A Token Bucket Model with Assured Forwarding for Web Traffic

Salvador Alcaraz[1], Katja Gilly[1], Carlos Juiz[2], and Ramon Puigjaner[2]

[1] Miguel Hernández University,
Departamento de Física y Arquitectura de Computadores,
Avda. del Ferrocarril, 03202 Elche (Spain)
{salcaraz,katya}@umh.es
[2] University of Balearic Islands,
Departament de Ciències Matemàtiques i Informàtica,
Carretera de Valldemossa, km 7.5, 07071 Palma de Mallorca (Spain)
{cjuiz,putxi}@uib.es

Abstract. In this paper we present PLF (Promotion of Long Flows). PLF tries to promote web traffic using the Token Bucket Model in a DiffServ framework. This algorithm preserves the high priority for short flows but tries to allocate some long flows in the high priority class level, in order to improve some performance parameters of the long ones. Finally, we present PLFwp (Promotion of Long Flows with Penalization) in order to detect (and cancel) the effect of these extremely long flows over the global performance. We analyze the results for packet loss and web transmission latency.

Keywords: Web traffic, DiffServ, Token bucket, QoS, Short and long flows.

1 Introduction

The best effort service model of the Internet, where all packets and flows have equal status is not being able to provide packet delivery guarantees. This model is unsuitable for applications with Quality of Service (QoS) constraints. The early 1990s saw a large number of frameworks being proposed for supporting QoS over the Internet.

The Integrated Services Model (IntServ) [1] was one of the IETF first proposal to achieve QoS over the Internet traffic. In this model, routing devices are required to hold the status information of each flow going through that device. One single reason why IntServ has not been accepted in the Internet is its absence of per-flow QoS scalability beyond the Intranet environment. Typically more than 250,000 simultaneous flows pass through Internet Core routers. Maintaining a state for such a large number of flows requires computing resources.

IETF proposed another framework, the Differentiated Services Model (DiffServ) [2] that could support a scalable form of QoS. DiffServ operates at class level, where a class is an aggregate of many such flows. For example, packets

T. Enokido, L. Barolli, and M. Takizawa (Eds.): NBiS 2007, LNCS 4658, pp. 298–307, 2007.
© Springer-Verlag Berlin Heidelberg 2007

coming from a set of source addresses or packets of a certain size, may fall into one class. DiffServ architecture adheres to the basic Internet philosophy, where the complexity is relegated to the Edge device while preserving simplicity of the Core device. Per-hop behaviors (PHBs) have been standardized into two classes by the IETF: *Expedited Forwarding (EF)* [3] and *Assured Forwarding (AF)*[4]. The main goal of the PHB-AF is to deliver the packets reliably. The PHB-AF is suitable for non-real time services such as TCP applications.

Several strategies have been proposed to achieve some kind of treatment for web traffic. Due to the web traffic's nature and the fact that the most of web traffic flows are short, in [5] the authors define a preferential treatment for short over long flows. In [6], QoS using control theory and predictability are applied both over DiffServ framework. In [7] uses RED Active Queue Management for provides better network performance for short-lived web traffic. In [8,9] the authors propose solutions based on load balancing and admission control.

In this paper, we propose a new algorithm (PLF, *Promotion of Long Flows*). It is in a DiffServ framework and PHB-AF, based on flow size, with a special handle of short flows in order to improve its performance for QoS. It also tries to achieve a certain QoS over the rest of long flows. We use the Token Bucket Model to assess the amount of priority available bandwidth to allocate long flows in the high level priority class.

The remainder of this paper is organized as follows: section 2 includes a short description of our architecture with DiffServ framework, Edge and Core devices and priority queueing scheduling. Section 3 describes the traffic and workload model. PLF algorithm is described in section 4. Afterwards, we present different performance metrics that we have analyzed. In section 6 we present another variant and the performance results. Finally, some concluding are presented.

2 System Architecture

Our experiments are based on a dummbell architecture (Fig. 1 (a)), where we can distinguish: *(i) Client area*, where the HTTP traffic is generated by HTTP client with HTTP request to HTTP servers, at the other side of the system; *(ii) Servers area*, where HTTP servers attend the incoming request from client; and *(iii) DiffServ area*, with the Edge router and Core router. The Edge router classifies packets by marking them and the Core router forwards/drops packets.

PHB-AF introduces two components in the operation of the DiffServ area: a packet marking mechanism administered by profile meters or traffic conditioners at Edge devices and a queue active management at Core devices. The packet marking mechanism monitors and marks packets according to the service profile at the Edge of a network. If the incoming traffic conforms the service profile, the packets are marked with a high priority and receive better service. Otherwise, the packets belonging to the non-conform part of a flow are marked with a low priority and receive low priority at the Core device. The Edge router classifies packets in function of the algorithm implemented, i.e. the action to perform with incoming packets: to drop them, to queue them in the queue-IN or in the