

NETMOD WG
Internet-Draft
Intended status: Standards Track
Expires: 21 September 2024

J. Clarke, Ed.
Cisco
M. Jethanandani, Ed.
Kloud Services
C. Wildes, Ed.
Cisco Systems Inc.
K. Koushik, Ed.
Verizon Wireless
20 March 2024

A YANG Data Model for Syslog Configuration
draft-ietf-netmod-syslog-model-32

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.

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 21 September 2024.

Copyright Notice

Copyright (c) 2024 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
1.1. Requirements Language	3
2. Terminology	3
3. NDMA Compliance	3
4. Editorial Note (To be removed by RFC Editor)	4
5. Design of the Syslog Model	4
5.1. Syslog Module	6
6. Syslog YANG Module	15
6.1. The ietf-syslog Module	15
7. Usage Examples	35
7.1. Syslog Configuration for Severity Critical	35
7.2. Remote Syslog Configuration	35
8. Acknowledgements	36
9. IANA Considerations	36
9.1. The IETF XML Registry	36
9.2. The YANG Module Names Registry	37
10. Security Considerations	37
11. References	38
11.1. Normative References	38
11.2. Informative References	40
Appendix A. Implementer Guidelines	40
A.1. Extending Facilities	41
A.2. Syslog Terminal Output	42
A.3. Syslog File Naming Convention	43
Authors' Addresses	43

1. Introduction

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.

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.

Such definitions of syslog protocol are defined in [RFC5424] , and are used in this RFC.

The YANG model in this document conforms to the Network Management Datastore Architecture defined in [RFC8342].

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.

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.

3. NDMA Compliance

The YANG model in this document conforms to the Network Management Datastore Architecture defined in [RFC8342] .

4. Editorial Note (To be removed by RFC Editor)

This document 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-crypto-types --> the assigned RFC value for draft-ietf-netconf-crypto-types
- * 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

5. Design of the Syslog Model

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

This document 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.

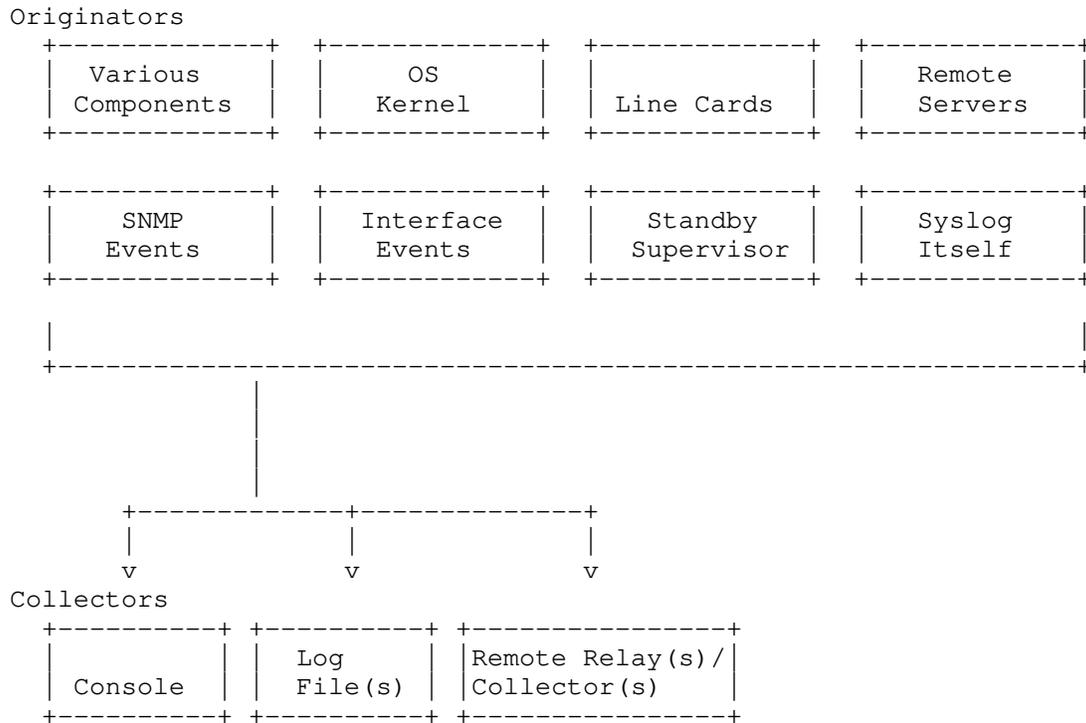


Figure 1. Syslog Processing Flow

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 filters specified by facility-severity pairs, 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, block the message, or stop 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).

5.1. Syslog Module

A simplified graphical representation of the data model is used in this document. Please see [RFC8340] 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?       identityref
          +--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? identityref
+--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-identity!
                |   +--rw (auth-type)
                |   |   +--:(certificate)
                |   |   |   {client-ident-x509-cert}?
                |   |   +--rw certificate
                |   |   |   +--rw (inline-or-keystore)
                |   |   |   |   +--:(inline)
                |   |   |   |   |   {inline-definitions-supp
orted}?
                |   |   |   |   |   +--rw inline-definition
                |   |   |   |   |   |   +--rw public-key-format?
                |   |   |   |   |   |   |   identityref
                |   |   |   |   |   +--rw public-key?
                |   |   |   |   |   |   binary
                |   |   |   |   +--rw private-key-format?
                |   |   |   |   |   identityref
                |   |   |   +--rw (private-key-type)
                |   |   |   |   +--:(cleartext-private-k
ey)
                |   |   |   |   |   {cleartext-priv
ate-keys}?
                |   |   |   |   |   +--rw cleartext-priv
ate-key?

```

					binary
					+---:(hidden-private-key)
-keys}?					{hidden-private
key?					+---rw hidden-private-
					empty
ey)					+---:(encrypted-private-k
ate-keys}?					{encrypted-priv
te-key					+---rw encrypted-priv
					+---rw encrypted-by
lue-format					+---rw encrypted-va
f					identityre
lue					+---rw encrypted-va
					binary
n					+---rw cert-data?
					end-entity-cert-cms
					+---n certificate-expiratio
n					{certificate-expira
tion-notification}?					+--- expiration-date
					yang:date-and-ti
me					+---x generate-csr
					{csr-generation}?
					+---w input
					+---w csr-format
					identityref
					+---w csr-info
					csr-info
					+---ro output
					+---ro (csr-type)
					+---:(p10-csr)
					+---ro p10-csr?
					p10-csr
					+---:(central-keystore)
					{central-keystore-supp
ted, asymmetric-keys}?					+---rw central-keystore-referen
ce					+---rw asymmetric-key?

```

ic-key-ref | | | | ks:central-asymmetr
           | | | | {central-keystore-s
supported, asymmetric-keys}?
           | | | | +---rw certificate?
           | | | | | leafref
           | | | | +---:(raw-public-key)
           | | | | | {client-ident-raw-public-key}?
           | | | | +---rw raw-private-key
           | | | | | +---rw (inline-or-keystore)
           | | | | | +---:(inline)
           | | | | | | {inline-definitions-supp
orted)?
           | | | | | +---rw inline-definition
           | | | | | | +---rw public-key-format?
           | | | | | | | identityref
           | | | | | | +---rw public-key?
           | | | | | | | binary
           | | | | | | +---rw private-key-format?
           | | | | | | | identityref
           | | | | | | +---rw (private-key-type)
           | | | | | | +---:(cleartext-private-k
ey)
           | | | | | | | {cleartext-priv
ate-keys}?
           | | | | | | | +---rw cleartext-priv
ate-key?
           | | | | | | | | binary
           | | | | | | | +---:(hidden-private-key)
           | | | | | | | | {hidden-private
-keys}?
           | | | | | | | | +---rw hidden-private-
key?
           | | | | | | | | | empty
           | | | | | | | | +---:(encrypted-private-k
ey)
           | | | | | | | | | {encrypted-priv
ate-keys}?
           | | | | | | | | | +---rw encrypted-priv
ate-key
           | | | | | | | | | | +---rw encrypted-by
           | | | | | | | | | | +---rw encrypted-va
lue-format
           | | | | | | | | | | | identityre
f
           | | | | | | | | | | | +---rw encrypted-va
lue
           | | | | | | | | | | | binary

```

```

+---:(central-keystore)
    {central-keystore-suppor
ted, asymmetric-keys}?
ce?
key-ref
+---:(tls12-psk)
    {client-ident-tls12-psk}?
+---rw tls12-psk
+---rw (inline-or-keystore)
    +---:(inline)
        {inline-definitions-supp
orted)?
+---rw inline-definition
+---rw key-format?
    | identityref
+---rw (key-type)
+---:(cleartext-symmetric
-key)
+---rw cleartext-symme
tric-key?
    binary
    {cleartext-sy
mmetric-keys}?
y)
+---:(hidden-symmetric-ke
ic-keys)?
    {hidden-symmetr
c-key?
+---rw hidden-symmetri
-key)
    empty
+---:(encrypted-symmetric
etric-keys)?
    {encrypted-symm
tric-key
+---rw encrypted-symme
+---rw encrypted-by
+---rw encrypted-va
lue-format
    identityre
f
+---rw encrypted-va
lue
    binary
+---:(central-keystore)
    {central-keystore-suppor

```

```

ted,symmetric-keys}?
ce?
ey-ref
    +---rw central-keystore-referen
    ks:central-symmetric-k
    +---rw id?
    |
    |   string
    +---:(tls13-epsk)
    |   {client-ident-tls13-epsk}?
    +---rw tls13-epsk
    |   +---rw (inline-or-keystore)
    |   |   +---:(inline)
    |   |   |   {inline-definitions-supp
orted)?
    |   |   |   +---rw inline-definition
    |   |   |   |   +---rw key-format?
    |   |   |   |   |   identityref
    |   |   |   |   +---rw (key-type)
    |   |   |   |   |   +---:(cleartext-symmetric
-key)
    |   |   |   |   |   |   +---rw cleartext-symme
tric-key?
    |   |   |   |   |   |   |   binary
    |   |   |   |   |   |   |   |   {cleartext-sy
mmetric-keys}?
    |   |   |   |   |   |   |   |   +---:(hidden-symmetric-ke
y)
    |   |   |   |   |   |   |   |   |   {hidden-symmetr
ic-keys}?
    |   |   |   |   |   |   |   |   |   +---rw hidden-symmetri
c-key?
    |   |   |   |   |   |   |   |   |   |   empty
    |   |   |   |   |   |   |   |   |   |   +---:(encrypted-symmetric
-key)
    |   |   |   |   |   |   |   |   |   |   |   {encrypted-symm
etric-keys}?
    |   |   |   |   |   |   |   |   |   |   |   +---rw encrypted-symme
tric-key
    |   |   |   |   |   |   |   |   |   |   |   |   +---rw encrypted-by
    |   |   |   |   |   |   |   |   |   |   |   |   |   +---rw encrypted-va
lue-format
    |   |   |   |   |   |   |   |   |   |   |   |   |   |   identityre
f
    |   |   |   |   |   |   |   |   |   |   |   |   |   |   +---rw encrypted-va
lue
    |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   binary
    |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   +---:(central-keystore)
    |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   {central-keystore-suppor

```


n-notification}??					trust-anchor-cert-cms
					+---n certificate-expiration
					{certificate-expiratio
					+--- expiration-date
					yang:date-and-time
certificates}??					+---:(central-truststore)
					{central-truststore-supported,
ef					+---rw central-truststore-reference?
					ts:central-certificate-bag-r
					+---rw raw-public-keys!
					{server-auth-raw-public-key}?
					+---rw (inline-or-truststore)
					+---:(inline)
?					{inline-definitions-supported}
					+---rw inline-definition
					+---rw public-key* [name]
					+---rw name
					string
					+---rw public-key-format
					identityref
					+---rw public-key
					binary
					+---:(central-truststore)
public-keys}??					{central-truststore-supported,
f					+---rw central-truststore-reference?
					ts:central-public-key-bag-re
					+---rw tls12-psks? empty
					{server-auth-tls12-psk}?
					+---rw tls13-epsks? empty
					{server-auth-tls13-epk}?
					+---rw hello-params {tlscmn:hello-params}?
					+---rw tls-versions
					+---rw min? identityref
					+---rw max? identityref
					+---rw cipher-suites
					+---rw cipher-suite*
					tlscsa:tls-cipher-suite-algorithm
					+---rw keepalives {tls-client-keepalives}?
					+---rw peer-allowed-to-send? empty
					+---rw test-peer-aliveness!
					+---rw max-wait? uint16
					+---rw max-attempts? uint8

```

+--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?       identityref
+--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 public-key-format?
|                   | identityref
|               +--rw public-key?          binary
|               +--rw private-key-format?
|                   | identityref
|               +--rw (private-key-type)
|                   +--:(cleartext-private-key)
|                       | {cleartext-private-keys}?
|                       | +--rw cleartext-private-key?  binary
|                   +--:(hidden-private-key)
|                       | {hidden-private-keys}?
|                       | +--rw hidden-private-key?    empty
|                   +--:(encrypted-private-key)
|                       | {encrypted-private-keys}?
|                       | +--rw encrypted-private-key
|                       |     +--rw encrypted-by
|                       |     +--rw encrypted-value-format
|                       |     | identityref
|                       |     +--rw encrypted-value
|                       |     binary
|               +--rw cert-data?
|                   | end-entity-cert-cms
|               +---n certificate-expiration
|                   | {certificate-expiration-notification}
|
|                   +-- expiration-date
|                       | yang:date-and-time
|               +---x generate-csr {csr-generation}?
|                   +---w input
|                       | +---w csr-format      identityref
|                       | +---w csr-info       csr-info

```

?


```
import ietf-crypto-types {
  prefix ct;
  reference
    "I-D.ietf-netconf-crypto-types: YANG Data Types for
    Cryptography";
}

organization
  "IETF NETMOD (Network Modeling) Working Group";
contact
  "WG Web: <https://datatracker.ietf.org/wg/netmod/>
  WG List: <mailto:netmod@ietf.org>

  Editor: Mahesh Jethanandani
         <mailto:mjethanandani@gmail.com>

  Editor: Joe Clarke
         <mailto:jclarke@cisco.com>

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

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

  Copyright (c) 2024 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 Revised BSD License
  set forth in Section 4.c of the IETF Trust's Legal
  Provisions Relating to IETF Documents
  (https://trustee.ietf.org/license-info).

  This version of this YANG module is part of RFC zzzz
  (https://www.rfc-editor.org/info/rfczzzz);
  see the RFC itself for full legal notices.

  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 (RFC 2119)
  (RFC 8174) when, and only when, they appear in all capitals,
```

```
    as shown here.";

revision 2024-03-21 {
    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.
    Note that a lower value is a higher severity. Comparisons
    of equal-or-higher severity mean equal or lower numeric
    value";
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";
}
```

```
identity action {
    description
        "Base identity for action for how a message will be
        handled.";
}

identity log {
    base action;
    description
        "This identity specifies that if the compare operation is
        true the message will be logged.";
}

identity block {
    base action;
    description
        "This identity specifies that if the compare operation is
        true the message will not be logged.";
}

identity stop {
    base action;
    description
        "This identity specifies that if the compare operation is
        true the message will not be logged and no further
        processing will occur for it.";
}

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, use
        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 identityref {
            base "action";
        }
        default "log";
        description

```

```
        "The action can be used to specify how the message
        should be handled. This may include logging the
        message, not logging the message (i.e., blocking
        it), or stopping further processing.";
    }
    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->block means
    messages that have a severity that are equal to the
    specified severity will not 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:.";
      reference
        "RFC 8089: The file URI Scheme";
    }
  }
  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

```



```
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 can be sent
        on any interface.";
}
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 ct:asymmetric-key-pair-with-cert-grou
ping;

        description
            "This is the certificate that
            is periodically sent to the
            remote receiver. The
            certificate is inherently
            associated with its private
            and public keys.";
    }
    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
```


Figure 2: Syslog YANG Model

7. Usage Examples

7.1. Syslog Configuration for Severity Critical

[note: '\ ' line wrapping for formatting only]

```
<!--  
  Enable console logging of syslogs of severity critical  
-->  
  
<?xml version="1.0" encoding="UTF-8"?>  
<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>
```

Figure 3: Syslog Configuration for Severity Critical

7.2. Remote Syslog Configuration

```
[note: '\ ' line wrapping for formatting only]

<!--
  Enable remote logging of syslogs to udp destination
  foo.example.com for facility auth, severity error
-->
<?xml version="1.0" encoding="UTF-8"?>
<syslog xmlns="urn:ietf:params:xml:ns:yang:ietf-syslog">
  <actions>
    <remote>
      <destination>
        <name>remotel</name>
        <udp>
          <address>foo.example.com</address>
        </udp>
        <facility-filter>
          <facility-list>
            <facility>auth</facility>
            <severity>error</severity>
          </facility-list>
        </facility-filter>
      </destination>
    </remote>
  </actions>
</syslog>
```

Figure 4: Remote Syslog Configuration

8. Acknowledgements

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

Andy Bierman, Martin Bjorklund, Alex Campbell, Alex Clemm, Francis Dupont, Jim Gibson, Jeffrey Haas, Bob Harold, John Heasley, Giles Heron, Lisa Huang, Mahesh Jethanandani, Warren Kumari, Jeffrey K Lange, Jan Lindblad, Chris Lonvick, Alexey Melnikov, Kathleen Moriarty, Tom Petch, Adam Roach, Juergen Schoenwaelder, Phil Shafer, Yaron Sheffer, Jason Sterne, Peter Van Horne, Kent Watsen, Bert Wijnen, Dale R Worley, and Aleksandr Zhdankin.

9. IANA Considerations

9.1. 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.

9.2. The YANG Module Names Registry

This document registers one YANG module in the YANG Module Names registry [RFC8525] . Following the format in [RFC7950] , the following registration is requested:

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

10. Security Considerations

The YANG module specified in this document defines a schema for data that is designed to be accessed via network management protocols such as NETCONF [RFC6241] or RESTCONF [RFC8040]. The lowest NETCONF layer is the secure transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC8446].

The NETCONF access control model [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF 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 should be considered sensitive or vulnerable in all network environments. Logging in particular is used to assess the state of systems and can be used to indicate a network compromise. If logging were to be disabled through malicious means, attacks may not be readily detectable. Therefore write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations and on network security.

In addition there are data nodes that require careful analysis and review. 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).

`remote/destination/signing/cert-signer`: When writing this subtree, implementations MUST NOT specify a private key that is used for any other purpose.

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:

`remote/destination/transport`: This subtree contains information about other hosts in the network, and the TLS transport certificate properties if TLS is selected as the transport protocol.

`remote/destination/signing`: This subtree contains information about the syslog message signing properties including signing certificate information.

There are no RPC operations defined in this YANG module.

11. References

11.1. Normative References

[I-D.ietf-netconf-crypto-types]

Watson, K., "YANG Data Types and Groupings for Cryptography", Work in Progress, Internet-Draft, draft-ietf-netconf-crypto-types-33, 1 March 2024, <<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-crypto-types-33>>.

[I-D.ietf-netconf-tls-client-server]

Watson, K., "YANG Groupings for TLS Clients and TLS Servers", Work in Progress, Internet-Draft, draft-ietf-netconf-tls-client-server-40, 1 March 2024, <<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-tls-client-server-40>>.

- [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>>.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <<https://www.rfc-editor.org/info/rfc3688>>.
- [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>>.
- [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>>.
- [RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <<https://www.rfc-editor.org/info/rfc6242>>.
- [RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <<https://www.rfc-editor.org/info/rfc6991>>.
- [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>>.
- [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>>.

- [RFC8089] Kerwin, M., "The "file" URI Scheme", RFC 8089, DOI 10.17487/RFC8089, February 2017, <<https://www.rfc-editor.org/info/rfc8089>>.
- [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>>.
- [RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <<https://www.rfc-editor.org/info/rfc8341>>.
- [RFC8343] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 8343, DOI 10.17487/RFC8343, March 2018, <<https://www.rfc-editor.org/info/rfc8343>>.
- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.
- [RFC8525] Bierman, A., Bjorklund, M., Schoenwaelder, J., Watsen, K., and R. Wilton, "YANG Library", RFC 8525, DOI 10.17487/RFC8525, March 2019, <<https://www.rfc-editor.org/info/rfc8525>>.
- [Std-1003.1-2008] Group, I. A. T. O., "'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/>>.

11.2. Informative References

- [RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <<https://www.rfc-editor.org/info/rfc8340>>.
- [RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", RFC 8342, DOI 10.17487/RFC8342, March 2018, <<https://www.rfc-editor.org/info/rfc8342>>.

Appendix A. Implementer 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):

[note: '\ ' line wrapping for formatting only]

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

  import ietf-syslog {
    prefix syslog;
  }

  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 2024-03-19 {
    description
      "Version 1.0";
    reference
      "Vendor SYSLOG Types: SYSLOG YANG Model";
  }

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

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

A.2. Syslog Terminal Output

Terminal output with requirements more complex than the console subtree currently provides, are expected to be supported via vendor extensions rather than handled via the file subtree.

A.3. Syslog File Naming Convention

The `syslog/file/log-file/file-rotation` container contains configuration parameters for syslog file rotation. This section describes how these fields might be used by an implementer to name syslog files in a rotation process. This information is offered as an informative guide only.

When an active syslog file with a name specified by `log-file/name`, reaches `log-file/max-file-size` and/or syslog events arrive after the period specified by `log-file/rollover`, the logging system can close the file, can compress it, and can name the archive file `<log-file/name>.0.gz`. The logging system can then open a new active syslog file `<log-file/name>`.

When the new syslog file reaches either of the size limits referenced above, `<log-file/name>.0.gz` can be renamed `<log-file/name>.1.gz` and the new syslog file can be closed, compressed and renamed `<log-file/name>.0.gz`. Each time that a new syslog file is closed, each of the prior syslog archive files named `<log-file/name>.<n>.gz` can be renamed to `<log-file/name>.<n + 1>.gz`.

Removal of archive log files could occur when either or both:

- `log-file/number-of-files` specified - the logging system can create up to `log-file/number-of-files` syslog archive files after which, the contents of the oldest archived file could be overwritten.
- `log-file/retention` specified - the logging system can remove those syslog archive files whose file expiration time (file creation time plus the specified `log-file/retention` time) is prior to the current time.

Authors' Addresses

Joe Clarke (editor)
Cisco
United States of America
Email: jclarke@cisco.com

Mahesh Jethanandani (editor)
Kloud Services
United States of America
Email: mjethanandani@gmail.com

Clyde Wildes (editor)
Cisco Systems Inc.
170 West Tasman Drive
San Jose, CA 95134
United States of America
Phone: +1 415 819-6111
Email: clyde@clydewildes.com

Kiran Koushik (editor)
Verizon Wireless
500 W Dove Rd.
Southlake, TX 76092
United States of America
Phone: +1 512 650-0210
Email: kirankoushik.agraharasreenivasa@verizonwireless.com