Approaches and Challenges for Guaranteeing Quality of Service in Next Generation Internet

Zoubir MAMMERI
IRIT - Paul Sabatier University
118, route de Narbonne, 31062 Toulouse - France
mammeri@irit.fr

Abstract: - The prime focus of today's Internet is on providing connectivity without assurance of service quality (QoS). However, many real-time applications such as teleconferencing and IP telephony require stringent QoS guarantees in delays and bandwidth which impose strict resource constraints on paths being used. In this paper we review the basic mechanisms and protocols to support QoS guarantees in the next generation Internet. We outline the various approaches and discuss their limitations and the challenges for the future. In particular, admission control, resource reservation, packet scheduling, and routing mechanisms are discussed.

Key-Words: - Quality of service, routing, scheduling, resource reservation, Internet, DiffServ, IntServ.

1 Introduction

Recently many interactive or real-time services have emerged to support new applications such as IP telephony, teleconferencing and many others. Transmitting real-time traffics is the greatest challenge in packet switching networks. The end-to-end transfer delay, the variation of transfer delay and packet loss must not exceed some limits otherwise the service being provided to the user may be severely disrupted.

The quality of service (QoS) is defined as set of quality requirements on the collective behavior of one or more objects in a network [7, 9, 11]. The main important parameters that define QoS are: delay (or end-to-end transfer delay), jitter (or transfer delay variation), throughput, and loss rate. Service availability and security also are other important aspects of QoS.

IP networks range in size from small clusters of routers situated within a given location, to thousands of routers, switches, and other components distributed all over the world. An *autonomous system* (AS) is a routing domain of Internet which has a common administrative authority and consistent internal routing policy. An AS may employ multiple intra-domain routing protocols internally and interfaces to other ASs via a common inter-domain routing protocol. The global Internet is composed of interconnected ASs.

Internet was mainly used for electronic mail and file transfer, which do not require the guarantee of timing constraints by the network. Contemporary Internet networks have three significant characteristics: (1) they provide real-time services, (2) they have become mission critical, and (3) their operating environments are very dynamic. The dynamic characteristics of IP networks can be attributed in part to fluctuations in demand, to the interaction between various network protocols and processes, to the rapid evolution of the infrastructure which demands the constant inclusion of new technologies and new network elements, and to transient and persistent impairments which occur within the system.

To achieve QoS in the Internet, two architectures have been proposed: Integrated Services (IntServ) and Differentiated Services (DiffServ). IntServ can potentially guarantee QoS on a per-flow basis. DiffServ achieves scalability by dealing with aggregates of flows, and seem to be more promising QoS technology in large scale networks.

Various different approaches have been proposed to provide QoS guarantees in switched networks in general and Internet in particular. They may be classified according to multiple criteria: connection oriented vs. connectionless, intra-domain vs. inter-domain, reservation-based vs. reservation-less, and static vs. dynamic.

The guarantee of QoS in packet switched networks, such as Internet, has many complex facets and requires the use of multiple mechanisms/functions: *admission control, negotiation and renegotiation, resource reservation, resource adaptation, packet classification, traffic shaping and conditioning, packet marking, flow policing, traffic scheduling, congestion control, routing, signaling protocols, QoS monitoring, QoS degradation and alert.*

The aim of this paper is not to provide an exhaustive review of existing mechanisms, products and platforms, but instead to give a perspective on the range of basic options available. The emphasis is on the need of adapting mechanisms to different situations and environments.

The rest of the paper is structured as follows: in section 2, we review and discuss the main mechanisms to provide QoS. In section 3, we review and discuss the main protocols and standards specified by the IETF for next generation Internet.

2 Mechanisms for QoS guarantee

QoS mechanisms work by controlling the allocation of network resources to application traffics in a manner that meets the application's service requirements. Although there are aspects of QoS that are taken into consideration in the design and implementation of every system, there is a wide spectrum of ways in which this can be done. At one extreme is the traditional 'static' approach to QoS, in which QoS is considered during the system design and configuration process and engineered 'statically' into the system. The use of multimedia and the extensive use of shared networks for many different and independent traffic streams, some of which may have stringent QoS requirements, are increasing the need for systems that can manage QoS *dynamically*. Such systems can respond to statements of QoS requirement, negotiate agreements about QoS, and then manage QoS by techniques such as resource reservation, routing, admission control, application adaptation, and so on. In consequence, QoS management encompasses a number of different functions, including static and dynamic aspects.

2.1 Traffic specification

The first thing to do is the specification of the traffic issued by users that the network should take into consideration. In the Internet, the specification of user traffic is currently done through a *TSpec* which is specified by means of parameters of a token bucket (a depth b and a rate r), a peak rate p, a minimum policed unit m and a maximum datagram size M [15]. The upper bound of the traffic in any time interval t is expressed in term of envelope $A(t)$ such that: $A(t) \leq min(M + pt, b + rt)$. Unfortunately, there are other types of traffics that cannot be specified using this model. Ideally, users want models that reflect with a high fidelity their needs. However, the number of traffic models should be kept low otherwise complex mechanisms are required to manage and control user traffics.

2.2 Admission control and resource management

Connection admission control (CAC). Admission control is the process to decide whether or not a new flow (or connection) should be admitted into the network. The main considerations behind this decision are current traffic load, current QoS, requested traffic profile, requested QoS, pricing and other policy considerations. The admission control must ensure that admitting a new flow does not result in violated QoS for the existing flows already inside the network. Sometimes, a negotiation process may be conducted between the user and the network to revise the requested QoS parameter values. If the network decides to accept a new flow, a QoS contract is established. The complexity of CAC algorithms depends on the complexity of traffic specification models. To have CAC algorithms with low complexity, the traffic models used must be simplified. Such a simplification may lead to the oversizing the network, and to the reject of connections that more accurate CAC algorithms should have admitted. Today, each autonomous system may have its own resource allocation policy, and there is no policy standard. Thus, many problems remain to solve to elaborate efficient CAC algorithms.

Resource reservation. The problem of allocating limited resources becomes even more complex if we consider that current computational systems are basically heterogeneous, subject to mobility and constant reconfiguration, but still have to provide a dependable and accurate service in a limited response time.

Resources (buffers and bandwidth) may be allocated in a deterministic fashion, in which case they are reserved for the activity in question, or statistically, in which case they are shared with other activities on the basis that the total available is estimated to be sufficient to meet all the needs, barring rare events. The resources may be allocated once and for all as part of the establishment phase, or they may be subject to re-allocation during the operational phase (i.e., dynamically).

Resource adaptation. The task of QoS management is to try to continue to meet the agreed user requirements in network overload conditions. The 'application adaptation' approach means the treatment of the degradations in the QoS available by providing a degraded, but still tolerable, service, and providing an improved service when higher QoS becomes available again. Reducing the QoS to the user may take many forms: for example, dropping some MPEG frames or changing picture size in video. QoS mechanisms have to be aware of the possibility of resource adaptation, making it transparent to the application whenever possible. When the agreed QoS is not reachable with the resources available, the application has to be informed that the agreed QoS has to be renegotiated. Applications holding resources that are subject to changes in their availability because of resource adaptation have to be able to degrade gracefully

when it occurs. It should be noted that application designers have to take into account resource adaptation earlier in the development of their applications; this is a new challenge for them.

2.3 Routing

Routing deployed in today's Internet is focused on connectivity and typically supports only one type of service, the best effort. Current Internet routing protocols (e.g. OSPF and RIP), use 'shortest path routing', i.e. routing that is optimized for a single arbitrary metric, administrative weight or hop count. QoS-based routing extends the current routing paradigm with mechanisms under which paths for flows are determined based on some knowledge of resource availability in the network as well as the QoS requirements of flows.

The main objectives of QoS-based routing are:

- dynamic determination of paths that have a good chance of accommodating the required QoS;
- optimization of resource usage;
- graceful performance degradation;
- keeping the current path as long as QoS requirements are met to avoid routing oscillations due to today's opportunistic routing;
- support for traffic using IntServ and DiffServ classes of services.

Routing is achieved at two levels: intra-domain routing and inter-domain routing. At the intra-domain level, the objective is to allow as much latitude as possible in addressing the QoS-based routing issues. Indeed, there are many ideas about how QoS-based routing services can be provisioned within an autonomous system (AS). These range from on-demand path computation based on current state information, to statically provisioned paths supporting a few service classes. The fundamental requirement on inter-domain QoS-based routing is scalability. This implies that inter-domain routing cannot be based on highly dynamic network state information. Rather, such routing must be aided by sound network engineering and relatively sparse information exchange between independent routing domains. This approach has the advantage that it can be realized by straightforward extensions of the present Internet inter-domain routing model. Support for QoS routing can be viewed as consisting of three major components: *metrics* (for collecting flow's requirements and characteristics, and information about the availability of resources), *advertisement of link state information*, and *path selection*.

QoS routing poses major challenges in terms of algorithmic design. On one hand, the path selection process is a complex task due to the need to deal with QoS requirements of connections. On the other hand, requests need to be handled promptly upon their arrival; hence there is limited time to spend on path selection. Some authors advise the precomputation of paths to reduce path selection [13]. Such an approach has limitations, because networks are highly changing environments.

Inaccuracy in the information used for computing QoS-aware routes arises naturally in number of environments. Guerin and Orda [8] proposed algorithms to select paths that are most likely successfully accommodate the desired QoS in the presence of uncertain network information. The proposed algorithms have a high overhead; that is the price to pay to increase routing scalability.

QoS-based routing in the Internet raises many questions that are not satisfactorily treated today; these questions are related to: usage efficiency of resources, granularity of routing decisions, routing metrics, performance objectives, mapping of QoS parameters between different autonomous systems, renegotiation of paths, fault-tolerance, routing overhead, administrative control, scalability, and interoperability between today's routing and QoS-aware routing.

QoS-based routing and resource reservation protocols. To simplify QoS routing, resource reservation and routing are sometimes combined as a single function. Nevertheless, there must clearly be a well-defined interface between routing and resource reservation protocols [6]. The nature of this interface, and the interaction between routing and resource reservation has to be determined carefully to avoid incompatibilities. The importance of this can be readily illustrated in the case of RSVP [4]. RSVP has been designed to operate independently of the underlying routing scheme. Under this model, RSVP 'Path' messages establish the reverse path for 'Resv' messages. In essence, this model is not compatible with QoS-based routing schemes that compute paths after receiver reservations are received. While this incompatibility can be resolved in a simple manner for unicast flows, multicast with heterogeneous receiver requirements is a more difficult case. For this, reconciliation between RSVP and QoS-based routing models is necessary. However, such a reconciliation may require some changes to the RSVP model depending on the QoS-based routing model.

2.4 Packet scheduling

The basic function of the scheduler is to allocate the output links to packets taking into account their constraints. The nature of scheduling employed greatly impacts the QoS guarantee that can be provided by the network. Routers mark traffics and use internal queuing mechanisms to enforce QoS guarantee. Most switches and routers enforce

priorities by assigning packets from different streams to different queues. Different types of traffic are held in different queues and traffics are served according to their priorities.

A variety of scheduling disciplines aimed at providing per-flow guarantees have been proposed. They mainly include Round Robin (RR), Virtual clock (VC), Weighted Fair Queuing (WFQ), Self Clocked Fair Queuing (SCFQ), Frame-Based Fair Queuing (FBFQ), Delay Earliest-Due-Date (DEDD), and Jitter Earliest-Due-Date (JEDD). Some of these service disciplines guarantee end-to-end transfer delay and, in certain cases, the delay jitter [5].

There are basically three types of queuing: priority, weighted, and class-based queuing. Routers using priority queuing classify traffic and set policies for high and low priority data. The high priority queues have to be emptied before lower priority traffic is transmitted. This approach works well for bursty traffic, but if policies aren't properly set then low priority traffic can be starved of bandwidth. This could lead to dropped packets and retransmission making the congestion problems worse. Another queuing approach is weighted fair queuing (WFQ). WFQ first ensures that there is enough capacity available for the low-bandwidth flows, and then splits the rest among the large-bandwidth flows. A primary goal of WFQ is to avoid bandwidth starvation for low-priority traffic. Finally, there is class-based queuing (CBQ) in which each queue is guaranteed a certain transmission rate. If the queue doesn't use all of its bandwidth, traffic from other classes can borrow as needed.

To provide some QoS guarantees, the same scheduling technique is implemented on all the routers of a given domain. Thus, the techniques (such as RR, VC, WFQ, SCFQ, FBFQ, DEDD, and JEDD) are all intended for scheduling within the same autonomous system. The only one approach that is intended for heterogeneous networks is CBQ (class-based queueing). CBQ provides aggregated service guarantees to a set of flows mapped into the same class.

The main challenges to guarantee QoS in heterogeneous and interconnected domains are:

- How to guarantee end-to-end QoS when packets traverse domains using different scheduling techniques?
- In the context of agreements between domains, one domain may guarantee bandwidth and another guarantees delay. Thus, how to translate QoS constraints of one type to another type to guarantee end-to-end QoS?

2.5 Congestion control

Since link speeds are often of several Mega bits per second, the amount of memory required to buffer traffic during transient periods of congestion can be large and exceed the amount of memory that routers/switches have. Some packets that arrive during congestion situation are dropped. To avoid haphazard behavior when a link experiences congestion, several different buffer management schemes have been defined, among which we have: EPD (Early Packet Discard) and RED (Random Early Discard) schemes where packets are discarded before the onset of congestion.

Generally, it is desirable to devise buffer management schemes that select the packets to drop that are the less important for the applications being supported and without always penalizing the same sources (fairness). The fairness of a buffer management scheme is a function of how it penalizes packets from non-conformant flows. Determining thresholds to control network congestion is not an obvious task because low thresholds may lead to the degradation of the network throughput. In a complex environment such as the Internet, network managers are tempted to fix thresholds with low values, by fear of congestion of their own network, but in doing so they limit the fluidity of the global Internet.

2.6 Traffic engineering

Traffic Engineering encompasses the application of technology and scientific principles to the measurement, characterization, modeling, and control of Internet traffic [1]. Traffic oriented performance measures include delay, delay variation, packet loss, and throughput. An important objective of Internet traffic engineering is to facilitate reliable network operations. This results in a minimization of the vulnerability of the network to service outages arising from errors, faults, etc.

A fundamental challenge in network operation, especially in a large scale public IP network, is to increase the efficiency of resource utilization while minimizing the possibility of congestion. It is not a question any more of making isolated decisions of network management, but a minimum of cooperation between the managers of neighboring networks is required to set up a global Internet with QoS providing. Large scale simulations and experiments are necessary to fix congestion thresholds. In today's Internet no significant (and commonly accepted) results exist to help domain managers to fix their parameters with a global view of a QoS-aware Internet; this is another challenge.

3 Protocols and standards

This section describes four protocols and standards specified by the IETF for providing QoS: IntServ, DiffServ, RSVP and MPLS.

3.1 Integrated Services (IntServ)

The integrated services (IntServ) model requires resources to be reserved a priori for a given traffic flow to ensure that the QoS requested by the traffic flow is satisfied [3]. It is important to notice that IntServ model is a flow-based model, i.e., resources are reserved to individual flows (or connections). With IntServ, each node is divided into two parts: background process and traffic forwarding. The background process takes care of routing, reservation setups and admission control. The traffic forwarding part classifies traffic based on information in the traffic control database and schedules the traffic based on this information.

In addition to the best effort service, two types of services are defined: *Guaranteed service* which provides absolute guarantees on the delay and loss [16], and *Controlled-load service* which provides service equivalent to that of an unloaded network [17]. A notable feature of the IntServ model is that it requires explicit signaling of QoS requirements from end systems to routers [18]. The RSVP Protocol performs this signaling function and is a critical component of the IntServ architecture.

Limitations of IntServ. The number of individual flows in a backbone network can be very large, and the number of control messages for making resource reservation for large number of flows can be large and may require a lot of processing power. Similarly, maintaining state information for all the flows can require a lot of storage capacity. Policy issues need to be resolved to determine who can make reservations. Similarly, security issues need to be resolved to ensure that unauthorized sources do not make spurious reservations. That is why it is believed that IntServ model is appropriate for small intranets where there are a small number of flows and where policy and security issues can be managed easily. Large backbone networks will need more scalable mechanisms for differentiating traffic and providing differentiated services.

3.2 Resource reservation protocol (RSVP)

RSVP (ReSource reserVation protocol) is specified as a signaling protocol to allocate resources from the network [4]. RSVP is independent from any architecture and can be used with a variety of QoS services. RSVP is used only for signaling; it does not deal with how the resources are actually reserved. In each router there is a module that reserves resources according to some policy specific to each router. The RSVP request is receiver initiated: this provides better scalability for large multicast receiver groups, more flexible group membership and diverse receiver requirements. The sender sends *Path* messages which record the route packets travel to receiver, and have traffic characterization information. On reception of a *Path* message the receiver sends a *Resv* message to reserve needed capacity from the network. This message travels hop-by-hop same route (other direction) the *Path* message traveled.

Disadvantages of RSVP

- RSVP is purely receiver based. The reservations are initiated by willing receivers. But in many cases, it is the sender who has the onus of initiating a QoS based flow.
- RSVP imposes maintenance of soft states at the routers. This implies that routers have to constantly monitor and update states on a per-flow basis. This increases the congestion probability.

3.3 Differentiated Services (DiffServ)

DiffServ model introduced the concept of 'aggregating flows' so that the number of flows in the backbone network remains manageably low [2]. The key features of DiffServ that overcome some of the limitations of IntServ are: (1) coarse differentiation, (2) no packet classification in the network, and (3) use of long-term static provisioning to establish service agreements with the users.

Traffic differentiation. There are two types of nodes with a DiffServ domain: boundary nodes and interior nodes. Boundary nodes (edge routers) connect the DiffServ cloud to other domains or end hosts to the network. Boundary nodes can be both ingress nodes and egress nodes depending on direction of traffic flow. Interior nodes (core routers) are connected to other interior nodes or boundary nodes - but they must be within the same DiffServ domain. The boundary nodes are assigned the duty of classifying ingress traffic so that incoming packets are marked appropriately to choose one of the Per Hop Behavior groups supported inside the domain. When data packets enter the DiffServ domain, they are classified, marked, shaped, and policed in the edge routers, typically on a per-user-flow basis. The packets that pass through the edges routers are marked as certain flow aggregates, each corresponding to a per-hop behavior (PHB). Each such aggregate is assigned a single DS (DiffServ) codepoint (i.e., one of the markups possible with the DS bits). The edge routers use the 8 bit ToS field in the IP packet header to mark the packet for preferential treatment by the core transit routers. Currently there are proposals for two PHB groups: *Assured Forwarding PHB* that provides reliable services to the users even in times of network congestion [10], and *Expedited Forwarding PHB*, which can be used to build a low loss, low latency, low jitter assured bandwidth, end-to-end service [12].

Service Level Agreements (SLAs). The resources are managed based on contracts between the neighboring DiffServ domains, called SLA (Service Level Agreements). The SLA may include traffic conditioning rules which (at least in part) constitute a Traffic Conditioning Agreement (TCA) and may specify rules such as for traffic remarking, actions to be taken for out-of-profile traffic etc. An SLA can be either qualitative (e.g. 'Traffic offered at service level L1 will be delivered with low latency') or quantitative (e.g. '90% of in profile traffic delivered at service level L2 will experience no more than 50 ms latency'').

DiffServ limitations

- Providing quality of service to traffic flows on a per-hop basis often cannot guarantee end-to-end QoS. Therefore, only premium service will work in a purely DiffServ setting.
- DiffServ assumes a static SLA configuration. But in the real world network topologies change fast.
- DiffServ is sender-oriented. In many flows, the receiver's requests have to be accounted for.
- Some long flows like high bandwidth videoconferencing require per-flow guarantees. But DiffServ only provides guarantees for flow aggregates.

It is worth noticing that many details regarding the realization of a DiffServ networks remain open and are the subject of debate.

3.4 Multi-Protocol Label Switching (MPLS)

The MPLS approach to IP QoS is different from DiffServ. MPLS has been proposed to be a combination of the better properties of ATM and IP. It proposes switching at the core based on labels on IP packets [14]. MPLS uses fields in the 4-byte label it adds to the IP packet. This label is intended to improve efficiency of the network and allow routers to forward packets using predetermined paths according to, among other things, specified QoS levels. At the edge of the MPLS network, a label is added to each packet containing information that alerts the next hop MPLS router to the packet's predefined path. As the packet traverses the network, it may be relabeled to travel a more efficient path. Upon leaving the MPLS network the packet is stripped of its label and restored to its original size. The labels are distributed by a dynamic Label Distribution Protocol (LDP) or by RSVP.

MPLS brings some mechanisms to provide QoS, and it may be integrated in DiffServ and IntServ architectures with several ways. Many problems related to this integration are still open (interaction between MPLS with lower layers, with resource reservation, with routing, etc.).

4 Conclusion

Currently there are two main efforts to provide control of QoS: IntServ and DiffServ architectures. It looks like there is a possible way to implement QoS: a combination where the DiffServ is used in the core networks and RSVP/IntServ in access network. There are many unanswered questions when it comes to determining the appropriate QoS mechanisms for each environment and this paper did not attempt to answer them, instead it tried to a emphasize their importance and the challenges for the development of new infrastructures for next generation Internet.

References:

[1] Awduche D. et al., Requirements for traffic engineering over MPLS, RFC 2702, IETF, 1999.

[2] Blake S. et al., An architecture for Differentiated Services, RFC 2475, IETF, December 1998.

[3] Braden R., et al., Integrated Services in the Internet architecture: an overview, RFC 1633, IETF, June 1994.

[4] Braden R. et al., Resource ReSerVation Protocol (RSVP), RFC 2205, IETF, September 1997.

[5] Cottet F., J. Delacroix, C. Kaiser, and Mammeri Z., *Scheduling in real-time systems*, Wiley, 2002.

[6] Crawley E. et al., A framework for QoS-based routing in the Internet, RFC 2386, IETF, Aug. 1998.

[7] Ferguson P., and Huston G., *Quality of service: delivering QoS on the Internet and in corporate networks,* Wiley Computer Publishing, 1998.

[8] Guerin R., and Orda A., QoS routing in networks with inaccurate information: theory and algorithms, *IEEE/ACM T.O.N.* 7(3):350-364, June 1999.

[9] Guerin R., and V. Peris, Quality-of-service in packet networks: basic mechanisms and direction, *Computer Networks*, 31(3):169-179, February 1999.

[10] Heinanen J. et al., Assured Forwarding PHB Group, RFC 2597, IETF, June 1999.

[11] ISO/IEC CD 13236-2, Quality of Service Framework - Part2: Basic Framework, July 1995.

[12] Jacobson V., et al., An Expedited Forwarding PHB, RFC 2598, IETF, June 1999.

[13] Orda A., and Sprintson A., QoS routing: the precomputation perspective, *In Proceedings of IEEE Infocom2000*, Tel Aviv, March 2000.

[14] Rosen E., et al., Multiprotocol Label Switching Architecture, RFC 3031, IETF, January 2001.

[15] Shenker S., and Wroclawski J., General characterization parameters for Integrated Service network elements, RFC 2215, IETF, Sep. 1997.

[16] Shenker S., Partridge C., and Guerin R., Specification of guaranteed quality of service, RFC 2212, IETF, September 1997.

[17] Wroclawski J., Specification of the controlled-load network element service, RFC 2211, IETF, Sep. 1997.

[18] Yavatkar R., Pendarakis D., and Guerin R., A framework for policy-based admission control, RFC 2753, IETF, January 2000.