

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: January 11, 2021

M. Veillette, Ed.
Trilliant Networks Inc.
I. Petrov, Ed.
Acklio
July 10, 2020

Constrained YANG Module Library
draft-ietf-core-yang-library-02

Abstract

This document describes a constrained version of the YANG library that provides information about the YANG modules, datastores, and datastore schemas used by a constrained network management server (e.g., a CORECONF server).

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 January 11, 2021.

Copyright Notice

Copyright (c) 2020 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 (<https://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 |
| 2. Terminology and Notation | 2 |
| 3. Overview | 3 |
| 3.1. Tree diagram | 4 |
| 3.2. Major differences between ietf-constrained-yang-library and ietf-yang-library | 4 |
| 4. YANG Module "ietf-constrained-yang-library" | 5 |
| 5. IANA Considerations | 13 |
| 5.1. YANG Module Registration | 13 |
| 5.2. YANG Namespace Registration | 13 |
| 6. Security Considerations | 13 |
| 7. Acknowledgments | 14 |
| 8. References | 14 |
| 8.1. Normative References | 14 |
| 8.2. Informative References | 16 |
| Authors' Addresses | 16 |

1. Introduction

There is a need for a standard mechanism to expose which YANG modules, datastores and datastore schemas are in use by a constrained network management server. This document defines the YANG module 'ietf-constrained-yang-library' that provides this information.

YANG module 'ietf-constrained-yang-library' shares the same data model and objectives as 'ietf-yang-library', only datatypes and mandatory requirements have been updated to minimize its size to allow its implementation by Constrained Nodes and/or Constrained Networks as defined by [RFC7228]. To review the list of objectives and the proposed data model, please refer to [RFC8525] section 2 and 3.

2. Terminology and Notation

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.

The following terms are defined in [RFC7950]: client, deviation, feature, module, submodule, and server.

The following term is defined in [I-D.ietf-core-sid]: YANG Schema Item iDentifier (SID).

The following terms are defined in [RFC8525]: YANG library and YANG library checksum.

3. Overview

The conceptual model of the YANG library is depicted in Figure 1.

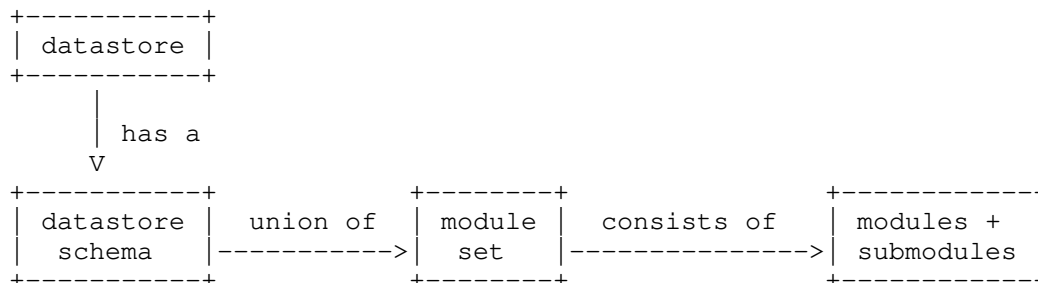


Figure 1: Conceptual model of the YANG library

It's expected that most constrained network management servers have one datastore (e.g. a unified datastore). However, some servers may have multiples datastore as described by NMDA [RFC8342]. The YANG library data model supports both cases.

In this model, every datastore has an associated datastore schema, which is the union of module sets, which is a collection of modules. Multiple datastores may refer to the same datastore schema and individual datastore schemas may share module sets.

For each module, the YANG library provides:

- o the YANG module identifier (i.e. SID)
- o its revision
- o its list of submodules
- o its list of imported modules
- o its set of features and deviations

YANG module namespace and location are also supported, but their implementation is not recommended for constrained servers.

3.1. Tree diagram

The tree diagram of YANG module `ietf-constrained-yang-library` is provided below. This graphical representation of a YANG module is defined in [RFC8340].

```

module: ietf-constrained-yang-library
  +--ro yang-library
    +--ro module-set* [index]
      +--ro index          uint8
      +--ro module* [identifier]
        +--ro identifier    sid:sid
        +--ro revision?     revision-identifier
        +--ro namespace?    inet:uri
        +--ro location*     inet:uri
        +--ro submodule* [identifier]
          +--ro identifier    sid:sid
          +--ro revision?     revision-identifier
          +--ro location*     inet:uri
        +--ro feature*      sid:sid
        +--ro deviation*    -> ../../module/identifier
      +--ro import-only-module* [identifier revision]
        +--ro identifier    sid:sid
        +--ro revision      union
        +--ro namespace?    inet:uri
        +--ro location*     inet:uri
        +--ro submodule* [identifier]
          +--ro identifier    sid:sid
          +--ro revision?     revision-identifier
          +--ro location*     inet:uri
    +--ro schema* [index]
      +--ro index          uint8
      +--ro module-set*    -> ../../module-set/index
    +--ro datastore* [identifier]
      +--ro identifier      ds:datastore-ref
      +--ro schema          -> ../../schema/index
    +--ro checksum          binary

  notifications:
    +---n yang-library-update
      +--ro checksum        -> /yang-library/checksum

```

3.2. Major differences between `ietf-constrained-yang-library` and `ietf-yang-library`

The changes between the reference data model '`ietf-yang-library`' and its constrained version '`ietf-constrained-yang-library`' are listed below:

- o module-set 'name' and schema 'name' are implemented using 8 bits unsigned integers and renamed 'index'.
- o module 'name', submodule 'name' and datastore 'name' are implemented using a SID (i.e. an unsigned integer) and renamed 'identifier'.
- o 'feature' and 'deviation' are implemented using a SID (i.e. an unsigned integer).
- o 'revision' fields are implemented using a 4 bytes binary string.
- o the mandatory requirement of the 'namespace' fields is removed, and implementation is not recommended. SIDs used by constrained devices and protocols don't require namespaces.
- o the implementation of the 'location' fields are not recommended, the use of the module SID as the handle to retrieve the associated YANG module is proposed instead.

4. YANG Module "ietf-constrained-yang-library"

RFC Ed.: update the date below with the date of RFC publication and remove this note.

```
<CODE BEGINS> file "ietf-constrained-yang-library@2019-03-28.yang"
module ietf-constrained-yang-library {
  yang-version 1.1;
  namespace
    "urn:ietf:params:xml:ns:yang:ietf-constrained-yang-library";
  prefix "yanglib";

  // RFC Ed.: update ietf-core-sid reference.

  import ietf-sid-file {
    prefix sid;
    reference "RFC YYYY: YANG Schema Item iDentifier (SID)";
    // RFC Editor: Please replace YYYY with RFC number of I-D.ietf-core-sid.
  }
  import ietf-inet-types {
    prefix inet;
    reference "RFC 6991: Common YANG Data Types.";
  }
  import ietf-datastores {
    prefix ds;
    reference
      "RFC 8342: Network Management Datastore Architecture (NMDA).";
  }
}
```

organization

"IETF CoRE Working Group";

contact

"WG Web: <<http://datatracker.ietf.org/wg/core/>>

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

Editor: Michel Veillette
<<mailto:michel.veillette@trilliantinc.com>>

Editor: Ivaylo Petrov
<<mailto:ivaylo@ackl.io>>;

description

"This module provides information about the YANG modules, datastores, and datastore schemas implemented by a constrained network management server.

Copyright (c) 2018 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; see the RFC itself for full legal notices.";

// RFC Editor: Please replace XXXX with RFC number and remove this note

revision 2019-03-28 {

description

"Second revision.";

reference

"RFC XXXX: Constrained YANG Module Library";

// RFC Editor: Please replace XXXX with RFC number and remove this note

}

/*

* Typedefs

*/

typedef revision-identifier {

type binary {

```
    length "4";
  }
  description
    "Revision date encoded as a binary string, each nibble
    representing a digit of the revision date. For example,
    revision 2018-09-21 is encoded as 0x20 0x18 0x09 0x21.";
}

/*
 * Groupings
 */

grouping module-identification-leafs {
  description
    "Parameters for identifying YANG modules and submodules.";

  leaf identifier {
    type sid:sid;
    mandatory true;
    description
      "SID assigned to this module or submodule.";
  }
  leaf revision {
    type revision-identifier;
    description
      "The YANG module or submodule revision date. If no
      revision statement is present in the YANG module
      or submodule, this leaf is not instantiated.";
  }
}

grouping location-leaf-list {
  description
    "Common location leaf list parameter for modules and
    submodules.";

  leaf-list location {
    type inet:uri;
    description
      "Contains a URL that represents the YANG schema resource
      for this module or submodule.

      This leaf is present in the model to keep the alignment
      with 'ietf-yang-library'. Support of this leaf in
      constrained devices is not necessarily required, nor
      expected. It is recommended that clients used the module
      or sub-module SID as the handle used to retrieve the
      corresponding YANG module";
  }
}
```

```
    }  
  }  
  
  grouping implementation-parameters {  
    description  
      "Parameters for describing the implementation of a module.";  
  
    leaf-list feature {  
      type sid:sid;  
      description  
        "List of all YANG feature names from this module that are  
        supported by the server, regardless whether they are  
        defined in the module or any included submodule.";  
    }  
    leaf-list deviation {  
      type leafref {  
        path "../..../module/identifier";  
      }  
      description  
        "List of all YANG deviation modules used by this server to  
        modify the conformance of the module associated with this  
        entry. Note that the same module can be used for  
        deviations for multiple modules, so the same entry MAY  
        appear within multiple 'module' entries.  
  
        This reference MUST NOT (directly or indirectly)  
        refer to the module being deviated.  
  
        Robust clients may want to make sure that they handle a  
        situation where a module deviates itself (directly or  
        indirectly) gracefully."  
    }  
  }  
  
  grouping module-set-parameters {  
    description  
      "A set of parameters that describe a module set.";  
  
    leaf index {  
      type uint8;  
      description  
        "An arbitrary number assigned of the module set.";  
    }  
    list module {  
      key "identifier";  
      description  
        "An entry in this list represents a module implemented  
        by the server, as per RFC 7950 section 5.6.5, with a
```



```
    particular set of supported features and deviations.";
reference
    "RFC 7950: The YANG 1.1 Data Modeling Language.";

uses module-identification-leafs;

leaf namespace {
    type inet:uri;
    description
        "The XML namespace identifier for this module.
        This leaf is present in the model to keep the alignment
        with 'ietf-yang-library'. Support of this leaf in
        constrained devices is not required, nor expected.";
}

uses location-leaf-list;

list submodule {
    key "identifier";
    description
        "Each entry represents one submodule within the parent
        module.";
    uses module-identification-leafs;
    uses location-leaf-list;
}

uses implementation-parameters;
}
list import-only-module {
    key "identifier revision";
    description
        "An entry in this list indicates that the server imports
        reusable definitions from the specified revision of the
        module, but does not implement any protocol accessible
        objects from this revision.

        Multiple entries for the same module name MAY exist.
        This can occur if multiple modules import the same
        module, but specify different revision-dates in the
        import statements.";

    leaf identifier {
        type sid:sid;
        description
            "The YANG module name.";
    }
    leaf revision {
        type union {
```

```
        type revision-identifier;
        type string {
            length 0;
        }
    }
    description
        "The YANG module revision date.";
}
leaf namespace {
    type inet:uri;
    description
        "The XML namespace identifier for this module.
        This leaf is present in the model to keep the alignment
        with 'ietf-yang-library'. Support of this leaf in
        constrained devices is not required, nor expected.";
}

uses location-leaf-list;

list submodule {
    key "identifier";
    description
        "Each entry represents one submodule within the
        parent module.";

    uses module-identification-leafs;
    uses location-leaf-list;
}
}

grouping yang-library-parameters {
    description
        "The YANG library data structure is represented as a grouping
        so it can be reused in configuration or another monitoring
        data structure.";

    list module-set {
        key index;
        description
            "A set of modules that may be used by one or more schemas.

            A module set does not have to be referentially complete,
            i.e., it may define modules that contain import statements
            for other modules not included in the module set.";

        uses module-set-parameters;
    }
}
```

```
list schema {
  key "index";
  description
    "A datastore schema that may be used by one or more
    datastores.

    The schema must be valid and referentially complete,
    i.e., it must contain modules to satisfy all used import
    statements for all modules specified in the schema.";

  leaf index {
    type uint8;
    description
      "An arbitrary reference number assigned to the schema.";
  }
  leaf-list module-set {
    type leafref {
      path "../../module-set/index";
    }
    description
      "A set of module-sets that are included in this schema.
      If a non import-only module appears in multiple module
      sets, then the module revision and the associated
      features and deviations must be identical.";
  }
}

list datastore {
  key "identifier";
  description
    "A datastore supported by this server.

    Each datastore indicates which schema it supports.

    The server MUST instantiate one entry in this list
    per specific datastore it supports.

    Each datastore entry with the same datastore schema
    SHOULD reference the same schema.";

  leaf identifier {
    type ds:datastore-ref;
    description
      "The identity of the datastore.";
  }
  leaf schema {
    type leafref {
      path "../../schema/index";
    }
  }
}
```

```
    }
    mandatory true;
    description
      "A reference to the schema supported by this datastore.
      All non import-only modules of the schema are
      implemented with their associated features and
      deviations.";
  }
}

/*
 * Top-level container
 */

container yang-library {
  config false;
  description
    "Container holding the entire YANG library of this server.";

  uses yang-library-parameters;

  leaf checksum {
    type binary;
    mandatory true;
    description
      "A server-generated checksum or digest of the contents of
      the 'yang-library' tree. The server MUST change the
      value of this leaf if the information represented by
      the 'yang-library' tree, except 'yang-library/checksum',
      has changed.";
  }
}

/*
 * Notifications
 */

notification yang-library-update {
  description
    "Generated when any YANG library information on the
    server has changed.";

  leaf checksum {
    type leafref {
      path "/yanglib:yang-library/yanglib:checksum";
    }
    mandatory true;
  }
}
```

```
        description
          "Contains the YANG library checksum or digest for the
           updated YANG library at the time the notification is
           generated.";
      }
    }
  }
<CODE ENDS>
```

5. IANA Considerations

5.1. YANG Module Registration

This document registers one YANG module in the YANG Module Names registry [RFC7950].

name: ietf-constrained-yang-library

namespace: urn:ietf:params:xml:ns:yang:ietf-constrained-yang-library

prefix: lib

reference: RFC XXXX

// RFC Ed.: replace XXXX with RFC number and remove this note

5.2. YANG Namespace Registration

This document registers the following XML namespace URN in the "IETF XML Registry", following the format defined in [RFC3688]:

URI: please assign urn:ietf:params:xml:ns:yang:ietf-constrained-yang-library

Registrant Contact: The IESG.

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

Reference: RFC XXXX

// RFC Ed.: replace XXXX with RFC number and remove this note

6. Security Considerations

The YANG module specified in this document defines a schema for data that is designed to be accessed via network management protocols such as NETCONF [RFC6241], RESTCONF [RFC8040] or CORECONF [I-D.ietf-core-comi]. The lowest NETCONF layer is the secure

transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC8446]. The lowest CORECONF layer is CoAP [RFC7252] and the mandatory-to-implement security transport is any one of DTLS [RFC6347] and OSCORE [RFC8613].

The Network Configuration Access Control Model (NACM) [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

Some of the readable data nodes in this YANG module may be considered sensitive or vulnerable in some network environments. It is thus important to control read access (e.g., via get, get-config, or notification) to these data nodes.

Specifically, the 'module' list may help an attacker to identify the server capabilities and server implementations with known bugs. Server vulnerabilities may be specific to particular modules, module revisions, module features, or even module deviations. This information is included in each module entry. For example, if a particular operation on a particular data node is known to cause a server to crash or significantly degrade device performance, then the module list information will help an attacker to identify server implementations with such a defect, in order to launch a denial of service attack on these devices.

7. Acknowledgments

The YANG module defined by this memo has been derived from an already existing YANG module, ietf-yang-library [RFC8525], we will like to thank the authors of this YANG module. A special thank also to Andy Bierman for his initial recommendations for the creation of this YANG module. The authors would also like to thank Tom Petch for his help during the development of this document and his useful comments during the review process.

8. References

8.1. Normative References

[I-D.ietf-core-comi]

Veillette, M., Stok, P., Pelov, A., Bierman, A., and I. Petrov, "CoAP Management Interface (CORECONF)", draft-ietf-core-comi-10 (work in progress), July 2020.

- [I-D.ietf-core-sid] Veillette, M., Pelov, A., and I. Petrov, "YANG Schema Item Identifier (YANG SID)", draft-ietf-core-sid-14 (work in progress), July 2020.
- [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>>.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <<https://www.rfc-editor.org/info/rfc3688>>.
- [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>>.
- [RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <<https://www.rfc-editor.org/info/rfc6242>>.
- [RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, DOI 10.17487/RFC6347, January 2012, <<https://www.rfc-editor.org/info/rfc6347>>.
- [RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <<https://www.rfc-editor.org/info/rfc6991>>.
- [RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <<https://www.rfc-editor.org/info/rfc7252>>.
- [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>>.
- [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>>.
- [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>>.

- [RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <<https://www.rfc-editor.org/info/rfc8340>>.
- [RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <<https://www.rfc-editor.org/info/rfc8341>>.
- [RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", RFC 8342, DOI 10.17487/RFC8342, March 2018, <<https://www.rfc-editor.org/info/rfc8342>>.
- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.
- [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>>.
- [RFC8613] Selander, G., Mattsson, J., Palombini, F., and L. Seitz, "Object Security for Constrained RESTful Environments (OSCORE)", RFC 8613, DOI 10.17487/RFC8613, July 2019, <<https://www.rfc-editor.org/info/rfc8613>>.

8.2. Informative References

- [RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, DOI 10.17487/RFC7228, May 2014, <<https://www.rfc-editor.org/info/rfc7228>>.

Authors' Addresses

Michel Veillette (editor)
Trilliant Networks Inc.
610 Rue du Luxembourg
Granby, Quebec J2J 2V2
Canada

Email: michel.veillette@trilliantinc.com

Ivaylo Petrov (editor)
Acklio
1137A avenue des Champs Blancs
Cesson-Sevigne, Bretagne 35510
France

Email: ivaylo@ackl.io