Chapter 12
Random Early Discard (RED) Queue Evaluation for Congestion Control

Md. Shohidul Islam
Dhaka University of Engineering & Technology, Bangladesh

Md. Niaz Morshed
Dhaka University of Engineering & Technology, Bangladesh

Sk. Shariful Islam
Dhaka University of Engineering & Technology, Bangladesh

Md. Mejbahul Azam
Dhaka University of Engineering & Technology, Bangladesh

ABSTRACT

Congestion is an un-avoiding issue of networking, and many attempts and mechanisms have been devised to avoid and control congestion in diverse ways. Random Early Discard (RED) is one of such type of algorithm that applies the techniques of Active Queue Management (AQM) to prevent and control congestion and to provide a range of Internet performance facilities. In this chapter, performance of RED algorithm has been measured from different point of views. RED works with Transmission Control Protocol (TCP), and since TCP has several variants, the authors investigated which versions of TCP behave well with RED in terms of few network parameters. Also, performance of RED has been compared with its counterpart Drop Tail algorithm. These statistics are immensely necessary to select the best protocol for Internet performance optimization.

DOI: 10.4018/978-1-4666-0203-8.ch012

INTRODUCTION

AQM is receiving wide attention as a promising technique to prevent and avoid congestion collapse in packet-switched networks. It is a form of router queue management based on a proactive approach. By providing advanced warning of incipient congestion, end nodes can respond to congestion before router buffer overflows and hence ensure improved performance (Jain, 1990). In this chapter, we represent a comprehensive performance analysis of Random Early Detection (RED) algorithm. We show that RED provides fair sharing of the bottleneck link capacity and avoids phenomena such as lock-out inherent in Tail Drop. The RED algorithm tends to drop packets from each connection in proportion to the transmission rate the flow has on the output link. Moreover, it can serve incoming traffic guaranteeing stable results as the data load and arrival conditions vary. We investigate how high priority User Datagram Protocol (UDP) traffic affects the performance of lower priority Transmission Control Protocol (TCP) traffic when they share the same bottleneck link with one or two classes of service. The RED algorithm does not minimize the number of dropped packets as expected, but it manages to achieve improved performance when compared to the Tail Drop. Moreover, we found out that even though the arrival of competing UDP traffic generally hurts the performance of the TCP connection running, Tail Drop provides better results to the TCP connection when the UDP flow starts simultaneously with the TCP traffic. Indeed, it prevents the Slow Start phase from being excessively aggressive thus avoiding phenomena such as Slow Start overshoot and severe congestion states. Our results confirm most of the theoretical properties of the RED algorithm. It is substantiated that RED will be useful to provide fair sharing of resources and improved performance in a wide range of environments, including a variable number of connections with different data loads and throughputs.

LITERATURE REVIEW

Active queue management has been recommended by the Internet Engineering Task force (IETF) as a way of mitigating the above stated performance limitations of TCP over drop tail networks. Random Early Detection (RED) is the first active queue management algorithm proposed for deployment in TCP/IP networks (Floyd, Jacobson, 1999). The basic idea behind an active queue management algorithm is to convey congestion notification early to the TCP endpoints so that they can reduce their transmission rates before queue overflow and sustained packet loss occur. It is now widely accepted that RED controlled queue performs better than a drop-tail queue. However, RED has some parameter tuning issues that need to be carefully addressed for it to give good performance under different network scenarios. Random early detection (RED), also known as Random Early Discard or Random Early Drop is an active queue management algorithm (Floyd, Jacobson, 1993). It is also a congestion avoidance algorithm. In the traditional tail drop algorithm, a router or other network component buffers as many packets as it can, and simply drops the ones it cannot buffer. If buffers are constantly full, the network is congested.

Tail drop distributes buffer space unfairly among traffic flows. Tail drop can also lead to TCP global synchronization as all TCP connections “hold back” simultaneously, and then step forward simultaneously. Networks become underutilized and flooded by turns. RED addresses these issues. It monitors the average queue size and drops (or marks when used in conjunction with ECN) packets based on statistical probabilities. If the buffer is almost empty, all incoming packets reaccepted. As the queue grows, the probability for dropping an incoming packet grows too. When the buffer is full, the probability has reached 1 and all incoming packets are dropped. RED is more fair than tail drop, in the sense that it does not possess a bias against bursts traffic that uses