QoS Monitoring System on IP Networks

Marcelo Borges Ribeiro, Lisandro Zambenedetti Granville,
Maria Janilce Bosquiroli Almeida, and Liane Margarida Rockenbach Tarouco

Federal University of Rio Grande do Sul – Institute of Informatics
Av. Bento Gonçalves, 9500 – Bloco IV – Porto Alegre, RS – Brazil
{mribeiro, granville, janilce, liane}@inf.ufrgs.br

Abstract. At the current development stage of computer networks, the emergence of new applications that use the high performance available is unavoidable. In this context, any service that requires high performance also requires network QoS (Quality of Service). To help the maintenance of QoS services and QoS provisioning mechanisms, this work proposes a QoS monitor. This monitor operates on IP-based networks and its goal is to measure current QoS parameters observed on the network and compare them with the negotiated QoS parameters. Thus, network managers can be informed about degradations, and proceed with proper actions in order to provide adequate conditions to applications that require strict time warranties in order to operate properly.

1 Introduction

Nowadays, there is a clear need to have IP networks offer more appropriate services to transport differentiated information. Multimedia applications, for instance, have more demanding time restrictions than the usual networked applications (e-mail, web browsing, etc.). In order to make differentiation possible and to allow time-restricted applications to run properly, IP networks must offer services with some level of warranty, i.e., IP networks must feature QoS services.

However, having the user or the application request services, or having the service provider and other involved parties acknowledge user demands is not enough for the service to work effectively according to needs. It is also necessary to have processes to monitor QoS and the network manager must bear in mind that the desired QoS might not be the obtained QoS. This difference causes performance degradation of the applications, which no longer perform properly. Thus, the presence of monitoring mechanisms that inform the manager of QoS service performance is a real necessary.

This work presents a QoS monitoring architecture that uses monitors developed to operate on IP networks. With the aid of the QoS monitor, the network manager can determine monitoring policies for some network links, so as to be warned when any of them presents a difference greater than a preset limit. Because QoS monitoring is distributed for increased efficiency it is necessary to have a central element that gathers such information and compiles it so that the manager can understand it. The development of a central element that collects and processes data from network monitors and detects degradation is also presented.

This paper is structured as follows: section 2 discusses QoS monitoring on IP networks. Section 3 is concerned with the monitor-related architecture and section 4 explains how the QoS monitor operates and section 5 finishes the paper with conclusions and future work.

E.S. Al-Shaer and G. Pacifici (Eds.): MMNS 2001, LNCS 2216, pp. 222–226, 2001.

2 Related Work

Due to the increase of QoS needs, some monitoring solutions present ways to analyze traffic on the managed network. Waldbusser [1] has proposed the extension of the RMON MIB functionalities. With RMON2, the agent (probe) has its reach (monitoring) extended beyond the link level. An RMON2 probe can analyze packets by accessing information on the network and transport levels (TCP and UDP). This is very useful to have a monitoring activity by analyzing the media occupation of each application through the observation of the ports used. However, RMON2 probes cannot perceive degradation because they cannot check delay, jitter, or loss. Since these parameters are critical in QoS, RMON2-based QoS monitoring is a poor option.

Brownlee et al. [2] have proposed an architecture called RTFM (Real Time Flow Measurement) to measure and monitor flow on networks. A flow is the path of data that starts at a point A and ends at a point B. Because it depends on source and target, this flow measure is also appropriate to monitor the application level. The risk of running path-based monitoring between end points is that we are working on a packet-switch-oriented environment. This feature harms the idea of flow if only end points are considered because packets of the same flow may even switch places halfway, causing delays. At best, in case a performance loss is detected, it is not possible to determine the node that has caused it.

Other tool for monitoring bandwidth utilization proposed by Deri et al [3] is called Ntop. It was proposed to act like the „top“ command used in UNIX systems that shows the amount of memory used by processes. The main difference between all the monitoring application and Ntop is the concern about how its data are presented to the user. It features a web interface showing statistics about link occupancy, protocols utilization, machines involved, etc. Because it is not a distributed architecture, the concept of flow used in Ntop is different of the used (and needed) by other applications. The manager itself must provide all calculations with the information from every machines running Ntop in consequence of that it cannot monitor flows outside the local network, thus any flow formed by an external host cannot be used.

3 Architecture of the Monitoring System

As seen before, QoS monitoring is a need, and distributed monitoring is specially important to determine degradation points. Given this context, this section presents a monitoring system that is part of the QAME QoS management environment [4]. The QoS monitor is subdivided into a central element and monitor agents (fig. 1). This division is necessary because monitoring is distributed. The central element will receive the QoS parameters as policies from the user environment, find out which agents operate on the „area“ to be monitored and divide the monitoring flow into tasks for each relevant agent, according to its location.

3.1 Central Element

The central or centralizing element supplies a communication interface with the user environment. The central element, from a technical point of view, is a service that is run on a machine the manager interacts with. The central element has to receive the monitoring policies and determine the best strategy to collect the relevant data for that policy from the agents, compile this information and then send it to the network

manager. For this constant interaction mechanism to work, the central element not only has to wait constantly for network manager requests, but also to program the agent(s) at run-time. The central element is also responsible for assessing degradation of QoS parameters such as loss and delay, since it only accesses the aggregate data of the monitoring agents.

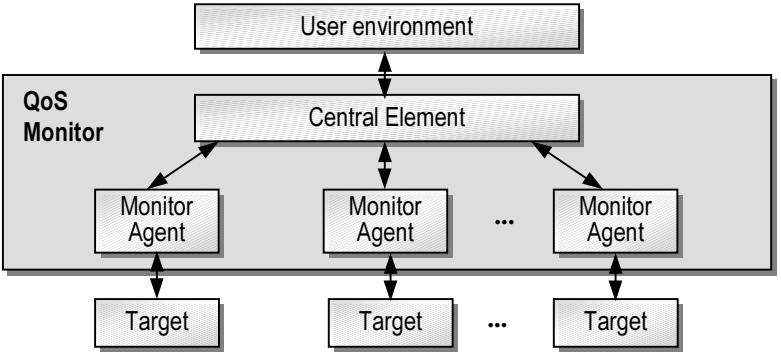

Fig. 1. QoS monitoring system

3.2 Monitoring Agents

The main tasks of a monitoring agent are to monitor flows, collect data and send information about the monitoring requests made by the central element. It is important to notice that monitors do not exchange messages with one another but with the central element only and they are able to inform about the link and current flow only.

If a critical flow has a smaller band than necessary, the monitoring agents associated to the problem node identify this condition and inform the central element. Another QoS parameter, which can be checked directly on agents, is jitter. When packets of the same flow or aggregate arrive at the interfaces of the monitored devices at irregular intervals, the variation is detected and the monitoring agent once again informs the central element of this event. Besides that, they are unable to process the collected information. They do not know what policy is under use at that time and, because they are limited to a sub-network, they would need data from other agents to generate any useful information, which is a task of the central element.

The degradation of other QoS parameters can only be detected at the central element. For instance, a packet delay is determined by an analysis of the departure time of this packet from the interface of a device and its arrival time at another interface of another device. This information is collected by accessing the monitoring agents located on each of the involved devices.

4 System Operation

Figure 2 shows an example of the operation of the QoS monitor in two sub-networks. The used network is made up of two collision domains which form segments A and B. Each segment has several workstations, represented by the squares. Two monitoring agents were placed on each segment, represented by the circles. Each

monitoring agent is, in fact, a workstation with a network interface operating on promiscuous mode.

The segments were connected to a main backbone through the use of two routers with at least two interfaces. Besides the routers, the backbone has a station which holds the central element of the QoS monitor, represented by the central rectangle. The dotted lines represent the indirect communications between monitoring agents and the central element.

Several IP flows/aggregates were monitored during the tests. To do so, each monitoring activity was defined by: sender and receiver IP addresses, sender and receiver ports, transport protocol and priority (represented by the DS field).

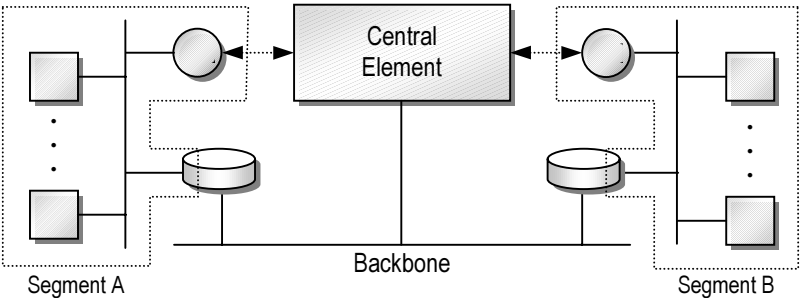

Fig. 2. Operation of the QoS monitor

The direct analysis on the monitoring agents allowed the assessment of flow bandwidth and jitter. The association of the information collected by the agents allowed the central element to verify delay and lost rate.

The analyses that were carried out were relevant, but limited by the type of implementation of the monitoring elements, which were, in this case, workstations with network interfaces in promiscuous mode. That limits the precision of the results obtained, since the verification time of the monitored packets on each agent is different from the real moment when the packet left the source, or arrived at the destination station. A more effective monitoring activity could be achieved if the monitoring agents were implemented directly on the routers.

The monitoring process starts when the network manager chooses a flow to be monitored (table 1). The management application must then send this information to the central element to have QoS monitoring started. As table 1 shows, the central element uses the monitoring policies to obtain enough information in order to distribute tasks to each agent that should be monitoring. The rows on table 1 are examples of the information the central element sends to an agent.

Table 1. Flows to be monitored

Sender IP	Receiver IP	Sender port	Receiver port	Protocol	DS field
192.168.0.101	192.168.0.4	1111	4444	TCP	7
192.168.0.101	192.168.0.23	2345	80	TCP	3

The network manager interacts with the central element only. When a monitoring policy is sent from the network manager to the central element it is broken down into simpler instructions, which are distributed to the monitoring agents that perform the

necessary measures within their monitoring scope. The collected data are then sent back to the central element, which processes the information and sends it to the network manager.

The only concern of the network manager is the communication with the central element (the whole QoS monitoring system, from the manager's viewpoint, is the central element only).

In case a data transmission service needs continuous flow, which may and most likely will go through more than one computer network, it is not enough to collect data on the client's end and to compare it with the sent data.

As previously discussed, performance loss may be observed throughout the entire „extension" of the transmission, i.e., monitors should be distributed along the transmission. This analysis causes a few problems. One must know which monitors are part of the monitored connection because it makes no sense to receive data from monitors that are not part of the connection under assessment. Besides that, the receipt of monitor messages has to be synchronized and the frequency of value sampling has to be chosen. A higher frequency results in a more precise analysis, but an excessive number of messages interferes with performance.

5 Conclusions and Future Work

QoS services are extremely important on a network, but very complex to design and install. The best effort paradigm of IP networks is not enough to ensure quality to performance-dependent services such as multimedia.

As the action to be taken most of the times a transmission degrades is already known (decrease the number of frames on a video, degrade sound quality on a song, etc.), what matters is to be aware of the link conditions of connections. The QoS monitor presented in this paper was designed for an environment that needs quality of service. We presented techniques to develop and install monitoring agents that would work as network „thermometers" by measuring QoS and submitting it to be compared with the hired value. This work also proposes a central element that „bridges the gap" between the network manager, via management application, and the agents, besides combining information gathered from several sources and giving the data meaningful interpretation.

References

1. Waldbusser,S.: Remote Network Monitoring Management Information Base Version2 using SMIv2. IETF RFC2021, (1997)
2. Brownlee, N.: Traffic Flow Measurement: Experiences with NeTraMet. IETF RFC 2123. (1997)
3. Deri, L.; Carbone, R; Suin, S.: Monitoring Networks Using Ntop Netikos S.p.A. (2001).
4. Granville, L. Z.; Tarouco, L.: QAME - An Environment to Support QoS Management Related Tasks on IP Networks. In: Proc. IEEE International Conference on Telecommunications - ICT (2001). Bucharest-Romania. (2001)