NETMOD Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 14, 2016

K. Watsen
Juniper Networks
T. Nadeau
Brocade Networks
September 11, 2015

NETMOD Operational State Requirements
draft-chairs-netmod-opstate-reqs-00

Abstract

This document captures consensus on operational state requirements by the NETMOD working group.

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 http://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 March 14, 2016.

Table of Contents

1. Terminology 2
2. Requirements 2
3. Security Considerations 4
4. IANA Considerations 4
5. Acknowledgements 4
6. References 4
6.1. Normative References 4
6.2. Informative References 4
Appendix A. Relation to Requirements in Other Drafts 5
Appendix B. Open Issues 5

1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

The following terms are defined in [draft-openconfig-netmod-opstate-01]:

- intended configuration - this data represents the state that the network operator intends the system to be in. This data is colloquially referred to as the 'configuration' of the system.

- applied configuration - this data represents the state that the network element is actually in, i.e., that which is currently being run by particular software modules (e.g., the BGP daemon), or other systems within the device (e.g., a secondary control-plane, or line card).

- derived state - this data represents information which is generated as part of the system's own interactions. For example, derived state may consist of the results of protocol interactions (the negotiated duplex state of an Ethernet link), statistics (such as message queue depth), or counters (such as packet input or output bytes).

2. Requirements

1. Ability to interact with both intended and applied configuration

   A. The ability to ask the operational components of a system (e.g., line cards) for the configuration that they are currently using. This is the "applied configuration".

   B. Applied configuration is read-only

C. The data model for the applied configuration is the same as the data model for the intended configuration (same leaves)

D. For asynchronous systems, when fully synchronized, the data in the applied configuration is the same as the data in the intended configuration.

2. Applied configuration as part of operational state

A. The ability to retrieve the applied configuration and derived state nodes in a single protocol operation.

3. Support for both transactional, synchronous management systems as well as distributed, asynchronous management systems

A. For asynchronous systems, the ability to request a protocol operation to not return (i.e. block) until the intended configuration has been fully synchronized.

B. The protocol operation's response would indicate the result of the operation (success, failure, partial, etc.)

4. Separation of configuration and operational state data; ability to retrieve them independently

A. Be able to retrieve only the derived state aspects of operational state

B. Be able to retrieve only the non-derived state aspects of operational state

5. Ability to retrieve operational state corresponding only to derived values, statistics, etc.

// this is a duplicate of # 4-a

6. Ability to relate configuration with its corresponding operational state

A. Ability to map intended config nodes to corresponding applied config nodes

B. Ability to map intended config nodes to associated derived state nodes

C. The mappings needs to be programmatically consumable

7. Ability for distinct modules to leverage a common model-structure

A. Scope is limited to IETF-defined modules

B. Multiple domain-specific trees are okay

C. Multiple namespaces are okay

3. Security Considerations

None

4. IANA Considerations

None

5. Acknowledgements

TBD

6. References

6.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

6.2. Informative References

[draft-openconfig-netmod-model-structure-00]
Shaikh, A., Shakir, R., D'Souza, K., and L. Fang, "Operational Structure and Organization of YANG Models", draft-openconfig-netmod-model-structure-00 (work in progress), 2015, <https://tools.ietf.org/html/draft-openconfig-netmod-model-structure-00>.

[draft-openconfig-netmod-opstate-01]
Shakir, R., Shaikh, A., and M. Hines, "Consistent Modeling of Operational State Data in YANG", draft-openconfig-netmod-opstate-01 (work in progress), 2015, <https://tools.ietf.org/html/draft-openconfig-netmod-opstate-01>.

Appendix A. Relation to Requirements in Other Drafts

The requirements in this document roughly map onto the requirements listed in [draft-openconfig-netmod-opstate-01] and [draft-openconfig-netmod-model-structure-00] as list below. Some liberty was taken to adjust the requirements based on what looked liked consensus from on list discussions:

1. draft-openconfig-netmod-opstate-01, Section 3
2. draft-openconfig-netmod-opstate-01, Section 4.1
3. draft-openconfig-netmod-opstate-01, Section 4.2
4. draft-openconfig-netmod-opstate-01, Section 4.3
5. draft-openconfig-netmod-opstate-01, Section 4.4
6. draft-openconfig-netmod-opstate-01, Section 4.5
7. draft-openconfig-netmod-model-structure-00 (no section)

Appendix B. Open Issues

All issues with this draft are tracked using GitHub issues. Please see: https://github.com/netmod-wg/opstate-reqs/issues to see currently opened issues.

Authors' Addresses

Kent Watsen
Juniper Networks

EMail: kwatsen@juniper.net

Thomas Nadeau
Brocade Networks

EMail: tnadeau@lucidvision.com