

Extensible YANG Model for YANG-Push Notifications

I-D: draft-netana-netconf-notif-envelope-02

A. Huang Feng, INSA-Lyon
P. Francois, INSA-Lyon
T. Graf, Swisscom
B. Claise, Huawei

November 10th 2024

Extensible YANG model for YANG-Push Notifications

Proposal of this I-D

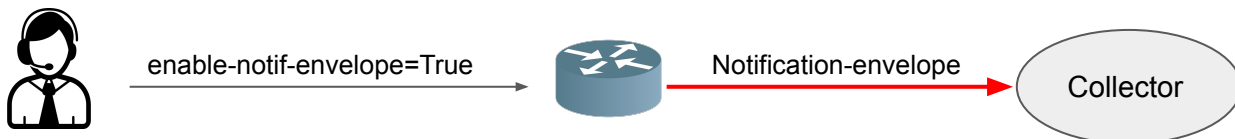
- Structure defined as a notification containing
 - event-time
 - metadata(s)
 - notification-contents

```
notifications:
  +---n envelope
    +---ro event-time          yang:date-and-time
    +---ro notification-contents? <anydata>
```

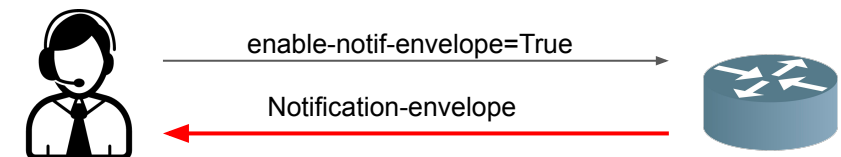
```
{
  "ietf-yp-notification:envelope": {
    "event-time": "2024-10-10T08:00:11.22Z",
    "notification-contents": {
      "ietf-yang-push:push-update": {
        "id": 1011,
        "datastore-contents": {
          "ietf-interfaces:interfaces": [
            {
              "interface": {
                "name": "eth0",
                "oper-status": "up"
              }
            }
          ]
        }
      }
    }
  }
}
```

JSON example without metadata

Configured Subscriptions



Dynamic Subscriptions



Extensible YANG model for YANG-Push Notifications

Proposal of this I-D

- YANG Notification structure for **YANG-Push Notifications** [RFC 8639/8641]
 - (1) Option to “opt-in” through a YANG-Push Subscription
 - (2) Able to discover the capability of this new header through “ietf-notification-capabilities”
 - (3) Extensible header defined in YANG 1.1
 - (4) Definition of each encoding (XML, JSON, CBOR)
 - (5) Defines the first base extensions (I-D.tgraf-netconf-notif-sequencing)

Extensible YANG model for YANG-Push Notifications

Feedback IETF 121 Dublin

- Envelope header is configurable per subscription
 - Too complex, suggested to enable it globally [Reshad, Rob]
- Let's simplify, don't allow configuring "which headers" we want
 - Suggested to let the client discover what is supported [Joe]
- The envelope need to be a sx:structure [Rob]
- Plenty of support of this I-D
 - Multiple contributors agreed to implement this I-D

Extensible YANG model for YANG-Push Notifications

Changes since -00

- (1) Envelope is enabled and disabled globally
- (2) Added Observation Timestamp extension [draft-tgraf-netconf-yang-push-observation-time]
- (3) Other changes
 - The XML namespace has been changed to “urn:ietf:params:xml:ns:yang:ietf-yp-notification”
 - Editorial changes
 - “The 'notification-contents' element SHOULD be located at the end of the notification envelope structure.”

Extensible YANG model for YANG-Push Notifications

(1) Enabling the envelope globally

- Global RPC “enable-notif-envelope”
 - This RPC is the only way to enabling/disabling the envelope
 - The node “/sn:subscriptions/enable-notification-envelope” becomes read-only
 - The enabling of the envelope MUST be configured before the creation of any dynamic subscriptions
 - If any subscriptions exist → “invalid-notification-envelope-config” error

```
module: ietf-yp-notification

augment /sn:subscriptions:
  +--ro enable-notification-envelope?  boolean
  +--ro metadata

rpcs:
  +---x enable-notif-envelope
    +---w input
      +---w enable-notification-envelope?  boolean
      +---w metadata
```

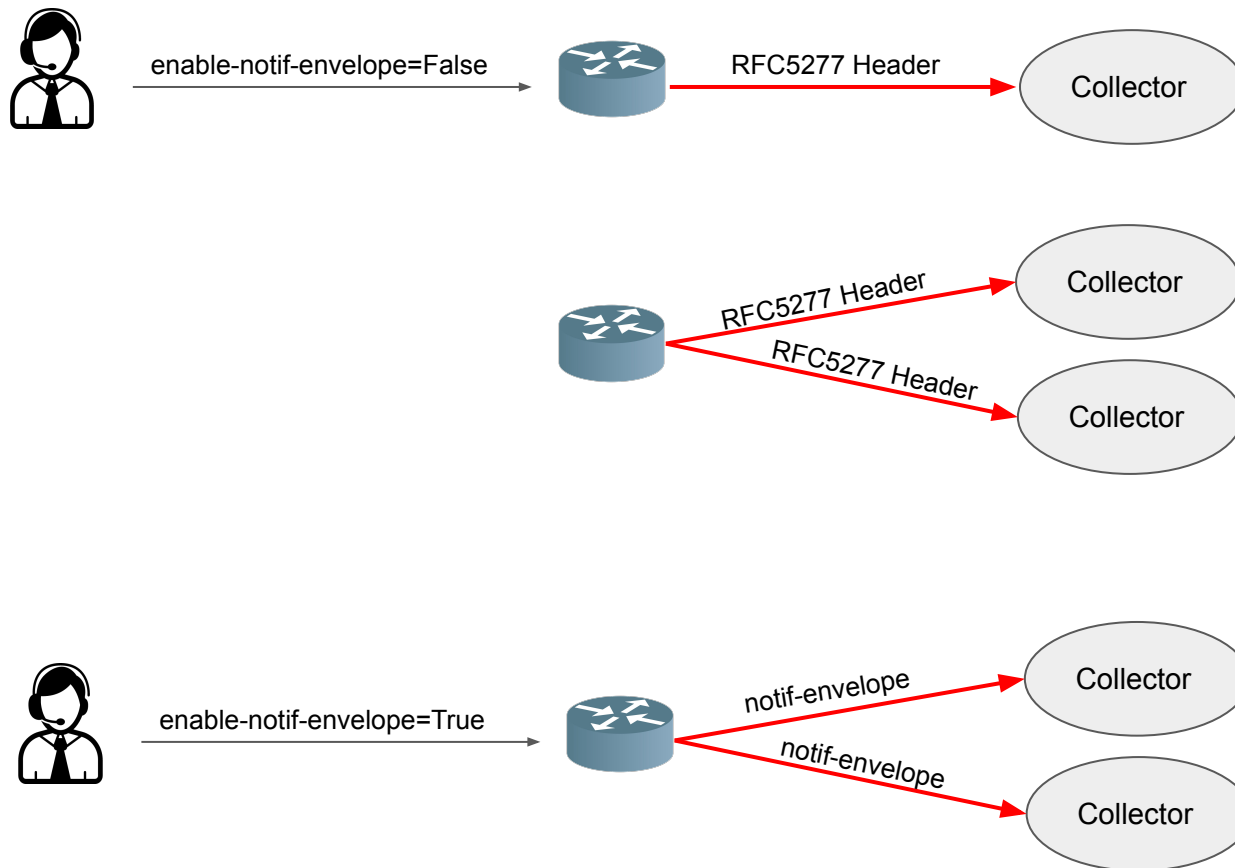
```
identity notif-envelope-error {
  description
    "Base identify for errors found while attempting to
    change configuration values during the
    'enable-notif-envelope' RPC requests.";
}

identity invalid-notification-envelope-config {
  base notif-envelope-error;
  description
    "This error is triggered and sent in the response of
    the RPC 'enable-notif-envelope' when attempting to change
    the value of the 'enable-notification-envelope' node
    while any Dynamic Subscription is active. The node
    'enable-notification-envelope' can only be changed prior to
    the creation of the Dynamic Subscription.";
}
```

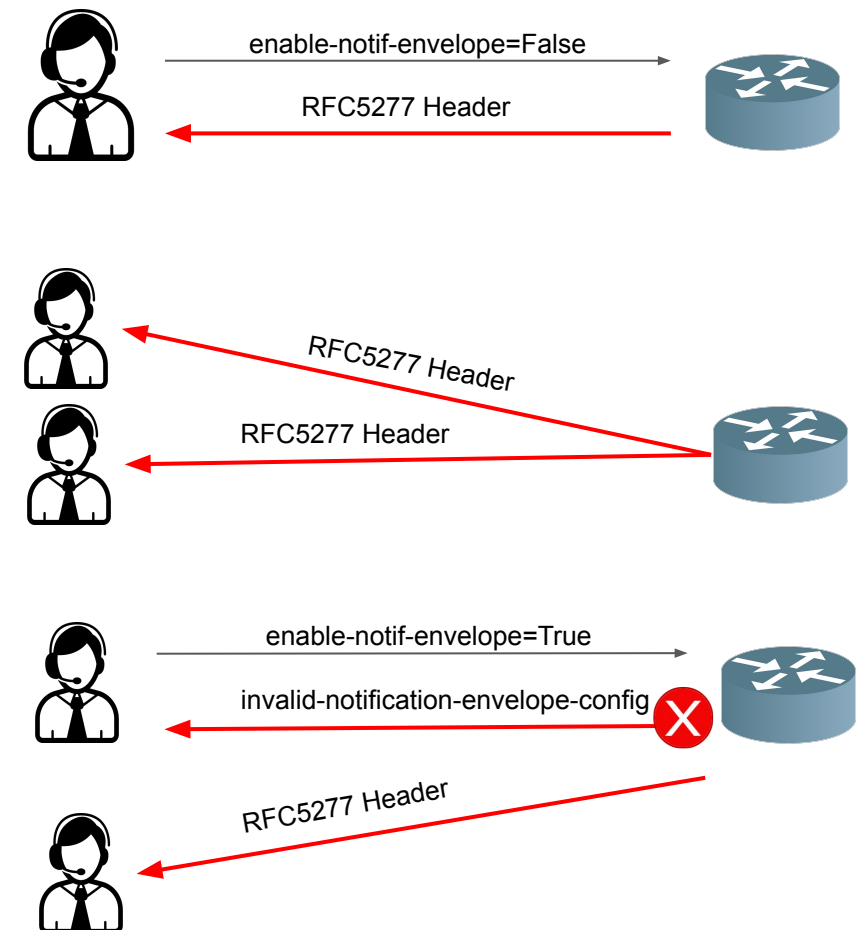
Extensible YANG model for YANG-Push Notifications

(1) Enabling the envelope globally

Configured Subscriptions



Dynamic Subscriptions



Extensible YANG model for YANG-Push Notifications

(2) Added Observation Timestamp extension

- Timestamp representing
 - The time the metrics were polled
 - The time the exported event occurred
- Extensions to YANG-Push
- Impact the following YANG-Push Notification only:
 - push-update
 - push-change-update
- Use same mechanism to get the support of this extension via “/sysc:system-capabilities/notc:subscription-capabilities”
- Based on: [draft-tgraf-netconf-yang-push-observation-time](#)

```
{
  "ietf-yp-notification:envelope": {
    "event-time": "2023-03-25T08:30:12.22Z",
    "hostname": "example-router",
    "sequence-number": 1,
    "notification-contents": {
      "ietf-yang-push:push-change-update": {
        "id": 2222,
        "ietf-yp-observation:timestamp": \
          "2023-03-25T08:30:11.22Z",
        "ietf-yp-observation:point-in-time": \
          "state-changed",
        "datastore-contents": {
          "yang-patch": {
            "patch-id": "patch_54",
            "comment": "Changing encoding to JSON and increasing \
              the period to 10 minutes",
            "edit": [
              {
                "edit-id": "id_change_1",
                "operation": "merge",
                "target": "/ietf-subscribed-notifications\subscriptions/subscription[id=2222]",
                "value": {
                  "ietf-subscribed-notifications:encoding": \
                    "ietf-subscribed-notifications:encode-json",
                  "ietf-yang-push:periodic": {
                    "period": 60000
                  }
                }
              }
            ]
          }
        }
      }
    }
  }
}
```


Extensible YANG model for YANG-Push Notifications

(3) Other minor changes

- The XML namespace has been changed to “urn:ietf:params:xml:ns:yang:ietf-yp-notification”
- Editorial changes, improve reading
- “The 'notification-contents' element SHOULD be located at the end of the notification envelope structure.”
 - Idea: *have the header located at the beginning of the message*

Extensible YANG model for YANG-Push Notifications

Discussion and open issues

- All the YANG notifications or only YANG-Push Notifications? So, far, current scope is fine (**YANG-Push**)
- Should this notification be defined as a “notification” or as a “sx:structure”? **sx:structure**
- XML namespace: which one to use?
 - urn:ietf:params:xml:ns:netconf:notification:2.0 → following RFC5277
 - **urn:ietf:params:xml:ns:yang:ietf-yp-notification** → following YANG guidelines
- Which notification and subscription extensions should be added?
 - Metadata sent by default when the envelope is enabled? **Yes, feedback IETF 121**
 - YANG versioning? [draft-ietf-netconf-yang-notifications-versioning (*currently adopted*)]; **Added**
 - Observation time? [draft-tgraf-netconf-yang-push-observation-time]; **Added**
 - Some of the extensions only impact a subset of YANG-Push notifications
 - How to deal with this? → **Current approach, extend YANG-Push header**

Extensible YANG model for YANG-Push Notifications

Discussion and open issues

- Does the changes reflect the will of the WG?
 - (1) Envelope is enabled and disabled globally
 - (2) Observation time as an extension
 - Other headers?
- Thanks Qiufang, Andy, Pierre Francois, and Reshad for the feedback.
- We will integrate this feedback on the next iterations

BACKUP

YANG model for NETCONF Event Notifications

Interim 2024-09-19 – draft-ahuang-netconf-notif-yang

- <https://datatracker.ietf.org/doc/minutes-interim-2024-netconf-02-202409191300/>
- Thorough review of draft-ahuang-netconf-notif-yang/YANG-Push/NETCONF Event Notifications
- Conclusion
 - draft-ahuang-netconf-notif-yang fixes a **gap** for YANG-Push but might be worth putting the effort on a brand new header
 - We need:
 - Bypass RFC5277, thus use YANG-Push only
 - Extensible header
 - be able to add new metadata (sequencing, versioning, others...)
 - A client should be able to “opt-in”
 - Clients that don’t support this new header should continue working seamlessly
 - The notification should be a YANG-based solution
 - Fix JSON and CBOR underspecification
 - including CBOR-SID allocation

YANG model for NETCONF Event Notifications

Problem statement - (draft-ahuang-netconf-notif-yang)

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2022-09-02T10:59:55.32Z</eventTime>
  <push-update xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-push">
    <id>101</id>
    <datastore-contents>
      <interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
        <interface>
          <name>eth0</name>
          <oper-status>up</oper-status>
        </interface>
      </interfaces>
    </datastore-contents>
  </push-update>
</notification>
```

```
{
  "ietf-notification:notification": {
    "eventTime": "2017-10-25T08:00:11.22Z",
    "ietf-yang-push:push-update": {
      "id": 1011,
      "datastore-contents": {
        "ietf-interfaces:interfaces": [
          "interface": {
            "name": "eth0",
            "oper-status": "up"
          }
        ]
      }
    }
  }
}
```

RFC 5277 - Netconf Event Notifications

RFC 8641 - YANG Push

YANG encodings:

- RFC 7950 - YANG XML
- RFC 7951 - YANG JSON
- RFC 9254 - YANG CBOR

Implementation Issues:

(1) YANG module not defined

(2) Non-existing Normative text defining this header

Thanks Andy for confirming the approach was not correct.

Extensible YANG model for YANG-Push Notifications

Proposal (comments)

- As requested
 - Scoped to YANG-Push (both dynamic and configured subscriptions)
 - Can be implemented with NETCONF and RESTCONF
 - Use a “notification” statement rather than a “sx:structure”
 - Given that it’s intended for YANG-Push, the following notifications are impacted:
 - push-update; push-change-update
 - subscription-started; subscription-modified; subscription-terminated
 - subscription-suspended; subscription-resumed; subscription-completed
 - replay-completed

Extensible YANG model for YANG-Push Notifications

(1) Option to “opt-in” through a YANG-Push Subscription

- Configuration on Globally on the server via the RPC “enable-notif-envelope”

```
rpcs:  
  +---x enable-notif-envelope  
    +---w input  
      +---w enable-notification-envelope?  boolean  
      +---w metadata
```

Currently
Default=False



enable-notification-envelope=false



Old Header as RFC5277

Collector



enable-notification-envelope=true



New envelope Header

Collector

Extensible YANG model for YANG-Push Notifications

(2) Able to discover the capability of this new header

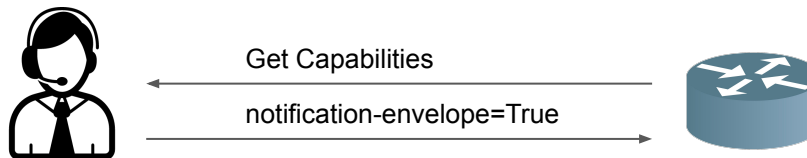
- Augmentation on notification capabilities (RFC9196)

module: ietf-yp-notification

```
augment /sn:subscriptions:
  +--ro enable-notification-envelope?  boolean
  +--ro metadata
augment /sysc:system-capabilities/notc:subscription-capabilities:
  +--ro notification-metadata
  +--ro notification-envelope?  boolean
  +--ro metadata
  +--ro hostname-sequence-number?  boolean
```

module: ietf-yp-observation

```
augment /yp:push-update:
  +--ro timestamp?      yang:date-and-time
  +--ro point-in-time?  enumeration
augment /yp:push-change-update:
  +--ro timestamp?      yang:date-and-time
  +--ro point-in-time?  enumeration
augment /sysc:system-capabilities/notc:subscription-capabilities:
  +--ro yang-push-observation-supported?
    inotifseq:notification-support
    {yang-push-observation-timestamp}?
```



Extensible YANG model for YANG-Push Notifications

(3) Extensible header defined in YANG

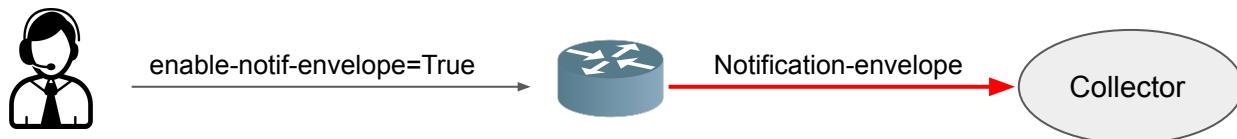
- Structure defined as a notification containing
 - event-time
 - metadata(s)
 - notification-contents

```
notifications:
  +---n envelope
    +---ro event-time          yang:date-and-time
    +---ro notification-contents? <anydata>
```

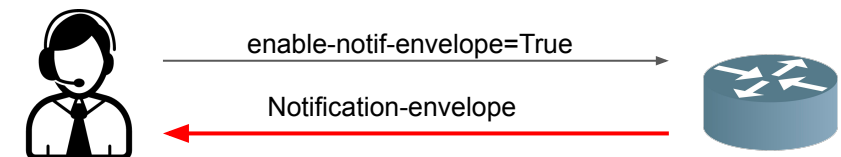
```
{
  "ietf-yp-notification:envelope": {
    "event-time": "2024-10-10T08:00:11.22Z",
    "notification-contents": {
      "ietf-yang-push:push-update": {
        "id": 1011,
        "datastore-contents": {
          "ietf-interfaces:interfaces": [
            {
              "interface": {
                "name": "eth0",
                "oper-status": "up"
              }
            }
          ]
        }
      }
    }
  }
}
```

JSON example without metadata

Configured Subscriptions



Dynamic Subscriptions



Extensible YANG model for YANG-Push Notifications

(4) Definition of each encoding (XML, JSON, CBOR)

- Explicit definition of the content of the “envelope” (**solving gap for JSON and CBOR**)
 - Definition of the namespace (urn:ietf:params:xml:ns:netconf:notification:2.0)
 - Mandatory event-time node
 - Mandatory notification-contents node
 - Metadata present when configured

A YANG notification encoded in XML is structured as a root "envelope" container. The namespace of this container is the namespace defined in the YANG module "ietf-yp-notification":

`urn:ietf:params:xml:ns:netconf:notification:2.0`

Two mandatory child nodes within the "envelope" container are expected, representing the event time and the notification payload. The "event-time" node is defined within the same XML namespace as the "envelope" container. The "event-time" node MUST be compliant with [\[RFC3339\]](#). Other metadata defined within the YANG module defined in [Section 5](#) MUST use the same XML namespace. See [Section 3.4](#) for more details.

Extensible YANG model for YANG-Push Notifications

(5) Extensions for hostname and sequence-number

- Definition of hostname and sequence-number extensions (draft-tgraf-netconf-notif-sequencing)
 - **Present** by default when the envelope is enabled
 - Discovery of support of this header through RFC9196

```
structure envelope:
  +-- event-time          yang:date-and-time
  +-- hostname?           inet:host
  +-- sequence-number?    yang:counter32
  +-- notification-contents? <anydata>
```

```
{
  "ietf-yp-notification:envelope": {
    "event-time": "2023-03-25T08:30:11.22Z",
    "hostname": "example-router",
    "sequence-number": 1,
    "notification-contents": {
      "ietf-yang-push:push-update": {
        "id": 6666,
        "datastore-contents": {
          "ietf-interfaces:interfaces": [
            {
              "interface": {
                "name": "eth0",
                "type": "iana-if-type:ethernetCsmacd",
                "oper-status": "up",
                "mtu": 1500
              }
            }
          ]
        }
      }
    }
  }
}
```