                 Considerations for Bandwidth Aggregation
                   draft-mrc-banana-considerations-01

Abstract

   This document lists a number of architectural and technical topics
   that should be considered in the design and implementation of
   Bandwidth Agregation mechanisms.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on January 9, 2017.

Copyright Notice

Table of Contents

1. Introduction

   There are currently several bandwidth aggregation solutions being
   discussed within the IETF or other parts of the Internet industry.
   This document discusses a number of technical and architectural facts
   that should be considered in the design and implementation of those
   solutions. This document is intended to provide useful information
   to the community, not to state requirements or advocate for a
   particular solution.

There is one simple thought underlying many of the considerations in this document: the goals of bandwidth aggregation are to increase the effective bandwidth available to customers and improve the reliability of customers' Internet access by using all of the available links, not just one of them. Intuitively, two links should have more bandwidth and reliability than one link, but experience shows that it is actually quite hard to design a bandwidth aggregation solution that will achieve the desired goals in all cases, and quite easy to design a solution that will reduce the effective bandwidth or decrease the reliability of Internet access in an unacceptably high number of cases. Many of the considerations in this document are intended to point out why that happens, so that solutions and implementations can avoid known pitfalls in this area.

[Note: This document is a work in progress. Feedback on the existing content is welcome, as well as feeback on other considerations that should be included. Please send any feedback to the Bandwidth Aggregation mailing list: banana@ietf.org]

2. What is Bandwidth Aggregation?

[TBD]

3. Taxonomy of Solutions

This section attempts to catergorize bandwidth aggregation solutions along several axes, providing a taxonomy tbat we can use to describe and reason about individual solutions. [Note: This section is largely TBD.]

3.1. Tunnel-Based Solutions

Many of the Bandwidth Aggregations currently under discussion are tunnel-based solutions. They tunnel traffic over the links that are being aggregated, and recombine the traffic on the remote end.

[Insert ASCII image of tunnel-based approach.]

There is at least one proposal for Bandwith Aggregation (the MP-TCP-based approach) that does not use tunnels. The considerations for tunnel-based solutions listed below may not apply to non-tunnel-based solutions.

3.2. Per-Packet vs. Per-Flow Multiplexing

The solutions currently under discussion use several different methods to determine which traffic will be sent over which interface.

These methods can be grouped into two categories: per-packet multiplexing and per-flow multiplexing.

Per-packet multiplexing aggregates the bandwidth by sending the desired proportion of packets over each interface. In these solutions, packets from single flow (such as a TCP connection) may be split across multiple interfaces and will need to be recombined at the remote end. However, the ability to multiplex on a per-packet basis makes it possible to most precisely apportion traffic across the available bandwidth.

Per-flow multiplexing involves choosing a single interface for each flow (i.e. TCP connection or application session) and sending all of the packets for a single flow across that interface. In these solutions, the flow do not need to be combined on the remote end. However, the ability to balance traffic between multiple links may be limited if there are only a small number of traffice flows active.

4. Considerations for All Solutions

   This section desribes potential issues that should be considered in the design and implementation of all bandwidth aggregation solutions.

4.1. Link Characteristics and Performance

4.2. Bypass Traffic

4.3. Capped or Tariffed Interfaces

   In some cases, bandwith aggregation may be performed between dedicated links and links that have traffic caps or tarifs associated with additional use. In these cases, customer may want to use bandwidth aggregation to increase the performance of some applications, while other applications (e.g. firmware upgrades or content downloads) may be limited to using the dedicated link. Solutions that wish to support this capability will need to support having a set of traffic that will be distributed using the bandwidth aggregation algorithms, and a set of traffic that will not.

4.4. Learning from History (Multilink PPP)

   The IETF has a venerable, standard, implemented solution to this sort of problem: Multilink PPP. Unfortunately, it is commonly said that experience with Multilink PPP did not find that it increased the effective bandwidth when it was used to share two indentical ISDN lines, compared to the bandwidth that was achieved from using only one line...

[Note: We should attempt to determine if this is true and, if so, find any research papers or other documentation that might help us understand why this was true, so that we might learn from history.]

5. Considerations for Tunnel-Based Solutions

5.1. Tunnel Overhead

   Tunneling involves more overhead than sending non-tunnelled traffic for two reasons: the extra IP and tunnel headers that must be included in each packet, and any tunnel management traffic that must be exchanged. This means that, in order to achieve increased effective bandwidth by aggregating traffic across more than one link, the raw bandwidth across multiple links must be higher than the bandwidth on a single link by a large enough margin to compensate for the tunnel overhead, so that increased effective bandwidth will result.

5.2. MTU Issues

   There are a number of MTU Issues associated with all tunneling mechanisms, and there is a different set of MTU issues associated with any mechanism that changes the MTU of packets within a given flow.

   [Note: This section is TBD.]

5.2.1. Fragmentation Issues

5.2.2. Issues with MTU Changes

6. Considerations for Per-Packet Solutions

6.1. Packet Ordering

6.2. Transport Layer Algorithms

   There are transport layer congestion control algorithms implemented in every TCP/IP stack. It is the purpose of these algorithms to ramp up the speed of a TCP connection slowly, and to back off at the first sign of congestion (i.e. packet loss). There are also algorithms which are designed to detect packet loss as quickly as possible by analyzing the protocol round-trip times, and deciding that a packet has been lost whenever there is a longer delay than expected before an acknowledgement is received. Per-packet solutions run the risk of interacting pathologically with these algorithms.

For example, if traffic from a single flow is being demultiplexed across two links with significantly different round-trip times (i.e. differnt latencies), the TCP retransmission algorithms may be triggered for packets that traverse the higher latency link. This may cause the TCP congestion control algorithms to inaccurately detect congestion (even when neither link is congested) and slow down the speed of the TCP connetion. In these cases, the throughput of each TCP connection may be reduced, thus reducing the performance of a customer's applications to the point where their applicaitons would have run faster over a single link.

This problem can potentially be avoided by avoiding aggregation of links with significantly different latencies. However, it may be desirable to perform bandwidth aggregation across those links in some cases.

7. Considerations for Per-Flow Solutions

This section describes some potential issues that should be considered in the design of per-flow bandwidth aggregation solutions.

7.1. Granularity Issues

Per-Flow demultiplexing is in widespread use for traffic engineering and load balancing in carrier and corporate networks. Within those networks, there are a very large number of flows, so being able to direct traffic on a per-flow basis will generally be sufficient to achieve acceptable load-balancing or link aggregation.

However, the number of flows generated by a single home or small office might not provide sufficient granularity to achieve the desire level of bandwidth aggregation. Also some flows, such as streaming video flows, might use far more bandwidth than other, such as downloading a single image on a web page. It is not always possible to predict which flows will be high-bandwidth flows, and which will require less bandwidth.

7.2. Aggregated Flows

Some Internet flows are aggregated into single, larger flows at the end-nodes. This would include VPN traffic that is tunnelled to a corporate intranet, or other tunnelled traffic such as Teredo traffic for IPv6. Use of these mechanisms can prevent proper classification of traffic into separate flows, thus exacerbating the granularity issues described above.

7.3. Encrypted Traffic

In some cases such as secure VPN traffic, the contents of packets may
be encrypted in a way that does not allow a middlebox to see the
transport-layer flow inforamtion (such as TCP or UDP ports). In
these cases, it might not be possible to properly separate multiple
flows between a single set of endpoints. This can exacerbate the
granularity issues described above.

8. Practical Considerations

8.1. Use Available Information

In many of the environments in which these mechanisms will be
deployed, there is already considerable informaiton available about
link quality, lost packets, traffic loads and effective bandwidth.
It is possible to use that information to actively tune a bandwidth
aggregation solution to achieve optimal effective bandwidth. This
information can also be used to detect situations in which the link
quality of a secondary link is not sufficient to provide enough
additional bandwidth to compensate for the bandwidth aggregation
overhead.

8.2. Theory is No Substitute for Experience

Because of the complexity of the algorithms implemented at multiple
layers of the TCP/IP Stack, many things that would appear to work in
theory or in limited simulation do not have the expected results when
deployed in a real-world environment. Therefore, it would be highly
desirable to have real-world experience running a bandwidth
aggregation mechanism in an operational network before standardizing
it within the IETF.

9. Security Considerations

9.1. Binding Tunnel Endpoints

10. Appendix A: List of Solutions

This is a (possibly incomplete) list of current or proposed solutions
for Bandwidth Aggregation. The descriptions in this section (when
present) were provided by the proponents of each solution. This list
is provided only as a source of information about possible solutions,
not as a recommendation for or against any of these solutions.

[Note: Insert information from Google Doc in this section.]

11. Informative References

   [RFC6126] Chroboczek, J., "The Babel Routing Protocol", RFC 6126,
              DOI 10.17487/RFC6126, April 2011,
              <http://www.rfc-editor.org/info/rfc6126>.

Authors' Addresses

   Margaret Cullen
   Painless Security
   14 Summer Street, Suite 202
   Malden, MA 02148
   USA

   Phone: +1 781 405-7464
   Email: margaret@painless-security.com
   URI: http://www.painless-security.com

   Mingui Zhang
   Huawei Technologies

   Email: zhangmingui@huawei.com