

Subscribing to YANG datastore push updates draft-clemm-netconf-yang-push-00.txt

Alexander Clemm, Alberto Gonzalez Prieto, Eric Voit
IETF 92, 24 March 2015

Motivation

- Many applications require continuous updates of datastore contents
 - Service assurance: continuous monitoring
 - Big Data: analyze network state & subscribe to continuous stream
 - SDN Controllers: caching of remote data
- Periodic polling has limitations (known from SNMP)
 - Additional load on network and devices
 - Lack of robustness, dealing with missed polling cycles
 - Difficult calibration, synchronization of polling cycles across network (makes polled data difficult to compare)
- Current interactions with datastore are request/response based
 - RFC 6470 defines configuration change notifications (root node+edit operation)
 - YANG datastores contain increasingly operational data; there is nothing that pushes actual datastore content

Solution overview

- Provide push mechanism as alternative to polling, centered around a datastore subscription service
- Requirements per draft-ietf-i2rs-pub-sub-requirements-00.txt
 - Consider this draft a “technical response”
- Subscription model
 - Subscription services
 - Subscription negotiation
 - Subscription management
- Push mechanism / transport
- Leverage RFC 5277
 - Some parameter extensions proposed
 - Require interleave support to support service primitives beyond create-subscription
 - Implies one subscription per session
- Alternative transport mappings conceivable but outside scope

Subscription primitives

- Create-subscription
 - Leverage RFC 5277 <create-subscription>
 - Subscribe to datastream “push-update”
 - Built-in conceptual continuous & perpetual stream of datastore updates
 - Alternatively, custom datastreams can be optionally configured
 - Parameters:
 - Filter (e.g. subtree filter, xpath filter, allow filter extensions)
 - Subscription policy
 - Periodic
 - » Frequency
 - On-change
 - » Dampening period: minimum elapsed time between updates
 - » Delta policy: optional, specify other change conditions such as magnitude of change
 - Start and stop time (like RFC 5277)

Subscription example

```
<netconf:rpc message-id="101"
  xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
  <create-subscription
    xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <stream>push-update</stream>
    <subscription-id xmlns="urn:ietf:params:xml:ns:netconf:
      datastore-push:1.0">
      my-sub
    </subscription-id>
    <filter netconf:type="xpath"
      xmlns:ex="http://example.com/foo/1.0"
      select="/ex:foo"/>
    <period xmlns="urn:ietf:params:xml:ns:netconf:datastore-push:1.0">
      500
    </period>
  </create-subscription>
</netconf:rpc>
```

More subscription primitives

- Delete-subscription
 - New RPC <delete-subscription>
 - Avoid need to tear down session
- Modify-subscription
 - Introduce a new RPC <modify-subscription>
 - Parameters analogous to create-subscription
 - Preferable over “modify by delete and create”
- Note: “interleave” capability (RFC 5277) required to permit other NETCONF operations within a notification subscription

Subscription negotiation

- Server may reject a subscription (or modification) request
 - Implementation limitations (e.g. on-change)
 - Resource limitations (e.g. update size, frequency)
 - Indicate as part of error response
- Response may include “acceptable” parameter settings
 - Increase chance of success of subsequent subscription request
 - Avoid try-and-error
- Additional notifications to indicate whenever server cannot keep “subscription promise”
 - E.g. “subscription-suspended”, “subscription-resumed”

Push updates

- Map to notification “push-update”
 - Subscription correlator
 - Datastore-contents (filtered per NACM rules)
- Leverage <notification> element (per RFC 5277)
- Alternative transport mappings conceivable but outside scope

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <subscription-id xmlns="urn:ietf:params:xml:ns:netconf:
    datastore-push:1.0">
    my-sub
  </subscription-id>
  <eventTime>2015-03-09T19:14:56Z</eventTime>
  <datastore-contents xmlns="urn:ietf:params:xml:ns:netconf:
    datastore-push:1.0">
    <foo>
      <bar>some_string</bar>
    </foo>
  </datastore-contents>
</notification>
```


Subscription management

- YANG Data Model to reflect subscriptions
 - Monitor subscriptions that are in effect
- Optional (if-feature) YANG Data Model to configure datastreams
 - Control the “faucet” – source of the stream, not the “nozzle” (filter over a stream)
 - Stream configuration model
 - Includes subscription policy, filters, etc.
 - Specify which conditions need to be met to generate a stream record
 - Configuration using <edit-config>

Subscription Data Model

```
module: ietf-datastore-push
+--rw subscriptions
  +--ro datastore-push-subscription* [subscription-id]
    +--ro subscription-id          subscription-identifier
    +--ro subscription-status?     identityref
    +--ro stream?                  string
    +--ro start-time?              yang:date-and-time
    +--ro stop-time?               yang:date-and-time
    +--ro subtree-filter?          subtree-filter
    +--ro xpath-filter?            yang:xpath1.0
    +--ro (update-trigger)?
      +--:(periodic)
        | +--ro period?              yang:timeticks
      +--:(on-change)
        +--ro dampening-period      yang:timeticks
        +--ro (change-policy)?
          +--:(delta-policy)
            +--ro delta?              uint32
```

Stream Configuration Model

```
+--rw streams {custom-streams}?
|   +--rw stream* [stream-name]
|       +--rw stream-name          string
|       +--ro stream-status?       identityref
|       +--rw subtree-filter?      subtree-filter
|       +--rw xpath-filter?        yang:xpath1.0
|       +--rw (update-trigger)?
|           +--:(periodic)
|               | +--rw period?          yang:timeticks
|           +--:(on-change)
|               +--rw dampening-period   yang:timeticks
|               +--rw (change-policy)?
|                   +--:(delta-policy)
|                       +--rw delta?      uint32
```

Conclusion

- Datastore-Push complements polling and request capabilities provided by Netconf
- Subscription model builds on, extends RFC 5277
- Address requirements from <https://datatracker.ietf.org/doc/draft-ietf-i2rs-pub-sub-requirements/>
- Details: <https://datatracker.ietf.org/doc/draft-clemm-netconf-yang-push/>
- Seeking WG Adoption