

Chapter 8

Standard SNMP Traps

This chapter summarizes the standard SNMP traps supported by the JUNOS software. For scalability reasons, the MPLS traps are generated by the ingress router only. For information on disabling the generation of MPLS traps, see the *JUNOS Internet Software Configuration Guide: MPLS Applications*.

The JUNOS software supports the following standard SNMP traps:

Standard SNMP Version 1 Traps on page 79

Standard SNMP Version 2 Traps on page 85

Standard SNMP Version 1 Traps

Table 9 provides an overview of the standard traps for SNMPv1. The traps are organized first by trap category and then by trap name and include their enterprise ID, generic trap number, and specific trap number. The system logging severity levels are listed for those traps that have them with their corresponding system log tag. For traps that do not have corresponding system logging severity levels, the cell in the table is marked with an em-dash (—).

For more information on system log messages, see the *JUNOS Internet Software System Log Messages Reference*. For more information about configuring system logging, see the *JUNOS Internet Software Configuration Guide: Getting Started*.

Table 9: Standard Supported SNMP Version 1 Traps

Trap Category	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	Syslog Tag
authentication	authenticationFailure	1.3.6.1.4.1.2636	4	0	notice	SNMPD_TRAP_GEN_FAILURE
link	linkDown	1.3.6.1.4.1.2636	2	0	info	SNMP_TRAP_LINK_DOWN
link	linkUp	1.3.6.1.4.1.2636	3	0	warning	SNMP_TRAP_LINK_UP
remote-operations	pingProbeFailed	1.3.6.1.2.1.80.0	6	1	info	SNMP_TRAP_PING_PROBE_FAILED
remote-operations	pingTestFailed	1.3.6.1.2.1.80.0	6	2	info	SNMP_TRAP_PING_TEST_FAILED
remote-operations	pingTestCompleted	1.3.6.1.2.1.80.0	6	3	info	SNMP_TRAP_PING_TEST_COMPLETED
remote-operations	traceRoutePathChange	1.3.6.1.2.1.81.0	6	1	info	SNMP_TRAP_TRACE_ROUTE_PATH_CHANGE
remote-operations	traceRouteTestFailed	1.3.6.1.2.1.81.0	6	2	info	SNMP_TRAP_TRACE_ROUTE_TEST_FAILED
remote-operations	traceRouteTestCompleted	1.3.6.1.2.1.81.0	6	3	info	SNMP_TRAP_TRACE_ROUTE_TEST_COMPLETED
rmon-alarm	fallingAlarm	1.3.6.1.2.1.16	6	2	—	—
rmon-alarm	risingAlarm	1.3.6.1.2.1.16	6	1	—	—
routing	bgpEstablished	1.3.6.1.2.1.15.7	6	1	—	—
routing	bgpBackwardTransition	1.3.6.1.2.1.15.7	6	2	—	—
routing	ospfVirtIfStateChange	1.3.6.1.2.1.14.16.2	6	1	—	—
routing	ospfNbrStateChange	1.3.6.1.2.1.14.16.2	6	2	—	—
routing	ospfVirtNbrStateChange	1.3.6.1.2.1.14.16.2	6	3	—	—
routing	ospflfConfigError	1.3.6.1.2.1.14.16.2	6	4	—	—
routing	ospfVirtIfConfigError	1.3.6.1.2.1.14.16.2	6	5	—	—
routing	ospfVirtIfConfigError	1.3.6.1.2.1.14.16.2	6	6	—	—
routing	ospflfAuthFailure	1.3.6.1.2.1.14.16.2	6	7	—	—
routing	ospflfRxBadPacket	1.3.6.1.2.1.14.16.2	6	8	—	—
routing	ospfVirtIfRxBadPacket	1.3.6.1.2.1.14.16.2	6	9	—	—
routing	ospfTxRetransmit	1.3.6.1.2.1.14.16.2	6	10	—	—
routing	ospfVirtIfTxRetransmit	1.3.6.1.2.1.14.16.2	6	11	—	—
routing	ospfOriginateLsa	1.3.6.1.2.1.14.16.2	6	12	—	—
routing	ospfMaxAgeLsa	1.3.6.1.2.1.14.16.2	6	13	—	—
routing	ospfLsdbOverflow	1.3.6.1.2.1.14.16.2	6	14	—	—
routing	ospfLsdbApproachingOverflow	1.3.6.1.2.1.14.16.2	6	15	—	—
routing	ospflfStateChange	1.3.6.1.2.1.14.16.2	6	16	—	—
startup	coldStart	1.3.6.1.4.1.2636	0	0	critical	SNMPD_TRAP_COLD_START
startup	warmStart	1.3.6.1.4.1.2636	1	0	error	SNMPD_TRAP_WARM_START
vrrp	vrrpTrapNewMaster	1.3.6.1.2.1.68	6	1	warning	VRRPD_NEWMASTER_TRAP
vrrp	vrrpTrapAuthFailure	1.3.6.1.2.1.68	6	2	warning	VRRPD_AUTH_FAILURE_TRAP

SNMPv1 also supports the following standard traps:

- SNMP Version 1 Standard Traps on page 81
- SNMP Version 1 Ping Traps MIB on page 82
- SNMP Version 1 Traceroute Traps MIB on page 83
- SNMP Version 1 VRRP Traps MIB on page 84

SNMP Version 1 Standard Traps

The JUNOS software supports the standard SNMP version 1 traps, which are taken from RFC 1215, *Convention for defining traps for use with the SNMP*.

```

coldStart          TRAP-TYPE
ENTERPRISE        snmp
DESCRIPTION
    "A coldStart trap signifies that the sending protocol entity is reinitializing itself such
    that the agent's configuration or the protocol entity implementation may be altered."
 ::= 0

warmStart          TRAP-TYPE
ENTERPRISE        snmp
DESCRIPTION
    "A warmStart trap signifies that the sending protocol entity is reinitializing itself such
    that neither the agent configuration nor the protocol entity implementation is altered."
 ::= 1

linkDown           TRAP-TYPE
ENTERPRISE        snmp
VARIABLES         { ifIndex }
DESCRIPTION
    "A linkDown trap signifies that the sending protocol entity recognizes a failure in one of
    the communication links represented in the agent's configuration."
 ::= 2

linkUp             TRAP-TYPE
ENTERPRISE        snmp
VARIABLES         { ifIndex }
DESCRIPTION
    "A linkUp trap signifies that the sending protocol entity recognizes that one of the
    communication links represented in the agent's configuration has come up."
 ::= 3

authenticationFailure TRAP-TYPE
ENTERPRISE        snmp
DESCRIPTION
    "An authenticationFailure trap signifies that the sending protocol entity is the addressee
    of a protocol message that is not properly authenticated. While implementations of the
    SNMP must be capable of generating this trap, they must also be capable of suppressing the
    emission of such traps via an implementation- specific mechanism."
 ::= 4

```

```

egpNeighborLoss TRAP-TYPE
  ENTERPRISE snmp
  VARIABLES { egpNeighAddr }
  DESCRIPTION
    "An egpNeighborLoss trap signifies that an EGP neighbor for whom the sending protocol entity
    was an EGP peer has been marked down and the peer relationship no longer obtains."
  ::= 5

```

SNMP Version 1 Ping Traps MIB

The JUNOS software supports the SNMP traps from RFC 2925, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*, converted to SNMPv1 format:

-definition of ping MIB traps

```

SNMP Version 1 Traceroute Traps MIB
pingProbeFailed TRAP-TYPE
  ENTERPRISE pingMIB
  VARIABLES {
    pingCtlTargetAddressType, pingCtlTargetAddress,
    pingResultsOperStatus, pingResultsIpTargetAddressType,
    pingResultsIpTargetAddress, pingResultsMinRtt,
    pingResultsMaxRtt, pingResultsAverageRtt,
    pingResultsProbeResponses, pingResultsSentProbes,
    pingResultsRttSumOfSquares, pingResultsLastGoodProbe
  }
  STATUS mandatory
  DESCRIPTION
    "Generated when a probe failure is detected when the
    corresponding pingCtlTrapGeneration object is set to
    probeFailure(0) subject to the value of
    pingCtlTrapProbeFailureFilter. The object
    pingCtlTrapProbeFailureFilter can be used to specify the
    number of successive probe failures that are required
    before this notification can be generated."
  ::= 1

pingTestFailed TRAP-TYPE
  ENTERPRISE pingMIB
  VARIABLES {
    pingCtlTargetAddressType, pingCtlTargetAddress,
    pingResultsOperStatus, pingResultsIpTargetAddressType,
    pingResultsIpTargetAddress, pingResultsMinRtt,
    pingResultsMaxRtt, pingResultsAverageRtt,
    pingResultsProbeResponses, pingResultsSentProbes,
    pingResultsRttSumOfSquares, pingResultsLastGoodProbe
  }
  STATUS mandatory
  DESCRIPTION
    "Generated when a ping test is determined to have failed
    when the corresponding pingCtlTrapGeneration object is
    set to testFailure(1). In this instance
    pingCtlTrapTestFailureFilter should specify the number of
    probes in a test required to have failed in order to
    consider the test as failed."
  ::= 2

```

```

pingTestCompleted          TRAP-TYPE
ENTERPRISE pingMIB
VARIABLES {
    pingCtlTargetAddressType, pingCtlTargetAddress,
    pingResultsOperStatus, pingResultsIpTargetAddressType,
    pingResultsIpTargetAddress, pingResultsMinRtt,
    pingResultsMaxRtt, pingResultsAverageRtt,
    pingResultsProbeResponses, pingResultsSentProbes,
    pingResultsRttSumOfSquares, pingResultsLastGoodProbe
}
STATUS                      mandatory
DESCRIPTION
    "Generated at the completion of a ping test when the
    corresponding pingCtlTrapGeneration object is set to
    testCompletion(4)."
 ::= 3

```

SNMP Version 1 Traceroute Traps MIB

The JUNOS software supports the SNMP traps from RFC 2925, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*, converted to SNMPv1 format:

-definition of traceroute traps

```

traceRoutePathChange      TRAP-TYPE
ENTERPRISE                traceRouteMIB
VARIABLES {
    traceRouteCtlTargetAddressType,
    traceRouteCtlTargetAddress,
    traceRouteResultsIpTgtAddrType,
    traceRouteResultsIpTgtAddr
}
STATUS                    mandatory
DESCRIPTION
    "The path to a target has changed."
 ::= 1

```

```

traceRouteTestFailed      TRAP-TYPE
ENTERPRISE                traceRouteMIB
VARIABLES {
    traceRouteCtlTargetAddressType,
    traceRouteCtlTargetAddress,
    traceRouteResultsIpTgtAddrType,
    traceRouteResultsIpTgtAddr
}
STATUS                    mandatory
DESCRIPTION
    "Could not determine the path to a target."
 ::= 2

```

```

traceRouteTestCompleted TRAP-TYPE
  ENTERPRISE traceRouteMIB
  VARIABLES {
    traceRouteCtlTargetAddressType,
    traceRouteCtlTargetAddress,
    traceRouteResultsIpTgtAddrType,
    traceRouteResultsIpTgtAddr
  }
  STATUS mandatory
  DESCRIPTION
    "The path to a target has just been determined."
 ::= 3

```

SNMP Version 1 VRRP Traps MIB

The JUNOS software supports the SNMP traps from RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol*, converted to SNMPv1 format:

-definition of vrrp traps

```

vrrpTrapNewMaster TRAP-TYPE
  ENTERPRISE vrrpMIB
  VARIABLES {
    vrrpOperMasterIpAddr
  }
  STATUS mandatory
  DESCRIPTION
    "The newMaster trap indicates that the sending agent
    has transitioned to 'Master' state."
 ::= 1

vrrpTrapAuthFailure TRAP-TYPE
  ENTERPRISE vrrpMIB
  VARIABLES {
    vrrpTrapPacketSrc
    vrrpTrapAuthErrorType
  }
  STATUS mandatory
  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."
 ::= 2

```

Standard SNMP Version 2 Traps

Table 10 provides an overview of the standard SNMPv2 traps supported by the JUNOS software. The traps are organized first by trap category and then by trap name and include their snmpTrapOID. The system logging severity levels are listed for those traps that have them with their corresponding system log tag. For traps that do not have corresponding system logging severity levels, the cell in the table is marked with an em-dash (—).

For more information about system log messages, see the *JUNOS Internet Software System Log Messages Reference*. For more information about configuring system logging, see the *JUNOS Internet Software Configuration Guide: Getting Started*.

Table 10: Standard Supported SNMP Version 2 Traps

Trap Category	Trap Name	snmpTrapOID	System Logging Severity Level	Syslog Tag
authentication	authenticationFailure	1.3.6.1.6.3.1.1.5.5	notice	SNMPD_TRAP_GEN_FAILURE
link	linkDown	1.3.6.1.6.3.1.1.5.4	info	SNMP_TRAP_LINK_DOWN
link	linkUp	1.3.6.1.6.3.1.1.5.4	warning	SNMP_TRAP_LINK_UP
remote-operations	pingProbeFailed	1.3.6.1.2.1.80.0.1	info	SNMP_TRAP_PING_PROBE_FAILED
remote-operations	pingTestFailed	1.3.6.1.2.1.80.0.2	info	SNMP_TRAP_PING_TEST_FAILED
remote-operations	pingTestCompleted	1.3.6.1.2.1.80.0.3	info	SNMP_TRAP_PING_TEST_COMPLETED
remote-operations	traceRoutePathChange	1.3.6.1.2.1.81.0.1	info	SNMP_TRAP_TRACE_ROUTE_PATH_CHANGE
remote-operations	traceRouteTestFailed	1.3.6.1.2.1.81.0.2	info	SNMP_TRAP_TRACE_ROUTE_TEST_FAILED
remote-operations	traceRouteTestCompleted	1.3.6.1.2.1.81.0.3	info	SNMP_TRAP_TRACE_ROUTE_TEST_COMPLETED
rmon-alarm	fallingAlarm	1.3.6.1.2.1.15.7.1	—	—
rmon-alarm	risingAlarm	1.3.6.1.2.1.15.7.2	—	—
routing	bgpEstablished	1.3.6.1.2.1.15.7.1	—	—
routing	bgpBackwardTransition	1.3.6.1.2.1.15.7.2	—	—
routing	ospfVirtIfStateChange	1.3.6.1.2.1.14.16.2.1	—	—
routing	ospfNbrStateChange	1.3.6.1.2.1.14.16.2.2	—	—
routing	ospfVirtNbrStateChange	1.3.6.1.2.1.14.16.2.3	—	—
routing	ospfIfConfigError	1.3.6.1.2.1.14.16.2.4	—	—
routing	ospfVirtIfConfigError	1.3.6.1.2.1.14.16.2.5	—	—
routing	ospfVirtIfConfigError	1.3.6.1.2.1.14.16.2.6	—	—
routing	ospfIfAuthFailure	1.3.6.1.2.1.14.16.2.7	—	—
routing	ospfIfRxBadPacket	1.3.6.1.2.1.14.16.2.8	—	—
routing	ospfVirtIfRxBadPacket	1.3.6.1.2.1.14.16.2.9	—	—
routing	ospfTxRetransmit	1.3.6.1.2.1.14.16.2.10	—	—
routing	ospfVirtIfTxRetransmit	1.3.6.1.2.1.14.16.2.11	—	—
routing	ospfOriginateLsa	1.3.6.1.2.1.14.16.2.12	—	—
routing	ospfMaxAgeLsa	1.3.6.1.2.1.14.16.2.13	—	—
routing	ospfLsdbOverflow	1.3.6.1.2.1.14.16.2.14	—	—

Trap Category	Trap Name	snmpTrapOID	System Logging Severity Level	Syslog Tag
routing	ospfLsdbApproachingOverflow	1.3.6.1.2.1.14.16.2.15	—	—
routing	ospfIfStateChange	1.3.6.1.2.1.14.16.2.16	—	—
startup	coldStart	1.3.6.1.6.3.1.1.5.1	critical	SNMPD_TRAP_COLD_START
startup	warmStart	1.3.6.1.6.3.1.1.5.2	error	SNMPD_TRAP_WARM_START
vrrp	vrrpTrapNewMaster	1.3.6.1.2.1.68.0.1	warning	VRRPD_NEWMASTER_TRAP
vrrp	vrrpTrapAuthFailure	1.3.6.1.2.1.68.0.2	warning	VRRPD_AUTH_FAILURE_TRAP

The JUNOS software supports the following standard SNMP version 2 traps:

SNMP Version 2 Standard Traps on page 86

SNMP Version 2 BGP Traps MIB on page 88

SNMP Version 2 OSPF Traps MIB on page 88

SNMP Version 2 Ping Traps MIB on page 92

SNMP Version 2 Traceroute Traps MIB on page 94

SNMP Version 2 VRRP Traps MIB on page 95

SNMP Version 2 Standard Traps

The JUNOS software supports the standard SNMP version traps, which are taken from RFCs 1907, *Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)*, and RFC 2863, *The Interfaces Group MIB*:

```

coldStart      NOTIFICATION-TYPE
STATUS        current
DESCRIPTION
  "A coldStart trap signifies that the SNMPv2 entity, acting
  in an agent role, is reinitializing itself and that its
  configuration may have been altered."
 ::= { snmpTraps 1 }

```

```

warmStart      NOTIFICATION-TYPE
STATUS        current
DESCRIPTION
  "A warmStart trap signifies that the SNMPv2 entity, acting
  in an agent role, is reinitializing itself such that its
  configuration is unaltered."
 ::= { snmpTraps 2 }

```

```

linkDown          NOTIFICATION-TYPE
OBJECTS {
  ifIndex
  ifAdminStatus
  ifOperStatus
}
STATUS            current
DESCRIPTION
  "A linkDown trap signifies that the SNMP entity, acting in
  an agent role, has detected that the ifOperStatus object for
  one of its communication links is about to enter the down
  state from some other state (but not from the notPresent
  state). This other state is indicated by the included value
  of ifOperStatus."
::= { snmpTraps 3 }

linkUp           NOTIFICATION-TYPE
OBJECTS {
  ifIndex
  ifAdminStatus
  ifOperStatus
}
STATUS            current
DESCRIPTION
  "A linkUp trap signifies that the SNMP entity, acting in an
  agent role, has detected that the ifOperStatus object for
  one of its communication links left the down state and
  transitioned into some other state (but not into the
  notPresent state). This other state is indicated by the
  included value of ifOperStatus."
::= { snmpTraps 4 }

authenticationFailureNOTIFICATION-TYPE
STATUS            current
DESCRIPTION
  "An authenticationFailure trap signifies that the SNMPv2
  entity, acting in an agent role, has received a protocol
  message that is not properly authenticated. While all
  implementations of the SNMPv2 must be capable of generating
  this trap, the snmpEnableAuthenTraps object indicates
  whether this trap will be generated."
::= { snmpTraps 5 }

```

SNMP Version 2 BGP Traps MIB

The JUNOS software supports the BGP standard SNMP version 2 traps. The following descriptions are taken from RFC 1657, *Definitions of Managed Objects for the Fourth Version of the Border Gateway Protocol (BGP-4) using SMIV2* :

```

bgpEstablished      NOTIFICATION-TYPE
OBJECTS {
    bgpPeerLastError
    bgpPeerState
}
STATUS              current
DESCRIPTION
    "The BGP Established event is generated when the BGP FSM enters the ESTABLISHED state."
::= { bgpTraps 1 }
bgpBackwardTransitionNOTIFICATION-TYPE
OBJECTS {
    bgpPeerLastError
    bgpPeerState
}
STATUS              current
DESCRIPTION
    "The BGPBackwardTransition Event is generated when the BGP FSM moves from a
    higher numbered state to a lower numbered state."
::= { bgpTraps 2 }

```

SNMP Version 2 OSPF Traps MIB

The JUNOS software supports the OSPF SNMP version 2 traps. The following descriptions are taken from RFC 1850, *OSPF Version 2 Management Information Base* :

```

ospflfStateChange  NOTIFICATION-TYPE
OBJECTS {
    ospfRouterId, -- The originator of the trap
    ospflfIpAddress,
    ospfAddressLessIf,
    ospflfState } -- The new state
STATUS              current
DESCRIPTION
    "An ospflfStateChange trap signifies that there has been a change in the state of a non-virtual
    OSPF interface. This trap should be generated when the interface state regresses (e.g., goes
    from Dr to Down) or progresses to a terminal state (i.e., Point-to-Point, DR Other, Dr, or
    Backup)."
::= { ospfTraps 16 }

ospfVirtIfStateChange NOTIFICATION-TYPE
OBJECTS              {ospfRouterId, -- The originator of the trap
    ospfVirtIfAreaId,
    ospfVirtIfNeighbor,
    ospfVirtIfState } -- The new state
STATUS              current
DESCRIPTION
    "An ospflfStateChange trap signifies that there has been a change in the state of an OSPF virtual
    interface. This trap should be generated when the interface state regresses (e.g., goes
    from Point-to-Point to Down) or progresses to a terminal state (i.e., Point-to-Point)."
::= { ospfTraps 1 }

```

```

ospfNbrStateChange NOTIFICATION-TYPE
OBJECTS      {ospfRouterId, -- The originator of the trap
              ospfNbrIpAddr,
              ospfNbrAddressLessIndex,
              ospfNbrRtrId,
              ospfNbrState } -- The new state
STATUS      current
DESCRIPTION
  "An ospfNbrStateChange trap signifies that there has been a change in the state of a non-
  virtual OSPF neighbor. This trap should be generated when the neighbor state regresses
  (e.g., goes from Attempt or Full to 1-Way or Down) or progresses to a terminal state (e.g.,
  2-Way or Full). When a neighbor transitions from or to Full on non-broadcast multi-access
  and broadcast networks, the trap should be generated by the designated router. A designated
  router transitioning to Down will be noted by ospfIfStateChange."
::= { ospfTraps 2 }

ospfVirtNbrStateChange NOTIFICATION-TYPE
OBJECTS      { ospfRouterId, -- The originator of the trap
              ospfVirtNbrArea,
              ospfVirtNbrRtrId,
              ospfVirtNbrState } -- The new state
STATUS      current
DESCRIPTION
  "An ospfIfStateChange trap signifies that there has been a change in the state of an OSPF vir-
  tual neighbor. This trap should be generated when the neighbor state regresses (e.g., goes
  from Attempt or Full to 1-Way or Down) or progresses to a terminal state (e.g., Full)."
::= { ospfTraps 3 }

ospfIfConfigError NOTIFICATION-TYPE
OBJECTS      {ospfRouterId, -- The originator of the trap
              ospfIfIpAddress,
              ospfAddressLessIf,
              ospfPacketSrc, -- The source IP address
              ospfConfigErrorType, -- Type of error
              ospfPacketType }
STATUS      current
DESCRIPTION
  "An ospfIfConfigError trap signifies that a packet has been received on a non-virtual in-
  terface from a router whose configuration parameters conflict with this router's confi-
  guration parameters. Note that the event optionMismatch should cause a trap only if it
  prevents an adjacency from forming."
::= { ospfTraps 4 }

ospfVirtIfConfigError NOTIFICATION-TYPE
OBJECTS      {ospfRouterId, -- The originator of the trap
              ospfVirtIfAreaId,
              ospfVirtIfNeighbor,
              ospfConfigErrorType, -- Type of error
              ospfPacketType }
STATUS      current
DESCRIPTION
  "An ospfConfigError trap signifies that a packet has been received on a virtual interface
  from a router whose configuration parameters conflict with this router's configuration
  parameters. Note that the event optionMismatch should cause a trap only if it prevents an ad-
  jacency from forming."
::= { ospfTraps 5 }

```

ospfIfAuthFailure NOTIFICATION-TYPE

OBJECTS {ospfRouterId, -- The originator of the trap
ospfIfIpAddress,
ospfAddressLessIf,
ospfPacketSrc, -- The source IP address
ospfConfigErrorType, -- authTypeMismatch or
-- authFailure
ospfPacketType }

STATUS current

DESCRIPTION

"An ospfIfAuthFailure trap signifies that a packet has been received on a non-virtual interface from a router whose authentication key or authentication type conflicts with this router's authentication key or authentication type."

::= { ospfTraps 6 }

ospfVirtIfAuthFailure NOTIFICATION-TYPE

OBJECTS {ospfRouterId, -- The originator of the trap
ospfVirtIfAreaId,
ospfVirtIfNeighbor,
ospfConfigErrorType, -- authTypeMismatch or
-- authFailure
ospfPacketType }

STATUS current

DESCRIPTION

"An ospfVirtIfAuthFailure trap signifies that a packet has been received on a virtual interface from a router whose authentication key or authentication type conflicts with this router's authentication key or authentication type."

::= { ospfTraps 7 }

ospfIfRxBadPacket NOTIFICATION-TYPE

OBJECTS {ospfRouterId, -- The originator of the trap
ospfIfIpAddress,
ospfAddressLessIf,
ospfPacketSrc, -- The source IP address
ospfPacketType }

STATUS current

DESCRIPTION

"An ospfIfRxBadPacket trap signifies that an OSPF packet has been received on a non-virtual interface that cannot be parsed."

::= { ospfTraps 8 }

ospfVirtIfRxBadPacket NOTIFICATION-TYPE

OBJECTS {ospfRouterId, -- The originator of the trap
ospfVirtIfAreaId,
ospfVirtIfNeighbor,
ospfPacketType }

STATUS current

DESCRIPTION

"An ospfVirtIfRxBadPacket trap signifies that an OSPF packet has been received on a virtual interface that cannot be parsed."

::= { ospfTraps 9 }

```

ospfTxRetransmit NOTIFICATION-TYPE
OBJECTS      {ospfRouterId, -- The originator of the trap
              ospfIfIpAddress,
              ospfAddressLessIf,
              ospfNbrRtrId, -- Destination
              ospfPacketType,
              ospfLsdbType,
              ospfLsdbLsid,
              ospfLsdbRouterId }
STATUS      current
DESCRIPTION
  "An ospfTxRetransmit trap signifies that an OSPF packet has been retransmitted on a non-
  virtual interface. All packets that may be re-transmitted are associated with an LSDB entry.
  The LS type, LS ID, and Router ID are used to identify the LSDB entry."
 ::= { ospfTraps 10 }

ospfVirtIfTxRetransmit NOTIFICATION-TYPE
OBJECTS      {
              ospfRouterId, -- The originator of the trap
              ospfVirtIfAreaId,
              ospfVirtIfNeighbor,
              ospfPacketType,
              ospfLsdbType,
              ospfLsdbLsid,
              ospfLsdbRouterId }
STATUS      current
DESCRIPTION
  "An ospfTxRetransmit trap signifies that an OSPF packet has been retransmitted on a virtual
  interface. All packets that may be retransmitted are associated with an LSDB entry. The LS
  type, LS ID, and Router ID are used to identify the LSDB entry."
 ::= { ospfTraps 11 }

ospfOriginateLsa NOTIFICATION-TYPE
OBJECTS      {
              ospfRouterId, -- The originator of the trap
              ospfLsdbAreaId, -- 0.0.0.0 for AS Externals
              ospfLsdbType,
              ospfLsdbLsid,
              ospfLsdbRouterId }
STATUS      current
DESCRIPTION
  "An ospfOriginateLsa trap signifies that a new LSA has been originated by this router. This
  trap should not be invoked for simple refreshes of LSAs (which happens every 30 minutes), but
  instead will only be invoked when an LSA is (re)originated due to a topology change. Addi-
  tionally, this trap does not include LSAs that are being flushed because they have reached
  MaxAge."
 ::= { ospfTraps 12 }

ospfMaxAgeLsa NOTIFICATION-TYPE
OBJECTS      {ospfRouterId, -- The originator of the trap
              ospfLsdbAreaId, -- 0.0.0.0 for AS Externals
              ospfLsdbType,
              ospfLsdbLsid,
              ospfLsdbRouterId }
STATUS      current
DESCRIPTION
  "An ospfMaxAgeLsa trap signifies that one of the LSAs in the router's link-state database has
  aged to MaxAge."
 ::= { ospfTraps 13 }

```

```

ospfLsdbOverflow NOTIFICATION-TYPE
OBJECTS      {
    ospfRouterId, -- The originator of the trap
    ospfExtLsdbLimit }
STATUS      current
DESCRIPTION
    "An ospfLsdbOverflow trap signifies that the number of LSAs in the router's link-state data-
    base has exceeded ospfExtLsdbLimit."
::= { ospfTraps 14 }

ospfLsdbApproachingOverflow NOTIFICATION-TYPE
OBJECTS      {\
    ospfRouterId, -- The originator of the trap
    ospfExtLsdbLimit }
STATUS      current
DESCRIPTION
    "An ospfLsdbApproachingOverflow trap signifies that the number of LSAs in the router's link-
    state database has exceeded ninety percent of ospfExtLsdbLimit."
::= { ospfTraps 15 }

```

SNMP Version 2 Ping Traps MIB

The following descriptions for the SNMPv2 ping traps are from RFC 2925, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*:

```

pingProbeFailed      NOTIFICATION-TYPE
OBJECTS {
    pingCtlTargetAddressType,
    pingCtlTargetAddress,
    pingResultsOperStatus,
    pingResultsIpTargetAddressType,
    pingResultsIpTargetAddress,
    pingResultsMinRtt,
    pingResultsMaxRtt,
    pingResultsAverageRtt,
    pingResultsProbeResponses,
    pingResultsSentProbes,
    pingResultsRttSumOfSquares,
    pingResultsLastGoodProbe
}
STATUS      current
DESCRIPTION
    "Generated when a probe failure is detected when the
    corresponding pingCtlTrapGeneration object is set to
    probeFailure(0) subject to the value of
    pingCtlTrapProbeFailureFilter. The object
    pingCtlTrapProbeFailureFilter can be used to specify the
    number of successive probe failures that are required
    before this notification can be generated."
::= { pingNotifications 1 }

```

```

pingTestFailed      NOTIFICATION-TYPE
OBJECTS {
    pingCtlTargetAddressType,
    pingCtlTargetAddress,
    pingResultsOperStatus,
    pingResultsIpTargetAddressType,
    pingResultsIpTargetAddress,
    pingResultsMinRtt,
    pingResultsMaxRtt,
    pingResultsAverageRtt,
    pingResultsProbeResponses,
    pingResultsSentProbes,
    pingResultsRttSumOfSquares,
    pingResultsLastGoodProbe
}
STATUS              current
DESCRIPTION
    "Generated when a ping test is determined to have failed
    when the corresponding pingCtlTrapGeneration object is
    set to testFailure(1). In this instance
    pingCtlTrapTestFailureFilter should specify the number of
    probes in a test required to have failed in order to
    consider the test as failed."
 ::= { pingNotifications 2 }

pingTestCompleted  NOTIFICATION-TYPE
OBJECTS {
    pingCtlTargetAddressType,
    pingCtlTargetAddress,
    pingResultsOperStatus,
    pingResultsIpTargetAddressType,
    pingResultsIpTargetAddress,
    pingResultsMinRtt,
    pingResultsMaxRtt,
    pingResultsAverageRtt,
    pingResultsProbeResponses,
    pingResultsSentProbes,
    pingResultsRttSumOfSquares,
    pingResultsLastGoodProbe
}
STATUS              current
DESCRIPTION
    "Generated at the completion of a ping test when the
    corresponding pingCtlTrapGeneration object is set to
    testCompletion(4)."
 ::= { pingNotifications 3 }

```

SNMP Version 2 Traceroute Traps MIB

The following descriptions for the SNMPv2 traceroute traps are from RFC 2925, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*:

```
traceRoutePathChange    NOTIFICATION-TYPE
```

```
  OBJECTS {
    traceRouteCtlTargetAddressType,
    traceRouteCtlTargetAddress,
    traceRouteResultsIpTgtAddrType,
    traceRouteResultsIpTgtAddr
  }
```

```
  STATUS                current
```

```
  DESCRIPTION
```

```
    "The path to a target has changed."
```

```
::= { traceRouteNotifications 1 }
```

```
traceRouteTestFailed    NOTIFICATION-TYPE
```

```
  OBJECTS {
    traceRouteCtlTargetAddressType,
    traceRouteCtlTargetAddress,
    traceRouteResultsIpTgtAddrType,
    traceRouteResultsIpTgtAddr
  }
```

```
  STATUS                current
```

```
  DESCRIPTION
```

```
    "Could not determine the path to a target."
```

```
::= { traceRouteNotifications 2 }
```

```
traceRouteTestCompleted NOTIFICATION-TYPE
```

```
  OBJECTS {
    traceRouteCtlTargetAddressType,
    traceRouteCtlTargetAddress,
    traceRouteResultsIpTgtAddrType,
    traceRouteResultsIpTgtAddr
  }
```

```
  STATUS                current
```

```
  DESCRIPTION
```

```
    "The path to a target has just been determined."
```

```
::= { traceRouteNotifications 3 }
```

SNMP Version 2 VRRP Traps MIB

The following descriptions for the SNMPv2 VRRP traps are from RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol*:

-- vrrp trap definitions

```

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."
 ::= { vrrpOperations 5 }

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."

```

