Abstract

This document defines two YANG modules: the first defines a grouping for configuring a generic TCP client, and the second defines a grouping for configuring a generic TCP server. It is intended that these groupings will be used by applications using the TCP protocol.

Editorial Note (To be removed by RFC Editor)

This draft contains many placeholder values that need to be replaced with finalized values at the time of publication. This note summarizes all of the substitutions that are needed. No other RFC Editor instructions are specified elsewhere in this document.

Artwork in this document contains placeholder values for the date of publication of this draft. Please apply the following replacement:

- "2019-03-09" --> the publication date of this draft

The following Appendix section is to be removed prior to publication:

- Appendix A. Change Log

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 September 10, 2019.
1. Introduction

This document defines two YANG 1.1 [RFC7950] modules: the first defines a grouping for configuring a generic TCP client, and the second defines a grouping for configuring a generic TCP server. It is intended that these groupings will be used by applications using the TCP protocol. For instance, these groupings could help define the configuration module for an SSH, TLS, or HTTP based application.
2. Terminology

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.

3. The TCP Client Model

3.1. Tree Diagram

This section provides a tree diagram [RFC8340] for the "ietf-tcp-client" module.

module: ietf-tcp-client

  grouping tcp-client-grouping
    +- remote-address    inet:host
    +- remote-port?      inet:port-number
    +- local-address?    inet:ip-address
    +- local-port?       inet:port-number
    ++ tcp-keepalives {tcp-client-keepalives}?
      +- idle-time?      uint16
      +- max-probes?     uint16
      +- probe-interval? uint16

  grouping ip-params-grouping
    +- remote-address    inet:host
    +- remote-port?      inet:port-number
    +- local-address?    inet:ip-address
    +- local-port?       inet:port-number

  grouping keepalives-grouping
    ++ tcp-keepalives {tcp-client-keepalives}?
      +- idle-time?      uint16
      +- max-probes?     uint16
      +- probe-interval? uint16

3.2. Example Usage

This section presents an example showing the tcp-client-grouping populated with some data.
3.3. YANG Module

This YANG module has normative references to [RFC6991].

<CODE BEGINS> file "ietf-tcp-client@2019-03-09.yang"
module ietf-tcp-client {
  yang-version 1.1;
  prefix tcpc;

  import ietf-inet-types {
    prefix inet;
    reference
      "RFC 6991: Common YANG Data Types";
  }

  organization
    "IETF NETCONF (Network Configuration) Working Group";

  contact
    "WG Web:  <http://datatracker.ietf.org/wg/netconf/>"
    "WG List:  <mailto:netconf@ietf.org>"
    "Author: Kent Watsen <mailto:kent+ietf@watsen.net>";

  description
    "This module defines reusable groupings for TCP clients that
    can be used as a basis for specific TCP client instances.

    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.

    Copyright (c) 2019 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.

revision 2019-03-09 {
  description
    "Initial version";
  reference
    "RFC XXXX: YANG Groupings for TCP Clients and TCP Servers";
}

// Features

feature tcp-client-keepalives {
  description
    "Per socket TCP keepalive parameters are configurable for
     TCP clients on the server implementing this feature.";
}

// Groupings

grouping tcp-client-grouping {
  description
    "A reusable grouping for configuring a TCP client.";
  uses ip-params-grouping;
  uses keepalives-grouping;
}

grouping ip-params-grouping {
  description
    "A reusable grouping for configuring TCP client IP level
     parameters.";
  leaf remote-address {
    type inet:host;
    mandatory true;
    description
    "The IP address or hostname of the remote peer to connect to.
     If a domain name is configured, then the DNS resolution
     should happen on each connection attempt. If the the DNS
     resolution results in multiple IP addresses, the IP addresses
are tried according to local preference order until a connection has been established or until all IP addresses have failed.

leaf remote-port {
  type inet:port-number;
  default "0";
  description
  "The IP port number for the remote peer to connect to. An invalid default value (0) is used (instead of ‘mandatory true’) so that an application level data model may ‘refine’ it with an application specific default port number value."
}

leaf local-address {
  type inet:ip-address;
  description
  "The local IP address/interface (VRF?) to bind to for when connecting to the remote peer. INADDR_ANY (‘0.0.0.0’) or INADDR6_ANY (‘0:0:0:0:0:0:0:0’a.k.a. ‘::’) MAY be used to explicitly indicate the implicit default, that the server can bind to any IPv4 or IPv6 addresses, respectively."
}

leaf local-port {
  type inet:port-number;
  default "0";
  description
  "The local IP port number to bind to for when connecting to the remote peer. The port number ‘0’, which is the default value, indicates that any available local port number may be used."
}

grouping keepalives-grouping {
  description
  "A reusable grouping for configuring TCP client keepalive parameters."
  container tcp-keepalives {
    if-feature "tcp-client-keepalives";
    description
    "Configures the keep-alive policy, to proactively test the aliveness of the TCP server. Not all clients will use all the values, based on capabilities of the underlying operating system. An unresponsive TCP server is dropped after approximately (idle-time * 60) + (max-probes * probe-interval) seconds."
    leaf idle-time {
      type uint16 {

range "1..max";
}
units "minutes"
description
"Sets the amount of time in minutes after which if no data has been received from the TCP server, a TCP-level probe message will be sent to test the aliveness of the TCP server. When 'idle-time' is not configured (the default) TCP keep-alives are disabled."
}
leaf max-probes {
  type uint16 {
    range "1..max";
  }
description
"Sets the maximum number of sequential keep-alive probes that can fail to obtain a response from the TCP server before assuming the TCP server is no longer alive. If no value is specified, then the operating system provided default value is used."
}
leaf probe-interval {
  type uint16 {
    range "1..max";
  }
  units "seconds"
description
"Sets the time interval between failed probes. If no value is specified, then the operating system provided default value is used."
}
}

<CODE ENDS>

4. The TCP Server Model

4.1. Tree Diagram

This section provides a tree diagram [RFC8340] for the "ietf-tcp-server" module.
module: ietf-tcp-server

grouping tcp-server-grouping
  +-- local-address     inet:ip-address
  +-- local-port?       inet:port-number
  +-- tcp-keepalives {tcp-server-keepalives}?
    +-- idle-time?       uint16
    +-- max-probes?      uint16
    +-- probe-interval?  uint16

grouping ip-params-grouping
  +-- local-address    inet:ip-address
  +-- local-port?      inet:port-number

grouping keepalives-grouping
  +-- tcp-keepalives {tcp-server-keepalives}?
    +-- idle-time?       uint16
    +-- max-probes?      uint16
    +-- probe-interval?  uint16

4.2. Example Usage

This section presents an example showing the tcp-server-grouping populated with some data.

<tcp-server xmlns="urn:ietf:params:xml:ns:yang:ietf-tcp-server">
  <local-address>10.20.30.40</local-address>
  <local-port>7777</local-port>
  <tcp-keepalives>
    <idle-time>15</idle-time>
    <max-probes>3</max-probes>
    <probe-interval>30</probe-interval>
  </tcp-keepalives>
</tcp-server>

4.3. YANG Module

This YANG module has normative references to [RFC6991].
This module defines reusable groupings for TCP servers that can be used as a basis for specific TCP server instances.

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.

Copyright (c) 2019 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.;

revision 2019-03-09 {
  description
    "Initial version";
  reference
    "RFC XXXX: YANG Groupings for TCP Clients and TCP Servers";
}

// Features

feature tcp-server-keepalives {
  description
    "Per socket TCP keepalive parameters are configurable for TCP servers on the server implementing this feature.";
}

// Groupings

Watsen                 Expires September 10, 2019               [Page 9]
grouping tcp-server-grouping {
    description
        "A reusable grouping for configuring a TCP server.";
    uses ip-params-grouping;
    uses keepalives-grouping;
}

grouping ip-params-grouping {
    description
        "A reusable grouping for configuring TCP server IP level parameters.";
    leaf local-address {
        type inet:ip-address;
        mandatory true;
        description
            "The local IP address to listen on for incoming TCP client connections. INADDR_ANY (0.0.0.0) or INADDR6_ANY (0:0:0:0:0:0:0:0 a.k.a. ::) MUST be used when the server is to listen on all IPv4 or IPv6 addresses, respectively.";
    }
    leaf local-port {
        type inet:port-number;
        default "0";
        description
            "The local port number to listen on for incoming TCP client connections. An invalid default value (0) is used (instead of "mandatory true") so that an application level data model may 'refine' it with an application specific default port number value.";
    }
}

grouping keepalives-grouping {
    description
        "A reusable grouping for configuring TCP server keepalive parameters.";
    container tcp-keepalives {
        if-feature "tcp-server-keepalives";
        description
            "Configures the keep-alive policy, to proactively test the aliveness of the TCP client. Not all servers will use all the values, based on capabilities of the underlying operating system. An unresponsive TCP client is dropped after approximately (idle-time * 60) + (max-probes * probe-interval) seconds.";
    }
    leaf idle-time {
        type uint16 {
            range "1..max";
5. Security Considerations

The YANG modules defined in this document are designed to be accessed via YANG based management protocols, such as NETCONF [RFC6241] and RESTCONF [RFC8040]. Both of these protocols have mandatory-to-implement secure transport layers (e.g., SSH, TCP) with mutual authentication.

The NETCONF access control model (NACM) [RFC8341] provides the means to restrict access for particular users to a pre-configured subset of all available protocol operations and content.
Since the modules defined in this document only define groupings, these considerations are primarily for the designers of other modules that use these groupings.

There are a number of data nodes defined in the YANG modules that are writable/creatable/deletable (i.e., config true, which is the default). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations. These are the subtrees and data nodes and their sensitivity/vulnerability:

NONE

Some of the readable data nodes in the YANG modules 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. These are the subtrees and data nodes and their sensitivity/vulnerability:

NONE

Some of the RPC operations in this YANG module may be considered sensitive or vulnerable in some network environments. It is thus important to control access to these operations. These are the operations and their sensitivity/vulnerability:

NONE

6. IANA Considerations

6.1. The IETF XML Registry

This document registers two URIs in the "ns" subregistry of the IETF XML Registry [RFC3688]. Following the format in [RFC3688], the following registrations are requested:

Registrant Contact: The NETCONF WG of the IETF.
XML: N/A, the requested URI is an XML namespace.

Registrant Contact: The NETCONF WG of the IETF.
XML: N/A, the requested URI is an XML namespace.
6.2. The YANG Module Names Registry

This document registers two YANG modules in the YANG Module Names registry [RFC6020]. Following the format in [RFC6020], the following registrations are requested:

name: ietf-tcp-client
prefix: tcpc
reference: RFC XXXX

name: ietf-tcp-server
prefix: tcps
reference: RFC XXXX

7. References

7.1. Normative References


7.2. Informative References


Author’s Address

Kent Watsen
Watsen Networks

EMail: kent+ietf@watsen.net