 (MS2) of which IP address is 203.232.127.20 is located in a subnet of KORNET. Control System (CS) of which IP address is 147.46.15.203 is located in a subnet of Seoul National University. Router 1's IP address is 147.46.14.65, Router 2's IP address is 147.46.15.2 and Router 3's IP address is 203.232.127.14. Router 1 is adjacent to Router 2 as shown in Fig. 8.

#### 3.5.2 Evaluation of Measurement Result

We measured one-way delay during a day from 0 AM on 2000/11/28 to 12 PM on 2000/11/28. We generated measurement packets in the frequency that the lambda of Poisson process is 2.

Fig. 9 shows one-way delay from MS1 to MS2 (Delay1) and Fig. 10 shows one-way delay from MS2 to MS1 (Delay2). X-axis of graph is time. The unit is 5 minutes. Y-axis is one-way delay. The unit is 1 micro-second (us). As representative values, we selected (a) Minimum delay, (b) 95<sup>th</sup> percentile and (c) Maximum delay in the period of 5 minutes. Because percentile is the most reasonable among three representatives, we compare two figures (Fig. 9 and Fig. 10) by 95<sup>th</sup> percentile. We can see that 95<sup>th</sup> percentile of Delay1 is from 100148[us] to 539724[us] and that 95<sup>th</sup> percentile of Delay2 is from 7923[us] to 16344[us]. As a result, we can see that the

one-way path from MS1 to MS2 (Path1) has bigger and more variable one-way delay than that from MS2 to MS1 (Path2).

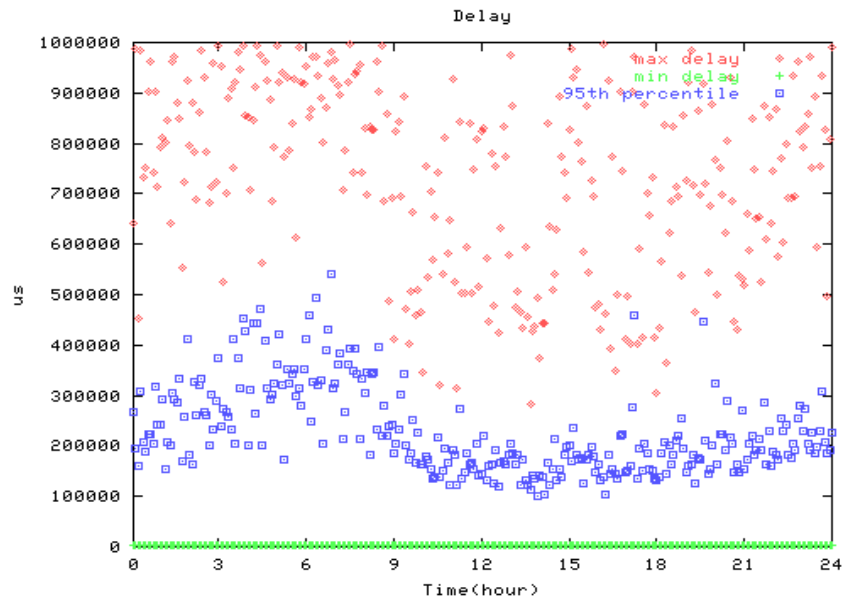


Fig. 9. One-way Delay from MS1 to MS2

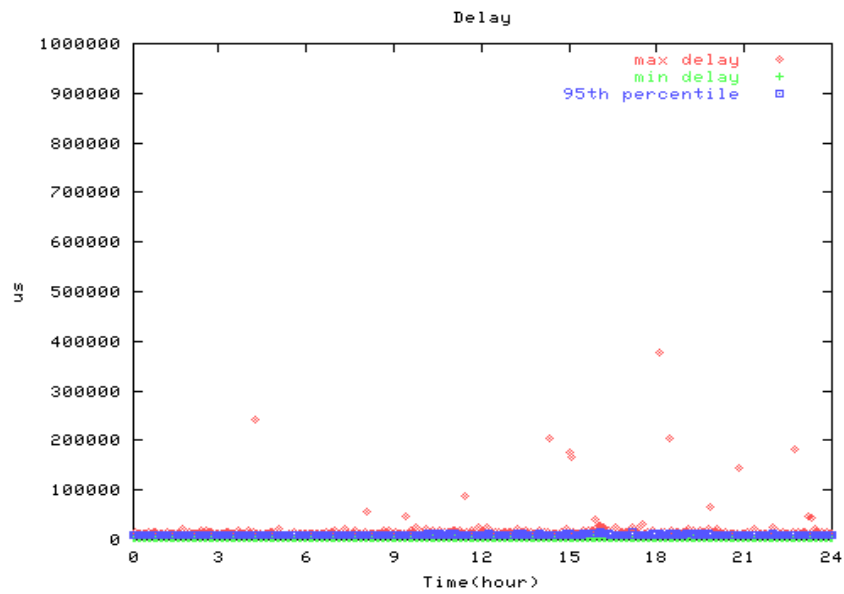
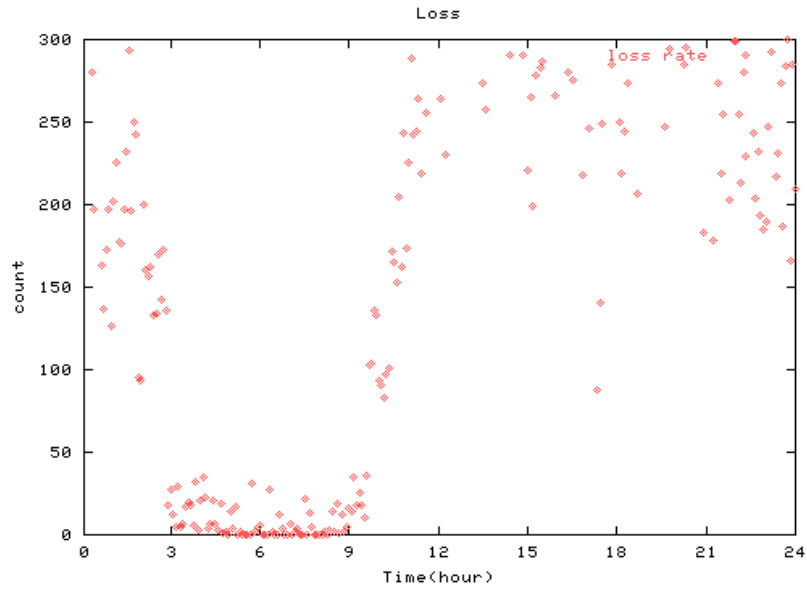
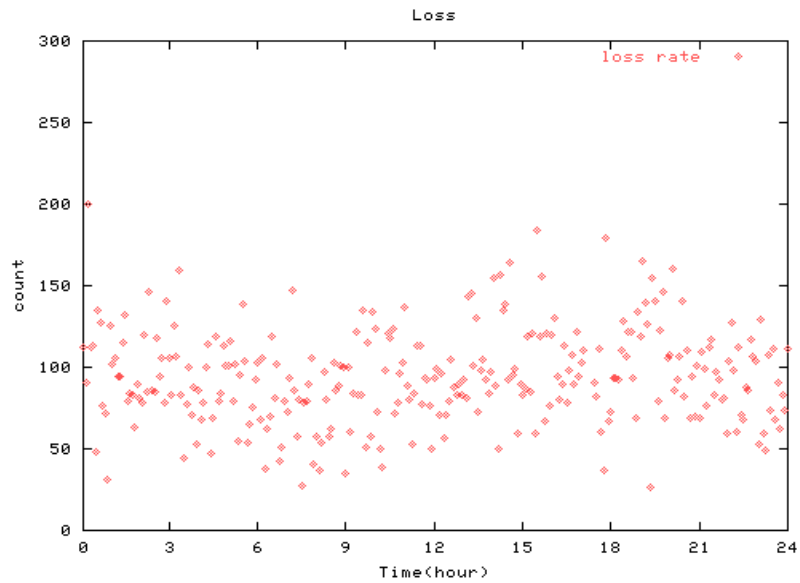


Fig. 10. One-way Delay from MS2 to MS1



**Fig. 11.** One-way Loss from MS1 to MS2



**Fig. 12.** One-way Loss from MS2 to MS1

Fig. 11 shows one-way loss from MS1 to MS2 (Loss1) and Fig. 12 shows one-way loss from MS2 to MS1 (Loss2). X-axis of graph is time. The unit is 5 minutes. Y-axis is one-way loss. The unit is the number of lost packets. We computed one-way loss by RFC 2680 [5]. We decided loss-threshold as 1[sec]. We consider a packet that has

bigger one-way delay than loss-threshold as a loss. We can see that Loss1 is from 0 to 295 and that Loss2 is from 26 to 180. As a result, we can see that the one-way path from MS1 to MS2 (Path1) has more variable and more one-way loss than that from MS2 to MS1 (Path2).

Through the above measurement, we can infer that Path1 may be more loaded than Path2 or that Path1 may have some problems (e.g., problem related to routing configuration). We can not find the above fact with 'Ping' that measures RTT between two end hosts. Like this, through one-way measurement we can get much useful information to grasp the state of network well for efficient network management

#### **4 Conclusion and Future Work**

One-way measurement is to be a kind of active measurement where measurement packets are injected in the path and are observed how they are served. With the measurement data obtained through active measurement, the network management can be performed effectively.

When we consider that the Internet is asymmetric, we can do active measurement better with one-way measurement tool such as Surveyor in order to grasp the state of network accurately than with two-way measurement tool such as Ping. One-way metrics that the IETF's IP Performance Metrics (IPPM) Working Group suggested are popularly used in one-way measurement. There are many cases where one-way measurement is useful. For example, if we have observed the network for a long time and have found that one-way delay increased much more than at ordinary times, we can guess that some problems have happened in the network. We can cope with the problems by resource relocation, load balancing through modification of routing configuration and so on.

In this paper, we suggested an architecture of measurement system (AMT: Active Measurement Tool) that can perform one-way measurement efficiently. We also described the procedure of measurement. We presented the analysis of result that we have obtained through measurement in the Internet including Korea Commercial Network (KORNET).

AMT is expected to be deployed in Korea Commercial Network (KORNET) and Asia Pacific Advanced Network (APAN) for the active measurement such as performance measurement at experiment related to QoS (e.g., DiffServ) and performance measurement of VoIP. We will also add some functions to AMT as follows; (a) Enhancement of function for self-troubleshooting, (b) Control of Control Server through Web, (c) Upgrade of AMT for IPv6 one-way performance measurement.

## References

1. V. Paxson, "End-to-End Internet Packet Dynamics", IEEE/ACM Transactions on Networking, Vol.7, No.3, pp.277 -292, June 1999.
2. V. Paxson, "Framework for IP Performance Metrics", RFC 2330, May 1998.
3. G. Almes et al., "A One-way Delay Metric for IPPM", RFC 2679, September 1999.
4. C. Demichelis and P. Chimento, "IP Packet Delay Variation Metric for IPPM", draft-ietf-ippm-ipdv-08.txt, November 2001.
5. G. Almes et al., "A One-way Packet Loss Metric for IPPM", RFC 2680, September 1999.
6. Skitter Home Page, <http://www.caida.org/tools/measurement/skitter/>
7. Surveyor Home Page, <http://www.advanced.org/surveyor/>
8. Sunil Kalidindi et al., "Surveyor: An Infrastructure for Internet Performance Measurements", presented at INET'99, San Jose, June 1999.
9. S. Shalunov et al., "A One-way Delay Measurement Protocol", draft-ietf-ippm-owdp-03.txt, February 2001.
10. Sunil Kalidindi, "OWDP Implementation, v1.0", Surveyor Technical Report 002.
11. FreeBSD Home Page, <http://www.freebsd.org/>
12. MySQL Home Page, <http://www.mysql.com/>
13. Motorola Home Page, <http://www.motorola.com/ies/GPS/>
14. NTP Home Page, <http://www.eecis.udel.edu/~ntp/>
15. David L. Mills, "Network Time Protocol (Version 3): Specification, Implementation and Analysis", RFC 1305, March 1992.
16. Gary R. Wright and W. Richard Stevens, "TCP/IP Illustrated, Volume 2: Implementation", Addison Wesley, 1995.