TSVWG   Y. Li
Internet-Draft   X. Zhou
Intended status: Informational   Huawei
Expires: January 14, 2021   M. Boucadair
                                                                  Orange
                                                                 J. Wang
                                                           China Telecom
                                                                  F. Qin
                                                            China Mobile
                                                           July 13, 2020

     **LOOPS (Localized Optimizations on Path Segments) Problem Statement and
        Opportunities for Network-Assisted Performance Enhancement
               draft-li-tsvwg-loops-problem-opportunities-06**

Abstract

   In various network deployments, end to end forwarding paths are
   partitioned into multiple segments. For example, in some cloud-based
   WAN communications, stitching multiple overlay tunnels are used for
   traffic policy enforcement matters such as to optimize traffic
   distribution or to select paths exposing a lower latency. Likewise,
   in satellite communications, the communication path is decomposed
   into two terrestrial segments and a satellite segment. Such long-
   haul paths are naturally composed of multiple network segments with
   various encapsulation schemes. Packet loss may show different
   characteristics on different segments.

   Traditional transport protocols (e.g., TCP) respond to packet loss
   slowly especially in long-haul networks: they either wait for some
   signal from the receiver to indicate a loss and then retransmit from
   the sender or rely on sender's timeout which is often quite long.
   With the increase of end-to-end transport encryption (e.g., QUIC),
   traditional PEP (performance enhancing proxy) techniques such as TCP
   splitting are no longer applicable.

   LOOPS (Local Optimizations on Path Segments) is a network-assisted
   performance enhancement over path segment and it aims to provide
   local in-network recovery to achieve better data delivery by making
   packet loss recovery faster. In an overlay network scenario, LOOPS
   can be performed over a variety of the existing, or purposely
   created, tunnel-based path segments.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

1.1. The Problem and Opportunity Overview

   Packet loss is ubiquitous in Internet. A reliable transport layer
   normally employs some end-to-end retransmission mechanisms which also
   address congestion control [RFC0793] [RFC5681]. The sender either
   waits for the receiver to send some signals on a packet loss or sets
   some form of timeout for retransmission. For unreliable transport
   protocols such as RTP [RFC3550], optional and limited usage of end-
   to-end retransmission is employed to recover from packet loss
   [RFC4585] [RFC4588]. End-to-end retransmission to recover lost
   packets is slow especially when the network is long-haul. For short-
   lived flows and transactional flows, latency suffers a lot from tail
   loss.

   Tunnels are widely deployed within many networks to achieve various
   engineering goals, including long-haul WAN interconnection or
   enterprise wireless access networks. A connection between two
   endpoints can be decomposed into many connection legs. As such, the
   corresponding forwarding path can be partitioned into multiple path
   segments that some of them are using network overlays by means of
   tunnels. This design serves a number of purposes such as steering
   the traffic, optimizing egress/ingress link utilization, optimizing
   traffic performance metrics (such as delay, delay variation, or
   loss), optimizing resource utilization by invoking resource bonding,
   provide high-availability, etc.

   When a path is partitioned into multiple path segments that are
   realized typically as overlay tunnels, LOOPS (Local Optimizations on
   Path Segments) aims to provide in-network recovery over segments to
   achieve better data delivery by making packet loss recovery faster.
   In an overlay network scenario, LOOPS can be performed over the
   existing, or purposely created, overlay tunnel based path segments.
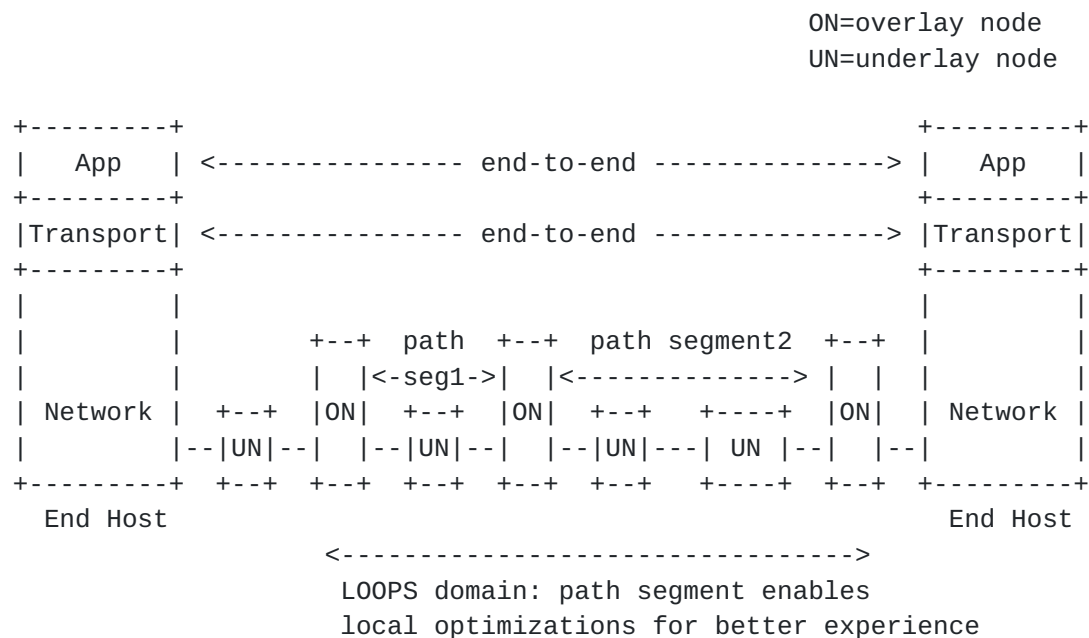   Figure 1 show an overall usage scenarios of LOOPS.

```
                                              ON=overlay node
                                              UN=underlay node


   +---------+                                          +---------+
   |   App   | <--------------- end-to-end --------------> |   App   |
   +---------+                                          +---------+
   |Transport| <--------------- end-to-end --------------> |Transport|
   +---------+                                          +---------+
   |         |                                          |         |
   |         |         +--+  path  +--+  path segment2  +--+  |         |
   |         |         |  |<-seg1->|  |<-------------> |  |  |         |
   | Network |   +--+  |ON|  +--+  |ON|  +--+   +----+  |ON|  | Network |
   |         |--|UN|--|  |--|UN|--|  |--|UN|---| UN |--|  |--|  |         |
   +---------+  +--+  +--+  +--+  +--+  +--+   +----+  +--+  +---------+
      End Host                                            End Host
                   <------------------------------->
                   LOOPS domain: path segment enables
                   local optimizations for better experience
```

                      Figure 1: LOOPS Usage Scenario

1.2. Sketching a Work Direction: Rationale & Goals

   This document sketches a proposal that is meant to experimentally
   investigate to what extent a network-assisted approach can contribute
   to increase the overall perceived quality of experience in specific
   situations (e.g., Sections 3.5 and 3.6 of [RFC8517]) without
   requiring access to internal transport primitives. The rationale
   beneath this approach is that some information (loss detection and
   segment characteristics, etc.) can be used to trigger local in-
   network recovery actions which have a faster effect while not
   impacting the end-to-end congestion control loop.

   To that aim, the work is structured into two (2) phased stages:

   o Stage 1: Network-assisted optimization. This one assumes that
      optimizations can be implemented at the network without requiring
      defining new interaction with the endpoint. Existing tools such
      as ECN will be used. Loss signal would be converted to CE
      (congestion experienced) signal to interact with the end-to-end
      control loop.

   o Stage 2: Collaborative networking optimization. This one requires
      more interaction between the network and an endpoint to implement
      coordinated and more surgical network-assisted optimizations based
      on information/instructions shared by an endpoint or sharing
      locally-visible information with endpoint for better and faster
      recovery.

The document focuses on the first stage. Effort related to the
second stage is out of scope of the initial planned work.

The proposed mechanism is not meant to be applied to all traffic, but
only to a subset which is particularly benefits from, and has been
selected for the network-assisted optimization service.

Which traffic is selected is deployment-specific and policy-based.
For example, techniques for dynamic information about optimization
function (e.g., SFC) may be leveraged to unambiguously identify the
aggregate of traffic that is eligible to the service. Such
identification may be triggered by subscription actions made by
customers or be provided by a network provider (e.g., specific
applications, during specific events such as during severe DDoS
attack or flash crowds events).

Likewise, whether the optimization function is permanently
instantiated or on-demand is deployment-specific.

This document does not intend to provide a comprehensive list of
target deployment cases. Sample scenarios are described to
illustrate some LOOPS potentials. Similar issues and optimizations
may be helpful in other deployments such as enhancing the reliability
of data transfer when a fleet of drones are used for specific
missions (e.g., site inspection, live streaming, and emergency
service). Captured data should be reliably transmitted via paths
involving radio connections.

It is not required that all segments are LOOPS-aware to benefit from
LOOPS advantages.

Section 3 presents the issues and opportunities found in some
multiple path segments scenarios. Section 3 describes the impact of
packet loss for different traffic. Section 5 describes the LOOPS
desired features and their impact on existing network technologies.
Section 6 shows the analysis on local retransmission and end-to-end
retransmission. Section 7 summarizes LOOPS key elements.

2. Terminology

This document makes use of the following terms:

LOOPS: Local Optimizations on Path Segments. LOOPS includes the
   local in-network (i.e., non end-to-end) recovery functions and
   other supporting features such as local measurement, loss
   detection, and congestion feedback.

LOOPS Node: A node supporting LOOPS functions.

Overlay Node (ON): A node having overlay functions (e.g., overlay
   protocol encapsulation/decapsulation, header modification, TLV
   inspection) and LOOPS functions in the LOOPS overlay network usage
   scenario.

Overlay Tunnel: A tunnel with designated ingress and egress nodes
   using some network overlay protocol as encapsulation, optionally
   with a specific traffic type.

Path segment: A LOOPS enabled tunnel-based network subpath. It is
   used interchangeably with overlay segment in this document when
   the context wants to emphasize on its overlay encapsulated nature.
   It is also called segment for simplicity in this document.

Overlay segment: Refers to path segment.

Underlay Node (UN): A node not participating in the overlay network.

3. Usage Scenarios

3.1. Cloud-Internet Overlay Network

CSPs (Cloud Service Providers) are connecting their data centers
using the Internet or via self-constructed networks/links. This
expands the traditional Internet's infrastructure and, together with
the original ISP's infrastructure, forms the Internet underlay.

Automation techniques and NFV (Network Function Virtualization) make
it easier to dynamically provision a new virtual node/function as a
workload in a cloud for CPU/storage intensive functions. Virtual
nodes can be in form of virtual machines or containers hosting the
workloads sharing a physical node's infrastructure. With the aid of
various mechanisms such as kernel bypassing and Virtual IO,
forwarding based on virtual nodes is becoming more and more
effective. The interconnection among the purposely positioned
virtual nodes and/or the existing nodes with virtualization functions
potentially form an overlay infrastructure. It is called the Cloud-
Internet Overlay Network (CION) in this document for short.

This architecture scenario makes use of overlay technologies to
direct the traffic going through the specific overlay path in order
to achieve better service delivery. It purposely creates or selects
overlay nodes (ON) from providers. By continuously measuring the
delay of path segments and use them as metrics for path selection,
when the number of overlay nodes is sufficiently large, there is a
high chance that a better path could be found
[DOI_10.1109_ICDCS.2016.49] [DOI_10.1145_3038912.3052560].
[DOI_10.1145_3038912.3052560] further shows all cloud providers

experience random loss episodes and random loss accounts for more
than 35% of total loss.

Figure 2 shows an example of an overlay path over large geographic
distances. An overlay node (ON) is usually a virtual node, though it
does not have to be. Three path segments, i.e., ON1-ON2, ON2-ON3,
ON3-ON4 are shown. Each segment transmits packets using some form of
network overlay protocol encapsulation. ON has the computing and
memory resources that can be used for some functions like packet loss
detection, network measurement and feedback, packet retransmission
and FEC (Forward Error Correction) computation. ONs here are managed
by a single administrator though they can be workloads created from
different CSPs.

```
                      _____
                     /   domain 1    \
                   /                    \
                 ___/           ------------\
               /                             \
    PoP1 ->--ON1                              \
          |   |                      ON4------>-- PoP2
          |   |    ON2                   ___|__/
          \__|_ |->|          _____      /   |
           | \|__|__       /       \    /    |
           |  |  |  \____/        \__/     |
          \|/ |  |           _____          |
           |  |  |        __/      \         |
           |  | \|/  /               \____     |
           |  |  |  /         domain 2 \ /|\
           |  |  | |           ON3        |  |
           |  |  |  \          |->|       |  |
           |  |  |   \____|__|_____/   |
           | /|\ |           | \|/         |
           |  |  |           |  |          |
           |  |  |          /|\ |          |
      +-----------------------------------------------+
      |     |  |  |         |  |           |  Internet |
      |     o--o  o---o->---o  o---o->--o--o  underlay |
      +-----------------------------------------------+
```
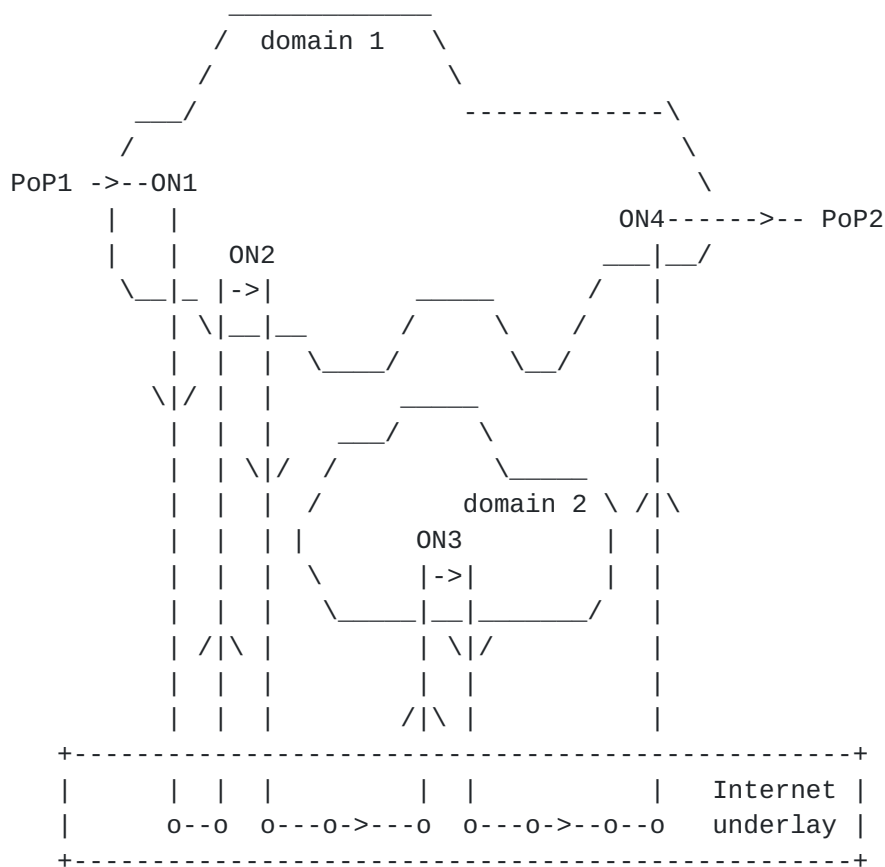
                 Figure 2: Cloud-Internet Overlay Network (CION)

We tested based on 37 overlay nodes from multiple cloud providers
globally. Each pair of the overlay nodes are used as sender and
receiver. When the traffic is not intentionally directed to go
through any intermediate virtual nodes, we call the path followed by
the traffic in the test the default path. When any of the virtual
nodes is intentionally used as an intermediate node to forward the

traffic, the path that the traffic takes is called an overlay path.
The preliminary experiments showed that the delay of a specifically
selected overlay path has lower latency than the one of the default
path in 69% of cases at 99% percentile and improvement is 17.5% at
99% percentile when we probe Ping packets every second for a week.
The average number of hops for an overlay path is 3.02. More
experimental information can be found in
[DOI_10.1109_INFCOMW.2019.8845208].

Lower average delay does not necessarily mean less or no packet loss.
Different path segments have different packet loss rates. Loss rate
is another major factor impacting the user experience, espcially for
the short-lived or transactional flows. From some customer
requirements, the target loss rate is set in the test to be less than
1% at 99% percentile and 99.9% percentile, respectively. The loss
was measured between any two overlay nodes, i.e., any potential path
segment. Two thousand Ping packets were sent every 20 seconds
between two overlay nodes for 55 hours. This preliminary experiment
showed that the packet loss rate satisfaction are only 44.27% and
29.51% at the 99% and 99.9% percentiles, respectively.

As CION naturally consists of multiple overlay segments, LOOPS can
leverage this to perform local optimizations on a single hop between
two overlay nodes. ("Local" here is a concept relative to end-to-
end, it does not mean such optimization is limited to LAN networks.)

3.2. Satellite Communication

Traditionally, satellite communications deploy PEP (performance
enhancing proxy [RFC3135]) nodes around the satellite link to enhance
end-to-end performance. TCP splitting is a common approach employed
by such PEPs, where the TCP connection is split into three: the
segment before the satellite hop, the satellite section (uplink,
downlink), and the segment behind the satellite hop. This requires
heavy interactions with the end-to-end transport protocols, usually
without the explicit consent of the end hosts. Unfortunately, this
is indistinguishable from a man-in-the-middle attack on TCP. With
end-to-end encryption moving under the transport (QUIC), this
approach is no longer useful.

Geosynchronous Earth Orbit (GEO) satellites have a one-way delay (up
to the satellite and back) on the order of 250 milliseconds. This
does not include queueing, coding and other delays in the satellite
ground equipment. The Round Trip Time for a TCP or QUIC connection
going over a satellite hop in both directions, in the best case, will
be on the order of 600 milliseconds. And, it may be considerably
longer. RTTs on this order of magnitude have significant performance
implications.

Packet loss recovery is an area where splitting the TCP connection
into different parts helps. Packets lost on the terrestrial links
can be recovered at terrestrial latencies. Packet loss on the
satellite link can be recovered more quickly by an optimized
satellite protocol between the PEPs and/or link layer FEC than they
could be end to end. Again, encryption makes TCP splitting no longer
applicable. Enhanced error recovery at the satellite link layer
helps for the loss on the satellite link but doesn't help for the
terrestrial links. Even when the terrestrial segments are short, any
loss must be recovered across the satellite link delay. And, there
are cases when a satellite ground station connects to the general
Internet with a potentially larger terrestrial segment (e.g., to a
correspondent host in another country). Faster recovery over such
long terrestrial segments is desirable.

There are two high level classes of solutions for making encrypted
transport traffic like QUIC work well over satellite:

o Hooks in the transport protocol which can adapt to large BDPs
   where both the bandwidth and the latency are large. This would
   require end to end enhancement.

o Capabilities (such as LOOPS) under the transport protocol to
   improve performance over specific segments of the path. In
   particular, separating the terrestrial from the satellite losses.
   Fixing the terrestrial loss quickly.

This document focuses on the latter.

3.3. Branch Office WAN Connection

Enterprises usually require network connections between the branch
offices, or between branch office and cloud data center over
geographic distances. With the increasing deployment of vCPE
(virtual CPE), services hosted on the CPE are moved to the provider
network from the customer site. Such vCPE approach enables some
value added service to be provided such as WAN optimization and
traffic steering.

Figure 3 shows a branch office access to public cloud via a selected
PoP (point of presence) for service access or reaching another branch
office via vPC (Virtual Private Cloud) interconnect. vCPE connects to
the PoP which can be hundreds of kilometers away via Internet. From
vCPE1 to vCPE2, it can consist of three segments, vCPE1-PoP1,
PoP1-PoP2 and PoP2-vCPE2. Packet loss can happen on any of them.
Segment based in-network recovery can be employed here to improve the
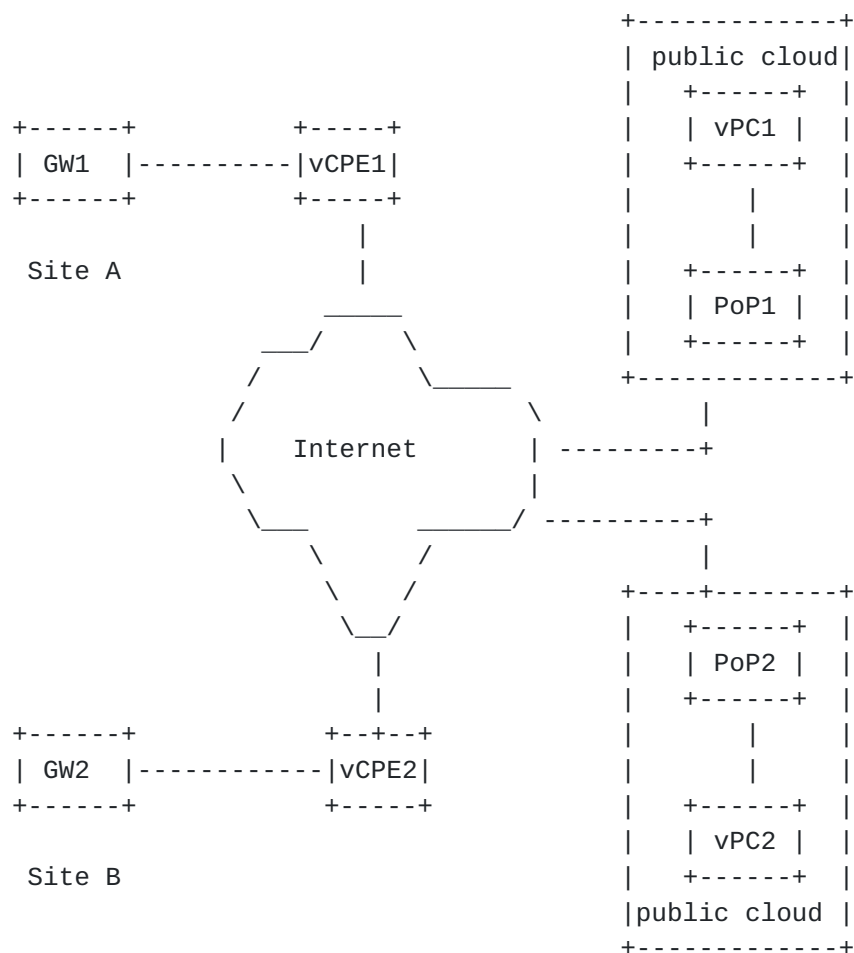WAN connection quality.

```
                                        +-------------+
                                        | public cloud|
                                        |    +------+  |
   +------+            +-----+          |    | vPC1 |  |
   | GW1  |-----------|vCPE1|           |    +------+  |
   +------+            +-----+          |       |      |
                          |             |       |      |
   Site A                 |             |    +------+  |
                        ____           |    | PoP1 |  |
                     ___/    \         |    +------+  |
                    /         \____    +-------------+
                   /               \         |
                  |    Internet     | ---------+
                   \               |
                    \___       ____/  ---------+
                     \   \    /               |
                      \   \  /          +----+---------+
                       \__/             |    +------+  |
                         |              |    | PoP2 |  |
                         |              |    +------+  |
   +------+           +--+--+           |       |      |
   | GW2  |------------|vCPE2|          |       |      |
   +------+           +-----+           |    +------+  |
                                        |    | vPC2 |  |
   Site B                               |    +------+  |
                                        |public cloud |
                                        +-------------+
```

Figure 3: Enterprise Cloud Access

4. Impact of Packet loss

4.1. Tail Loss or Loss in Short Flows

   When the lost segments are at the end of a transaction, TCP's fast
   retransmit algorithm does not work as there are no ACKs to trigger
   it. When a sender does not receive an ACK for a given segment within
   a certain amount of time called retransmission timeout (RTO), it re-
   sends the segment [RFC6298]. RTO can be as long as several seconds.
   Hence the recovery of lost segments triggered by RTO is lengthy.
   [I-D.dukkipati-tcpm-tcp-loss-probe] indicates that large RTOs make a
   significant contribution to the long tail on the latency statistics
   of short flows such as loading web pages.

   The short-lived flows often complete in one or two RTTs. Even when
   the lost packet is not an exact tail, it can possibly add another RTT

because there may not be enough packets in flight to trigger the fast
retransmit). In long-haul networks, it can result in extra time of
tens or hundreds of milliseconds. For ant short lived or
transactional flows, it affects the latency greatly.

An overlay segment transmits the aggregated flows from ON to ON. As
short-lived flows are aggregated, the probability of tail loss over
this specific overlay segment decreases compared to an individual
flow. The overlay segment is much shorter than the end-to-end path,
hence loss recovery over an overlay segment helps to obtain low
latency.

4.2. Packet Loss in Real Time Media Streams

The Real-time transport protocol (RTP) is widely used in interactive
audio and video. Packet loss degrades the quality of the received
media. When the latency tolerance of the application is sufficiently
large, the RTP sender may use RTCP NACK feedback from the receiver
[RFC4585] to trigger the retransmission of the lost packets before
the playout time is reached at the receiver.

The end-to-end path over WAN can be hundreds of milliseconds, so the
end-to-end feedback based retransmission may be not be very useful
when applications can not tolerate one more RTT. Loss recovery over
an overlay segment can then be used for the scenarios in which a
shorter delayed retransmission can catch up with the playout time.

5. Features to be Considered for LOOPS

This section provides an overview of the LOOPS features. This
section is not meant to document a detailed specification, but it is
meant to highlight some design choices that may be followed during
the solution design phase.

5.1. Local Recovery

LOOPS (Local Optimizations on Path Segments) aims to provide in-
network recovery over segments to achieve better data delivery by
making packet loss recovery faster. This is viable because LOOPS
nodes will be instantiated to partition the path into segments. At
the same time, LOOPS does not replace the end-to-end loss recovery
(if any). With the advent of automation and technologies like NFV
and virtual IO, it is possible to dynamically instantiate functions
to nodes. The enabling of LOOPS is expected to be dynamic. When to
enable this function is out of scope. The operator or administrator
can make the decision based on their historical experience or real-
time monitoring.

There are two ways to recover packet, retransmission and Forward
Error Correction (FEC). A document to specify the generic elements
for loss detection, sequence number space, acknowledgment generation
and state transition is available in [I-D.welzl-loops-gen-info].

5.2. Congestion Control Interaction

When a TCP-like transport layer protocol is used, local recovery in
LOOPS has to interact with the upper layer transport congestion
control. Classic TCP adjusts the congestion window when a loss is
detected and then fast retransmit is invoked. LOOPS performs in-
network recovery which may cause a loss event not being observed by
the TCP sender. Then TCP sender may overshoot then.

To solve this issue, LOOPS needs to report the loss to end-to-end
congestion control LOOPS. LOOPS can CE(Congestion Experienced) marks
its recovered packets as the loss signal to end-to-end. Converting a
packet loss signal to CE marking signal brings the benefits of
reducing Head-of-Line blocking and probability of RTO expiry
[RFC8087] without affecting TCP sender's loss based congestion
control behaviour while enjoying the faster local recovery. ECN
based indication is equivalent to a loss event at the TCP sender
[RFC3168]. In this way, a requirement is set for applying LOOPS.
Only ECT (ECN-Capable Transport) flows should be directed to an LOOPS
enabled path segment.

5.3. Overlay Protocol Extensions

Some tunnel protocols such as VXLAN [RFC7348], GENEVE
[I-D.ietf-nvo3-geneve], LISP [RFC6830] or CAPWAP [RFC5415] are
employed in overlay network. They are used in various ways. A path
can have single overlay tunnel as a sub-path or stitch multiple
segments together, like VXLAN [RFC7348] or GENEVE
[I-D.ietf-nvo3-geneve], or specify a sequence of intermediate nodes,
as in SRv6 [RFC8754].

LOOPS does not look into the inner packet. LOOPS information is
required to be embedded in the overlay protocol header. An example
shown in Figure 4. The current protocol focus is GENEVE
[I-D.ietf-nvo3-geneve]. The specific information is to be defined in
separate documents.

```
 +------------+-----------+-----------------+---------+---------+
 |Outer IP hdr|Overlay hdr|LOOPS information|Inner hdr|payload  |
 +------------+-----------+-----------------+---------+---------+
```

Figure 4: LOOPS Extension Header Example

6. **Local in-network Recovery and End-to-end Retransmission**

   Most transport layer protocols have their own end-to-end
   retransmission to recover the lost packet. When LOOPS is in use, its
   local recovery can affect the end-to-end one. This section talks
   about such impacts.

   There are two ways to perform local recovery, retransmission and FEC
   (Forward Error Correction). They are possibly used together in some
   cases. Such approaches between two overlay nodes recover the lost
   packet in relatively shorter distance and thus shorter latency.
   Therefore the local recovery is generally faster compared to end-to-
   end.

   End-to-end retransmission is normally triggered by a NACK as in RTCP,
   multiple duplicate ACKs as in traditional TCP or time based detection
   as in RACK [I-D.ietf-tcpm-rack].

   When FEC is used for local recovery, it may come with a buffer to
   make sure the recovered packets delivered are in order subsequently.
   Therefore the receiver side is unlikely to see the out-of-order
   packets and then send a NACK or multiple duplicate ACKs. The side
   effect to unnecessarily trigger end-to-end retransmit is minimum.
   When FEC is used in this way, if redundancy and block size are
   determined, extra latency required to recover lost packets is also
   bounded. Then RTT variation caused by it is predictable. In some
   extreme case like a large number of packet loss caused by persistent
   burst, FEC may not be able to recover it. Then end-to-end retransmit
   will work as a last resort. In summary, when FEC is used as local
   recovery, the impact on end-to-end retransmission is limited.

   When local retransmission is used, it has the following impacts on
   the end-to-end retransmission.

   For packet loss in RTP streaming, local retransmission can recover
   those packets which would not be retransmitted end-to-end otherwise
   due to long RTT. Therefore when the segment(s) being retransmitted
   on is a small portion of the whole end to end path, the
   retransmission will have a significant effect of improving the
   quality at receiver.

   When the sender also re-transmits the packet based on a NACK
   received, the receiver may receive the duplicated retransmitted
   packets.

   For packet loss in TCP flows, TCP RENO and CUBIC use duplicate ACKs
   as a loss signal to trigger the fast retransmit. Though we are not

standardize the buffering feature of a LOOPS egress, an introductory
analysis is given as follows.

o The egress overlay node can buffer the out-of-order packets for a
   while, giving a limited time for a packet being retransmitted
   somewhere in the overlay path to reach it. The retransmitted
   packet and the buffered packets caused by it may increase the RTT
   variation at the sender. When the retransmitted latency is a
   small portion of RTT or the loss is rare, such RTT variation will
   be smoothed without much impact.

   The buffer management is nontrivial in this case. It has to be
   determined how many out-of-order packets can be buffered at the
   egress overlay node before it gives up waiting for a successful
   local retransmission. In some extreme case the lost packet is not
   recovered successfully locally, the sender may invoke end-to-end
   fast retransmit slower than it would be in classic TCP.

o If LOOPS network does not buffer the out-of-order packets caused
   by packet loss, TCP sender which uses a time based loss detection
   like RACK [I-D.ietf-tcpm-rack] will perform well here. It uses
   the notion of time to replace the conventional DUPACK threshold
   approach to detect losses. Hence it prevents the TCP sender from
   invoking fast retransmit too early. Local retransmission will not
   interfere the sender's retransmission generally in this case. If
   time based loss detection is not supported at the sender, end to
   end retransmission may be invoked as usual. It consumes extra
   bandwidth Because the lost packets (i.e. recovered packet) is
   normally a very small percentage of the total packets. Then extra
   bandwidth cost is not significant.

7. Summary

   LOOPS will extend the existing overlay protocols in data plane,
   potential starting from GENEVE [I-D.ietf-nvo3-geneve] which has good
   extensibility. Path or segment selection can be feature provided by
   the overlay protocols via SDN techniques [RFC7149] or other
   approaches and is not a part of LOOPS. LOOPS is a set of functions
   to be implemented on Overlay Nodes as a tunnel transport with best
   effort reliability. LOOPS targets the following features.

1. Local recovery: Local recovery: Retransmission, FEC, or
    combination thereof can be used as local recovery method. Such
    recovery mechanism is in-network. It is performed by two network
    nodes with computing and memory resources.

   2. Local measurement: Ingress/Egress overlay nodes measure the local
       segment RTT, loss and/or throughput to immediately get the
       overlay segment status for loss detection.

   3. Interact with end-to-end congestion control: Convert a packet
       loss signal to an ECN-marking signal to notify the end host
       sender.

8. Security Considerations

   LOOPS does not require access to the traffic payload in clear, so
   encrypted payload does not affect functionality of LOOPS.

   The use of LOOPS introduces some issues which impact security. ON
   with LOOPS function represents a point in the network where the
   traffic can be potentially manipulated and intercepted by malicious
   nodes. Means to ensure that only legitimate nodes are involved
   should be considered.

   Denial of service attack can be launched from an ON. A rogue ON
   might be able to spoof packets as if it come from a legitimate ON.
   It may also modify the ECN CE marking in packets to influence the
   sender's rate. In order to protected from such attacks, the overlay
   protocol itself should have some built-in security protection which
   is used by LOOPS. The operator should use some authentication
   mechanism to make sure ONs are valid and non-compromised.

9. IANA Considerations

   No IANA action is required.

10. Acknowledgements

   Thanks to etosat mailing list about the discussion about the SatCom
   and LOOPS use case.

11. Informative References

   [RFC0793] Postel, J., "Transmission Control Protocol", STD 7,
              RFC 793, DOI 10.17487/RFC0793, September 1981,
              <https://www.rfc-editor.org/info/rfc793>.

   [RFC3135] Border, J., Kojo, M., Griner, J., Montenegro, G., and Z.
              Shelby, "Performance Enhancing Proxies Intended to
              Mitigate Link-Related Degradations", RFC 3135,
              DOI 10.17487/RFC3135, June 2001,
              <https://www.rfc-editor.org/info/rfc3135>.

   [RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition
              of Explicit Congestion Notification (ECN) to IP",
              RFC 3168, DOI 10.17487/RFC3168, September 2001,
              <https://www.rfc-editor.org/info/rfc3168>.

   [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V.
              Jacobson, "RTP: A Transport Protocol for Real-Time
              Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550,
              July 2003, <https://www.rfc-editor.org/info/rfc3550>.

   [RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey,
              "Extended RTP Profile for Real-time Transport Control
              Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585,
              DOI 10.17487/RFC4585, July 2006,
              <https://www.rfc-editor.org/info/rfc4585>.

   [RFC4588] Rey, J., Leon, D., Miyazaki, A., Varsa, V., and R.
              Hakenberg, "RTP Retransmission Payload Format", RFC 4588,
              DOI 10.17487/RFC4588, July 2006,
              <https://www.rfc-editor.org/info/rfc4588>.

   [RFC5415] Calhoun, P., Ed., Montemurro, M., Ed., and D. Stanley,
              Ed., "Control And Provisioning of Wireless Access Points
              (CAPWAP) Protocol Specification", RFC 5415,
              DOI 10.17487/RFC5415, March 2009,
              <https://www.rfc-editor.org/info/rfc5415>.

   [RFC5681] Allman, M., Paxson, V., and E. Blanton, "TCP Congestion
              Control", RFC 5681, DOI 10.17487/RFC5681, September 2009,
              <https://www.rfc-editor.org/info/rfc5681>.

   [RFC6298] Paxson, V., Allman, M., Chu, J., and M. Sargent,
              "Computing TCP's Retransmission Timer", RFC 6298,
              DOI 10.17487/RFC6298, June 2011,
              <https://www.rfc-editor.org/info/rfc6298>.

   [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The
              Locator/ID Separation Protocol (LISP)", RFC 6830,
              DOI 10.17487/RFC6830, January 2013,
              <https://www.rfc-editor.org/info/rfc6830>.

   [RFC7149] Boucadair, M. and C. Jacquenet, "Software-Defined
              Networking: A Perspective from within a Service Provider
              Environment", RFC 7149, DOI 10.17487/RFC7149, March 2014,
              <https://www.rfc-editor.org/info/rfc7149>.

   [RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger,
              L., Sridhar, T., Bursell, M., and C. Wright, "Virtual
              eXtensible Local Area Network (VXLAN): A Framework for
              Overlaying Virtualized Layer 2 Networks over Layer 3
              Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014,
              <https://www.rfc-editor.org/info/rfc7348>.

   [RFC8087] Fairhurst, G. and M. Welzl, "The Benefits of Using
              Explicit Congestion Notification (ECN)", RFC 8087,
              DOI 10.17487/RFC8087, March 2017,
              <https://www.rfc-editor.org/info/rfc8087>.

   [RFC8517] Dolson, D., Ed., Snellman, J., Boucadair, M., Ed., and C.
              Jacquenet, "An Inventory of Transport-Centric Functions
              Provided by Middleboxes: An Operator Perspective",
              RFC 8517, DOI 10.17487/RFC8517, February 2019,
              <https://www.rfc-editor.org/info/rfc8517>.

   [RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J.,
              Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header
              (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020,
              <https://www.rfc-editor.org/info/rfc8754>.

   [I-D.dukkipati-tcpm-tcp-loss-probe]
              Dukkipati, N., Cardwell, N., Cheng, Y., and M. Mathis,
              "Tail Loss Probe (TLP): An Algorithm for Fast Recovery of
              Tail Losses", draft-dukkipati-tcpm-tcp-loss-probe-01 (work
              in progress), February 2013.

   [I-D.ietf-nvo3-geneve]
              Gross, J., Ganga, I., and T. Sridhar, "Geneve: Generic
              Network Virtualization Encapsulation", draft-ietf-
              nvo3-geneve-16 (work in progress), March 2020.

   [I-D.ietf-tcpm-rack]
              Cheng, Y., Cardwell, N., Dukkipati, N., and P. Jha, "RACK:
              a time-based fast loss detection algorithm for TCP",
              draft-ietf-tcpm-rack-08 (work in progress), March 2020.

   [I-D.welzl-loops-gen-info]
              Welzl, M. and C. Bormann, "LOOPS Generic Information Set",
              draft-welzl-loops-gen-info-03 (work in progress), March
              2020.

   [DOI_10.1109_ICDCS.2016.49]
             Cai, C., Le, F., Sun, X., Xie, G., Jamjoom, H., and R.
             Campbell, "CRONets: Cloud-Routed Overlay Networks", 2016
             IEEE 36th International Conference on Distributed
             Computing Systems (ICDCS), DOI 10.1109/icdcs.2016.49, June
             2016.

   [DOI_10.1145_3038912.3052560]
             Haq, O., Raja, M., and F. Dogar, "Measuring and Improving
             the Reliability of Wide-Area Cloud Paths", Proceedings of
             the 26th International Conference on World Wide Web,
             DOI 10.1145/3038912.3052560, April 2017.

   [DOI_10.1109_INFCOMW.2019.8845208]
             Xu, Z., Ju, R., Gu, L., Wang, W., Li, J., Li, F., and L.
             Han, "Using Overlay Cloud Network to Accelerate Global
             Communications", IEEE INFOCOM 2019 - IEEE Conference on
             Computer Communications Workshops (INFOCOM WKSHPS),
             DOI 10.1109/infcomw.2019.8845208, April 2019.

Authors' Addresses

   Yizhou Li
   Huawei Technologies

   Email: liyizhou@huawei.com

   Xingwang Zhou
   Huawei Technologies

   Email: zhouxingwang@huawei.com

   Mohamed Boucadair
   Orange

   Email: mohamed.boucadair@orange.com

   Jianglong Wang
   China Telecom

   Email: wangjl1.bri@chinatelecom.cn

   Fengwei Qin
   China Mobile

   Email: qinfengwei@chinamobile.com