Subscribing to YANG datastore push updates
draft-ietf-netconf-yang-push-02
NETCONF WG
IETF #95 Buenos Aires
7-April-2015

Alexander Clemm
Alberto Gonzalez Prieto
Eric Voit
Ambika Prasad Tripathy
Einar Nilsen-Nygaard

<alex|albertgo|evoit|ambripa|einar nn@cisco.com>
Updates since IETF #94

• Two revision updates (00→01→02)
• Establish-subscription replaces create-subscription
  – Use for negotiated subscription vs unnegotiated (RFC 5277)
• Updated RPC definitions for establish, modify, delete subscription
  – e.g. error return codes
• Added push source configuration for static subscriptions (e.g., VRF)
• Incorporated content from draft-voit-netconf-restconf-yang-push
  – Subscription prioritization, state machine, YANG
  – NOT transport (RESTCONF, HTTP, HTTP/2)
• Adopted capabilities from OC-Telemetry.yang
  – Multiple static receivers, DSCP
• Made associated YANG updates
Vision

• Common subscription control mechanisms
  – YANG-Push for YANG Datastore updates
  – Event Notifications for General Notifications

• Robust, Flexible, Extensible

• Incrementally addressable components / drafts
## NETCONF Event Notification & YANG Push Drafts - Key Elements

<table>
<thead>
<tr>
<th></th>
<th>Event Notifications</th>
<th>YANG Push</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.0</td>
<td>1.1</td>
</tr>
<tr>
<td>Types of Subscription</td>
<td>Dynamic</td>
<td>Dynamic and Static</td>
</tr>
<tr>
<td>Subscriptions per Session</td>
<td>one</td>
<td>many</td>
</tr>
<tr>
<td>Negotiation</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>RPCs</td>
<td>create</td>
<td>establish, modify, delete</td>
</tr>
<tr>
<td>Control Plane Notifications</td>
<td>None</td>
<td>started, suspended, resumed, terminated, modified</td>
</tr>
<tr>
<td>Data Plane Notifications</td>
<td>notification</td>
<td>+subscription-id</td>
</tr>
<tr>
<td></td>
<td></td>
<td>push-update, push-change-update</td>
</tr>
<tr>
<td>Transport</td>
<td>NETCONF</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>RESTConf, HTTP, HTTP2</td>
<td>No</td>
</tr>
</tbody>
</table>

*draft-ietf-netconf-yang-push*
IETF95 Hackathon – YANG Push

• YANG-PUSH Client from OpenDaylight Beryllium

• XR Router Demo Code

• Extended implementation: draft-ietf-netconf-yang-push
  – Create Subscription
  – Delete Subscription
  – On-Change
  – Periodic

• Hands-on Demo at tonight’s Bits-N-Bytes
Model overview: streams + filters

module: ietf-yang-push
  +--ro update-streams
    |  +--ro update-stream*  update-stream
  +--rw filters
    |  +--rw filter*  [filter-id]
    |    +--rw filter-id  filter-id
    |    +--rw (filter-type)?
    |    |    +--:(subtree)
    |    |    |    +--rw subtree-filter
    |    |    +--:(xpath)
    |    |    |    +--rw xpath-filter?  yang:xpath1.0
    |    |    +--:(rfc5277)
    |    |    |    +--rw filter
    |    +--rw filter
Model overview:
Subscription configuration

```
+--rw subscription-config {configured-subscriptions}?
  |   +--rw yang-push-subscription* [subscription-id]
  |   |   +--rw subscription-id subscription-id
  |   |   +--rw stream? update-stream
  |   |   +--rw encoding? encoding
  |   |   +--rw subscription-start-time? yang:date-and-time
  |   |   +--rw subscription-stop-time? yang:date-and-time
  |   |   +--rw (filterspec)?
  |   |   |   +--:(inline)
  |   |   |   |   +--rw (filter-type)?
  |   |   |   |   |   +--:(subtree)
  |   |   |   |   |   |   +--rw subtree-filter
  |   |   |   |   |   |   +--:(xpath)
  |   |   |   |   |   |   |   +--rw xpath-filter? yang:xpath1.0
  |   |   |   |   |   |   +--:(rfc5277)
  |   |   |   |   |   |   |   +--rw filter
  |   |   |   |   |   |   |   +--:(by-reference)
  |   |   |   |   |   |   |   |   +--rw filter-ref?
  |   |   |   |   |   |   |   |   |   filter-ref
```

Optional feature, only applies to static subscriptions
Model overview:
Subscription configuration (contd)

```
+--rw subscription-config {configured-subscriptions}?  |
   +--rw yang-push-subscription* [subscription-id]

...  |
   +--rw (update-trigger)?                 |
   |   +--:(periodic)                     |
   |   |   +--rw period                    yang:timeticks
   |   +--:(on-change) {on-change}?       |
   |   |   +--rw no-synch-on-start?         empty
   |   |   +--rw dampening-period           yang:timeticks
   |   |   +--rw excluded-change*           change-type
   |   +--rw receiver* [address]          |
   |   |   +--rw address inet:host         |
   |   |   +--rw port? inet:port-number    |
   |   +--rw protocol? transport-protocol|
   +--rw (push-source)?                 |
   |   +--:(interface-originated)        |
   |   |   +--rw source-interface? if:interface-ref
   |   |   +--:(address-originated)         |
   |   |   +--rw source-vrf? uint32        |
   |   |   +--rw source-address            inet:ip-address-no-zone
   |   +--rw dscp? inet:dscp {configured-subscriptions}?|
   +--rw subscription-priority? uint8   |
   +--rw subscription-dependency? String
```

Note: items in blue are new or modified since -00
Model overview:

Subscription monitoring

• Applies to both dynamic and static subscriptions
• Mirrors subscription parameters from RPC or config

Note: items in blue are new or modified since -00
Model overview: RPCs (input)

+---x establish-subscription
  | +---w input
  |    | +---w stream? update-stream
  |    | +---w encoding? encoding
  |    | +---w subscription-start-time? yang:date-and-time
  |    | +---w subscription-stop-time? yang:date-and-time
  |    | +---w (filterspec)?
  |    |    | +--:(inline)
  |    |    |    | +---w (filter-type)?
  |    |    |    |    | +--:(subtree)
  |    |    |    |    |    | +---w subtree-filter
  |    |    |    |    |    | +--:(xpath)
  |    |    |    |    |    |    | +---w xpath-filter? yang:xpath1.0
  |    |    |    |    |    | +--:(rfc5277)
  |    |    |    |    |    |    | +---w filter
  |    |    |    |    |    |    | +--:(by-reference)
  |    |    |    |    |    |    |    | +---w filter-ref? filter-ref
  |    |    |    |    |    |    | +---w (update-trigger)?
  |    |    |    |    |    |    |    | +--:(periodic)
  |    |    |    |    |    |    |    |    | +---w period yang:timeticks
  |    |    |    |    |    |    |    |    | +--:(on-change) {on-change}?
  |    |    |    |    |    |    |    |    |    | +---w no-synch-on-start? empty
  |    |    |    |    |    |    |    |    |    | +---w dampening-period yang:timeticks
  |    |    |    |    |    |    |    |    |    | +---w excluded-change* change-type
  |    |    |    |    |    |    |    |    |    | +---w dscp? inet:dscp {configured-subscriptions}?
  |    |    |    |    |    |    |    |    |    | +---w subscription-priority? uint8
  |    |    |    |    |    |    |    |    |    | +---w subscription-dependency? string

• Input parameters mirror those of static subscription, except for receiver specification
• Same underlying groupings

Note: items in blue are new or modified since -00
Model overview: RPCs (output)

+---x establish-subscription
  | +--ro output
  |   | +--ro subscription-result
  |   |     | +--ro (result)?
  |   |     |     | +--:(success)
  |   |     |     |     | +--ro subscription-id
  |   |     | +--:(no-success)
  |   |     |     | +--ro stream?
  |   |     |     |     | +--ro encoding?
  |   |     |     |     | +--ro subscription-start-time? yang:date-and-time
  |   |     |     |     | +--ro subscription-stop-time? yang:date-and-time
  |   |     |     | +--ro (filterspec)?
  |   |     |     |     | +--:(inline)
  |   |     |     |     |     | +--ro (filter-type)?
  |   |     |     |     |     |     | +--:(subtree)
  |   |     |     |     |     |     |     | +--ro subtree-filter
  |   |     |     |     |     |     |     | +--:(xpath)
  |   |     |     |     |     |     |     |     | +--ro xpath-filter? yang:xpath1.0
  |   |     |     |     |     |     |     | +--:(rfc5277)
  |   |     |     |     |     |     |     |     | +--ro filter
  |   |     |     |     |     |     | +--:(by-reference)
  |   |     |     |     |     |     |     | +--ro filter-ref?
  |   |     |     |     |     |     | +--ro (update-trigger)?
  |   |     |     |     |     |     |     | +--:(periodic)
  |   |     |     |     |     |     |     |     | +--ro period yang:timeticks
  |   |     |     |     |     |     |     | +--:(on-change) {on-change}?
  |   |     |     |     |     |     |     |     | +--ro no-synch-on-start? empty
  |   |     |     |     |     |     |     |     | +--ro dampening-period yang:timeticks
  |   |     |     |     |     |     |     |     | +--ro excluded-change* change-type
  |   |     |     |     |     |     |     |     | +--ro dscp? inet:dscp {configured-subscriptions}?
  |   |     |     |     |     |     |     |     | +--ro subscription-priority? uint8
  |   |     |     |     |     |     |     |     | +--ro subscription-dependency? string

• “No success” returns parameter settings that should be changed to lead to future success (subscription negotiation)

Note: items in blue are new or modified since -00
Model overview (contd.)

• Not depicted:
  – RPCs for modify, delete subscription
  – Notifications
    • Push updates
      – Push-update, push-change-update
    • Control notifications
      – Subscription started, stopped, suspended, resumed
Next Steps

• At this point, YANG-push is “self-sufficient” for YANG Subscriptions over NETCONF

• At the end of WG, discuss compartmentalizing functionality in different drafts:
  – Transport independence (NETCONF, RESTCONF, HTTP, IPFIX?, other?)
  – Common subscription model for Event Notifications and YANG Datastore Push

• Recommendation: split subscription mgmt and transports
Thank you!
**NETCONF Event Notification & Datastore Push Drafts - Context**

<table>
<thead>
<tr>
<th>What you need</th>
<th>Consume a stream of Publisher generated messages at the cadence determined by the Publisher</th>
<th>Consume a stream of Publisher generated YANG data updates at a cadence negotiated with the Subscriber</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed</td>
<td>RFC5277 draft-gonzalez-netconf-5277bis</td>
<td>draft-ietf-netconf-yang-push draft-voit-netconf-restconf-yang-push</td>
</tr>
</tbody>
</table>