

UDP based Publication Channel for Streaming Telemetry

draft-zheng-netconf-udp-pub-channel-00

Guangying Zheng

Tianran Zhou

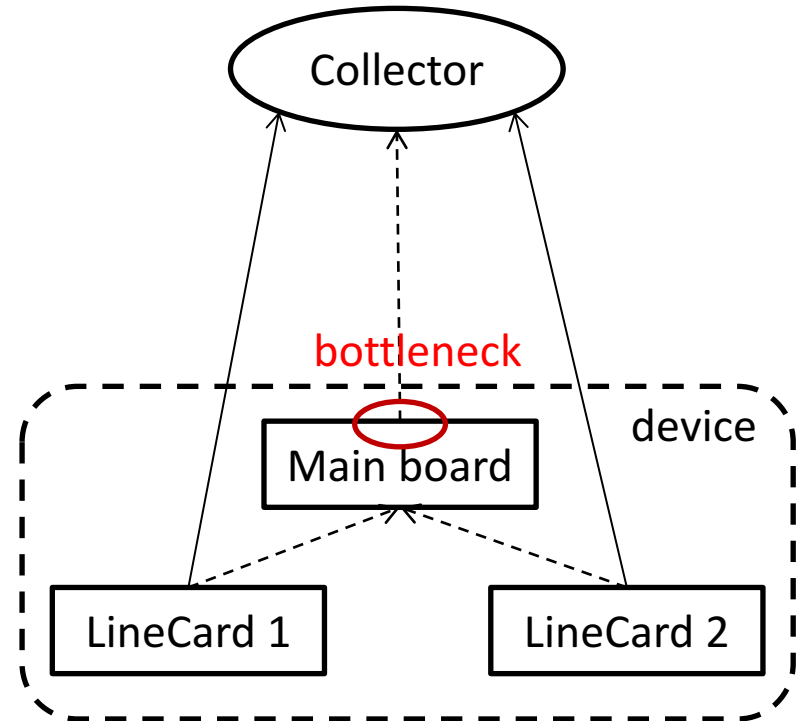
Alexander Clemm

Related Work

- Subscribing to YANG datastore push updates
 - *ietf-netconf-yang-push-07*
- Subscribing to Event Notifications
 - *ietf-netconf-subscribed-notifications-03*
- Netconf Transport
 - *ietf-netconf-netconf-event-notifications-04*
- Restconf + HTTP Transport
 - *ietf-netconf-restconf-notif-02*

Problem to Solve

- Large amount of data collection from devices with main board and line cards.
- Existing solution consider only one push server reside in the main board.
 - Result in performance bottleneck when data are forwarded to the main board and converged to one consolidated stream.
- Request for **distributed data collection mechanism** which can directly push data from line cards to a collector.



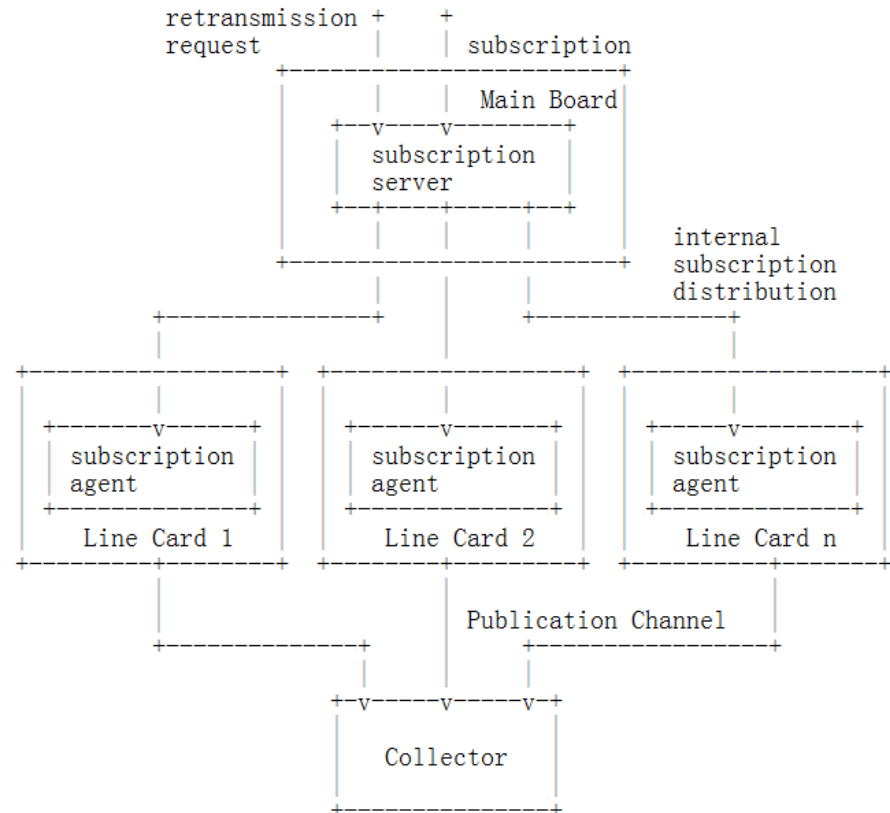
Why UDP based Publication Channel

- Separate the management and control of subscriptions from the transport that is used to actually stream and deliver the data.
- Existing transport including Netconf and HTTP2 are TCP based.
 - Data collector will suffer a lot of TCP connections from many line cards equipped on different devices.
 - Because of the lightweight UDP encapsulation, higher frequency and better transit performance can be achieved, which is important for streaming telemetry.
 - As no connection state needs to be maintained, UDP encapsulation can be easily implemented by hardware which will further improve the performance.

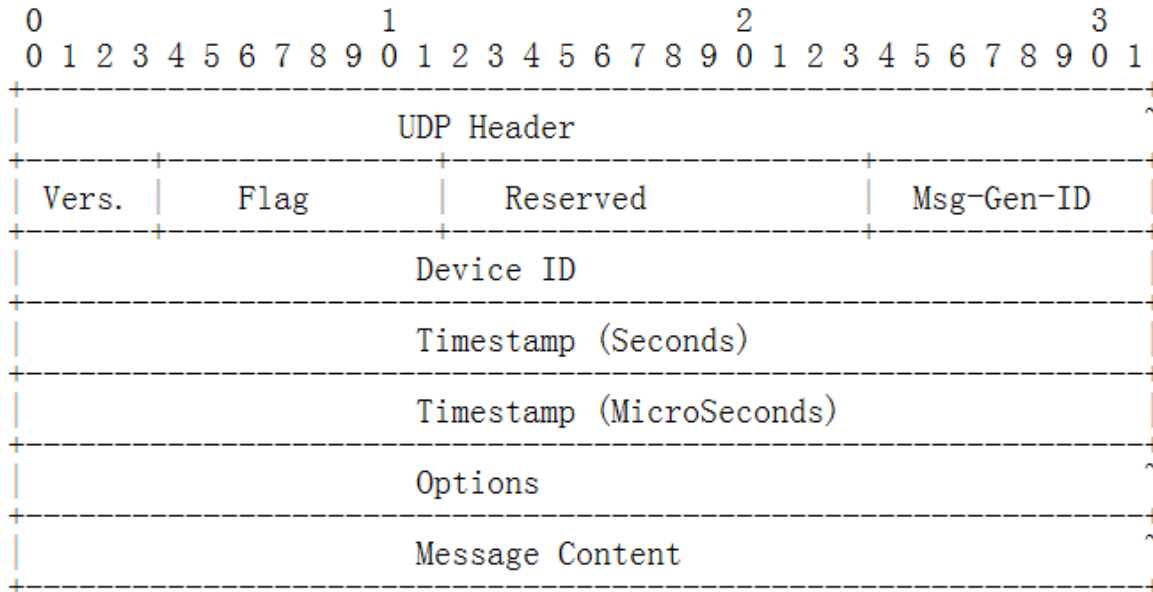
Solution Overview

Technique points:

- Subscription model that allow a single subscription to control multiple internal data originators
- Interaction between Subscription server and Subscription agent
- UDP based message header
- Retransmission procedure



UDP Telemetry Header



- Flag: indicates supported features on reliability, authentication, encryption, etc
- Message generator ID: e.g. Line card number
- Device ID: global unique number within the management domain to indentify a device.
- Timestamp: time when the message is generated/sent
- Options: parameters related to features enabled by the flag.

Thank you