MASV Accelerator™
Technology Overview

Introduction

Most internet applications, FTP and HTTP to name a few, achieve network transport via the ubiquitous TCP protocol. But TCP suffers from latency, packet loss, congestion and sub-optimal TCP stacks, which leads to long transfer times and a degraded user experience.

MASV Accelerator can increase your throughput by 25% to 85% over real-world networks, and we've seen 2X to 10X improvement on impaired networks or wild public networks like airports, hotels and coffee shops.

What follows is an overview of the technology behind MASV Accelerator and some discussion about test strategies for evaluating MASV Accelerator performance.

MASV Accelerator Overview

MASV Accelerator is an auto-tuning, per-connection TCP optimizer that makes more efficient use of available bandwidth, speeding up file transfers and video stream delivery.

Available as a quick-install network driver (Win, Mac, and Linux), MASV Accelerator can be deployed as a single-sided solution on the sending
device: on client hardware for upload acceleration, and on cloud infrastructure for download acceleration.

By design, regular TCP aims to ramp up transfer speed carefully, then progressively faster, until it perceives network congestion via packet loss detection or “buffer bloat”.

When congestion is detected, regular TCP stacks drastically reduce data transmission speed, in an effort to be network-friendly. Once its perception of network congestion subsides, TCP speeds up again, probing bandwidth availability until it again perceives network congestion via packet loss – this cycle continues for the duration of the transfer.

We often see an overreaction to packet loss, as an indicator of network congestion, which in turn, leads to uneven performance and loss of bandwidth opportunity, as depicted below.

On the other hand, MASV Accelerator intelligently measures congestion, accounts for network volatility and displays a smoother, often near-maximum throughput characteristics – making the most of your network connections.

**Congestion Avoidance**

TCP stacks aim to avoid congestion using goal-seeking algorithms. These algorithms (e.g. cubic, reno, vegas, etc.) each have their pros and cons by addressing different types of networks and different network conditions.

Direct LAN transfers exhibit different performance characteristics than transfers to/from the cloud, because they have different network
characteristics such as ping time, jitter, maximum bitrate, congestion severity and packet loss.

Wi-Fi networks present different challenges than purely wired connections, such as Wi-Fi contention or noise issues.

So it’s difficult to conclude that any single algorithm will fit all network paths, in all circumstances.

What’s required, is an implementation of TCP that can make sense of the underlying end-to-end network, and that can deploy the right optimizers at the right time, on a per-connection basis – there is such a solution and it’s called MASV Accelerator.

**TCP Stack Tuning**

Operating systems allow system administrators to tune the TCP stack. However, the number of parameters involved, their impact on various aspects of TCP performance, as well as the complexity in considering application use-cases, mixed software deployments and network topologies, make this a particularly challenging endeavour.

The question becomes, what would you rather be doing with your time? On the other hand, we’ve been working on MASV Accelerator since 2012 and MASV Accelerator is now accelerating millions of devices.
How is MASV Accelerator Different?

• Deployment is simple: it’s single-sided. Unlike other TCP and UDP solutions, MASV Accelerator is only required on the sending device.

• License keys can be shared with your partners and clients, and can be moved seamlessly between machines, depending on your subscription plan.

• MASV Accelerator performs dynamic auto-tuning on a per-connection basis: it’s a single solution to handle all network paths, simultaneously.

• Unlike other TCP solutions, MASV Accelerator can fill huge pipes, those with a large bandwidth-delay product (BDP) that you might find between datacenters or on high-end private networks.

• When operating over perfect network conditions, MASV Accelerator becomes your insurance policy: should conditions degrade, users might not even notice.

• All TCP applications can be optimized, including FTP, HTTP/S, HTTP/2, ssh, rsync, NFS, Samba, RDP, ICA, SNMP and more.

From moving or sharing files, to synchronizing databases across regions, to streaming live video, MASV Accelerator has your back.
Performance

MASV Accelerator is a software solution and runs as a subsystem of the operating system – as a driver or kernel module. It optimizes the use of existing bandwidth where regular TCP fails to do so. It can generate throughput gains and reduce turnaround time, freeing up resources more quickly.

Obviously, MASV Accelerator cannot add new bandwidth; the only way to add new bandwidth is to upgrade the network underpinning the slowest link.

Typical MASV Accelerator throughput gains fall into the following categories:

- **Wow**: 2X - 10X (1 hour transfer time reduced to 30 minutes or less)
- **Great**: 50% to 100% (1 hour transfer time reduced to 35 minutes)
- **Good**: 10% to 50% (1 hour transfer time reduced to 45 minutes)
- **No Change**: Conditions are close to optimal or resources are at maximum (CPU, bandwidth, memory)

SaaS providers with a large following will typically see a mix of results that vary based on their users’ geographical distribution, underlying network conditions, time of day and Wi-Fi connections. We consider this mix of results a success if MASV Accelerator improves speeds for many users and provides an insurance policy for others, should conditions degrade.

In some situations, there is no opportunity to improve performance. This can be seen for example, on networks where the
bandwidth is being rate-limited, or where the bandwidth-delay product (BDP) is small enough that regular TCP does a fine job on its own, or the network conditions are already optimal, or the sender/receiver are CPU-limited, or some combination of these factors.

If muted results are surprising or unexpected, it can suggest the existence of an unknown (or forgotten) intervening device that is terminating the TCP span, effectively segmenting the TCP connection.

As an example, consider the case of a well-intentioned installation of MASV Accelerator on a server that is front-ended by an NGINX reverse-proxy with TLS offloading. In this scenario, there are 2 TCP spans: one span between the server and NGINX, and another span between NGINX and the mobile client.

If the server and the NGINX reverse proxy are in the same LAN and the ping time is near zero, there’s no opportunity to improve the throughput between these two devices.

However, if MASV Accelerator is installed on the NGINX reverse-proxy, it will likely improve the throughput between the reverse-proxy and the user, as the HTTPS traffic would traverse several networks towards the edge, and across a Wi-Fi hop, all in one TCP span.

Alternatively, if servers are connected across regions, it would make sense to install MASV Accelerator on both sides and optimize throughput in both directions.
Reference Topologies

The following diagram attempts to depict deployment options and indicates where MASV Accelerator might be positioned to speed up data transfers.

- A: Inter-region data sync, bidirectional acceleration
  **Large BDP, public or private networks**
- B: Enterprise file sharing with download acceleration
  **Corporate, wired network**
- C: Enterprise file sharing with bidirectional acceleration
  **MASV Accelerator deployed on-premise**
- D: Regional proxy for download acceleration
  **Serving mobile users and edge nodes**
- E: Road warrior equipped with MASV Accelerator for upload acceleration
  **Laptop/desktop, SOHO or on the road**
- F: Edge node providing download acceleration
  **Do-it-yourself CDN**
Evaluating MASV Accelerator

There are two high level methodologies that we use to evaluate MASV Accelerator performance:

- Head-to-head A/B tests
- Back-to-back A/B tests
- Head-to-head A/B Test

In head-to-head testing, MASV Accelerator and native TCP performance are compared by performing a simultaneous transfer over the same network, effectively setting up a resource competition (CPU, bandwidth, memory) between senders A and B.

The method ensures that the underlying network conditions are the same for both competitors and is particularly useful in filtering out noise in the results when going over public networks.
Each sender (A and B) can be deployed on separate server instances assuming they are co-located and share the same specifications.

On the receiving end, there may be one or two receivers. In a dual receiver setup, they should also be co-located and of the same ilk.

Back-to-back A/B test

In back-to-back testing, MASV Accelerator and native TCP performance are compared by performing sequential-in-time data transfers (first A, then B) over the same network, effectively setting up a high-score competition (throughput or turnaround time) between senders A and B.

Clarity in the results is best achieved in a private network, since each transfer may be subject to completely different network conditions in real-world environments.
However, public networks can also be used as long as outlier results are filtered out and min/mean/average/max results are compared.

Each sender (A and B) can be placed on the same server instance (MASV Accelerator on/off), or on separate server instances assuming they are co-located and share the same specifications. On the receiving end, only one device is required.

![Graph showing Bandwidth Opportunity]

**Bandwidth Opportunity**

Whether you’re looking at MASV Accelerator to address technical issues or you’re wanting to build the best possible product, we want to help you understand where best to deploy MASV Accelerator in order to get the most value.

Taking a step back, the best way to optimize data transfer, is to not have to send the data at all. If this peaks your interest, ask us about our LiveSync and CDN Shield solutions.
In terms of TCP, the following sections will help us identify parts of the network that are likely to benefit from MASV Accelerator acceleration.

**Parameters Affecting Performance**

TCP performance is sensitive to the following network characteristics:

- Round-trip time
- Wi-Fi packet loss
- Packet loss due to congestion
- TCP sender and receiver window sizes

**Packet Loss & Round-trip Time**

The well-known ping utility can be used to calculate round-trip time and estimate packet loss. However, the target host must support ping; i.e. its firewall must allow ping packets to enter and the host must support replying to ping requests.

The following excerpt shows a large variation in RTT, from the minimum time required to travel from the East Coast to California and back at 78 ms, to the maximum reported time of 368 ms.

```
1 $ ping -c 100 54.67.46.39
2 PING 54.67.46.39 (54.67.46.39) 56(84) bytes of data.
3 64 bytes from 54.67.46.39: icmp_seq=1 ttl=53 time=79.5 ms
4 64 bytes from 54.67.46.39: icmp_seq=2 ttl=53 time=83.2 ms
5   ... 
6 64 bytes from 54.67.46.39: icmp_seq=28 ttl=53 time=367 ms
7 64 bytes from 54.67.46.39: icmp_seq=29 ttl=53 time=386 ms
8 64 bytes from 54.67.46.39: icmp_seq=30 ttl=53 time=111 ms
9   -- 54.67.46.39 ping statistics --
10 100 packets transmitted, 90 received, 5% packet loss, time 9936ms
11 rtt min/avg/max/mdev = 78.166/109.315/368.924/53.573 ms
```
This variation in RTT is an indicator that there is quite a bit of network congestion (buffer bloat), as packets queue up in routers or servers until they arrive at their final destination.

Further, the trace shows that 5 requests were dropped. While ping reports this as 5% packet loss, it’s unlikely that the network actually suffers from 5% packet loss on average.

It is more likely that there is volatile network congestion, characterized by high instantaneous loss rates, while the average loss rate is near zero. Still, packet loss can have a deflating effect on TCP throughput, as a small amount of packet loss can mean a large reduction of the sending rate. Thus, an instantaneous problem has long-lasting consequences on data transfer. MASV Accelerator is likely to show substantial improvement under such conditions.

Another interesting tool named MTR (aka My Traceroute) on Linux and Mac, as well as its Windows counterpart WinMTR, can discover intermediate hops, while also providing an estimation of packet loss and round-trip time.

In the above trace, we can see that there were 4 dropped requests in my office (I was using a Wi-Fi enabled Macbook Pro on a 100 Mbps corporate connection).
When packets are dropped on a network path where it is quite unlikely that congestion is at play, we can begin to suspect that Wi-Fi contention and/or Wi-Fi interference may have been at play. Similarly to packets dropped at deeper points within the network, Wi-Fi packet loss can have a disproportionate effect on TCP throughput reduction.

In this kind of Wi-Fi environment, backed by substantial upstream bandwidth, MASV Accelerator is likely to show upload improvement when installed on my Mac, and download improvement if installed in the Cloud.

Applications

In past experiments, we’ve noticed that some applications are quite sensitive to network conditions. Some of them seem to set small buffer sizes, others unsuccessfully attempt to control throughput from the application layer, and others suffer from application behaviour not being optimally paired with TCP.

If you’re using any of these applications and/or your network suffers from some of the impairments we’ve discussed, it’s likely that MASV Accelerator can improve user experience.

- **FileZilla**
  The well-known FTP application used by millions to transfer files; gains of 2X-5X are possible.
- **NGINX**
  The most loved Linux-based reverse-proxy and web server; gains on the order of 25%-85% are likely.
- **Apache**
One of the most popular web servers; gains on the order of 25%-85% are likely.

- **RDP**
  Perhaps the most popular remote desktop protocol; gains of 2X-10X are possible.

- **rsync, scp**
  These file copy and synchronization tools are well known to developers; gains of 25%-85% are likely and 2X-10X are possible.

- **Video streaming**
  Whether video is delivered over HTTP or RTMP, to wired desktops of Wi-Fi apps; gains on the order of 25%-85% are likely and 2X-10X are possible.