NETMOD WG
Internet-Draft
Intended status: Standards Track
Expires: August 13, 2018

C. Wildes, Ed.
Cisco Systems Inc.
K. Koushik, Ed.
Verizon Wireless

February 09, 2018

A YANG Data Model for Syslog Configuration
draft-ietf-netmod-syslog-model-20

Abstract

This document defines a YANG data model for the configuration of a syslog process. It is intended this model be used by vendors who implement syslog in their systems.

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 shorthand references to drafts in progress. Please apply the following replacements:

- "I-D.ietf-netconf-keystore" --> the assigned RFC value for draft-ietf-netconf-keystore

- "I-D.ietf-netconf-tls-client-server" --> the assigned RFC value for draft-ietf-netconf-tls-client-server

- "zzzz" --> the assigned RFC value for this draft

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 August 13, 2018.

Copyright Notice

Copyright (c) 2018 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 3
  1.1. Requirements Language 3
  1.2. Terminology 3
2. Problem Statement 4
3. Design of the Syslog Model 4
  3.1. Syslog Module 6
4. Syslog YANG Module 8
  4.1. The ietf-syslog Module 8
5. Usage Examples 27
6. Acknowledgements 27
7. IANA Considerations 28
  7.1. The IETF XML Registry 28
  7.2. The YANG Module Names Registry 28
8. Security Considerations 28
9. References 29
  9.1. Normative References 29
  9.2. Informative References 30
Appendix A. Implementor Guidelines 32
  A.1. Extending Facilities 32
Authors' Addresses 33

1. Introduction

Operating systems, processes and applications generate messages indicating their own status or the occurrence of events. These messages are useful for managing and/or debugging the network and its services. The BSD syslog protocol is a widely adopted protocol that is used for transmission and processing of the message.

Since each process, application and operating system was written somewhat independently, there is little uniformity to the content of syslog messages. For this reason, no assumption is made upon the formatting or contents of the messages. The protocol is simply designed to transport these event messages. No acknowledgement of the receipt is made.

Essentially, a syslog process receives messages (from the kernel, processes, applications or other syslog processes) and processes them. The processing may involve logging to a local file, and/or displaying on console, and/or relaying to syslog processes on other machines. The processing is determined by the "facility" that originated the message and the "severity" assigned to the message by the facility.

We are using definitions of syslog protocol from [RFC5424] in this RFC.

1.1. Requirements Language

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.

1.2. Terminology

The term "originator" is defined in [RFC5424]: an "originator" generates syslog content to be carried in a message.

The term "relay" is defined in [RFC5424]: a "relay" forwards messages, accepting messages from originators or other relays and sending them to collectors or other relays

The term "collectors" is defined in [RFC5424]: a "collector" gathers syslog content for further analysis.

The term "action" refers to the processing that takes place for each syslog message received.

2. Problem Statement

This document defines a YANG [RFC7950] configuration data model that may be used to configure the syslog feature running on a system. YANG models can be used with network management protocols such as NETCONF [RFC6241] to install, manipulate, and delete the configuration of network devices.

The data model makes use of the YANG "feature" construct which allows implementations to support only those syslog features that lie within their capabilities.

This module can be used to configure the syslog application conceptual layers as implemented on the target system.

3. Design of the Syslog Model

The syslog model was designed by comparing various syslog features implemented by various vendors' in different implementations.

This draft addresses the common leafs between implementations and creates a common model, which can be augmented with proprietary features, if necessary. This model is designed to be very simple for maximum flexibility.

Some optional features are defined in this document to specify functionality that is present in specific vendor configurations.

Syslog consists of originators and collectors. The following diagram shows syslog messages flowing from originators, to collectors where filtering can take place.

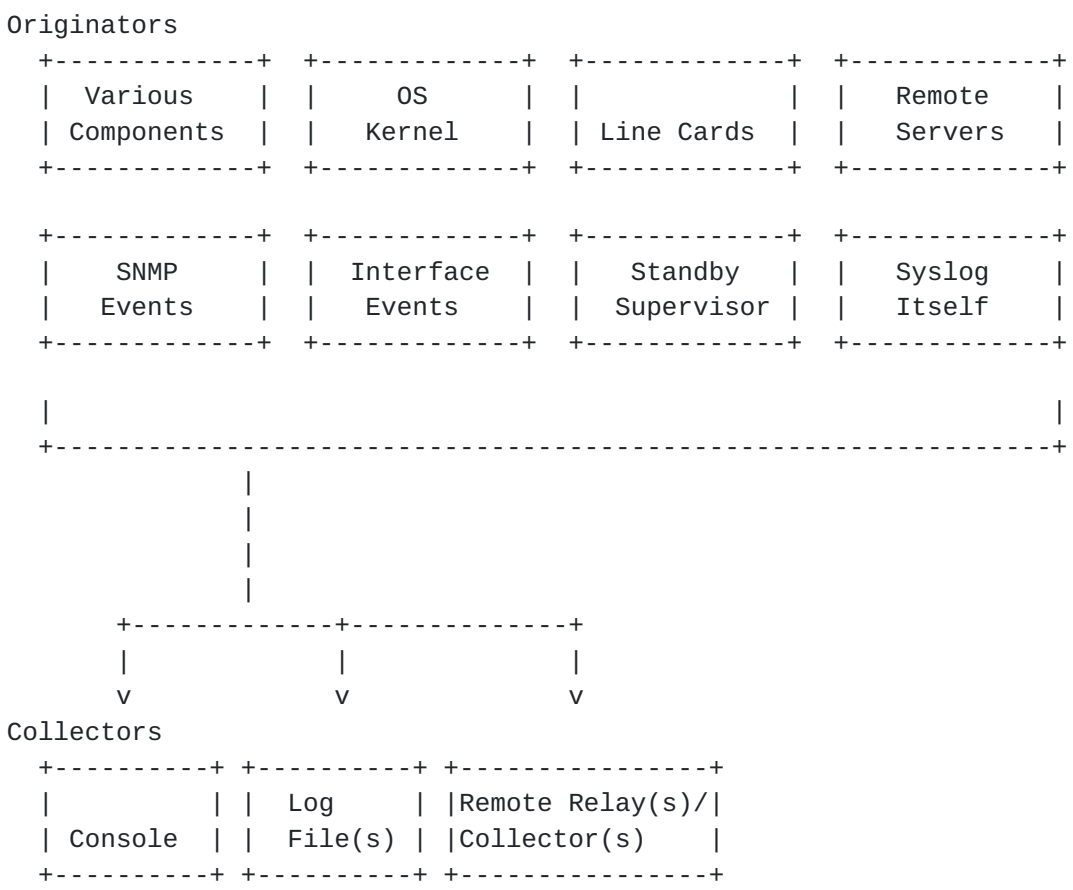


Figure 1. Syslog Processing Flow

Collectors are configured using the leaves in the syslog model "actions" container which correspond to each message collector:

console

log file(s)

remote relay(s)/collector(s)

Within each action, a selector is used to filter syslog messages. A selector consists of a list of one or more facility-severity matches, and, if supported via the select-match feature, an optional regular expression pattern match that is performed on the [RFC5424] field.

A syslog message is processed if:

```
        There is an element of facility-list (F, S) where
            the message facility matches F
            and the message severity matches S
        and/or the message text matches the regex pattern (if it
            is present)
```

The facility is one of a specific syslog-facility, or all facilities.

The severity is one of type syslog-severity, all severities, or none. None is a special case that can be used to disable a filter. When filtering severity, the default comparison is that messages of the specified severity and higher are selected to be logged. This is shown in the model as "default equals-or-higher". This behavior can be altered if the select-adv-compare feature is enabled to specify a compare operation and an action. Compare operations are: "equals" to select messages with this single severity, or "equals-or-higher" to select messages of the specified severity and higher. Actions are used to log the message or block the message from being logged.

Many vendors extend the list of facilities available for logging in their implementation. An example is included in Extending Facilities (Appendix A.1).

3.1. Syslog Module

A simplified graphical representation of the data model is used in this document. Please see [I-D.ietf-netmod-yang-tree-diagrams] for tree diagram notation.

```
module: ietf-syslog
   +--rw syslog!
      +--rw actions
         +--rw console! {console-action}?
         |  +--rw facility-filter
         |  |  +--rw facility-list* [facility severity]
         |  |     +--rw facility            union
         |  |     +--rw severity            union
         |  |     +--rw advanced-compare {select-adv-compare}?
         |  |        +--rw compare?   enumeration
         |  |        +--rw action?    enumeration
         |  +--rw pattern-match?     string {select-match}?
         +--rw file {file-action}?
         |  +--rw log-file* [name]
         |     +--rw name               inet:uri
         |     +--rw facility-filter
         |     |  +--rw facility-list* [facility severity]
         |     |     +--rw facility            union
         |     |     +--rw severity            union
```

```
|     |     +--rw advanced-compare {select-adv-compare}?
|     |        +--rw compare?   enumeration
|     |        +--rw action?    enumeration
|     +--rw pattern-match?     string {select-match}?
|     +--rw structured-data?   boolean {structured-data}?
|     +--rw file-rotation
|        +--rw number-of-files?   uint32 {file-limit-size}?
|        +--rw max-file-size?     uint32 {file-limit-size}?
|        +--rw rollover?          uint32
|        |       {file-limit-duration}?
|        +--rw retention?         uint32
|                 {file-limit-duration}?
+--rw remote {remote-action}?
   +--rw destination* [name]
      +--rw name                 string
      +--rw (transport)
      |  +--:(udp)
      |  |  +--rw udp
      |  |     +--rw address?   inet:host
      |  |     +--rw port?      inet:port-number
      |  +--:(tls)
      |     +--rw tls
      |        +--rw address?        inet:host
      |        +--rw port?           inet:port-number
      |        +--rw client-auth
      |        |  +--rw (auth-type)?
      |        |     +--:(certificate)
      |        |        +--rw certificate?   leafref
      |        +--rw server-auth
      |        |  +--rw pinned-ca-certs?       leafref
      |        |  +--rw pinned-server-certs?   leafref
      |        +--rw hello-params
      |                {tls-client-hello-params-config}?
      |           +--rw tls-versions
      |           |  +--rw tls-version*   identityref
      |           +--rw cipher-suites
      |              +--rw cipher-suite*   identityref
      +--rw facility-filter
      |  +--rw facility-list* [facility severity]
      |     +--rw facility            union
      |     +--rw severity            union
      |     +--rw advanced-compare {select-adv-compare}?
      |        +--rw compare?   enumeration
      |        +--rw action?    enumeration
      +--rw pattern-match?       string {select-match}?
      +--rw structured-data?     boolean {structured-data}?
      +--rw facility-override?   identityref
      +--rw source-interface?    if:interface-ref
```

```
|       {remote-source-interface}?
+--rw signing! {signed-messages}?
   +--rw cert-signers
      +--rw cert-signer* [name]
      |  +--rw name              string
      |  +--rw cert
      |  |  +--rw algorithm?
      |  |  |       identityref
      |  |  +--rw private-key?
      |  |  |       union
      |  |  +--rw public-key?
      |  |  |       binary
      |  |  +---x generate-private-key
      |  |  |  +---w input
      |  |  |     +---w algorithm?
      |  |  |             identityref
      |  |  +--rw certificates
      |  |  |  +--rw certificate* [name]
      |  |  |     +--rw name     string
      |  |  |     +--rw value?   binary
      |  |  +---x generate-certificate-signing-request
      |  |     +---w input
      |  |     |  +---w subject       binary
      |  |     |  +---w attributes?   binary
      |  |     +--ro output
      |  |        +--ro certificate-signing-request
      |  |                binary
      |  +--rw hash-algorithm?   enumeration
      +--rw cert-initial-repeat?   uint32
      +--rw cert-resend-delay?     uint32
      +--rw cert-resend-count?     uint32
      +--rw sig-max-delay?         uint32
      +--rw sig-number-resends?    uint32
      +--rw sig-resend-delay?      uint32
      +--rw sig-resend-count?      uint32
```

Figure 2. ietf-syslog Module Tree

4. Syslog YANG Module

4.1. The ietf-syslog Module

This module imports typedefs from [RFC7223], groupings from [I-D.ietf-netconf-keystore], and [I-D.ietf-netconf-tls-client-server], and it references [RFC5424], [RFC5425], [RFC5426], and [RFC5848] and [Std-1003.1-2008].

<CODE BEGINS> file "ietf-syslog@2018-02-09.yang"

```
   module ietf-syslog {
     yang-version 1.1;

     namespace "urn:ietf:params:xml:ns:yang:ietf-syslog";
     prefix syslog;

     import ietf-inet-types {
       prefix inet;
       reference
         "RFC 6991: INET Types Model";
     }

     import ietf-interfaces {
       prefix if;
       reference
         "RFC 7223: Interfaces Model";
     }

     import ietf-tls-client {
       prefix tlsc;
       reference
         "I-D.ietf-netconf-tls-client-server:
          TLS Client and Server Models";
     }

     import ietf-keystore {
       prefix ks;
       reference
         "I-D.ietf-netconf-keystore: Keystore Model";
     }

     organization
         "IETF NETMOD (Network Modeling) Working Group";

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

        Editor:   Kiran Agrahara Sreenivasa
                  <mailto:kirankoushik.agraharasreenivasa@
                          verizonwireless.com>

        Editor:   Clyde Wildes
                  <mailto:cwildes@cisco.com>";
     description
       "This module contains a collection of YANG definitions
        for syslog configuration.
```

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).

The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL', 'SHALL
NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'MAY', and
'OPTIONAL' in the module text are to be interpreted as
described in RFC 2119 (http://tools.ietf.org/html/rfc2119).

This version of this YANG module is part of RFC zzzz
(http://tools.ietf.org/html/rfczzzz); see the RFC itself for
full legal notices.";

```
  revision 2018-02-09 {
    description
      "Initial Revision";
    reference
      "RFC zzzz: Syslog YANG Model";
  }

  feature console-action {
    description
      "This feature indicates that the local console action is
       supported.";
  }

  feature file-action {
    description
      "This feature indicates that the local file action is
       supported.";
  }

  feature file-limit-size {
    description
      "This feature indicates that file logging resources
       are managed using size and number limits.";
  }

  feature file-limit-duration {
    description
      "This feature indicates that file logging resources
       are managed using time based limits.";
```

```
    }

    feature remote-action {
      description
        "This feature indicates that the remote server action is
         supported.";
    }

    feature remote-source-interface {
      description
        "This feature indicates that source-interface is supported
         supported for the remote-action.";
    }

    feature select-adv-compare {
      description
        "This feature represents the ability to select messages
         using the additional comparison operators when comparing
         the syslog message severity.";
    }

    feature select-match {
      description
        "This feature represents the ability to select messages
         based on a Posix 1003.2 regular expression pattern match.";
    }

    feature structured-data {
      description
        "This feature represents the ability to log messages
         in structured-data format.";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    feature signed-messages {
      description
        "This feature represents the ability to configure signed
         syslog messages.";
      reference
        "RFC 5848: Signed Syslog Messages";
    }

    typedef syslog-severity {
      type enumeration {
        enum "emergency" {
          value 0;
          description
```

```
            "The severity level 'Emergency' indicating that the
             system is unusable.";
          }
          enum "alert" {
            value 1;
            description
              "The severity level 'Alert' indicating that an action
               must be taken immediately.";
          }
          enum "critical" {
            value 2;
            description
              "The severity level 'Critical' indicating a critical
               condition.";
          }
          enum "error" {
            value 3;
            description
              "The severity level 'Error' indicating an error
               condition.";
          }
          enum "warning" {
            value 4;
            description
              "The severity level 'Warning' indicating a warning
               condition.";
          }
          enum "notice" {
            value 5;
            description
              "The severity level 'Notice' indicating a normal but
               significant condition.";
          }
          enum "info" {
            value 6;
            description
              "The severity level 'Info' indicating an informational
               message.";
          }
          enum "debug" {
            value 7;
            description
              "The severity level 'Debug' indicating a debug-level
               message.";
          }
        }
        description
          "The definitions for Syslog message severity.";
```

```
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity syslog-facility {
      description
        "This identity is used as a base for all syslog facilities.";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity kern {
      base syslog-facility;
      description
        "The facility for kernel messages (0).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity user {
      base syslog-facility;
      description
        "The facility for user-level messages (1).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity mail {
      base syslog-facility;
      description
        "The facility for the mail system (2).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity daemon {
      base syslog-facility;
      description
        "The facility for the system daemons (3).";
      reference
        "RFC 5424: The Syslog Protocol";
     }

    identity auth {
      base syslog-facility;
      description
        "The facility for security/authorization messages (4).";
      reference
```

```
        "RFC 5424: The Syslog Protocol";
    }

    identity syslog {
      base syslog-facility;
      description
        "The facility for messages generated internally by syslogd
         facility (5).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity lpr {
      base syslog-facility;
      description
        "The facility for the line printer subsystem (6).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity news {
      base syslog-facility;
      description
        "The facility for the network news subsystem (7).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity uucp {
      base syslog-facility;
      description
        "The facility for the UUCP subsystem (8).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity cron {
      base syslog-facility;
      description
        "The facility for the clock daemon (9).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity authpriv {
      base syslog-facility;
      description
        "The facility for privileged security/authorization messages
```

```
          (10).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity ftp {
      base syslog-facility;
      description
        "The facility for the FTP daemon (11).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity ntp {
      base syslog-facility;
      description
        "The facility for the NTP subsystem (12).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity audit {
      base syslog-facility;
      description
        "The facility for log audit messages (13).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity console {
      base syslog-facility;
      description
        "The facility for log alert messages (14).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity cron2 {
      base syslog-facility;
      description
        "The facility for the second clock daemon (15).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity local0 {
      base syslog-facility;
      description
```

```
        "The facility for local use 0 messages (16).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity local1 {
      base syslog-facility;
      description
        "The facility for local use 1 messages (17).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity local2 {
      base syslog-facility;
      description
        "The facility for local use 2 messages (18).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity local3 {
      base syslog-facility;
      description
        "The facility for local use 3 messages (19).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity local4 {
      base syslog-facility;
      description
        "The facility for local use 4 messages (20).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity local5 {
      base syslog-facility;
      description
        "The facility for local use 5 messages (21).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity local6 {
      base syslog-facility;
      description
```

```
        "The facility for local use 6 messages (22).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    identity local7 {
      base syslog-facility;
      description
        "The facility for local use 7 messages (23).";
      reference
        "RFC 5424: The Syslog Protocol";
    }

    grouping severity-filter {
      description
        "This grouping defines the processing used to select
         log messages by comparing syslog message severity using
         the following processing rules:
          - if 'none', do not match.
          - if 'all', match.
          - else compare message severity with the specified severity
            according to the default compare rule (all messages of the
            specified severity and greater match) or if the
            select-adv-compare feature is present, the advance-compare
            rule.";
      leaf severity {
        type union {
          type syslog-severity;
          type enumeration {
            enum none {
              value 2147483647;
              description
                "This enum describes the case where no severities
                 are selected.";
            }
            enum all {
              value -2147483648;
              description
                "This enum describes the case where all severities
                 are selected.";
            }
          }
        }
        mandatory true;
        description
          "This leaf specifies the syslog message severity.";
      }
      container advanced-compare {
```

```
          when '../severity != "all" and
               ../severity != "none"' {
            description
              "The advanced compare container is not applicable for
               severity 'all' or severity 'none'";
          }
          if-feature select-adv-compare;
          leaf compare {
            type enumeration {
              enum equals {
                description
                  "This enum specifies that the severity comparison
                   operation will be equals.";
              }
              enum equals-or-higher {
                description
                  "This enum specifies that the severity comparison
                   operation will be equals or higher.";
              }
            }
            default equals-or-higher;
            description
              "The compare can be used to specify the comparison
               operator that should be used to compare the syslog message
               severity with the specified severity.";
          }
          leaf action {
            type enumeration {
              enum log {
                description
                  "This enum specifies that if the compare operation is
                   true the message will be logged.";
               }
              enum block {
                description
                  "This enum specifies that if the compare operation is
                   true the message will not be logged.";
              }
            }
            default log;
            description
              "The action can be used to spectify if the message should
               be logged or blocked based on the outcome of the compare
               operation.";
          }
          description
            "This container describes additional severity compare
             operations that can be used in place of the default
```

```
            severity comparison. The compare leaf specifies the type of
            the compare that is done and the action leaf specifies the
            intended result.
            Example: compare->equals and action->no-match means
            messages that have a severity that is not equal to the
            specified severity will be logged.";
        }
      }

      grouping selector {
        description
          "This grouping defines a syslog selector which is used to
           select log messages for the log-actions (console, file,
           remote, etc.). Choose one or both of the following:
             facility [<facility> <severity>...]
             pattern-match regular-expression-match-string
           If both facility and pattern-match are specified, both must
           match in order for a log message to be selected.";
        container facility-filter {
          description
            "This container describes the syslog filter parameters.";
          list facility-list {
            key "facility severity";
            ordered-by user;
            description
              "This list describes a collection of syslog
               facilities and severities.";
            leaf facility {
              type union {
                type identityref {
                  base syslog-facility;
                }
                type enumeration {
                  enum all {
                    description
                      "This enum describes the case where all
                       facilities are requested.";
                  }
                }
              }
              description
                "The leaf uniquely identifies a syslog facility.";
            }
            uses severity-filter;
          }
        }
        leaf pattern-match {
          if-feature select-match;
```

```
        type string;
        description
          "This leaf describes a Posix 1003.2 regular expression
           string that can be used to select a syslog message for
           logging. The match is performed on the SYSLOG-MSG field.";
        reference
          "RFC 5424: The Syslog Protocol
           Std-1003.1-2008 Regular Expressions";
      }
    }

    grouping structured-data {
      description
        "This grouping defines the syslog structured data option
         which is used to select the format used to write log
         messages.";
      leaf structured-data {
        if-feature structured-data;
        type boolean;
        default false;
        description
          "This leaf describes how log messages are written.
           If true, messages will be written with one or more
           STRUCTURED-DATA elements; if false, messages will be
           written with STRUCTURED-DATA = NILVALUE.";
        reference
          "RFC 5424: The Syslog Protocol";
      }
    }

    container syslog {
      presence "Enables logging.";
      description
        "This container describes the configuration parameters for
         syslog.";
      container actions {
        description
          "This container describes the log-action parameters
           for syslog.";
        container console {
          if-feature console-action;
          presence "Enables logging to the console";
          description
            "This container describes the configuration parameters
             for console logging.";
          uses selector;
        }
        container file {
```

```
            if-feature file-action;
            description
              "This container describes the configuration parameters for
               file logging. If file-archive limits are not supplied, it
               is assumed that the local implementation defined limits
               will be used.";
            list log-file {
              key "name";
              description
                "This list describes a collection of local logging
                 files.";
              leaf name {
                type inet:uri {
                  pattern 'file:.*';
                }
                description
                  "This leaf specifies the name of the log file which
                   MUST use the uri scheme file:.";
              }
              uses selector;
              uses structured-data;
              container file-rotation {
                description
                  "This container describes the configuration
                   parameters for log file rotation.";
                leaf number-of-files {
                  if-feature file-limit-size;
                  type uint32;
                  default 1;
                  description
                    "This leaf specifies the maximum number of log
                     files retained. Specify 1 for implementations
                     that only support one log file.";
                }
                leaf max-file-size {
                  if-feature file-limit-size;
                  type uint32;
                  units "megabytes";
                  description
                    "This leaf specifies the maximum log file size.";
                }
                leaf rollover {
                  if-feature file-limit-duration;
                  type uint32;
                  units "minutes";
                  description
                    "This leaf specifies the length of time that log
                     events should be written to a specific log file.
```

```
                Log events that arrive after the rollover period
                cause the current log file to be closed and a new
                log file to be opened.";
            }
            leaf retention {
              if-feature file-limit-duration;
              type uint32;
              units "hours";
              description
                "This leaf specifies the length of time that
                 completed/closed log event files should be stored
                 in the file system before they are deleted.";
            }
          }
        }
      }
      container remote {
        if-feature remote-action;
        description
          "This container describes the configuration parameters
           for forwarding syslog messages to remote relays or
           collectors.";
        list destination {
          key "name";
          description
            "This list describes a collection of remote logging
             destinations.";
          leaf name {
            type string;
            description
              "An arbitrary name for the endpoint to connect to.";
          }
          choice transport {
            mandatory true;
            description
              "This choice describes the transport option.";
            case udp {
              container udp {
                description
                  "This container describes the UDP transport
                   options.";
                reference
                  "RFC 5426: Transmission of Syslog Messages over
                   UDP";
                leaf address {
                  type inet:host;
                  description
                    "The leaf uniquely specifies the address of
```

```
                    the remote host. One of the following must be
                    specified: an ipv4 address, an ipv6 address,
                    or a host name.";
                }
                leaf port {
                  type inet:port-number;
                  default 514;
                  description
                    "This leaf specifies the port number used to
                     deliver messages to the remote server.";
                }
              }
            }
            case tls {
              container tls {
                description
                  "This container describes the TLS transport
                   options.";
                reference
                  "RFC 5425: Transport Layer Security (TLS)
                   Transport Mapping for Syslog ";
                leaf address {
                  type inet:host;
                  description
                    "The leaf uniquely specifies the address of
                     the remote host. One of the following must be
                     specified: an ipv4 address, an ipv6 address,
                     or a host name.";
                }
                leaf port {
                  type inet:port-number;
                  default 6514;
                  description
                    "TCP port 6514 has been allocated as the default
                     port for syslog over TLS.";
                }
                uses tlsc:tls-client-grouping;
              }
            }
          }
          uses selector;
          uses structured-data;
          leaf facility-override {
            type identityref {
              base syslog-facility;
            }
            description
              "If specified, this leaf specifies the facility used
```

```
          to override the facility in messages delivered to
          the remote server.";
      }
      leaf source-interface {
        if-feature remote-source-interface;
        type if:interface-ref;
        description
          "This leaf sets the source interface to be used to
           send messages to the remote syslog server. If not
           set, messages sent to a remote syslog server will
           contain the IP address of the interface the syslog
           message uses to exit the network element";
      }
      container signing {
        if-feature signed-messages;
        presence
          "If present, syslog-signing options is activated.";
        description
          "This container describes the configuration
           parameters for signed syslog messages.";
        reference
          "RFC 5848: Signed Syslog Messages";
        container cert-signers {
          description
            "This container describes the signing certificate
             configuration for Signature Group 0 which covers
             the case for administrators who want all Signature
             Blocks to be sent to a single destination.";
          list cert-signer {
            key "name";
            description
              "This list describes a collection of syslog
               message signers.";
            leaf name {
              type string;
              description
                "This leaf specifies the name of the syslog
                 message signer.";
            }
            container cert {
              uses ks:private-key-grouping;
              uses ks:certificate-grouping;
              description
               "This is the certificate that is periodically
                sent to the remote receiver. Selection of the
                certificate also implicitly selects the private
                key used to sign the syslog messages.";
            }
```

```
      leaf hash-algorithm {
        type enumeration {
          enum SHA1 {
            value 1;
            description
              "This enum describes the SHA1 algorithm.";
          }
          enum SHA256 {
            value 2;
            description
              "This enum describes the SHA256 algorithm.";
          }
        }
        description
          "This leaf describes the syslog signer hash
           algorithm used.";
      }
    }
    leaf cert-initial-repeat {
      type uint32;
      default 3;
      description
      "This leaf specifies the number of times each
       Certificate Block should be sent before the first
       message is sent.";
    }
    leaf cert-resend-delay {
      type uint32;
      units "seconds";
      default 3600;
      description
        "This leaf specifies the maximum time delay in
         seconds until resending the Certificate Block.";
    }
    leaf cert-resend-count {
      type uint32;
      default 0;
      description
        "This leaf specifies the maximum number of other
         syslog messages to send until resending the
         Certificate Block.";
    }
    leaf sig-max-delay {
      type uint32;
      units "seconds";
      default 60;
      description
        "This leaf specifies when to generate a new
```

```
                    Signature Block. If this many seconds have
                    elapsed since the message with the first message
                    number of the Signature Block was sent, a new
                    Signature Block should be generated.";
                }
                leaf sig-number-resends {
                  type uint32;
                  default 0;
                  description
                    "This leaf specifies the number of times a
                     Signature Block is resent. (It is recommended to
                     select a value of greater than 0 in particular
                     when the UDP transport RFC 5426 is used.).";
                }
                leaf sig-resend-delay {
                  type uint32;
                  units "seconds";
                  default 5;
                  description
                    "This leaf specifies when to send the next
                     Signature Block transmission based on time. If
                     this many seconds have elapsed since the previous
                     sending of this Signature Block, resend it.";
                }
                leaf sig-resend-count {
                  type uint32;
                  default 0;
                  description
                    "This leaf specifies when to send the next
                     Signature Block transmission based on a count.
                     If this many other syslog messages have been
                     sent since the previous sending of this
                     Signature Block, resend it. A value of 0 means
                     that you don't resend based on the number of
                     messages.";
                }
              }
            }
          }
        }
      }
    }
  }
  <CODE ENDS>
```

Figure 3. ietf-syslog Module

5. Usage Examples

```
Requirement:
Enable console logging of syslogs of severity critical

Here is the example syslog configuration xml:
<syslog xmlns="urn:ietf:params:xml:ns:yang:ietf-syslog">
  <actions>
    <console>
      <facility-filter>
        <facility-list>
          <facility>all</facility>
          <severity>critical</severity>
        </facility-list>
      </facility-filter>
    </console>
  </actions>
</syslog>

Enable remote logging of syslogs to udp destination
2001:db8:a0b:12f0::1 for facility auth, severity error

<syslog xmlns="urn:ietf:params:xml:ns:yang:ietf-syslog">
  <actions>
    <remote>
      <destination>
        <name>remote1</name>
        <udp>
          <address>foo.eample.com</address>
        </udp>
        <facility-filter>
          <facility-list>
            <facility>auth</facility>
            <severity>error</severity>
          </facility-list>
        </facility-filter>
      </destination>
    </remote>
  </actions>
</syslog>
```

Figure 4. ietf-syslog Examples

6. Acknowledgements

The authors wish to thank the following who commented on this proposal:

Andy Bierman, Martin Bjorklund, Alex Campbell, Alex Clemm, Jim Gibson, Jeffrey Haas, John Heasley, Giles Heron, Lisa Huang, Mahesh Jethanandani, Jeffrey K Lange, Jan Lindblad, Chris Lonvick, Tom Petch, Juergen Schoenwaelder, Phil Shafer, Jason Sterne, Peter Van Horne, Kent Watsen, Bert Wijnen, Dale R Worley, Aleksandr Zhdankin

7. IANA Considerations

7.1. The IETF XML Registry

This document registers one URI in the IETF XML registry [RFC3688]. Following the format in [RFC3688], the following registration is requested:

```
URI: urn:ietf:params:xml:ns:yang:ietf-syslog
Registrant Contact: The IESG.
XML: N/A, the requested URI is an XML namespace.
```

7.2. The YANG Module Names Registry

This document registers one YANG module in the YANG Module Names registry [RFC7895]. Following the format in [RFC7950], the following registration is requested:

```
name:         ietf-syslog
namespace:    urn:ietf:params:xml:ns:yang:ietf-syslog
prefix:       ietf-syslog
reference:    RFC zzzz
```

8. Security Considerations

The YANG module defined in this document is 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, TLS) with mutual authentication.

The NETCONF access control model (NACM) [RFC6536] provides the means to restrict access for particular users to a pre-configured subset of all available protocol operations and content.

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

facility-filter/pattern-match: When writing this node, implementations MUST ensure that the regular expression pattern match is not constructed to cause a regular expression denial of service attack due to a pattern that causes the regular expression implementation to work very slowly (exponentially related to input size).

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.

There are no RPC operations defined in this YANG module.

9. References

9.1. Normative References

[I-D.ietf-netconf-keystore]
Watsen, K., "YANG Data Model for a "Keystore" Mechanism", draft-ietf-netconf-keystore-04 (work in progress), October 2017.

[I-D.ietf-netconf-tls-client-server]
Watsen, K. and G. Wu, "YANG Groupings for TLS Clients and TLS Servers", draft-ietf-netconf-tls-client-server-05 (work in progress), October 2017.

[I-D.ietf-netmod-yang-tree-diagrams]
Bjorklund, M. and L. Berger, "YANG Tree Diagrams", draft-ietf-netmod-yang-tree-diagrams-05 (work in progress), January 2018.

[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>.

[RFC5424] Gerhards, R., "The Syslog Protocol", RFC 5424, DOI 10.17487/RFC5424, March 2009, <https://www.rfc-editor.org/info/rfc5424>.

[RFC5425] Miao, F., Ed., Ma, Y., Ed., and J. Salowey, Ed., "Transport Layer Security (TLS) Transport Mapping for Syslog", RFC 5425, DOI 10.17487/RFC5425, March 2009, <https://www.rfc-editor.org/info/rfc5425>.

[RFC5426] Okmianski, A., "Transmission of Syslog Messages over UDP", RFC 5426, DOI 10.17487/RFC5426, March 2009, <https://www.rfc-editor.org/info/rfc5426>.

[RFC5848] Kelsey, J., Callas, J., and A. Clemm, "Signed Syslog Messages", RFC 5848, DOI 10.17487/RFC5848, May 2010, <https://www.rfc-editor.org/info/rfc5848>.

[RFC7223] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 7223, DOI 10.17487/RFC7223, May 2014, <https://www.rfc-editor.org/info/rfc7223>.

[RFC7895] Bierman, A., Bjorklund, M., and K. Watsen, "YANG Module Library", RFC 7895, DOI 10.17487/RFC7895, June 2016, <https://www.rfc-editor.org/info/rfc7895>.

[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>.

[Std-1003.1-2008] The Open Group, ""Chapter 9: Regular Expressions". The Open Group Base Specifications Issue 6, IEEE Std 1003.1-2008, 2016 Edition.", September 2016, <http://pubs.opengroup.org/onlinepubs/9699919799/>.

9.2. Informative References

[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>.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, DOI 10.17487/RFC6536, March 2012, <https://www.rfc-editor.org/info/rfc6536>.

[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>.

Appendix A. Implementor Guidelines

A.1. Extending Facilities

Many vendors extend the list of facilities available for logging in their implementation. Additional facilities may not work with the syslog protocol as defined in [RFC5424] and hence such facilities apply for local syslog-like logging functionality.

The following is an example that shows how additional facilities could be added to the list of available facilities (in this example two facilities are added):

```
   module example-vendor-syslog-types {
     namespace "http://example.com/ns/vendor-syslog-types";
     prefix vendor-syslogtypes;

     import ietf-syslog {
       prefix syslogtypes;
     }

     organization "Example, Inc.";
     contact
       "Example, Inc.
        Customer Service

        E-mail: syslog-yang@example.com";

     description
       "This module contains a collection of vendor-specific YANG type
        definitions for SYSLOG.";

     revision 2017-08-11 {
       description
         "Version 1.0";
       reference
         "Vendor SYSLOG Types: SYSLOG YANG Model";
     }

     identity vendor_specific_type_1 {
       base syslogtypes:syslog-facility;
       description
         "Adding vendor specific type 1 to syslog-facility";
     }

     identity vendor_specific_type_2 {
       base syslogtypes:syslog-facility;
       description
         "Adding vendor specific type 2 to syslog-facility";
     }
   }
```

Authors' Addresses

Clyde Wildes (editor)
Cisco Systems Inc.
170 West Tasman Drive
San Jose, CA 95134
US

Phone: +1 408 527-2672
EMail: cwildes@cisco.com

Kiran Koushik (editor)
Verizon Wireless
500 W Dove Rd.
Southlake, TX 76092
US

Phone: +1 512 650-0210
EMail: kirankoushik.agraharasreenivasa@verizonwireless.com