Network Working Group
Internet-Draft
Intended status: Informational
Expires: September 18, 2020

R. Wilton, Ed.
Cisco Systems, Inc.
March 17, 2020

YANG Versioning Solution Overview
draft-ietf-netmod-yang-solutions-00

Abstract

This document gives an overview of the different documents that comprise a full solution to the YANG versioning requirements document. The purpose of this document is to help readers understand how the discrete parts of the YANG versioning solution fit together during working group development of the solution documents.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

[I-D.ietf-netmod-yang-versioning-reqs] documents the requirements for any solution to the YANG [RFC7950] versioning problem. In particular, chapter 5 lists the formal requirements that a solution requires.

The complete solution to all of the YANG versioning requirements is comprised of five documents, each addressing different aspects of the solution. These documents are:

1. [I-D.verdt-netmod-yang-module-versioning]

2. [I-D.verdt-netmod-yang-semver]

3. [I-D.rwilton-netmod-yang-packages]

4. [I-D.wilton-netmod-yang-ver-selection]

5. [I-D.verdt-netmod-yang-schema-comparison]

The aim of this document is to help readers understand how these different solution documents fit together, and also which documents contribute solutions that address particular individual requirements.

Open issues, across all of the solution documents are tracked at <https://github.com/netmod-wg/yang-ver-dt/issues>.

2. Solution Documents

2.1. Updated YANG Module Revision Handling

In summary, [I-D.verdt-netmod-yang-module-versioning] specifies minimal extensions and updates to the YANG language, YANG Library, and YANG author guidelines to provide more flexible YANG module revision handling. The intent is that these changes and extensions could be folded into future revisions of the updated specifications. The document provides a base solution for all requirements except Req 2.2, Req 3.1 and Req 3.2.

The extensions and changes in the document can be summarized thus:

- It defines a YANG extension statement to indicate where non-backwards-compatible changes have occurred in a module's revision history.

- It relaxes the rules for the module revision history to allow for a non-linear module revision history. I.e., any given module revision may have multiple revisions directly derived from it.

- It defines a new import extension statement that restricts the allowed module revisions that satisfy the import to only those derived from a specified module revision.

- It defines a revision label extension statement to allow an informative name to be associated with a particular revision date, and to be used in import statements, YANG module filenames, and is available in YANG library. One example of how the revision label could be used is to associate a semantic versioning scheme to YANG module revisions.

- It updates the YANG rules for changes between module revisions that are allowed to be classified as backwards-compatible. In particular, marking a node as obsolete is no longer classified as a backwards compatible change.

- It provides updated guidance on how servers handle deprecated and obsolete YANG nodes and augments YANG library with additional leaves to report the server's behavior to clients.

- It provides an extension statement to allow a description statement to be associated with a YANG status statement, providing more information about why the status has changed.

- It defines how versioning relates to YANG instance data.

- It refines the guidelines for updating modules, taking into consideration that non-backwards-compatible changes are sometimes necessary for various pragmatic reasons.

2.2. YANG Semantic Versioning

[I-D.verdt-netmod-yang-semver] defines a semantic versioning scheme, derived from the semver.org 2.0.0 specification, that can be used in conjunction with the revision label extension statement defined in Section 2.1 to allow semantic version numbers to be used to manage the revision lifecycle of YANG modules and other related YANG assets, e.g., YANG packages. This document provides an enhanced solution for Req 2.1, but organizations authoring modules are not obliged to use this specific versioning scheme, and could choose a different overlaid versioning scheme, or none at all and rely solely on revision dates.

The aims of the YANG semantic versioning scheme are:

- to generally allow clients to determine whether NBC changes have occurred between two revisions from the version number alone, without having to check the full revision history;

- to give a more informative identifier for a branched revision history over revision dates alone;

- to allow revision branches that contain fixes for published non-latest releases.

2.3. Versioned YANG packages

The two previous solution documents primarily address version and revision management of individual modules. [I-D.rwilton-netmod-yang-packages] provides a mechanism to group sets of related YANG modules revisions together, into constructs called YANG packages, and to apply a versioning scheme to the groups.

The core part of this document are YANG module definitions that define a YANG package. The definitions are used as an augmentation to YANG library and also in YANG instance data documents for offline access.

The principle aims of YANG packages are:

To define an efficient hierarchical structure that can precisely specify a YANG schema.

To provide an simple alternative mechanism to manage conformance of modules. Rather than checking conformance against a set of individual YANG module revisions and enabled features, it should be easier to check for conformance against a much smaller set of YANG package versions.

To provide a more efficient mechanism for servers to share conformance information with clients. Rather that downloading and comparing all individual module revisions and features via YANG library, the client can just check whether the package version is compatible instead. The package definition could be retrieved and cached from multiple sources.

To define constructs that can be used for YANG schema selection.

Although the YANG packages document does not satisfy any versioning requirements directly, it provides foundational building blocks for the schema selection solution, described in Section 2.4, that does address two of the requirements.

2.4. Dynamic YANG schema selection

[I-D.wilton-netmod-yang-ver-selection] specifies a solution for requirements 3.1 and 3.2 via the use of [I-D.rwilton-netmod-yang-packages] and a model and protocol based schema selection scheme that can be used by clients to choose which schemas to use when interacting with the device from the available schema that are supported and advertised by the server.

The dynamic YANG schema selection solution:

allows servers to define named 'schema-sets' which specify the schema for each supported datastore via references to YANG packages;

can support clients choosing a single default schema-set (from those advertised by the server) that is used for all NETCONF/RESTCONF protocol sessions;

can support clients enabling multiple compatible secondary schema-sets that can be used on separate NETCONF/RESTCONF protocol sessions;

can support clients configuring named custom schema-sets that can be selected as default or secondary schema-sets;

can support different module versions via placing them in different schema-sets;

can support different schema families (e.g., IETF YANG modules , native vendor, or OpenConfig);

allows considerable freedom in the schema selection capabilities that servers choose to support.

2.5. YANG Schema Comparison

The final piece of the solution jigsaw is a document that describes how to algorithmically compare YANG schema, addressing Req 2.2.

[I-D.verdt-netmod-yang-schema-comparison] specifies an algorithm that can be used to compare two revisions of a YANG schema to determine the overall scope of the changes, and a list of the specific changes, between the two revisions.

The YANG Schema Comparison solution:

defines a algorithm for comparing two YANG schema, identifying the differences and classifying them as backwards-compatible, non-backwards-compatible or editorial;

can be used to compare individual YANG module revisions;

can be used to compare YANG schema defined using YANG packages;

can filter the comparison output to the subset of the schema nodes that are of interest, providing a more precise answer for clients to determine whether they would likely be affected when upgrading between two schema versions;

defines YANG extensions to improve the accuracy of the comparison algorithm by explicitly annotating the type of change to statements within a YANG module, for use where the type of change would otherwise be ambiguous to a simple programmatic comparison algorithm.

3. Contributors

This document grew out of the YANG module versioning design team that started after IETF 101. The following individuals are (or have been) members of that design team and have contributed to defining the problem, specifying the requirements, and working on a solution:

o Balazs Lengyel

o Benoit Claise

- o Ebben Aries
- o Jason Sterne
- o Joe Clarke
- o Juergen Schoenwaelder
- o Mahesh Jethanandani
- o Michael (Wangzitao)
- o Qin Wu
- o Reshad Rahman
- o Rob Wilton
- o Susan Hares
- o Wu Bo

4. Security Considerations

The document does not define any new protocol or data model. There is no security impact.

5. IANA Considerations

None.

6. References

6.1. Normative References

[I-D.ietf-netmod-yang-versioning-reqs]
Clarke, J., "YANG Module Versioning Requirements", draft-ietf-netmod-yang-versioning-reqs-02 (work in progress), December 2019.

[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>.

6.2. Informative References

[I-D.rwilton-netmod-yang-packages]
Wilton, R., Rahman, R., Clarke, J., Sterne, J., and W. Bo, "YANG Packages", draft-rwilton-netmod-yang-packages-03 (work in progress), February 2020.

[I-D.verdt-netmod-yang-module-versioning]
Claise, B., Clarke, J., Rahman, R., Wilton, R., Lengyel, B., Sterne, J., and K. D'Souza, "Updated YANG Module Revision Handling", draft-verdt-netmod-yang-module-versioning-01 (work in progress), October 2019.

[I-D.verdt-netmod-yang-schema-comparison]
Wilton, R., "YANG Schema Comparison", draft-verdt-netmod-yang-schema-comparison-00 (work in progress), November 2019.

[I-D.verdt-netmod-yang-semver]
Claise, B., Clarke, J., Rahman, R., Wilton, R., Lengyel, B., Sterne, J., and K. D'Souza, "YANG Semantic Versioning", draft-verdt-netmod-yang-semver-01 (work in progress), October 2019.

[I-D.wilton-netmod-yang-ver-selection]
Wilton, R., Rahman, R., Clarke, J., Sterne, J., and W. Bo, "YANG Schema Selection", draft-wilton-netmod-yang-ver-selection-02 (work in progress), February 2020.

Author's Address

Robert Wilton (editor)
Cisco Systems, Inc.

Email: rwilton@cisco.com