

Network Working Group
Internet-Draft
Updates: [7950](#) (if approved)
Intended status: Standards Track
Expires: April 14, 2018

M. Bjorklund
Tail-f Systems
J. Schoenwaelder
Jacobs University
P. Shafer
K. Watsen
Juniper Networks
R. Wilton
Cisco Systems
October 11, 2017

NETCONF Model for NMDA
draft-ietf-netconf-nmda-netconf-00

Abstract

The "Network Management Datastore Architecture" (NMDA) improves on NETCONF by adding new features to give more accurate handling of configuration and operational data. These include ability to inspect the current operational values of configuration data, allowing clients to use identical paths for retrieving the configured values and the operational values. These new features require additional operations in network management applications such as NETCONF. This draft details those new operations.

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 April 14, 2018.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1.](#) Introduction [2](#)
- [1.1.](#) Keywords [3](#)
- [1.2.](#) Terminology [3](#)
- [2.](#) The NMDA Model for NETCONF [3](#)
- [2.1.](#) Operations [3](#)
- [2.1.1.](#) The <get-data> Operation [3](#)
- [2.1.2.](#) The <edit-data> Operation [4](#)
- [2.2.](#) Augmentations to the Base NETCONF Model [4](#)
- [2.3.](#) RPCs and Actions [5](#)
- [2.4.](#) YANG Library Capability [5](#)
- [3.](#) YANG Model [5](#)
- [4.](#) IANA Considerations [11](#)
- [5.](#) Security Considerations [11](#)
- [6.](#) Informative References [11](#)
- [Appendix A.](#) Examples [12](#)
- Authors' Addresses [12](#)

1. Introduction

This document provides a YANG model that adds NETCONF ([\[RFC6241\]](#)) support for the "Network Management Datastore Architecture" (NMDA) [\[I-D.ietf-netmod-revised-datastores\]](#). NMDA defines a framework for datastores, a fundamental concept binding network management data models to network management protocols, enabling data models to be written in a network management protocol agnostic way. NETCONF operations currently refer to the datastores defined in the original model, so new operations are required to allow references to the new datastores.

Operations like <copy-config>, <lock> and <unlock> are augmented to allow them to target additional datastores.

In addition the original <get> operation is deprecated, since the information it returns is no longer needed. <get>'s deficiencies were a major motivation for the NMDA.

1.1. Keywords

The keywords "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..

1.2. Terminology

This document uses the terminology defined by the NMDA [\[I-D.ietf-netmod-revised-datastores\]](#).

2. The NMDA Model for NETCONF

This section describes the changes needed for NMDA support. These changes are contained in a new YANG ([\[RFC7950\]](#)) model "ietf-netconf-datastores".

These changes include the use of source and target parameters based on the "datastore" identity defined in the "ietf-datastores" from [\[I-D.ietf-netmod-revised-datastores\]](#). The use of identities allows future expansion in a way that the choice-based strategy from the original operations (e.g. <get-config>, <edit-config>) do not.

2.1. Operations

Support for the NMDA includes two new operations defined in this document.

2.1.1. The <get-data> Operation

The <get-data> operation retrieves data from a specific NMDA datastore. This operation is similar to NETCONF's "get-config" operation, but adds flexibility in naming the target datastore.

The "source" parameter indicates the datastore which is the source of the data to be retrieved. This is a datastore identity.

The <get-data> operation accepts a content filter parameter, similar to the "filter" parameter of "get-config", but using explicit nodes for subtree filtering and XPath filtering.

Additional filters are defined to retrieve only "config true" nodes or nodes matching a given "origin" value.

The "get-data" operation also supports the "with-defaults" parameter as defined in [[RFC6243](#)]. The supported values follow the constraints given by the "with-defaults" capability.

2.1.1.1. Origin Attribute

The "get-data" operation adds a new boolean parameter named "with-origin", which requests that the server returns the "origin" information as detailed in the NMDA. This parameter is only valid for <operational> and any datastores with identities derived from the "operational" identity.

Data from <operational> can come from multiple sources. The server should return the most accurate value for the "origin" attribute as possible, indicating the source of the operational value.

When encoding the origin attribute for a hierarchy of returned nodes, the origin attribute may be omitted when the value matches that of the parent node.

2.1.2. The <edit-data> Operation

The <edit-data> operation changes the contents of a specific datastore, similar to the <edit-config> operation, but with additional flexibility in naming the target datastore.

The "target" parameter is a datastore identity that indicates the desired target datastore where changes should be made.

The "edit-content" parameter from "edit-config" it is modified to allow use "type anydata" for configuration content, rather than the "edit-config"'s use of "type anyxml".

The "default-operation" parameter mirrors the parameter of the "edit-config" operation.

2.2. Augmentations to the Base NETCONF Model

Several of the operations defined in the base NETCONF data model (ietf-netconf@2011-06-01.yang) will continue to be used under the NMDA. The <lock>, <unlock>, and <validate> operations are augmented with a new "datastore" leaf can indicate a desired NMDA datastore.

Only writable datastores can be locked.

2.3. RPCs and Actions

RPC operations and actions can be defined in YANG modules. The evaluation context for constraints and references in operation and actions is <operational>.

2.4. YANG Library Capability

RFC Ed.: Update 201X-XX-XX below with correct date.

Support for NMDA requires the server to implement at least revision 201X-XX-XX of the "ietf-yang-library" module defined in [[I-D.nmdsd-netconf-rfc7895bis](#)]. The server MUST advertise the following capability in the <hello> message (line breaks and whitespaces are used for formatting reasons only):

```
urn:ietf:params:netconf:capability:yang-library:1.1?
  revision=<date>&checksum=<checksum-value>
```

The parameter "revision" has the same value as the revision date of the "ietf-yang-library" module implemented by the server. This parameter MUST be present.

The parameter "checksum" has the same value as the leaf "/yang-library/checksum" from "ietf-yang-library". This parameter MUST be present.

With this mechanism, a client can cache the supported modules for a server and only update the cache if the "checksum" value in the <hello> message changes.

This document updates [[RFC7950](#)], [section 5.6.4](#), to allow servers to advertise the capability :yang-library:1.1 instead of :yang-library:1.0, and to implement the subtree "/yang-library" [[I-D.nmdsd-netconf-rfc7895bis](#)] instead of "/modules-state".

3. YANG Model

```
<CODE BEGINS> file "ietf-netconf-datastores@2017-08-24.yang"

module ietf-netconf-datastores {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-netconf-datastores";
  prefix ncds;

  import ietf-yang-types {
    prefix yang;
  }
}
```



```
import ietf-inet-types {
  prefix inet;
}
import ietf-datastores {
  prefix ds;
}
import ietf-origin {
  prefix or;
}
import ietf-netconf {
  prefix nc;
}
import ietf-netconf-with-defaults {
  prefix ncwd;
}
```

organization

"IETF NETCONF Working Group";

contact

"WG Web: <<https://datatracker.ietf.org/wg/netconf/>>

WG List: <<mailto:netconf@ietf.org>>

Author: Martin Bjorklund
<<mailto:mbj@tail-f.com>>

Author: Juergen Schoenwaelder
<<mailto:j.schoenwaelder@jacobs-university.de>>

Author: Phil Shafer
<<mailto:phil@juniper.net>>

Author: Kent Watsen
<<mailto:kwatsen@juniper.net>>

Author: Rob Wilton
<rwilton@cisco.com>";

description

"This YANG module defines a set of NETCONF operations for the Network Management Datastore Architecture (NMDA).

Copyright (c) 2017 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in [Section 4.c](#) of the IETF Trust's Legal Provisions

Relating to IETF Documents
(<http://trustee.ietf.org/license-info>).

This version of this YANG module is part of RFC XXXX
(<http://www.rfc-editor.org/info/rfcxxxx>); see the RFC itself
for full legal notices.";

```
revision 2017-08-24 {
  description
    "Initial revision.";
  reference "RFC XXXX: NETCONF Support for NMDA";
}

feature origin {
  description
    "Indicates that the server supports the 'origin' annotation.";
  reference "RFC XXXX: Network Management Datastore Architecture";
}

typedef datastore {
  type identityref {
    base ds:datastore;
  }
  description
    "An NMDA datastore.";
  reference "RFC XXXX: Network Management Datastore Architecture";
}

rpc get-data {
  description
    "Retrieve data from an NMDA datastore.";
  input {
    leaf source {
      type ncds:datastore;
      mandatory true;
      description
        "Datastore from which to retrieve data.";
    }
  }

  choice filter-spec {
    description
      "The content filter specification for this request.";

    anydata subtree-filter {
      description
        "This parameter identifies the portions of the
        target datastore to retrieve.";
      reference "RFC 6241, Section 6";
    }
  }
}
```



```
    }
    leaf xpath-filter {
      if-feature nc:xpath;
      type yang:xpath1.0;
      description
        "This parameter contains an XPath expression
        identifying the portions of the target
        datastore to retrieve.";
    }
  }
}

container where {
  description
    "Filter content with the specified criteria. All given
    criteria are logically AND:ed.";

  leaf config {
    type boolean;
    description
      "Filter for nodes with the given value for their
      'config' property.";
  }
  leaf origin {
    if-feature origin;
    type identityref {
      base or:origin;
    }
    description
      "Filter based on 'origin' annotation. A node matches the
      filter if its 'origin' annotation is derived from or
      equal to the given filter value.";
  }
}

leaf with-origin {
  when 'derived-from-or-self(..../source, "or:operational")';
  if-feature origin;
  type boolean;
  default false;
  description
    "If this parameter is 'true', the server will return
    the 'origin' annotation for the nodes that has one.";
}

uses ncwd:with-defaults-parameters;
}

output {
```



```
    anydata data {
      description
        "Copy of the source datastore subset which matched
        the filter criteria (if any). An empty data
        container indicates that the request did not
        produce any results.";
    }
  }
}
```

```
rpc edit-data {
  description
    "Edit data in an NMDA datastore.";
  input {
    leaf target {
      type ncds:datastore;
      description
        "Datastore which data affects.";
    }
    leaf default-operation {
      type enumeration {
        enum "merge" {
          description
            "The default operation is merge.";
        }
        enum "replace" {
          description
            "The default operation is replace.";
        }
        enum "none" {
          description
            "There is no default operation.";
        }
      }
      default "merge";
      description
        "The default operation to use.";
    }
  }
  choice edit-content {
    mandatory true;
    description
      "The content for the edit operation.";

    anydata config {
      description
        "Inline Config content.";
    }
    leaf url {
```



```
        if-feature nc:url;
        type inet:uri;
        description
            "URL based config content.";
    }
}
}
}

/*
 * Augment the lock and unlock operations with a
 * "datastore" parameter.
 */

augment "/nc:lock/nc:input/nc:target/nc:config-target" {
    description
        "Add NMDA Datastore as target.";
    leaf datastore {
        type ncds:datastore;
        description
            "Datastore to lock.";
    }
}

augment "/nc:unlock/nc:input/nc:target/nc:config-target" {
    description
        "Add NMDA Datastore as target.";
    leaf datastore {
        type ncds:datastore;
        description
            "Datastore to unlock.";
    }
}

/*
 * Augment the validate operation with a
 * "datastore" parameter.
 */

augment "/nc:validate/nc:input/nc:source/nc:config-source" {
    description
        "Add NMDA Datastore as source.";
    leaf datastore {
        type ncds:datastore;
        description
            "Datastore to validate.";
    }
}
}
```


<CODE ENDS>

4. IANA Considerations

This document registers one capability identifier URN from the "Network Configuration Protocol (NETCONF) Capability URNs" registry:

Index	Capability Identifier
-----	-----
:yang-library	urn:ietf:params:netconf:capability:yang-library:1.1

This document registers a URI in the "IETF XML Registry" [[RFC3688](#)]. Following the format in [RFC 3688](#), the following registration has been made.

URI: urn:ietf:params:xml:ns:yang:ietf-netconf-datastores

Registrant Contact: The IESG.

XML: N/A, the requested URI is an XML namespace.

This document registers a YANG module in the "YANG Module Names" registry [[RFC6020](#)].

name:	ietf-netconf-datastores
namespace:	urn:ietf:params:xml:ns:yang:ietf-netconf-datastores
prefix:	ncds
reference:	RFC XXXX

5. Security Considerations

This document has no security considerations.

6. Informative References

[I-D.ietf-netmod-revised-datastores]

Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture", [draft-ietf-netmod-revised-datastores-04](#) (work in progress), August 2017.

[I-D.nmdsd-netconf-rfc7895bis]

Bierman, A., Bjorklund, M., and K. Watsen, "YANG Library", [draft-nmdsd-netconf-rfc7895bis-01](#) (work in progress), July 2017.

- [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>>.
- [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>>.
- [RFC6243] Bierman, A. and B. Lengyel, "With-defaults Capability for NETCONF", [RFC 6243](#), DOI 10.17487/RFC6243, June 2011, <<https://www.rfc-editor.org/info/rfc6243>>.
- [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>>.
- [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>>.

Appendix A. Examples

Authors' Addresses

Martin Bjorklund
Tail-f Systems

Email: mbj@tail-f.com

Juergen Schoenwaelder
Jacobs University

Email: j.schoenwaelder@jacobs-university.de

Phil Shafer
Juniper Networks

Email: phil@juniper.net

Kent Watsen
Juniper Networks

Email: kwatsen@juniper.net

Robert Wilton
Cisco Systems

Email: rwilton@cisco.com