NETCONF Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 7, 2017

K. Watsen
Juniper Networks

November 3, 2016

TLS Client and Server Models
draft-ietf-netconf-tls-client-server-01

Abstract

This document defines two YANG modules, one defines groupings for a generic TLS client and the other defines groupings for a generic TLS server. It is intended that these groupings will be used by applications using the TLS 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.

This document contains references to other drafts in progress, both in the Normative References section, as well as in body text throughout. Please update the following references to reflect their final RFC assignments:

- draft-ietf-netconf-keystore

Artwork in this document contains shorthand references to drafts in progress. Please apply the following replacements:

- "XXXX" --> the assigned RFC value for this draft
- "YYYY" --> the assigned RFC value for draft-ietf-netconf-keystore

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

- "2016-11-02" --> the publication date of this draft

The following two Appendix sections are to be removed prior to publication:

- Appendix A. Change Log
- Appendix B. Open Issues

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This document defines two YANG [RFC6020] modules, one defines groupings for a generic TLS client and the other defines groupings for a generic TLS server (TLS is defined in [RFC5246]). It is intended that these groupings will be used by applications using the TLS protocol. For instance, these groupings could be used to help define the data model for an HTTPS [RFC2818] server or a NETCONF over TLS [RFC7589] based server.

The two YANG modules in this document each define two groupings. One grouping defines everything other than what's needed for the TCP [RFC793] protocol layer. The other grouping uses the first grouping while adding TCP layer specifics (e.g., addresses to connect to, ports to listen on, etc.). This separation is done in order to enable applications the opportunity to define their own strategy for how the underlying TCP connection is established. For instance, applications supporting NETCONF Call Home [draft-ietf-netconf-call-home] could use the first grouping for the TLS parts it provides, while adding data nodes for the reversed TCP layer.

1.1. Terminology

The keywords "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].

1.2. Tree Diagrams

A simplified graphical representation of the data models is used in this document. The meaning of the symbols in these diagrams is as follows:

- Brackets "[" and "]" enclose list keys.

- Braces "{" and "}" enclose feature names, and indicate that the named feature must be present for the subtree to be present.

- Abbreviations before data node names: "rw" means configuration (read-write) and "ro" state data (read-only).

- Symbols after data node names: "?" means an optional node, "!" means a presence container, and "*" denotes a list and leaf-list.

- Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (":").

- Ellipsis ("...") stands for contents of subtrees that are not shown.

2. The TLS Client Model

EDITOR NOTE: Please ignore this section, it is incomplete.

The TLS client model presented in this section contains two YANG groupings, one for a client that initiates the underlying TCP connection and another for a client that has had the TCP connection opened for it already (e.g., call home).

Both of these groupings reference data nodes defined by the Keystore model [draft-ietf-netconf-keystore]. For instance, a reference to the keystore model is made to indicate which trusted CA certificate a client should use to authenticate the server's certificate.

2.1. Tree Diagram

The following tree diagram presents the data model for the two groupings defined in the ietf-tls-client module.

```
module: ietf-tls-client
  groupings:
  initiating-tls-client-grouping
      +---- some-TBD-tcp-client-stuff?   string
      +---- some-TBD-tls-client-stuff?   string

  non-initiating-tls-client-grouping
      +---- some-TBD-tls-client-stuff?   string
```

2.2. Example Usage

This section shows how it would appear if the initiating-tls-client-grouping were populated with some data. This example is consistent with the examples presented in Section 2.2 of [draft-ietf-netconf-keystore].

FIXME

2.3. YANG Model

This YANG module has a normative references to [RFC6991] and [draft-ietf-netconf-keystore].

<CODE BEGINS> file "ietf-tls-client@2016-11-02.yang"

```
// Editor's Note:
// This module is incomplete at this time.  Below is
// just a skeleton so there's something in the draft.
// Please ignore this module for now!

module ietf-tls-client {
  yang-version 1.1;

  namespace "urn:ietf:params:xml:ns:yang:ietf-tls-client";
  prefix "tlsc";
/*
  import ietf-inet-types {
    prefix inet;
    reference
      "RFC 6991: Common YANG Data Types";
  }

  import ietf-keystore {
    prefix ks;
    reference
      "RFC YYYY: Keystore Model";
  }
*/
  organization
   "IETF NETCONF (Network Configuration) Working Group";

  contact
   "WG Web:   <http://tools.ietf.org/wg/netconf/>
    WG List:  <mailto:netconf@ietf.org>

    WG Chair: Mehmet Ersue
              <mailto:mehmet.ersue@nsn.com>

    WG Chair: Mahesh Jethanandani
              <mailto:mjethanandani@gmail.com>

    Editor:   Kent Watsen
              <mailto:kwatsen@juniper.net>";
```

```
     description
      "This module defines a reusable grouping for a TLS client that
       can be used as a basis for specific TLS client instances.

       Copyright (c) 2014 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 "2016-11-02" {
       description
        "Initial version";
       reference
        "RFC XXXX: TLS Client and Server Models";
     }

     grouping initiating-tls-client-grouping {
       description
         "A reusable grouping for a TLS client that initiates the
          underlying TCP transport connection.";
       leaf some-TBD-tcp-client-stuff {
         type string;
         description "";
       }
       uses non-initiating-tls-client-grouping;
     }

    grouping non-initiating-tls-client-grouping {
       description
         "A reusable grouping for a TLS client that does not initiate
          the underlying TCP transport connection.";
       leaf some-TBD-tls-client-stuff {
         type string;
         description "";
       }
     }

   }
```

<CODE ENDS>

3. The TLS Server Model

The TLS server model presented in this section contains two YANG groupings, one for a server that opens a socket to accept TCP connections and another for a server that has had the TCP connection opened for it already (e.g., inetd).

Both of these groupings reference data nodes defined by the Keystore model [draft-ietf-netconf-keystore]. For instance, a reference to the keystore model is made to indicate the certificate a server should present.

3.1. Tree Diagram

The following tree diagram presents the data model for the two groupings defined in the ietf-tls-server module.

```
module: ietf-tls-server
  groupings:
  listening-tls-server-grouping
      +---- address?        inet:ip-address
      +---- port?           inet:port-number
      +---- certificates
      |  +---- certificate* [name]
      |     +---- name?   -> /ks:keystore/private-keys/private-key/cert
ificate-chains/certificate-chain/name
      +---- client-auth
         +---- trusted-ca-certs?       -> /ks:keystore/trusted-certific
ates/name
         +---- trusted-client-certs?   -> /ks:keystore/trusted-certific
ates/name

  non-listening-tls-server-grouping
      +---- certificates
      |  +---- certificate* [name]
      |     +---- name?   -> /ks:keystore/private-keys/private-key/cert
ificate-chains/certificate-chain/name
      +---- client-auth
         +---- trusted-ca-certs?       -> /ks:keystore/trusted-certific
ates/name
         +---- trusted-client-certs?   -> /ks:keystore/trusted-certific
ates/name
```

3.2. Example Usage

This section shows how it would appear if the listening-tls-server-grouping were populated with some data. This example is consistent with the examples presented in Section 2.2 of [draft-ietf-netconf-keystore].

```
<listening-tls-server
  xmlns="urn:ietf:params:xml:ns:yang:ietf-tls-server">
  <port>6513</port>
  <certificates>
    <certificate>
      <name>ex-key-sect571r1-cert</name>
    </certificate>
  </certificates>
  <client-auth>
    <trusted-ca-certs>
      deployment-specific-ca-certs
    </trusted-ca-certs>
    <trusted-client-certs>
      explicitly-trusted-client-certs
    </trusted-client-certs>
  </client-auth>
</listening-tls-server>
```

3.3. YANG Model

This YANG module has a normative references to [RFC6991], and [draft-ietf-netconf-keystore].

```
<CODE BEGINS> file "ietf-tls-server@2016-11-02.yang"

module ietf-tls-server {
  yang-version 1.1;

  namespace "urn:ietf:params:xml:ns:yang:ietf-tls-server";
  prefix "tlss";

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

  import ietf-keystore {
    prefix ks;
    reference
```

```
        "RFC YYYY: Keystore Model";
    }

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

    contact
     "WG Web:   <http://tools.ietf.org/wg/netconf/>
      WG List:  <mailto:netconf@ietf.org>

      WG Chair: Mehmet Ersue
                <mailto:mehmet.ersue@nsn.com>

      WG Chair: Mahesh Jethanandani
                <mailto:mjethanandani@gmail.com>

      Editor:   Kent Watsen
                <mailto:kwatsen@juniper.net>";


    description
     "This module defines a reusable grouping for a TLS server that
      can be used as a basis for specific TLS server instances.

      Copyright (c) 2014 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 "2016-11-02" {
      description
       "Initial version";
      reference
       "RFC XXXX: TLS Client and Server Models";
    }


    // grouping
    grouping non-listening-tls-server-grouping {
      description
```

```
        "A reusable grouping for a TLS server that can be used as a
         basis for specific TLS server instances.";
      container certificates {
        description
          "The list of certificates the TLS server will present when
           establishing a TLS connection in its Certificate message,
           as defined in Section 7.4.2 in RRC 5246.";
        reference
          "RFC 5246:
             The Transport Layer Security (TLS) Protocol Version 1.2";
        list certificate {
          key name;
          min-elements 1;
          description
            "An unordered list of certificates the TLS server can pick
             from when sending its Server Certificate message.";
          reference
            "RFC 5246: The TLS Protocol, Section 7.4.2";
          leaf name {
            type leafref {
              path "/ks:keystore/ks:private-keys/ks:private-key/"
                   + "ks:certificate-chains/ks:certificate-chain/"
                   + "ks:name";
            }
            description
              "The name of the certificate in the keystore.";
          }
        }
      }

      container client-auth {
        description
          "A reference to a list of trusted certificate authority (CA)
           certificates and a reference to a list of trusted client
           certificates.";
        leaf trusted-ca-certs {
          type leafref {
            path "/ks:keystore/ks:trusted-certificates/ks:name";
          }
          description
            "A reference to a list of certificate authority (CA)
             certificates used by the TLS server to authenticate
             TLS client certificates.";
        }

        leaf trusted-client-certs {
          type leafref {
            path "/ks:keystore/ks:trusted-certificates/ks:name";
```

```
          }
          description
            "A reference to a list of client certificates used by
             the TLS server to authenticate TLS client certificates.
             A clients certificate is authenticated if it is an
             exact match to a configured trusted client certificate.";
        }
      }
    }


    grouping listening-tls-server-grouping {
      description
        "A reusable grouping for a TLS server that can be used as a
         basis for specific TLS server instances.";
      leaf address {
        type inet:ip-address;
        description
         "The IP address of the interface to listen on.  The TLS
          server will listen on all interfaces if no value is
          specified. Please note that some addresses have special
          meanings (e.g., '0.0.0.0' and '::').";
      }
      leaf port {
        type inet:port-number;
        description
         "The local port number on this interface the TLS server
          listens on.  When this grouping is used, it is RECOMMENDED
          that refine statement is used to either set a default port
          value or to set mandatory true.";
      }
      uses non-listening-tls-server-grouping;
    }
  }


  <CODE ENDS>
```

4. Security Considerations

5. IANA Considerations

5.1. The IETF XML Registry

This document registers two URIs in the IETF XML registry [RFC2119]. Following the format in [RFC3688], the following registrations are requested:

```
URI: urn:ietf:params:xml:ns:yang:ietf-tls-client
Registrant Contact: The NETCONF WG of the IETF.
XML: N/A, the requested URI is an XML namespace.

URI: urn:ietf:params:xml:ns:yang:ietf-tls-server
Registrant Contact: The NETCONF WG of the IETF.
XML: N/A, the requested URI is an XML namespace.
```

5.2. The YANG Module Names Registry

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

```
name:         ietf-tls-client
namespace:    urn:ietf:params:xml:ns:yang:ietf-tls-client
prefix:       tlsc
reference:    RFC XXXX

name:         ietf-tls-server
namespace:    urn:ietf:params:xml:ns:yang:ietf-tls-server
prefix:       tlss
reference:    RFC XXXX
```

6. Acknowledgements

The authors would like to thank for following for lively discussions on list and in the halls (ordered by last name): Andy Bierman, Martin Bjorklund, Benoit Claise, Mehmet Ersue, David Lamparter, Alan Luchuk, Ladislav Lhotka, Radek Krejci, Tom Petch, Juergen Schoenwaelder, Phil Shafer, Sean Turner, and Bert Wijnen.

7. References

7.1. Normative References

[draft-ietf-netconf-keystore]
Watsen, K., "Keystore Model", draft-ieft-netconf-keystore-00 (work in progress), 2016, <https://datatracker.ietf.org/html/draft-ieft-netconf-keystore>.

[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>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <http://www.rfc-editor.org/info/rfc6020>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <http://www.rfc-editor.org/info/rfc6991>.

[RFC7589] Badra, M., Luchuk, A., and J. Schoenwaelder, "Using the NETCONF Protocol over Transport Layer Security (TLS) with Mutual X.509 Authentication", RFC 7589, DOI 10.17487/RFC7589, June 2015, <http://www.rfc-editor.org/info/rfc7589>.

7.2. Informative References

[draft-ietf-netconf-call-home] Watsen, K., "NETCONF Call Home and RESTCONF Call Home", draft-ieft-netconf-call-home-17 (work in progress), 2015, <https://datatracker.ietf.org/html/draft-ieft-netconf-call-home-17>.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, DOI 10.17487/RFC2818, May 2000, <http://www.rfc-editor.org/info/rfc2818>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <http://www.rfc-editor.org/info/rfc3688>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <http://www.rfc-editor.org/info/rfc5246>.

[RFC793] Postel, J., "TRANSMISSION CONTROL PROTOCOL", STD 7, September 1981, <https://www.ietf.org/rfc/rfc793.txt>.

Appendix A. Change Log

A.1. server-model-09 to 00

- This draft was split out from draft-ietf-netconf-server-model-09.
- Noted that '0.0.0.0' and '::' might have special meanings.

Appendix B. Open Issues

Please see: https://github.com/netconf-wg/tls-client-server/issues.

Author's Address

Kent Watsen
Juniper Networks

EMail: kwatsen@juniper.net