NETMOD Working Group Q. Sun
Internet-Draft I. Farrer
Intended status: Standards Track Deutsche Telekom AG
Expires: September 10, 2015 March 9, 2015

A YANG Data Model for Transferring Files
draft-sf-netmod-file-transfer-yang-00

Abstract

This document defines a YANG data model for the transfer of files between devices. The data model includes operation data and state data.

Requirements Language

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 [RFC2119].

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This document defines a YANG [RFC6020] data model for the transfer of files between devices.

The model augments the state YANG model for System Management [RFC7317] to query the status of the file transfer process and the downloaded files. It defines new RPCs, enabling the NETCONF client to instruct the NETCONF server to tranfer a file.

This YANG model can be used to configure a device, to upload or download files to a remote host. The model enables the operator to easily manage the content on a device via Netconf.

1.1. Terminology

1.1.1. Requirements Notations

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 [RFC2119].

1.1.2. NETCONF Terms

The following terms are defined in [RFC6241] and are not redefined here:

- Client

- Server

- Remote Procedure Call (RPC)

- Operation

1.1.3. YANG Terms

The following terms are defined in [RFC6020] and are not redefined here:

- Data node

- Module

- Namespace

- YANG

1.1.4. Tree Diagrams

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

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

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

This document defines a YANG data model that can be used to manage file transfer between devices. In order to collect and store the status of the transfer process and files, the model augments the state subtree of the System Management YANG model [RFC7317].

The model defines a new RPC, "transfer". The RPC provides the server with the action (upload/download), the URI of the target file, the credentials parameters and some other information about failover. The server will respond to the RPC with the index to find the file.

The basic procedure is that the client instructs the server to transfer a specific file to/from the provided destination. The client can then retrieve the state of the transfer operation.

3. File Transfer YANG Tree Diagrams

3.1. File Transfer State subtree

This section describes the state data model for file transfer, which augments the system-state subtree:

```
augment /sys:system-state:
   +--ro file-transfer-state
      +--ro file-transfered*
         +--ro index                 uint32
         +--ro file-name             string
         +--ro remote-file-path      inet:uri
         +--ro local-file-path       inet:uri
         +--ro credential-method?    enumeration
         +--ro transfer-status
         |  +--ro status             enumeration
         |  +--ro status-msg?        string
         |  +--ro progress           percent
         +--ro retried-times?        uint8
```

Figure 1: File Transfer State Tree

- o remote-file-path: a URI for the remote file path. This can be a URI of type HTTP/HTTPS/FTP/SCP, depending on the protocol which is being used for the transfer.

- o local-file-path: A URI for the local file path. This is a URI of type file.

- credential-method: indicates the method used for supplying credentials for the transfer.

- transfer-status: The status for the downloading/uploading.

3.2. File Transfer RPCs

This section describes the Remote Procedure Call data model, which is used to instruct the server to transfer a specific file:

```
   rpcs:
      +---x transfer
         +---w input
         |  +---w action                      enumeration
         |  +---w file-name?                  string
         |  +---w local-file-path             inet:uri
         |  +---w remote-file-path            inet:uri
         |  +---w (trans-method)
         |  |  +--:(scp)
         |  |  |  +---w scp! {transfer-scp}?
         |  |  +--:(ftp)
         |  |  |  +---w ftp! {transfer-ftp}?
         |  |  +--:(http)
         |  |  |  +---w http! {transfer-http}?
         |  |  +--:(https)
         |  |     +---w https! {transfer-https}?
         |  +---w (credentials)?
         |  |  +--:(password)
         |  |  |  +---w password!
         |  |  |     +---w password           string
         |  |  +--:(certificate)
         |  |  |  +---w certificate!
         |  |  |     +---w local-path         inet:uri
         |  |  +--:(ssh-key)
         |  |     +---w ssh-key!
         |  |        +---w local-path         inet:uri
         |  +---w credential-valid-time?      lifetime
         |  +---w md5-checksum?               string
         |  +---w timeframe?                  uint32
         |  +---w retry-times?                uint8
         |  +---w retry-interval?             uint32
         |  +---w retry-interval-increment?   percent
         +--ro output
            +--ro index    uint32
```

Figure 2: File Transfer RPC Tree

- action: Specifies the transfer is "upload" or "download".

- remote-file-path: A URI for the remote file path. This can be a URI of type HTTP/HTTPS/FTP/SCP, depending on the protocol which is being used for the transfer. The transfer protocol is specified in the choice of 'trans-method'.

- local-file-path: a URI for the local file path. This MUST be a URI of type file.

- trans-method: Specifies the method used for file transfer. This is necessary as different transfer methods require different formats of URI.

- credential: Specifies the credentials method used for the transfer.

- credential-valid-time: Indicates the valid time for authentication of the current connection.

- output: The server sends back the index of the current transfer process, which can be used by the client to collect information of that transferring procedure.

4. File Transfer YANG Model

This module imports typedefs from [RFC6991] and augments to the system-state in [RFC7317].

```
<CODE BEGINS> file "ietf-file-transfer@2015-03-09.yang"
module "file-transfer" {

        namespace "urn:ietf:params:xml:ns:yang:file-trans";
        prefix "ft";

        import "ietf-system" {
                prefix "sys";
        }
        import "ietf-inet-types" {
                prefix "inet";
        }
        description "File transfer YANG model";
        organization "NETMOD Working Group";
        contact
        "qui.sun@external.telekom.de
         ian.farrer@telekom.de
        ";
    revision "2015-03-09" {
```

```
        description
            "Initial version for file transfer YANG model.
             This model can be used for file transfering and uploading.
            ";
    }

        feature "transfer-scp" {
                description "Indicate the transfer is through scp";
        }
        feature "transfer-ftp" {
                description "Indicate the transfer is through ftp";
        }
        feature "transfer-http" {
                description "Indicate the transfer is through http";
        }
        feature "transfer-https" {
                description "Indicate the transfer is through https";
        }

    /* typedefs */
        typedef "percent" {
                type "uint8" {
                        range "0 .. 100";
                }
        }
    typedef "lifetime" {
        type uint32;
        units seconds;
    }

    /* augments */
        augment "/sys:system-state" {
                container "file-transfer-state" {
            config false;
                        list "file-transfered" {
                                leaf "index" {
                                        type "uint32";
                    mandatory true;
                                }
                                leaf "file-name" {
                                        type "string";
                    mandatory true;
                                }
                                leaf "remote-file-path" {
                                        type "inet:uri";
                    mandatory true;
                    description
                      "A URI for the remote file path. This can be a URI of
```

```
                       type HTTP/HTTPS/FTP/SCP, depending on the protocol
                       which is being used for the transfer.  The transfer
protocol
                       is indicated through the URI. ";
                                }
                                leaf "local-file-path" {
                                        type "inet:uri";
                    mandatory true;
                    description
                      "A URI for the local file path. This is a URI of type
file. ";
                                }
                                leaf "credential-method" {
                    type enumeration {
                        enum password;
                        enum certificate;
                        enum ssh-key;
                    }
                                }
                container "transfer-status" {
                    config false;
                    leaf status {
                        type enumeration {
                            enum planned;
                            enum in-progress;
                            enum transfer-failed;
                            enum md5check-failed;
                            enum verification-failed;
                            enum done;
                        }
                    mandatory true;
                    }
                    leaf status-msg {
                        type string;
                        description "Additional system messages.";
                    }
                    leaf progress {
                        when "../status = 'in-progress'";
                        type percent;
                        mandatory true;
                    }
                }
                leaf retried-times {
                    type uint8;
                }
                        }
                }
        }
```

```
    rpc transfer {
        input {
            leaf action {
                mandatory true;
                type enumeration {
                    enum upload {
                        description
                            "Specify the upload action. The server sends
                             the file identified by the local-file-path
                             to the URI identified by the remote-file-path.";
                    }
                    enum download {
                        description
                            "Specify the download action. The server retrives
                            the file identified by the remote-file-path
                            to the URI identified by the local-file-path.";
                    }
                }
            }
            leaf file-name {
                type string;
            }
            leaf local-file-path {
                type inet:uri;
                mandatory true;
                description
                    "A URI for the local file path. This MUST be a URI
                    of type file.
                    ";
            }
            leaf remote-file-path {
                type inet:uri;
                mandatory true;
                description
                  "A URI for the remote file path. This can be a URI of
                   type HTTP/HTTPS/FTP/SCP, depending on the protocol
                   which is being used for the transfer. The transfer
                   protocol is specified in the choice of 'trans-method'.";
            }
                        choice "trans-method" {
                mandatory true;
                case "scp" {
                                    container "scp" {
                                        presence "true";
                        if-feature "transfer-scp";
                                    }
                }
                case "ftp" {
                                    container "ftp" {
                                            presence "true";
```

```
                        if-feature "transfer-ftp";
                                    }
                }
                                case "http" {
                    container "http" {
                                            presence "true";
                        if-feature "transfer-http";
                    }
                }
                case "https" {
                                    container "https" {
                                            presence "true";
                        if-feature "transfer-https";
                                    }
                }
            }
                        choice "credentials" {
                                case "password" {
                    container "password" {
                        presence true;
                        leaf password {
                            type string;
                            mandatory true;
                        }
                    }
                }
                case "certificate" {
                    container "certificate" {
                        presence "true";
                        leaf local-path {
                            type inet:uri;
                            mandatory true;
                        }
                    }
                }
                case "ssh-key" {
                    container "ssh-key" {
                        presence "true";
                        leaf local-path {
                            type inet:uri;
                            mandatory true;
                        }
                    }
                }
                        }
            leaf credential-valid-time {
                type lifetime;
                description
```

```
                    "The valid duration for the credential method
                     above.";
            }
            leaf md5-checksum {
                type string;
                description
                    "MD5 checking is optional.";
            }
            leaf timeframe {
                type uint32;
            }
            leaf retry-times {
                type uint8;
            }
            leaf retry-interval {
                type uint32;
            }
            leaf retry-interval-increment {
                type percent;
            }
        }
        output {
            leaf index {
                type uint32;
                mandatory true;
            }
            description
                " When the server receives the RPC request of
                  'transfer', if the transfering process starts
                  successfully, the server MUST assign an index
                  to that process and reply it to the client as
                  an output parameter.
                ";
        }
    }
}



<CODE ENDS>
```

5. Security Considerations

The YANG module defined in this memo is designed to be accessed via the NETCONF protocol [RFC6241]. The lowest NETCONF layer is the secure transport layer and the mandatory to implement secure transport is SSH [RFC6242]. The NETCONF access control model [RFC6536] provides the means to restrict access for particular

NETCONF users to a pre-configured subset of all available NETCONF protocol operations and content.

There are a number of data nodes defined in this YANG module which 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:

subtrees and data nodes and state why they are sensitive

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. These are the subtrees and data nodes and their sensitivity/vulnerability:

subtrees and data nodes and state why they are sensitive

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:

subtrees and data nodes and state why they are sensitive

6. IANA Considerations

TBD

7. Acknowledgements

The authors would like to thank Calvin Cheng for his valuable inputs.

8. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6021] Schoenwaelder, J., "Common YANG Data Types", RFC 6021, October 2010.

[RFC6087] Bierman, A., "Guidelines for Authors and Reviewers of YANG Data Model Documents", RFC 6087, January 2011.

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, June 2011.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, March 2012.

[RFC6991] Schoenwaelder, J., "Common YANG Data Types", RFC 6991, July 2013.

[RFC7317] Bierman, A. and M. Bjorklund, "A YANG Data Model for System Management", RFC 7317, August 2014.

Authors' Addresses

Qi Sun
Deutsche Telekom AG
CTO-ATI,Landgrabenweg 151
Bonn, NRW 53227
Germany

Email: qui.sun@external.telekom.de

Ian Farrer
Deutsche Telekom AG
CTO-ATI,Landgrabenweg 151
Bonn, NRW 53227
Germany

Email: ian.farrer@telekom.de