silver peak™

White Paper

Taking Forward Error Correction (FEC) to the Next Level

Using Packet-Level FEC to Improve Network Integrity in WAN Environments

Taking Forward Error Correction (FEC) to the Next Level

Using Packet-Level FEC to Improve Network Integrity in WAN Environments

Introduction

Enterprises rely on a variety of different technologies to build Wide Area Networks (WANs), such as Time Division Multiplexing, Frame Relay, and Virtual Private Networks. These technologies use well established, link-layer techniques to deliver error-free, well-formed Protocol Data Units (PDUs) to upper layers. However, different network-layer characteristics result in unique packet delivery behavior across these technologies.

Even when the physical layer of a WAN is error-free, some technologies and provisioning practices still lead to packet loss at the network layer. In fact, it is not unusual to see network packet loss rates as high as 8% in some networks. When this type of loss is coupled with high latency and the retransmission and congestion avoidance behavior inherent to the Transmission Control Protocol (TCP), it is not surprising that application performance suffers across a WAN.

Forward Error Correction (FEC) is a technology that is well known for its ability to correct bit errors at the physical-layer. However, this technology can also be adapted to operate on packets at the network layer to improve application performance across WANs that have high-loss characteristics. With packet-level FEC, network equipment can reconstitute lost packets at the far end of a WAN link, avoiding delays that come with multiple round-trip retransmissions. This enables WANs to easily recover from packet loss due to a variety of network layer conditions, such as queue overflows and constrained bandwidth links. With packet level FEC, enterprises commonly see significant improvements in application performance — up to a 10-fold performance increase in some WAN environments.

Network-layer loss in enterprise wans

Enterprise WANs generally employ one or more of the following technologies:

1. Time Division Multiplexing (TDM) Private Line
2. Frame Relay or Asynchronous Transfer Mode (ATM)
3. Internet Protocol (IP) Virtual Private Network (VPN)

TDM private lines are circuits provisioned in a point-to-point fashion between two enterprise locations. A dedicated physical circuit is established and maintained through a service provider's network to provide this type of service. TDM is used to provision and guarantee bandwidth within these circuits. Because there is no statistical multiplexing at the packet level, there is no packet loss with this technology.

Frame Relay and Asynchronous Transfer Mode (ATM) are packet switching technologies employed by service providers to deliver WAN services. Service providers employ these technologies because they allow the physical links within the service provider's network to be shared among the provider's many customers. This is achieved with statistical multiplexing.

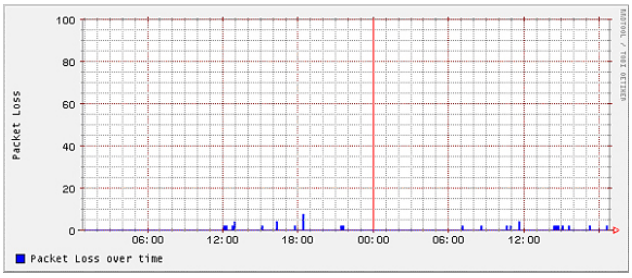

Figure 1: Packet-loss at ISP "A"

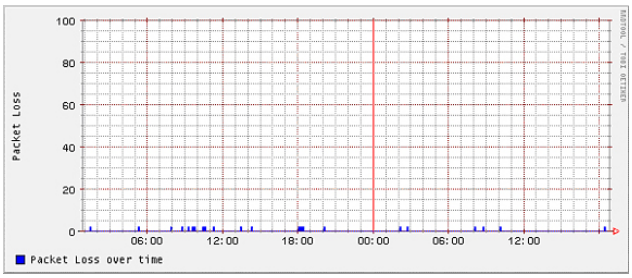

Figure 2: Packet-loss at ISP "B"

Statistical multiplexing relies on the fact that not all customers will use all of the network's available bandwidth between any two points at any given time. The link's bandwidth is divided among the customers using packet switching. Contention for link resources is mitigated by buffering packets in queues in the Frame Relay or ATM switches. Queues, however, have finite capacity and will overflow during periods of high traffic and when there is competition for the same link. Service providers attempt to mitigate this by engineering their networks for peak loads under normal traffic patterns. Traffic patterns, however, are unpredictable and therefore it is not possible to guarantee that packets will never get dropped due to congestion. As a result, it is common to see 1% or more packet loss on a Frame Relay or ATM WAN link. Some service providers can honor packet-level QoS markings to decide how to prioritize traffic when links become congested. In most networks, however, packets are discarded indiscriminately under heavy load conditions.

IP VPNs rely on the Internet to connect remote enterprise locations. VPN technology is used to guarantee privacy of enterprise traffic as it traverses the Internet. As the cost of Internet access continues to fall and as enterprises grow more comfortable with maturing VPN technology, this option is becoming more and more commonplace in enterprise environments. The problem, however, is that packet loss on the Internet can be quite high — usually much worse than a typical service provider's Frame Relay or ATM network. In addition, there is a wider range of loss levels in these types of environments.

The website, http://weather.nacs.uci.edu/ hosted by the University of California at Irvine, collects and maintains statistics on many Internet Service Providers (ISPs). Figure 1 and Figure 2 show packet loss statistics from two representative ISPs collected from this site on a typical afternoon. In Figure 1, the current packet loss rate is 1.87%; the maximum loss rate over the previous 24-hour period is 8%. In Figure 2, the current packet loss rate is 2%; the maximum loss rate over the previous 24-hour period is also 2%. It is important to note that not only are these packet loss rates significant, but also that they vary greatly over the 24-hour period observed here.

The Effects Of Loss On Tcp Transfers

It may seem that a packet loss rate of 8% should only affect application performance by a similar amount. The reality, though, is quite different when packet loss is coupled with latency and TCP's retransmission and congestion avoidance behavior.

TCP employs an end-to-end congestion avoidance mechanism implemented in the client and server hosts that are communicating on behalf of applications within an enterprise network. When TCP senses packet loss due to a transmission time-out, not only does it retransmit the lost packets, but it interprets the loss as a sign that there is congestion within the network, which triggers its congestion avoidance mechanisms. These mechanisms cause end stations to exponentially back off the rate of their transmissions, and then slowly increase their rate of transmissions over time. This means that packet loss results in a drastic decrease in throughput across poorly utilized networks.

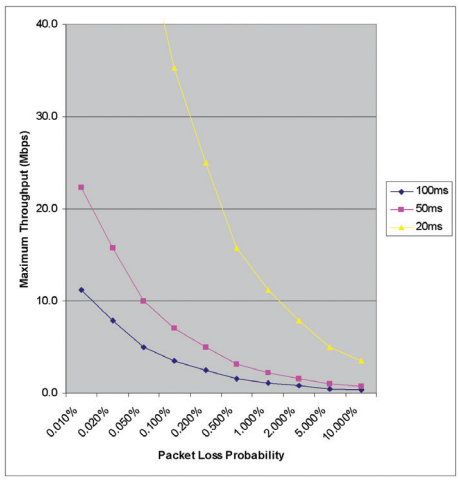

Figure 3: Maximum throughput for different packet loss ratios and round-trip times

Figure 3 shows how TCP's maximum theoretical throughput decreases exponentially as packet loss increases. The effect gets worse as latency increases. The yellow curve assumes a Round Trip Time (RTT) of 20ms (that is, an end-to-end delay or latency of 10ms) and shows how increasing packet loss exponentially decreases TCP's theoretical maximum throughput. The magenta and blue curves show the impact of loss on throughput when the RTT is 50ms and 20ms respectively.

The situation is exacerbated by the fact that TCP has a built-in (un)fairness assumption. TCP's throughput is inversely proportional to Round Trip Time (RTT), ensuring that "shorter" (lower latency) connections are guaranteed more bandwidth than "longer" (higher latency) connections. For example, a connection that has half the latency of another connection will be allocated twice the bandwidth. This effect can clearly be seen in Figure 3, where lower latency connections have significantly higher bandwidth than higher latency connections (that is, the yellow curve is much higher on the chart than the magenta and blue curves).

TCP was designed for a network of cooperators. In the presence of packet loss, it avoids total congestion collapse but it does not deliver optimal throughput for all users. TCP was not optimized for an environment when enterprises pay service-providers for guaranteed levels of bandwidth. As each enterprise is sharing the service provider's network with other customers, congestion and packet loss are unavoidable. TCP back-off, however is not the best reaction to packet loss because one customer should not be restricted from using the bandwidth they paid for just because another customer is competing for the same bandwidth. Furthermore, service providers cannot penalize customers with higher latencies by delivering them lower throughput.

A better solution is required, which allows an enterprise to negate the effects of packet loss while ensuring that the enterprise has full access to all the bandwidth it has paid for. Packet-level Forward Error Correction (FEC) is the answer.

Using Forward Error Correction (Fec) To Mitigate Packet Loss

While FEC is traditionally used at the physical link level to correct bit errors, it has since been adapted to recover from packet loss at the network level. Packet-level FEC (see Figure 4) works by adding an additional loss recovery packet for every "N" packets that are sent. This additional loss recovery packet enables network-layer equipment to reconstitute lost packets at the far end of a WAN link, before the packets are delivered to TCP or other transport layers. This avoids transport-layer retransmissions and, in the case of TCP, prevents TCP's congestion avoidance mechanism from stepping in and lowering the throughput available to the application. For the modest overhead of an additional loss recovery packet, FEC reduces packet losses dramatically, enabling applications to benefit from the maximum throughput that the WAN link can support.

To fully understand the impact that packet-level FEC can have on packet loss rates within a network, it is worth considering some specific cases. For example, introducing one loss recovery packet for every 10 "regular" packets (1:10 FEC) can reduce a 1% packet loss to less than 0.09%. Introducing one loss recovery packet for every five packets (1:5 FEC) can reduce that same 1% packet loss to less than 0.04%. In the extreme, 1:5 FEC can even reduce a 5% loss to less than 1%. In the context of Figure 3, this has the effect of moving the operating point "up and to the left" — clearly a dramatic increase in throughput.

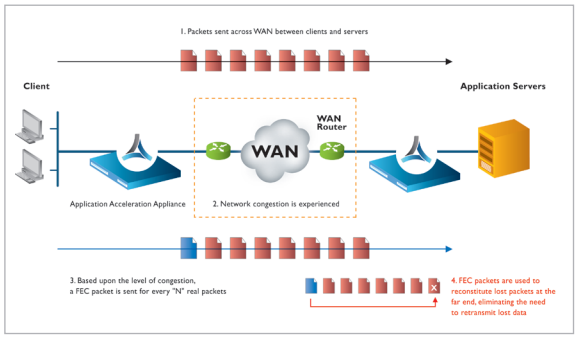

Figure 4: Packet-level FEC

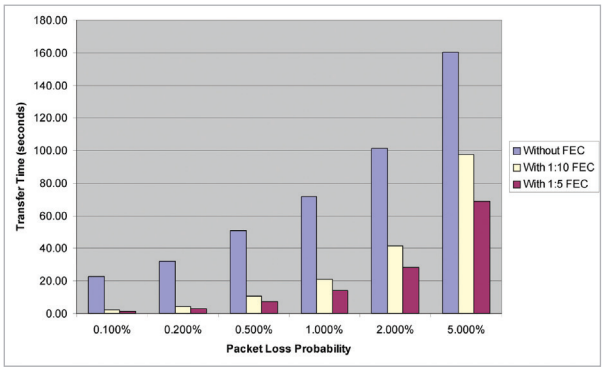

Figure 5: File transfer times for different packet loss ratios — without FEC and with 1:5 FEC

To work in real-world environments, FEC needs to be adaptive. This is because packet loss is random in nature and occurs due to traffic bursts in periods of heavy network usage. As a result, packet loss varies greatly over time, even over periods as short as 24 hours.

An optimal implementation will vary the amount of FEC packets to accommodate different loss conditions. In other words, more loss recovery packets will be used during period of high network packet loss; fewer FEC packets will be used during periods of small packet loss. This helps to balance the performance benefits of FEC with the overhead that the solution inherently introduces.

Silver Peak offers an adaptive FEC solution. In other words, the FEC ratio (of loss-recovery packets to data packets) is tied to dynamic network measurements. When a link is experiencing no loss, FEC is disabled and no overhead is incurred. When loss is detected (due to a network event or during periods of congestion), FEC automatically steps in, reducing loss by an order of magnitude or more. The FEC ratio is adjusted dynamically to ensure that application performance is maximized while overhead is kept at a minimum.

FEC in Action

The chart in Figure 5 shows FEC in action, mitigating the effects of loss on a file transfer across a typical WAN link with 100ms of delay and varying rates of packet loss.

When there is 0.1% packet loss on the link, the file transfer takes 23 seconds without FEC. Introducing 1:5 FEC reduces this time to 1.5 seconds, a fifteen-fold improvement.

At high loss rates, FEC still adds significant value. For example, with 2% loss, the file transfer takes 101 seconds without FEC and 28 seconds with FEC, almost a four-fold improvement.

At 5% loss, the file transfer takes 69 seconds with FEC enabled. This is less than the 72 seconds it takes to transfer the same file with a 1% packet-loss rate and no FEC.

CONCLUSION

Although WAN links may have no bit errors at the link layer, packets may still be lost, dropped, or delayed at the network layer at various congestion points. This is true for any packet switching–based WAN technology, such as Frame Relay, ATM, or MPLS, which relies on statistical multiplexing to deliver service. It is especially true for IP VPNs, where it is common to see packet loss rates as high as 8%. When these loss rates are combined with high latencies inherent to WANs and the unique effects associated with TCP's retransmission and congestion avoidance mechanisms, the result is poor application performance and dissatisfied end users.

By applying FEC at the packet level, enterprises can easily recover from instances of packet loss in the service provider's network. This ensures that enterprise WANs deliver maximum throughput to applications for optimal performance. For example, a large file transfer on a typical WAN link with 0.1% packet loss can be improved 15-fold by introducing one FEC loss recovery packet for every five data packets.

FEC works best on a high-rate aggregate flow, rather than on individual flows. As a result, it is best implemented in environments that leverage tunnels or aggregated flows when transferring traffic across a WAN, as is the case with the Silver Peak solution. An ideal FEC implementation, such as Silver Peak, will also adapt the amount of overhead to accommodate changing WAN conditions.

When implemented properly, FEC provides a significant increase in application performance under a wide range of network conditions, making it indispensable to enterprise application delivery across a WAN.