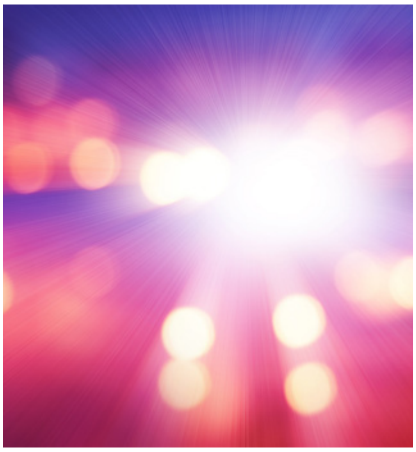
SANDVINE



TCP Accelerator

Take your network to the next level by taking control of TCP

Sandvine's TCP Accelerator allows communications service providers (CSPs) to dramatically improve subscriber quality of experience (QoE) and network efficiency, without making changes to the network infrastructure.

In today's networks, TCP's very nature causes problems:

- When additional bandwidth is available, TCP may not be fast enough
- When too little bandwidth is available, TCP may be too fast
- When there are many concurrent TCP connections, the collection as a whole behaves inefficiently

For CSPs, these problems have two significant consequences:

1. Network efficiency is harmed, causing performance and return on investment to fall below expectations
2. Subscribers are dissatisfied due to poor QoE with applications

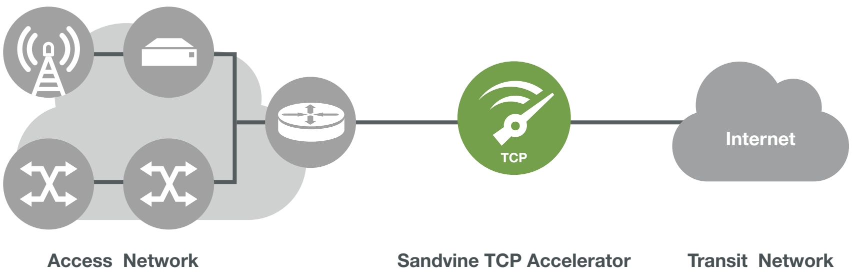
KEY SYSTEM BENEFITS

- Improved Subscriber QoE: faster data transmissions and increased application performance lead to measurably, consistently better subscriber quality of experience
- Increased Network Performance: higher ratio of goodput to throughput, better resource utilization, reduced retransmissions, and extended legacy infrastructure lifetimes
- Increased Revenue Opportunities: subscribers tend to use more data when they are enjoying faster, more reliable connections
- Rapid ROI: more efficient resource utilization defers capital, and lets access network resources run 'hotter' without compromising QoE

ENGINEERED FOR TODAY'S NETWORKS

The Sandvine TCP Accelerator is engineered to make TCP run better in today's networks:

- When additional bandwidth is available, TCP Accelerator ensures TCP doesn't become a bottleneck by accelerating the slow-start phase and by making sure there is always enough data ready to be served
- When too little bandwidth is available (e.g., from congestion, spotty mobile coverage, or a low-speed service plan), TCP Accelerator reduces effective latency by preventing bufferbloat in access network resources
- By transparently bridging the access network with the transit network, TCP Accelerator is in a position to manage all the network's TCP connections as a collective whole, to ensure maximum efficiency



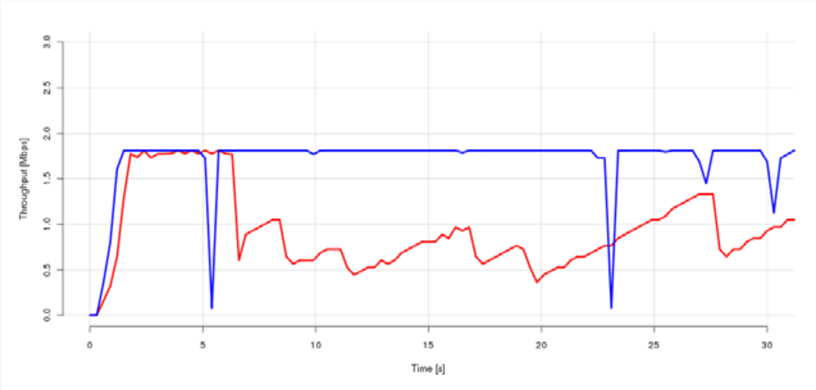
Transfer Size	10KB-50KB	50KB-100KB	100KB-500KB	500KB-1MB	1MB-5MB	5MB-10MB	>10MB
Uploads	+18%	+12%	+32%	+36%	+38%	+38%	+31%
Downloads	+11%	+17%	+24%	+43%	+43%	+33%	+24%

This table shows speed-up results (i.e., how much faster data was transmitted) in a customer's LTE network.

GET THE MOST OUT OF YOUR NETWORK

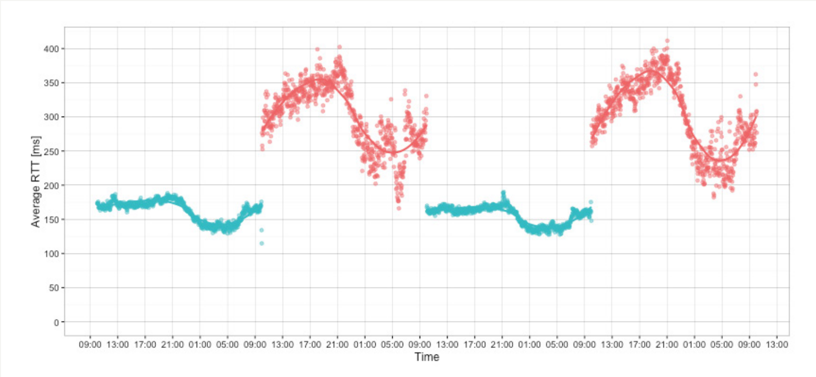
More Efficient Data Transfers

TCP Acceleration lets you (and your subscribers) get the most out of your network. The graph below, showing a single TCP flow, illustrates how the accelerated traffic (blue) gets up to speed more quickly, is faster overall, is more consistent, and recovers more quickly after a genuine packet loss.



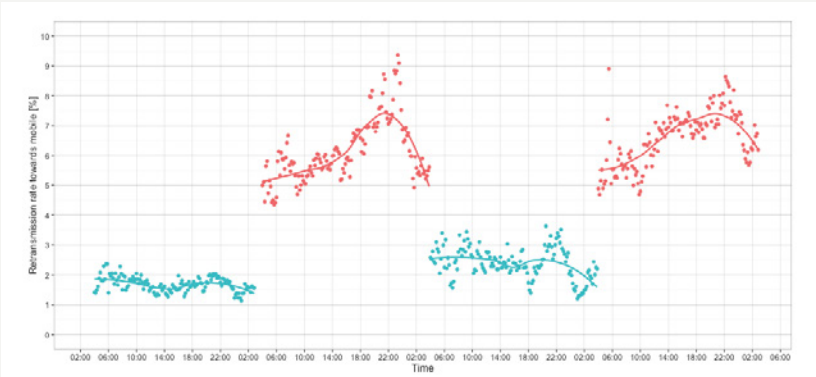
Faster and More Consistent Round-Trip Times

The graph on the right shows the difference in flow round-trip times when TCP Acceleration (including buffer management) is enabled and when it is not enabled. In this mobile network, the average round-trip time was cut in half (164ms average versus 320ms average), and round-trip times showed much greater consistency.



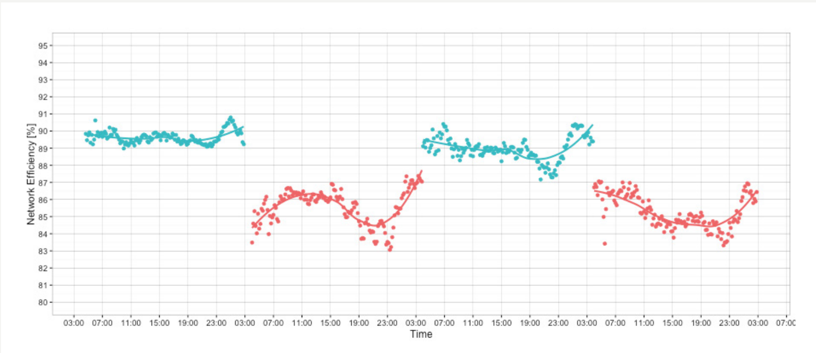
Fewer Retransmissions

This graph, from a CDMA network, shows the enormous positive effect on retransmission rate: when TCP acceleration is enabled, the retransmission rate is much lower, and much more consistent.



MORE EFFICIENT NETWORKS

The graph below, also from a CDMA network, shows the overall positive impact on network efficiency as a result of TCP acceleration. In this example, network efficiency (goodput as a percentage of all throughput) improved by 5% and exhibited much tighter consistency.



KEY FEATURES OF TCP ACCELERATOR

Transparency

The Sandvine TCP Accelerator behaves as a bridge and doesn't terminate the TCP connection, so the acceleration is completely transparent to the endpoints, yielding a number of benefits:

- Connection migration: connections can be migrated from optimized to non-optimized without affecting the connection from the perspective of either endpoint
- Connection resumption: long-lived but inactive connections time-out after a prolonged period of inactivity to recycle state memory; because of transparency, if a connection times out of TCP Accelerator, but then resumes, the connection is able to resume seamlessly
- Sequence number preservation: sequence numbers are untouched, which ensures that the TCP connection state in the client and server will be the same after the TCP handshake
- Step-out for easy upgrades: when acceleration is halted and the stores and forward buffers are cleared, TCP Accelerator can be stopped within minutes, without disrupting traffic
- TCP option transparency: if new TCP options or features are introduced, the endpoints can still negotiate its use (unlike terminating proxies, which would prevent its usage entirely)

Powerful TCP Acceleration Techniques

TCP represents between 85% to 90% of fixed access Internet traffic, and as much as 96% of mobile traffic, and TCP Accelerator applies acceleration to practically all of it, including:

- Uploads and downloads: acceleration is applied to upload and download traffic
- Any application: the default configuration accelerates traffic agnostic of application, protocol, service, etc.
- Encrypted traffic: with 70% of traffic expected to be encrypted by the end of 2016, it's critical that solutions work with encrypted traffic, and TCP Accelerator does
- HTTP2: fully supported

The only TCP that isn't accelerated is the traffic that shouldn't be accelerated. For instance, if the traffic originates from a specialized device (e.g., a sensor or probe) that should be omitted from TCP acceleration, then TCP acceleration is not applied.

In addition to the buffer management features described below, TCP is accelerated with a combination of techniques, including:

- Two-sided acceleration: most configuration variables can be set separately for the access side and the Internet side
- Reduced packet loss effect during slow-start: tunable system for mitigating the impact of TCP packet loss early in the connection, by allowing such packet loss to be ignored for the purposes of adjusting the congestion window
- Congestion control: distinguishes between genuine congestion events and radio glitches (in mobile networks) and handles each accordingly
- Fast retransmit: supports TCP Fast Retransmit, with configurable behavior
- Improved retransmission handling: replaces the standard timeout-based TCP retransmit entirely for the access side of the connection, which has an enormous positive impact on mobile networks (where hand-overs and jitter can cause spurious retransmissions)

Plus, TCP Accelerator is in a position to manage all the network's TCP connections as a collective whole, to ensure maximum efficiency.

TCP Buffer Management

When high-bandwidth downloads fill the buffers (queues) in the access network, other traffic gets stuck behind and the subscriber experience suffers.

To prevent this scenario—called bufferbloat—in the access network, round-trip times need to be carefully managed; at the same time, a minimum amount of data has to be sent so as not to starve the radio network.

TCP Accelerator manages buffer queues by adjusting the sending rate to correspond to the level of buffered data in the access network. When TCP Accelerator determines that too much data is being queued for a particular subscriber, it sends less data for the subscriber in order to give time for the queue to shrink. TCP Accelerator learns the base latency for each subscriber, which accounts for the varying radio access technologies and roaming subscribers.

Importantly, TCP Accelerator prevents latency-insensitive applications from being favored over latency-sensitive ones. This approach treats all traffic fairly, and has an enormous positive impact on subscriber quality of experience because it ensures sensitive applications aren't starved.

Egress Burst Control

When a large amount of data is released at once from a server close to the core network, this traffic can overwhelm any switch where there is a speed differential (e.g., going from 10G to 1G); the buffers in the switch may be insufficient to accommodate micro-bursts resulting from the speed differential, and as a result they might be forced to simply drop packets. This behavior has a negative impact on the experience of all subscribers on the impacted switch.

To prevent this scenario from unfolding, Sandvine's TCP Accelerator prevents buffer overflow by limiting the amount of traffic sent to a subscriber in a configurable fraction of a second (typically a millisecond). This limiting allows the switch to handle the burst without exhausting its buffers, significantly reducing retransmission rates.

This functionality can relieve a network operator from replacing a huge number of switches.

Universal Access Support

TCP Accelerator can be deployed in any type of access network, with any combination of access technologies (e.g., Cable, DSL, 3G, 4G, WiFi, Satellite, WiMAX, etc.).

In mobile networks, TCP Accelerator even supports deployment in shared RAN environments with a Multi-Operator Core Network (MOCN).

Multiple Acceleration Profiles

Distinct acceleration profiles (consisting of tuning parameters) can be created using any Ethernet, IP, or TCP header fields, such as IP ranges, TCP ports, etc. These profiles can then be applied to specific traffic; for instance, to manage high-bandwidth points of presence and low-bandwidth points of presence differently.

Carrier-Grade Performance

The Sandvine platform scales to support the world's largest networks, so you can enjoy the benefits of TCP acceleration no matter the scale.

Multiple Operational Modes

The system has three operational modes:

- Forward mode: a software pass-through (i.e., no traffic is accelerated) that is used to perform measurements and provide visibility into the performance of TCP without acceleration in place; in Forward mode, all traffic is forwarded.
- Accelerate mode: all available acceleration mechanisms are applied to all relevant traffic. Accelerate mode can be applied to a subset of traffic; for instance, for the purpose of comparing the performance of accelerated traffic to traffic that isn't accelerated (this traffic is simply forwarded). If any resource within TCP Accelerator (e.g., processing, memory) becomes exhausted, then new flows are simply forwarded until resources become available.
- Bypass mode: specialized network interface cards physically bypass TCP Accelerator, triggered by a watchdog process; if TCP Accelerator is not deployed on network interface cards with physical bypass, then it can be configured to drop traffic matching a standard PCAP filter while operating in Bypass mode. For instance, using this technique to drop BFD frames or specific ICMP traffic can trigger a failsafe by configuring next-hop routers to convergence based on BFD/IP-SLA features.

Audit Records and Historic Reporting

TCP performance measurements and statistics are logged and can be used for audit purposes or examined for business and operational intelligence.

DEPLOYING TCP ACCELERATOR

TCP Accelerator is deployed as a software image running on COTS hardware, with two deployment models

- Mobile Access Networks: Installed on SGi/Gi-LAN as a layer-2 transparent node
- Fixed Access Networks: Installed anywhere with symmetric traffic and supported encapsulation

TCP ACCELERATOR HARDWARE SPECIFICS

- Assumes a 90% TCP downlink
- 1-2TB disk storage

Hardware Spec	Gbps
Intel® Xeon® E5-2630 v2 (2.6GHz/6c) – 16GB RAM	1
Intel® Xeon® E5-2650 v3 (2.3GHz/10c) x2 – 32GB RAM	5
Intel® Xeon® E5-2690v2 (3.0GHz/10c) x2 – 32GB RAM	10
Intel® Xeon® E5-4627v3 (2.6GHz/10c) x4 – 128GB RAM	20
Intel® Xeon® E7-8890v3 (2.5GHz/18c) x4 – 256GB RAM	40

TCP ACCELERATION IN ACTION

TCP Accelerator transparently takes control of TCP connections that are eligible for acceleration; non-TCP packets, including any non-IP packets, are simply forwarded, as are TCP packets that are to be omitted from acceleration.

The basic techniques of TCP acceleration are best explained via TCP flow diagrams; the diagrams below represent an HTTP connection, and show how latency-splitting is fundamental to TCP acceleration.

Accelerating Slow-Start (See Figure 1)

When TCP Accelerator receives a SYN packet (1) that is eligible for acceleration, it creates a connection record internally corresponding to the 5-tuple of the SYN packet.

The SYN is sent to the destination server as-is. Likewise, the SYN/ACK response is sent as-is to the client. Importantly, TCP Accelerator does not split the connection.

After the SYN, SYN/ACK, and the three-way handshake is completed with the ACK packet (2), the connection is established between the client and the server. At this point, TCP Accelerator has all the necessary information to represent the client when communicating with the server, and vice versa.

TCP Accelerator acknowledges data segments on behalf of the receiver (3); the ACK packet is created by TCP Accelerator, complete with relevant information and addressing (e.g., MAC addressing, TTL, sequence numbers, receive window, etc.).

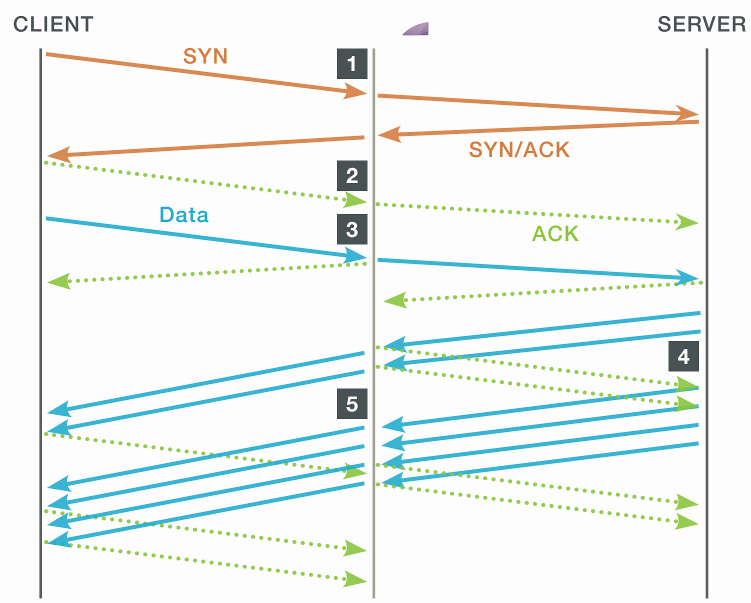
The data segment received at point (3) is sent to the destination, and is buffered in case it needs to be retransmitted. The optimization works symmetrically for downloads and uploads, but TCP Accelerator's TCP/IP stack is tuned differently for each side of the data transfer.

To maximize efficiency, the buffers and advertised windows are adjusted by taking into account the speed observed for the connection on the access side and Internet side of TCP Accelerator.

As the ACKs from TCP Accelerator arrive at the server, they trigger more data transmission (4). At this point, the server is in TCP Slow-Start: each ACK triggers the sending of one more data segment than before. TCP Accelerator is able to ACK much faster than the client (i.e., the connection to the server will have less latency and higher reliability), so the slow-start phase is accelerated.

The four additional data segments arriving from the server are sent towards the client immediately, with the two previously sent segments still pending in-flight (5); this happens because the 'initial congestion window' of TCP Accelerator is set to a high enough value to match the bandwidth of the network.

Figure 1



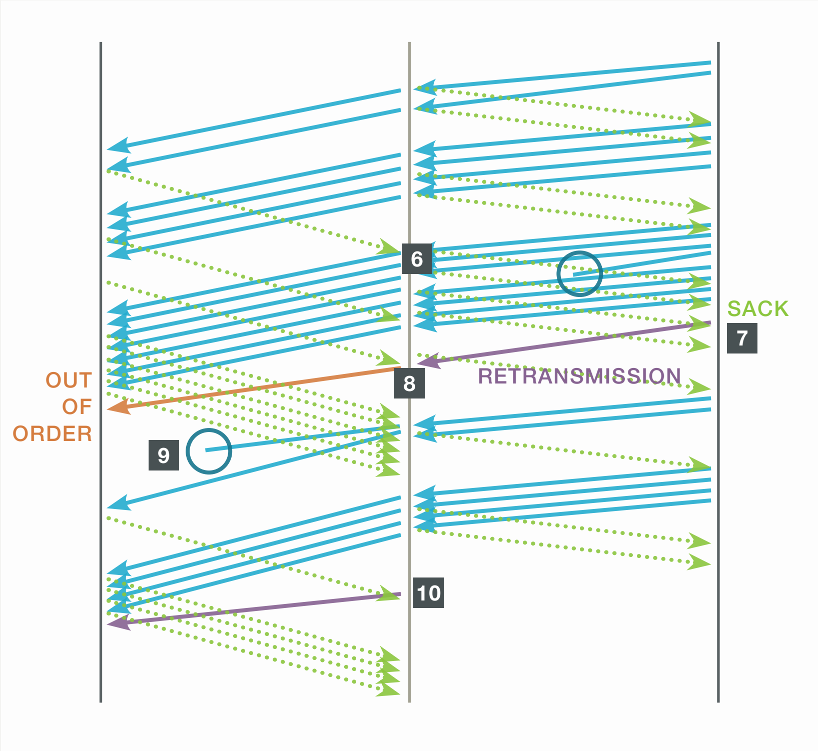
Faster Resolution from Packet Loss (See Figure 2 below)

If a packet is lost on the Internet side of TCP Accelerator (6), it will respond with selective acknowledgements (SACK); the SACK lets the server determine which packet is lost. Since TCP Accelerator works on a per-packet basis, transparently, it does not need to wait until the Internet side packet loss is resolved, and instead can forward data segments that arrive after the lost packet (6).

The SACK triggers the server to retransmit the lost segment (7); this retransmitted packet is a 'gap-filling' packet from the point of view of TCP Accelerator, and is forwarded directly to the client (8).

If a packet is lost on the access side (9), then the client responds with a SACK and TCP Accelerator retransmits directly from its buffer (10).

Figure 2



v20180129

ABOUT SANDVINE

Sandvine helps organizations run world-class networks with Active Network Intelligence, leveraging machine learning analytics and closed-loop automation to identify and adapt to network behavior in real-time. With Sandvine, organizations have the power of a highly automated platform from a single vendor that delivers a deep understanding of their network data to drive faster, better decisions. For more information, visit **sandvine.com** or follow Sandvine on Twitter at **@Sandvine.**

SANDVINE

USA
47448 Fremont Blvd,
Fremont,
CA 94538,
USA
T. +1 510.230.2777

EUROPE
Birger Svenssons Väg
28D
432 40 Varberg,
Sweden
T. +46 (0)340.48 38 00

CANADA
408 Albert Street,
Waterloo,
Ontario N2L 3V3,
Canada
T. +1 (0)519.880.2600

ASIA
Ardash Palm Retreat,
Bellandur, Bangalore,
Karnataka 560103,
India
T. +91 80677.43333