WHITE PAPER

TCP Testing

RFC 6349 based TPC throughput tests and stress load tests give network operators valuable information on network TCP performance

OVERVIEW

Enterprises will typically sign a Service Level Agreement (SLA) with a service provider for their communication path through a network. The SLA contains worst case values for parameters like bandwidth, latency, packet jitter, frame loss ratio and availability. The service provider can conduct RFC 2544 and Y.1564 tests to verify that the requirements in the SLA are met. However, even if the tests show that all criteria are fulfilled, the enterprises may complain that they get less bandwidth than expected or that they experience long response times from the applications they use.

The reason for the complaints will in many cases be non-optimal configuration of the Transmission Control Protocol (TCP). TCP improves the "best effort" nature of IP networks by adding mechanisms guaranteeing that data sent to a recipient will actually be delivered and in the right order. To provide this functionality TCP buffer the data at both sending and receiving end of a connection. If the buffers are not dimensioned correctly, customers may experience performance degradation.

In RFC 6349 the Internet Engineering Task Force (IETF) has defined a "Framework for TCP Throughput Testing", providing a methodology for testing sustained TCP Layer performance. In addition to finding the TCP throughput at the optimal buffer size, RFC 6349 presents metrics that can be used to better understand the results. The Xena layer 4-7 test solutions XenaScale and XenaAppliance support powerful TCP testing based on RCF 6349. In addition XenaScale and XenaAppliance support extreme RFC 6349 testing with millions of concurrent TCP connections, giving service providers valuable information on the number of users and connections the network can handle, highlighting capacity bottlenecks.

"TCP turns the 'best effort' nature of IP networks into reliable communication services. Tests are however needed to ensure optimal performance."

RFC 6349 based TCP throughput testing

Contents

WHITE PAPER

INTRODUCTION

Communication is essential for the modern society and its enterprises. Many enterprises cover a large geographical area with a number of remote branch offices that also need to be included in the enterprise communication network to get access to applications and services needed for their business.

Figure 1: Enterprise communication

To ensure they get the performance and quality they need, enterprises typically sign a Service Level Agreement (SLA) with a service provider for their communication through the service provider's network. The SLA will be based on layer 2-3 parameters and contains worst case values for parameters like bandwidth, latency, packet jitter, frame loss ratio and availability.

The service provider can verify that the SLA requirements are met using RFC 2544 and Y.1564 test methodologies. However even if the tests show that all criteria are fulfilled, the enterprises may complain that they get less bandwidth than expected or that they experience long response times from the applications they use.

The reason for the complaints will in many cases be non-optimal configuration of the layer 4 Transmission Control Protocol (TCP). TCP is the de-facto standard way of interconnecting hosts over the Internet. TCP turns the "best effort" nature of IP networks into reliable communication services by adding mechanisms, which guarantee that data sent over a network will actually be delivered to the recipient in the right order. To achieve this TCP needs to buffer the data at both sending and receiving end of a connection. If these buffers are dimensioned incorrectly, customers may experience bandwidth or response time issues.

Even though end-to-end TCP connections typically is handled by equipment managed by enterprise IT departments, the service provider will often get complains about degraded network performance in case of TCP layer problems. Therefore the service providers will benefit from having tools to test and verify the network's TCP performance and discuss the TCP issues with the enterprises.

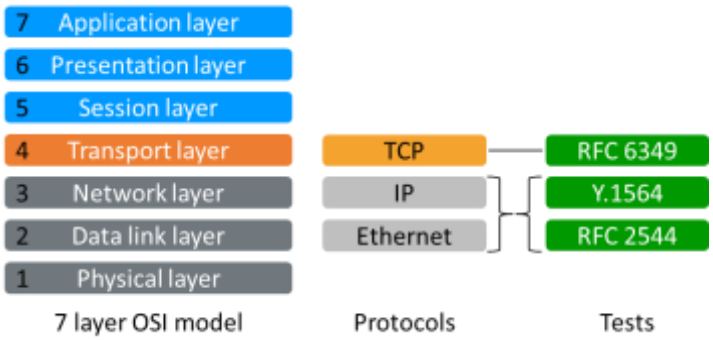

Figure 2: The 7 layer OSI model, protocols and related test methodologies

Figure 2 shows the 7 layer OSI model, some of the protocols related to layer 2 to 4 and corresponding test methodologies. Normally layers 1-3 are the responsibility of the service providers/network operators, while layers 5-7 are the responsibility of enterprise IT departments. Layer 4 is in-between and can be an area for discussion.

In RFC 6349 the Internet Engineering Task Force (IETF) has defined a "Framework for TCP Throughput Testing", providing a methodology for testing sustained TCP Layer performance. In addition to finding the TCP throughput at the optimal buffer size, RFC 6349 presents metrics that can be used to better understand the results. With RFC 6349 service providers can document their network's TCP performance to their customers.

TCP

The Transmission Control Protocol (TCP) is a transport protocol carried over the Internet Protocol (IP) – together the two protocols are often called the TCP/IP protocol stack. The TCP is a host-to-host protocol; its scope is to provide a reliable process-to-process communication in a multi network environment. Where IP provides no guarantee that sent packets are actually delivered to the intended recipient in the right order, TCP detects these problems, re-transmits lost data, removes duplicated data and rearranges out-of-order data. Hereby TCP provides reliable, ordered and error-checked data delivery between applications communicating through an IP network. Many applications like World Wide Web, email and file transfer use TCP and with many users on a network accessing these applications, the number of concurrent TCP connections in the network can be very high. Applications that do not require reliable data delivery can use the connectionless User Datagram Protocol (UDP), which provides reduced latency – and reliability.

TCP is a connection-oriented protocol and includes connection establishment and closing phases. Data received from higher protocol layers are divided into segments with sequence numbers and sent through the network. The receiving end must acknowledge the segments; if the sending end does not receive the acknowledgement (ACK) within a certain time frame, the segment will be retransmitted. TCP sequence numbers are 32 bits long; a sequence number is assigned to each byte sent. When a packet containing a TCP segment is sent, the TCP header will include the sequence number of first byte in the segment. ACK packets include the next sequence number expected by the receiver.

Buffers – or windows – are used at sending and receiving ends to avoid throughput reduction. For each connection, TCP maintains a Congestion Window (CWND), limiting the total number of unacknowledged segments that may be in transit (or be "in-flight") end-to-end. If the CWND is too large the network may be congested i.e. receive more data than it can handle and it will drop some of the data. This will lead to retransmissions, which will reduce the effective TCP throughput. Therefore algorithms have been defined to avoid undue congestion by adjusting the size of the CWND. The algorithms will increase the CWND size until packets are lost and then find a lower CWND size for the connection. The original TCP congestion avoidance algorithm was known as "TCP Tahoe", but later many other algorithms have been defined (e.g. TCP Reno, TCP New Reno, TCP Vegas, FAST TCP and TCP Hybla).

The sending part can send all the contents of the CWND without receiving any ACKs. When ACKs are received the related part of the CWND is released and can be used for new segments.

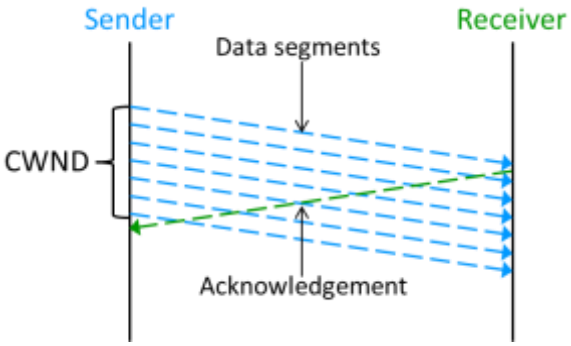

Figure 3: Acknowledgement of sent segments

Figure 3 illustrates how data segments are sent and acknowledged. If an ACK is not received before all segments in the CWND are sent, the transmission is stopped until the receipt of the ACK. The Round Trip Time (RTT) (the time it takes from a segment is sent until the related ACK is received) is important as it is the time the CWND will have to wait for the ACK. Another important parameter is the maximum bandwidth (or the bottleneck bandwidth (BB)) available on the paths going from the sender to the receiver and back. Multiplying RTT and BB gives the Bandwidth-Delay Product (BDP):

$$BDP = BB \times RTT$$

BDP is the number of bits that can be sent before the sender gets an ACK from the receiver and the BDP is the minimum size of the CWND if you want to avoid that the transmission stops waiting for the next ACK. Therefore if the congestion avoidance algorithm used for a connection for some reason (e.g. frame loss not caused by congestion avoidance algorithm itself) adjusts the CWDN to a value lower than the BDP, optimal throughput will not be obtained as the sending part will be inactive until the next ACK arrives.

The receiving end will send an ACK to acknowledge the received segments to the sending end. The ACK will include information on the buffer capacity still available at the receiving end in the

Window Size parameter in the ACK header. When all buffer capacity is used, the Window Size is set to 0. A message with Window Size > 0 is sent when buffer capacity is available again.

For the receiving end the TCP Receive Window (RWND) buffers received segments until they can be delivered to the higher layer applications. If segments are received in wrong order they will be re-ordered correctly before passing them on to the higher layer applications.

The RWND operates as a sliding window as illustrated in figure 4. With every ACK the receiving end will indicate a range of acceptable sequence numbers beyond the last segment successfully received. This indicates the amount of data the sending part may transmit before receiving further permission. The RWND size should equal the BDP to ensure sufficient buffer capacity.

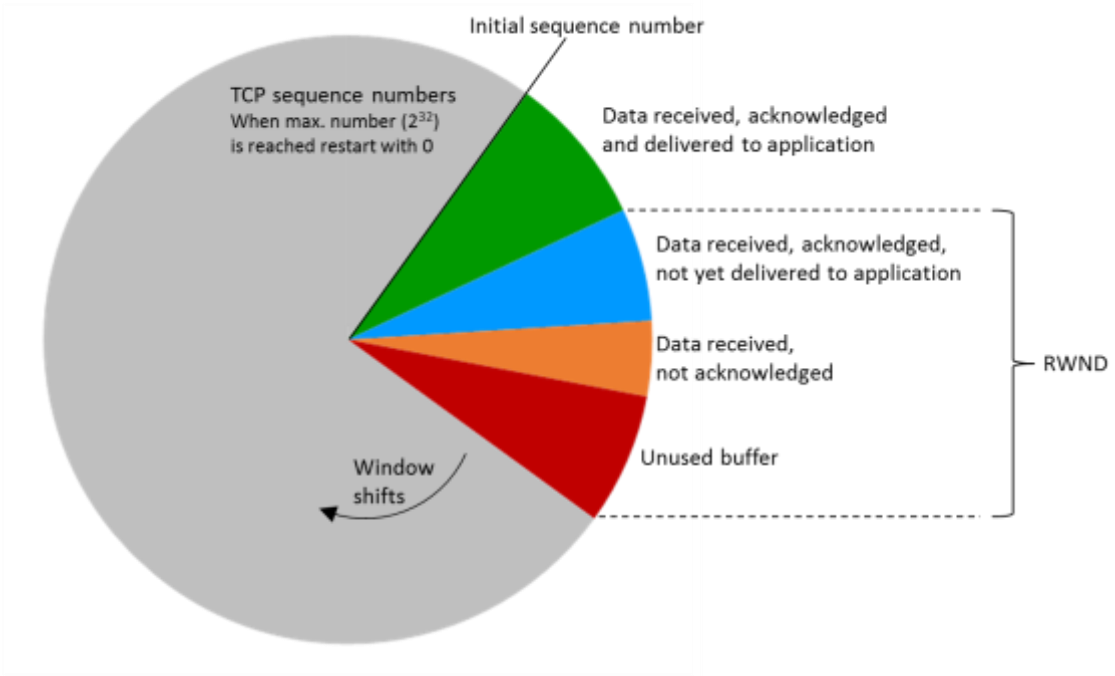

Figure 4: Sliding window operation

A TPC connection is identified as the combination of its source IP address, destination IP address, TCP source port number and TCP destination port number. IP addresses are 32 bit values (IPv4) and TCP port numbers are 16 bit values. This means that in theory $2^{32} \times 2^{16} = 2^{48}$ connections can be established from a single TCP/IP source. Although this will not happen in real life, a very high number of concurrent TCP connections can exist in a network considering the many TCP based applications users are running.

TCP Specifications

TCP is defined in a number of RFC documents:

RFC #	Description	Published
RFC 793	Transmission Control Protocol - the basic TCP specification	September 1981
RFC 1122	Requirements for Internet Hosts - Communication Layers	October 1989
RFC 2581	TCP Congestion Control	April 1999
RFC 3168	The Addition of Explicit Congestion Notification (ECN) to IP	September 2001
RFC 7414	A Roadmap for Transmission Control Protocol (TCP) Specification Documents	February 2015

Table 1: A selection of TCP related specifications

TCP Testing

RFC 6349 based Throughput Testing

The RFC 6349 "Framework for TCP Throughput Testing" provides a methodology for testing sustained TCP Layer performance. In addition to finding the TCP throughput at the optimal buffer size, RFC 6349 presents metrics that can be used to better understand the results.

RFC 6349 testing is done in 3 steps:

- Identify the Path Maximum Transmission Unit (MTU)
- Identify the Baseline Round-Trip Time (RTT) and the Bottleneck Bandwidth (BB)
- Perform the TCP Connection Throughput Tests

However before starting the TCP tests, RFC 6349 recommends that layer 2/3 tests are conducted to verify the integrity of the network. This may be manual measurements of throughput, loss, and delay. This can also be done with RFC 2544 tests (although RFC 2544 was not intended for use outside a lab) or Y.1564 tests.

The Path MTU should be identified in accordance with RFC 4821 to identify the largest frame size that can be sent without the network needs to fragment the frame. The MTU is size of the maximum TCP payload plus IP header and TCP header. RFC 4821 describes a method for Path MTU Discovery (PMTUD) where an Internet path is tested with increasing packet sizes to identify at what frame size fragmentation starts.

It is important that the device performing the TCP tests can be configured to avoid fragmentation, as fragmentation may cause that corrupted data are delivered to higher protocol layers. On the other hand the larger frames that can be sent, more payload data can be transferred per packet

relative to the overhead contained in the packet (see figure 5) leading to higher TCP throughput. With the information in figure 5 you can calculate the maximum possible TCP payload throughput, which is the actual layer 1 data rate multiplied by (1460 bytes / 1538 bytes). This means that for a 100 Mbps layer 1 line rate the maximum possible TCP payload throughput is 100 Mbps X (1460 bytes / 1538 bytes) = 94.928 Mbps (or 11.866 Mbyte/sec). If the TCP segment size is reduced, the maximum possible TCP payload throughput will also be reduced. As TCP will carry higher layer protocols that use a part of the TCP payload for their own overhead the resulting end-user payload throughput will in any case be lower than the throughput for the TCP payload.

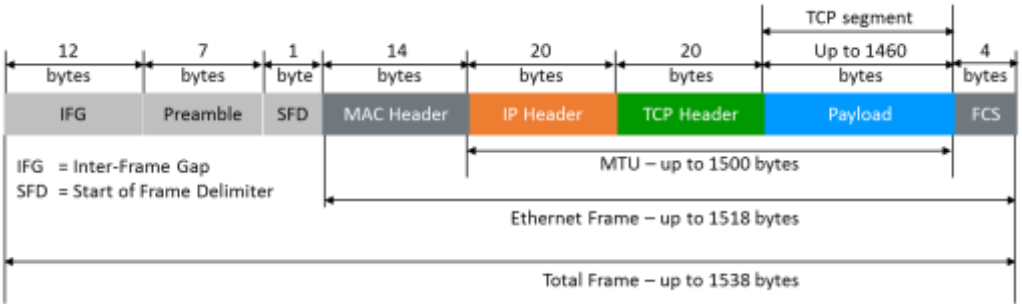

Figure 5: How the MTU and the TCP segment fits into an Ethernet/IP/TCP frame

The maximum Ethernet frame size is 1518 bytes including MAC address and FCS, which gives a MTU of 1500 bytes (figure 5). If it is necessary to verify that this frame size can actually be sent un-fragmented through a network, this can be done with a RFC 2544 Throughput test using increasing test packet sizes and the don't fragment bit set in the IP header. Network elements that need to fragment packets larger than a given size will drop these packets when the don't fragment bit is set. Hereby the MTU can be identified as the maximum frame size that passes through the network during the RFC 2544 Throughput test.

After the MTU has been identified, the inherent RTT and the BB of the end-to-end network path must be measured. Based on these measurements TCP CWND and RWND sizes can be estimated for the TCP Connection Throughput Tests: You calculate the Bandwidth-Delay Product (BDP) = BB x RTT and set the CWND and RWND sizes (which are in bytes) at (or higher) than BDP/8.

The bandwidth through a network is in some cases defined by a Committed Information Rate (CIR) agreed between a network operator and a customer. In such cases the CIR can be used as BB. In other cases the BB is defined by the line rate of the access network. In cases where the BB is unknown it can be measured with an RFC 2544 throughput test where the traffic load is increased step-by-step.

Last step in a RFC 6349 test is the TCP Throughput Tests. Single- and multiple-TCP-connection throughput tests should be performed to identify the network performance. Windows sizes at sending and receiving end should be set to match the BDP.

RFC 6349 metrics

Together with the TCP throughput measurement, RFC 6349 presents metrics that can be used to better understand the results, including:

- The TCP Transfer Time Ratio
- The TCP Efficiency Percentage

These metrics must be measured in each direction.

The TCP Transfer Time Ratio is the ratio between the time it actually takes to transfer a block of data compared with the ideal TCP transfer time i.e. what should be possible considering the BB of the Network Under Test (NUT) and the shortest possible end-to-end transfer time:

$$\text{TCP Transfer time ratio} = \frac{\text{Actual TCP Transfer Time}}{\text{Ideal TCP Transfer Time}}$$

The TCP Efficiency Percentage gives the percentage of Bytes that were not retransmitted and shows the effective data transfer relative to the total number of bytes sent:

$$\text{TCP Efficiency \%} = \frac{\text{Transmitted Bytes} - \text{Retransmitted Bytes}}{\text{Transmitted Bytes}} \times 100$$

Interpretation of TCP Throughput Test Results

The measured TCP throughput may be compared with calculated values of the ideally achievable throughput. If measured and calculated throughput matches, the network performs as you can expect. If there is a significant difference the cause should be investigated. The metrics can also be checked to better understand the issues. RFC 6349 suggest several causes, including:

- Network congestion that may cause frame loss and retransmissions
- Non-optimal windows size setting, which can cause reduced TCP throughput
- Rate limiting in the network by policing instead of shaping: If the traffic is bursty, policing may cut traffic peak loads, causing frame losses and retransmissions. Shaping smoothen the bursty traffic eliminating the high traffic peaks. Hereby frame losses may be avoided.

Additional TCP throughput tests

RFC 6349 recommends to run the TCP Throughput tests in each direction independently first to see if there are differences in the performance in the two directions. After that the test should be run in both directions simultaneously to see how this affects the performance. In all cases the measured results can be compared with calculated values of the throughput that ideally can be achieved.

In addition the TCP throughput can be measured at different window sizes up to the BDP. This may reveal cases where network performance degrades as the traffic load increases. This can also show what throughput that can be achieved if the windows are less than the BDP.

IP networks may support Differentiated Services or DiffServ, which is used for classifying, prioritizing and managing traffic through the network, based on information in the Differentiated Services Code Point (DSCP) bits in the Differentiated Services (DS) field in the IP header. Hereby different flows of traffic can be carried through the network with transfer characteristics best suited for the type of traffic and in case the network is overloaded DiffServ can indicate what part of the traffic that has lowest priority and may be discarded. This can be tested by generating traffic including several concurrent connections with different DiffServ/DSCP values.

Extreme RFC 6349 Stress Load Testing

In addition to the tests defined in RFC 6349 it is also important to examine a networks behavior when many users run TCP based applications simultaneously and see if the network capacity is exceeded. The RFC 6349 will measure the TCP performance with a single or maybe a few connections. In large networks many thousand – up to millions – of users can simultaneously activate network connections through apps on their smartphones or mobile devices. This will generate huge amounts of TCP based traffic in mobile backhaul and core networks, clouds and in data center networks, which can be stressed to the extreme. The performance of load balancers, firewalls and other devices in the networks may degrade with very high TCP traffic load, which can reduce the quality of experience (QoE) for the users of the network. To check this extreme RFC 6349 stress load testing is necessary, generating many hundred thousand or even millions of concurrent TCP connections together with a very high traffic load. This will give the service providers valuable information on the behavior of their network with under realistic conditions, which can help them to improve their position in a very competitive market.

Extreme stress load RFC 6349 testing is also relevant when networks are based on the new Software-Defined Networking (SDN) and Network Function Virtualization (NFV) like Software-Defined Wide Area Networks (SD-WAN). In these networks software based appliances run as Virtual Network Functions (VNF) on Commercial Off-The-Shelf (COTS) hardware to provide the required functionality. In this case the performance of the network does not only depend on the network itself. The VNFs and the dynamic nature of SDNs/SD-WANs will also influence the performance. With pre-stage testing or lab testing of the data plane performance on trial networks the effect of typical scenarios can be benchmarked before they are introduced into live networks.

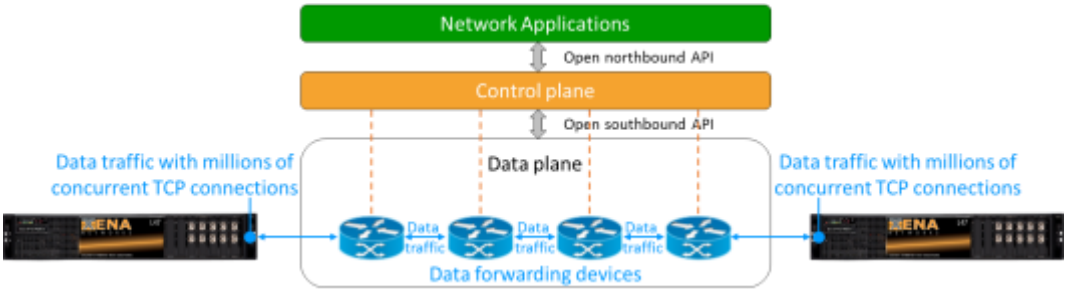

Figure 6: Extreme RFC 6349 stress load pre-stage testing or lab testing of the data plane performance in a SD-WAN

The results of extreme RFC 6349 testing can be compared with tests where just a few connections fill the available bandwidth to see how the number of concurrent connections affects the measured TCP throughput or other parameters.

WHITE PAPER

TCP Testing with Xena Networks Test Solutions

RFC 6349 based TCP throughput testing and stress load testing is of course supported by Xena Networks test solutions. To generate test signals with stateful TCP traffic for throughput testing and for generation of a very high number of concurrent TCP connections the Xena Networks testers supporting layer 4-7 - XenaScale and XenaAppliance – are the obvious choice.

Testing at lower layers like RFC 2544 testing is supported by the XenaBay and XenaCompact test chassis equipped with relevant test modules.

The Xena Networks application note: "TCP Throughput testing" explains in details how to test TCP throughput performance based on RFC 6349 with Xena Networks testers.

Testing above Layer 3

Figure 7: The powerful Xena Networks Layer 4-7 testers XenaScale and XenaAppliance

Xena Network's XenaScale and XenaAppliance can be used to generate TCP, HTTP/TCP and UDP traffic streams simultaneously. In addition both products offer stateful end-to-end testing of network appliances such as switches, firewalls, routers, NAT routers, proxies, load-balancers, bandwidth shapers and more. The platform is also suitable to characterize entire network infrastructure performance for TCP. Top features include:

- Wire-speed stateful TCP traffic generation and analysis
- Stateful TCP traffic load generation for millions of connections
- Configuration and tuning of Ethernet, IP and TCP header fields for advanced traffic scenarios
- Stateful TCP connection blasting
- HTTP get/put/head/post blasting
- Extensive live stats and test reports
- 1G – 10G Ethernet interfaces
- 40G Ethernet interfaces (XenaScale)
- High port density – up to 12 x 10 GigE (XenaScale)
- Configurable allocation of processing resources to Ethernet test ports
- Free XenaConnect traffic generation and analysis software makes basic testing quick and easy
- Wire-speed traffic capture
- Switched and routed network topologies, TCP proxy and NAT support
- Export packet capture to industry standard pcap/Wireshark

Testing up to Layer 3

Figure 8: The versatile and powerful Xena Networks Layer 2-3 testers XenaBay and XenaCompact

Testing at lower layers is supported by the XenaBay and XenaCompact test chassis equipped with relevant test modules, which can support data rates up to 100 Gbps. Up to 12 test modules can be installed in the XenaBay.

Based on Xena's advanced architecture, XenaBay and XenaCompact equipped with relevant test modules are proven solutions for Ethernet testing at layers 2 and 3. Advanced test scenarios can be performed using the free Xena test applications:

XenaManager-2G test software is used to configure and generate streams of Ethernet traffic between Xena test equipment and Devices Under Test (DUTs) and analyze the results.

Xena2544 offers full support for the 4 test types specified in RFC 2544: Throughput, Latency, Frame loss and Back-to-back frames; Jitter (Frame Delay Variation) is also supported.

Xena1564 provides full support for both the configuration and performance test types described in Y.1564 for complete validation of Ethernet Service Level Agreements (SLAs) in a single test.

Xena2889 is an application for benchmarking the performance of Layer 2 LAN switches in accordance with RFC 2889.

Xena3918 makes it easy to create, edit and execute all test types specified in RFC 3918. RFC 3918 describes tests for measuring and reporting the throughput, forwarding, latency and Internet Group Management Protocol (IGMP) group membership characteristics of devices that support IP multicast protocols.

XenaScripting is another free application for XenaBay and XenaCompact. It is a powerful and easy-to-use command-line-interface (CLI) scripting API that makes test automation easier for test engineers.

CONCLUSION

Incorrect dimensioning of buffers for the layer 4 Transmission Control Protocol (TCP) may cause that customers experience performance degradation even though service providers can prove that their layer 2/3 IP network operates in accordance with a Service Level Agreement (SLA) signed with the customer.

RFC 6349 provides a methodology for testing sustained TCP Layer performance. In addition to finding the TCP throughput at the optimal buffer size, RFC 6349 presents metrics that can be used to better understand the results.

The Xena layer 4-7 test solutions XenaScale and XenaAppliance together with the layer 2-3 test solutions XenaBay and XenaCompact support powerful TCP testing based on RCF 6349. In addition XenaScale and XenaAppliance support extreme RFC 6349 testing with millions of concurrent TCP connections, giving service providers valuable information on the number of users and connections the network can handle, highlighting capacity bottlenecks.