Export User Flow Telemetry Data by Postcard Packets
draft-song-ippm-postcard-based-telemetry-00

Abstract

   This document describes a proposal which allows network OAM
   applications to collect telemetry data about any user packet. Unlike
   similar techniques such as INT and in-situ OAM, the Postcard-Based
   Telemetry (PBT) does not require inserting telemetry data into user
   packets, but directly exports the telemetry data to a collector
   through separated OAM packets called postcards. Two variations of
   PBT are described in this document: one requires inserting an
   instruction header to user packets to guide the data collection and
   the other only marks the user packets to invoke the data collection.
   Each has its own pros and cons. Either way, the postcards for a
   single user packet are from multiple devices and need to be
   correlated at the collector, which is a unique issue pertaining to
   PBT. Whereas, PBT provides an alternative to INT and address several
   implementation and deployment challenges of the INT-based solutions.

Status of This Memo

Table of Contents

1. Motivation

   In order to gain detailed data plane visibility to support effective
   network OAM, it is important to be able to examine the trace of user
   packets along their forwarding paths. Such path-associated data
   reflect the state and status of each user packet's real-time
   experience and provide valuable information for network monitoring,
   measurement, and diagnosis.

   The telemetry data include the detailed forwarding path, the
   timestamp/latency at each network node, and other information. The

emerging programmable data plane devices allow more sophisticated
data to be retrieved [I-D.song-opsawg-dnp4iq]. Such path-associated
flow data can only be derived from the live user packets. The data
complement with the other data acquired through passive and active
OAM mechanisms such as IPFIX [RFC7011] and ICMP [RFC2925].

In-band Network Telemetry (INT) was designed to cater exactly this
need. in-situ OAM (iOAM) [I-D.brockners-inband-oam-requirements]
represents the related standardization efforts. Essentially, INT
augments user packets with instructions to tell each network node on
the forwarding paths which data to collect. The requested data are
inserted into and travel along with the user packets. The path-end
node strips off the data trace and exports it to a data collector for
processing.

While the concept is simple and straightforward, INT faces several
technical challenges:

o Issue 1: INT header and data processing needs to be done in data
   plane fast path. It may interfere with the normal traffic
   forwarding (e.g., leading to forwarding performance degradation)
   and lead to inaccurate measurements (e.g., resulting in longer
   latency measurements than usual). This undesirable "observer
   effect" is problematic to carrier networks where stringent SLA
   must be observed.

o Issue 2: INT may significantly increase the user packet's original
   size by adding the instruction header and data at each traversed
   node. The longer the forwarding path and the more the data
   collected, the larger the packet will become. The size may exceed
   the path MTU so either INT cannot apply or the packet needs to be
   fragmented. Limiting the data size or path length reduces the
   effectiveness of INT. On the other hand, the INT header and data
   can be deeply embedded in a packet due to various transport
   protocol and tunnel configurations. The required deep packet
   header inspection and processing may be infeasible to some data
   plane fast path where only a limited number of header bytes are
   accessible.

o Issue 3: INT requires attaching an instruction header to user
   packets to inform network nodes what types of data to collect.
   Due to the header overhead constraint and hardware-friendly
   consideration, TLV is undesirable for data type encoding.
   Instead, iOAM use a bitmap where each bit indicates one pre-
   defined data type [I-D.ietf-ippm-ioam-data]. However, new use
   cases may require new data types. The current allocated 16-bit
   bitmap limits the data type scalability. The proposed bitmap
   extension in [I-D.song-ippm-ioam-data-extension] provides a method

to support more data types but it also increases the iOAM header
size.

o Issue 4: INT header need to be encapsulated into user packets for
   transport. [I-D.brockners-inband-oam-transport] has discussed
   several encapsulation approaches for different transport
   protocols. However, it is difficult to encapsulate extra header
   in MPLS and IPv4 networks which happens to be the most widely
   deployed and where the path-associated telemetry data is most
   wanted by operators. The proposed NVGRE encapsulation for IPv4 in
   [I-D.brockners-inband-oam-transport] requires a tunnel to be built
   between each pair of nodes which may be unrealistic for plain IP
   networks.

o Issue 5: The INT header and data are vulnerable to eavesdropping
   and tampering as well as DoS attack. Extra protective measurement
   is difficult on the fast data path.

o Issue 6: Since INT only exports the telemetry data at the
   designated end node, if the packet is dropped in the network, the
   data will be lost. It cannot pinpoint the packet drop location
   which is required for fault diagnosis.

The above issues are inherent to the INT-based solutions.
Nevertheless, the path-associated data acquired by INT are valuable
for network operators. Therefore, alternative approaches which can
collect the same data but avoid or mitigate the above issues are
desired. This document provides a new proposal named Postcard-Based
Telemetry (PBT) with two different implementation variations, each
having its own trade-off and addressing some or all of the above
issues. The basic idea of PBT is simple: at each node, instead of
inserting the collected data into the user packets, the data are
directly exported through dedicated OAM packets. Such "postcard"
approach is in contrast to the "passport stamps" approach adopted by
INT [DOI_10.1145_2342441.2342453]. The OAM packets or postcards can
be transported in band or out of band, independent of the original
user packets.

2. PBT-M: Postcard-based Telemetry with Packet Marking

This section describes the first variation of PBT. PBT-M aims to
address all the challenges of INT listed above and introduce some new
benefits. We first list all the design requirements of PBT-M.

o Req. 1: We should avoid augmenting user packets with new headers
   or introducing new data plane protocols. This helps to alleviate
   or eliminate the issue 1, 2, 4, and 5. We expect the OAM data
   collecting signaling remains in data plane. Simple packet marking
   techniques suffice to serve this purpose. It is also possible to
   configure the OAM data collecting from the control plane.

o Req. 2: We should make the scheme extensible for collecting
   arbitrary new data to support possible future use cases. The data
   set to be collected is preferred to be configured through
   management plane or control plane. Since there is no limitation
   on the types of data, any data including those generated by
   customized DNPs [I-D.song-opsawg-dnp4iq] can be collected. Since
   there is no size constraints any more, it is free to use the more
   flexible TLV for data type definition. This addresses the issue 2
   and 3.

o Req. 3: We should avoid interfering the normal forwarding and
   affecting the forwarding performance when conducting data plane
   OAM tasks. Hence, the collected data are better to be transported
   independently by dedicated OAM packets through in-band or out-of-
   band channels. The data collecting, processing, assembly,
   encapsulation, and transport are therefore decoupled from the
   forwarding of the corresponding user packets and can be performed
   in data plane slow path if necessary. This addresses the issue 1,
   4, and 5.

o Req. 4: The data collected from each node is not necessarily
   identical, depending on application requirements and node
   capability. Data for different operation modes can be collected
   at the same time. These requirements are either impossible or
   very difficult to be supported by INT in which data types
   collected per node are supposed to be identical and for a single
   mode.

o Req. 5: The flow's path-associated data can be sensitive and the
   security concerns need to be carefully addressed. Sending OAM
   data with independent packets also makes it easy to secure the
   collected data without exposing it to unnecessary entities. For
   example, the data can be encrypted before being sent to the
   collector so passive eavesdropping and man-in-the-middle attack
   can both be deterred. This addresses the issue 5.

o Req. 6: Even if a user packet under inspection is dropped in
   network, the OAM data that have been collected should still be

exported and help to diagnose the packet drop location and reason.
This addresses the issue 6.

2.2. Solution Description

In light of the above discussion, the sketch of the proposed
solution, PBT-M, is as follows. The user packet, if its path-
associated data need to be collected, is marked at the path head
node. At each PBT-aware node, a postcard (i.e., the dedicated OAM
packet triggered by a marked user packet) is generated and sent to a
collector. The postcard contains the data requested by the
management plane. The requested data are configured by the
management plane through data templates (as in IPFIX [RFC7011]) or
other means. Once the collector receives all the postcards for a
single user packet, it can infer the packet's forwarding path and
analyze the data set. The path end node is configured to unmark the
packets to its original format if necessary.

The overall architecture of PBT-M is depict in Figure 1.

```
                  +------------+        +-----------+
                  | Network    |        | Telemetry |
                  | Management |(-------| Data      |
                  |            |        | Collector |
                  +-----:------+        +-----------+
                        :                     ^
                        :configurations       |postcards (OAM pkts)
                        :                      |
            .......................:......................|........
            :           :          :          :      |     :
            :    +---------:---+-----------:---+--+-------:---+
            :    |         :   |           :   |  |       :   |
            V    |         V   |           V   |  |       V   |
         +------+-+     +-----+--+     +------+-+     +------+-+
  usr pkts | Head   |     | Path   |     | Path   |     | End    |
     ====>| Node   |====>| Node   |====>| Node   |====>| Node   |====>
          |        |     | A      |     | B      |     |        |
          +--------+     +--------+     +--------+     +--------+
          gen postcards  gen postcards  gen postcards  gen postcards
          mark usr pkts                                unmark usr pkts
```
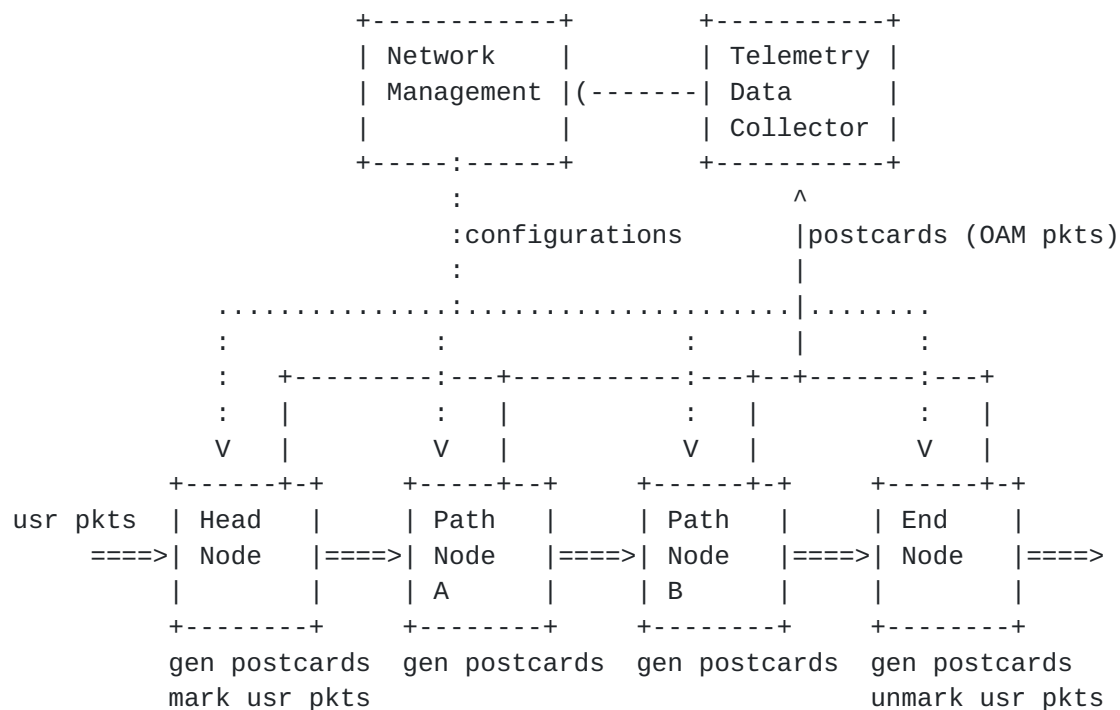
Figure 1: Architecture of PBT-M

Next we discuss the details of the PBT-M solution and the potential
standard gaps.

2.3. **New Challenges**

   Although PBT-M solves the issues of INT, it does introduce a few new
   challenges.

   o Challenge 1: A user packet needs to be marked in order to trigger
      the path-associated data collection. Since we do not want to
      augment user packets with any new header fields (i.e., Req. 1), we
      must take advantage of the existing header fields.

   o Challenge 2: Since the packet header will not carry OAM
      instructions any more, the data plane devices need to be
      configured to know what data to collect. However, in general, the
      forwarding path of a flow packet (due to ECMP or dynamic routing)
      is unknown beforehand. Configuring the data set for each flow at
      all data plane devices is expensive in terms of configuration load
      and data plane resources.

   o Challenge 3: Due to the variable transport latency, the dedicated
      OAM packets for a single packet may arrive at the collector out of
      order or be dropped in networks for some reason. In order to
      infer the packet forwarding path, the collector needs some
      information from the OAM packets to identify the user packet
      affiliation and the order of path node traversal.

2.4. **Considerations on PBT-M Design**

   To address the above challenges, we propose several design details of
   PBT-M.

2.4.1. **Packet Marking**

   Instead of stuffing new header fields into user packets, it is
   preferred to reuse some existing header fields. To trigger the path-
   associated data collection, usually a single bit is sufficient.
   While no such bit is available, other packet marking techniques are
   needed. we discuss three possible application scenarios.

   o IPv4. IPFPM [I-D.ietf-ippm-alt-mark] is an IP flow performance
      measurement framework which also requires a single bit for packet
      coloring. The difference is that IPFPM does in-network
      measurement while PBT only collects and exports data at network
      nodes (i.e., the data analysis is done at the collector). IPFPM
      suggests to use some reserved bit of the Flag field or some unused
      bit of the TOS field. Actually, IPFPM can be considered a subcase
      of PBT so the same bit can be used for PBT. The management plane
      is responsible to configure the actual operation mode.

   o SFC NSH. The OAM bit in NSH header can be used to trigger the
      path-associated data collection [I-D.ietf-sfc-nsh]. PBT does not
      add any other metadata to NSH.

   o MPLS. Instead of choosing a header bit, we take advantage of the
      synonymous flow label [I-D.bryant-mpls-synonymous-flow-labels]
      approach to mark the packets. A synonymous flow label indicates
      the path-associated data should be collected and forwarded through
      a postcard.

2.4.2. Flow Path Discovery

   By default, all PBT-aware nodes are configured to react to the marked
   packets by exporting some basic data such as node ID and TTL before a
   data set template for that flow is configured. This way, the
   management plane can learn the flow path.

   If the management plane wants to collect the path-associated data for
   some flow, it configures the head node(s) with a probability or time
   interval for the flow packet marking. When the first marked packet
   is forwarded in the network, the PBT-aware nodes will export the
   basic data to the collector. Hence, the flow path is identified. If
   other types of data need to be collected, the management plane can
   further configure the data set template to the target nodes. The
   PBT-aware nodes would collect and export data accordingly if the
   packet is marked and a data set template is present.

   If for any reason, the flow path is changed. The new path nodes can
   be learnt immediately by the collector, so the management plane
   controller can be informed to configure the new path nodes. The
   outdated configuration can be automatically timed out or explicitly
   revoked by the management plane controller.

2.4.3. Packet Identity for Export Data Correlation

   The collector needs to correlate all the OAM packets for a single
   user packet. Once this is done, the TTL (or the timestamp, if the
   network time is synchronized) can be used to infer the flow
   forwarding path. The key issue here is to uniquely identify the user
   packet affiliation of the OAM packet.

   The first possible approach is to include the flow ID plus the user
   packet ID in the OAM packets. The user packet ID can be some unique
   information pertaining to a user packet (e.g., the sequence number of
   a TCP packet).

   If the packet marking interval is long enough, then the flow ID
   itself is enough to identify the user packet. That is, we can assume

all the exported OAM packets for the same flow during a short period
of time belong to the same user packet.

If the network is synchronized, then the flow ID plus the timestamp
at each node can also infer the packet identity. However, some
errors may occur under some circumstances. For example, if two
consecutive user packets from the same flows are both marked and one
exported OAM packet from a node is lost, then it is difficult for the
collector to decide which user packet the remaining OAM packet
belongs to. In many cases, such rare errors may be tolerable.

3. PBT-I: Postcard-based Telemetry with Instruction Header

Since PBT-M has some challenges as listed in Section 2.3, this
section describes another variation of PBT, which essentially
compromises some of the design requirements listed in Section 2.1,
yet retains most of the benefits of PBT.

PBT-I can be seen as a trade-off between INT/iOAM and PBT-M. PBT-I
needs to add a fixed length instruction header to user packets for
OAM data collection. However, the collected data will be exported
through dedicated OAM packets. On the one hand, PBT-I violates the
Req. 1 in Section 2.1. It also makes it harder to meet the Req. 2.
On the other hand, the overhead of the instruction header is under
control and user packets will not inflate with path length or
telemetry data amount. We also introduce an optimization to mitigate
the impact on Req. 2. In return, PBT-I addresses all the challenges
of PBT-M:

o There is no need to find an existing header field to mark a user
   packet. The encapsulation of the PBT-I instruction header can use
   the same method for iOAM. So far, the iOAM header encapsulation
   methods have been defined for several protocols, including IPv6,
   VXLAN-GPE, NSH, SRv6
   [I-D.brockners-inband-oam-transport],[I-D.ietf-sfc-ioam-nsh],
   GENEVE [I-D.brockners-ippm-ioam-geneve], and GRE
   [I-D.weis-ippm-ioam-gre]. [I-D.song-mpls-extension-header]
   describes the approach to encapuslate the instruction header into
   MPLS packets.

o There is no need to configure the node about the set of data to be
   collected since all the information is carried in the instruction
   header. We adopt the optimization in
   [I-D.song-ippm-ioam-data-extension] to address the data
   scalability issue incurred by the instruction header.

o The instruction header contains enough information to help
   correlate the OAM packets belonging to a user packets. Even

better, new fields are added to track the flow and the packet, so
any packet under inspection can be easily identified even in
tunnels and the collector can easily check if any user packet
under inspection or its OAM data packet is missing.

3.1. Solution Description

The sketch of the proposed solution, PBT-I, is as follows. If the
path-associated data need to be collected for a user packet, an
instruction header named Telemetry Instruction Header (TIH) is
inserted into the packet at the path head node. At each PBT-aware
node, a postcard is generated and sent to a collector. Once the
collector receives all the postcards for a single user packet, it can
combine and analyze the data set. The path end node is configured to
remove the TIH.

The overall architecture of PBT-I is depict in Figure 2.

```
                            +-----------+
                            | Telemetry |
                            | Data      |
                            | Collector |
                            +-----------+
                                 ^
                                 |postcards (OAM pkts)
                                 |
                                 |
                                 |
            +--------------+------+-------+--------------+
            |              |             |              |
            |              |             |              |
        +---+----+     +---+----+    +---+----+     +---+----+
 usr pkts | Head   |     | Path   |    | Path   |     | End    |
     ====>| Node   |====>| Node   |===>| Node   |====>| Node   |====>
         |        |     | A      |    | B      |     |        |
         +--------+     +--------+    +--------+     +--------+
         insert TIH                                 remove TIH
         gen postcards  gen postcards gen postcards gen postcards
```
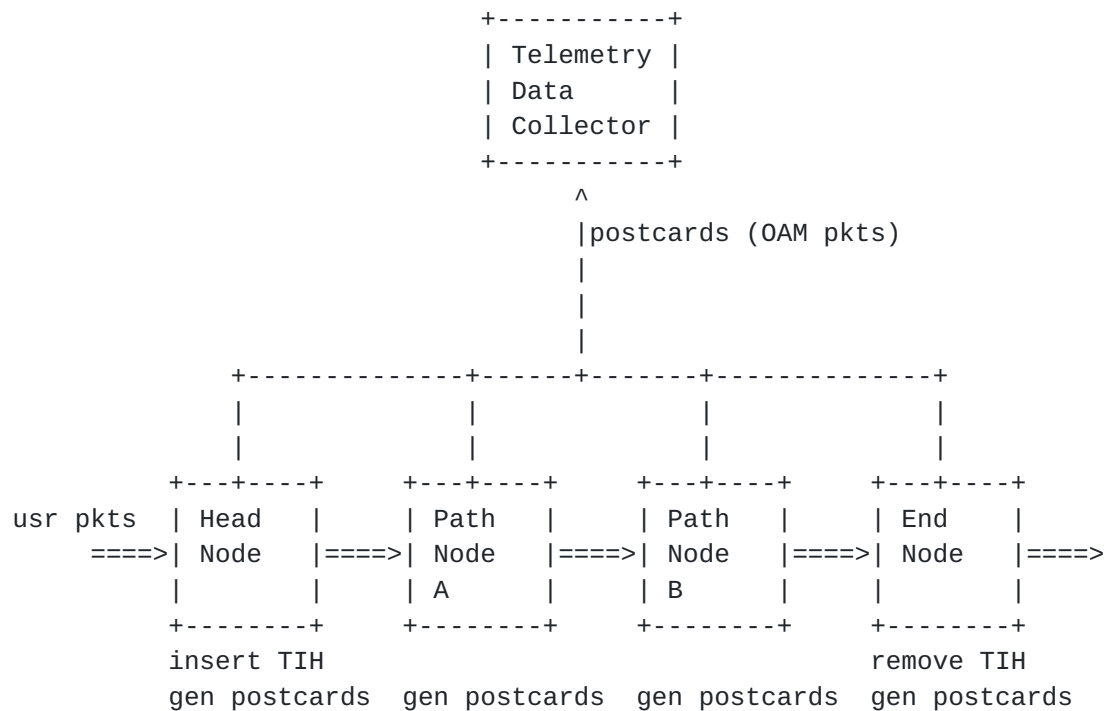
Figure 2: Architecture of PBT-I

3.2. **PBT-I Telemetry Instruction Header**

   The proposed format of TIH is shown in Figure 3.

```
    0              0 0            1 1            2 2            3
    0              7 8            5 6            3 4            1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Next Header   |  TIH Length  |    Reserved   |   Hop Count   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                          Flow ID                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                          Flow ID                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                       Sequence Number                        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Data Element Bitmap                     |E|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           Data Element Bitmap Extension (optional)          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

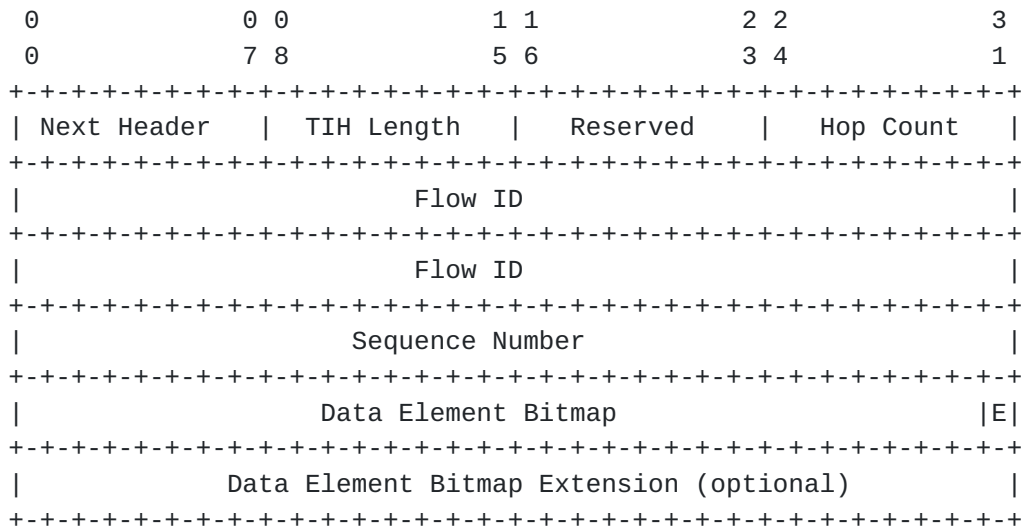   Figure 3: TIH Format

   o Next Header: the 8-bit indicator indicating the next protocol
      after TIH.

   o TIH Length: the 8-bit Instruction Header Length field. The value
      is in the unit of 4-octet words.

   o Hop Count: the 8-bit Hop Count field. It is used to count the
      hops of the TIH-aware nodes, starting for 0 and incremented by 1
      at each PBT-aware node.

   o Flow ID: the 64-bit flow ID field. If the actual flow ID is
      shorter than 64 bits, it is right aligned with the leading bits
      being filled with 0. The field is set at the head node.

   o Sequence Number: the 32-bit sequence number starting from 0 and
      increasing by 1 for each following monitored packet from the same
      flow at the head node.

   o Data Element Bitmap: the 31-bit bitmap indicating the list of
      required data elements.

   o E bit: Data Extension Indicator. This bit has a default value of
      0. If set, it indicates a following data element bitmap extension
      word, as described in [I-D.song-ippm-ioam-data-extension].

The specification of the data element bitmap is as follows:

o Bit 0: Node ID (32b)

o Bit 1: TTL (8b), derived from the TTL field in the packet header

o Bit 2: Ingress Interface ID (32b)

o Bit 3: Egress Interface ID (32b)

o Bit 4: Ingress Timestamp (64b)

o Bit 5: Egress Timestamp (64b)

o Bit 6: Transit Delay (32b)

o Bit 7: Egress Queue Depth (32b)

o Bit 8: Ingress MTU (32b)

o Bit 9: Egress MTU (32b)

o Bit 10: Egress Packet Length (32b)

o Bit 11-30: Undefined

3.3. **Considerations on PBT-I Design**

4. **Security Considerations**

Several security issues need to be considered.

o Eavesdrop and tamper: the OAM packets can be encrypted and
   authenticated.

o DoS attack: PBT can be limited to a single administration domain,
   or the enforce mark or instruction header are checked at the
   domain edge. The node can rate limit the extra traffic incurred
   by the OAM data.

5. **IANA Considerations**

TBD.

6. Contributors

   TBD.

7. Acknowledgments

   TBD.

8. Informative References

   [DOI_10.1145_2342441.2342453]
              Handigol, N., Heller, B., Jeyakumar, V., MaziA(C)res, D.,
              and N. McKeown, "Where is the debugger for my software-
              defined network?", Proceedings of the first workshop on
              Hot topics in software defined networks - HotSDN '12,
              DOI 10.1145/2342441.2342453, 2012.

   [I-D.brockners-inband-oam-requirements]
              Brockners, F., Bhandari, S., Dara, S., Pignataro, C.,
              Gredler, H., Leddy, J., Youell, S., Mozes, D., Mizrahi,
              T., <>, P., and r. remy@barefootnetworks.com,
              "Requirements for In-situ OAM", draft-brockners-inband-
              oam-requirements-03 (work in progress), March 2017.

   [I-D.brockners-inband-oam-transport]
              Brockners, F., Bhandari, S., Govindan, V., Pignataro, C.,
              Gredler, H., Leddy, J., Youell, S., Mizrahi, T., Mozes,
              D., Lapukhov, P., and R. Chang, "Encapsulations for In-
              situ OAM Data", draft-brockners-inband-oam-transport-05
              (work in progress), July 2017.

   [I-D.brockners-ippm-ioam-geneve]
              Brockners, F., Bhandari, S., Govindan, V., Pignataro, C.,
              Gredler, H., Leddy, J., Youell, S., Mizrahi, T., Mozes,
              D., Lapukhov, P., and R. Chang, "Geneve encapsulation for
              In-situ OAM Data", draft-brockners-ippm-ioam-geneve-01
              (work in progress), June 2018.

   [I-D.bryant-mpls-synonymous-flow-labels]
              Bryant, S., Swallow, G., Sivabalan, S., Mirsky, G., Chen,
              M., and Z. Li, "RFC6374 Synonymous Flow Labels", draft-
              bryant-mpls-synonymous-flow-labels-01 (work in progress),
              July 2015.

   [I-D.clemm-netconf-push-smart-filters-ps]
             Clemm, A., Voit, E., Liu, X., Bryskin, I., Zhou, T.,
             Zheng, G., and H. Birkholz, "Smart filters for Push
             Updates - Problem Statement", draft-clemm-netconf-push-
             smart-filters-ps-00 (work in progress), October 2017.

   [I-D.ietf-ippm-alt-mark]
             Fioccola, G., Capello, A., Cociglio, M., Castaldelli, L.,
             Chen, M., Zheng, L., Mirsky, G., and T. Mizrahi,
             "Alternate Marking method for passive and hybrid
             performance monitoring", draft-ietf-ippm-alt-mark-14 (work
             in progress), December 2017.

   [I-D.ietf-ippm-ioam-data]
             Brockners, F., Bhandari, S., Pignataro, C., Gredler, H.,
             Leddy, J., Youell, S., Mizrahi, T., Mozes, D., Lapukhov,
             P., Chang, R., and d. daniel.bernier@bell.ca, "Data Fields
             for In-situ OAM", draft-ietf-ippm-ioam-data-00 (work in
             progress), September 2017.

   [I-D.ietf-netconf-udp-pub-channel]
             Zheng, G., Zhou, T., and A. Clemm, "UDP based Publication
             Channel for Streaming Telemetry", draft-ietf-netconf-udp-
             pub-channel-01 (work in progress), November 2017.

   [I-D.ietf-netconf-yang-push]
             Clemm, A., Voit, E., Prieto, A., Tripathy, A., Nilsen-
             Nygaard, E., Bierman, A., and B. Lengyel, "YANG Datastore
             Subscription", draft-ietf-netconf-yang-push-12 (work in
             progress), December 2017.

   [I-D.ietf-sfc-ioam-nsh]
             Brockners, F., Bhandari, S., Govindan, V., Pignataro, C.,
             Gredler, H., Leddy, J., Youell, S., Mizrahi, T., Mozes,
             D., Lapukhov, P., and R. Chang, "NSH Encapsulation for In-
             situ OAM Data", draft-ietf-sfc-ioam-nsh-00 (work in
             progress), May 2018.

   [I-D.ietf-sfc-nsh]
             Quinn, P., Elzur, U., and C. Pignataro, "Network Service
             Header (NSH)", draft-ietf-sfc-nsh-28 (work in progress),
             November 2017.

   [I-D.sambo-netmod-yang-fsm]
             Sambo, N., Castoldi, P., Fioccola, G., Cugini, F., Song,
             H., and T. Zhou, "YANG model for finite state machine",
             draft-sambo-netmod-yang-fsm-00 (work in progress), October
             2017.

   [I-D.song-ippm-ioam-data-extension]
             Song, H. and T. Zhou, "In-situ OAM Data Type Extension",
             draft-song-ippm-ioam-data-extension-00 (work in progress),
             October 2017.

   [I-D.song-ippm-ioam-tunnel-mode]
             Song, H., Li, Z., Zhou, T., and Z. Wang, "In-situ OAM
             Processing in Tunnels", draft-song-ippm-ioam-tunnel-
             mode-00 (work in progress), June 2018.

   [I-D.song-mpls-extension-header]
             Song, H., Li, Z., Zhou, T., and L. Andersson, "MPLS
             Extension Header", draft-song-mpls-extension-header-01
             (work in progress), August 2018.

   [I-D.song-opsawg-dnp4iq]
             Song, H. and J. Gong, "Requirements for Interactive Query
             with Dynamic Network Probes", draft-song-opsawg-dnp4iq-01
             (work in progress), June 2017.

   [I-D.talwar-rtgwg-grpc-use-cases]
             Specification, g., Kolhe, J., Shaikh, A., and J. George,
             "Use cases for gRPC in network management", draft-talwar-
             rtgwg-grpc-use-cases-01 (work in progress), January 2017.

   [I-D.weis-ippm-ioam-gre]
             Weis, B., Brockners, F., crhill@cisco.com, c., Bhandari,
             S., Govindan, V., Pignataro, C., Gredler, H., Leddy, J.,
             Youell, S., Mizrahi, T., Kfir, A., Gafni, B., Lapukhov,
             P., and M. Spiegel, "GRE Encapsulation for In-situ OAM
             Data", draft-weis-ippm-ioam-gre-00 (work in progress),
             March 2018.

   [RFC2925] White, K., "Definitions of Managed Objects for Remote
             Ping, Traceroute, and Lookup Operations", RFC 2925,
             DOI 10.17487/RFC2925, September 2000,
             <https://www.rfc-editor.org/info/rfc2925>.

   [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed.,
             and A. Bierman, Ed., "Network Configuration Protocol
             (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011,
             <https://www.rfc-editor.org/info/rfc6241>.

   [RFC7011] Claise, B., Ed., Trammell, B., Ed., and P. Aitken,
             "Specification of the IP Flow Information Export (IPFIX)
             Protocol for the Exchange of Flow Information", STD 77,
             RFC 7011, DOI 10.17487/RFC7011, September 2013,
             <https://www.rfc-editor.org/info/rfc7011>.

Authors' Addresses

    Haoyu Song (editor)
    Huawei
    2330 Central Expressway
    Santa Clara, 95050
    USA

    Email: haoyu.song@huawei.com

    Tianran Zhou
    Huawei
    156 Beiqing Road
    Beijing, 100095
    P.R. China

    Email: zhoutianran@huawei.com

    Zhenbin Li
    Huawei
    156 Beiqing Road
    Beijing, 100095
    P.R. China

    Email: lizhenbin@huawei.com