Internet Engineering Task Force (IETF)
Request for Comments: 8640
Category: Standards Track
ISSN: 2070-1721

E. Voit
Cisco Systems
A. Clemm
Futurewei
A. Gonzalez Prieto
Microsoft
E. Nilsen-Nygaard
A. Tripathy
Cisco Systems

September 2019

Dynamic Subscription to YANG Events and Datastores over NETCONF

Abstract

This document provides a Network Configuration Protocol (NETCONF) binding to the dynamic subscription capability of both subscribed notifications and YANG-Push.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8640.

Copyright Notice

Table of Contents

1. Introduction

This document specifies the binding of a stream of events that form part of a dynamic subscription to the Network Configuration Protocol (NETCONF) [RFC6241]. Dynamic subscriptions are defined in [RFC8639]. In addition, as [RFC8641] is itself built upon [RFC8639], this document enables a NETCONF client to request via a dynamic subscription, and receive, updates from a YANG datastore located on a NETCONF server.

This document assumes that the reader is familiar with the terminology and concepts defined in [RFC8639].

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The following terms are defined in [RFC8639]: dynamic subscription, event stream, notification message, publisher, receiver, subscriber, and subscription. This document does not define any additional terms.

3. Compatibility with <create-subscription> as Defined in RFC 5277

A publisher is allowed to concurrently support dynamic subscription RPCs as defined in [RFC8639] at the same time as the <create-subscription> RPC defined in [RFC5277]. However, a single NETCONF transport session MUST NOT support both this specification and a subscription established by the <create-subscription> RPC defined in [RFC5277]. To protect against any attempts to use a single NETCONF transport session in this way:

- A solution MUST reply with the <rpc-error> element [RFC6241] containing the "error-tag" value of "operation-not-supported" if a <create-subscription> RPC is received on a NETCONF session where an established subscription per [RFC8639] exists.

- A solution MUST reply with the <rpc-error> element [RFC6241] containing the "error-tag" value of "operation-not-supported" if an "establish-subscription" request has been received on a NETCONF session where the <create-subscription> RPC [RFC5277] has successfully created a subscription.

If a publisher supports this specification but not subscriptions via [RFC5277], the publisher MUST NOT advertise "urn:ietf:params:netconf:capability:notification:1.0".

4. Mandatory XML, Event Stream, and Datastore Support

The "encode-xml" feature of [RFC8639] MUST be supported. This indicates that XML is a valid encoding for RPCs, state change notifications, and subscribed content.

A NETCONF publisher supporting event stream subscription via [RFC8639] MUST support the "NETCONF" event stream identified in that document.

5. NETCONF Connectivity and Dynamic Subscriptions

Management of dynamic subscriptions occurs via RPCs as defined in [RFC8641] and [RFC8639]. For a dynamic subscription, if the NETCONF session involved with the "establish-subscription" terminates, the subscription MUST be terminated.

For a dynamic subscription, any "modify-subscription", "delete-subscription", or "resync-subscription" RPCs MUST be sent using the same NETCONF session upon which the referenced subscription was established.

6. Notification Messages

Notification messages transported over NETCONF MUST be encoded in a <notification> message as defined in [RFC5277], Section 4. And per the <eventTime> object definition provided in [RFC5277], <eventTime> is populated with the event occurrence time.

For dynamic subscriptions, all notification messages MUST use the NETCONF transport session used by the "establish-subscription" RPC.

7. Dynamic Subscriptions and RPC Error Responses

When an RPC error occurs as defined in [RFC8639], Section 2.4.6 and [RFC8641], Appendix A, the NETCONF RPC reply MUST include an <rpc-error> element per [RFC6241] with the error information populated as follows:

- An "error-type" node of "application".
- An "error-tag" node, where the value is a string that corresponds to an identity associated with the error. For the mechanisms specified in this document, this "error-tag" will correspond to the error identities in either (1) [RFC8639], Section 2.4.6, for general subscription errors:

error identity	uses error-tag
dscp-unavailable	invalid-value
encoding-unsupported	invalid-value
filter-unsupported	invalid-value
insufficient-resources	resource-denied
no-such-subscription	invalid-value
replay-unsupported	operation-not-supported

or (2) [RFC8641], Appendix A.1, for subscription errors specific to YANG datastores:

error identity	uses error-tag
cant-exclude	operation-not-supported
datastore-not-subscribable	invalid-value
no-such-subscription-resync	invalid-value
on-change-unsupported	operation-not-supported
on-change-sync-unsupported	operation-not-supported
period-unsupported	invalid-value
update-too-big	too-big
sync-too-big	too-big
unchanging-selection	operation-failed

- An "error-severity" of "error" (this MAY be included).

- An "error-app-tag" node, where the value is a string that corresponds to an identity associated with the error, as defined in [RFC8639], Section 2.4.6 for general subscriptions and [RFC8641], Appendix A.1 for datastore subscriptions. The specific identity to use depends on the RPC for which the error occurred. Each error identity will be inserted as the "error-app-tag" following the form <modulename>:<identityname>. An example of such a valid encoding would be "ietf-subscribed-notifications:no-such-subscription". Viable errors for different RPCs are as follows:

RPC	has base identity
establish-subscription	establish-subscription-error
modify-subscription	modify-subscription-error
delete-subscription	delete-subscription-error
kill-subscription	delete-subscription-error
resync-subscription	resync-subscription-error

- In the case of error responses to an "establish-subscription" or "modify-subscription" request, there is the option of including an "error-info" node. This node may contain XML-encoded data with hints for parameter settings that might lead to successful RPC requests in the future. The yang-data structures from [RFC8639] and [RFC8641] that may be returned are as follows:

establish-subscription	returns hints in yang-data structure
target: event stream	establish-subscription-stream-error-info
target: datastore	establish-subscription-datastore-error-info

modify-subscription	returns hints in yang-data structure
target: event stream	modify-subscription-stream-error-info
target: datastore	modify-subscription-datastore-error-info

  The yang-data included in "error-info" SHOULD NOT include the optional leaf "reason", as such a leaf would be redundant with information that is already placed in the "error-app-tag".

In the case of an RPC error resulting from a "delete-subscription", "kill-subscription", or "resync-subscription" request, no "error-info" needs to be included, as the "subscription-id" is the only RPC input parameter and no hints regarding this RPC input parameter need to be provided.

8. Security Considerations

This document does not introduce additional security considerations for dynamic subscriptions beyond those discussed in [RFC8639]. But there is one consideration worthy of more refinement based on the connection-oriented nature of NETCONF. Specifically, if a buggy or compromised NETCONF subscriber sends a number of "establish-subscription" requests, then these subscriptions accumulate and may use up system resources. In such a situation, subscriptions MAY be terminated by terminating the underlying NETCONF session. The publisher MAY also suspend or terminate a subset of the active subscriptions on that NETCONF session in order to reclaim resources and preserve normal operation for the other subscriptions.

9. IANA Considerations

This document has no IANA actions.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5277] Chisholm, S. and H. Trevino, "NETCONF Event Notifications", RFC 5277, DOI 10.17487/RFC5277, July 2008, <https://www.rfc-editor.org/info/rfc5277>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8639] Voit, E., Clemm, A., Gonzalez Prieto, A., Nilsen-Nygaard, E., and A. Tripathy, "Subscription to YANG Notifications", RFC 8639, DOI 10.17487/RFC8639, September 2019, <https://www.rfc-editor.org/info/rfc8639>.

[RFC8641] Clemm, A. and E. Voit, "Subscription to YANG Notifications for Datastore Updates", RFC 8641, DOI 10.17487/RFC8641, September 2019, <https://www.rfc-editor.org/info/rfc8641>.

[W3C.REC-xml-20081126] Bray, T., Paoli, J., Sperberg-McQueen, M., Maler, E., and F. Yergeau, "Extensible Markup Language (XML) 1.0 (Fifth Edition)", World Wide Web Consortium Recommendation REC-xml-20081126, November 2008, <https://www.w3.org/TR/2008/REC-xml-20081126>.

10.2. Informative References

[RFC8347] Liu, X., Ed., Kyparlis, A., Parikh, R., Lindem, A., and M. Zhang, "A YANG Data Model for the Virtual Router Redundancy Protocol (VRRP)", RFC 8347, DOI 10.17487/RFC8347, March 2018, <https://www.rfc-editor.org/info/rfc8347>.

[XPATH] Clark, J. and S. DeRose, "XML Path Language (XPath) Version 1.0", November 1999, <https://www.w3.org/TR/1999/REC-xpath-19991116>.

Appendix A. Examples

This appendix is non-normative. Additionally, the subscription "id" values of 22, 23, 39, and 99 used below are just examples. In production, the actual values of "id" might not be small integers.

A.1. Event Stream Discovery

As defined in [RFC8639], an event stream exposes a continuous set of events available for subscription. A NETCONF client can retrieve the list of available event streams from a NETCONF publisher using the <get> operation against the top-level "streams" container defined in [RFC8639], Section 3.1.

The following XML example [W3C.REC-xml-20081126] illustrates the retrieval of the list of available event streams:

```
<rpc message-id="101"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter type="subtree">
      <streams
     xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications"/>
    </filter>
  </get>
</rpc>
```

Figure 1: <get> Request for Retrieval of Event Streams

After such a request, the NETCONF publisher returns a list of available event streams as well as additional information that might exist in the container.

A.2. Dynamic Subscriptions

A.2.1. Establishing Dynamic Subscriptions

Figure 2 shows two successful "establish-subscription" RPC requests as per [RFC8639]. The first request is given a subscription "id" of 22, and the second is given an "id" of 23.

```
              +------------+                 +-----------+
              | Subscriber |                 | Publisher |
              +------------+                 +-----------+
                    |                              |
                    |    Capability Exchange       |
                    |<---------------------------->|
                    |                              |
                    |                              |
                    |    establish-subscription    |
                    |----------------------------->|  (a)
                    | RPC Reply: OK, id = 22       |
                    |<-----------------------------|  (b)
                    |                              |
                    | notification message (for 22)|
                    |<-----------------------------|
                    |                              |
                    |                              |
                    |    establish-subscription    |
                    |----------------------------->|
                    | notification message (for 22)|
                    |<-----------------------------|
                    | RPC Reply: OK, id = 23       |
                    |<-----------------------------|
                    |                              |
                    |                              |
                    | notification message (for 22)|
                    |<-----------------------------|
                    | notification message (for 23)|
                    |<-----------------------------|
                    |                              |
```

Figure 2: Multiple Subscriptions over a NETCONF Session

To provide examples of the information being transported, example messages for interactions (a) and (b) in Figure 2 are detailed below (Figures 3 and 4):

```
<rpc message-id="102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <establish-subscription
      xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
    <stream-xpath-filter xmlns:ex="https://example.com/events">
      /ex:foo/
    </stream-xpath-filter>
    <stream>NETCONF</stream>
    <dscp>10</dscp>
  </establish-subscription>
</rpc>
```

Figure 3: "establish-subscription" Request (a)

As the NETCONF publisher was able to fully satisfy the request (a), the publisher sends the subscription "id" of the accepted subscription in its reply message (b):

```
  <rpc-reply message-id="102"
    xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <id
      xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
      22
    </id>
  </rpc-reply>
```

Figure 4: A Successful "establish-subscription" (b)

If the NETCONF publisher had not been able to fully satisfy the request or the subscriber has no authorization to establish the subscription, the publisher would have sent an RPC error response. For instance, if the "dscp" value of 10 asserted by the subscriber in Figure 3 proved unacceptable, the publisher may have returned:

```
<rpc-reply message-id="102"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <rpc-error>
   <error-type>application</error-type>
   <error-tag>invalid-value</error-tag>
   <error-severity>error</error-severity>
   <error-app-tag>
     ietf-subscribed-notifications:dscp-unavailable
   </error-app-tag>
  </rpc-error>
</rpc-reply>
```

Figure 5: An Unsuccessful "establish-subscription"

The subscriber can use this information in future attempts to establish a subscription.

A.2.2. Modifying Dynamic Subscriptions

An existing subscription may be modified. The following exchange shows a negotiation of such a modification via several exchanges between a subscriber and a publisher. This negotiation consists of a failed RPC modification request/response followed by a successful one.

```
              +------------+                 +-----------+
              | Subscriber |                 | Publisher |
              +------------+                 +-----------+
                    |                               |
                    | notification message (for 23) |
                    |<------------------------------|
                    |                               |
                    | modify-subscription (id = 23) |
                    |------------------------------>|   (c)
                    | RPC error (with hint)         |
                    |<------------------------------|   (d)
                    |                               |
                    | modify-subscription (id = 23) |
                    |------------------------------>|
                    | RPC Reply: OK                 |
                    |<------------------------------|
                    |                               |
                    | notification message (for 23) |
                    |<------------------------------|
                    |                               |
```

Figure 6: Interaction Model for Successful Subscription Modification

If the subscription being modified in Figure 6 is a datastore subscription as per [RFC8641], the modification request made in (c) may look like that shown in Figure 7. As can be seen, the modifications being attempted are the application of a new XPath filter as well as the setting of a new periodic time interval.

```
<rpc message-id="303"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <modify-subscription
       xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications"
       xmlns:yp="urn:ietf:params:xml:ns:yang:ietf-yang-push">
    <id>23</id>
    <yp:datastore-xpath-filter xmlns:ex="https://example.com/datastore">
        /ex:foo/ex:bar
    </yp:datastore-xpath-filter>
    <yp:periodic>
      <yp:period>500</yp:period>
    </yp:periodic>
  </modify-subscription>
</rpc>
```

Figure 7: Subscription Modification Request (c)

If the NETCONF publisher can satisfy both changes, the publisher sends a positive result for the RPC. If the NETCONF publisher cannot satisfy either of the proposed changes, the publisher sends an RPC error response (d). Figure 8 shows an example RPC error response for (d) that includes a hint. This hint is an alternative time period value that might have resulted in a successful modification:

```
<rpc-reply message-id="303"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <rpc-error>
    <error-type>application</error-type>
    <error-tag>invalid-value</error-tag>
    <error-severity>error</error-severity>
    <error-app-tag>
        ietf-yang-push:period-unsupported
    </error-app-tag>
    <error-info>
      <modify-subscription-datastore-error-info
          xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-push">
        <period-hint>
            3000
        </period-hint>
      </modify-subscription-datastore-error-info>
    </error-info>
  </rpc-error>
</rpc-reply>
```

Figure 8: "modify-subscription" Failure with Hint (d)

A.2.3. Deleting Dynamic Subscriptions

Figure 9 demonstrates the deletion of a subscription. This subscription may have been to either a stream or a datastore.

```
<rpc message-id="103"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <delete-subscription
    xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
    <id>22</id>
  </delete-subscription>
</rpc>
```

Figure 9: "delete-subscription"

If the NETCONF publisher can satisfy the request, the publisher returns a reply indicating success.

If the NETCONF publisher cannot satisfy the request, the publisher sends an <rpc-error> element indicating that the modification didn't work. Figure 10 shows a valid response for an existing valid subscription "id", but that subscription "id" was created on a different NETCONF transport session:

```
<rpc-reply message-id="103"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <rpc-error>
    <error-type>application</error-type>
    <error-tag>invalid-value</error-tag>
    <error-severity>error</error-severity>
    <error-app-tag>
        ietf-subscribed-notifications:no-such-subscription
    </error-app-tag>
  </rpc-error>
</rpc-reply>
```

Figure 10: An Unsuccessful "delete-subscription"

A.3. Subscription State Notifications

A publisher will send subscription state notifications for dynamic subscriptions according to the definitions in [RFC8639].

A.3.1. "subscription-modified"

As per Section 2.7.2 of [RFC8639], a "subscription-modified" might be sent over NETCONF if the definition of a configured filter changes. A subscription state notification encoded in XML would look like:

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2007-09-01T10:00:00Z</eventTime>
  <subscription-modified
      xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
    <id>39</id>
    <stream-xpath-filter xmlns:ex="https://example.com/events">
      /ex:foo
    </stream-xpath-filter>
    <stream>NETCONF</stream>
  </subscription-modified>
</notification>
```

Figure 11: "subscription-modified" Subscription State Notification

A.3.2. "subscription-resumed" and "replay-complete"

A "subscription-resumed" would look like:

```
<notification
  xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2007-09-01T10:00:00Z</eventTime>
  <subscription-resumed
    xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
    <id>39</id>
  </subscription-resumed>
</notification>
```

Figure 12: "subscription-resumed" Notification

The "replay-complete" is virtually identical, with "subscription-resumed" simply being replaced by "replay-complete".

A.3.3. "subscription-terminated" and "subscription-suspended"

A "subscription-terminated" would look like:

```
<notification
  xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2007-09-01T10:00:00Z</eventTime>
  <subscription-terminated
    xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
    <id>39</id>
    <reason>
       suspension-timeout
    </reason>
  </subscription-terminated>
</notification>
```

Figure 13: "subscription-terminated" Subscription State Notification

The "subscription-suspended" is virtually identical, with "subscription-terminated" simply being replaced by "subscription-suspended".

A.4. Filter Examples

This appendix provides examples that illustrate both XPath and subtree methods of filtering event record contents. The examples are based on the YANG notification "vrrp-protocol-error-event" as defined per the ietf-vrrp YANG data model in [RFC8347]. Event records based on this specification that are generated by the publisher might appear as:

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2018-09-14T08:22:33.44Z</eventTime>
  <vrrp-protocol-error-event
       xmlns="urn:ietf:params:xml:ns:yang:ietf-vrrp">
     <protocol-error-reason>checksum-error</protocol-error-reason>
  </vrrp-protocol-error-event>
</notification>
```

Figure 14: Example VRRP Notification per RFC 8347

Suppose that a subscriber wanted to establish a subscription that only passes instances of event records where there is a "checksum-error" as part of a Virtual Router Redundancy Protocol (VRRP) protocol event. Also, assume that the publisher places such event records into the NETCONF stream. To get a continuous series of matching event records, the subscriber might request the application of an XPath filter against the NETCONF stream. An "establish-subscription" RPC to meet this objective might be:

```
<rpc message-id="601" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <establish-subscription
    xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
    <stream>NETCONF</stream>
    <stream-xpath-filter xmlns="urn:ietf:params:xml:ns:yang:ietf-vrrp">
      /vrrp-protocol-error-event[
         vrrp:protocol-error-reason="vrrp:checksum-error"]
    </stream-xpath-filter>
  </establish-subscription>
</rpc>
```

Figure 15: Establishing a Subscription Error Reason via XPath

For more examples of XPath filters, see [XPATH].

Suppose that the "establish-subscription" in Figure 15 was accepted. And suppose that a subscriber decided later on that they wanted to broaden this subscription to cover all VRRP protocol events (i.e., not just those with a "checksum-error"). The subscriber might attempt to modify the subscription in a way that replaces the XPath filter with a subtree filter that sends all VRRP protocol events to a subscriber. Such a "modify-subscription" RPC might look like:

```
<rpc message-id="602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <modify-subscription
     xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
    <id>99</id>
    <stream-subtree-filter>
     <vrrp-protocol-error-event
            xmlns="urn:ietf:params:xml:ns:yang:ietf-vrrp"/>
    </stream-subtree-filter>
  </modify-subscription>
</rpc>
```

Figure 16: Example "modify-subscription" RPC

For more examples of subtree filters, see [RFC6241], Section 6.4.

Acknowledgments

We wish to acknowledge the helpful contributions, comments, and suggestions that were received from Andy Bierman, Yan Gang, Sharon Chisholm, Hector Trevino, Peipei Guo, Susan Hares, Tim Jenkins, Balazs Lengyel, Martin Bjorklund, Mahesh Jethanandani, Kent Watsen, Qin Wu, and Guangying Zheng.

Authors' Addresses

Eric Voit
Cisco Systems

Email: evoit@cisco.com

Alexander Clemm
Futurewei

Email: ludwig@clemm.org

Alberto Gonzalez Prieto
Microsoft

Email: alberto.gonzalez@microsoft.com

Einar Nilsen-Nygaard
Cisco Systems

Email: einarnn@cisco.com

Ambika Prasad Tripathy
Cisco Systems

Email: ambtripa@cisco.com