****

ATM Forum Document Number: ATM_Forum/96-0177

****

Title: TBE and TCP/IP traffic

****

Abstract:

The effect of TCP traffic over ATM ABR is studied with the ERICA switch algorithm. ABR implements its rate-based traffic control at switches (via ER algorithms like ERICA) and at sources (via source rules, such as, Rule 6, which uses TBE parameter). TCP implements its own traffic controls via slow start window control. We study the interaction between the two mechanisms. In particular, this contribution concentrates on the effect of Transient Buffer Exposure (TBE) parameter and Source End System Rule 6 on TCP/IP connections in a Wide Area Network (WAN).

****

Source:

Raj Jain, Shiv Kalyanaraman, Rohit Goyal, Sonia Fahmy, and Fang Lu
The Ohio State University

Raj Jain is now at Washington University in Saint Louis, jain@cse.wustl.edu http://www.cse.wustl.edu/~jain/

Saragur Srinidhi
NASA Lewis Research Center and Sterling Software
Scientific Systems Division
21000 Brookpark Road, MS 141-1
Cleveland,OH 44135
Phone: 216-433-8987, Fax: 216-433-8000

The presentation of this contribution at the ATM Forum is sponsored by NASA.

****

Date: February, 1996, Los Angeles

****

Distribution: ATM Forum Technical Working Group Members
(Traffic Management)

****

****

Given the popularity of TCP/IP, it is important to verify that all source, switch, and destination rules specified for ABR perform as expected for TCP/IP traffic. We intend to do a thorough study of various rules and their associated parameters.

This contribution concentrates on the Transient Buffer Exposure (TBE). Source End System rule 6 states requires that a source reduce its ACR if the source does not receive a backward RM cell after having sent TBE cells (or CRM=TBE/Nrm RM cells). This is a source-based control such that the source reduces its load without explicit instructions from the network.

TCP's CONGESTION MECHANISM:

TCP is one of the few transport protocols, which has its own congestion control mechanisms. Compared to other transport protocols, it relies less on network based control mechanisms. The key TCP congestion mechanism is the so called "Slow start." TCP connections use an end-to-end flow control window to limit the number of packets that the source sends. Whenever a TCP connection loses a packet, the source does not receive an acknowledgement and it times out. The source remembers the window value at which it lost the packet by setting a threshold variable SSTHRESH at half the window. The source resets the window (called congestion window in TCP) to one.

The source then retransmits the lost packet and increases its window by one every time a packet is acknowledged. We call this phase "exponential increase phase" since the window when plotted as a function of time increases exponentially. This continues until the window is equal to SSTHRESH. After that, the window w is increased by 1/w for every packet that is acked. This is called "linear increase phase" since the window graph as a function of time is approximately a straight line. After the window reaches the maximum window size (specified by the destination based on its buffer), the window remains constant. We call this the "steady-state."

Reducing window to 1 on a packet loss is similar to reducing the ACR on not receiving the backward RM cells. Both control loops are source-based and so it is important to study the interaction between the two.

Note that TCP's congestion control mechanism does not respond if there is no loss (assuming that the RTT estimators don't trigger false timeouts). The retransmission algorithm retransmits all the packets starting from the lost packet, besides reducing the window and threshold size.

SOURCE MODEL:

For the initial simulations that we have done, we used an infinite source model at the application layer in the sense that the TCP always has a packet to send as long as its window will permit it. We find that in spite of the infinite source application, the traffic seen by the ATM network is sometimes bursty and continuous at other times.

Whenever the network drops a packet, TCP stops putting additional load on the network. Only after the retransmitted packet reaches the destination and is acked, the source increases its window. Thus, the path is practically cleared of all packets from that connection (and becomes idle for one round trip unless there is other traffic). Once the ack reaches the source, the source

starts sending additional traffic and enters the exponential rise phase. During this phase, the ATM network sees a burst of traffic. Once the TCP layer reaches the maximum window, there is a continuous flow of traffic at all layers and the ATM layer's load is similar that for infinite ATM sources.

TCP OPTIONS:

We use a TCP maximum segment size (MSS) of 512 bytes. The MTU size used by IP is generally 9180 bytes and so there is no segmentation caused by IP. We implemented the window scaling option so that the throughput is not limited by path length. Without the window scaling option, the maximum window size is 2**16 bytes or 64 kB. We use a window of 16x64 kB or 1024 kB. The network consists of three links of 1000 km each and therefore has a one-way delay of 15 ms (or 291 kB at 155 Mbps). In our initial simulations, we have not implemented "fast retransmit and recovery." This will be included later.

TCP PERFORMANCE WITHOUT BACKGROUND TRAFFIC:

If there is no background traffic, the network capacity is constant. The TCP sources may loose a few packets initially but soon enter the steady state. In this state, the load entering the network is limited by the maximum window size and ACR granted by the network. We found that with proper (congestion avoiding) switch algorithm like ERICA [1], the queues in the switches are small (close to 1). Most of the cells are waiting at the source itself. The source queues are long and depend upon the maximum window size and the path length.

The ABR parameters, like TBE, have no effect in this case since rule 6 is not triggered.

In an explicit rate-based ABR network, the network can respond to source activity within one feedback delay. The feedback delay is the time between the instant a switch wants to change load and the instant that the switch feels the impact of the change. With a quick responding switch algorithm like ERICA, the feedback delay is less than a full round-trip delay. For an established flow it is close to the inter-RM cell time plus the round-trip delay between the bottleneck switch and the source. Further since ERICA (or other similar congestion avoiding switch algorithm) try to keep the switch queues small while keeping the utilizations high, we find that ATM layer reaches its steady state operating point much before TCP reaches its maximum window size. There are no queues in the network and the utilization is high.

During steady state, the TCP load is limited by the ACR granted by the switches and not so much by the window. Increasing window simply results in increasing queues at the source network interface card (NIC).

TCP PERFORMANCE WITH BACKGROUND VBR TRAFFIC:

The case when the network capacity for ABR varies continuously due to higher priority VBR sources is more interesting and realistic. In this case, the network may allow the sources to go at a higher rate but suddenly find its ABR capacity diminish due to VBR. Queues build up and some cells may be lost. This is the case that we study in detail and find the effect of various parameters.

We found that in the presence of VBR traffic, a lower TBE value performs better than higher TBE values. Disabling rule 6 is equivalent to setting TBE to infinity. This applies particularly to WANs. LAN cases do not indicate any significant impact of TBE since the round trip times and feedback delays are much smaller.

We also found that even a single packet drop results in the drain of the entire NIC queue. After the queue is empty, one round trip of time is lost until the ack for the retransmitted packet returns to the source. Actually some capacity is lost even before the source retransmits since the detection (timeout) takes several round-trip delay. Further, during retransmission the source sends all the packets again, possibly wasting considerable bandwidth (for large windows). However, the successive packet drops result in less damage since TCP is smart enough to take precautionary measures after each loss.

Since TCP cycles between exponential/linear increase phases and idle time (due to loss), the switch may allocate a high ACR during idle period and may find it flooded with ABR traffic (during exponential rise phase). If this happens to coincide with arrival of VBR traffic as well, the packet loss is inevitable. As discussed earlier, packet drop in TCP causes a significant reduction in link utilization due to long timeout intervals.

Rule 6 limits the size of the TCP burst following an idle period to TBE. This limits queues during the exponential rise phase. So, even though the lower TBE values may cause lower throughput initially (when the control loop is not yet set up), it can avoid packet drop in the network. In effect it moderates the exponential rise (increase by congestion window every RTT) by reducing ACR and hence increasing the RTT experienced by TCP. Hence it not only shields the network against bursts, but also the source against fluctuating network capacity and packet loss.

In summary, we find that lower TBE values result in better overall performance. Of course, TBE values have to be set in relation to the round-trip delays. Larger TBE values may be necessary for long-delay paths. The optimization of TBE is yet to be studied.

EFFECT OF TIMER GRANULARITY:

The damage caused by a packet loss depends upon the timeout interval, which in turn depends upon the round-trip delay. TCP implementations measure round-trip delays only in units of 100 ms or 500 ms. This parameter is called timer granularity.

Round-trip delays less than one unit of time are counted as one unit. For example, if the timer granularity is 100 ms, and the round trip delay is only 5 ms, TCP will base its timeouts on a round trip delay of 100 ms.

The timer granularity has a significant impact on the performance since it determines the damage caused by packet loss in most LAN and WAN situations. Larger granularity resulting in lower performance.

COMPARISON WITH OUR EARLIER WORK:

These results differ from our earlier analysis of infinite sources without transport layer congestion control [2]. In that analysis, we had not implemented higher layer protocols and had assumed that the packet loss does not result in any load reductions by higher layers. The packet loss was found to be as

high as 30%. That analysis applies to Non-TCP transport protocols (e.g., UDP) that do not have their own congestion control algorithms. (It must be pointed out that UDP is used by NFS and several other popular applications.) As a result of this analysis, we find that TCP gets much better performance (in terms of packet loss at least) than expected due to its congestion mechanism.

SIMULATION RESULTS:

At the forum presentation, we will present detailed simulation results justifying the conclusions mentioned here.

REFERENCES:

[1] R.Jain, S.Kalyanaraman, R. Goyal, "A Sample Switch Algorithm," ATM Forum/95-0178R1, February 1995.

[2] R.Jain, S.Fahmy, S.Kalyanaraman, R. Goyal, F.Lu, "More Strawvote Comments: TBE vs Queue sizes," AF-TM 95-1661, December 1995.

All our past ATM forum contributions and presentations can be obtained on-line:

http://www.cse.wustl.edu/~jain/