VRRP working group
Internet Draft
Document: draft-ietf-vrrp-ipv6-mib-00.txt
Expires: December 2002

k. Tata
Nokia inc.
k. Karlekar
Nokia inc.
B. Jewell
Copper Mountain
Networks Inc.
June 2003

Definitions of Managed Objects for the VRRP IPv6
draft-ietf-vrrp-ipv6-mib-00.txt

Status of this Memo

Abstract

This specification defines an extension to the Management Information Base (MIB) for use with SNMP-based network management. In particular, it defines objects for configuring, monitoring, and controlling routers that employ the Virtual Router Redundancy Protocol for IPv6 as defined in draft-ietf-vrrp-ipv6-spec-04.txt [19].

This memo specifies a MIB module in a manner that is compliant with SMIv2 [5], and semantically identical to the SMIv1 definitions [2].

Conventions used in this document

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 [2].

Table of Contents

1. The SNMP Management Framework

The SNMP Management Framework presently consists of five major components:

- An overall architecture, described in RFC 2571 [1].
- Mechanisms for describing and naming objects and events for the purpose of management. The first version of this Structure of Management Information (SMI) is called SMIv1 and described in STD 16, RFC 1155 [2], STD 16, RFC 1212 [3] and RFC 1215 [4]. The second version, called SMIv2, is described in STD 58, RFC 2578 [5], STD 58, RFC 2579 [6] and STD 58, RFC 2580 [7].
- Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and described in STD 15, RFC 1157 [8]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in RFC 1901 [9] and RFC 1906 [10]. The third version of the message protocol is called SNMPv3 and described in RFC 1906 [10], RFC 2572 [11] and RFC 2574 [12].
- Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in STD 15, RFC 1157 [8]. A second set of protocol operations and associated PDU formats is described in RFC 1905 [13].
- A set of fundamental applications described in RFC 2573 [14] and the view-based access control mechanism described in RFC 2575 [15].

A more detailed introduction to the current SNMP Management Framework can be found in RFC 2570 [16].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the mechanisms defined in the SMI.

This memo specifies a MIB module that is compliant to the SMIv2. A MIB conforming to the SMIv1 can be produced through the appropriate translations. The resulting translated MIB must be semantically equivalent, except where objects or events are omitted because no translation is possible (use of Counter64). Some machine readable information in SMIv2 will be converted into textual descriptions in SMIv1 during the translation process. However, this loss of machine readable information is not considered to change the semantics of the MIB.

2. Overview

VRRP protocols as defined in RFC 2338 [17] and draft-ietf-vrrp-ipv6-spec-04.txt [19] are inherently IP version specific. Though both the protocols are similar they are not identical and can coexist on a network element. Network managers should be able to configure and monitor these protocols independently. RFC 2787 [21] defines managed objects that are specific to VRRP protocol for IPv4 networks as defined in RFC 2338 [17]. This document defines managed objects for configuring and monitoring VRRP protocol for IPv6 networks as defined in draft-ietf-vrrp-ipv6-spec-04.txt [19]

IPv6 hosts on a LAN will usually learn about one or more default routers by receiving Router Advertisements sent using the IPv6 Neighbor Discovery protocol [ND]. The Router Advertisements are multicast periodically at a rate that the hosts will learn about the default routers in a few minutes. They are not sent frequently enough to rely on the absence of the router advertisement to detect router failures.

The Virtual Router Redundancy Protocol for IPv6 provides a much faster switch over to an alternate default router than can be obtained using standard ND procedures. Using VRRP for IPv6 a backup router can take over for a failed default router in around three seconds (using VRRP for IPv6 default parameters). This is done with out any interaction with the hosts and a minimum amount of VRRP traffic.

2.1 Virtual Router Redundancy Protocol for IPv6

This MIB is based on the following characteristics of VRRP as defined in the VRRP for IPv6 specification [19].

- A "VRRP IPv6 router" is one that is configured to run the VRRP for IPv6 protocol in conjunction with one or more other VRRP IPv6 routers attached to a LAN.
- A VRRP IPv6 router can be running one or more instances of a virtual router.
- A "IPv6 virtual router" is an abstraction, which consists of two or more physical routers associated by a IPv6 Virtual Router Identifier (IPv6VRID).
- An instance of a virtual router (on a physical VRRP IPv6 router), can be uniquely identified by a combination of the 'ifIndex' [18] and "Virtual Router Identifier" (IPv6VRID).

2.2 VRRP IPv6 MIB Structure

The VRRP IPv6 MIB contains three conformance groups:

- vrrpIpv6Operations Group: Objects related to VRRP IPv6 router's configuration and control.
- vrrpIpv6Statistics Group: Objects containing information useful in monitoring the operation of VRRP IPv6 routers.
- vrrpIpv6Notifications Group: Consists of objects and definitions for use in SNMP notifications sent by VRRP Ipv6 routers.

Tables in the MIB include the following:

(1)The vrrpIpv6OperTable, which contains objects that define the operational characteristics of a VRRP IPv6 router. Rows in this table correspond to instances of virtual routers.

(2)The vrrpIpv6RouterStatsTable which contains the operating statistics for a VRRP IPv6 router.

2.3 VRRP for IPv6 MIB Table Design

The tables in the VRRP for IPv6 MIB are structured with the assumption that a VRRP network management application would likely be designed to display information or provide configuration about a IPv6 VRRP router on a "per-IPv6-virtual-router basis". Thus, the tables defined in the MIB consist of conceptual rows which are grouped in a manner to present a view of individual virtual routers with a minimal number of SNMP operations.

2.4 Relation to Interface Group (RFC 2233) [18]

Since a router can be participating in VRRP over IPv6 on one or more physical interfaces, "ifIndex" is used as an index into the tables defined in the VRRP IPv6 MIB.

2.5 VRRP IPv6 Scenario

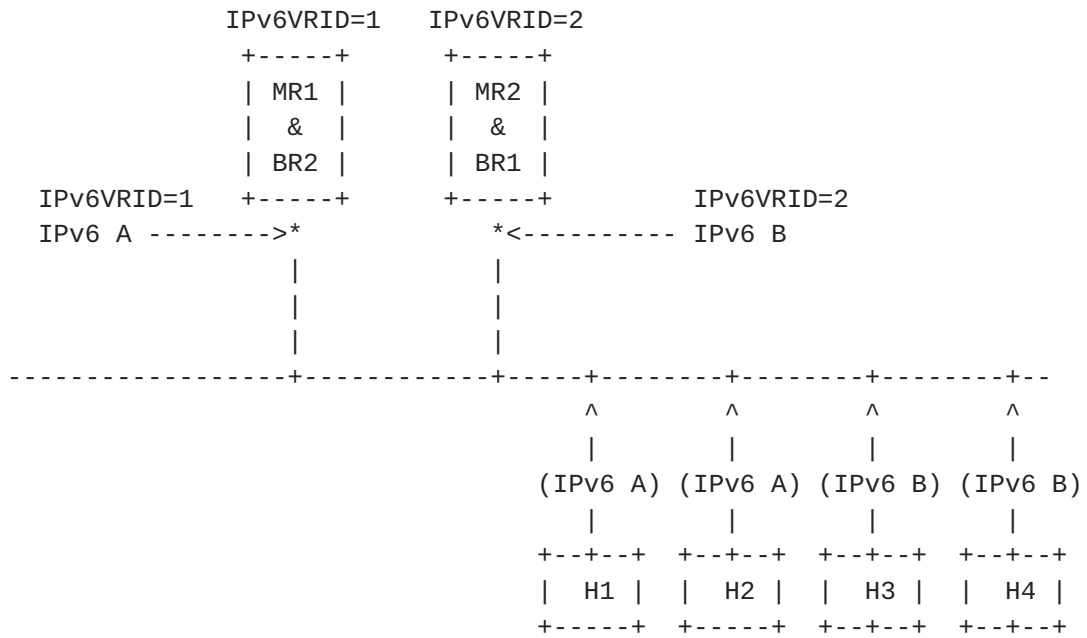
The following section provides examples of how some of the objects in this MIB are instantiated for two different VRRP IPv6 scenarios.

KEY:

The labels in the following tables and diagrams correspond to the actual MIB objects as follows:

```
if      = vrrpIpv6OperIfIndex
IPv6VrId= vrrpIpv6OperVrId
State   = vrrpIpv6OperState
Prior   = vrrpIpv6OperPriority
AddrType= vrrpIpv6OperIpAddrType
IpAddr  = vrrpIpv6OperMasterIpAddr
RowStat = vrrpIpv6OperRowStatus
```

The following figure shows a simple network with two VRRP IPv6 routers configured with two virtual routers. This sample topology is taken from the VRRP specification [17]. Addresses in '()' indicate the IPv6 address of the default gateway for a given host, H1 - H4. In the diagram, "Interface" is used in the context defined in IF-MIB [18].



----- MIB Tables For VRRP IPv6 Router "IPv6 A": -----

vrrpOperTable

if	IPv6VrId	State	Prior	AddrType	IpAddr	...	RowStat
I1	01	M	255	2	A		active
I1	02	B	1-254	2	B		active

----- MIB Tables For VRRP IPv6 Router "IPv6 B": -----

vrrpIPv6OperTable

if	IPv6VrId	State	Prior	AddrType	IpAddr	...	RowStat
I2	01	B	1-254	2	A		active
I2	02	M	255	2	B		active

NOTES:

1) "I1" and "I2" are used to designate IF indices on each respective router.
2) For "State": M = Master; B = Backup.
3) In the vrrpIpv6OperTable, a "priority" of 255 indicates that the respective router owns the IPv6 address, e.g., this IPv6 address is native to the router (i.e., "the IPv6 Address Owner" [17]).

3. Definitions

```
VRRP-IPv6-MIB DEFINITIONS ::= BEGIN

IMPORTS
     MODULE-IDENTITY, OBJECT-TYPE,
     NOTIFICATION-TYPE, Counter32,
     Integer32, mib-2         FROM SNMPv2-SMI

     RowStatus,
     MacAddress, TruthValue, TimeStamp   FROM SNMPv2-TC

     MODULE-COMPLIANCE, OBJECT-GROUP,
     NOTIFICATION-GROUP                  FROM SNMPv2-CONF
     ifIndex                             FROM IF-MIB
     VrId                                FROM VRRP-MIB
     InetAddressType, InetAddress        FROM INET-ADDRESS-MIB;


vrrpIpv6MIB  MODULE-IDENTITY
     LAST-UPDATED "200304200000Z"
     ORGANIZATION "IETF VRRP Working Group"
     CONTACT-INFO "Kalyan Tata
     Postal: Nokia, Inc.
             313, Fair child Dr.
             Mountain View, California 94303
     Tel:    +1 408  896 6493
     E-Mail: kalyan.tata@nokia.com"

     DESCRIPTION
         "This MIB describes objects used for managing Virtual Router
          Redundancy Protocol (VRRP) for IPv6 routers.

          Copyright (C) The Internet Society (2003).  This version
          of this MIB module is part of RFC XXXX: see the RFC
          itself for full legal notices.
          RFC Ed.: replace XXXX with assigned number & remove
          this note.
         "
     ::= { mib-2 xx } -- To be assigned by IANA

--*****************************************************************
--  VRRP MIB Groups
--*****************************************************************

 vrrpIpv6Operations      OBJECT IDENTIFIER ::= { vrrpIpv6MIB 1 }
 vrrpIpv6Statistics      OBJECT IDENTIFIER ::= { vrrpIpv6MIB 2 }
 vrrpIpv6Conformance     OBJECT IDENTIFIER ::= { vrrpIpv6MIB 3 }
```

```
--*****************************************************************
--  Start of MIB objects
--*****************************************************************

 vrrpIpv6NodeVersion  OBJECT-TYPE
     SYNTAX       Integer32
     MAX-ACCESS   read-only
     STATUS       current
     DESCRIPTION
         "This value identifies the particular version of the VRRP
         over IPv6 supported by this node. Version 3 for this
         document."
     ::= { vrrpIpv6Operations 1 }

 vrrpIpv6NotificationCntl  OBJECT-TYPE
     SYNTAX       INTEGER {
         enabled     (1),
         disabled    (2)
     }
     MAX-ACCESS   read-write
     STATUS       current
     DESCRIPTION
        "Indicates whether the VRRP-IPv6-enabled router will generate
         SNMP traps for events defined in this MIB. 'Enabled'
         results in SNMP traps; 'disabled', no traps are sent."
     DEFVAL { enabled }
     ::= { vrrpIpv6Operations 2 }

--*****************************************************************
-- VRRP IPv6 Operations Table
--*****************************************************************

 vrrpIpv6OperTable OBJECT-TYPE
     SYNTAX       SEQUENCE OF VrrpIpv6OperEntry
     MAX-ACCESS   not-accessible
     STATUS       current
     DESCRIPTION
         "Operations table for a VRRP router which consists of a
         sequence (i.e., one or more conceptual rows) of
         'vrrpIpv6OperEntry' items."

     ::= { vrrpIpv6Operations 3 }

 vrrpIpv6OperEntry OBJECT-TYPE
     SYNTAX       VrrpIpv6OperEntry
     MAX-ACCESS   not-accessible
     STATUS       current
```

```
        DESCRIPTION
            "An entry in the vrrpIpv6OperTable containing the
            operational characteristics of a virtual router. On a VRRP
            IPv6 router, a given virtual router is identified by a
            Combination of the IF index and IPv6VRID.

             Rows in the table cannot be modified unless the value
             of `vrrpIpv6OperAdminState' is `disabled' and the
             `vrrpIpv6OperState' has transitioned to `initialize'."

        INDEX    { ifIndex, vrrpIpv6OperVrId }
        ::= { vrrpIpv6OperTable 1 }

    VrrpIpv6OperEntry ::=
        SEQUENCE {
            vrrpIpv6OperVrId
                VrId,
            vrrpIpv6OperVirtualMacAddr
                MacAddress,
            vrrpIpv6OperState
                INTEGER,
            vrrpIpv6OperAdminState
                INTEGER,
            vrrpIpv6OperPriority
                Integer32,
            vrrpIpv6OperIpAddrType
                InetAddressType,
            vrrpIpv6OperMasterIpAddr
                InetAddress,
            vrrpIpv6OperAdvInterval
                Integer32,
            vrrpIpv6OperPreemptMode
                TruthValue,
            vrrpIpv6OperVirtualRouterUpTime
                TimeStamp,
            vrrpIpv6OperProtocol
                INTEGER,
            vrrpIpv6OperRowStatus
                RowStatus
    }

    vrrpIpv6OperVrId OBJECT-TYPE
        SYNTAX       VrId
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "This object contains the IPv6 Virtual Router Identifier
            (IPv6VRID)."
```

```
        ::= { vrrpIpv6OperEntry 1 }
```

```
    vrrpIpv6OperVirtualMacAddr OBJECT-TYPE
        SYNTAX       MacAddress
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
          "The virtual MAC address of the virtual router. Although this
          object can be derived from the 'vrrpIpv6OperVrId'object, it is
          defined so that it is easily obtainable by a management
          application and can be included in VRRP-related SNMP traps."
        ::= { vrrpIpv6OperEntry 2 }

    vrrpIpv6OperState OBJECT-TYPE
        SYNTAX       INTEGER {
            initialize(1),
            backup(2),
            master(3)
        }
        MAX-ACCESS   read-only
        STATUS       current

        DESCRIPTION
            "The current state of the virtual router. This object has
            three defined values:

            - `initialize', which indicates that all the
                virtual router is waiting for a startup event.

            - `backup', which indicates the virtual router is
                monitoring the availability of the master router.

            - `master', which indicates that the virtual router
                is forwarding packets for IPv6 addresses that are
                associated with this router.

            Setting the `vrrpIpv6OperAdminState' object(below) initiates
            transitions in the value of this object."
        ::= { vrrpIpv6OperEntry 3 }

    vrrpIpv6OperAdminState OBJECT-TYPE
        SYNTAX       INTEGER {
            up(1),
            down(2)
        }
        MAX-ACCESS   read-create
        STATUS       current
        DESCRIPTION
            "This object will enable/disable the virtual router
            function. Setting the value to `up', will transition
```

```
            the state of the virtual router from `initialize' to
            `backup' or `master', depending on the value of
            `vrrpIpv6OperPriority'. Setting the value to `down', will
            transition  the router from `master' or `backup' to
            `initialize'. State transitions may not be immediate; they
            sometimes depend on other factors, such as the interface
            (IF) state.

            The `vrrpIpv6OperAdminState' object must be set to `down'
            prior to modifying the other read-create objects in the
            conceptual row. The value of the `vrrpIpv6OperRowStatus'
            object (below) must be `active', signifying that the
            conceptual row is valid (i.e., the objects are correctly
            set), in order for this object to be set to `up'."
        DEFVAL    { down }
        ::= { vrrpIpv6OperEntry 4 }

    vrrpIpv6OperPriority OBJECT-TYPE
        SYNTAX       Integer32 (0..255)
        MAX-ACCESS   read-create
        STATUS       current
        DESCRIPTION
            "This object specifies the priority to be used for the
            virtual router master election process. Higher values imply
            higher priority.

            A priority of '0', although not settable, is sent by
            the master router to indicate that this router has ceased
            to participate in VRRP and a backup virtual router should
            transition  to become a new master.

            A priority of 255 is used for the router that owns the
            associated IP address(es)."
        DEFVAL       { 100 }
        ::= { vrrpIpv6OperEntry 5 }

    vrrpIpv6OperIpAddrType OBJECT-TYPE
        SYNTAX       InetAddressType
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "This specifies the the type of inetAddress in this row.
             This should allways be ipv6(2) for this document."
        ::= { vrrpIpv6OperEntry 6 }

    vrrpIpv6OperMasterIpAddr OBJECT-TYPE
        SYNTAX       InetAddress
        MAX-ACCESS   read-only
```

```
        STATUS       current
```

```
        DESCRIPTION
            "The master router's real (primary) IPv6 address. This is
            the IPv6 address listed as the source in VRRP advertisement
            last received by this virtual router."
        ::= { vrrpIpv6OperEntry 7 }

    vrrpIpv6OperAdvInterval OBJECT-TYPE
        SYNTAX       Integer32 (1..255)
        UNITS        "seconds"
        MAX-ACCESS   read-create
        STATUS       current
        DESCRIPTION
            "The time interval, in seconds, between sending
            advertisement messages. Only the master router sends
            VRRP advertisements."
        DEFVAL       { 1 }
        ::= { vrrpIpv6OperEntry 8 }

    vrrpIpv6OperPreemptMode OBJECT-TYPE
        SYNTAX       TruthValue
        MAX-ACCESS   read-create
        STATUS       current
        DESCRIPTION
            "Controls whether a higher priority virtual router will
            preempt a lower priority master."
        DEFVAL       { true }
        ::= { vrrpIpv6OperEntry 9 }

    vrrpIpv6OperVirtualRouterUpTime OBJECT-TYPE

        SYNTAX       TimeStamp
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "This is the value of the `sysUpTime' object when this
            virtual router (i.e., the `vrrpIpv6OperState') transitioned
            out of `initialized'."
        ::= { vrrpIpv6OperEntry 10 }

    vrrpIpv6OperProtocol OBJECT-TYPE
        SYNTAX   INTEGER {
            ip (1),
            bridge (2),
            decnet (3),
            other (4)
        }
        MAX-ACCESS read-create
        STATUS     current
```

DESCRIPTION

```
            "The particular protocol being controlled by this Virtual
            Router.

            New enumerations to this list can only be added via a new
            RFC on the standards track."
        DEFVAL { ip }
        ::= { vrrpIpv6OperEntry 11 }

    vrrpIpv6OperRowStatus OBJECT-TYPE
        SYNTAX       RowStatus
        MAX-ACCESS   read-create
        STATUS       current
        DESCRIPTION
            "The row status variable, used in accordance to installation
            and removal conventions for conceptual rows. The rowstatus
            of a currently active row in the vrrpIpv6OperTable is
            constrained by the operational state of the corresponding
            virtual router.
            When `vrrpIpv6OperRowStatus' is set to active(1), no other
            objects in the conceptual row, with the exception of
            `vrrpIpv6OperAdminState', can be modified. Prior to setting
            the `vrrpIpv6OperRowStatus' object from `active' to a
            different value, the `vrrpIpv6OperAdminState' object must be
            set to `down' and the `vrrpIpv6OperState' object be
            transitioned to `initialize'.

            To create a row in this table, a manager sets this object
            to either createAndGo(4) or createAndWait(5). Until
            instances of all corresponding columns are appropriately
            configured, the value of the corresponding instance of the
            `vrrpIpv6OperRowStatus' column will be read as notReady(3).
            In particular, a newly created row cannot be made active(1)
            until (minimally) the corresponding instance of
            `vrrpIpv6OperVrId' has been set."
        ::= { vrrpIpv6OperEntry 12
    }


    --*****************************************************************
    -- VRRP IPv6 Router Statistics
    --*****************************************************************

    vrrpIpv6RouterChecksumErrors OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The total number of VRRP packets received with an invalid
```

VRRP checksum value."

```
        ::= { vrrpIpv6Statistics 1 }

    vrrpIpv6RouterVersionErrors OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The total number of IPv6 VRRP packets received with an
            unknown or unsupported version number."
        ::= { vrrpIpv6Statistics 2 }

    vrrpIpv6RouterVrIdErrors OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current

        DESCRIPTION
            "The total number of IPv6 VRRP packets received with an
            invalid IPv6VRID for this virtual router."
        ::= { vrrpIpv6Statistics 3 }

   --*****************************************************************
   -- VRRP IPv6 Router Statistics Table
   --*****************************************************************

    vrrpIpv6RouterStatsTable OBJECT-TYPE
        SYNTAX       SEQUENCE OF VrrpRouterStatsEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
            "Table of virtual router statistics."
        ::= { vrrpIpv6Statistics 4 }

    vrrpIpv6RouterStatsEntry OBJECT-TYPE
        SYNTAX       VrrpRouterStatsEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
            "An entry in the table, containing statistics information
            about a given virtual router."
        AUGMENTS    { vrrpIpv6OperEntry }
        ::= { vrrpIpv6RouterStatsTable 1 }

    VrrpRouterStatsEntry ::=
        SEQUENCE {
            vrrpIpv6StatsBecomeMaster
                Counter32,
            vrrpIpv6StatsAsMasterUpTime
```

```
                TimeStamp,
```

```
            vrrpIpv6StatsAdvRcvd
                Counter32,
            vrrpIpv6StatsAdvIntervalErrors
                Counter32,
            vrrpIpv6StatsHopLimitErrors
                Counter32,
            vrrpIpv6StatsPriZeroPktsRcvd
                Counter32,
            vrrpIpv6StatsPriZeroPktsSent
                Counter32,
            vrrpIpv6StatsInvalidTypePktsRcvd
                Counter32,
            vrrpIpv6StatsAddressListErrors
                Counter32,
            vrrpIpv6StatsPacketLengthErrors
                Counter32
        }

    vrrpIpv6StatsBecomeMaster OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The total number of times that this virtual router's state
            has transitioned to MASTER."
        ::= { vrrpIpv6RouterStatsEntry 1 }

    vrrpIpv6StatsAsMasterUpTime OBJECT-TYPE
        SYNTAX      TimeStamp
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The total number of seconds this router is UP in master
            state."
        ::= { vrrpIpv6RouterStatsEntry 2 }

    vrrpIpv6StatsAdvRcvd OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The total number of IPv6 VRRP advertisements received by
            this virtual router."
        ::= { vrrpIpv6RouterStatsEntry 3 }

    vrrpIpv6StatsAdvIntervalErrors OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
```

```
        STATUS       current
```

```
        DESCRIPTION
            "The total number of IPv6 VRRP advertisement packets
            received for which the advertisement interval is different
            than the one configured for the local virtual router."
        ::= { vrrpIpv6RouterStatsEntry 4 }

    vrrpIpv6StatsHopLimitErrors OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current


        DESCRIPTION
            "The total number of IPv6 VRRP packets received by the
            virtual router with IPv6 hop limit not equal to 255."
        ::= { vrrpIpv6RouterStatsEntry 5 }

    vrrpIpv6StatsPriZeroPktsRcvd OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The total number of IPv6 VRRP packets received by the
            virtual router with a priority of '0'."
        ::= { vrrpIpv6RouterStatsEntry 6 }

    vrrpIpv6StatsPriZeroPktsSent OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The total number of IPv6 VRRP packets sent by the virtual
            router with a priority of '0'."
        ::= { vrrpIpv6RouterStatsEntry 7 }

    vrrpIpv6StatsInvalidTypePktsRcvd OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The number of VRRP packets received by the virtual router
            with an invalid value in the 'type' field."
        ::= { vrrpIpv6RouterStatsEntry 8 }

    vrrpIpv6StatsAddressListErrors OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
```

DESCRIPTION

```
            "The total number of packets received for which the address
            list does not match the locally configured list for the
            virtual router."
        ::= { vrrpIpv6RouterStatsEntry 9 }

    vrrpIpv6StatsPacketLengthErrors OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The total number of packets received with a packet length
            less than the length of the VRRP header."
        ::= { vrrpIpv6RouterStatsEntry 10 }

--**********************************************************************
-- Trap Definitions
--**********************************************************************

    vrrpIpv6Notifications   OBJECT IDENTIFIER ::= { vrrpIpv6MIB 0 }

    vrrpIpv6TrapNewMasterReason OBJECT-TYPE
        SYNTAX        INTEGER {
            priority (0),
            preempted (1),
            masterNoResponse (2)
        }
        MAX-ACCESS   accessible-for-notify
        STATUS       current
        DESCRIPTION
            "This indicates the reason for NewMaster trap.
            Used by vrrpIpv6TrapNewMaster trap."
        ::= { vrrpIpv6Operations 6 }

    vrrpIpv6TrapProtoErrReason OBJECT-TYPE
        SYNTAX        INTEGER {
            hopLimitError (0),
            versionError (1),
            checksumError (2),
            vridError(3)
        }
        MAX-ACCESS   accessible-for-notify
        STATUS       current
        DESCRIPTION
            "This indicates the reason for protocol error trap.
            Used by vrrpIpv6TrapProtoError trap."
        ::= { vrrpIpv6Operations 7 }

    vrrpIpv6TrapNewMaster NOTIFICATION-TYPE
```

```
        OBJECTS      { vrrpIpv6OperMasterIpAddr,
```

```
                    vrrpIpv6TrapNewMasterReason
                  }
      STATUS      current
      DESCRIPTION
          "The newMaster trap indicates that the sending agent
          has transitioned to 'Master' state. The
          vrrpIpv6TrapNewMasterReason  indicates the reason due to
          which the sending agent transitioned to æmasterÆ state.ö
          ::= { vrrpIpv6Notifications 1 }

   vrrpIpv6TrapProtoError NOTIFICATION-TYPE
      OBJECTS     { vrrpIpv6TrapProtoErrReason
                  }
      STATUS      current
      DESCRIPTION
          "The error trap indicates that the sending agent has
          encountered the protocol error indicated by ErrorReason."
      ::= { vrrpIpv6Notifications 2 }



   --*****************************************************************
   -- Conformance Information
   --*****************************************************************

   vrrpIpv6MIBCompliances  OBJECT IDENTIFIER
      ::= { vrrpIpv6Conformance 1 }
   vrrpIpv6MIBGroups       OBJECT IDENTIFIER
      ::= { vrrpIpv6Conformance 2 }

   --.................................................................
   -- Compliance Statements
   --.................................................................

   vrrpIpv6MIBCompliance MODULE-COMPLIANCE
      STATUS current
      DESCRIPTION
         "The core compliance statement for all VRRP IPv6
          implementations."
      MODULE -- this module
      MANDATORY-GROUPS  {
          vrrpIpv6OperGroup,
          vrrpIpv6StatsGroup
      }
      OBJECT        vrrpIpv6OperPriority
      WRITE-SYNTAX  Integer32 (1..255)
      DESCRIPTION  "SETable values are from 1 to 255."

      ::= { vrrpIpv6MIBCompliances 1 }
```

```
   --..............................................................
   -- Conformance Groups
   --..............................................................

    vrrpIpv6OperGroup  OBJECT-GROUP
        OBJECTS  {
            vrrpIpv6NodeVersion,
            vrrpIpv6NotificationCntl,
            vrrpIpv6OperVirtualMacAddr,
            vrrpIpv6OperState,
            vrrpIpv6OperAdminState,
            vrrpIpv6OperPriority,
            vrrpIpv6OperIpAddrType,
            vrrpIpv6OperMasterIpAddr,
            vrrpIpv6OperAdvInterval,
            vrrpIpv6OperPreemptMode,
            vrrpIpv6OperVirtualRouterUpTime,
            vrrpIpv6OperProtocol,
            vrrpIpv6OperRowStatus
            }
        STATUS current
        DESCRIPTION
           "Conformance group for VRRP operations."
        ::= { vrrpIpv6MIBGroups 1 }

    vrrpIpv6StatsGroup  OBJECT-GROUP
        OBJECTS  {
            vrrpIpv6RouterChecksumErrors,
            vrrpIpv6RouterVersionErrors,
            vrrpIpv6RouterVrIdErrors,
            vrrpIpv6StatsBecomeMaster,
            vrrpIpv6StatsAsMasterUpTime,
            vrrpIpv6StatsAdvRcvd,
            vrrpIpv6StatsAdvIntervalErrors,
            vrrpIpv6StatsHopLimitErrors,
            vrrpIpv6StatsPriZeroPktsRcvd,
            vrrpIpv6StatsPriZeroPktsSent,
            vrrpIpv6StatsInvalidTypePktsRcvd,
            vrrpIpv6StatsAddressListErrors,
            vrrpIpv6StatsPacketLengthErrors
            }
        STATUS current
        DESCRIPTION
           "Conformance group for VRRP statistics."
        ::= { vrrpIpv6MIBGroups 2 }
```

```
    vrrpIpv6TrapGroup  OBJECT-GROUP
```

```
        OBJECTS  {
            vrrpIpv6TrapNewMasterReason,
            vrrpIpv6TrapProtoErrReason
            }
        STATUS current
        DESCRIPTION
           "Conformance group for objects contained in VRRP
            notifications."
        ::= { vrrpIpv6MIBGroups 3 }

    vrrpIpv6NotificationGroup NOTIFICATION-GROUP
        NOTIFICATIONS {
            vrrpIpv6TrapNewMaster,
            vrrpIpv6TrapProtoError
            }
        STATUS current
        DESCRIPTION
           "The VRRP MIB Notification Group."
        ::= { vrrpIpv6MIBGroups 4 }

   END
```

4. Security considerations

There are a number of management objects defined in this MIB that have a MAX-ACCESS clause of read-write or read-create. Such objects may be considered sensitive or vulnerable to security attacks in some networking environments. The support for SET operations in a non-secure environment without proper protection can have a negative effect on VRRP router operations.

A number of objects in the vrrpIpv6OperTable possess the read-create attribute. Manipulation of these objects is capable of affecting the operation of a virtual router.

Specific examples of this include, but are not limited to:

o The vrrpIpv6OperAdminState object which could be used to disable a virtual router.

SNMPv1 by itself is not a secure environment. 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.

It is recommended that the implementers consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model RFC 2574 [RFC2574] and the View-based Access Control Model RFC 2575 [RFC2575] is recommended.

It is then a customer/user responsibility to ensure that the SNMP entity giving access to an instance of this MIB, 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.

5. Normative References

[19] Robert Hinden, "Virtual Router Redundancy Protocol for IPv6", (draft-ietf-vrrp-ipv6-spec-04.txt), May 2003.

6. Informative References

[1] Harrington, D., Presuhn, R. and B. Wijnen, "An Architecture for Describing SNMP Management Frameworks", RFC 2571, April 1999.

[2] Rose, M. and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", STD 16, RFC 1155, May 1990.

[3] Rose, M. and K. McCloghrie, "Concise MIB Definitions", STD 16, RFC 1212, March 1991.

[4] Rose, M., "A Convention for Defining Traps for use with the SNMP", RFC 1215, March 1991.

[5] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.

[6] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.

[7] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.

[8] Case, J., Fedor, M., Schoffstall, M. and J. Davin, "Simple Network Management Protocol", STD 15, RFC 1157, May 1990.

[9] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Introduction to Community-based SNMPv2", RFC 1901, January 1996.

[10] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1906, January 1996.

[11] Case, J., Harrington D., Presuhn R. and B. Wijnen, "Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)", RFC 2572, April 1999.

[12] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", RFC 2574, April 1999.

[13] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1905, January 1996.

[14] Levi, D., Meyer, P. and B. Stewart, "SNMPv3 Applications", RFC 2573, April 1999.

[15] Wijnen, B., Presuhn, R. and K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)", RFC 2575, April 1999

[16] Case, J., Mundy, R., Partain, D. and B. Stewart, "Introduction to Version 3 of the Internet-standard Network Management Framework", RFC 2570, April 1999

[17] Knight, S., Weaver, D., Whipple, D., Hinden, R., Mitzel, D., Hunt, P., Higginson, P., Shand, M. and Lindem, A., "Virtual Router Redundancy Protocol", RFC 2338, November 1997.

[18] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB using SMIv2", RFC 2233, November 1997.

[20] M. Daniele, "Textual Conventions for Internet Network Addresses", RFC 3291, May 2002.

[21] Jewell & Chuang, "Definitions of Managed Objects for the Virtual Router Redundancy Protocol", RFC 2787, March 2000.

7. Acknowledgements

This specification is based on RFC 2787 [21]. The authors of RFC2787 are Brian Jewell and David Chuang

8. IANA Considerations

VRRP IPv6 MIB requires an OID assigned under mib-2 and this should be entered into section 3.

Author's Addresses

Kalyan Tata
Nokia Inc.
313 Fair Child Dr.
Mountain View, California 94087
US

Phone: +1 408-896 6493
Email: kalyan.tata@nokia.com

Kripakaran karlekar
Nokia Inc.
313 Fair Child Dr.
Mountain View, California 94087
US
Phone: +1 -
Email: kripakaran.karlekar@nokia.com

Brian R. Jewell
Copper Mountain Networks, Inc.
2470 Embarcadero Way
Palo Alto, California 94303
US

Phone: +1 650 687 3367
Email: bjewell@coppermountain.com

9. Changes from RFC 2787

- General rewrite to change MIB definition to accommodate protocol changes in virtual router functionality from IPv4 to IPv6.
- Change all definitions of IPaddress to InetAddress and add InetAddressType as defined in RFC 3291 [20] in the following :
  - o vrrpIPv6OperTable defined with vrrpIpv6OperIpAddrType and vrrpIPv6OperMasterIpAddr.

- There is no definition to support multiple IP addresses per virtual router in VRRP-IPv6 and hence the following have been changed
  - o No corresponding definition to vrrpAssoAddrTable.
  - o No definition corresponding to vrrpPrimaryIPAddress
  - o No definition corresponding to vrrpIPAddrCount
- Added vrrpIpv6StatsAsMasterUpTime to vrrpIPv6Statisctcs group.
- There is no authentication mechanism defined in VRRP for IPv6 protocol. So all authentication related configuration, statistics and notifications are removed.
- Changed vrrpIpv6StatsIpTtlErrors to vrrpIpv6HopLimitErrors
- Added new trap to indicate various errors encountered by the VRRP protocol.
- Added ErrorReason to indicate the reason for vrrpIpv6ErrorTrap.
- Updated mib description with copyright information.
- Modify conformance statement to reflect changes in vrrpIpv6OperGroup and vrrpIPv6StatsGroup.