Distributed Data Collection

T. Zhou, G. Zheng   Huawei

E. Voit   Cisco Systems

A. Clemm   Huawei

A. Bierman   YumaWorks

History

- Extensions to NETCONF for distributed data collection.

- draft-ietf-netconf-udp-pub-channel

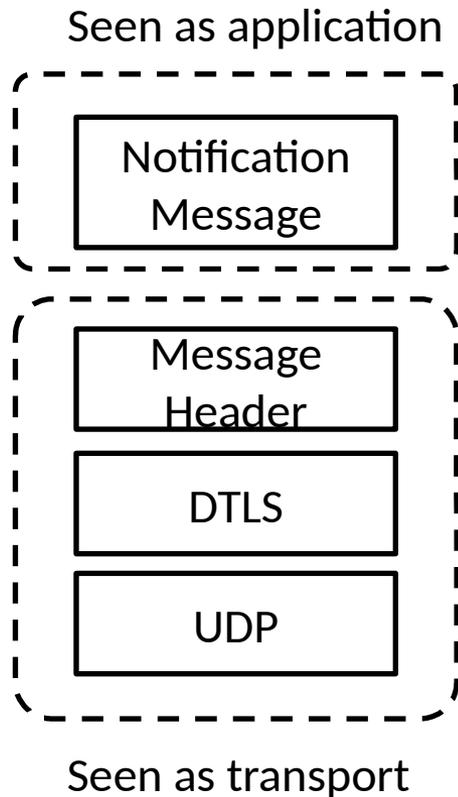| draft-zhou-netconf-multi-stream-originators: -- The generic issues on distributed data collection	draft-ietf-netconf-udp-pub-channel: -- UDP based transport for the publication channel

UDP based Publication Channel for Streaming Telemetry

draft-ietf-netconf-udp-pub-channel

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 RESTCONF are TCP based.
  - Data collector will suffer a lot of TCP connections from many line cards equipped on different devices.
  - As no connection state needs to be maintained, UDP encapsulation can be easily implemented by hardware which will further improve the performance.
  - Because of the lightweight UDP encapsulation, higher frequency and better transit performance can be achieved, which is important for streaming telemetry.

Solutions

Seen as application

```
┌ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┐
│  ┌─────────────┐  │
   │ Notification │
│  │   Message    │  │
   └─────────────┘
└ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┘

┌ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┐
│  ┌─────────────┐  │
   │   Message    │
│  │   Header     │  │
   └─────────────┘
│  ┌─────────────┐  │
   │    DTLS      │
│  └─────────────┘  │
   ┌─────────────┐
│  │    UDP       │  │
   └─────────────┘
└ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┘
```

Seen as transport

- DTLS: provide reusable security and authentication functions over UDP
- Message Header: some important information before de-serializing the notification.
  – Encoding method: GPB, CBOR, JSON, XML
  – Message generator ID
  – Time stamp
  – Sequence number
  – Fragmentation
  – Options for extensibility
- Notification Message:
  – include a notification header, as defined in draft-ietf-netconf-notification-messages-02
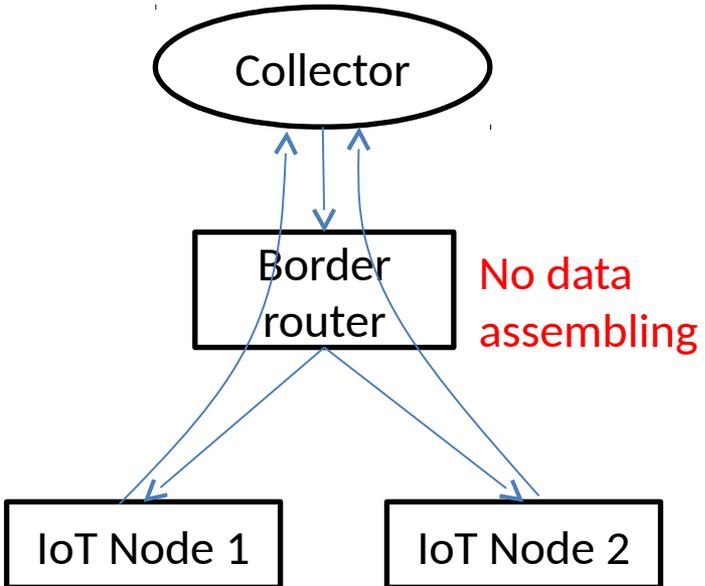 – Encoded with the content.

Subscription to Multiple Steam Originators

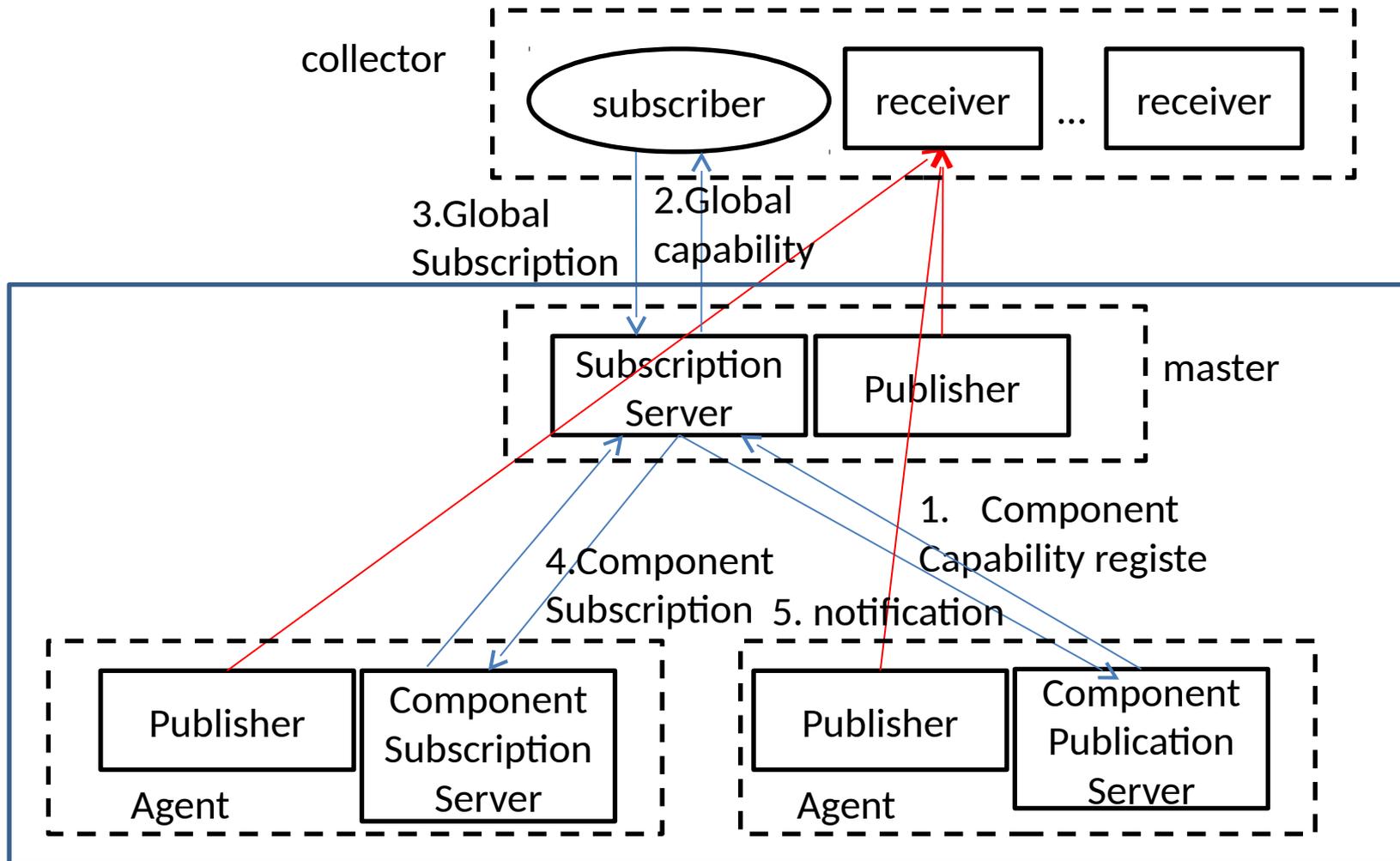draft-zhou-netconf-multi-stream-originators-01

Use Cases

Collector

bottleneck

device

Main board

LineCard 1   LineCard 2
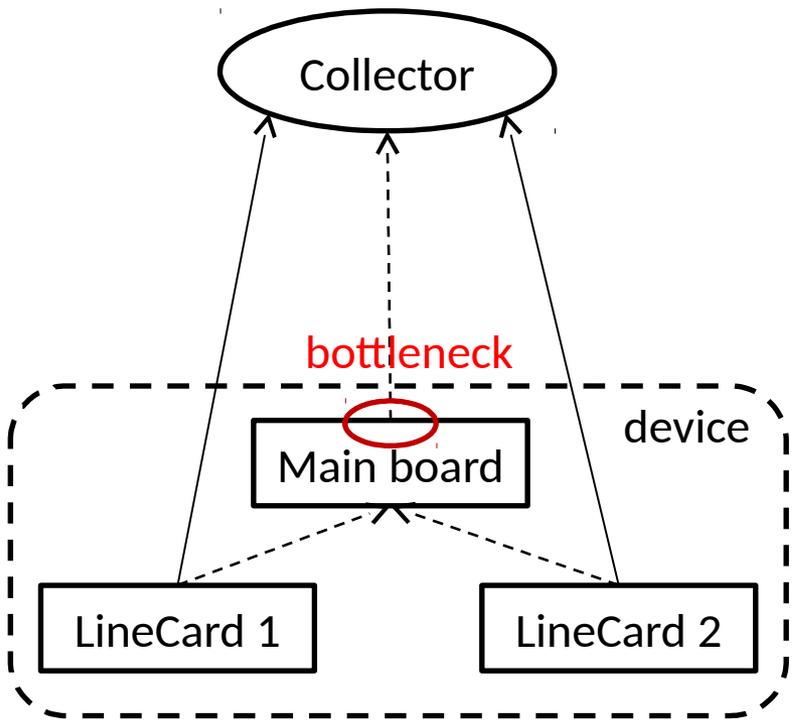
Use case 1: Data Collection from Devices with Main-board and Line-cards

Collector

Border router

No data assembling

IoT Node 1   IoT Node 2

Use case 2: IoT Data Collection

Solution Overview

Issues Being Worked

- Subscription Decomposition
  - Keep track of resources and the associated publisher
  - Make decision on decomposing the global subscription into multiple component subscriptions.
- Publication Composition
  - Compose the component notifications into one.
- Subscription Management
  - Error codes related to the Subscription Decomposition and Component Subscription
- Notifications on Subscription State Changes
  - Each component subscription maintains its own subscription state and is responsible for sending its own OAM notifications.
- Potential Issues
  - Synchronization

Next step

- Encourage comments and suggestions.
- Should the WG adopt this draft?