NETMOD Working Group K. Watsen
Internet-Draft Juniper Networks
Intended status: Informational T. Nadeau
Expires: June 17, 2016 Brocade Networks
December 15, 2015

NETMOD Operational State Requirements
draft-ietf-netmod-opstate-reqs-01

Abstract

This document defines requirements for servers enabling better visibility and control over the server's operational state. To achieve this end, this document also defines terminology describing a conceptual model enabling the requirements to be expressed.

Status of This Memo

This Internet-Draft will expire on June 17, 2016.

Copyright Notice

Table of Contents

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 term "server" is used throughout this document to refer to what is many times known as the "device", "system", or "network element". This definition is intended to be consistent with the term "server" defined in [RFC6241], Section 1.1, but free of any association to a particular protocol.

This document defines the following terms:

Applied Configuration: This data represents the configuration state that the server is actually in. That is, the configuration state which is currently being used by server components (e.g., control plane daemons, operating system kernels, line cards).

> NOTE: The server's ability to report applied configuration accurately may be limited in some cases, such as when the configuration goes through an intermediate layer without an ability to inspect the lower layer.

Asynchronous Configuration Operation: A configuration request to update the running configuration of a server that is applied asynchronously with respect to the client request. The server MUST update its intended configuration (see term) before replying to the client indicating whether the request will be processed. This reply to the client only indicates whether there are any errors in the original request. The server's applied configuration state (see term) is updated after the configuration change has been fully effected to all impacted components in the

server. Once applied, there MUST be a mechanism for the client to determine when the request has completed processing and whether the intended config is now fully effective or there are any errors from applying the configuration change, which could be from an asynchronous notification or via a client operation.

Continue On Error: Continue to process configuration data on error; error is recorded, and negative response is generated if any errors occur.

Derived State: This data represents information which is generated as part of the server'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).

Intended Configuration: This data represents the configuration state that the network operator intends the server to be in, and that has been accepted by the server as valid configuration.

Operational State: Operational State is the current state of the system as known to the various components of the system (e.g., control plane daemons, operating system kernels, line cards). The operational state includes both applied configuration and derived state.

Rollback On Error: If an error condition occurs such that part of applying the configuration fails, the server will stop processing the configuration operation and restore the specified configuration to its complete state at the start of this configuration operation.

Synchronous Configuration Operation: A configuration request to update the running configuration of a server that is applied synchronously with respect to the client request (i.e. a blocking call). The server MUST fully attempt to apply the configuration change to all impacted components in the server, updating both the server's intended and applied configuration (see terms), before replying to the client. The reply to the client indicates whether there are any errors in the request or errors from applying the configuration change.

2. Requirements

1. Ability to interact with both intended and applied configuration

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

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. When a configuration change for any intended configuration node has been successfully applied to the server (e.g. not failed, nor deferred due to absent hardware) then the existence and value of the corresponding applied configuration node must match the intended configuration node.

2. Support for both synchronous and asynchronous configuration operations (see terms)

A. A server may support only synchronous configuration operations, or only asynchronous configuration operations, or both synchronous and asynchronous configuration operations on a client-specified per-operation basis.

B. Servers that support asynchronous configuration operations MAY also provide a verify operation that a client can request from the server to return information regarding the difference between the intended and applied configurations.

C. The configuration protocol MUST specify how configuration errors are handled. Errors may be handled by "stop on error", "continue on error" or "rollback on error" semantics (see terms). Support for "rollback on error" SHOULD be provided.

3. Separation of the applied configuration and derived state aspects of operational state; ability to retrieve them independently and together

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

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

C. Be able to retrieve both the applied configuration and derived state aspects of operational state together

4. 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

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

    A. Focus on the IETF-defined modules, and ideally provides guidance to other SDOs

    B. Multiple domain-specific model-structure trees are okay

    C. Model-structures may be defined in multiple modules with distinct namespaces

3. Security Considerations

None

4. IANA Considerations

None

5. Acknowledgements

The authors would like to thank the following for contributing to this document (in alphabetic order): Acee Lindem, Andy Bierman, Anees Shaikh, Benoit Claise, Carl Moberg, Dan Romascanu, Dean Bogdanovic, Gert Grammel, Jonathan Hansford, Juergen Schoenwaelder, Lou Berger, Mahesh Jethanandani, Martin Bjorklund, Phil Shafer, Randy Presuhn, Rob Shakir, Robert Wilton, Sterne, Jason.

6. References

6.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

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>.

[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, <http://www.rfc-editor.org/info/rfc6241>.

Appendix A. Relation to Terms Defined in Other Drafts

The following terms were originally defined in [RFC6241], but since modified by the NETMOD WG:

- continue-on-error
- stop-on-error
- rollback-on-error

The following terms were originally defined in [draft-openconfig-netmod-opstate-01], but since modified by the NETMOD WG:

- Intended Configuration
- Applied Configuration
- Derived State

Appendix B. 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 4.2
3. draft-openconfig-netmod-opstate-01, Section 4.3
4. draft-openconfig-netmod-opstate-01, Section 4.5
5. draft-openconfig-netmod-model-structure-00 (no section)

Appendix C. 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