Quality of Service on the Internet: Evaluation of the IntServ Architecture on the Linux Operating System

Elisabete Reis
elreis@dei.uc.pt
Polytechnic Institute of Coimbra

Fernando Melo
fmelor@dei.uc.pt
Polytechnic Institute of Guarda

Marília Oliveira, Gonçalo Quadros, Edmundo Monteiro
{marilia,quadros,edmundo}@dei.uc.pt
Tel: +351-239-790018 Fax: +351-239-701266
Communication and Telematics Laboratory
Center for Informatics and Systems of the University of Coimbra
Pólo II, Pinhal de Marrocos, 3030 Coimbra

Preference for oral presentation – Networks and Multimedia

Abstract[1]

Recent Linux kernels offer a wide variety of traffic control functions[2]. These cover the mechanisms required to support the Integrated Services architecture developed by the IETF[3]. This document analyses the ability of Traffic Control enabled routers, under the Linux operating system, to provide the desired service.

Key Words

Quality of Service, Integrated Services, Linux.

1 Introduction

Today's Internet provides only a best-effort service, where all traffic is equally treated and there are no delivery guarantees. However, emerging applications have very diverse Quality of Service (QoS) expectations and thus, the current Internet model becomes inadequate and limiting. One proposal to support QoS on the Internet is the IETF Integrated Services[4] (IntServ) approach. This proposal specifies the Guaranteed Service [Shenker1997] and the Controlled-Load Service [Wroclawski1997]. Its implementation requires that network elements contain the following modules: Admission Control, Classifier and Scheduler of packets. These modules constitute the Traffic Controller.

Under the Controlled-Load Service, applications are treated as if it is deployed a best-effort service with light loads. This service is designed for adaptive real-time applications that can tolerate variance in packet delays. The Guaranteed Service provides an assured level of bandwidth, with firm bounds on end-to-end queuing delays, and guarantees that packets will not be discarded due to queue overflows. This service is intended for applications that are highly sensitive to delays, such as real-time video. The specification of IntServ needs a signalling mechanism such that applications can require the appropriate QoS level. The *Resource Reservation Protocol* (RSVP) [Braden1997] is the standard signalling protocol for the IntServ architecture.

2 Validation and Sharing Throughput Tests

In this paper we describe the capability of Linux Traffic Control to support the Integrated Services architecture. We used a testbed with two end hosts connected through three serially interconnected routers. All computers run the Linux operating system (RedHat 5.2, with kernel 2.2.7), and have RSVP installed. The Traffic Control modules were installed in all routers. The discipline Class-Based Queuing[5] (CBQ) was activated and configured on the output interfaces of all routers. In the tests we generated several combinations of data flows, belonging to the QoS classes considered (guaranteed, controlled-load and best-effort). The evaluation of the services

[1] *Trabalho parcialmente financiado pelo Programa de Investigação PRAXIS XII do Ministério da Ciência e Tecnologia, sob o contrato PRAXIS/P/EEI/10168/1998 (Projecto QoS II – Qualidade de Serviço em Sistemas de Comunicação por Computador).*
[2] *<kuznet@ms2.inr.ac.ru>*
[3] *<http://www.ietf.org>*
[4] *< http://www.ietf.org/html.charters/intserv-charter.html>*
[5] *Linux Traffic Control supports some queuing disciplines, including Class Based Queueing (CBQ), Priority, Stochastic Fair Queueing (SFQ), and Random Early Detection (RED). We used the CBQ discipline because it is the only that has the necessary features to interact with RSVP.*

provided consisted on the measurement of the QoS parameters that reflect the characteristics of the services, namely delay, loss rate and bandwidth.

The first tests concerned the validation of the Admission Control and Packet Classifier mechanisms. The Admission Control module showed a satisfactory behaviour. However, we verified that the bandwidth provided by the CBQ discipline to the IntServ class was not completely used by the flows with reservation. The evaluation of the Classifier was successful, since we were able to verify that the packets were put on the correct queues.

In the second set of tests we evaluated the ability of the CBQ discipline to guarantee, to each class, the bandwidth that was previously allocated, under different traffic loads. This mechanism had the appropriate behaviour.

The borrowing mechanism was also evaluated. When this mechanism is activated for a class, that becomes "not bounded", the class should be able to use the available bandwidth not necessary for other classes. However, we verified that when the borrowing mechanism was activated for flows with reservation, these flows did not benefit from it.

3 Performance Measurements and Analysis

The guarantees provided by the Guaranteed Service (maximum queuing delay and absence of losses due to queues overflow) depend on the reserved bandwidth and token bucket depth. There must be a trade-off in the definition of these parameters. If we want to have low delays, we must use shorter queues; however this can induce a higher loss rate. On the other hand, if we lower delay by increasing the reserved bandwidth, resource wasting may occur, limiting the number of QoS flows that can be accepted. In order to evaluate the relevance of these parameters, we conducted two sets of tests. In the first set, we allocated various levels of bandwidth to guaranteed flows, for the same bucket depth. In the second set, the bandwidth allocated was kept constant, and we varied the bucket depth.

The analysis of the results of these tests showed that, in many cases, the maximum delays exceeded significantly the maximum delays expected according to the service specification. We are presently conducting tests in order to explain this unpredicted behaviour.

We also evaluated the service provided to flows using the Controlled-Load Service. Since this service does not give firm bounds on delay and losses, we did a qualitative evaluation. This was achieved comparing the service degradation given to these flows, under the different traffic levels induced in the test network.

4 Conclusions and Future Work

In this paper we presented the analysis of the performance of the IntServ architecture on Linux. We verified that the classes received the allocated bandwidth. Nevertheless, the classes with reservation did not benefit form the borrowing mechanism. The major drawback of the tested implementation concerns the guaranteed service and relates to the fact that the maximum queuing delay has unacceptable values according to the service specification.

Future work will include a more detailed evaluation of the Controlled-Load Service. The evaluation of the Integrated Services architecture when the Controlled-Load Service is used with the Guaranteed Service will also be performed.

References

[Braden1997] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, "Resource ReServation Protocol (RSVP) – Version 1 Functional Specification", Request For Comments 2205, September 1997.

[Shenker1997] S. Shenker, C. Partridge, R. Guérin, "Specification of Guaranteed Quality of Service", Internet Engineering Task Force, Network Working Group, Request For Comments 2212, September 1997.

[Wroclawski1997] J. Wroclawski, "Specification of the Controlled-load Network Element Service", Internet Engineering Task Force, Network Working Group, Request For Comments 2211, September 1997.