

NETMOD WG
Internet-Draft
Intended status: Informational
Expires: Apr 16, 2016

Clyde Wildes
Kiran Koushik
Cisco Systems Inc.
Oct 16, 2015

SYSLOG YANG model
draft-ietf-netmod-syslog-model-05

Abstract

This document describes a data model for Syslog protocol which is used to convey event notification messages.

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 <http://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 Jan 06, 2016.

Copyright Notice

Copyright (c) 2015 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 (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1. Introduction](#) [2](#)
- [1.1. Definitions and Acronyms](#) [3](#)
- [2. Problem Statement](#) [3](#)
- [3. Design of the SYSLOG Model](#) [3](#)
- [3.1. SYSLOG Module](#) [4](#)
- [4. SYSLOG YANG Models](#) [6](#)
- [4.1. SYSLOG TYPES Module](#) [6](#)
- [4.2. SYSLOG module](#) [10](#)
- [4.3. A SYSLOG Example](#) [18](#)
- [5. Implementation Status](#) [19](#)
- [6. Security Considerations](#) [19](#)
- [7. IANA Considerations](#) [20](#)
- [8. Acknowledgements](#) [20](#)
- [9. Change log \[RFC Editor: Please remove\]](#) [20](#)
- [10. References](#) [20](#)
- Authors' Addresses [21](#)

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 messages.

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 those. The processing involves logging to a local file, displaying on console, user terminal, 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 [[RFC3164](#)] in this draft.

1.1. Definitions and Acronyms

IP: Internet Protocol

IPv4: Internet Protocol version 4

IPv6: Internet Protocol version 6

UDP: User Datagram Protocol

VRF: Virtual Routing and Forwarding

2. Problem Statement

This document defines a YANG [[RFC6020](#)] configuration data model that may be used to monitor and control one or more syslog processes running on a system. YANG models can be used with network management agents such as NETCONF [[RFC6241](#)] to install, manipulate, and delete the configuration of network devices.

This module makes use of the YANG "feature" construct which allows implementations to support only those Syslog features that lie within their capabilities.

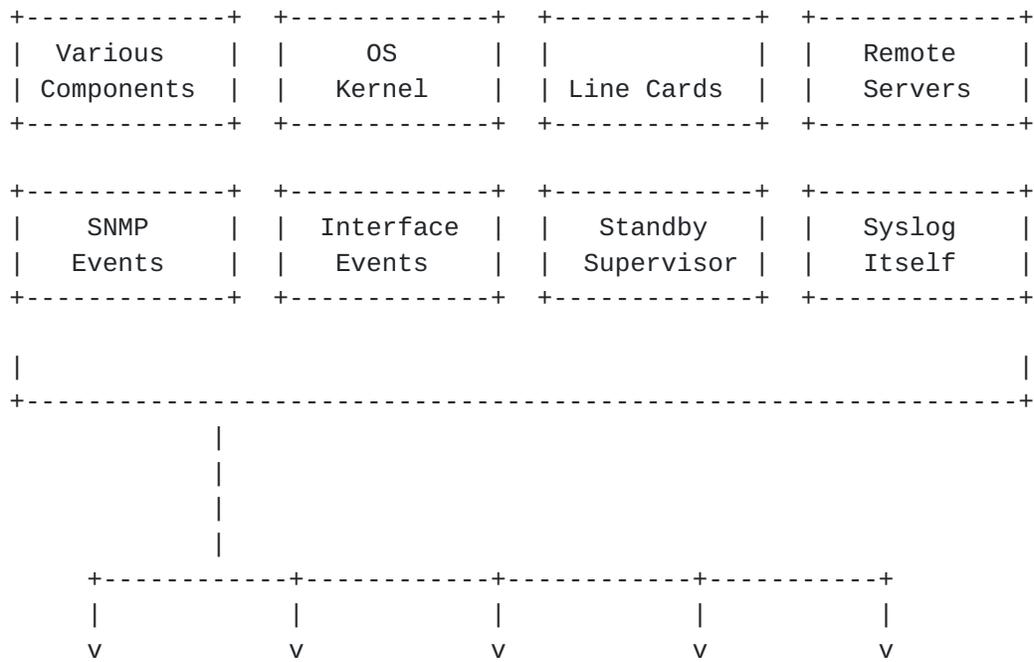
3. Design of the SYSLOG Model

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

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

Syslog consists of message producers, a group level suppression filter, and message distributors. The following diagram shows syslog messages flowing from a message producer, through the group level suppression filter, and if passed by the group filter to message distributors where further suppression filtering can take place.

Message Producers



Message Distributors



The leaves in the base syslog model correspond to the group level suppression filter and each message distributor:

- console
- log buffer
- log file(s)
- user terminals
- remote server(s).

Optional features are used to specified fields that are not present in all vendor configurations.

3.1. SYSLOG Module

```

module: ietf-syslog
  +-rw syslog
    +-rw log-actions
      +-rw console!
        | +-rw log-selector
        |   +-rw (selector-facility)
        |     | +--:(no-log-facility)
        |     | | +-rw no-facilities? empty
        |     | | +--:(log-facility)
        |     |   +-rw log-facility* [facility]
        |     |     +-rw facility union
        |     |     +-rw severity union
        |     |     +-rw severity-operator? enumeration {selector-
severity-operator-config}?
        |     +-rw pattern-match? string {selector-match-processing-
config}?
        +-rw buffer
          | +-rw log-buffer* [name]
          |   +-rw name string
          |   +-rw log-selector
          |     | +-rw (selector-facility)
          |     | | +--:(no-log-facility)
          |     | | | +-rw no-facilities? empty
          |     | | | +--:(log-facility)
          |     | |   +-rw log-facility* [facility]
          |     | |     +-rw facility union
          |     | |     +-rw severity union
          |     | |     +-rw severity-operator? enumeration {selector-
severity-operator-config}?
          |     | +-rw pattern-match? string {selector-match-processing-
config}?
          |     +-rw buffer-size-bytes? uint64 {buffer-limit-bytes}?
          |     +-rw buffer-size-messages? uint64 {buffer-limit-messages}?
          +-rw file
            | +-rw log-file* [name]
            |   +-rw name inet:uri
            |   +-rw log-selector
            |     | +-rw (selector-facility)
            |     | | +--:(no-log-facility)
            |     | | | +-rw no-facilities? empty
            |     | | | +--:(log-facility)
            |     | |   +-rw log-facility* [facility]
            |     | |     +-rw facility union
            |     | |     +-rw severity union
            |     | |     +-rw severity-operator? enumeration {selector-
severity-operator-config}?
            |     +-rw pattern-match? string {selector-match-processing-
config}?

```

```

|     +--rw structured-data?    boolean {structured-data-config}?
|     +--rw file-archive
|         +--rw number-of-files?  uint32 {file-limit-size}?
|         +--rw max-file-size?    uint64 {file-limit-size}?
|         +--rw rollover?         uint32 {file-limit-duration}?
|         +--rw retention?        uint16 {file-limit-duration}?
+--rw remote
|   +--rw destination* [name]
|     +--rw name                  string
|     +--rw (transport)
|       | +--:(tcp)
|       | | +--rw tcp
|       | |   +--rw address?    inet:host
|       | |   +--rw port?      inet:port-number
|       | +--:(udp)
|       |   +--rw udp
|       |     +--rw address?    inet:host
|       |     +--rw port?      inet:port-number
|     +--rw log-selector
|       | +--rw (selector-facility)
|       | | +--:(no-log-facility)
|       | | | +--rw no-facilities?  empty
|       | | +--:(log-facility)
|       | |   +--rw log-facility* [facility]
|       | |     +--rw facility      union
|       | |     +--rw severity     union
|       | |     +--rw severity-operator?  enumeration {selector-
severity-operator-config}?
|       | |   +--rw pattern-match?  string {selector-match-processing-
config}?
|       +--rw destination-facility?  identityref
|       +--rw source-interface?      if:interface-ref
|     +--rw syslog-sign! {signed-messages-config}?
|       +--rw cert-initial-repeat    uint16
|       +--rw cert-resend-delay      uint16
|       +--rw cert-resend-count      uint16
|       +--rw sig-max-delay          uint16
|       +--rw sig-number-resends     uint16
|       +--rw sig-resend-delay       uint16
|       +--rw sig-resend-count       uint16
+--rw terminal
  +--rw (user-scope)
  +--:(all-users)
  | +--rw all-users
  |   +--rw log-selector
  |     +--rw (selector-facility)
  |       | +--:(no-log-facility)
  |       | | +--rw no-facilities?  empty
  |       | +--:(log-facility)
  |       |   +--rw log-facility* [facility]
  |       |     +--rw facility      union

```

```

|          |          +--rw severity          union
|          |          +--rw severity-operator? enumeration
{selector-severity-operator-config}?
|          +--rw pattern-match?  string {selector-match-
processing-config}?
+--:(per-user) {terminal-facility-user-logging-config}?
+--rw user-name* [uname]
+--rw uname          string
+--rw log-selector
+--rw (selector-facility)
| +--:(no-log-facility)
| | +--rw no-facilities?  empty
| +--:(log-facility)
|   +--rw log-facility* [facility]
|     +--rw facility          union
|     +--rw severity          union
|     +--rw severity-operator? enumeration
{selector-severity-operator-log-config}?
+--rw pattern-match?  string {selector-match-
processing-config}?

```

[4.](#) SYSLOG YANG Models

[4.1.](#) SYSLOG-TYPES module

```
<CODE BEGINS> file "ietf-syslog-types.yang"
module ietf-syslog-types {
  namespace "urn:ietf:params:xml:ns:yang:ietf-syslog-types";
  prefix syslogtypes;

  organization "IETF NETMOD (NETCONF Data Modeling Language) Working
    Group";
  contact
    "WG Web: <http://tools.ietf.org/wg/netmod/>
    WG List: <mailto:netmod@ietf.org>

    WG Chair: Tom Nadeau
              <mailto:tnadeau@lucidvision.com>

    WG Chair: Kent Watson
              <mailto:kwatsen@juniper.net>

    Editor: Ladislav Lhotka
           <mailto:lhotka@nic.cz>";
  description
    "This module contains a collection of YANG type definitions for
    SYSLOG.";

  revision 2015-10-14 {
    description
      "Initial Revision";
    reference
      "This model references RFC 5424 - The Syslog Protocol,
      and RFC 5848 - Signed Syslog Messages.";
  }

  typedef severity {
    type enumeration {
      enum "emergency" {
        value 0;
        description
          "Emergency Level Msg";
      }
      enum "alert" {
        value 1;
        description
          "Alert Level Msg";
      }
      enum "critical" {
        value 2;
```

```
description
    "Critical Level Msg";
}
enum "error" {
    value 3;
    description
        "Error Level Msg";
}
```

Wildes, et al. Expires Apr 16, 2016

[Page 6]

```
    enum "warning" {
      value 4;
      description
        "Warning Level Msg";
    }
    enum "notice" {
      value 5;
      description
        "Notification Level Msg";
    }
    enum "info" {
      value 6;
      description
        "Informational Level Msg";
    }
    enum "debug" {
      value 7;
      description
        "Debugging Level Msg";
    }
  }
  description
    "The definitions for Syslog message severity as per RFC 5424.";
}

identity syslog-facility {
  description
    "The base identity to represent syslog facilities";
}

identity kern {
  base syslog-facility;
  description
    "The facility for kernel messages as defined in RFC 5424.";
}

identity user {
  base syslog-facility;
  description
    "The facility for user-level messages as defined in RFC 5424.";
}

identity mail {
  base syslog-facility;
  description
    "The facility for the mail system as defined in RFC 5424.";
}

identity daemon {
```

```
base syslog-facility;
description
  "The facility for the system daemons as defined in RFC 5424";
}

identity auth {
  base syslog-facility;
  description
    "The facility for security/authorization messages as defined
    in RFC 5424";
}
```

```
identity syslog {
  base syslog-facility;
  description
    "The facility for messages generated internally by syslogd
    facility as defined in RFC 5424.";
}

identity lpr {
  base syslog-facility;
  description
    "The facility for the line printer subsystem as defined in
    RFC 5424.";
}

identity news {
  base syslog-facility;
  description
    "The facility for the network news subsystem as defined in
    RFC 5424.";
}

identity uucp {
  base syslog-facility;
  description
    "The facility for the UUCP subsystem as defined in RFC 5424.";
}

identity cron {
  base syslog-facility;
  description
    "The facility for the clock daemon as defined in RFC 5424.";
}

identity authpriv {
  base syslog-facility;
  description
    "The facility for privileged security/authorization messages
    as defined in RFC 5424.";
}

identity ftp {
  base syslog-facility;
  description
    "The facility for the FTP daemon as defined in RFC 5424.";
}

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

```
    "The facility for the NTP subsystem as defined in RFC 5424.";
}

identity audit {
    base syslog-facility;
    description
        "The facility for log audit messages as defined in RFC 5424.";
}
```

```
identity console {
  base syslog-facility;
  description
    "The facility for log alert messages as defined in RFC 5424.";
}

identity cron2 {
  base syslog-facility;
  description
    "The facility for the second clock daemon as defined in
    RFC 5424.";
}

identity local0 {
  base syslog-facility;
  description
    "The facility for local use 0 messages as defined in
    RFC 5424.";
}

identity local1 {
  base syslog-facility;
  description
    "The facility for local use 1 messages as defined in
    RFC 5424.";
}

identity local2 {
  base syslog-facility;
  description
    "The facility for local use 2 messages as defined in
    RFC 5424.";
}

identity local3 {
  base syslog-facility;
  description
    "The facility for local use 3 messages as defined in
    RFC 5424.";
}

identity local4 {
  base syslog-facility;
  description
    "The facility for local use 4 messages as defined in
    RFC 5424.";
}

identity local5 {
```

```
base syslog-facility;
description
  "The facility for local use 5 messages as defined in
  RFC 5424.";
}

identity local6 {
  base syslog-facility;
  description
    "The facility for local use 6 messages as defined in
    RFC 5424.";
}

identity local7 {
  base syslog-facility;
  description
    "The facility for local use 7 messages as defined in
    RFC 5424.";
}
}
```

<CODE ENDS>

4.2. SYSLOG module

```
<CODE BEGINS> file "ietf-syslog.yang"
module ietf-syslog {
  namespace "urn:ietf:params:xml:ns:yang:ietf-syslog";
  prefix syslog;

  import ietf-inet-types {
    prefix inet;
  }

  import ietf-interfaces {
    prefix if;
  }

  import ietf-syslog-types {
    prefix syslogtypes;
  }

  organization "IETF NETMOD (NETCONF Data Modeling Language)
  Working Group";
  contact
    "WG Web: <http://tools.ietf.org/wg/netmod/>
    WG List: <mailto:netmod@ietf.org>

    WG Chair: Tom Nadeau
              <mailto:tnadeau@lucidvision.com>

    WG Chair: Kent Watson
              <mailto:kwatsen@juniper.net>

    Editor: Ladislav Lhotka
           <mailto:lhotka@nic.cz>";
  description
    "This module contains a collection of YANG definitions
    for Syslog configuration.";

  revision 2015-10-14 {
    description
      "Initial Revision";
    reference
      "RFC 5424: The Syslog Protocol
      RFC 5848: Signed Syslog Messages";
  }

  feature buffer-limit-bytes {
    description
      "This feature indicates that local memory logging buffers
      are limited in size using a limit expressed in bytes.";
  }
}
```

```
feature buffer-limit-messages {  
  description  
    "This feature indicates that local memory logging buffers  
    are limited in size using a limit expressed in number  
    of messages."  
}
```

```
feature structured-data-config {  
  description  
    "This feature represents the ability to log messages  
    in structured-data format as per RFC 5424."  
}
```

```
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 terminal-facility-user-logging-config {
  description
    "This feature represents the ability to adjust
    log message settings for individual terminal users.";
}

feature selector-severity-operator-config {
  description
    "This feature represents the ability to select messages
    using the additional operators equal to, or not equal to
    when comparing the Syslog message severity.";
}

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

feature signed-messages-config {
  description
    "This feature represents the ability to configure signed
    syslog messages according to RFC 5848.";
}

grouping syslog-severity {
  description
    "This grouping defines the Syslog severity which is used to
    select log messages.";
  leaf severity {
    type union {
      type syslogtypes:severity;
      type enumeration {
        enum all {
          value -1;
          description
            "This enum describes the case where all severities
```

```
        are requested.";
    }
    enum none {
        value -2;
        description
            "This enum describes the case where no severities
            are requested.";
    }
}
mandatory true;
description
    "This leaf specifies the Syslog message severity. When
    severity is specified, the default severity comparison
    is all messages of the specified severity and greater are
    logged. 'all' is a special case which means all severities
    are requested. 'none' is a special case which means that
    no severity selection should occur.";
}
```

```
leaf severity-operator {
  if-feature selector-severity-operator-config;
  type enumeration {
    enum equals-or-higher {
      description
        "This enum specifies all messages of the specified
        severity and higher are logged according to the
        given log-action";
    }
    enum equals {
      description
        "This enum specifies all messages that are for
        the specified severity are logged according to the
        given log-action";
    }
    enum not-equals {
      description
        "This enum specifies all messages that are not for
        the specified severity are logged according to the
        given log-action";
    }
  }
  default equals-or-higher;
  description
    "This leaf describes the option to specify how the
    severity comparison is performed.";
}

grouping syslog-selector {
  description
    "This grouping defines a Syslog selector which is used to
    select log messages for the log-action (buffer, file,
    etc). Choose one of the following:
    no-log-facility
    log-facility [<facility> <severity>...]";
  container log-selector {
    description
      "This container describes the log selector parameters
      for Syslog.";
    choice selector-facility {
      mandatory true;
      description
        "This choice describes the option to specify no
        facilities, or a specific facility which can be
        all for all facilities.";
      case no-log-facility {
        description
          "This case specifies no facilities will match when
```

```
    comparing the Syslog message facility. This is a
    method that can be used to effectively disable a
    particular log-action (buffer, file, etc).";
leaf no-facilities {
    type empty;
    description
    "This leaf specifies that no facilities are selected
    for this log-action.";
}
}
```

```
case log-facility {
  description
    "This case specifies one or more specified facilities
    will match when comparing the Syslog message facility.";
  list log-facility {
    key facility;
    description
      "This list describes a collection of Syslog
      facilities and severities.";
    leaf facility {
      type union {
        type identityref {
          base syslogtypes: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 syslog-severity;
  }
}

leaf pattern-match {
  if-feature selector-match-processing-config;
  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 RFC 5424
    SYSLOG-MSG field.";
}

container syslog {
  description
    "This container describes the configuration parameters for
    Syslog.";
  container log-actions {
    description
      "This container describes the log-action parameters
      for Syslog.";
  }
}
```

```
container console {
  presence "Enables logging console configuration";
  description
    "This container describes the configuration parameters for
    console logging.";
  uses syslog-selector;
}
```

```
container buffer {
  description
    "This container describes the configuration parameters for
    local memory buffer logging. The buffer is circular in
    nature, so newer messages overwrite older messages after
    the buffer is filled.";
  list log-buffer {
    key name;
    description
      "This list describes a collection of local logging
      memory buffers.";
    leaf name {
      type string;
      description
        "This leaf specifies the name of the log buffer.";
    }
    uses syslog-selector;
    leaf buffer-size-bytes {
      if-feature buffer-limit-bytes;
      type uint64;
      units "bytes";
      description
        "This leaf configures the amount of memory
        (in bytes) that will be dedicated to the local
        memory logging buffer. The default value varies
        by implementation.";
    }
    leaf buffer-size-messages {
      if-feature buffer-limit-messages;
      type uint64;
      units "log messages";
      description
        "This leaf configures the amount number of log
        messages that can be stored in the local memory
        logging buffer. The default value varies by
        implementation.";
    }
  }
}
}
}
container file {
  description
    "This container describes the configuration parameters for
    file logging.";
  list log-file {
    key "name";
    description
      "This list describes a collection of local logging
      files.";
    leaf name {
```

```
type inet:uri;
description
  "This leaf specifies the name of the log file which
  MUST use the uri scheme file:.";
}
```

Wildes, et al.

Expires Apr 16, 2016

[Page 14]

```
uses syslog-selector;
leaf structured-data {
  if-feature structured-data-config;
  type boolean;
  default false;
  description
    "This leaf describes how log messages are written to
    the log file. If true, messages will be written
    with one or more STRUCTURED-DATA elements as per
    RFC5424; if false, messages will be written with
    STRUCTURED-DATA = NILVALUE.";
}
container file-archive {
  description
    "This container describes the configuration
    parameters for log file archiving.";
  leaf number-of-files {
    if-feature file-limit-size;
    type uint32;
    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 uint64;
    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 uint16;
    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 {
  description
    "This container describes the configuration parameters for
    remote logging.";
  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 tcp {
      container tcp {
        description
          "This container describes the TCP transport
          options.";
        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 udp {
    container udp {
      description
        "This container describes the UDP transport
        options.";
      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.";  
}  
}  
}  
}
```

```
uses syslog-selector;
leaf destination-facility {
  type identityref {
    base syslogtypes:syslog-facility;
  }
  default syslogtypes:local7;
  description
    "This leaf specifies the facility used in messages
    delivered to the remote server.";
}
leaf source-interface {
  type if:interface-ref;
  description
    "This leaf sets the source interface for the remote
    Syslog server. Either the interface name or the
    interface IP address can be specified. 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 syslog-sign {
  if-feature signed-messages-config;
  presence
    "If present, syslog-sign is activated.";
  description
    "This container describes the configuration
    parameters for signed syslog messages as described
    by RFC 5848.";
  reference
    "RFC 5848: Signed Syslog Messages";
  leaf cert-initial-repeat {
    type uint16;
    mandatory true;
    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 uint16;
    mandatory true;
    description
      "This leaf specifies the maximum time delay in
      seconds until resending the Certificate Block.";
  }
  leaf cert-resend-count {
    type uint16;
    mandatory true;
    description
```

```
        "This leaf specifies the maximum number of other
        syslog messages to send until resending the
        Certificate Block.";
    }
    leaf sig-max-delay {
        type uint16;
        mandatory true;
        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.";
    }
}
```



```
    </console>
  </log-actions>
</syslog>
</config>
</edit-config>
</rpc>

<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Enable remote logging of syslogs to udp destination 1.1.1.1
for facility auth, severity error

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <candidate/>
    </target>
    <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
      <syslog xmlns="urn:ietf:params:xml:ns:yang:ietf-syslog"
        xmlns:syslog="urn:ietf:params:xml:ns:yang:ietf-syslog">
        <log-actions>
          <remote>
            <destination>
              <name>remote1</name>
              <udp>
                <address>1.1.1.1</address>
              </udp>
              <log-selector>
                <log-facility>
                  <facility xmlns:syslogtypes=
                    "urn:ietf:params:xml:ns:yang:ietf-syslog-types">
                    syslogtypes:auth</facility>
                  <severity>error</severity>
                </log-facility>
              </log-selector>
            </destination>
          </remote>
        </log-actions>
      </syslog>
    </config>
  </edit-config>
</rpc>

<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

5. Implementation Status

[Note to RFC Editor: Please remove this section before publication.]

This section records the status of known implementations of the Syslog YANG model at the time of posting of this Internet-Draft.

Cisco Systems, Inc. has implemented the proposed IETF Syslog model

for the Nexus 7000 NXOS OS as a prototype, together with an augmentation model for operating system specific Syslog configuration features.

Five leaves were implemented in the base IETF model and three leaves were implemented in the Cisco specific augmentation model as follows:

| Leaf XPATH | Sample NXOS CLI Command(s) |
|---------------------------------------|---------------------------------------------------------------------------------------------|
| syslog:log-actions/console | logging console 1 |
| syslog:log-actions/file | logging logfile mylog.log 2 4096 |
| syslog:log-actions/terminal | logging monitor 2 |
| syslog:log-actions/remote | *logging server server.cisco.com 2 facility user *logging source-interface loopback 0 |
| cisco-syslog:logging-timestamp-config | logging timestamp milli-seconds |
| cisco-syslog:origin-id-cfg | logging origin-id string abcdef |
| cisco-syslog:module-logging | logging module 1 |

*The "logging server" and "logging source-interface" commands were combined into one base model leaf.

The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs.

6. Security Considerations

The YANG module defined in this memo is designed to be accessed via the NETCONF protocol [[RFC6241](#)] [[RFC6241](#)]. The lowest NETCONF layer is the secure transport layer and the mandatory-to-implement secure transport is SSH [[RFC6242](#)] [[RFC6242](#)]. The NETCONF access control model [[RFC6536](#)] [[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 the 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.

TBD: List specific Subtrees and data nodes and their sensitivity/vulnerability.

7. IANA Considerations

This document registers a URI in the IETF XML registry [[RFC3688](#)] [[RFC3688](#)]. Following the format in [RFC 3688](#), the following registration is requested to be made:

URI: urn:ietf:params:xml:ns:yang:syslog

Registrant Contact: The IESG.

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

This document registers a YANG module in the YANG Module Names registry [[RFC6020](#)].

name: syslog namespace: urn:ietf:params:xml:ns:yang:syslog
prefix: syslog reference: RFC XXXX

8. Acknowledgements

The authors wish to thank the following who commented on versions 01 through 05 of this proposal:

Martin Bjorklund <mbjorklu@cisco.com>
Jim Gibson <gibson@cisco.com>
Jeffrey Haas <jhaas@pfrc.org>
John Heasley <heas@shrubbery.net>
Giles Heron <giheron@cisco.com>
Lisa Huang <yihuan@cisco.com>
Jeffrey K Lange <jeffrey.K.lange@ge.com>
Jan Lindblad <jlindbla@cisco.com>
Chris Lonvick <lonvick@gmail.com>
Juergen Schoenwaelder <j.schoenwaelder@jacobs-university.de>
Jason Sterne <jason.sterne@alcatel-lucent.com>
Peter Van Horne <petervh@cisco.com>
Bert Wijnen <bertietf@bwinen.net>
Aleksandr Zhdankin <azhdanki@cisco.com>

9. Change log [RFC Editor: Please remove]

10. References

[RFC3164] Lonvick, C., "The BSD syslog Protocol", [BCP 81](#), [RFC 3164](#), August 2001.

[RFC3688] Mealling, M., "The IETF XML Registry", [BCP 81](#), [RFC 3688](#), March 2004.

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

[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.

Authors' Addresses

Clyde Wildes
Cisco Systems Inc.
Email: cwildes@cisco.com

Kiran Agrahara Sreenivasa
Cisco Systems, Inc.
Email: kkoushik@cisco.com