NETCONF
Transaction ID

2022-07-25
IETF 114
Jan Lindblad <jlindebla@cisco.com>
Changes and additions to the protocol extension:

• An optional last-modified txid mechanism has been added (back)
• YANG Push functionality has been added

These additions will be explained shortly.
Improvements to the RFC draft document:

- Changed name of "versioned elements". They are now called "versioned nodes".
- Clarified txid behavior for transactions toward the Candidate datastore.
- Examples provided for the abstract mechanism level with simple message flow diagrams.
- More examples on protocol level, and with ietf-access-control as example target module instead of ietf-interfaces.
- Explicit list of XPaths to clearly state where etag or last-modified attributes may be added by clients and servers.
- Document introduction restructured to remove duplication between sections and to allow multiple (etag and last-modified) txid mechanisms.
Problem #1
get-config slow + costly for change detection

Client
---
get-config
data
get-config
data

Server
---

Compare. No change.
Problem #2
YANG Push needs echo-cancellation
Problem #3
Vulnerability to clashing changes

Client

get-config

data

get-config

data

Compare. No change.

Server

Client

get-config

data

No change. Go ahead.

Server

edit-config

data

edit-config

ok

Undetected config clash.
Optional last-modified txid mechanism
Optional last-modified txid mechanism

**txid:last-modified mechanism over NETCONF/XML**

```xml
<rpc-reply message-id="4"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns:txid="urn:ietf:params:xml:ns:netconf:txid:1.0">
  <data>
    <acls xmlns="urn:ietf:params:xml:ns:yang:ietf-access-control-list"
      txid:last-modified="2022-03-20T16:20:11.333444Z">
      <acl txid:last-modified="2022-03-20T16:20:11.333444Z">
        <name>A1</name>
        <aces txid:last-modified="2022-03-20T16:20:11.333444Z">
          <ace txid:last-modified="2022-03-20T16:20:11.333444Z">
            <name>R1</name>
            <matches>
              <ipv4>
                <protocol>udp</protocol>
              </ipv4>
            </matches>
          </ace>
        </aces>
      </acl>
    </acls>
  </data>
</rpc-reply>
```

**txid:etag mechanism over NETCONF/XML**

```xml
<rpc-reply message-id="3"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns:txid="urn:ietf:params:xml:ns:netconf:txid:1.0">
  <data>
    <acls xmlns="urn:ietf:params:xml:ns:yang:ietf-access-control-list"
      txid:etag="nc5152">
      <acl txid:etag="nc4711">
        <name>A1</name>
        <aces txid:etag="nc4711">
          <ace txid:etag="nc4711">
            <name>R1</name>
            <matches>
              <ipv4>
                <protocol>udp</protocol>
              </ipv4>
            </matches>
          </ace>
        </aces>
      </acl>
    </acls>
  </data>
</rpc-reply>
```
Optional last-modified txid mechanism

**txid:last-modified mechanism over NETCONF/XML**

```xml
<rpc-reply message-id="4"
    xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
    xmlns:txid="urn:ietf:params:xml:ns:netconf:txid:1.0">
  <data>
    <acls xmlns="urn:ietf:params:xml:ns:yang:ietf-access-control-list"
          txid:last-modified="2022-03-20T16:20:11.333444Z">
      <acl txid:last-modified="2022-03-20T16:20:11.333444Z">
        <name>R1</name>
        <matches>
          <ipv4><protocol>udp</protocol></ipv4>
        </matches>
      </acl>
    </acls>
  </data>
</rpc-reply>
```

**txid:last-modified mechanism over RESTCONF/XML**

```xml
<rpc-reply message-id="4"
    xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
    xmlns:txid="urn:ietf:params:xml:ns:netconf:txid:1.0">
  <data>
    <acls xmlns="urn:ietf:params:xml:ns:yang:ietf-access-control-list"
          txid:last-modified="2022-03-20T16:20:11.333444Z">
      <acl txid:last-modified="2022-03-20T16:20:11.333444Z">
        <name>R1</name>
        <matches>
          <ipv4><protocol>udp</protocol></ipv4>
        </matches>
      </acl>
    </acls>
  </data>
</rpc-reply>
```

Question: Which timestamp format is best?
- Keep format with 1s resolution, aligned with RESTCONF Last-Modified retrieval mechanism
- Switch to high resolution time format, more useful and aligned with NETCONF txid mechanism
YANG Push with txid
YANG Push with txid

**Push subscribe**

```
<netconf:rpc message-id="13"
  xmlns:netconf="urn:...:netconf:base:1.0">
  <establish-subscription
    xmlns="urn:...:ietf-subscribed-notifications"
    xmlns:yp="urn:...:ietf-yang-push"
    xmlns:ietf-netconf-txid-yp="urn:...:ietf-txid-yang-push">
    <yp:datastore xmlns:ds="urn:...:ietf-datastores">
      ds:running
    </yp:datastore>
    <yp:datastore-xpath-filter
      xmlns:acl="urn:...:ietf-access-control-list">
      /acl:acls
    </yp:datastore-xpath-filter>
    <yp:periodic>
      <yp:period>500</yp:period>
    </yp:periodic>
  </establish-subscription>
</netconf:rpc>
```

**Push update**

```
<notification xmlns="urn:...:notification:1.0">
  <eventTime>2022-04-04T06:00:24.16Z</eventTime>
  <push-change-update xmlns="urn:...:ietf-yang-push">
    <id>89</id>
    <datastore-changes>
      <yang-patch>
        <patch-id>0</patch-id>
        <edit txid:etag="nc8008">
          <edit-id>edit1</edit-id>
          <operation>delete</operation>
          <target xmlns:acl="urn:...:ietf-access-control-list">
            /acl:acls
          </target>
          <value>
            <acl xmlns="urn:...:ietf-access-control-list">
              <name>A1</name>
            </acl>
          </value>
        </edit>
      </yang-patch>
    </datastore-changes>
</notification>
```
Thank you
Bonus material
Simplistic, Top-Level TXID Solution

No change. Massive effort.

No change. Small effort.
Draft Solution
Simulated Results

• Real world management application running 1h in lab:
  • 569 => 378 roundtrips, down 34% (network load, delay)
  • 1002 kB => 547 kB, down 45% (network load, processing)

• No datastore locking outside `<edit-config>` until `<commit>`
• No window of vulnerability to clashing configurations