International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT

IJIEMR Transactions, online available on 19st July 2017. Link :
http://www.ijiemr.org/downloads.php?vol=Volume-6&issue=ISSUE-5

Title: Multimedia Data Transmission Through Tcp/Ip Using Transparent Error Correction.

Volume 06, Issue 05, Page No: 1925 – 1930.

Paper Authors

*** THIPPAGALLA PUSPA RANI, K.LJRAIL.**

* Dept of CSE, CVRT.

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

MULTIMEDIA DATA TRANSMISSION THROUGH TCP/IP USING TRANSPARENT ERROR CORRECTION

***THIPPAGALLA PUSPA RANI, **K.LJRAIL**
*PG Scholar, Dept of CSE, CVRT, AP, India
**Assistant Professor, Dept of CSE, CVRT, AP, India

ABSTRACT:
The global network of data centers is emerging as an important distributed systems paradigm — commodity clusters running high-performance applications, connected by high-speed 'lambda' networks across hundreds of milliseconds of network latency. Packet loss on long-haul networks can cripple the performance of applications and protocols — a loss rate as low as 0.1% is sufficient to reduce TCP/IP throughput by an order of magnitude on a 1 Gbps link with 50ms one-way latency. Maelstrom is an edge appliance that masks packet loss transparently and quickly from inter-cluster protocols, aggregating traffic for high-speed encoding and using a new Forward Error Correction scheme to handle bursty loss.

I. INTRODUCTION
THE emergence of commodity clusters and data centers has enabled a new class of globally distributed high performance applications that coordinate over vast geographical distances. For example, a financial firm's New York City data center may receive real-time updates from a stock exchange in Switzerland, conduct financial transactions with banks in Asia, cache data in London for locality and mirror it to Kansas for disaster-tolerance. To interconnect these bandwidth-hungry data centers across the globe, organizations are increasingly deploying private 'lambda' networks. Raw bandwidth is ubiquitous and cheaply available in the form of existing 'dark fiber'; however, running and maintaining high-quality loss-free networks over this fiber is difficult and expensive. Though high-capacity optical links are almost never congested, they drop packets for numerous reasons – dirty/degraded fiber [1], misconfigured/Malfunctioning hardware [2], [3] and switching contention [4], for example – and in different patterns, ranging from singleton drops to extended bursts [5], [6].

Non congestion loss has been observed on long-haul networks as well-maintained as Abilene/Internet2 and National Lambda Rail The inadequacy of commodity TCP/IP in high bandwidth delay product networks is extensively documented.TCP/IP has three major problems when used over such networks. First, TCP/IP suffers throughput collapse if the network is even slightly prone to packet loss. Conservative flow control mechanisms designed to deal with the systematic congestion of the commodity Internet react too sharply to

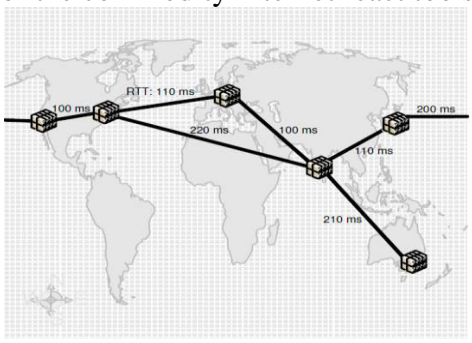

Fig. 1: Example Lambda Network

ephemeral loss on over-provisioned links — a single packet in ten thousand is enough to

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

reduce TCP/IP throughput to a third over a 50 ms gigabit link, and one in a thousand drops it by an order of magnitude. Second, real-time or interactive applications are impacted by the reliance of reliability mechanisms on acknowledgments and retransmissions, limiting the latency of packet recovery to at least the Round Trip Time (RTT) of the link. If delivery is sequenced, as in TCP/IP, each lost packet acts as a virtual 'road-block' in the FIFO channel until it is recovered. Third, TCP/IP requires massive buffers at the communicating endhosts to fully exploit the bandwidth of a long-distance high speed link, even in the absence of packet loss. Deploying new loss-resistant alternatives to TCP/IP is not feasible in corporate data centers, where standardization is the key to low and predictable maintenance costs; neither is eliminating loss events on a network that could span thousands of miles. Accordingly, there is a need to mask loss on the link from the commodity protocols running at end-hosts, and to do so rapidly and transparently. Rapidly, because recovery delays for lost packets translate into dramatic reductions in application-level throughput; and transparently, because applications and OS networking stacks in commodity data centers cannot be rewritten from scratch.

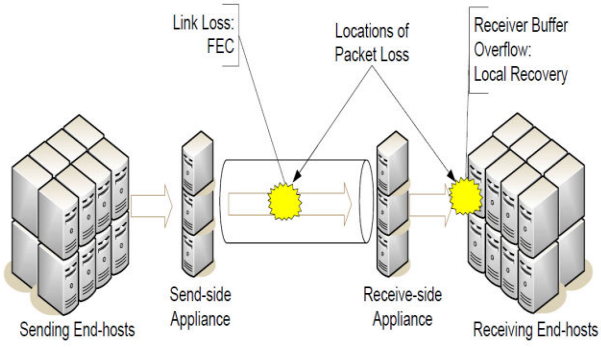

Fig. 2: Maelstrom Communication Path

Forward Error Correction (FEC) is a promising solution for reliability over long-haul links [11] — packet recovery latency is independent of the RTT of the link. While FEC codes have been used for decades within link-level hardware solutions, faster commodity processors have enabled packet-level FEC at end-hosts [12], [13]. End-to-end FEC is very attractive for communication between data centers: it's inexpensive, easy to deploy and customize, and does not require specialized equipment in the network linking the data centers. However, endhost FEC has two major issues — First, it's not transparent, requiring modification of the end-host application/OS. Second, it's not necessarily rapid; FEC works best over high, stable traffic rates and performs poorly if the data rate in the channel is low and sporadic [14], as in a single end-to-end channel. In this paper, we present the Maelstrom Error Correction appliance — a rack of proxies residing between a data center and its WAN link (see Figure 2). Maelstrom encodes FEC packets over traffic flowing through it and routes them to a corresponding appliance at the destination data center, which decodes them and recovers lost data. Maelstrom is completely transparent — it does not require modification of end-host software and is agnostic to the network connecting the data centers. Also, it eliminates the dependence of FEC recovery latency on the data rate in any single node-to-node channel by encoding over the aggregated traffic leaving the data center. Additionally, Maelstrom uses a new encoding scheme called layered interleaving, designed especially for time-sensitive packet recovery in the presence of bursty loss.

II. MODEL

Loss Model: Packet loss typically occurs at two points in an end-to-end communication path between two data centers, as shown in Figure 2 — in the wide-area network connecting them and at the receiving end-hosts. Loss in the lambda link can occur for many reasons, as stated previously: transient congestion, dirty or degraded fiber, malfunctioning or misconfigured equipment, low receiver power and burst switching contention are some reasons. Loss can also occur at receiving end-

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

hosts within the destination data center; these are usually cheap commodity machines prone to temporary overloads that cause packets to be dropped by the kernel in bursts [14] — this loss mode occurs with UDP-based traffic but not with TCP/IP, which advertises receiver windows to prevent end-host buffer ovWhat are typical loss rates on long-distance optical The answer to this question is surprisingly hard to determine, perhaps because such

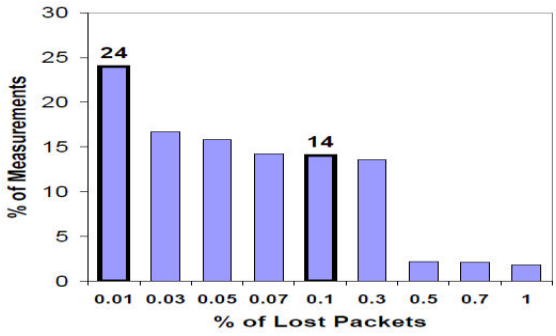

Fig. 3: Loss Rates on TeraGrid

links are a relatively recent addition to the networking landscape and their ownership is still mostly restricted to commercial organizations disinclined to reveal such information. One source of information is TeraGrid [17], an optical network interconnecting major supercomputing sites in the US. TeraGrid has a monitoring framework within which ten sites periodically send each other 1 Gbps streams of UDP packets and measure the resulting loss rate [18]. Each site measures the loss rate to every other site once an hour, resulting in a total of 90 loss rate measurements collected across the network every hour. Figure 3 shows that between Nov 1, 2007 and Jan 25, 2008, 24% of all such measurements were over 0.01% and a surprising 14% of them were over 0.1%. After eliminating a single site (Indiana University) that dropped incoming packets steadily at a rate of 0.44%, 14% of the remainder were over 0.01% and 3% were over 0.1%. These numbers may look small in absolute terms, but they are sufficient to bring TCP/IP throughput crashing down on high-speed long-distance links. Conventional wisdom states Nthat optical links do not drop packets; most carrier-grade optical equipment is configured to shut down beyond bit error rates of 10□12 — one out of a trillion bits. However, the reliability of the lambda network is clearly not equal to the sum of its optical parts; in fact, it's less reliable by orders of magnitude. As a result, applications and protocols – such as TCP/IP – which expect extreme reliability from the high-speed network are instead subjected to unexpectedly high loss rates. Of course, these numbers reflect the loss rate specifically experienced by UDP traffic on an end-to-end path and may not generalize to TCP packets. Also, we do not know if packets were dropped within the optical network or at intermediate devices within either data center, though it's unlikely that they were dropped at the end-hosts; many of the measurements lost just one or two packets whereas kernel/NIC losses are known to be bursty [14]. Further, loss occurred on paths whereLevels of optical link utilization (determined by 20-second moving averages) were consistently lower than 20%, ruling out congestion as a possible cause, a conclusion supported by dialogue with the network administrators

III. EXISTING RELIABILITY OPTIONS

TCP/IP is the default reliable communication option for contemporary networked applications, with deep, exclusive embeddings in commodity operating systems and networking APIs. Consequently, most applications requiring reliable communication over any form of network use TCP/IP. As noted earlier, TCP/IP has three major problems when used over high-speed long-distance networks:

1. Throughput Collapse in Lossy Networks: TCP/IP is unable to distinguish between ephemeral loss modes — due to transient congestion, switching errors, or bad fiber — and persistent congestion. The loss of one

packet out of ten thousand is sufficient to reduce TCP/IP throughput to a third of its lossless maximum; if one packet is lost out of a thousand, throughput collapses to a thirtieth of the maximum. The root cause of throughput collapse is TCP/IP's fundamental reliance on loss as a signal of congestion. While recent approaches have sought to replace loss with delay as a congestion signal [23], or to specifically identify loss caused by non-congestion causes [24], older variants — prominently Reno — remain ubiquitously deployed.

2. Recovery Delays for Real-Time Applications: Conventional TCP/IP uses positive acknowledgments and retransmissions to ensure reliability — the sender buffers packets until their receipt is acknowledged by the receiver, and resends if an acknowledgment is not received within some time period. Hence, a lost packet is received in the form of a retransmission that arrives no earlier than 1.5 RTTs after the original send event. The sender has to buffer each packet until it's acknowledged, which takes 1 RTT in lossless operation, and it has to perform additional work to retransmit the packet if it does not receive the acknowledgment. Also, any packets that arrive with higher sequence numbers than that of a lost packet must be queued while the receiver waits for the lost packet to arrive. Consider a high-throughput financial banking application running in a data center in New York City, sending updates to a sister site in Switzerland. The RTT value between these two centers is typically 100 milliseconds; i.e., in the case of a lost packet, all packets received within the 150 milliseconds or more between the original packet send and the receipt of its retransmission have to be buffered at the receiver. As a result, the loss of a single packet stops all traffic in the channel to the application for a seventh of a second; a sequence of such blocks can have devastating effect on a high-throughput system where every spare cycle counts. Further, in applications with many fine-grained components, a lost packet can potentially trigger a butterfly effect of missed deadlines along a distributed workflow. During high-activity periods, overloaded networks and end-hosts can exhibit continuous packet loss, with each lost packet driving the system further and further out of sync with respect to its real-world deadlines.

3. Massive Buffering Needs for High Throughput Applications: TCP/IP uses fixed size buffers at receivers to prevent overflows; the sender never pushes more unacknowledged data into the network than the receiver is capable of holding. In other words, the size of the fluctuating window at the sender is bounded by the size of the buffer at the receiver. In high-speed long-distance networks, the quantity of inflight unacknowledged data has to be extremely high for the flow to saturate the network. Since the size of the receiver window limits the sending envelope, it plays a major role in determining TCP/IP's throughput. The default receiver buffer sizes in many standard TCP/IP implementations are in the range of tens of kilobytes, and consequently inadequate receiver buffering is the first hurdle faced by most practical deployments. A natural solution is to increase the size of the receiver buffers; however, in many cases the receiving end-host may not have the spare memory capacity to buffer the entire bandwidth-delay product of the long-distance network. The need for larger buffers is orthogonal to the flow control mechanisms used within TCP/IP and impacts all variants equally.

IV. RELATED WORK

Maelstrom lies in the intersection of two research areas that have seen major innovations in the last decade — high-speed long-haul communication and forward error correction. TCP/IP variants such as Compound TCP [37] and CUBIC [38] use transmission delay to detect backed up routers, replacing or

supplementing packet loss as a signal of congestion. While such protocols solve the congestion collapse experienced by conventional TCP/IP on high-speed long-haul networks, they cannot mitigate the longer packet delivery latencies caused by packet loss, and they do not eliminate the need for larger buffers at end-hosts. FEC has seen major innovations in the last fifteen years. Packet-level FEC was first described for high-speed WAN networks as early as 1990 [39]. Subsequently, it was applied by researchers in the context of ATM networks [40]. Interest in packet-level FEC for IP networks was revived in 1996 [13] in the context of both reliable multicast and long-distance communication. Rizzo subsequently provided a working implementation of a software packet-level FEC engine [11]. As a packet-level FEC proxy, Maelstrom represents a natural evolution of these ideas. The emphasis on applying error correcting codes at higher levels of the software stack has been accompanied by advances in the codes themselves. Prior to the mid-90s, the standard encoding used was Reed-Solomon, an erasure code that performs excellently at small scale but does not scale to large sets of data and error correcting symbols. This scalability barrier resulted in the development of new variants of Low Density Parity Check (LPDC) codes [41] such as Tornado [42], LT [43] and Raptor [44] codes, which are orders of magnitude faster than Reed-Solomon and much more scalable in input size, but require slightly more data to be received at the decoder. While the layered interleaving code used by Maelstrom is similar to the Tornado, LT and Raptor codes in its use of simple XOR operations, it differs from them in one very important aspect — it seeks to minimize the latency between the arrival of a packet at the send-side proxy and its successful reception at the receive-side proxy. In contrast, codes such as Tornado encode over a fixed set of input symbols, without treating symbols differently based on their sequence in the data stream. In addition, as mentioned in Section IV-C, layered interleaving is unique in allowing the recovery latency of lost packets to depend on the actual burst size experienced, as opposed to the maximum tolerable burst size as with other encoding schemes.

V. CONCLUSION

Modern distributed systems are compelled by real-world imperatives to coordinate across data centers separated by thousands of miles. Packet loss cripples the performance of such systems, and reliability and flow-control protocols designed for LANs and/or the commodity Internet fail to achieveoptimal performance on the high-speed long-haul 'lambda' networks linking data centers. Deploying new protocols is not an option for commodity clusters where standardization is critical for cost mitigation. Maelstrom is an edge appliance that uses Forward Error Correction to mask packet loss from endto-end protocols, improving TCP/IP throughput and latency by orders of magnitude when loss occurs. Maelstrom is easy to install and deploy, and is completely transparent to Mapplications and protocols — literally providing reliability in an inexpensive box.

REFERENCES

[1] R. Habel, K. Roberts, A. Solheim, and J. Harley, "Optical Domain Performance Monitoring," in OFC 2000: The Optical Fiber Communication Conference, Baltimore, MD, 2000.

[2] Internet2, "End-to-end performance initiative: When 99% isn't quite enough - educause bad connection," http://e2epi.internet2.edu/ case studies/EDUCAUSE/index.html.

[3] ——, "End-to-end performance initiative: Hey! where did my performance go? - rate limiting rears its ugly head," http://e2epi.internet2.edu/ case-studies/UMich/index.html.

[4] A. Kimsas, H. Øverby, S. Bjornstad, and V. L. Tuft, "A Cross Layer Study of Packet Loss in All-Optical Networks," in AICT/ICIW, Guadelope, French Caribbean, 2006.

[5] D. C. Kilper, R. Bach, D. J. Blumenthal, D. Einstein, T. Landolsi, L. Ostar, M. Preiss, and A. E. Willner, "Optical Performance Monitoring," Journal of Lightwave Technology, vol. 22, no. 1, pp. 294–304, 2004.

[6] T. J. Hacker, B. D. Noble, and B. D. Athey, "The Effects of Systemic Packet Loss on Aggregate TCP Flows," in Supercomputing '02: ACM/IEEE Conference on Supercomputing, Baltimore, MD, 2002.

[7] T. J. Hacker, B. D. Athey, and B. D. Noble, "The End-to-End Performance Effects of Parallel TCP Sockets on a Lossy Wide-Area Network," in IPDPS 2002: International Parallel and Distributed Processing Symposium, Fort Lauderdale, FL, 2002.

[8] T. Lakshman and U. Madhow, "The Performance of TCP/IP for Networks with High Bandwidth-Delay Products and Random Loss," IEEE/ACM Transactions on Networking (TON), vol. 5, no. 3, pp. 336– 350, 1997.

[9] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCPThroughput: A Simple Model and its Empirical Validation," ACM SIGCOMM Computer Communication Review, vol. 28, no. 4, pp. 303–314, 1998.

[10] D. Katabi, M. Handley, and C. Rohrs, "Congestion Control for High Bandwidth-Delay Product Networks," in ACM SIGCOMM, Pittsburgh, PA, 2002.