HyperIP: FTP Application Note

Introduction

HyperIP is a Linux software application that runs on an appliance that quantifiably and measurably enhances large data movement over big bandwidth and long-haul IP networks. HyperIP is used to mitigate TCP performance problems over long-distance wide-area networks. HyperIP is the most recent iteration of production-hardened transport technologies from NetEx that eliminates the negative impacts of packet loss, congestion, out of order sequence issues and network 'WAN' latency on TCP/IP data transfers.

FTP (File Transfer Protocol) is a standard TCP file transfer application. It allows users to copy files between their local system and any system they can reach on the TCP/IP network.

FTP over the WAN can be a frustrating experience when packet loss, congestion, or jitter is high. Those conditions cause significant amounts of data retransmission. Standard FTP does not handle retransmissions well because it must retransmit the entire file.

HyperIP eliminates that frustration by overcoming the limitations of TCP/IP.

TCP high bandwidth long haul issues and limitations

Several characteristics of TCP/IP cause FTP to perform poorly over high bandwidth and long distances:

- ***Window Size***

Window size is the amount of data allowed to be outstanding (in-the-air) at any given point-in-time. The available window size on a given bandwidth pipe is the rate of the bandwidth times the round-trip delay or latency. Using a cross-country OC-3 link (approximately 60 ms based on a total 6000-mile roundtrip) creates an available data window of 155Mbps x 60ms = 1,163Kbytes. A DS3 satellite connection (540 ms roundtrip) creates an available data window of 45Mbps X 540ms = 3,038Kbytes.

When this is contrasted with standard and even enhanced versions of TCP, there is a very large gap between the available window and the window utilized. Most standard TCP implementations are limited to 65Kbytes windows. There are a few enhanced TCP versions capable of using up to 512Kbytes or larger windows. Either case means an incredibly large amount of "dead air" and very inefficient bandwidth utilization.

- ***Acknowledgement Scheme***

TCP causes the entire stream from any lost portion to be retransmitted in its entirety. In high packet loss scenarios this will cause large amounts of bandwidth to be wasted in resending data that has already been successfully received, all with the long latency time of the path. Each retransmission is additionally subjected to the performance penalty issues of "Slow Start".

- ***Slow start***

TCP data transfers start slowly to avoid congestion due to large numbers of sessions that may be competing for the bandwidth, and ramp-up to their maximum transfer rate, resulting in poor performance for short sessions.

- ***Session free-for-all***

Each TCP session is throttled and contends for network resources independently, which can cause over-subscription of resources relative to each individual session. The net result of these issues is very poor bandwidth utilization. The typical bandwidth utilization for large data transfers over long-haul networks is usually less than 30% and more often less than 10%.

As fast as bandwidth costs are dropping, they are still not free.

The cost effective solution: NetEx HyperIP

HyperIP was designed specifically for moving large amounts of data over big bandwidth and distance, and to be highly efficient regardless of the packet loss, congestion, or jitter. HyperIP is a standard TCP/IP network appliance requiring no modifications to LAN/WAN infrastructures and uses no proprietary hardware. It provides transparent "acceleration" across long-haul high bandwidth WANs.

HyperIP provides the following benefits:

- ***Window size***

The HyperIP transport protocol keeps the available network bandwidth pipe full. The results are 90 + percent efficient link utilization. It eliminates the discrepancy between maximum available bandwidth and the results provided by native TCP/IP.

- ***Acknowledgement scheme***

HyperIP transport protocol retransmits only the NAK'd segments and not all the data that has already been successfully sent.

- ***Fast start***

Configuration parameters allow HyperIP to start transmissions at a close approximation of the available session bandwidth.

- ***Dynamic adjustments***

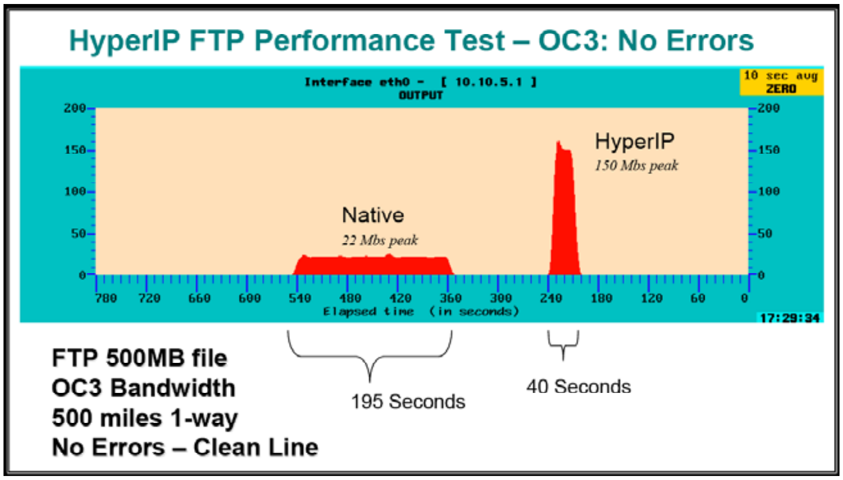
When feedback from the receiver in the acknowledgement protocol is received, HyperIP quickly "zeroes-in" on the appropriate send rate for current conditions.

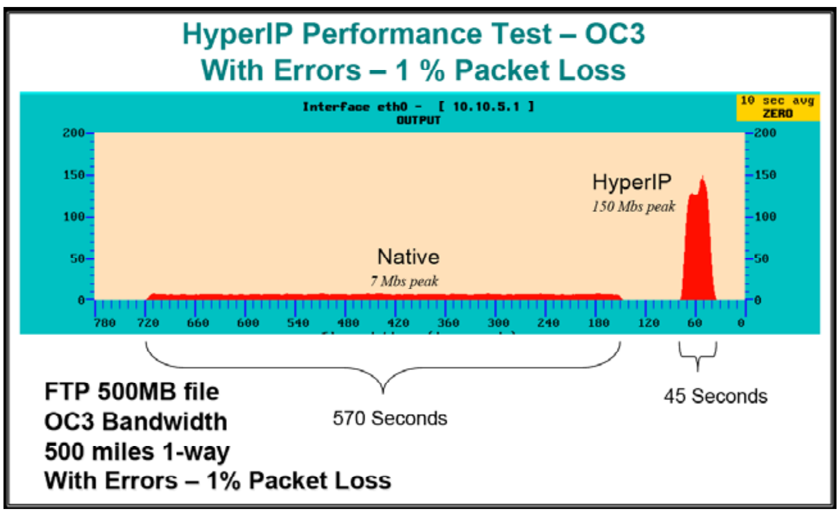
- ***Session pipeline***

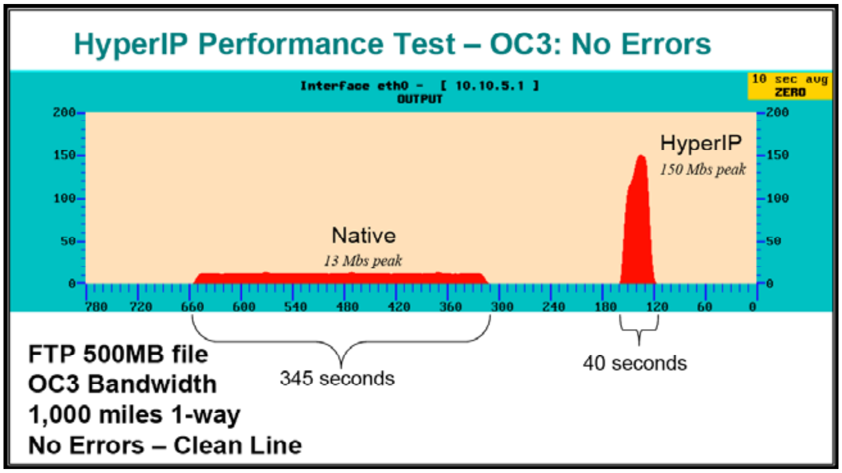
HyperIP's design allows traffic from multiple TCP sessions to be aggregated over a smaller set of connections between the HyperIP appliances, enabling a more efficient use of the bandwidth and less protocol overhead.

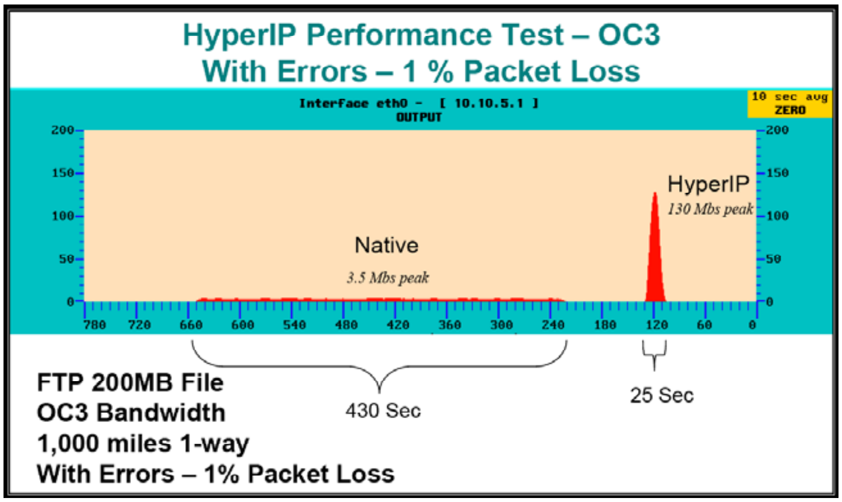
HyperIP's incredible FTP test results over long-haul IP WANs

Testing results with FTP have been outstanding. Bandwidth utilization approached 100% both without and with packet loss on an OC3 (155Mb/s) circuit.









While none of the above performance charts include compression the bandwidth utilization can be further reduced depending upon the compressibility of the data. Typical compression ratios range from 2:1 to 15:1 depending upon the data.

- ***What this means to end-users***

FTP on HyperIP provides the highest possible throughput. HyperIP saves bandwidth, time, and money.

HyperIP software components

- ***IP-Packet Edge Intercept***

This component intercepts IP packets, optimizes for performance, and reroutes over the HyperIP protocol on the network.

When a message is intercepted and rerouted, the original IP addressing information is retained and sent as additional protocol information. This allows each message to be reconstructed with the original addressing information at the destination side. A pre-built configuration file describing the HyperIP configuration is processed at initialization.

- ***IP Accelerator***

This component establishes and maintains connections with other HyperIP nodes on the IP network. This IP Accelerator receives intercepted packets from each of the Edge processes. It aggregates these packets into more efficient buffers, and then passes these buffers to the HyperIP Transport component, which sends them to the HyperIP node on the other side of the network.

The remote HyperIP receives these aggregated buffers on the network and passes them on to the IP Accelerator, which sends the packets from the buffer on to the appropriate Edge Interface process.

- ***HyperIP Transport***

The transport component provides the transport delivery mechanism between HyperIP nodes. It receives the optimized buffers from the IP Accelerator and delivers them to the destination HyperIP node for subsequent delivery to the end destination. It is responsible for maintaining acknowledgements of data buffers and resending buffers when required. It maintains a flow control mechanism on each connection, and optimizes the performance of each connection to match the available bandwidth and network capacity.

Since HyperIP provides a complete transport mechanism for managing data delivery, it uses UDP socket calls as an efficient, low overhead, data streaming protocol to read and write from the network.

- ***Compression Engine***

The HyperIP LZO-based software compression engine compresses the aggregated packets that are in the highly efficient IP Accelerator buffers. This provides an even greater level of compression efficiency since a large block of data is compressed at once rather than multiple small packets being compressed individually.

Summary and Conclusion

HyperIP provides the highest possible throughput for FTP over TCP/IP WANs. It does this with a “Production Hardened Transport” that provides:

- Bandwidth utilization that consistently exceeds 90 + percent regardless of packet loss, congestion or jitter;
- 2:1 to 15:1 compression;
- Elimination of TCP/IP network latency.

FTP over TCP/IP with HyperIP is the solution of choice.