     Definition of Managed Objects for the Manet Simplified Multicast
                   Framework Relay Set Process
                      draft-ietf-manet-smf-mib-10

Abstract

   This memo defines a portion of the Management Information Base (MIB)
   for use with network management protocols in the Internet community.
   In particular, it describes objects for configuring aspects of the
   Simplified Multicast Forwarding (SMF) process for Mobile Ad-Hoc
   Networks (MANETs). The SMF-MIB module also reports state
   information, performance information, and notifications. In addition
   to configuration, the additional state and performance information is
   useful to operators troubleshooting multicast forwarding problems.

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 September 9, 2014.

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

   This memo defines a portion of the Management Information Base (MIB)
   for use with network management protocols in the Internet community.
   In particular, it describes objects for configuring aspects of a
   process implementing Simplified Multicast Forwarding (SMF) [RFC6621]
   for Mobile Ad-Hoc Networks (MANETs). SMF provides multicast
   Duplicate Packet Detection (DPD) and supports algorithms for
   constructing an estimate of a MANET Minimum Connected Dominating Set
   (MCDS) for efficient multicast forwarding. The SMF-MIB module also
   reports state information, performance information, and
   notifications. In addition to configuration, this additional state
   and performance information is useful to operators troubleshooting
   multicast forwarding problems.

2. The Internet-Standard Management Framework

   For a detailed overview of the documents that describe the current
   Internet-Standard Management Framework, please refer to section 7 of
   RFC 3410 [RFC3410].

   Managed objects are accessed via a virtual information store, termed
   the Management Information Base or MIB. MIB objects are generally
   accessed through the Simple Network Management Protocol (SNMP).
   Objects in the MIB are defined using the mechanisms defined in the
   Structure of Management Information (SMI). This memo specifies a MIB
   module that is compliant to the SMIv2, which is described in STD 58,
   RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580
   [RFC2580].

3. Conventions

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in RFC 2119 [RFC2119].

4. Overview

   SMF provides methods for implementing Duplicate Packet Detection
   (DPD)-based multicast forwarding with the optional use of Connected
   Dominating Set (CDS)-based relay sets. The CDS provides a complete
   connected coverage of the nodes comprising the MANET. The Minimum
   CDS (MCDS) is the smallest set of MANET nodes (comprising a connected
   cluster) which cover all the nodes in the cluster with their
   transmissions. As the density of the MANET nodes increase, the
   fraction of nodes required in an MCDS decreases. Using the MCDS as a
   multicast forwarding set then becomes an efficient multicast
   mechanism for MANETs.

Various algorithms for the construction of estimates of the MCDS
exist. The Simplified Multicast Framework [RFC6621] describes some
of these. It further defines various operational modes for a node
which is participating in the collective creation of the MCDS
estimates. These modes depend upon the set of related MANET routing
and discovery protocols and mechanisms in operation in the specific
MANET node.

A SMF router's MIB module contains SMF process configuration
parameters (e.g. specific CDS algorithm), state information (e.g.,
current membership in the CDS), performance counters (e.g., packet
counters), and notifications.

4.1. SMF Management Model

This section describes the management model for the SMF node process.

Figure 1 (reproduced from Figure 1 of [RFC6621]) shows the
relationship between the SMF Relay Set selection algorithm and the
related algorithms, processes and protocols running in the MANET
nodes. The Relay Set Selection Algorithm (RSSA) can rely upon
topology information gotten from the MANET Neighborhood Discovery
Protocol (NHDP), from the specific MANET routing protocol running on
the node, or from Layer 2 information passed up to the higher layer
protocol processes.

```
     _____               _____
    |               |             |             |
    | Neighborhood  |             | Relay Set   |
    |  Discovery    |------------>| Selection   |
    |               |  neighbor   |             |
    |_____|    info     |_____|
          \                              /
           \                            /
       neighbor\                       / forwarding
         info    \    _____   /    status
              \    |             |   /
               `-->| Forwarding  |<--'
                   |   Process   |
   ---------------->|_____|---------------->
     incoming packet,               forwarded packets
     interface id , and
     previous hop
```
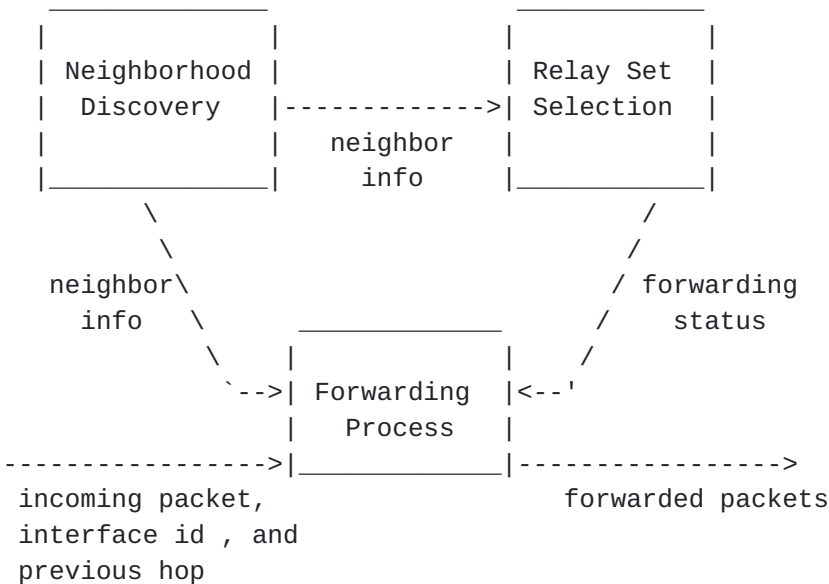
                 Figure 1: SMF Router Architecture

4.2. Terms

   The following definitions apply throughout this document:

   o Configuration Objects - switches, tables, objects which are
      initialized to default settings or set through the management
      interfaces such as defined by this MIB module.

   o Tunable Configuration Objects - objects whose values affect timing
      or attempt bounds on the SMF Relay Set (RS) process.

   o State Objects - automatically generated values which define the
      current operating state of the SMF RS process in the router.

   o Performance Objects - automatically generated values which help an
      administrator or automated tool to assess the performance of the
      CDS multicast process on the router and the overall multicast
      performance within the MANET routing domain.

5. Structure of the MIB Module

   This section presents the structure of the SMF-MIB module. The
   objects are arranged into the following groups:

   o smfMIBNotifications - defines the notifications associated with
      the SMF process.

o smfMIBObjects - defines the objects forming the basis for the SMF-
   MIB module. These objects are divided up by function into the
   following groups:

   * Capabilities Group - This group contains the SMF objects that
      the device uses to advertise its local capabilities with
      respect to, e.g., the supported RSSAs.

   * Configuration Group - This group contains the SMF objects that
      configure specific options that determine the overall operation
      of the SMF process and the resulting multicast performance.

   * State Group - Contains information describing the current state
      of the SMF process such as the Neighbor Table.

   * Performance Group - Contains objects which help to characterize
      the performance of the SMF process, typically counters for
      statistical computations.

o smfMIBConformance - defines two, i.e., minimal and full,
   conformance implementations for the SMF-MIB module.

5.1. Textual Conventions

The textual conventions defined within the SMF-MIB module are:

o The SmfStatus is defined within the SMF-MIB module. This contains
   the current operational status of the SMF process on an interface.

The textual conventions defined for the SMF-MIB module and maintained
by IANA are:

o The IANAsmfOpModeIdTC represents an index that identifies a
   specific SMF operational mode. This textual convention is
   maintained by IANA in the IANAsmfOpModeID-MIB.

o The IANAsmfRssaIdTC represents an index that identifies, through
   reference, a specific RSSA available for operation on the device.
   This textual convention is maintained by IANA in the
   IANAsmfRssaID-MIB.

5.2. The Capabilities Group

The SMF device supports a set of capabilities. The list of
capabilities which the device can advertise are:

   o Operational Mode - topology information from NHDP, CDS-aware
      unicast routing or Cross-layer from Layer 2.

   o SMF RSSA - the specific RSSA operational on the device. Note that
      configuration, state and performance objects related to a specific
      RSSA must be defined within a separate MIB module.

5.3. The Configuration Group

   The SMF device is configured with a set of controls. Some of the
   prominent configuration controls for the SMF device are:

   o Operational Mode - determines where topology information is
      derived from, e.g., NHDP, CDS-aware unicast routing or Cross-layer
      from Layer 2.

   o SMF RSSA - the specific RSSA operational on the device.

   o Duplicate Packet detection for IPv4 - Identification-based or
      Hash-based DPD.

   o Duplicate Packet detection for IPv6 - Identification-based or
      Hash-based DPD.

   o SMF Type Message TLV - if NHDP mode is selected, then the SMF Type
      Message TLV MAY be included in the NHDP exchanges.

   o SMF Address Block TLV - if NHDP mode is selected, then the SMF
      Address Block TLV SHOULD be included in the NHDP exchanges.

   o SMF Address Forwarding Table - a table identifying configured
      multicast addresses to be forwarded by the SMF process.

5.4. The State Group

   The State sub-tree reports current state information, e.g.,

   o Node RSSA State - identifies whether the node is currently in or
      out of the Relay Set.

   o Neighbors Table - a table containing current one-hop neighbors and
      their operational RSSA.

5.5. The Performance Group

   The Performance sub-tree reports primarily counters that relate to
   SMF RSSA performance. The SMF performance counters consists of per
   node and per interface objects:

o Total multicast packets received.

o Total multicast packets forwarded.

o Total duplicate multicast packets detected.

o Per interface statistics table with the following entries:

   * Multicast packets received.

   * Multicast packets forwarded.

   * Duplicate multicast packets detected.

5.6. The Notifications Group

The Notifications Sub-tree contains the list of notifications
supported within the SMF-MIB module and their intended purpose and
utility.

5.7. Tables and Indexing

The SMF-MIB module contains a number of tables which record data
related to:

o configuration and operation of packet forwarding on the local
   router,

o configuration and operation of local MANET interfaces on the
   router, and

o configuration and operation of various RSSA algorithms for packet
   forwarding.

The SMF-MIB module's tables are indexed via the following constructs:

o smfCapabilitiesIndex - the index identifying the combination of
   SMF mode and SMF RSSA available on this device.

o smfCfgAddrForwardingIndex - the index to configured multicast
   addresses lists which are forwarded by the SMF process.

o smfCfgIfIndex - the IfIndex of the interface on the local router
   on which SMF is configured.

o smfStateNeighborIpAddrType, smfStateNeighborIpAddr, and
   smfStateNeighborPrefixLen - the interface index set of specific
   one-hop neighbor nodes to this local router.

These tables and their associated indexing are:

o smfCapabilitiesTable - identifies the resident set of (SMF
   Operational Modes, SMF RSSA algorithms) available on this router.
   This table has 'INDEX { smfCapabilitiesIndex }.

o smfCfgAddrForwardingTable - contains information on multicast
   addresses which are to be forwarded by the SMF process on this
   device. This table has 'INDEX { smfCfgAddrForwardingIndex }'.

o smfCfgInterfaceTable - describes the SMF interfaces on this device
   that are participating in the SMF packet forwarding process. This
   table has 'INDEX { smfCfgIfIndex }'.

o smfStateNeighborTable - describes the current neighbor nodes,
   their addresses and the SMF RSSA and the interface on which they
   can be reached. This table has 'INDEX {
   smfStateNeighborIpAddrType, smfStateNeighborIpAddr,
   smfStateNeighborPrefixLen }'.

o smfPerfIpv4InterfacePerfTable - contains the IPv4 related SMF
   statistics per each SMF interface on this device. This table has
   'INDEX { smfCfgIfIndex }'.

o smfPerfIpv6InterfacePerfTable - contains the IPv6 related SMF
   statistics per each SMF interface on this device. This table has
   'INDEX { smfCfgIfIndex }'.

6. Relationship to Other MIB Modules

6.1. Relationship to the SNMPv2-MIB

The 'system' group in the SNMPv2-MIB module [RFC3418] is defined as
being mandatory for all systems, and the objects apply to the entity
as a whole. The 'system' group provides identification of the
management entity and certain other system-wide data. The SMF-MIB
module does not duplicate those objects.

6.2. MIB modules required for IMPORTS

The textual conventions imported for use in the SMF-MIB module are as
follows. The MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
Counter32, Unsigned32, Integer32 and mib-2 textual conventions are
imported from RFC 2578 [RFC2578]. The TEXTUAL-CONVENTION, RowStatus
and TruthValue textual conventions are imported from RFC 2579
[RFC2579]. The MODULE-COMPLIANCE, OBJECT-GROUP and NOTIFICATION-
GROUP textual conventions are imported from RFC 2580 [RFC2580]. The
InterfaceIndexOrZero textual convention is imported from RFC 2863

[RFC2863]. The SnmpAdminString textual convention is imported from
RFC 3411 [RFC3411]. The InetAddress, InetAddressType and
InetAddressPrefixLength textual conventions are imported from RFC
4001 [RFC4001].

6.3. Relationship to the Future RSSA-MIB Moduless

In a sense, the SMF-MIB module is a general front-end to a set of,
yet to be developed, RSSA-specific MIB modules. These RSSA-specific
MIB modules will define the objects for the configuration, state,
performance and notification required for the operation of these
specific RSSAs. The SMF-MIB module Capabilities Group allows the
remote management station the ability to query the router to discover
the set of supported RSSAs.

7. SMF-MIB Definitions

```
SMF-MIB DEFINITIONS ::= BEGIN

IMPORTS

   MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
   Counter32, Integer32, TimeTicks, experimental
      FROM SNMPv2-SMI                        -- [RFC2578]

   TEXTUAL-CONVENTION, RowStatus, TruthValue
      FROM SNMPv2-TC                         -- [RFC2579]

   MODULE-COMPLIANCE, OBJECT-GROUP,
   NOTIFICATION-GROUP
      FROM SNMPv2-CONF                       -- [RFC2580]

   InterfaceIndexOrZero
      FROM IF-MIB                            -- [RFC2863]

   SnmpAdminString
      FROM SNMP-FRAMEWORK-MIB                -- [RFC3411]

   InetAddress, InetAddressType,
   InetAddressPrefixLength
      FROM INET-ADDRESS-MIB                  -- [RFC4001]

   IANAsmfOpModeIdTC
           FROM IANAsmfOpModeID-MIB

   IANAsmfRssaIdTC
```

```
               FROM IANAsmfRssaID-MIB
       ;


   smfMIB MODULE-IDENTITY
       LAST-UPDATED "201403081300Z"  -- March 8, 2014
       ORGANIZATION "IETF MANET Working Group"
       CONTACT-INFO
          "WG E-Mail: manet@ietf.org

           WG Chairs: sratliff@cisco.com
                      jmacker@nrl.navy.mil


           Editors:   Robert G. Cole
                      US Army CERDEC
                      Space and Terrestrial Communications
                      6010 Frankford Road
                      Aberdeen Proving Ground, MD 21005
                      USA
                      +1 443 395-8744
                      robert.g.cole@us.army.mil

                      Joseph Macker
                      Naval Research Laboratory
                      Washington, D.C. 20375
                      USA
                      macker@itd.nrl.navy.mil

                      Brian Adamson
                      Naval Research Laboratory
                      Washington, D.C. 20375
                      USA
                      adamson@itd.nrl.navy.mil"

       DESCRIPTION
          "This MIB module contains managed object definitions for
           the Manet SMF RSSA process defined in:

           [SMF] Macker, J.(ed.),
           Simplified Multicast Forwarding, RFC 6621,
           May 2012.

           Copyright (C) The IETF Trust (2012). This version
           of this MIB module is part of RFC xxxx; see the RFC
           itself for full legal notices."

         -- Revision History
         REVISION    "201403081300Z"   -- March 8, 2014
```

```
          DESCRIPTION
             "The first version of this MIB module,
              published as RFC xxxx.
              "
          -- RFC-Editor assigns xxxx
          ::= { experimental xxxx }   -- to be assigned by IANA



    --
    -- TEXTUAL CONVENTIONs
    --

    SmfStatus ::= TEXTUAL-CONVENTION
        STATUS       current
        DESCRIPTION
           "An indication of the operability of a SMF
           function or feature.  For example, the status
           of an interface: 'enabled' indicates that
           this interface is performing SMF functions,
           and 'disabled' indicates that it is not.
           Similarly for the status of the device:
           'enabled' indicates that the device has
           enabled the SMF functions on the device and
           'disabled' means that the device and all interfaces
           have disabled all SMF functions."
        SYNTAX  INTEGER {
                        enabled (1),
                        disabled (2)
                }


    --
    -- Top-Level Object Identifier Assignments
    --

    smfMIBNotifications OBJECT IDENTIFIER ::= { smfMIB 0 }
    smfMIBObjects       OBJECT IDENTIFIER ::= { smfMIB 1 }
    smfMIBConformance   OBJECT IDENTIFIER ::= { smfMIB 2 }



    --
    -- smfMIBObjects Assignments:
    --       smfCapabilitiesGroup  - 1
    --       smfConfigurationGroup - 2
    --       smfStateGroup         - 3
    --       smfPerformanceGroup   - 4
    --
```

```
     --
     -- smfCapabilitiesGroup
     --
     --    This group contains the SMF objects that identify specific
     --    capabilities within this device related to SMF functions.
     --

     smfCapabilitiesGroup  OBJECT IDENTIFIER ::= { smfMIBObjects 1 }


     --
     -- SMF Capabilities Table
     --

     smfCapabilitiesTable OBJECT-TYPE
         SYNTAX      SEQUENCE OF SmfCapabilitiesEntry
         MAX-ACCESS  not-accessible
         STATUS      current
         DESCRIPTION
             "The smfCapabilitiesTable identifies the
              resident set of SMF Operational Modes and
              RSSA combinations that can run on this
              forwarder."
         REFERENCE
           "See Section 7.2. 'Reduced Relay Set Forwarding',
            Section 8.1.1. 'SMF Message TLV Type', and
            the Appendices A, B and C in
            RFC 6621 - Simplified Multicast Forwarding
            (SMF), Macker, J., May 2012."
         ::= { smfCapabilitiesGroup 1 }

     smfCapabilitiesEntry OBJECT-TYPE
         SYNTAX      SmfCapabilitiesEntry
         MAX-ACCESS  not-accessible
         STATUS      current
         DESCRIPTION
             "Information about a particular operational
              mode and RSSA combination.
              "
         INDEX   { smfCapabilitiesIndex }
         ::= { smfCapabilitiesTable 1 }

     SmfCapabilitiesEntry ::= SEQUENCE {
         smfCapabilitiesIndex               Integer32,
         smfCapabilitiesOpModeID            IANAsmfOpModeIdTC,
         smfCapabilitiesRssaID              IANAsmfRssaIdTC
     }

     smfCapabilitiesIndex      OBJECT-TYPE
```

```
        SYNTAX        Integer32 (1..2147483647)
        MAX-ACCESS  not-accessible
        STATUS        current
        DESCRIPTION
            "The index for this entry; a unique value,
             greater than zero, for each combination of
             a particular operational mode and RSSA
             algorithm available on this device.
             It is recommended that values are assigned
             contiguously starting from 1.

             Rows in this table are automatically
             populated by the entity's management system
             on initialization.

             By default, the agent should support at least the
             Classical Flooding 'cF' algorithm.  All compliant
             SMF forwarders must support Classical Flooding.
             Hence, the first entry in this table MUST exist
             and MUST be defined as:

                smfCapabilitiesIndex i '1'
                smfCapabilitiesOpModeID i 'cfOnly(1)'
                smfCapabilitiesRssaID i 'cF(1)'

             The value for each combination MUST remain
             constant at least from one re-initialization
             of the entity's management system to the
             next re-initialization."
        ::= { smfCapabilitiesEntry 1 }

    smfCapabilitiesOpModeID     OBJECT-TYPE
        SYNTAX        IANAsmfOpModeIdTC
        MAX-ACCESS  read-only
        STATUS        current
        DESCRIPTION
            "This object identifies
             the particular operational mode for this device."
        ::= { smfCapabilitiesEntry 2 }

    smfCapabilitiesRssaID     OBJECT-TYPE
        SYNTAX        IANAsmfRssaIdTC
        MAX-ACCESS  read-only
        STATUS        current
        DESCRIPTION
            "This object identifies
             the particular RSSA algorithm in this MIB
             module.  Example RSSAs are found in the
```

                 appendix of RFC 6621."
            REFERENCE
               "See, e.g., Section 8.1.1. 'SMF Message TLV Type',
                and the Appendices A, B and C in
                RFC 6621 - Simplified Multicast Forwarding
                (SMF), Macker, J., May 2012."
            ::= { smfCapabilitiesEntry 3 }

      --
      -- smfConfigurationGroup
      --
      --   This group contains the SMF objects that configure specific
      --   options that determine the overall performance and operation
      --   of the multicast forwarding process for the router device
      --   and its interfaces.
      --

      smfConfigurationGroup OBJECT IDENTIFIER ::= { smfMIBObjects 2 }

      smfCfgAdminStatus OBJECT-TYPE
         SYNTAX   SmfStatus
         MAX-ACCESS   read-write
         STATUS   current
         DESCRIPTION
            "The configured status of the SMF process
             on this device. 'enabled(1)' means that
             SMF is configured to run on this device.
             'disabled(2)' mean that the SMF process
             is configured off.

             Prior to SNM functions being performed over
             specific interfaces, this object must first
             be 'enabled'. If this object is 'disabled',
             then no SMF functions are being performed on
             the device and all smfIfAdminStatus objects
             MUST also be set to 'disabled'. When this
             object is changed from 'enabled' to 'disabled'
             by the manager, then all smfIfAdminStatus
             objects MUST also be automatically set to
             'disabled' by the agent.

             The default value for this object SHOULD be
             'enabled'.

             This object is persistent and when written
             the entity SHOULD save the change to
             non-volatile storage."

```
     DEFVAL { enabled }
   ::= { smfConfigurationGroup 1 }

   smfCfgSmfSysUpTime OBJECT-TYPE
      SYNTAX  TimeTicks
      MAX-ACCESS  read-only
      STATUS  current
      DESCRIPTION
         "The time (in hundredths of a second) since the
          system SMF process was last re-initialized.
          The SMF process is re-initialized when the
          value of the 'smfCfgAdminStatus' object
          transitions to 'enabled' from either a prior
          value of 'disabled' or upon initialization
          of this device."
   ::= { smfConfigurationGroup 2 }

   smfCfgRouterIDAddrType  OBJECT-TYPE
      SYNTAX      InetAddressType { ipv4(1), ipv6(2) }
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
         "The address type of the address used for
          SMF ID of this router as specified
          in the 'smfCfgRouterID' next.

          Only the values ipv4(1) and ipv6(2)
          are supported.

          This object is persistent and when written
          the entity SHOULD save the change to
          non-volatile storage."
      DEFVAL { ipv4 }
   ::= { smfConfigurationGroup 3 }

   smfCfgRouterID  OBJECT-TYPE
      SYNTAX      InetAddress (SIZE(4|16))
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
         "The IP address used as the SMF router ID.
          This can be set by the management station.
          If not explicitly set, then the device
          SHOULD select a routable IP address
          assigned to this router for use as
          the 'smfCfgRouterID'.

          The smfCfgRouterID is a logical identification
```

          that MUST be consistent across interoperable
          SMF neighborhoods and it is RECOMMENDED to be
          chosen as the numerically largest address
          contained in a node's 'Neighbor Address List'
          as defined in NHDP. A smfCfgRouterID MUST be
          unique within the scope of the operating
          MANET network regardless of the method used
          for selecting it.

          This object is persistent and when written
          the entity SHOULD save the change to
          non-volatile storage."
       REFERENCE
          "See, e.g.,

          Appendix Section A.1. 'E-CDS Relay Set
              Selection Overview' and

          Appendix Secdtion C.1. 'MPR-CDS Relay
              Set Selection Overview'

          in RFC 6621 - Simplified Multicast Forwarding
          (SMF), Macker, J., May 2012."
      ::= { smfConfigurationGroup 4 }

    smfCfgOperationalMode OBJECT-TYPE
       SYNTAX   Integer32 (1..2147483647)
       MAX-ACCESS   read-write
       STATUS   current
       DESCRIPTION
          "The SMF RSS node operational mode and
           RSSA algorithm combination active on this
           local forwarder. This object is defined
           to be equal to the smfCapabilitiesIndex
           which identifies the specific active
           operational mode and RSSA.

           The default value for this object is
           '1' which corresponds to:

              smfCapabilitiesOpModeID i 'cfOnly(1)'
              smfCapabilitiesRssaID i 'cF(1)'

           This object is persistent and when written
           the entity SHOULD save the change to
           non-volatile storage."
       REFERENCE
           "See Section 7.2. 'Reduced Relay Set Forwarding',

          and the Appendices A, B and C in
          RFC 6621 - Simplified Multicast Forwarding
          (SMF), Macker, J., May 2012."
      DEFVAL { 1 }
   ::= { smfConfigurationGroup 5 }

   smfCfgRssaMember OBJECT-TYPE
      SYNTAX   INTEGER {
                       potential(1),
                       always(2),
                       never(3)
                       }
      MAX-ACCESS read-write
      STATUS   current
      DESCRIPTION
         "The RSSA downselects a set of forwarders for
          multicast forwarding. Sometimes it is useful
          to force an agent to be included or excluded
          from the resulting RSS. This object is a
          switch to allow for this behavior.

          The value 'potential(1)' allows the selected
          RSSA to determine if this agent is included
          or excluded from the RSS.

          The value 'always(2)' forces the selected
          RSSA include this agent in the RSS.

          The value 'never(3)' forces the selected
          RSSA to exclude this agent from the RSS.

          The default setting for this object is
          'potential(1)'. Other settings could pose
          operational risks under certain conditions.

          This object is persistent and when written
          the entity SHOULD save the change to
          non-volatile storage."
      REFERENCE
         "See Section 7. 'Relay Set Selection' in
          RFC 6621 - Simplified Multicast Forwarding
          (SMF), Macker, J., May 2012."
      DEFVAL { potential }
   ::= { smfConfigurationGroup 6 }

   smfCfgIpv4Dpd OBJECT-TYPE
      SYNTAX   INTEGER {
                       hashBased(1),

```
                         identificationBased(2)
                         }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
           "The current method for IPv4 duplicate packet
            detection.

            The value 'hashBased(1)' indicates that the
            routers duplicate packet detection is based
            upon comparing a hash over the packet fields.
            This is the default setting for this object.

            The value 'identificationBased(2)'
            indicates that the duplicate packet
            detection relies upon header information
            in the multicast packets to identify
            previously received packets.

            This object is persistent and when written
            the entity SHOULD save the change to
            non-volatile storage."
        REFERENCE
           "See Section 6.2. 'IPv4 Duplicate Packet
            Detection' in  RFC 6621 - Simplified
            Multicast Forwarding (SMF), Macker, J.,
            May 2012."
        DEFVAL { hashBased }
     ::= { smfConfigurationGroup 7 }

     smfCfgIpv6Dpd  OBJECT-TYPE
        SYNTAX      INTEGER {
                         hashBased(1),
                         identificationBased(2)
                         }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
           "The current method for IPv6 duplicate packet
            detection.

            The values indicate the type of method used
            for duplicate packet detection as described
            the previous description for the object
            `smfCfgIpv4Dpd'.

            The default value for this object is
            'hashBased(1)'.
```

           This object is persistent and when written
           the entity SHOULD save the change to
           non-volatile storage."
      REFERENCE
          "See Section 6.1. 'IPv6 Duplicate Packet
           Detection' in RFC 6621 - Simplified
           Multicast Forwarding (SMF), Macker, J.,
           May 2012."
      DEFVAL { hashBased }
   ::= { smfConfigurationGroup 8 }

   smfCfgMaxPktLifetime OBJECT-TYPE
      SYNTAX   Integer32 (0..65535)
      UNITS   "Seconds"
      MAX-ACCESS   read-write
      STATUS   current
      DESCRIPTION
          "The estimate of the network packet
           traversal time.

           This object is persistent and when written
           the entity SHOULD save the change to
           non-volatile storage."
      REFERENCE
          "See Section 6. 'SMF Duplicate Packet
           Detection' in RFC 6621 - Simplified
           Multicast Forwarding (SMF), Macker, J.,
           May 2012."
      DEFVAL { 60 }
   ::= { smfConfigurationGroup 9 }

   smfCfgDpdEntryMaxLifetime OBJECT-TYPE
      SYNTAX   Integer32 (0..65525)
      UNITS   "Seconds"
      MAX-ACCESS   read-write
      STATUS   current
      DESCRIPTION
          "The maximum lifetime of a cached DPD
           record in the local device storage.

           If the memory is running low prior to the
           MaxLifetime being exceeded, the local SMF
           devices should purge the oldest records first.

           This object is persistent and when written
           the entity SHOULD save the change to
           non-volatile storage."
      REFERENCE

         "See Section 6. 'SMF Duplicate Packet
          Detection' in RFC 6621 - Simplified
          Multicast Forwarding (SMF), Macker, J.,
          May 2012."
      DEFVAL { 600 }
   ::= { smfConfigurationGroup 10 }

   --
   -- Configuration of messages to be included in
   -- NHDP message exchanges in support of SMF
   -- operations.
   --

   smfCfgNhdpRssaMesgTLVIncluded OBJECT-TYPE
      SYNTAX   TruthValue
      MAX-ACCESS read-write
      STATUS   current
      DESCRIPTION
         "Indicates whether the associated NHDP messages
          include the RSSA Message TLV, or not. This
          is an optional SMF operational setting.
          The value 'true(1)' indicates that this TLV is
          included; the value 'false(2)' indicates that it
          is not included.

          It is RECOMMENDED that the RSSA Message TLV
          be included in the NHDP messages.

          This object is persistent and when written
          the entity SHOULD save the change to
          non-volatile storage."
      REFERENCE
         "See Section 8.1.1. 'SMF Message TLV Type' in
          RFC 6621 - Simplified Multicast Forwarding
          (SMF), Macker, J., May 2012."
      DEFVAL { true }
   ::= { smfConfigurationGroup 11 }

   smfCfgNhdpRssaAddrBlockTLVIncluded OBJECT-TYPE
      SYNTAX   TruthValue
      MAX-ACCESS read-write
      STATUS   current
      DESCRIPTION
         "Indicates whether the associated NHDP messages
          include the RSSA Address Block TLV, or not.
          This is an optional SMF operational setting.
          The value 'true(1)' indicates that this TLV is

          included; the value 'false(2)' indicates that it
          is not included.

          The smfCfgNhdpRssaAddrBlockTLVIncluded is optional
          in all cases as it depends on the existence of
          an address block which may not be present.
          If this SMF device is configured with NHDP,
          then this object SHOULD be set to 'true(1)'.

          This object is persistent and when written
          the entity SHOULD save the change to
          non-volatile storage."
      REFERENCE
         "See Section 8.1.2. 'SMF Address Block TLV
          Type' in RFC 6621 - Simplified Multicast
          Forwarding (SMF), Macker, J., May 2012."
      DEFVAL { true }
   ::= { smfConfigurationGroup 12 }

   --
   -- Table identifying configured multicast addresses to be forwarded.
   --

   smfCfgAddrForwardingTable OBJECT-TYPE
      SYNTAX   SEQUENCE OF SmfCfgAddrForwardingEntry
      MAX-ACCESS not-accessible
      STATUS   current
      DESCRIPTION
         "The (conceptual) table containing information on multicast
          multicast addresses which are to be forwarded by the SMF
          process. This table represents an IP filters table for
          forwarding (or not) packets based upon their IP
          multicast address.

          The SMF process can be configured to forward only those
          multicast addresses found within the
          smfCfgAddrForwardingTable. As such, addresses which are
          to be forwarded by the SMF process MUST be found within
          the address ranges configured within this table, unless
          this table is empty.

          Each row is associated with a range of multicast
          addresses, and ranges for different rows must be disjoint.
          Different rows MAY share a common
          smfCfgAddrForwardingGroupName to administratively
          associate different rows.

              The objects in this table are persistent and when written
              the entity SHOULD save the change to non-volatile storage."
        REFERENCE
            "See Section 9.1. 'Forwarded Multicast Groups' in
             RFC 6621 - Simplified Multicast Forwarding
             (SMF), Macker, J., May 2012."
    ::= { smfConfigurationGroup 13 }

    smfCfgAddrForwardingEntry OBJECT-TYPE
        SYNTAX   SmfCfgAddrForwardingEntry
        MAX-ACCESS not-accessible
        STATUS   current
        DESCRIPTION
            "An entry (conceptual row) containing the information on a
             particular multicast scope."
        INDEX { smfCfgAddrForwardingIndex }
        ::= { smfCfgAddrForwardingTable 1 }

    SmfCfgAddrForwardingEntry ::= SEQUENCE {
        smfCfgAddrForwardingIndex   Integer32,
        smfCfgAddrForwardingGroupName   SnmpAdminString,
        smfCfgAddrForwardingAddrType   InetAddressType,
        smfCfgAddrForwardingAddress   InetAddress,
        smfCfgAddrForwardingAddrPrefixLength
                                          InetAddressPrefixLength,
        smfCfgAddrForwardingStatus   RowStatus
    }

    smfCfgAddrForwardingIndex   OBJECT-TYPE
        SYNTAX   Integer32 (1..2147483647)
        MAX-ACCESS not-accessible
        STATUS   current
        DESCRIPTION
            "This object identifies an unique entry
             for a forwarding group. The index for
             this entry is a unique value,
             greater than zero, for each row.
             It is recommended that values are assigned
             contiguously starting from 1.

             The value for each row index MUST remain
             constant from one re-initialization
             of the entity's management system to the
             next re-initialization."
        ::= { smfCfgAddrForwardingEntry 1 }

    smfCfgAddrForwardingGroupName OBJECT-TYPE
        SYNTAX   SnmpAdminString

```
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
           "This object identifies a group name for a set of
            row entries in order to administratively associate
            a set of address ranges.

            If there is no group name or this object is
            otherwise not applicable, then this object contains
            a zero-length string.

            This object is persistent and when written
            the entity SHOULD save the change to
            non-volatile storage."
       ::= { smfCfgAddrForwardingEntry 2 }

    smfCfgAddrForwardingAddrType OBJECT-TYPE
        SYNTAX      InetAddressType { ipv4(1), ipv6(2) }
        MAX-ACCESS read-create
        STATUS      current
        DESCRIPTION
           "The type of the addresses in the multicast
            forwarding ranges identified by this table.

            Only the values ipv4(1) and  ipv6(2) are
            supported.

            This object is persistent and when written
            the entity SHOULD save the change to
            non-volatile storage."
       ::= { smfCfgAddrForwardingEntry 3 }

    smfCfgAddrForwardingAddress OBJECT-TYPE
        SYNTAX      InetAddress (SIZE(4|16))
        MAX-ACCESS read-create
        STATUS      current
        DESCRIPTION
           "The multicast group address which, when
            combined with smfCfgAddrForwardingAddrPrefixLength,
            gives the group prefix for this forwarding range.
            The InetAddressType is given by
            smfCfgAddrForwardingAddrType.

            This address object is only significant up to
            smfCfgAddrForwardingAddrPrefixLength bits.  The
            remaining address bits are set to zero.  This is
            especially important for this index field,
            Any non-zero bits would signify an entirely
```

```
        different entry.

        Legal values correspond to the subset of address
        families for which multicast address allocation
        is supported.

        This object is persistent and when written
        the entity SHOULD save the change to
        non-volatile storage."
::= { smfCfgAddrForwardingEntry 4 }

smfCfgAddrForwardingAddrPrefixLength OBJECT-TYPE
   SYNTAX     InetAddressPrefixLength
   MAX-ACCESS read-create
   STATUS     current
   DESCRIPTION
      "The length in bits of the mask which, when
       combined with smfCfgAddrForwardingAddress,
       gives the group prefix for this forwarding
       range.

       This object is persistent and when written
       the entity SHOULD save the change to
       non-volatile storage."
::= { smfCfgAddrForwardingEntry 5 }

smfCfgAddrForwardingStatus OBJECT-TYPE
   SYNTAX     RowStatus
   MAX-ACCESS read-create
   STATUS     current
   DESCRIPTION
      "The status of this row, by which new entries may be
       created, or old entries deleted from this table."
::= { smfCfgAddrForwardingEntry 6 }



--
-- SMF Interfaces Configuration Table
--

smfCfgInterfaceTable  OBJECT-TYPE
   SYNTAX      SEQUENCE OF SmfCfgInterfaceEntry
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "The SMF Interface Table describes the SMF
       interfaces that are participating in the
```

         SMF packet forwarding process. The ifIndex is
         from the interfaces group defined in the
         Interfaces Group MIB module (RFC 2863). As such,
         this table 'sparse augments' the ifTable
         specifically when SMF is to be configured to
         operate over this interface.

         A conceptual row in this table exists if and only
         if either a manager has explicitly created the row
         or there is an interface on the managed device
         that supports and runs SMF.

         The manager can create a row by setting
         rowStatus to 'createAndGo' or 'createAndWait'.
         Row objects having associated DEFVAL clauses are
         automatically defined by the agent with these
         values during row creation, unless the manager
         explicitly defines these object values during the
         row creation.

         If the corresponding entry with ifIndex value
         is deleted from the Interface Table, then the entry
         in this table is automatically deleted and
         SMF is disabled on this interface,
         and all configuration and state information
         related to this interface is to be removed
         from memory.

         If the value of the smfCfgAdminStatus object is changed
         from 'enabled' to 'disabled', then all rows in this
         table are to be deleted and all
         configuration and state information
         related to this interface is to be removed
         from memory."
    REFERENCE
       "RFC 2863 - The Interfaces Group MIB, McCloghrie,
         K., and F. Kastenholtz, June 2000."
    ::= { smfConfigurationGroup 14 }

    smfCfgInterfaceEntry OBJECT-TYPE
       SYNTAX   SmfCfgInterfaceEntry
       MAX-ACCESS not-accessible
       STATUS   current
       DESCRIPTION
          "The SMF interface entry describes one SMF
            interface as indexed by its ifIndex.

            The objects in this table are persistent and when

```
        written the device SHOULD save the change to
        non-volatile storage.  For further information
        on the storage behavior for these objects, refer
        to the description for the smfCfgIfRowStatus
        object."
   INDEX { smfCfgIfIndex }
::= { smfCfgInterfaceTable 1 }

SmfCfgInterfaceEntry ::=
   SEQUENCE {
      smfCfgIfIndex       InterfaceIndexOrZero,
      smfCfgIfName        SnmpAdminString,
      smfCfgIfAdminStatus  SmfStatus,
      smfCfgIfSmfUpTime    TimeTicks,
      smfCfgIfRowStatus    RowStatus
      }

smfCfgIfIndex  OBJECT-TYPE
   SYNTAX      InterfaceIndexOrZero
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "The ifIndex for this SMF interface. This value
       MUST correspond to an ifIndex referring
       to a valid entry in The Interfaces Table.
       If the manager attempts to create a row
       for which the ifIndex does not exist on the
       local device, then the agent SHOULD issue
       a return value of 'inconsistentValue' and
       the operation SHOULD fail."
   REFERENCE
      "RFC 2863 - The Interfaces Group MIB, McCloghrie,
       K., and F. Kastenholtz, June 2000."
   ::= { smfCfgInterfaceEntry 1 }

smfCfgIfName  OBJECT-TYPE
   SYNTAX      SnmpAdminString
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "The textual name of the interface.  The value of this
       object SHOULD be the name of the interface as assigned by
       the local device.  This can be a text-name, such as 'le0'
       or a simple port number, such as '1',
       depending on the interface-naming syntax of the device.

       If there is no local name or this object is otherwise not
       applicable, then this object contains a zero-length string."
```

```
     ::= { smfCfgInterfaceEntry 2 }

smfCfgIfAdminStatus OBJECT-TYPE
   SYNTAX       SmfStatus
   MAX-ACCESS   read-create
   STATUS       current
   DESCRIPTION
      "The SMF interface's administrative status.
      The value 'enabled' denotes that the interface
      is running the SMF forwarding process.
      The value 'disabled' denotes that the interface is
      currently external to the SMF forwarding process.

      The default value for this object is 'enabled(1)'.

      This object SHOULD be persistent and when
      written the device SHOULD save the change to
      non-volatile storage."
   DEFVAL { enabled }
   ::= { smfCfgInterfaceEntry 3 }

smfCfgIfSmfUpTime OBJECT-TYPE
   SYNTAX  TimeTicks
   MAX-ACCESS   read-only
   STATUS  current
   DESCRIPTION
      "The time (in hundredths of a second) since
      this interface SMF process was last
      re-initialized. The interface SMF process
      is re-initialized when the corresponding
      'smfCfgIfRowStatus' object transits to
      the active(1) state."
   ::= { smfCfgInterfaceEntry 4 }

smfCfgIfRowStatus  OBJECT-TYPE
   SYNTAX       RowStatus
   MAX-ACCESS   read-create
   STATUS       current
   DESCRIPTION
      "This object permits management of the table
       by facilitating actions such as row creation,
       construction, and destruction. The value of
       this object has no effect on whether other
       objects in this conceptual row can be
       modified.

       An entry may not exist in the active(1) state unless all
       objects in the entry have a defined appropriate value.  For
```

          objects with DEFVAL clauses, the management station
          does not need to specify the value of this object in order
          for the row to transit to the active(1) state; the default
          value for this object is used. For objects that do not
          have DEFVAL clauses, then the network manager MUST
          specify the value of this object prior to this row
          transitioning to the active(1) state.

          When this object transitions to active(1), all objects
          in this row SHOULD be written to non-volatile (stable)
          storage. Read-create objects in this row MAY be modified.
          When an object in a row with smfCfgIfRowStatus of active(1)
          is changed, then the updated value MUST be reflected in SMF
          and this new object value MUST be written to non-volatile
          storage.

          If this object is not equal to active(1), all associated
          entries in the smfPerfIpv4InterfacePerfTable and the
          smfPerfIpv6InterfacePerfTable MUST be deleted."
      ::= { smfCfgInterfaceEntry 5 }

   --
   -- smfStateGroup
   --
   --   Contains information describing the current state of the SMF
   --   process such as the current inclusion in the RS or not.
   --

   smfStateGroup OBJECT IDENTIFIER ::= { smfMIBObjects 3 }

   smfStateNodeRsStatusIncluded OBJECT-TYPE
      SYNTAX   TruthValue
      MAX-ACCESS   read-only
      STATUS   current
      DESCRIPTION
         "The current status of the SMF node in the context of
          the MANETs relay set. A value of 'true(1)' indicates
          that the node is currently part of the MANET Relay
          Set. A value of 'false(2)' indicates that the node
          is currently not part of the MANET Relay Set."
      REFERENCE
         "See Section 7. 'Relay Set Selection' in
          RFC 6621 - Simplified Multicast Forwarding
          (SMF), Macker, J., May 2012."
   ::= { smfStateGroup 1 }

   smfStateDpdMemoryOverflow OBJECT-TYPE

```
        SYNTAX        Counter32
        UNITS         "DPD Records"
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
           "The number of DPD records that had to be flushed to
            prevent memory overruns for caching of these records.
            The number of records to be flushed upon a buffer
            overflow is an implementation specific decision.

            There is the potential for a counter discontinuity
            in this object if the system SMF process had been
            disabled and later enabled.  In order to check for
            the occurrence of such a discontinuity when monitoring
            this counter object, it is recommended that the
            smfCfgSmfSysUpTime object also be monitored."
        REFERENCE
           "See Section 6. 'SMF Duplicate Packet
            Detection' in RFC 6621 - Simplified
            Multicast Forwarding (SMF), Macker, J.,
            May 2012."
    ::= { smfStateGroup 2 }


    --
    -- SMF Neighbor Table
    --

    smfStateNeighborTable  OBJECT-TYPE
        SYNTAX        SEQUENCE OF SmfStateNeighborEntry
        MAX-ACCESS    not-accessible
        STATUS        current
        DESCRIPTION
           "The SMF StateNeighborTable describes the
            current one-hop neighbor nodes, their address
            and SMF RSSA and the interface on which
            they can be reached."
        REFERENCE
           "See Section 7. 'SMF Neighborhood Discovery' and
            Section 8.1. 'SMF Relay Algorithm TLV
            Types' in RFC 6621 - Simplified Multicast
            Forwarding (SMF), Macker, J., May 2012."
    ::= { smfStateGroup 3 }

    smfStateNeighborEntry  OBJECT-TYPE
        SYNTAX        SmfStateNeighborEntry
        MAX-ACCESS    not-accessible
        STATUS        current
```

```
      DESCRIPTION
         "The SMF Neighbor Table contains the
          set of one-hop neighbors, the interface
          they are reachable on and the SMF RSSA
          they are currently running."
      INDEX { smfStateNeighborIpAddrType,
              smfStateNeighborIpAddr,
              smfStateNeighborPrefixLen }
   ::= { smfStateNeighborTable 1 }

   SmfStateNeighborEntry ::=
      SEQUENCE {
         smfStateNeighborIpAddrType       InetAddressType,
         smfStateNeighborIpAddr           InetAddress,
         smfStateNeighborPrefixLen        InetAddressPrefixLength,
         smfStateNeighborRSSA             IANAsmfRssaIdTC,
         smfStateNeighborNextHopInterface InterfaceIndexOrZero
         }

   smfStateNeighborIpAddrType  OBJECT-TYPE
      SYNTAX      InetAddressType { ipv4(1), ipv6(2) }
      MAX-ACCESS  not-accessible
      STATUS      current
      DESCRIPTION
         "The one-hop neighbor IP address type.

          Only the values 'ipv4(1)' and
          'ipv6(2)' are supported."
   ::= { smfStateNeighborEntry 1 }

   smfStateNeighborIpAddr  OBJECT-TYPE
      SYNTAX      InetAddress (SIZE(4|16))
      MAX-ACCESS  not-accessible
      STATUS      current
      DESCRIPTION
         "The one-hop neighbor Inet IPv4 or IPv6
         address.

         Only IPv4 and IPv6 addresses
         are supported."
   ::= { smfStateNeighborEntry 2 }

   smfStateNeighborPrefixLen  OBJECT-TYPE
      SYNTAX      InetAddressPrefixLength
      UNITS       "bits"
      MAX-ACCESS  not-accessible
      STATUS      current
      DESCRIPTION
```

```
            "The prefix length. This is a decimal value that
             indicates the number of contiguous, higher-order
             bits of the address that make up the network
             portion of the address."
    ::= { smfStateNeighborEntry 3 }

    smfStateNeighborRSSA  OBJECT-TYPE
       SYNTAX        IANAsmfRssaIdTC
       MAX-ACCESS    read-only
       STATUS        current
       DESCRIPTION
          "The current RSSA running on the neighbor."
    ::= { smfStateNeighborEntry 4 }

    smfStateNeighborNextHopInterface OBJECT-TYPE
       SYNTAX        InterfaceIndexOrZero
       MAX-ACCESS    read-only
       STATUS        current
       DESCRIPTION
          "The interface ifIndex over which the
           neighbor is reachable in one-hop."
    ::= { smfStateNeighborEntry 6 }


    --
    -- SMF Performance Group
    --
    --     Contains objects which help to characterize the
    --     performance of the SMF RSSA process, such as statistics
    --     counters. There are two types of SMF RSSA statistics:
    --     global counters and per interface counters.
    --

    smfPerformanceGroup  OBJECT IDENTIFIER ::= { smfMIBObjects 4 }

    smfPerfGobalGroup  OBJECT IDENTIFIER ::= { smfPerformanceGroup 1 }

    --
    -- IPv4 packet counters
    --

    smfPerfIpv4MultiPktsRecvTotal  OBJECT-TYPE
       SYNTAX        Counter32
       UNITS         "Packets"
       MAX-ACCESS    read-only
       STATUS        current
       DESCRIPTION
          "A counter of the total number of
```

```
        multicast IPv4 packets received by the
        device and delivered to the SMF process.

        There is the potential for a counter discontinuity
        in this object if the system SMF process had been
        disabled and later enabled.  In order to check for
        the occurrence of such a discontinuity when monitoring
        this counter object, it is recommended that the
        smfCfgSmfSysUpTime object also be monitored."
 ::= { smfPerfGobalGroup 1 }

 smfPerfIpv4MultiPktsForwardedTotal  OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "Packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
       "A counter of the total number of
        multicast IPv4 packets forwarded by the
        device.

        There is the potential for a counter discontinuity
        in this object if the system SMF process had been
        disabled and later enabled.  In order to check for
        the occurrence of such a discontinuity when monitoring
        this counter object, it is recommended that the
        smfCfgSmfSysUpTime object also be monitored."
 ::= { smfPerfGobalGroup 2 }

 smfPerfIpv4DuplMultiPktsDetectedTotal  OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "Packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
       "A counter of the total number of duplicate
        multicast IPv4 packets detected by the
        device.

        There is the potential for a counter discontinuity
        in this object if the system SMF process had been
        disabled and later enabled.  In order to check for
        the occurrence of such a discontinuity when monitoring
        this counter object, it is recommended that the
        smfCfgSmfSysUpTime object also be monitored."
    REFERENCE
       "See Section 6.2. 'IPv4 Duplicate Packet
        Detection' in RFC 6621 - Simplified Multicast
```

              Forwarding (SMF), Macker, J., May 2012."
       ::= { smfPerfGobalGroup 3 }

       smfPerfIpv4DroppedMultiPktsTTLExceededTotal OBJECT-TYPE
          SYNTAX   Counter32
          UNITS   "Packets"
          MAX-ACCESS read-only
          STATUS   current
          DESCRIPTION
             "A counter of the total number of dropped
              multicast IPv4 packets by the
              device due to TTL exceeded.

              There is the potential for a counter discontinuity
              in this object if the system SMF process had been
              disabled and later enabled. In order to check for
              the occurrence of such a discontinuity when monitoring
              this counter object, it is recommended that the
              smfCfgSmfSysUpTime object also be monitored."
          REFERENCE
             "See Section 5. 'SMF Packet Processing and
              Forwarding' in RFC 6621 - Simplified
              Multicast Forwarding (SMF), Macker, J.,
              May 2012."
       ::= { smfPerfGobalGroup 4 }

       smfPerfIpv4TTLLargerThanPreviousTotal OBJECT-TYPE
          SYNTAX   Counter32
          UNITS   "Packets"
          MAX-ACCESS read-only
          STATUS   current
          DESCRIPTION
             "A counter of the total number of IPv4 packets
              recieved which have a TTL larger than that
              of a previously received identical packet.

              There is the potential for a counter discontinuity
              in this object if the system SMF process had been
              disabled and later enabled. In order to check for
              the occurrence of such a discontinuity when monitoring
              this counter object, it is recommended that the
              smfCfgSmfSysUpTime object also be monitored."
          REFERENCE
             "See Section 5. 'SMF Packet Processing and
              Forwarding' in RFC 6621 - Simplified Multicast
              Forwarding (SMF), Macker, J., May 2012."
       ::= { smfPerfGobalGroup 5 }

```
   --
   -- IPv6 packet counters
   --

   smfPerfIpv6MultiPktsRecvTotal  OBJECT-TYPE
      SYNTAX       Counter32
      UNITS        "Packets"
      MAX-ACCESS   read-only
      STATUS       current
      DESCRIPTION
         "A counter of the total number of
          multicast IPv6 packets received by the
          device and delivered to the SMF process.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled.  In order to check for
          the occurrence of such a discontinuity when monitoring
          this counter object, it is recommended that the
          smfCfgSmfSysUpTime object also be monitored."
   ::= { smfPerfGobalGroup 6 }

   smfPerfIpv6MultiPktsForwardedTotal  OBJECT-TYPE
      SYNTAX       Counter32
      UNITS        "Packets"
      MAX-ACCESS   read-only
      STATUS       current
      DESCRIPTION
         "A counter of the total number of
          multicast IPv6 packets forwarded by the
          device.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled.  In order to check for
          the occurrence of such a discontinuity when monitoring
          this counter object, it is recommended that the
          smfCfgSmfSysUpTime object also be monitored."
   ::= { smfPerfGobalGroup 7 }

   smfPerfIpv6DuplMultiPktsDetectedTotal  OBJECT-TYPE
      SYNTAX       Counter32
      UNITS        "Packets"
      MAX-ACCESS   read-only
      STATUS       current
      DESCRIPTION
         "A counter of the total number of duplicate
          multicast IPv6 packets detected by the
```

```
      device.

      There is the potential for a counter discontinuity
      in this object if the system SMF process had been
      disabled and later enabled.  In order to check for
      the occurrence of such a discontinuity when monitoring
      this counter object, it is recommended that the
      smfCfgSmfSysUpTime object also be monitored."
   REFERENCE
      "See Section 6.1. 'IPv6 Duplicate Packet
      Detection' in RFC 6621 - Simplified Multicast
      Forwarding (SMF), Macker, J., May 2012."
 ::= { smfPerfGobalGroup 8 }

smfPerfIpv6DroppedMultiPktsTTLExceededTotal  OBJECT-TYPE
   SYNTAX      Counter32
   UNITS       "Packets"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "A counter of the total number of dropped
      multicast IPv6 packets by the
      device due to TTL exceeded.

      There is the potential for a counter discontinuity
      in this object if the system SMF process had been
      disabled and later enabled.  In order to check for
      the occurrence of such a discontinuity when monitoring
      this counter object, it is recommended that the
      smfCfgSmfSysUpTime object also be monitored."
   REFERENCE
      "See Section 5. 'SMF Packet Processing and
      Forwarding' in RFC 6621 - Simplified
      Multicast Forwarding (SMF), Macker, J.,
      May 2012."
 ::= { smfPerfGobalGroup 9 }

smfPerfIpv6TTLLargerThanPreviousTotal  OBJECT-TYPE
   SYNTAX      Counter32
   UNITS       "Packets"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "A counter of the total number of IPv6 packets
      recieved which have a TTL larger than that
      of a previously recived identical packet.

      There is the potential for a counter discontinuity
```

```
            in this object if the system SMF process had been
            disabled and later enabled.  In order to check for
            the occurrence of such a discontinuity when monitoring
            this counter object, it is recommended that the
            smfCfgSmfSysUpTime object also be monitored."
        REFERENCE
            "See Section 5. 'SMF Packet Processing and
            Forwarding' in RFC 6621 - Simplified Multicast
            Forwarding (SMF), Macker, J., May 2012."
    ::= { smfPerfGobalGroup 10 }

    smfPerfIpv6HAVAssistsReqdTotal  OBJECT-TYPE
        SYNTAX      Counter32
        UNITS       "Packets"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "A counter of the total number of IPv6 packets
            received which required the HAV assist for DPD.

            There is the potential for a counter discontinuity
            in this object if the system SMF process had been
            disabled and later enabled.  In order to check for
            the occurrence of such a discontinuity when monitoring
            this counter object, it is recommended that the
            smfCfgSmfSysUpTime object also be monitored."
        REFERENCE
            "See Section 6.1.1. 'IPv6 SMF_DPD Option Header'
            in RFC 6621 - Simplified Multicast Forwarding
            (SMF), Macker, J., May 2012."
    ::= { smfPerfGobalGroup 11 }

    smfPerfIpv6DpdHeaderInsertionsTotal  OBJECT-TYPE
        SYNTAX      Counter32
        UNITS       "Packets"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "A counter of the total number of IPv6 packets
            recieved which the device inserted the
            DPD header option.

            There is the potential for a counter discontinuity
            in this object if the system SMF process had been
            disabled and later enabled.  In order to check for
            the occurrence of such a discontinuity when monitoring
            this counter object, it is recommended that the
            smfCfgSmfSysUpTime object also be monitored."
```

       REFERENCE
           "See Section 6.1.2. 'IPv6 Identification-Based
            DPD' in RFC 6621 - Simplified Multicast
            Forwarding (SMF), Macker, J., May 2012."
    ::= { smfPerfGobalGroup 12 }

    --
    -- Per SMF Interface Performance Table
    --

    smfPerfInterfaceGroup OBJECT IDENTIFIER ::= { smfPerformanceGroup 2 }

    smfPerfIpv4InterfacePerfTable OBJECT-TYPE
       SYNTAX   SEQUENCE OF SmfPerfIpv4InterfacePerfEntry
       MAX-ACCESS   not-accessible
       STATUS   current
       DESCRIPTION
          "The SMF Interface Performance Table
           describes the SMF counters per
           interface."
    ::= { smfPerfInterfaceGroup 1 }

    smfPerfIpv4InterfacePerfEntry OBJECT-TYPE
       SYNTAX   SmfPerfIpv4InterfacePerfEntry
       MAX-ACCESS   not-accessible
       STATUS   current
       DESCRIPTION
          "The SMF Interface Performance entry
           describes the statistics for a particular
           node interface."
       INDEX { smfCfgIfIndex }
    ::= { smfPerfIpv4InterfacePerfTable 1 }

    SmfPerfIpv4InterfacePerfEntry ::=
       SEQUENCE {
          smfPerfIpv4MultiPktsRecvPerIf   Counter32,
          smfPerfIpv4MultiPktsForwardedPerIf   Counter32,
          smfPerfIpv4DuplMultiPktsDetectedPerIf   Counter32,
          smfPerfIpv4DroppedMultiPktsTTLExceededPerIf Counter32,
          smfPerfIpv4TTLLargerThanPreviousPerIf   Counter32
          }

    smfPerfIpv4MultiPktsRecvPerIf OBJECT-TYPE
       SYNTAX   Counter32
       UNITS   "Packets"
       MAX-ACCESS   read-only
       STATUS   current

```
      DESCRIPTION
         "A counter of the number of multicast IP
          packets received by the SMF process on
          this device on this interface.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled on this interface.
          In order to check for the occurrence of such a
          discontinuity when monitoring this counter object,
          it is recommended that the smfCfgIfSmfUpTime
          object also be monitored."
      ::= { smfPerfIpv4InterfacePerfEntry 1 }

   smfPerfIpv4MultiPktsForwardedPerIf  OBJECT-TYPE
      SYNTAX      Counter32
      UNITS       "Packets"
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
         "A counter of the number of
          multicast IP packets forwarded by the
          SMF process on this device
          on this interface.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled on this interface.
          In order to check for the occurrence of such a
          discontinuity when monitoring this counter object,
          it is recommended that the smfCfgIfSmfUpTime
          object also be monitored."
      ::= { smfPerfIpv4InterfacePerfEntry 2 }

   smfPerfIpv4DuplMultiPktsDetectedPerIf  OBJECT-TYPE
      SYNTAX      Counter32
      UNITS       "Packets"
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
         "A counter of the number of duplicate
          multicast IP packets detected by the
          SMF process on this device
          on this interface.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled on this interface.
```

```
            In order to check for the occurrence of such a
            discontinuity when monitoring this counter object,
            it is recommended that the smfCfgIfSmfUpTime
            object also be monitored."
      ::= { smfPerfIpv4InterfacePerfEntry 3 }

   smfPerfIpv4DroppedMultiPktsTTLExceededPerIf  OBJECT-TYPE
      SYNTAX      Counter32
      UNITS       "Packets"
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
         "A counter of the total number of dropped
          multicast IPv4 packets by the SMF process
          on this device on this interface
          due to TTL exceeded.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled on this interface.
          In order to check for the occurrence of such a
          discontinuity when monitoring this counter object,
          it is recommended that the smfCfgIfSmfUpTime
          object also be monitored."
      ::= { smfPerfIpv4InterfacePerfEntry 4 }

   smfPerfIpv4TTLLargerThanPreviousPerIf  OBJECT-TYPE
      SYNTAX      Counter32
      UNITS       "Packets"
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
         "A counter of the total number of IPv4 packets
          received by the SMF process on this device
          on this interface which have a TTL larger than
          that of a previously received identical packet.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled on this interface.
          In order to check for the occurrence of such a
          discontinuity when monitoring this counter object,
          it is recommended that the smfCfgIfSmfUpTime
          object also be monitored."
      ::= { smfPerfIpv4InterfacePerfEntry 5 }


   smfPerfIpv6InterfacePerfTable OBJECT-TYPE
```

```
        SYNTAX       SEQUENCE OF SmfPerfIpv6InterfacePerfEntry
     MAX-ACCESS   not-accessible
     STATUS       current
     DESCRIPTION
        "The SMF Interface Performance Table
         describes the SMF counters per
         interface."
  ::= { smfPerfInterfaceGroup 2 }

  smfPerfIpv6InterfacePerfEntry OBJECT-TYPE
     SYNTAX       SmfPerfIpv6InterfacePerfEntry
     MAX-ACCESS   not-accessible
     STATUS       current
     DESCRIPTION
        "The SMF Interface Performance entry
         describes the counters for a particular
         node interface."
     INDEX { smfCfgIfIndex }
  ::= { smfPerfIpv6InterfacePerfTable 1 }

  SmfPerfIpv6InterfacePerfEntry ::=
     SEQUENCE {
        smfPerfIpv6MultiPktsRecvPerIf            Counter32,
        smfPerfIpv6MultiPktsForwardedPerIf       Counter32,
        smfPerfIpv6DuplMultiPktsDetectedPerIf    Counter32,
        smfPerfIpv6DroppedMultiPktsTTLExceededPerIf Counter32,
        smfPerfIpv6TTLLargerThanPreviousPerIf    Counter32,
        smfPerfIpv6HAVAssistsReqdPerIf           Counter32,
        smfPerfIpv6DpdHeaderInsertionsPerIf      Counter32
        }

  smfPerfIpv6MultiPktsRecvPerIf  OBJECT-TYPE
     SYNTAX     Counter32
     UNITS      "Packets"
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
        "A counter of the number of
         multicast IP packets received by the
         SMF process on this device
         on this interface.

         There is the potential for a counter discontinuity
         in this object if the system SMF process had been
         disabled and later enabled on this interface.
         In order to check for the occurrence of such a
         discontinuity when monitoring this counter object,
         it is recommended that the smfCfgIfSmfUpTime
```

```
      object also be monitored."
   ::= { smfPerfIpv6InterfacePerfEntry 1 }

   smfPerfIpv6MultiPktsForwardedPerIf  OBJECT-TYPE
      SYNTAX       Counter32
      UNITS        "Packets"
      MAX-ACCESS   read-only
      STATUS       current
      DESCRIPTION
         "A counter of the number of
          multicast IP packets forwarded by the
          SMF process on this device
          on this interface.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled on this interface.
          In order to check for the occurrence of such a
          discontinuity when monitoring this counter object,
          it is recommended that the smfCfgIfSmfUpTime
          object also be monitored."
   ::= { smfPerfIpv6InterfacePerfEntry 2 }

   smfPerfIpv6DuplMultiPktsDetectedPerIf  OBJECT-TYPE
      SYNTAX       Counter32
      UNITS        "Packets"
      MAX-ACCESS   read-only
      STATUS       current
      DESCRIPTION
         "A counter of the number of duplicate
          multicast IP packets detected by the
          SMF process on this device
          on this interface.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled on this interface.
          In order to check for the occurrence of such a
          discontinuity when monitoring this counter object,
          it is recommended that the smfCfgIfSmfUpTime
          object also be monitored."
   ::= { smfPerfIpv6InterfacePerfEntry 3 }

   smfPerfIpv6DroppedMultiPktsTTLExceededPerIf  OBJECT-TYPE
      SYNTAX       Counter32
      UNITS        "Packets"
      MAX-ACCESS   read-only
      STATUS       current
```

```
      DESCRIPTION
         "A counter of the number of dropped
          multicast IP packets by the
          SMF process on this device
          on this interface due to TTL
          exceeded.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled on this interface.
          In order to check for the occurrence of such a
          discontinuity when monitoring this counter object,
          it is recommended that the smfCfgIfSmfUpTime
          object also be monitored."
   ::= { smfPerfIpv6InterfacePerfEntry 4 }

   smfPerfIpv6TTLLargerThanPreviousPerIf  OBJECT-TYPE
      SYNTAX       Counter32
      UNITS        "Packets"
      MAX-ACCESS   read-only
      STATUS       current
      DESCRIPTION
         "A counter of the total number of IPv6 packets
          received which have a TTL larger than that
          of a previously received identical packet
          by the SMF process on this device on this
          interface.

          There is the potential for a counter discontinuity
          in this object if the system SMF process had been
          disabled and later enabled on this interface.
          In order to check for the occurrence of such a
          discontinuity when monitoring this counter object,
          it is recommended that the smfCfgIfSmfUpTime
          object also be monitored."
   ::= { smfPerfIpv6InterfacePerfEntry 5 }

   smfPerfIpv6HAVAssistsReqdPerIf  OBJECT-TYPE
      SYNTAX       Counter32
      UNITS        "Packets"
      MAX-ACCESS   read-only
      STATUS       current
      DESCRIPTION
         "A counter of the total number of IPv6 packets
          received by the SMF process on this device
          on this interface which required the
          HAV assist for DPD.
```

```
            There is the potential for a counter discontinuity
            in this object if the system SMF process had been
            disabled and later enabled on this interface.
            In order to check for the occurrence of such a
            discontinuity when monitoring this counter object,
            it is recommended that the smfCfgIfSmfUpTime
            object also be monitored."
     ::= { smfPerfIpv6InterfacePerfEntry 6 }

     smfPerfIpv6DpdHeaderInsertionsPerIf  OBJECT-TYPE
        SYNTAX        Counter32
        UNITS         "Packets"
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
           "A counter of the total number of IPv6 packets
            received by the SMF process on this device
            on this interface which the device inserted the
            DPD header option.

            There is the potential for a counter discontinuity
            in this object if the system SMF process had been
            disabled and later enabled on this interface.
            In order to check for the occurrence of such a
            discontinuity when monitoring this counter object,
            it is recommended that the smfCfgIfSmfUpTime
            object also be monitored."
     ::= { smfPerfIpv6InterfacePerfEntry 7 }



     --
     -- Notifications
     --

smfMIBNotifObjects OBJECT IDENTIFIER ::= { smfMIBNotifications 0 }
smfMIBNotifControl OBJECT IDENTIFIER ::= { smfMIBNotifications 1 }


     -- smfMIBNotifObjects

     smfNotifAdminStatusChange NOTIFICATION-TYPE
          OBJECTS { smfCfgRouterIDAddrType, -- The originator of
                                            --   the notification.
                    smfCfgRouterID,         -- The originator of
                                            --   the notification.
                    smfCfgAdminStatus       -- The new status of the
                                            --   SMF process.
```

```
               }
          STATUS       current
          DESCRIPTION
             "smfCfgAdminStatusChange is a notification sent when a
              the 'smfCfgAdminStatus' object changes."
          ::= { smfMIBNotifObjects 1 }

   smfNotifConfiguredOpModeChange NOTIFICATION-TYPE
          OBJECTS { smfCfgRouterIDAddrType, -- The originator of
                                            --   the notification.
                    smfCfgRouterID,         -- The originator of
                                            --   the notification.
                    smfCfgOperationalMode   -- The new Operations
                                            --   Mode of the SMF
                                            --   process.
                  }
          STATUS       current
          DESCRIPTION
             "smfNotifConfiguredOpModeChange is a notification
              sent when the 'smfCfgOperationalMode' object
              changes."
          ::= { smfMIBNotifObjects 2 }

   smfNotifIfAdminStatusChange NOTIFICATION-TYPE
          OBJECTS { smfCfgRouterIDAddrType, -- The originator of
                                            --   the notification.
                    smfCfgRouterID,         -- The originator of
                                            --   the notification.
                    smfCfgIfName,           -- The interface whose
                                            --   status has changed.
                    smfCfgIfAdminStatus     -- The new status of the
                                            --   SMF interface.
                  }
          STATUS       current
          DESCRIPTION
             "smfCfgIfAdminStatusChange is a notification sent when a
              the 'smfCfgIfAdminStatus' object changes."
          ::= { smfMIBNotifObjects 3 }

   smfNotifDpdMemoryOverflowEvent NOTIFICATION-TYPE
          OBJECTS { smfCfgRouterIDAddrType,   -- The originator of
                                              --   the notification.
                    smfCfgRouterID,           -- The originator of
                                              --   the notification.
                    smfStateDpdMemoryOverflow -- The counter of
                                              --   the overflows.
                  }
          STATUS       current
```

```
        DESCRIPTION
          "smfNotifDpdMemoryOverflowEvents is sent when the
           number of memory overflow events exceeds the
           the 'smfNotifDpdMemoryOverflowThreshold' within the
           previous number of seconds defined by the
           'smfNotifDpdMemoryOverflowWindow'."
        ::= { smfMIBNotifObjects 4 }



    -- smfMIBNotifControl
    smfNotifDpdMemoryOverflowThreshold OBJECT-TYPE
        SYNTAX       Integer32 (0..255)
        UNITS        "Events"
        MAX-ACCESS   read-write
        STATUS       current
        DESCRIPTION
          "A threshold value for the
           `smfNotifDpdmemoryOverflowEvents' object.
           If the number of occurences exceeds
           this threshold within the previous
           number of seconds
           'smfNotifDpdMemoryOverflowWindow',
           then the `smfNotifDpdMemoryOverflowEvent'
           notification is sent.

           The default value for this object is
           '1'."
        DEFVAL { 1 }
         ::= { smfMIBNotifControl 1 }

    smfNotifDpdMemoryOverflowWindow OBJECT-TYPE
        SYNTAX       TimeTicks
        MAX-ACCESS   read-write
        STATUS       current
        DESCRIPTION
          "A time window value for the
           `smfNotifDpdmemoryOverflowEvents' object.
           If the number of occurences exceeds
           the `smfNotifDpdMemoryOverflowThreshold'
           within the previous number of seconds
           'smfNotifDpdMemoryOverflowWindow',
           then the `smfNotifDpdMemoryOverflowEvent'
           notification is sent.

           The default value for this object is
           '1'."
        DEFVAL { 1 }
```

```
           ::= { smfMIBNotifControl 2 }



   --
   -- Compliance Statements
   --

   smfCompliances   OBJECT IDENTIFIER ::= { smfMIBConformance 1 }
   smfMIBGroups     OBJECT IDENTIFIER ::= { smfMIBConformance 2 }

   smfBasicCompliance  MODULE-COMPLIANCE
      STATUS current
      DESCRIPTION "The basic implementation requirements for
                   managed network entities that implement
                   the SMF RSSA process."
      MODULE  -- this module
      MANDATORY-GROUPS { smfCapabObjectsGroup,
                         smfConfigObjectsGroup }
   ::= { smfCompliances 1 }

   smfFullCompliance MODULE-COMPLIANCE
      STATUS current
      DESCRIPTION "The full implementation requirements for
                   managed network entities that implement
                   the SMF RSSA process."
      MODULE  -- this module
      MANDATORY-GROUPS { smfCapabObjectsGroup,
                         smfConfigObjectsGroup,
                         smfStateObjectsGroup,
                         smfPerfObjectsGroup,
                         smfNotifObjectsGroup,
                         smfNotificationsGroup
                       }
   ::= { smfCompliances 2 }

   --
   -- Units of Conformance
   --

   smfCapabObjectsGroup OBJECT-GROUP
      OBJECTS {
              smfCapabilitiesOpModeID,
              smfCapabilitiesRssaID
      }
      STATUS  current
      DESCRIPTION
         "Set of SMF configuration objects implemented
          in this module."
```

```
   ::= { smfMIBGroups 1 }


smfConfigObjectsGroup OBJECT-GROUP
   OBJECTS {
           smfCfgAdminStatus,
           smfCfgSmfSysUpTime,
           smfCfgRouterIDAddrType,
           smfCfgRouterID,
           smfCfgOperationalMode,
           smfCfgRssaMember,
           smfCfgIpv4Dpd,
           smfCfgIpv6Dpd,
           smfCfgMaxPktLifetime,
           smfCfgDpdEntryMaxLifetime,
           smfCfgNhdpRssaMesgTLVIncluded,
           smfCfgNhdpRssaAddrBlockTLVIncluded,

           smfCfgAddrForwardingGroupName,
           smfCfgAddrForwardingAddrType,
           smfCfgAddrForwardingAddress,
           smfCfgAddrForwardingAddrPrefixLength,
           smfCfgAddrForwardingStatus,

           smfCfgIfName,
           smfCfgIfAdminStatus,
           smfCfgIfSmfUpTime,
           smfCfgIfRowStatus
   }
   STATUS  current
   DESCRIPTION
      "Set of SMF configuration objects implemented
       in this module."
::= { smfMIBGroups 2 }

smfStateObjectsGroup  OBJECT-GROUP
   OBJECTS {
           smfStateNodeRsStatusIncluded,
           smfStateDpdMemoryOverflow,

           smfStateNeighborRSSA,
           smfStateNeighborNextHopInterface
   }
   STATUS  current
   DESCRIPTION
      "Set of SMF state objects implemented
       in this module."
::= { smfMIBGroups 3 }
```

```
    smfPerfObjectsGroup  OBJECT-GROUP
       OBJECTS {
               smfPerfIpv4MultiPktsRecvTotal,
               smfPerfIpv4MultiPktsForwardedTotal,
               smfPerfIpv4DuplMultiPktsDetectedTotal,
               smfPerfIpv4DroppedMultiPktsTTLExceededTotal,
               smfPerfIpv4TTLLargerThanPreviousTotal,

               smfPerfIpv6MultiPktsRecvTotal,
               smfPerfIpv6MultiPktsForwardedTotal,
               smfPerfIpv6DuplMultiPktsDetectedTotal,
               smfPerfIpv6DroppedMultiPktsTTLExceededTotal,
               smfPerfIpv6TTLLargerThanPreviousTotal,
               smfPerfIpv6HAVAssistsReqdTotal,
               smfPerfIpv6DpdHeaderInsertionsTotal,

               smfPerfIpv4MultiPktsRecvPerIf,
               smfPerfIpv4MultiPktsForwardedPerIf,
               smfPerfIpv4DuplMultiPktsDetectedPerIf,
               smfPerfIpv4DroppedMultiPktsTTLExceededPerIf,
               smfPerfIpv4TTLLargerThanPreviousPerIf,

               smfPerfIpv6MultiPktsRecvPerIf,
               smfPerfIpv6MultiPktsForwardedPerIf,
               smfPerfIpv6DuplMultiPktsDetectedPerIf,
               smfPerfIpv6DroppedMultiPktsTTLExceededPerIf,
               smfPerfIpv6TTLLargerThanPreviousPerIf,
               smfPerfIpv6HAVAssistsReqdPerIf,
               smfPerfIpv6DpdHeaderInsertionsPerIf
       }
       STATUS  current
       DESCRIPTION
          "Set of SMF performance objects implemented
           in this module by total and per interface."
    ::= { smfMIBGroups 4 }

    smfNotifObjectsGroup  OBJECT-GROUP
       OBJECTS {
               smfNotifDpdMemoryOverflowThreshold,
               smfNotifDpdMemoryOverflowWindow
       }
       STATUS  current
       DESCRIPTION
          "Set of SMF notification control
           objects implemented in this module."
    ::= { smfMIBGroups 5 }

    smfNotificationsGroup  NOTIFICATION-GROUP
```

```
     NOTIFICATIONS {
             smfNotifAdminStatusChange,
             smfNotifConfiguredOpModeChange,
             smfNotifIfAdminStatusChange,
             smfNotifDpdMemoryOverflowEvent
     }
     STATUS  current
     DESCRIPTION
        "Set of SMF notifications implemented
         in this module."
    ::= { smfMIBGroups 6 }


     END
```

8. IANAsmfOpModeID-MIB Definitions

This section contains the IANAsmfOpModeID-MIB module. This MIB
module defines the single textual convention for which IANA SHOULD
maintain and keep synchronized with the registry identified below
within the IANAsmfOpModeIdTC TEXTUAL-CONVENTION.

The IANAsmfOpModeIdTC defines an index that identifies through
reference to a specific SMF operations mode. The index is an integer
valued named-number enumeration consisting of an integer and label.
IANA is to create and maintain this textual convention. Future
assignments are made to anyone on a first come, first served basis.
There is no substantive review of the request, other than to ensure
that it is well-formed and does not duplicate an existing assignment.
However, requests must include a minimal amount of clerical
information, such as a point of contact (including an email address)
and a brief description of the method being identified as a new SMF
operations mode.

```
     IANAsmfOpModeID-MIB DEFINITIONS ::= BEGIN

     IMPORTS
         MODULE-IDENTITY, mib-2
                   FROM SNMPv2-SMI
         TEXTUAL-CONVENTION
                   FROM SNMPv2-TC;

     ianasmfOpModeID MODULE-IDENTITY
         LAST-UPDATED "201403081300Z"  -- March 8, 2014
         ORGANIZATION "IANA"
         CONTACT-INFO "Internet Assigned Numbers Authority
```

```
                     Postal: ICANN
                             4676 Admiralty Way, Suite 330
                             Marina del Rey, CA 90292

                     Tel:    +1 310 823 9358
                     E-Mail: iana@iana.org"
        DESCRIPTION   "This MIB module defines the
                       IANAsmfOpModeIdTC Textual
                       Convention, and thus the enumerated values of
                       the smfCapabilitiesOpModeID object defined in
                       the SMF-MIB."
        REVISION      "201403081300Z"  -- March 8, 2014
        DESCRIPTION   "Initial version of this MIB as published in
                       RFC KKKK."
        ::= { mib-2 kkkk }

     IANAsmfOpModeIdTC ::= TEXTUAL-CONVENTION
        STATUS        current
        DESCRIPTION
           "An index that identifies through reference to a specific
            SMF operations mode.  There are basically three styles
            of SMF operation with reduced relay sets currently
            identified:

              Independent operation 'independent(1)' -
                  SMF performs its own relay
                  set selection using information from an associated
                  MANET NHDP process.

              CDS-aware unicast routing operation 'routing(2)'-
                  a coexistent unicast routing
                  protocol provides dynamic relay
                  set state based upon its own control plane
                  CDS or neighborhood discovery information.

              Cross-layer operation 'crossLayer(3)' -
                  SMF operates using neighborhood
                  status and triggers from a
                  cross-layer information base for dynamic relay
                  set selection and maintenance.

            IANA MUST update this textual convention accordingly.

            The definition of this textual convention with the
            addition of newly assigned values is published
            periodically by the IANA, in either the Assigned
            Numbers RFC, or some derivative of it specific to
            Internet Network Management number assignments.  (The
```

```
            latest arrangements can be obtained by contacting the
            IANA.)

            Requests for new values SHOULD be made to IANA via
            email (iana@iana.org)."
        REFERENCE
            "See Section 7.2. 'Reduced Relay Set Forwarding',
            and the Appendices A, B and C in
            RFC 6621 - Simplified Multicast Forwarding
            (SMF), Macker, J., May 2012."
        SYNTAX  INTEGER {
                        independent (1),
                        routing (2),
                        crossLayer (3)
                        -- future (4-255)
        }

    END
```

9. IANAsmfRssaID-MIB Definitions

This section contains the IANAsmfRssaID-MIB module. This MIB module
defines the single textual convention for which IANA SHOULD maintain
and keep synchronized with the registry identified below within the
IANAsmfRssadTC TEXTUAL-CONVENTION.

The IANAsmfRssaIdTC defines an index that identifies through
reference to a specific Reduced Set Selection Algorithm (RSSA). The
index is an integer valued named-number enumeration consisting of an
integer and label. IANA is to create and maintain this textual
convention.

Future assignments for the index range 5-127 require an RFC
publication (either as an IETF submission or as an RFC Editor
Independent submission [RFC3932]). The type of RFC MUST be Standards
Track. The specific RSSA algorithms MUST be documented in sufficient
detail so that interoperability between independent implementations
is possible.

Future assignments for the index range 128-239 are private or local
use only, with the type and purpose defined by the local site. No
attempt is made to prevent multiple sites from using the same value
in different (and incompatible) ways. There is no need for IANA to
review such assignments (since IANA will not record these) and
assignments are not generally useful for broad interoperability. It
is the responsibility of the sites making use of the Private Use
range to ensure that no conflicts occur (within the intended scope of
use).

Future assignments for the index range 240-255 are to facilitate
experimentation. These require an RFC publication (either as an IETF
submission or as an RFC Editor Independent submission [RFC3932]).
The type of RFC MUST be Experimental. The RSSA algorithms MUST be
documented in sufficient detail so that interoperability between
independent implementations is possible.


```
 IANAsmfRssaID-MIB DEFINITIONS ::= BEGIN

 IMPORTS
     MODULE-IDENTITY, mib-2
             FROM SNMPv2-SMI
     TEXTUAL-CONVENTION
             FROM SNMPv2-TC;

 ianasmfRssaID MODULE-IDENTITY
     LAST-UPDATED "201403081300Z"  -- March 8, 2014
     ORGANIZATION "IANA"
     CONTACT-INFO "Internet Assigned Numbers Authority

                    Postal: ICANN
                            4676 Admiralty Way, Suite 330
                            Marina del Rey, CA 90292

                    Tel:    +1 310 823 9358
                    E-Mail: iana@iana.org"
     DESCRIPTION  "This MIB module defines the
                   IANAsmfRssaIdTC Textual
                   Convention, and thus the enumerated values of
                   the smfCapabilitiesRssaID object defined in
                   the SMF-MIB."
     REVISION     "201403081300Z"  -- March 8, 2014
     DESCRIPTION  "Initial version of this MIB as published in
                   RFC LLLL."

     ::= { mib-2 llll }

  IANAsmfRssaIdTC ::= TEXTUAL-CONVENTION
     STATUS      current
     DESCRIPTION
         "An index that identifies through reference to a specific
          RSSA algorithms.  Several are currently defined
          in the Appendix A, B and C of RFC 6621.

          Examples of RSSA algorithms already identified within
          this TC are:
```

           Classical Flooding (cF(1)) - is the standard
              flooding algorithm where each node in the next
              retransmits the information on each of its interfaces.

           Source-Based Multipint Relay (sMPR(2)) -
              this algorithm is used by Optimized Link State Routing
              (OLSR) and OLSR version 2 (OLSRv2) protocols for the
              relay of link state updates and other control
              information [RFC3626]. Since each router picks
              its neighboring relays independently, sMPR
              forwarders depend upon previous hop information
              (e.g., source MAC address) to operate correctly.

           Extended Connected Dominating Set (eCDS(3)) -
              defined in [RFC5614] this algorithm forms a single
              CDS mesh for the SMF operating region. Its
              packet-forwarding rules are not dependent upon
              previous hop knowledge in contrast to sMPR.

           Multipoint Relay Connected Dominating Set (mprCDS(4)) -
              This algorithm is an extension to the basic sMPR
              election algorithm that results in a shared
              (non-source-specific) SMF CDS. Thus, its forwarding
              rules are not dependent upon previous hop information,
              similar to eCDS.

        IANA MUST update this textual convention accordingly.

        The definition of this textual convention with the
        addition of newly assigned values is published
        periodically by the IANA, in either the Assigned
        Numbers RFC, or some derivative of it specific to
        Internet Network Management number assignments. (The
        latest arrangements can be obtained by contacting the
        IANA.)

        Requests for new values SHOULD be made to IANA via
        email (iana@iana.org)."
    REFERENCE
       "See, e.g.,

        Section 8.1.1. 'SMF Message TLV Type',
        Appendix A. 'Essential Connecting Dominating Set (E-CDS)
             Algorithm',
        Appendix B. 'Source-Based Multipoint Relay (S-MPR)
             Algorithm', and
        Appendix C. 'Multipoint Relay Connected Dominating Set
              (MPR-CDS) Algorithm'

```
        in RFC 6621 - Simplified Multicast Forwarding
       (SMF), Macker, J., May 2012."
    SYNTAX      INTEGER {
                     cF(1),
                     sMPR(2),
                     eCDS(3),
                     mprCDS(4)
                     -- future(5-127)
                     -- noStdAction(128-239)
                     -- experimental(240-255)
                }

 END
```

10. Security Considerations

   This section discusses security implications of the choices made in
   this SMF-MIB module.

   There are a number of management objects defined in this MIB module
   with a MAX-ACCESS clause of read-write and/or read-create. Such
   objects may be considered sensitive or vulnerable in some network
   environments. The support for SET operations in a non-secure
   environment without proper protection can have a negative effect on
   network operations. These are the tables and objects and their
   sensitivity/vulnerability:

   o 'smfCfgAdminStatus' - this writable configuration object controls
      the operational status of the SMF process. If this setting is
      configured inconsistently across the MANET multicast domain, then
      delivery of multicast data may be inconsistent across the domain;
      some nodes may not receive multicast data intended for them.

   o 'smfCfgRouterIDAddrType' and 'smfCfgRouterID' - these writable
      configuration objects define the ID of the SMF process. These
      objects should be configured with a routable address defined on
      the local SMF device. The smfCfgRouterID is a logical
      identification that MUST be configured as unique across inter-
      operating SMF neighborhoods and it is RECOMMENDED to be chosen as
      the numerically largest address contained in a node's 'Neighbor
      Address List' as defined in NHDP. A smfCfgRouterID MUST be unique
      within the scope of the operating MANET network regardless of the
      method used for selecting it. If these objects are mis-configured
      or configured in-consistently across the MANET, then the ability
      of various RSSA algorithms, e.g., ECDS, may be compromised. This
      would potentially result in some routers within the MANET not
      receiving multicast packets destine to them. Hence, intentionally
      mis-configuring these objects could pose a form of Denial-of-

Service (DOS) attack against the MANET.

o 'smfCfgOpMode' - this writable configuration objects define the
   operational mode of the SMF process. The operational mode defines
   how the SMF process receives its data to form its local estimate
   of the CDS. It is recommended that the value for this object be
   set consistently across the MANET to ensure proper operation of
   the multicast packet forwarding. If the value for this object is
   set inconsistently across the MANET, the result may be that
   multicast packet delivery will be compromised within the MANET.
   Hence, intentionally mis-configuring this object could pose a form
   DOS attack against the MANET.

o 'smfCfgRssa' - this writable configuration object sets the
   specific Reduced Set Selection Algorithm (RSSA) for the SMF
   process. If this object is set inconsistently across the MANET
   domain, multicast delivery of data will likely fail. Hence,
   intentionally mis-configuring this object could pose a form DOS
   attack against the MANET.

o 'smfCfgRssaMember' - this writable configuration object sets the
   'interest' of the local SMF node in participating in the CDS.
   Setting this object to 'never(3)' on a highly highly connected
   device could lead to frequent island formation. Setting this
   object to 'always(2)' could support data ex-filtration from the
   MANET domain.

o 'smfCfgIpv4Dpd' - this writable configuration object sets the
   duplicate packet detection method, i.e., H-DPD or I-DPD, for
   forwarding of IPv4 multicast packets. Forwarders may operate with
   mixed H-DPD and I-DPD modes as long as they consistently perform
   the appropriate DPD routines outlined [RFC6621]. However, it is
   RECOMMENDED that a deployment be configured with a common mode for
   operational consistency.

o 'smfCfgIpv6Dpd' - this writable configuration object sets the
   duplicate packet detection method for forwarding of IPv6 multicast
   packets. Since IPv6 SMF does specifies an option header, the
   interoperability constraints are not as loose as in the IPv4
   version, and forwarders SHOULD not operate with mixed H-DPD and
   I-DPD modes. Hence the value for this object SHOULD be
   consistently set within the forwarders comprising the MANET, else
   inconsistent forwarding may result unnecessary multicast packet
   dropping.

o 'smfCfgMaxPktLifetime' - this writable configuration object sets
   the estimate of the network packet traversal time. If set too
   small, this could lead to poor multicast data delivery ratios

throughout the MANET domain. This could serve as a form of DOS
attack if this object value is set too small.

o 'smfCfgDpdEntryMaxLifetime' - this writable configuration object
   sets the maximum lifetime (in seconds) for the cached DPD records
   for the combined IPv4 and IPv6 methods. If the memory is running
   low prior to the MaxLifetime being exceeded, the local SMF devices
   should purge the oldest records first. If this object value is
   set too small, then the effectiveness of the SMF DPD algorithms
   may become greatly diminished causing a higher than necessary
   packet load on the MANET.

o 'smfCfgNhdpRssaMesgTLVIncluded' - this writable configuration
   object indicates whether the associated NHDP messages include the
   the RSSA Message TLV, or not. It is highly RECOMMENDED that this
   object be set to 'true(1)' when the SMF operation mode is set to
   independent as this information will inform the local forwarder of
   the RSSA algorithm implemented in neighboring forwarders and is
   used to ensure consistent forwarding across the MANET. While it
   is possible that SMF neighbors MAY be configured differently with
   respect to the RSSA algorithm and still operate cooperatively, but
   these cases will vary dependent upon the algorithm types
   designated and this situation SHOULD be avoided.

o 'smfCfgNhdpRssaAddrBlockTLVIncluded' - this writable configuration
   object indicates whether the associated NHDP messages include the
   the RSSA Address Block TLV, or not. The
   smfNhdpRssaAddrBlockTLVIncluded is optional in all cases as it
   depends on the existence of an address block which may not be
   present. If this SMF device is configured with NHDP, then this
   object should be set to 'true(1)' as this TLV enables CDS relay
   algorithm operation and configuration to be shared among 2-hop
   neighborhoods. Some relay algorithms require 2-hop neighbor
   configuration in order to correctly select relay sets.

o 'smfCfgAddrForwardingTable' - the writable configuration objects
   in this table indicate which multicast IP addresses are to be
   forwarded by this SMF node. Misconfiguration of rows within this
   table can limit the ability of this SMF device to properly forward
   multicast data.

o 'smfCfgInterfaceTable' - the writable configuration objects in
   this table indicate which SMF node interfaces are participating in
   the SMF packet forwarding process. Misconfiguration of rows
   within this table can limit the ability of this SMF device to
   properly forward multicast data.

Some of the readable objects in this MIB module (i.e., objects with a

MAX-ACCESS other than not-accessible) may be considered sensitive or
vulnerable in some network environments. It is thus important to
control even GET and/or NOTIFY access to these objects and possibly
to even encrypt the values of these objects when sending them over
the network via SNMP. These are the tables and objects and their
sensitivity/vulnerability:

o 'smfNodeRsStatusIncluded' - this readable state object indicates
   that this SMF node is part of the CDS, or not. Being part of the
   CDS makes this node a distinguished device. It could be exploited
   for data ex-filtration, or denial of service attacks.

o 'smfStateNeighborTable' - the readable state objects in this table
   indicate current neighbor nodes to this SMF node. Exposing this
   information to an attacker could allow the attacker easier access
   to the larger MANET domain.

The remainder of the objects in the SMF-MIB module are performance
counter objects. While these give an indication of the activity of
the SMF process on this node, it is not expected that exposing these
values pose a security risk to the MANET network.

SNMP versions prior to SNMPv3 did not include adequate security.
Even if the network itself is secure (for example by using IPSec),
even then, there is no control as to who on the secure network is
allowed to access and GET/SET (read/change/create/delete) the objects
in this MIB module.

Implementations MUST provide the security features described by the
SNMPv3 framework (see [RFC3410]), including full support for
authentication and privacy via the User-based Security Model (USM)
[RFC3414] with the AES cipher algorithm [RFC3826]. Implementations
MAY also provide support for the Transport Security Model (TSM)
[RFC5591] in combination with a secure transport such as SSH
[RFC5592] or TLS/DTLS [RFC6353].

Further, deployment of SNMP versions prior to SNMPv3 is NOT
RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to
enable cryptographic security. It is then a customer/operator
responsibility to ensure that the SNMP entity giving access to an
instance of this MIB module is properly configured to give access to
the objects only to those principals (users) that have legitimate
rights to indeed GET or SET (change/create/delete) them.

11. Applicability Statement

This document describes objects for configuring parameters of the
Simplified Multicast Forwarding [RFC6621] process on a Mobile Ad-Hoc

Network (MANET) router. This MIB module, denoted SMF-MIB, also
reports state and performance information and notifications. This
section provides some examples of how this MIB module can be used in
MANET network deployments. A fuller discussion of MANET network
management use cases and challenges will be provided elsewhere.

SMF is designed to allow MANET routers to forward IPv4 and IPv6
packets over the MANET and cover the MANET nodes through the
automatic discovery of efficient estimates of the Minimum Connected
Dominating Set (MCDS) of nodes within the MANET. The MCDS are
estimated using the Relay Set Selection Algorithms (RSSAs) discussed
within this document. In the following, three scenarios are listed
where this MIB module is useful, i.e.,

o For a Parking Lot Initial Configuration Situation - it is common
   for the vehicles comprising the MANET being forward deployed at a
   remote location, e.g., the site of a natural disaster, to be off-
   loaded in a parking lot where an initial configuration of the
   networking devices is performed. The configuration is loaded into
   the devices from a fixed location Network Operation Center (NOC)
   at the parking lot and the vehicles are stationary at the parking
   lot while the configuration changes are made. Standards-based
   methods for configuration management from the co-located NOC are
   necessary for this deployment option. The set of interesting
   configuration objects for the SMF process are listed within this
   MIB module.

o For Mobile vehicles with Low Bandwidth Satellite Link to a Fixed
   NOC - Here the vehicles carrying the MANET routers carry multiple
   wireless interfaces, one of which is a relatively low-bandwidth
   on-the-move satellite connection which interconnects a fix NOC to
   the nodes of the MANET. Standards-based methods for monitoring
   and fault management from the fixed NOC are necessary for this
   deployment option.

o For Fixed NOC and Mobile Local Manager in Larger Vehicles - for
   larger vehicles, a hierarchical network management arrangement is
   useful. Centralized network management is performed from a fixed
   NOC while local management is performed locally from within the
   vehicles. Standards-based methods for configuration, monitoring
   and fault management are necessary for this deployment option.

Here we provide an example of the simplest of configurations to
establish an operational multicast forwarding capability in a MANET.
This discussion only identifies the configuration necessary through
the SMF-MIB module and assumes that other configuration has occurred.
Assume that the MANET is to support only IPv4 addressing and that the
MANET nodes are to be configured in the context of the Parking Lot

Initialization case above. Then the SMF-MIB module defines ten
configuration OIDs and two configuration tables, i.e., the
smfCfgAddrForwardingTable and the smfCfgInterfaceTable. Of the ten
OIDs defined, all but one, i.e., the smfCfgRouterID, have DEFVAL
clauses which allow for a functional configuration of the SMF process
within the MANET. The smfCfgRouterIDType defaults to 'ipv4' so the
smfCfgRouterID can be set as, e.g.,:

snmpset [options] <smfCfgRouterID_OID>.0 a 192.168.1.100

If the smfCfgAddrForwardingTable is left empty, then the SMF local
forwarder will forward all multicast addresses. So this table does
not require configuration if you want to have the MANET forward all
multicast addresses.

All that remains is to configure at least one row in the
smfCfgInterfaceTable. Assume that the node has a wireless interface
with an <ifName>='wlan0' and an <ifIndex>='1'. All of the objects in
the rows of the smfCfgInterfaceTable have a DEFVAL clause, hence only
the RowStatus object needs to be set. So the SMF process will be
activated on the 'wlan0' interface by the following network manager
snmpset command:

snmpset [options] <smfCfgIfRowStatus>.1 i active(1)

At this point, the configured forwarder will begin a Classical
Flooding algorithm to forward all multicast addresses IPv4 packets it
receives. If the

To provide a more efficient multicast forwarding within the MANET,
the network manager could walk the smfCapabilitiesTable to identify
other SMF operational modes, e.g.,:

snmpwalk -c public -v 2c router <smfCapabilitiesTable>

SMF-MIB::smfCapabilitiesIndex.1 = INTEGER: 1

SMF-MIB::smfCapabilitiesIndex.2 = INTEGER: 2

SMF-MIB::smfCapabilitiesOpModeID.1 = INTEGER: cfOnly(1)

SMF-MIB::smfCapabilitiesOpModeiD.2 = INTEGER: independent(2)

SMF-MIB::smfCapabilitiesRssaID.1 = INTEGER: cF(1)

SMF-MIB::smfCapabilitiesRssaID.2 = INTEGER: eCDS(3)

In this example, the forwarding device also supports the Extended

Connected Dominating Set (eCDS) RSSA with the forwarder in the
'independent(2)' operational mode. If the network manager were to
then issue an snmpset, e.g.,:

snmpset [options] <smfCfgOperationalMode>.0 i 2

then the local forwarder would switch if forwarding behavior from
Classical Flooding to the more efficient eCDS flooding.

12. IANA Considerations

The MIB module in this document uses the following IANA-assigned
OBJECT IDENTIFIER values recorded in the SMI Numbers registry:

```
    Descriptor            OBJECT IDENTIFIER value
    ----------            -----------------------
    SMF-MIB               { experimental XXXX }
    IANAsmfOpModeID-MIB   { experimental KKKK }
    IANAsmfRssaID-MIB     { experimental LLLL }

    IANA EDITOR NOTE: please assign XXXX, KKKK and LLLL and
    remove this note.
```

13. Contributors

This MIB document uses the template authored by D. Harrington which
is based on contributions from the MIB Doctors, especially Juergen
Schoenwaelder, Dave Perkins, C.M.Heard and Randy Presuhn.

14. Acknowledgements

The authors would like to acknowledge the valuable comments from Sean
Harnedy in the early phases of the development of this MIB module and
from Dan Romascanu in the final reviews of this MIB module. The
authors would like to thank James Nguyen for his careful review and
comments on this MIB module and his work on the definitions of the
follow-on MIB modules to configure specific RSSA algorithms related
to SMF. Further, the authors would like to acknowledge to work of
James Nguyen, Brian Little, Ryan Morgan and Justin Dean on their
software development of the SMF-MIB.

15. References

15.1. Normative References

[RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group
           MIB", RFC 2863, June 2000.

   [RFC3411]   Harrington, D., Presuhn, R., and B. Wijnen, "An
               Architecture for Describing Simple Network Management
               Protocol (SNMP) Management Frameworks", STD 62, RFC 3411,
               December 2002.

   [RFC3418]   Presuhn, R., "Management Information Base (MIB) for the
               Simple Network Management Protocol (SNMP)", STD 62,
               RFC 3418, December 2002.

   [RFC4001]   Daniele, M., Haberman, B., Routhier, S., and J.
               Schoenwaelder, "Textual Conventions for Internet Network
               Addresses", RFC 4001, February 2005.

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

   [RFC2578]   McCloghrie, K., Ed., Perkins, D., Ed., and J.
               Schoenwaelder, Ed., "Structure of Management Information
               Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.

   [RFC2579]   McCloghrie, K., Ed., Perkins, D., Ed., and J.
               Schoenwaelder, Ed., "Textual Conventions for SMIv2",
               STD 58, RFC 2579, April 1999.

   [RFC2580]   McCloghrie, K., Perkins, D., and J. Schoenwaelder,
               "Conformance Statements for SMIv2", STD 58, RFC 2580,
               April 1999.

   [RFC6621]   Macker, J., "Simplified Multicast Forwarding", RFC 6621,
               May 2012.

15.2. Informative References

   [RFC3410]   Case, J., Mundy, R., Partain, D., and B. Stewart,
               "Introduction and Applicability Statements for Internet-
               Standard Management Framework", RFC 3410, December 2002.

   [RFC3414]   Blumenthal, U. and B. Wijnen, "User-based Security Model
               (USM) for version 3 of the Simple Network Management
               Protocol (SNMPv3)", STD 62, RFC 3414, December 2002.

   [RFC3826]   Blumenthal, U., Maino, F., and K. McCloghrie, "The
               Advanced Encryption Standard (AES) Cipher Algorithm in the
               SNMP User-based Security Model", RFC 3826, June 2004.

   [RFC5591]   Harrington, D. and W. Hardaker, "Transport Security Model
               for the Simple Network Management Protocol (SNMP)",
               RFC 5591, June 2009.

   [RFC5592] Harrington, D., Salowey, J., and W. Hardaker, "Secure
              Shell Transport Model for the Simple Network Management
              Protocol (SNMP)", RFC 5592, June 2009.

   [RFC6353] Hardaker, W., "Transport Layer Security (TLS) Transport
              Model for the Simple Network Management Protocol (SNMP)",
              RFC 6353, July 2011.

Appendix A.

   ****
   * Note to the RFC Editor (to be removed prior to publication) *
   *   *
   *   *
   * 1) The reference to RFCXXXX throughout this document point   *
   * to the current draft-ietf-manet-smf-xx.txt. This needs   *
   * to be replaced with the XXXX RFC number for the SMF   *
   * publication.   *
   *   *
   * 2) This document also contains the IANAsmfOpModeID-MIB   *
   * module which is defined by this specification above.   *
   * IANA should take over the IANAsmfOpModeID-MIB and keep it   *
   * synchronized with the registry identified within the   *
   * contained IANAsmfOpModeIdTC TEXTUAL-CONVENTION.   *
   *   *
   * 3) This document contains the IANAsmfRssaID-MIB   *
   * module which is defined by this specification above.   *
   * IANA should take over the IANAsmfRssaID-MIB and keep it   *
   * synchronized with the registry identified within the   *
   * contained IANAsmfRssaID TEXTUAL-CONVENTION.   *
   *   *
   ****

Authors' Addresses

   Robert G. Cole
   US Army CERDEC
   6010 Frankford Road
   Aberdeen Proving Ground, Maryland 21005
   USA

   Phone: +1 443 395 8744
   EMail: robert.g.cole@us.army.mil

Joseph Macker
Naval Research Laboratory
Washington, D.C.   20375
USA

EMail: macker@itd.nrl.navy.mil

Brian Adamson
Naval Research Laboratory
Washington, D.C.   20375
USA

EMail: adamson@itd.nrl.navy.mil