NMOP
Internet-Draft
Intended status: Standards Track
Expires: 4 January 2026

A. Elhassany
T. Graf
Swisscom

3 July 2025

Validating anydata in YANG Library context
draft-netana-nmop-yang-anydata-validation-00

Abstract

This document describes a method to use YANG RFC 8525 and standard YANG validation rules in RFC 7950 to validate YANG data nodes that are children of an "anydata" data node.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 4 January 2026.

Table of Contents

1. Introduction

Section 7.10 of [RFC7950] defines the "anydata" statement to represent an unknown set of YANG nodes for which the data model is not known at module design time. However, Section 7.10 of [RFC7950] left the verification of the "anydata" tree open to become known through protocol signaling or other means. Several IETF models, e.g., [RFC8526], [RFC9144], [RFC8639], [RFC8641], and [RFC8040], use "anydata" in their definitions. Current YANG implementations accept syntactically valid YANG data nodes as children of an "anydata" node but do not check the data type of these data nodes against a YANG schema. This creates a real problem for any YANG data consumer when validating YANG data against YANG data tree. For instance, a YANG Message Broker Consumer described in Section 4.6 of [I-D.ietf-nmop-yang-message-broker-integration] is not able to fully validate the received messages published by the network nodes.

YANG Schema Mount [RFC8528] allows mounting complete data models at implementation and run time. While powerful, schema mount cannot address use cases where the user selects an arbitrary subset of an instantiated data tree, such as [RFC8641]. A current proposed approach, YANG Full Include [I-D.jouqui-netmod-yang-full-include], complements YANG Schema Mount and applies at design time, yet cannot address dynamic filtering of an instantiated YANG data tree.

This document propeses using the [RFC8525] to define the context in which anydata trees are validated. This would require the YANG tooling to implement additional flags that enables validating "anydata" subtrees in the context of a YANG Library.

1.1. Requirements Language

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.

2. Terminology

This document uses the terminology defined in YANG [RFC7950] for schema node and schema tree but refines data node and data tree to be more precise.

* data node: A node in the schema tree that can be instantiated in a data tree. One of container, leaf, leaf-list, list, anydata, and anyxml. This document does not change how YANG handles anyxml data nodes.

* instantiated data node: an instantiated instance of a data node that contains before fully qualified name (module namespace + identifier) for the data node and the data modeled within YANG.

* instantiated data tree: is what [RFC7950] defines as "data tree". Adding the term "instantiated" precisely indicates that this tree is an instance of specific data modeled with YANG.

* data tree: a tree of data nodes (with no values).

3. Instantiated data node schema lookup

This document builds on the fact that when a YANG validator examines a node in an instantiated data tree, it can find the corresponding data node in a YANG schema. For the existing YANG encodings, the following rules are defined to encode instantiated data nodes:

* In YANG XML encoding [RFC7950], The element’s local name is the data node identifier, and its namespace is the module’s XML namespace.

* In JSON encoding [RFC7951], each object member must be identical to the corresponding YANG data node identifier or namespace-qualified - the data node identifier is prefixed with the name of the module in which the data node is defined, separated from the data node identifier by the colon character (":").

* In CBOR [RFC9254] encoding, node should include information that would allow each node to be - identified in a stateless way, for instance, the SID number associated with the node, the SID delta from another SID in the application payload, the namespace-qualified name, or the instance-identifier.

Given the encoding rules that maintain complete information to identify the corresponding data node for each instantiated data node, the YANG validator can easily find the schema for the data node in the YANG Library.

4. Validating "anydata" Data Tree

This document introduces two new YANG validation options: anydata-complete and anydata-candidate. These two options align with Section 8.3.3 of [RFC7950], such that the complete validation validates the contents of the anydata subtree, which MUST obey all validation rules defined in the corresponding schema in the YANG Library. The candidate does not apply the constraint checks.

5. Implementation Status

Note to the RFC-Editor: Please remove this section before publishing.

anydata-candidate validation is implemented for libyang and avaiable at https://github.com/ahassany/libyang/tree/anydata-strict-parsing

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

TBD

8. References

8.1. 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>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC7951] Lhotka, L., "JSON Encoding of Data Modeled with YANG", RFC 7951, DOI 10.17487/RFC7951, August 2016, <https://www.rfc-editor.org/info/rfc7951>.

[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>.

[RFC8525] Bierman, A., Bjorklund, M., Schoenwaelder, J., Watsen, K., and R. Wilton, "YANG Library", RFC 8525, DOI 10.17487/RFC8525, March 2019, <https://www.rfc-editor.org/info/rfc8525>.

[RFC9254] Veillette, M., Ed., Petrov, I., Ed., Pelov, A., Bormann, C., and M. Richardson, "Encoding of Data Modeled with YANG in the Concise Binary Object Representation (CBOR)", RFC 9254, DOI 10.17487/RFC9254, July 2022, <https://www.rfc-editor.org/info/rfc9254>.

8.2. Informative References

[I-D.ietf-nmop-yang-message-broker-integration] Graf, T. and A. Elhassany, "An Architecture for YANG-Push to Message Broker Integration", Work in Progress, Internet-Draft, draft-ietf-nmop-yang-message-broker-integration-07, 3 March 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-nmop-yang-message-broker-integration-07>.

[I-D.jouqui-netmod-yang-full-include] Quilbeuf, J., Claise, B., and T. Joubert, "YANG Full Embed", Work in Progress, Internet-Draft, draft-jouqui-netmod-yang-full-include-02, 5 July 2024, <https://datatracker.ietf.org/doc/html/draft-jouqui-netmod-yang-full-include-02>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8526] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "NETCONF Extensions to Support the Network Management Datastore Architecture", RFC 8526, DOI 10.17487/RFC8526, March 2019, <https://www.rfc-editor.org/info/rfc8526>.

[RFC8528] Bjorklund, M. and L. Lhotka, "YANG Schema Mount", RFC 8528, DOI 10.17487/RFC8528, March 2019, <https://www.rfc-editor.org/info/rfc8528>.

[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>.

[RFC9144] Clemm, A., Qu, Y., Tantsura, J., and A. Bierman, "Comparison of Network Management Datastore Architecture (NMDA) Datastores", RFC 9144, DOI 10.17487/RFC9144, December 2021, <https://www.rfc-editor.org/info/rfc9144>.

Acknowledgements

The authors would like to thank Jean Quilbeuf, Benoit Claise, and Alex Huang Feng for their review and valuable comments.

Authors' Addresses

Ahmed Elhassany
Swisscom
Binzring 17
CH- Zurich 8045
Switzerland
Email: ahmed.elhassany@swisscom.com

Thomas Graf
Swisscom
Binzring 17
CH-8045 Zurich
Switzerland
Email: thomas.graf@swisscom.com