Definitions of Managed Objects for the 
Virtual Router Redundancy Protocol

<draft-ietf-vrrp-mib-08.txt>

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This internet draft expires on December 4, 1999.

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 (VRRP) [17].

This memo specifies a MIB module in a manner that is compliant with both the SNMPv2 SMI [19], and semantically identical to the SNMPv1 definitions [2].
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 RFC 1155 [2], RFC 1212 [3] and RFC 1215 [4]. The second version, called SMIv2, is described in RFC 2578 [5], RFC 2579 [6] and RFC 2580 [7].

- Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and described in 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 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

This memo identifies the set of objects for configuring, monitoring, and controlling the Virtual Router Redundancy Protocol (VRRP), as defined in RFC 2338 [17].

VRRP specifies an election protocol that will allow one or more associated IP addresses to be assumed by another router in the event of a failure of the IP address(es) owner. Thus, IP traffic from a host using a failed router as a default gateway is transparently forwarded by the VRRP router that has assumed control. VRRP provides redundancy in routed networks without requiring configuration of dynamic routing or router discovery protocols on every end-host.

2.1. VRRP MIB Structure

The VRRP MIB contains three conformance groups:

- vrrpOperations Group: Objects related to VRRP router’s configuration and control.

- vrrpStatistics Group: Objects containing information useful in monitoring the operation of VRRP routers.

- vrrpNotifications Group: Consists of objects and definitions for use in SNMP traps sent by VRRP routers.

Tables in the MIB include the following:

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

2. The vrrpAssoIpAddrTable, which contains the addresses of the virtual router(s) that a given VRRP router is backing up.

3. The vrrpRouterStatsTable which contains the operating statistics for a VRRP router.

2.2. Virtual Router Redundancy Protocol

This MIB is based on the following characteristics of VRRP as defined in the VRRP specification [17].

- A "VRRP router" is one that is configured to run the VRRP protocol in conjunction with one or more other VRRP routers attached to a LAN.
- A VRRP router can be running one or more instances of a virtual router.

- A "virtual router" is an abstraction which consists of two or more physical routers associated by a Virtual Router Identifier (VRID).

- An instance of a virtual router (on a physical VRRP router), can be uniquely identified by a combination of the 'IF Index' [18] and "Virtual Router Identifier" (VRID).

- For each VRID there is a set of one or more "associated IP addresses" that are backed-up by the virtual router.

2.3. VRRP MIB Table Design

The tables in the VRRP MIB are structured with the assumption that a VRRP network management application would likely be designed to display information or provide configuration about a VRRP router on a "per-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.

2.3.1. Relation to Interface Group (RFC 2233) [23].

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

2.4. VRRP Scenarios

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

KEY:

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

if     = vrrpOperIfIndex
VrId   = vrrpOperVrId
State  = vrrpOperState
Prior  = vrrpOperPriority
AddrCnt = vrrpOperIpAddrCount
2.4.1. VRRP Scenario #1

The following figure shows a simple network with two VRRP routers configured with two virtual routers. This sample topology is taken from the VRRP specification [17]. Addresses in ‘()’ indicate the IP address of the default gateway for a given host, H1 - H4. In the diagram, "Interface" is used in the context defined in MIB-II [4].

```
VRID=1     VRID=2
+-----+      +-----+
| MR1 |      | MR2 |
| &   |      | &   |
| BR2 |      | BR1 |
+-----+      +-----+

IP A ----------> *            *<---------- IP B
Interface=I1 | | |            | | | Interface=I2
```

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

```
vrrpOperTable
-------------
<table>
<thead>
<tr>
<th>if</th>
<th>VrId</th>
<th>State</th>
<th>Prior</th>
<th>AddrCnt</th>
<th>IpAddr</th>
<th>...</th>
<th>RowStat</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1</td>
<td>01</td>
<td>M</td>
<td>255</td>
<td>1</td>
<td>A</td>
<td></td>
<td>active</td>
</tr>
<tr>
<td>I1</td>
<td>02</td>
<td>B</td>
<td>1-254</td>
<td>1</td>
<td>B</td>
<td></td>
<td>active</td>
</tr>
</tbody>
</table>
```
### vrrpAssoIpAddrTable

<table>
<thead>
<tr>
<th>if</th>
<th>VrId</th>
<th>IP</th>
<th>RowStat</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1</td>
<td>01</td>
<td>A</td>
<td>active</td>
</tr>
<tr>
<td>I1</td>
<td>02</td>
<td>B</td>
<td>active</td>
</tr>
</tbody>
</table>

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

### vrrpOperTable

<table>
<thead>
<tr>
<th>if</th>
<th>VrId</th>
<th>State</th>
<th>Prior</th>
<th>AddrCnt</th>
<th>IpAddr</th>
<th>...</th>
<th>RowStat</th>
</tr>
</thead>
<tbody>
<tr>
<td>I2</td>
<td>01</td>
<td>B</td>
<td>1-254</td>
<td>1</td>
<td>A</td>
<td></td>
<td>active</td>
</tr>
<tr>
<td>I2</td>
<td>02</td>
<td>M</td>
<td>255</td>
<td>1</td>
<td>B</td>
<td></td>
<td>active</td>
</tr>
</tbody>
</table>

### vrrpAssoIpAddrTable

<table>
<thead>
<tr>
<th>if</th>
<th>VrId</th>
<th>IP</th>
<th>RowStat</th>
</tr>
</thead>
<tbody>
<tr>
<td>I2</td>
<td>01</td>
<td>A</td>
<td>active</td>
</tr>
<tr>
<td>I2</td>
<td>02</td>
<td>B</td>
<td>active</td>
</tr>
</tbody>
</table>
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 vrrpOperTable, a "priority" of 255 indicates that the respective router owns the IP address, e.g., this IP address is native to the router (i.e., "the IP Address Owner" [17]).

2.4.2. VRRP Scenario #2

The following figure shows a simple network with two virtual routers. Here, a single interface has been configured with two IP addresses. Again, addresses in () indicate the IP address of the default gateway for a given host, H1 - H2.

```
+-----+      +-----+  
| MR1 |      | MR2 |  
| &   |      | &   |  
| BR2 |      | BR1 |  
+-----+      +-----+  

IP A ---------->*            *<---------- IP B
IP C
Interface=I1

------------------+------------+-----+--------+
          ^        ^
          (IP A)    (IP B)
          |        |
          H1      H2
          +-----+  +-----+
```

```
```

```
```
MIB Tables For VRRP Router "IP A":

### vrrpOperTable

<table>
<thead>
<tr>
<th>if</th>
<th>VrId</th>
<th>State</th>
<th>Prior</th>
<th>AddrCnt</th>
<th>IpAddr</th>
<th>...</th>
<th>RowStat</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1</td>
<td>01</td>
<td>M</td>
<td>255</td>
<td>2</td>
<td>A</td>
<td></td>
<td>active</td>
</tr>
<tr>
<td>I1</td>
<td>02</td>
<td>B</td>
<td>1-254</td>
<td>1</td>
<td>B</td>
<td></td>
<td>active</td>
</tr>
</tbody>
</table>

### vrrpAssoIpAddrTable

<table>
<thead>
<tr>
<th>if</th>
<th>VrId</th>
<th>IP</th>
<th>RowStat</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1</td>
<td>01</td>
<td>A</td>
<td>active</td>
</tr>
<tr>
<td>I1</td>
<td>01</td>
<td>C</td>
<td>active</td>
</tr>
<tr>
<td>I1</td>
<td>02</td>
<td>B</td>
<td>active</td>
</tr>
</tbody>
</table>
### MIB Tables For VRRP Router "IP B":

#### vrrpOperTable

<table>
<thead>
<tr>
<th>if</th>
<th>VrId</th>
<th>State</th>
<th>Prior</th>
<th>AddrCnt</th>
<th>IpAddr</th>
<th>...</th>
<th>RowStat</th>
</tr>
</thead>
<tbody>
<tr>
<td>I2</td>
<td>01</td>
<td>B</td>
<td>1-254</td>
<td>2</td>
<td>A</td>
<td></td>
<td>active</td>
</tr>
<tr>
<td>I2</td>
<td>02</td>
<td>M</td>
<td>255</td>
<td>1</td>
<td>B</td>
<td></td>
<td>active</td>
</tr>
</tbody>
</table>

#### vrrpAssoIpAddrTable

<table>
<thead>
<tr>
<th>if</th>
<th>VrId</th>
<th>IP</th>
<th>RowStat</th>
</tr>
</thead>
<tbody>
<tr>
<td>I2</td>
<td>01</td>
<td>A</td>
<td>active</td>
</tr>
<tr>
<td>I2</td>
<td>01</td>
<td>C</td>
<td>active</td>
</tr>
<tr>
<td>I2</td>
<td>02</td>
<td>B</td>
<td>active</td>
</tr>
</tbody>
</table>
3. Definitions

VRRP-MIB DEFINITIONS ::= BEGIN

IMPORTS
MODULE-IDENTITY, OBJECT-TYPE,
NOTIFICATION-TYPE, Counter32,
Integer32, IpAddress, mib-2 FROM SNMPv2-SMI -- RFC 1902[19]
TEXTUAL-CONVENTION, RowStatus,
MacAddress, TruthValue, TimeStamp FROM SNMPv2-TC -- RFC 1903[20]
MODULE-COMPLIANCE, OBJECT-GROUP FROM SNMPv2-CONF -- RFC 1904[21]
ifIndex FROM IF-MIB; -- RFC 2233

vrrp MODULE-IDENTITY
LAST-UPDATED "9906240900Z"
ORGANIZATION "IETF VRRP Working Group"
CONTACT-INFO
"Brian R. Jewell
Postal: 3Com Corporation
5400 BayFront Plaza
Santa Clara, California 95052
Tel:    +1 408 326 6173
E-Mail: bjewell@3com.com"

DESCRIPTION
"The MIB module to describe the VRRP Version 2 Protocol."
 ::= { mib-2 68 }

vrrpMIB OBJECT IDENTIFIER ::= { vrrp 1 }

-- *******************************************************************
--  Textual Conventions
-- *******************************************************************

VrId ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
"A number which, along with an interface index (ifIndex),
serves to uniquely identify a virtual router on a given VRRP
router. A set of one or more associated addresses is assigned
to a VRID."
SYNTAX Integer32 (1..255)
vrrpNodeVersion  OBJECT-TYPE
SYNTAX       Integer32
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION
"This value identifies the particular version of the VRRP
supported by this node."
REFERENCE
"Virtual Router Redundancy Protocol, RFC 2338"
DEFVAL { 2 }
::= { vrrpOperations 1 }

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

...
vrrpOperTable OBJECT-TYPE
SYNTAX       SEQUENCE OF VrrpOperEntry
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
  'vrrpOperEntry' items."
 ::= { vrrpOperations 3 }

vrrpOperEntry OBJECT-TYPE
SYNTAX       VrrpOperEntry
MAX-ACCESS   not-accessible
STATUS       current
DESCRIPTION
  "An entry in the vrrpOperTable containing the operational
  characteristics of a virtual router. On a VRRP router,
  a given virtual router is identified by a combination
  of the IF index and VRID.

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

INDEX    { ifIndex, vrrpOperVrId }
 ::= { vrrpOperTable 1 }
VrrpOperEntry ::= 
  SEQUENCE { 
    vrrpOperVrId 
     VrId, 
    vrrpOperVirtualMacAddr 
     MacAddress, 
    vrrpOperState 
     INTEGER, 
    vrrpOperAdminState 
     INTEGER, 
    vrrpOperPriority 
     Integer32, 
    vrrpOperIpAddrCount 
     Integer32, 
    vrrpOperMasterIpAddr 
     IpAddress, 
    vrrpOperPrimaryIpAddr 
     IpAddress, 
    vrrpOperAuthType 
     INTEGER, 
    vrrpOperAuthKey 
     OCTET STRING, 
    vrrpOperAdvertisementInterval 
     Integer32, 
    vrrpOperPreemptMode 
     TruthValue, 
    vrrpOperVirtualRouterUpTime 
     TimeStamp, 
    vrrpOperProtocol 
     INTEGER, 
    vrrpOperRowStatus 
     RowStatus 
  } 

vrrpOperVrId OBJECT-TYPE 
SYNTAX VrId 
MAX-ACCESS  not-accessible 
STATUS       current 
DESCRIPTION 
  "This object contains the Virtual Router Identifier (VRID)."
 ::= { vrrpOperEntry 1 }
vrrpOperVirtualMacAddr 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 ‘vrrpOperVrId’ object, it is defined so that it is easily obtainable by a management application and can be included in VRRP-related SNMP traps."
::= { vrrpOperEntry 2 }

vrrpOperState 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 IP addresses that are associated with this router.

Setting the ‘vrrpOperAdminState’ object (below) initiates transitions in the value of this object."
::= { vrrpOperEntry 3 }
vrrpOperAdminState 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'; 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 'vrrpOperAdminState' object must be set to 'down' prior
  to modifying the other read-create objects in the conceptual
  row. The value of the 'vrrpOperRowStatus' 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 }
::= { vrrpOperEntry 4 }
vrrpOperPriority 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 }
::= { vrrpOperEntry 5 }

vrrpOperIpAddrCount OBJECT-TYPE
SYNTAX Integer32 (0..255)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of IP addresses that are associated with this virtual router. This number is equal to the number of rows in the vrrpAssoIpAddrTable that correspond to a given IF index/VRID pair."
::= { vrrpOperEntry 6 }

vrrpOperMasterIpAddr OBJECT-TYPE
SYNTAX IpAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The master router's real (primary) IP address. This is the IP address listed as the source in VRRP advertisement last received by this virtual router."
::= { vrrpOperEntry 7 }
vrrpOperPrimaryIpAddr OBJECT-TYPE
SYNTAX       IpAddress
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION
"In the case where there is more than one IP address for a given 'ifIndex', this object is used to specify the IP address that will become the 'vrrpOperMasterIpAddr', should the virtual router transition from backup to master. If this object is set to 0.0.0.0, the IP address which is numerically lowest will be selected."
DEFVAL       { '00000000'H } -- 0.0.0.0
::= { vrrpOperEntry 8 }

vrrpOperAuthType OBJECT-TYPE
SYNTAX       INTEGER {
    noAuthentication(1),       -- VRRP protocol exchanges are not authenticated.
    simpleTextPassword(2),     -- Exchanges are authenticated by a clear text password.
    ipAuthenticationHeader(3)  -- Exchanges are authenticated using the IP authentication header.
}
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION
"Authentication type used for VRRP protocol exchanges between virtual routers. This value of this object is the same for a given ifIndex."
DEFVAL       { noAuthentication } 
::= { vrrpOperEntry 9 }

vrrpOperAuthKey OBJECT-TYPE
SYNTAX       OCTET STRING (SIZE (0..16))
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION
"The Authentication Key. This object is set according to the value of the 'vrrpOperAuthType' object ('simpleTextPassword' or 'ipAuthenticationHeader'). If the length of the value is less than 16 octets, the agent will left adjust and zero fill to 16 octets. The value of this object is the same for a given ifIndex.

When read, vrrpOperAuthKey always returns an Octet String of length zero."
::= { vrrpOperEntry 10 }
vrrpOperAdvertisementInterval OBJECT-TYPE
SYNTAX Integer32 (1..255)
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 }
 ::= { vrrpOperEntry 11 }

vrrpPreemptMode 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 }
 ::= { vrrpOperEntry 12 }

vrrpOperVirtualRouterUpTime 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 'vrrpOperState') transitioned
out of 'initialized'."
 ::= { vrrpOperEntry 13 }

vrrpOperProtocol 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."
DEFVAL { ip }
 ::= { vrrpOperEntry 14 }
vrrpOperRowStatus 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. In the text
that follows, the characteristics (i.e., whether readable,
settable or both) of each value are indicated in parenthesis.

The state that this object transitions to when it is set
is based on a determination of whether the read-create
objects in the row have been correctly initialized for
virtual router operation. A row in which not all of the
objects are correctly set is considered 'incomplete'.

The rowstatus of a currently active row in the vrrpOperTable
is also constrained by the operational state of the
 corresponding virtual router. Prior to setting this object
from 'active' to a different value, the 'vrrpOperAdminState'
object must be set to 'down', and the 'vrrpOperState' object
 be transitioned to 'initialize'.

The rowstatus column has six defined values:

- 'active' (read/set). When this value is read, it
  indicates that all the read-create objects (in the row)
  required for virtual router operation have been
  correctly initialized such that the respective virtual
  router can be made operational by setting the
  'vrrpOperAdminState' to 'up'. When set to 'active',
  no other objects in the conceptual row, with the
  exception of 'vrrpOperAdminState', can be modified.

  When set, the agent will transition the value of
  rowstatus to 'notReady' if the row is incomplete.

- 'notInService' (read/set), which when set, allows the
  objects in the row to be modified by a management station,
  thus changing the operational characteristics of the
  corresponding virtual router.

- 'notReady' (read). The agent sets the object to this
  state to indicate that the conceptual row exists,
  but is lacking initialization of one or more objects
  required for virtual router operation.

- 'createAndGo' (set), which is set by a management
station wishing to create a new instance of a virtual router and to have its status automatically set to 'active', making it available for use by a virtual router. Upon receiving a request to set the rowstatus to this value, the agent transitions the rowstatus to 'active' if the other settable objects in the row have been correctly initialized. If the row is incomplete, the agent transitions the state to 'notReady'.

- 'createAndWait' (set), which is set by a management station wishing to create a new instance of a virtual router but not make it available for use. When this value is set, rowstatus transitions to 'notInService' if the row has been correctly initialized; if the row is incomplete, rowstatus will become 'notReady'.

- 'destroy' (set), which deletes the conceptual row, and hence, the corresponding instance of a virtual router.

::= { vrrpOperEntry 15 }

-- *******************************************************************
-- VRRP Associated IP Address Table
-- *******************************************************************

vrrpAssoIpAddrTable OBJECT-TYPE
SYNTAX       SEQUENCE OF VrrpAssoIpAddrEntry
MAX-ACCESS   not-accessible
STATUS       current
DESCRIPTION   "The table of addresses associated with this virtual router."
::= { vrrpOperations 4 }
vrrpAssoIpAddrEntry OBJECT-TYPE
SYNTAX VrrpAssoIpAddrEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"An entry in the table contains an IP address that is
associated with a virtual router. The number of rows for
a given ifIndex and VrId will equal the number of IP
addresses associated (e.g., backed up) by the virtual
router (equivalent to 'vrrpOperIpAddrCount').

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

INDEX { ifIndex, vrrpOperVrId, vrrpAssoIpAddr }
::= { vrrpAssoIpAddrTable 1 }

VrrpAssoIpAddrEntry ::= SEQUENCE {
vrrpAssoIpAddr IpAddress,
vrrpAssoIpAddrRowStatus RowStatus
}

vrrpAssoIpAddr OBJECT-TYPE
SYNTAX IpAddress
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"The assigned IP addresses that a virtual router is
responsible for backing up."
::= { vrrpAssoIpAddrEntry 1 }

vrrpAssoIpAddrRowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The row status variable, used according to installation
and removal conventions for conceptual rows. Setting this
object to ‘active’ or ‘createAndGo’ results in the addition
of an associated address for a virtual router. Destroying
the entry or setting it to ‘notInService’ removes the
associated address from the virtual router. The use of
other values is implementation-dependent."
::= { vrrpAssoIpAddrEntry 2 }
-- *******************************************************************
--  VRRP Router Statistics
-- *******************************************************************

vrrpRouterChecksumErrors OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "The total number of VRRP packets received with an invalid
VRRP checksum value."
::= { vrrpStatistics 1 }

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

vrrpRouterVrIdErrors OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "The total number of VRRP packets received with an invalid
VRID for this virtual router."
::= { vrrpStatistics 3 }

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

vrrpRouterStatsTable OBJECT-TYPE
SYNTAX SEQUENCE OF VrrpRouterStatsEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "Table of virtual router statistics."
::= { vrrpStatistics 4 }
vrrpRouterStatsEntry 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    { vrpOperEntry }
 ::= { vrrpRouterStatsTable 1 }

VrrpRouterStatsEntry ::= SEQUENCE {
  vrrpStatsBecomeMaster
    Counter32,
  vrrpStatsAdvertiseRcvd
    Counter32,
  vrrpStatsAdvertiseIntervalErrors
    Counter32,
  vrrpStatsAuthFailures
    Counter32,
  vrrpStatsPriorityZeroPktsRcvd
    Counter32,
  vrrpStatsPriorityZeroPktsSent
    Counter32,
  vrrpStatsInvalidTypePktsRcvd
    Counter32,
  vrrpStatsInvalidTypePktsSent
    Counter32,
  vrrpStatsAddressListErrors
    Counter32,
  vrrpStatsInvalidAuthType
    Counter32,
  vrrpStatsAuthTypeMismatch
    Counter32,
  vrrpStatsPacketLengthErrors
    Counter32
}

vrrpStatsBecomeMaster 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."
 ::= { vrrpRouterStatsEntry 1 }
vrrpStatsAdvertiseRcvd OBJECT-TYPE
SYNTAX       Counter32
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION
  "The total number of VRRP advertisements received by this virtual router."
 ::= { vrrpRouterStatsEntry 2 }

vrrpStatsAdvertiseIntervalErrors OBJECT-TYPE
SYNTAX       Counter32
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION
  "The total number of VRRP advertisement packets received for which the
   advertisement interval is different than the one configured for the local virtual router."
 ::= { vrrpRouterStatsEntry 3 }

vrrpStatsAuthFailures OBJECT-TYPE
SYNTAX       Counter32
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION
  "The total number of VRRP packets received that do not pass the authentication check."
 ::= { vrrpRouterStatsEntry 4 }

vrrpStatsIpTtlErrors OBJECT-TYPE
SYNTAX       Counter32
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION
  "The total number of VRRP packets received by the virtual router with IP TTL (Time-To-Live) not equal to 255."
 ::= { vrrpRouterStatsEntry 5 }

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

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

vrrpStatsInvalidTypePktsRcvd 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."
::= { vrrpRouterStatsEntry 8 }

vrrpStatsAddressListErrors 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."
::= { vrrpRouterStatsEntry 9 }

vrrpStatsInvalidAuthType OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The total number of packets received with an unknown
authentication type."
::= { vrrpRouterStatsEntry 10 }

vrrpStatsAuthTypeMismatch OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The total number of packets received with 'Auth Type' not
equal to the locally configured authentication method
('vrrpOperAuthType')."
::= { vrrpRouterStatsEntry 11 }
vrrpStatsPacketLengthErrors 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."
::= { vrrpRouterStatsEntry 12 }

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

vrrpTraps       OBJECT IDENTIFIER ::= { vrrpNotifications 1 }

vrrpTrapPacketSrc OBJECT-TYPE
SYNTAX       IpAddress
MAX-ACCESS   accessible-for-notify
STATUS       current
DESCRIPTION  "The IP address of an inbound VRRP packet. Used by
vrrpTrapAuthFailure trap."
::= { vrrpTraps 1 }

vrrpTrapAuthErrorType OBJECT-TYPE
SYNTAX        INTEGER {
  invalidAuthType (1),
  authTypeMismatch (2),
  authFailure (3)
}
MAX-ACCESS   accessible-for-notify
STATUS       current
DESCRIPTION  "Potential types of configuration conflicts.
Used by vrrpAuthFailure trap."
::= { vrrpTraps 2 }

vrrpTrapNewMaster NOTIFICATION-TYPE
OBJECTS      { ifIndex,
    vrrpOperVrId
  }
STATUS       current
DESCRIPTION  "The newMaster trap indicates that the sending agent
has transitioned from 'Backup' state to 'Master' state."
::= { vrrpTraps 3 }
vrrpTrapAuthFailure NOTIFICATION-TYPE
   OBJECTS { ifIndex,
              vrrpOperVrId,
              vrrpTrapPacketSrc,
              vrrpTrapAuthErrorType
   }
   STATUS current
   DESCRIPTION
      "A vrrpAuthFailure trap signifies that a packet has
      been received from a router whose authentication key
      or authentication type conflicts with this router’s
      authentication key or authentication type. Implementation
      of this trap is optional."
   ::= { vrrpTraps 4 }

-- Conformance Information
---------------------------------------------------------------

vrrpMIBCompliances OBJECT IDENTIFIER ::= { vrrpConformance 1 }
vrrpMIBGroups OBJECT IDENTIFIER ::= { vrrpConformance 2 }

-- Compliance Statements
---------------------------------------------------------------

vrrpMIBCompliance MODULE-COMPLIANCE
   STATUS current
   DESCRIPTION
      "The core compliance statement for all VRRP implementations."
   MODULE -- this module
      MANDATORY-GROUPS { vrrpOperGroup , vrrpStatsGroup }
   ::= { vrrpMIBCompliances 1 }
vrrpOperGroup  OBJECT-GROUP
   OBJECTS {
      vrrpNodeVersion,
      vrrpTrapCntl,
      vrrpOperVrId,
      vrrpOperVirtualMacAddr,
      vrrpOperState,
      vrrpOperAdminState,
      vrrpOperPriority,
      vrrpOperIpAddrCount,
      vrrpOperMasterIpAddr,
      vrrpOperPrimaryIpAddr,
      vrrpOperAuthType,
      vrrpOperAuthKey,
      vrrpOperAdvertisementInterval,
      vrrpOperPreemptMode,
      vrrpOperVirtualRouterUpTime,
      vrrpOperProtocol,
      vrrpOperRowStatus,
      vrrpAssoIpAddr,
      vrrpAssoIpAddrRowStatus
   }

   STATUS current
   DESCRIPTION
      "Conformance group for VRRP operations."
   ::= { vrrpMIBGroups 1 }


vrrpStatsGroup OBJECT-GROUP
  OBJECTS {
    vrrpRouterChecksumErrors,
    vrrpRouterVersionErrors,
    vrrpRouterVrIdErrors,
    vrrpStatsBecomeMaster,
    vrrpStatsAdvertiseRcvd,
    vrrpStatsAdvertiseIntervalErrors,
    vrrpStatsAuthFailures,
    vrrpStatsIpTtlErrors,
    vrrpStatsPriorityZeroPktsRcvd,
    vrrpStatsPriorityZeroPktsSent,
    vrrpStatsInvalidTypePktsRcvd,
    vrrpStatsAddressListErrors,
    vrrpStatsInvalidAuthType,
    vrrpStatsAuthTypeMismatch,
    vrrpStatsPacketLengthErrors
  }
  STATUS current
  DESCRIPTION
    "Conformance group for VRRP statistics."
  ::= { vrrpMIBGroups 2 }
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 vrrpOperTable 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:

- The vrrpOperAdminState object which could be used to disable a virtual router.
- The vrrpOperPrimaryIpAddr object which, if compromised, could allow assignment of an invalid IP address to a master router.
- The authentication type/key related objects which could potentially render the VRRP security mechanisms ineffective.

Of additional concern is the ability to disable the transmission of traps. This would nullify the capability of a virtual router to provide notification in the event of an authentication failure.

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. Acknowledgements

The authors would like to thank Danny Mitzel, Venkat Prasad, Al Pham, Robert Hinden, Venkat Prasad, Barbera Denny, Fred Baker, Jeff Case, Flavio Fernandes, Acee Lindem and Scott Barvick for their comments and suggestions.

6. References


[22] Postel, J., Reynolds, J., "Instructions to RFC Authors", RFC 2223, October 1997.


7. Authors’ Addresses

Brian R. Jewell
3Com, Inc.
5400 Bayfront Plaza
Santa Clara, CA 95052
US

Phone: +1 326 764 6173
EMail: bjewell@3com.com

David Chuang
3Com, Inc.
5400 Bayfront Plaza
Santa Clara, CA 95052
US

Phone: +1 408 326 6610
EMail: david_chuang@3com.com
This section tracks changes made during revisions of this document. It will be deleted when the document is published as an RFC.

* June 1999: Changes in 8th revision (draft-ietf-vrrp-mib-08.txt):
  - Changed import of "ifIndex" from RFC 1213 to IF-MIB.
  - Added new IANA assigned MIB OID (68).

* June 1999: Changes in 7th revision (draft-ietf-vrrp-mib-07.txt):
  - Added MIB Boilerplate for "The SNMP Management Framework" section.
  - Added "Security Considerations" section.
  - Removed references to "SNMPv2" in header and footers.

* May, 1999: Changes in 6th revision (draft-ietf-vrrp-mib-06.txt):
  - Removed description of how "vrrpOperVirtualMacAddr" is specified as per changes to the VRRP Spec (addition of Token Ring).
  - Added citations for RFC 1903 and RFC 1904 in references.
  - Modified reference to RFC 1573, which has been obsoleted by RFC 2233.

* March, 1999: Changes in 5th revision (draft-ietf-vrrp-mib-05.txt):
  - Objects under "vrrpRouterStatsEntry" were renumbered. OID’s became out-of-sequence due to changes made in the last revision (04).
  - Format of section numbers was changed (‘.’ added) to be consistent with other RFC’s.
  - Changed the type of "vrrpNodeVersion" from INTEGER to Integer32. Also made DEFVAL (for this object) a decimal number.
  - A few spelling and typographical corrections were made.
  - Revisions were made to the format of some of the references to make them more concise and consistent.
  - Reworded the Overview section to attempt to make the description of VRRP more in-line with the VRRP RFC.

  - "vrrpNodeVersion" SYNTAX changed from OCTET STRING to INTEGER.
  - Removed the REFERENCE clause from "vrrpOperVirtualMacAddr", as it
was deemed redundant.
- The range of vrrpOperAddrCount was changed from (1..255) to (0..255). This reflects the fact that a row might be created in the vrrpOperTable *prior* to creating a row in the vrrpAssoIpAddrTable.
- Removed `DEFVAL` clause from "vrrpOperAuthKey" as unnecessary.
- Moved the following objects out of the "vrrpRouterStatsTable" to become "global" VRRP router objects: "vrrpStatsChecksumErrors", "vrrpStatsVersionErrors", "vrrpStatsVrIdErrors". Each of these was renamed "vrrpRouter*", accordingly (since they are no longer members of the vrrpRouterStatsTable. The reason for this change is that, when any of these events occur, attempting to map the packet, with possible corruption of the VRID (or other fields), into a row of the vrrpRouterStatsTable would be risky, at best.
- Removed "vrrpOperVirtualMacAddr" from the "vrrpTrapNewMaster". Since the virtual MAC can be derived from the VR ID, this was extra payload in the trap.
- removed "vrrpTrapConfigErrorType" and "vrrpTrapPacketSrc" from conformance groups, as these are not mandatory objects.
- Added new object, "vrrpOperPrimaryIpAddr" to vrrpOperTable. This object allows for designation of the vrrp IP primary address on an interface with multiple IP addresses, should this virtual router transition from backup to master.
- Changed vrrpRouterStatsEntry to AUGMENT the vrrpOperEntry, since there is a one-to-one correspondence between these two objects and they are indexed with identical parameters.
- Addded new object, vrrpStatsPacketLengthErrors, to the statistics table.
- Changed the description for "vrrpStatsSecurityViolations" to apply to *all* types of authentication failures, as opposed to cleartext auth failures only, which it previously stated.
- Renamed "vrrpTrapConfigErrorType" object to "vrrpTrapAuthErrorType" and added a new enumeration for "authentication failures".
- Renamed "vrrpStatsAuthTypeErrors" to "vrrpStatsAuthTypeMismatch".
- Renamed "vrrpStatsSecurityViolations" to "vrrpStats-AuthenticationFailures"


- Changed name of "vrrpAssoIpAddrStatus" object to "vrrpAssoIpAddrRowStatus" for consistancy with similar object in vrrpOperTable.
- Changed name of "vrrpOperControl" object to "vrrpOperAdminState" and enhanced description clause.
- Renumbered the objects in both the vrrpOperTable and vrrpAssoIpAddrTable. The first objects in the tables started with an OID of '2', when they should have started with '1'.

draft-ietf-vrrp-mib-08.txt [Page 36]
- Changed the name of "vrrpStatsUnknownAuthType" to "vrrpStatsInvalidAuthType" as per suggestion that it was more concise.
- Rewrote the descriptions for the table Entry objects and the rowstatus objects to make the process of row creation and deletion more clear. Each enumeration of the rowstatus object in the "vrrpOperTable" is now clearly explained.
- Changed MAX-ACCESS to "read-create" on the "vrrpOperAuthType" and "vrrpOperAuthKey" objects. This was done with the assumption the the VRRP RFC will change to make these attributes defined on a "per-virtual-router-basis" as was proposed at the August (1998) IETF Meeting.


- A number of changes were made to the textual content of the draft as per feedback received at the last WG meeting. These changes were too numerous to document individually in this section.
- The size of the "vrrpNodeVersion" object has been changed from 2 octets to 1 octet.
- References to obsoleted RFC’s have been replaced by references to later documents. Notably, references to RFC’s 1442, 1443 and 1444 have been replaced by RFC 1902, RFC 1903 and RFC 1904, respectively.
- The description for the "vrrpOperControl" was changed to reflect the fact that a virtual router does not necessarily transition directly from initialize state -> backup state. The description for the "vrrpStatsBecomeMaster" was also changed to more accurately convey this fact.
- The SYNTAX of the "vrrpOperIpAddrCount" was changed to reflect the fact that a virtual router can support only up to 255 backup IP addresses.
- Descriptions for vrrpOperAuthType and vrrpOperAuthKey expanded to indicate the per-interface assignment.
- SYNTAX of vrrpOperPreemptMode object changed from INTEGER to 'truthValue'
- The OIDs for the VRRP traps were fixed; incorrect identifiers (‘vrrpOperations’) had been used in OID assignments.
- The SYNTAX for the ‘vrrpOperPriority’ object was corrected to indicate that this can have a value of ’0’.
- The vrrpOperHMACMD5Key object was deleted. It was combined with the vrrpOperAuthKey object, whose SYNTAX was adjusted accordingly.
- OID for ‘vrrpTraps’ changed to '{ vrrpNotifications 1 }'
- The ‘vrrpStatsPasswdSecurityViolations’ and ‘vrrpStatsHmacSecurityViolations’ objects have been combined into a
single 'vrrpStatsSecurityViolations' object; this was suggested to avoid redundancy.
- As per the last WG meeting, the 'vrrpAssoIpAddrIndex' object has been deleted from the 'vrrpAssoIpAddrTable' and replaced by 'vrrpAssoIpAddr'.
- Removed references to 'vrrpAssoIpAddrIndex' in samples.
- Added new object 'vrrpOperProtocol' to 'VrrpOperEntry'.
- MAX-ACCESS for the 'vrrpOperVrId' object changed to 'not-accessible', as per RFC1902 (auxiliary objects).
- SYNTAX for 'vrrpOperVirtualRouterUpTime' changed to 'TimeStamp'.
- Added importation of 'TruthValue' and 'TimeStamp' to accommodate changes listed above. Deleted importation of 'TimeTicks'.
- Changed MAX-ACCESS to 'accessible-for-notify' for 'vrrpTrapPacketSrc' and 'vrrpTrapConfigErrorType' objects.
- In the sample tables, the "if" values were incorrect for the sample tables for "IP B" (they used to read "II").
- MAX-ACCESS for 'vrrpOperAuthType' and 'vrrpOperAuthKey' changed to 'read-only', since these objects are defined on a per-interface basis.
- Overall review and editing of Section 5.0 (References) with deletion of references not used in this document. Also, added reference '9'.

Concerns:
---------
- Keep the 'vrrpTrapAuthFailure' trap?
- Remove 'vrrpTrapPacketSrc' and 'vrrpTrapConfigErrorType' from compliances? Only used for 'vrrpTrapAuthFailure' trap.


- Overall review and editing of document text and object descriptions from feedback received at December 1997 IETF meeting. More specifically the changes made along these lines include the following:

  -- Cleaned up Section 5.0 ("References") and validated.
  -- Modified diagram for VRRP Scenario #2 (Section 2.4.2).
  -- VRRP Overview (Section 2.0) rewritten to be more in-line with latest revision (05) of VRRP RFC Draft.
  -- Description of "VrId" Textual convention rewritten.

  - MAX-ACCESS of vrrpTrapCntl object changed to "read-write".
  - Changed SYNTAX of "vrrpOperVirtualMacAddr" object from "PhyAddress" to "MacAddress".
  - Changed the syntax of the "vrrpOperPriority" object so upper
value can be 255.
- Change MAX-ACCESS for "vrrpOperAuthType" object from read-only to read-create.
- Added three new objects to "vrrpRouterStatsTable": "vrrpStatsAddressListErrors", "vrrpStatsUnknownauthType" and "vrrpStatsAuthTypeErrors".
- Removed "vrrpStatsUnknownPacketType" object, as it was redundant (with "vrrpStatsInvalidPacketType").
- Removed the "vrrpOperIfIndex" object to use "ifIndex" (which is actually equivalent) instead.

* 11/19/97 (vft-ietf-vrrp-mib-00.txt) - Initial Draft