Transferring Earth observation data around the globe

Christos Argyropoulos, GRNET NREN Network-Research Engineer
Keywords: big-data, large-scale, high-performance-networking, host-tuning, long-fat-networks

Presentation Description:

Our presentation will describe the network obstacles we faced and eventually overcame trying to deliver European Space Agency Sentinel (ESA) satellites' data faster to the scientific community.

Abstract:

The path for the Earth observation data, from the European Space Agency Sentinel (ESA) satellites [https://sentinels.copernicus.eu/web/sentinel/home] all the way down to the end-user is a complex one, starting from the satellite, involving receiving stations as far away as Svallbard, including distributed processing centres for data calibration and geocoding, especially considering the data should be delivered as fast as possible to the user communities in order to run models and make predictions for environmental effects. GRNET is participating to this effort as one of the service/content distribution points for ESA's Earth observation data, contributing to the European Union's flagship Copernicus programme [http://www.copernicus.eu/].

Taking part to the network design and operation process we had to assure that the data would be transferred fast enough from the main service distribution DC, located to central Europe, to our DC in Greece which acts as complementary DC.

From capacity planning perspective, we knew that our WAN network and our upstream peerings to the pan-European research and education network (GÉANT) could achieve speeds of many Gbits per second. But, somehow, when the testing period started, the transfer speed was not adequate and the number of the datasets (files) per day, with a total size of many Terabytes, that were in the queue waiting to be transferred to our DC was constantly increasing day-to-day. It was easy to see the problem, but difficult to understand what was the cause. One of our first thoughts, that of the geographical distance (a.k.a propagation delay in networking) leading to TCP throttling was indeed the source of the problem, but an invisible hunter of our skills was drawing a red herring across the path.

Problem statement

The network requirement could be described as: "A constant flow of large files should be delivered daily, as fast as possible, from one DC to another DC residing thousands miles away, over Long Fat Networks (LFN)".

Data are distributed from the main DC of the ESA, in central Europe, to other DCs across Europe. One of these DCs is a newly-built GRNET DC connected to our WAN network with multiple 10Gbit links. The software handling the file download is using TCP connections for data transferring and resides on multiple Virtual Machines on multiple servers across the DC.

After setting up the entire DC infrastructure we started to download datasets. The operation team noticed that the backlog of the files that were waiting to be downloaded was large. Even worse, new datasets were becoming available with fresh data from the satellites to the main DC.

Troubleshooting Approach

Our newly-built DC is using a spine-leaf collapsed CLOS topology with EVPN-VXLAN implementation as main pillars for the network fabric. The spine switches that also act as DC routers are connected to carrier routers that are running MPLS/IS-IS and provide L2 services in order to connect the DC routers to our main IP routers. The links among the carrier routers are implemented as optical services from our DWDM optical network. Each layer of the aforementioned WAN and DC Fabric are redundant to node, link and routing engine level. Some optical links are also protected. The overall number of the paths between end-hosts located to two different GRNET's DCs is at least 128.

Servers that host the service VMs are also multihomed to Top-Of-the-Rack (TOR) leaf switches. Ganetti/KVM is the Infrastructure-as-a-Service (IaaS) platform we are using. And on top of that the service specific software is running in order to download the file and make smart things that are beyond the scope of this text. Needless to say that with so many software and hardware components taking part to the service provisioning there was no other path rather than trying to make an educated guess as a starting point and adopt a top-down approach for our troubleshooting.

The educated guess

The main DC is located in central Europe and our capacity view is limited to the GRNET and GEANT network, which are the two of the multiple pieces of the network topology puzzle. We had no view to the rest of the network links. Nonetheless, (1) taking into consideration the report that another DC somewhere in central Europe is downloading with higher speeds from the main DC, (2) not knowing exactly what the numbers behind this vague description are, (3) having assured there is no congestion inside our DC and WAN topology, we start thinking the TCP characteristics. After consideration our educated guess was that the bandwidth-delay product effect is throttling the TCP bandwidth of each connection.

Fully-controlled environment is a good starting-point for experiments

In order to have full control of the end-nodes and the network we decided to deploy a small testbed inside our premises between hosts located to GRNET's distant DCs (200 miles away).

Production environment for the ultimate trial

Real-life conditions which define the environment we made the measurements/observations:

1) Use of perfSONAR monitoring suite for measurements [https://www.perfsonar.net].

2) Use of ESnet servers' acting as the other end-host, outside our DC.

3) The intermediate links were not congested as they were belonging to overprovisioned research networks such as GEANT and ESnet.

4) We performed multiple bandwidth tests with single TCP flow between a server located to GRNET DC facilities and ESnet perfSonar servers to 3 different locations (CERN, London, New York) and an intra-DC test.

5) For each value of buffer tune set we ran iterations with 1500 and 9000 bytes MTU (Maximum transmission unit) size to record the differences.

Conclusions

We reached the following experimental conclusions:

1) The read buffer size of the fat flow receiver has the highest impact to tcp flow performance as it defines the TCP window size and should be tuned taking into consideration the Round-Trip-Time (RTT).

2) The write buffer size of the fat flow sender must also be tuned consecutively, as it defines the maximum number of UN-acknowledged bytes that the sender side will allow to fly on wire.

3) For very large RTTs, over 100ms, or/and especially for 10/40/100Gbit links you need huge buffers, even 30x or 40x bigger than the default kernel values.

4) TCP read buffer max_value defines the maximum window size of the receiver throttling the maximum throughput of the TCP connection over the path and the TCP send buffer max_value defines the maximum number of bytes on flight.

5) MTU size affects the bandwidth performance of a single TCP flow when this TCP flow consumes all the available bandwidth and the window size is not a bottleneck.

6) MTU is also beneficial since it requires a smaller number of packets to be generated for the same amount of data, requiring less packet processing to the end-hosts. Another advantage, not investigated in our test, is that with jumbo frames you can get faster recovery rate from potential packet loss events.

7) Kernel's window-size auto-tuning, with unchanged read buffer default_value, is importing delays to the desired high bitrate build-up in large RTT TCP connections.

Short CV

Dr. Christos Argyropoulos received the Diploma in Electrical Engineering and Computer Science from the University of Patras and the Ph.D. degree in Electrical Engineering from the National Technical University of Athens (NTUA). He worked as a research associate to the Network Management and Optimal Design Laboratory (NETMODE) at NTUA participating in several European FP7 projects (NOVI, GÉANT GN3 and GN3+ etc.). His main research interests lie in the area of computer networks with emphasis on network virtualization and software defined networking. Currently he is with the Greek Research and Education Network (GRNET).