Available online at www.sciencedirect.com
ScienceDirect
Computer Networks 51 (2007) 1815–1832
Computer Networks
www.elsevier.com/locate/comnet

Determining an appropriate sending rate over an underutilized network path

Pasi Sarolahti [a,*], Mark Allman [b], Sally Floyd [b]

[a] *Nokia Research Center, P.O. Box 407, FI-00045 Nokia Group, Helsinki, Finland*
[b] *ICIR/ICSI, 1947 Center Street, Suite 600, Berkeley, CA 94704-1198, USA*

Available online 29 November 2006

Abstract

Determining an appropriate sending rate when beginning data transmission into a network with unknown characteristics is a fundamental issue in best-effort networks. Traditionally, the slow-start algorithm has been used to probe the network path for an appropriate sending rate. This paper provides an initial exploration of the efficacy of an alternate scheme called *Quick-Start*, which is designed to allow transport protocols to explicitly request permission from the routers along a network path to send at a higher rate than allowed by slow-start. Routers may approve, reject or reduce a sender's requested rate. Quick-Start is not a general purpose congestion control mechanism, but rather an *anti*-congestion control scheme; Quick-Start does not detect or respond to congestion, but instead, when successful, gets permission to send at a high sending rate on an underutilized path. Before deploying Quick-Start there are many questions that need to be answered. However, before tackling all the thorny engineering questions we need to understand whether Quick-Start provides enough benefit to even bother. Therefore, our goal in this paper is to start the process of determining the efficacy of Quick-Start, while also highlighting some of the issues that will need to be addressed to realize a working Quick-Start system.

Keywords: TCP; Congestion control; Explicit feedback

1. Introduction

A fundamental aspect of communication in general-purpose, best-effort packet-switched networks is determining an appropriate *sending rate*. The appropriate sending rate depends on the characteristics of the network path between the two peers (bandwidth, propagation delay, etc.), as well as the amount of load being placed on the network by competing traffic at the given time. Traditionally, TCP [20] has used a set of congestion control algorithms for determining an appropriate sending rate [11]. The rate is controlled using a congestion window (*cwnd*), which is an upper bound on the amount of unacknowledged data that can be injected into the network.

TCP's traditional method for determining the capacity of a network path with unknown characteristics is to use the *slow start* algorithm [11], which

* Corresponding author. Tel.: +358 50 4876607; fax: +358 7180 36850.
E-mail addresses: pasi.sarolahti@iki.fi (P. Sarolahti), mallman@icir.org (M. Allman), floyd@icir.org (S. Floyd).

1389-1286/$ - see front matter
doi:10.1016/j.comnet.2006.11.006

initializes *cwnd* to between one and four segments and then increases *cwnd* exponentially during each subsequent round-trip time (RTT) of the connection. In the best case slow-start takes $\log_2 N - 1$ RTTs and requires sending $N - 3$ packets before reaching a *cwnd* of N packets [11]. When there is contention for resources along the network path, slow start is a reasonable procedure. However, over underutilized paths that could support large congestion windows, possibly allowing an entire data transfer to be sent in one RTT, slow start can take much time, and require much data to be transmitted before achieving the desired sending rate.

In this paper, we provide an initial investigation of the usefulness of setting the initial sending rate using *Quick-Start*, a mechanism that allows a sender to advertise a desired sending rate, while the network can approve, reject or reduce the requested rate. While Quick-Start is designed to be used with a range of transport protocols, in this paper we consider its use with TCP. When using Quick-Start, a TCP sender may use the SYN packet to advertise a desire to transmit at X bytes/s. Each hop along the path may (i) explicitly approve the rate request in the SYN, (ii) explicitly reject the connection's use of a higher-than-standard initial sending rate, (iii) reduce the rate from X to some X' or (iv) do nothing, which implicitly prevents the connection's use of a higher-than-standard initial sending rate. Assuming some rate request X' arrives at the receiver, that rate is echoed back to the sender in the ACK of the SYN. The sender can then determine if all routers along the path approved the rate request, and if so, the sender can fairly safely transmit at X' bytes/s. If the request is rejected the sender will fall back to standard slow-start. As outlined in Section 3, routers supporting Quick-Start do not reserve bandwidth, and do not promise that the approved rate will be available one round-trip time later. Rather, routers "allocate" aggregate Quick-Start bandwidth only in the sense that this allocation is used by the router in deciding whether to grant future Quick-Start requests. Connections are not guaranteed the capacity "allocated"—though steps are taken in the allocation process to try to make failure a rare event.

This paper makes a number of contributions, as follows. (i) We present the first, if preliminary, well-rounded evaluation of Quick-Start. (ii) While alternate faster-than-slow-start schemes have been proposed, Quick-Start is the first scheme to allow a large data transfer in the first round-trip time after connection set-up, explicitly involving all nodes along a network path in arriving at an explicit appropriate sending rate. (iii) We introduce the notion of *anti-congestion control*. In other words, Quick-Start only provides a quick check to determine whether a network with unknown conditions is underutilized (uncongested) and permits a large initial sending rate. Quick-Start does not attempt to control the sending rate over the lifetime of a connection, but rather yields to standard congestion control for that task. (iv) We introduce and explore the notion of rate requests for best-effort traffic. (v) Because Quick-Start is so explicit and inclusive in choosing an initial sending rate, the scheme can serve as a baseline for evaluating alternate schemes.

This paper represents only a start to the evaluation of the costs and benefits of Quick-Start. Before Quick-Start could see wide use, a variety of questions need to be answered. This paper makes some assumptions that could not be made in the real world; for example, while Section 7 briefly discusses deployment issues such as interactions with middleboxes, IP tunnels, or non-IP queues, we do not address these issues in this paper; these issues are discussed in some detail in [9]. We investigate Web transfers, focusing on medium-sized flows that are shown to get the most benefit from using Quick-Start, and assume that the TCP sender is able to determine the desired sending rate for the Quick-Start request at the time when TCP connection is being established, based on the amount of data that is going to be sent. The assumptions made in the paper are not intended to minimize the required effort needed to realize a working Quick-Start system. Rather, the assumptions are part of the process of understanding the potential usefulness of Quick-Start separately from puzzling through the array of details that need to be nailed down for a Quick-Start deployment.

While our conclusion is that Quick-Start's benefits make it an attractive area for future work we are not convinced that Quick-Start would be feasible for the global Internet. However, many smaller (but, not small) networks that are within a single administrative domain—and therefore are not subject to the same concerns present on the global Internet—may find Quick-Start to be an attractive mechanism. For instance, [18] shows that within one particular enterprise typical network utilization is 2–3 orders of magnitude less than the raw capacity of the network and therefore Quick-Start could help applications to better use these untapped resources. Further, [2] notes that within long-delay satellite networks faster slow start is desirable.